

ADAM GRZECH

**STEROWANIE RUCHEM
W SIECIACH
TELEINFORMATYCZNYCH**



**OFICyna WYDAWNICZA POLITECHNIKI WROCLAWSKIEJ
WROCLAW 2002**

RECENZENCI

Zdzisław Papier

Józef Woźniak

OPRACOWANIE REDAKCYJNE

Alicja Kordas

KOREKTA

Aleksandra Wawrzynkowska-Dwojak

PROJEKT OKŁADKI

Zofia i Dariusz Godlewscy

© Copyright by Oficyna Wydawnicza Politechniki Wrocławskiej, Wrocław 2002

ISBN 83-7085-637-3

OFICyna WYDAWNICZA POLITECHNIKI WROCLAWSKIEJ

Wybrzeże Wyspiańskiego 27, 50-370 Wrocław

Ark. wyd. 50,75. Ark. druk. 39¼. Papier offset. kl. III, 70 g, B1.
Drukarnia Oficyny Wydawniczej Politechniki Wrocławskiej. Zam. nr 545/2002.

W monografii przedstawiono zakres oraz ogólne cele i metody sterowania ruchem w sieciach teleinformatycznych z komutacją jednostek danych, w których mogą być realizowane różne koncepcje dostarczania jakości usług sieciowych. Ogólne zadania sterowania ruchem zilustrowano przykładowymi rozwiązaniami szczegółowych zadań związanych ze sterowaniem przepływem i zapobieganiem przeciążeniom. Charakterystyka omawianych zadań i metod ich rozwiązywania ilustruje ewolucję zakresu funkcjonalnego zadań sterowania ruchem w sieciach teleinformatycznych oraz ewolucję koncepcji, metod i narzędzi stosowanych do rozwiązywania zadań dostarczania jakości usług sieciowych.

Książka jest adresowana do pracowników naukowych i studentów kierunków związanych z informatyką i teleinformatyką wyższych uczelni technicznych oraz osób zajmujących się i zainteresowanych zagadnieniami jakości usług sieci teleinformatycznych.

**Wydawnictwa Politechniki Wrocławskiej
są do nabycia w następujących księgarniach:
„Politechnika”**

**Wybrzeże Wyspiańskiego 27, 50-370 Wrocław
budynek A-1 PWr., tel. (0-71) 320-25-34,
„Tech”**

**plac Grunwaldzki 13, 50-377 Wrocław
budynek D-1 PWr., tel. (0-71) 320-32-52**

Prowadzimy sprzedaż wysyłkową

ISBN 83-7085-637-3

SPIS RZECZY

WPROWADZENIE	13
1. ARCHITEKTURA I PROTOKOŁY JAKOŚCI USŁUG SIECI TELEINFORMATYCZNYCH	18
1.1. Aplikacje w sieciowych systemach teleinformatycznych.....	19
1.2. Jakość usług sieci teleinformatycznych.....	20
1.2.1. Jakość usług.....	20
1.2.2. Różnicowanie jakości usług	21
1.2.3. System jakości usług	23
1.2.4. Funkcje systemu jakości usług	25
1.3. Architektury warstwowe systemów jakości usług.....	26
1.3.1. Łańcuchy jakości usług.....	26
1.3.2. Konceptcje jakości usług w sieciach.....	28
1.3.3. Sterowanie ruchem w sieciach.....	29
1.4. Podsumowanie.....	30
Literatura	30
2. MIARY JAKOŚCI USŁUG SIECI TELEINFORMATYCZNYCH	35
2.1. Klasy ruchu.....	35
2.2. Miary jakości usług w sieciach komputerowych.....	36
2.2.1. Miary jakości połączeń.....	37
2.2.1.1. Czas ustanowienia połączenia	37
2.2.1.2. Czas rozłączania połączenia	37
2.2.1.3. Prawdopodobieństwo akceptacji połączenia	37
2.2.2. Miary jakości transferu.....	38
2.2.2.1. Stopa błędów	39
2.2.2.2. Współczynnik strat jednostek danych.....	39
2.2.2.3. Współczynnik „wtrącania” jednostek danych	40
2.2.2.4. Opóźnienie transferu jednostek danych.....	41
2.2.2.5. Model opóźnienia przepływów.....	44
2.2.2.6. Zmienność opóźnienia.....	47
2.2.2.7. Asymetria	54
2.2.2.8. Czas oczekiwania na odtwarzanie	54
2.2.2.9. Wierność odtwarzania	54
2.3. Podsumowanie.....	55
Literatura	56

3. ODWZOROWANIE WARTOŚCI PARAMETRÓW JAKOŚCI USŁUG W MODELU WARSTWOWYM	58
3.1. Model łączenia systemów otwartych.....	58
3.1.1. Warstwowość.....	58
3.1.2. Model warstwowy łączenia systemów otwartych ISO/OSI.....	60
3.1.2.1. Warstwa fizyczna	60
3.1.2.2. Warstwa łącza danych	61
3.1.2.3. Warstwa sieciowa	62
3.1.2.4. Warstwa transportowa	63
3.1.2.5. Warstwa sesji.....	64
3.1.2.6. Warstwa prezentacji	64
3.1.2.7. Warstwa aplikacji	64
3.1.3. Algorytmy.....	64
3.2. Hierarchia jakości usług w modelu warstwowym	65
3.3. Parametry jakości usług.....	68
3.3.1. Maksymalne opóźnienie jednostek danych	68
3.3.2. Zmienność opóźnienia jednostek danych	68
3.3.3. Zmienność czasu transmisji jednostek danych	69
3.3.4. Niezawodność.....	69
3.3.5. Przepustowość	70
3.4. Odwzorowanie parametrów jakości usług	70
3.4.1. Odwzorowanie niezawodności	70
3.4.2. Odwzorowania miar jakości usług w procesach segmentacji i desegmentacji ...	71
3.4.2.1. Opóźnienia.....	72
3.4.2.2. Zmienność opóźnienia	73
3.4.2.3. Przepustowość	74
3.4.2.4. Niezawodność.....	75
3.4.3. Odwzorowania miar jakości usług w procesach łączenia jednostek danych	76
3.4.3.1. Opóźnienia.....	76
3.4.3.2. Zmienność opóźnienia	77
3.4.3.3. Przepustowość	78
3.4.3.4. Niezawodność.....	78
3.4.3.5. Kompensowanie opóźnienia.....	79
3.4.4. Odwzorowania miar jakości usług w procesach uzupełniania jednostek danych	81
3.4.4.1. Opóźnienia.....	81
3.4.4.2. Przepustowość	81
3.4.4.3. Niezawodność.....	81
3.4.5. Odwzorowania miar jakości usług w procesach kodowania i kompresji.....	82
3.4.5.1. Opóźnienia.....	83
3.4.5.2. Zmienność opóźnienia	83
3.4.5.3. Przepustowość	84
3.4.5.4. Niezawodność.....	84
3.4.6. Odwzorowania miar jakości usług w procesach wygładzania.....	84
3.4.6.1. Opóźnienia.....	85
3.5. Podsumowanie.....	85
Literatura	86

4. ZADANIA KOMUNIKACJI W SIECIACH O ARCHITEKTURACH WARSTWOWYCH	88
4.1. Zadania komunikacji w architekturze warstwowej.....	88
4.2. Hierarchia zadań komunikacji z architekturach warstwowych.....	88
4.3. Podział funkcjonalny zadania komunikacji	89
4.4. Warstwy systemu komunikacji danych	92
4.5. Redundancja funkcji w systemie komunikacji danych	93
4.6. Obsługa zadań komunikacji w modelu warstwowym	94
4.7. Przykład dekompozycji zadania komunikacji	97
4.7.1. Procedura retransmisji pojedynczych	99
4.7.2. Procedura retransmisji grupowej	100
4.7.3. Procedura retransmisji selektywnej	101
4.8. Podsumowanie.....	103
Literatura	103
5. WYZNACZANIE TRAS I STEROWANIE PRZEPLYWEM.....	105
5.1. Zadania wyznaczania tras i przepływów	106
5.2. Zadanie wyznaczania tras	107
5.3. Zadanie wyznaczania tras optymalnych	108
5.4. Moc sieci	111
5.5. Zadanie sterowania przepływem	113
5.6. Łączne zadanie wyznaczania tras i sterowania przepływem	116
5.7. Sprawiedliwy rozdział zasobów w sieci w zadaniach sterowania przepływem.....	126
5.8. Maksymalno-minimalne sterowanie przepływem	129
5.9. Implementacja procedur sterowania przepływami	135
5.9.1. Mechanizm okna.....	135
5.9.2. Wyznaczanie rozmiaru okna.....	137
5.9.2.1. Sterowanie przepływem w procesie komunikacyjnym.....	138
5.9.2.2. Ograniczenia sterowania przepływem z wykorzystaniem mechanizmu okna	144
5.9.2.3. Sterowanie przepływem z wykorzystaniem hierarchii mechanizmów okna	151
5.10. Podsumowanie.....	154
Literatura	155
6. STEROWANIE PRZEPLYWEM.....	158
6.1. Alokacja zasobów w sieciach teleinformatycznych	158
6.2. Zadania sterowania przepływem	160
6.2.1. Degradacja przepustowości	160
6.2.2. niesprawiedliwy podział zasobów	170
6.2.3. Zakleszczenia.....	173
6.3. Funkcje sterowania przepływem	186
6.3.1. Architektura systemów sterowania przepływem	188
6.4. Poziomy sterowania przepływem	191
6.4.1. Sterowanie przepływem na poziomie łącza danych	191
6.4.1.1. Cel sterowania przepływem na poziomie łącza danych.....	191
6.4.1.2. Klasyfikacja schematów sterowania przepływem na poziomie łącza danych.....	192

6.4.1.3. Sterowanie przepływem dla klas jednostek danych wyróżnianych ze względu na wyjściowe kanały transmisyjne.....	194
6.4.1.3.1. Podział całkowity.....	195
6.4.1.3.2. Podział równomierny.....	196
6.4.1.3.3. Podział z maksymalną długością kolejki.....	196
6.4.1.3.4. Podział z minimalną długością kolejki.....	197
6.4.1.3.5. Podział z minimalną i maksymalną długością kolejki.....	198
6.4.1.3.6. Podział pojemności pamięci buforowych w węzłach wyjściowych ruchu.....	199
6.4.1.4. Sterowanie przepływem dla klas ruchu jednostek danych wyróżnianych ze względu na czas przebywania w sieci.....	199
6.4.2. Sterowanie przepływem w połączeniach wirtualnych.....	206
6.4.2.1. Sieci datagramowe i sieci połączeń wirtualnych.....	206
6.4.2.2. Krokowe sterowanie przepływem w połączeniu wirtualnym.....	210
6.4.2.3. Przykładowe implementacje metod sterowania przepływem w połączeniach wirtualnych.....	213
6.4.2.3.1. Sterowanie przepływem w połączeniach wirtualnych sieci TYMNET.....	213
6.4.2.3.2. Sterowanie przepływem w połączeniach wirtualnych sieci TRANSPAC.....	214
6.4.2.3.3. Sterowanie przepływem w połączeniach wirtualnych sieci GMDNET.....	216
6.4.2.3.4. Sterowanie przepływem w połączeniach wirtualnych sieci CODEX.....	218
6.4.3. Sterowanie przepływem na poziomie trasy.....	219
6.4.3.1. Zapobieganie przeciążeniom na trasie.....	219
6.4.3.2. Modele kolejkowe.....	220
6.4.3.3. Sterowanie przepływem na trasie.....	225
6.4.3.4. Lokalne sterowanie przepływem.....	230
6.4.3.5. Cel sterowania przepływem na trasie.....	232
6.4.3.6. Przykładowe implementacje metod sterowania przepływem na poziomie trasy.....	235
6.4.3.6.1. Sterowanie przepływem na poziomie trasy w sieci ARPANET.....	236
6.4.3.6.2. Sterowanie przepływem na poziomie trasy w sieci GMDNET.....	238
6.4.3.6.3. Sterowanie przepływem na poziomie trasy w sieci DATAPAC.....	241
6.4.3.6.4. Sterowanie przepływem na poziomie trasy w architekturze SNA.....	244
6.4.3.6.5. Sterowanie przepływem na poziomie trasy w sieci X.25.....	247
6.4.3.7. Metoda okna w zadaniach sterowania przepływem na poziomie trasy.....	252
6.4.4. Sterowanie przepływem na poziomie dostępu do sieci.....	253
6.4.4.1. Cel sterowania przepływem na poziomie dostępu do sieci.....	253
6.4.4.2. Metoda izarytmiczna.....	255

6.4.4.3. Metoda bufora wejściowego.....	259
6.4.4.4. Implementacja metody bufora wejściowego	260
6.4.4.5. Metoda z pakietem dławiącym	264
6.4.4.6. Całościowe i selektywne sterowanie przepływem.....	266
6.4.5. Sterowanie przepływem na poziomie transportowym	267
6.4.5.1. Cel sterowanie przepływem na poziomie transportowym.....	267
6.4.5.2. Przykładowe implementacje sterowania przepływem na poziomie transportowym	269
6.4.5.2.1. Procedura sterowania przepływem na poziomie transportowym sieci ARPANET	269
6.4.5.2.2. Procedura sterowania przepływem na poziomie sieci CYCLADES	270
6.4.5.2.3. Procedura sterowania przepływem na poziomie transportowym w architekturze ISO/OSI	270
6.4.5.2.4. Procedura sterowania przepływem na poziomie transportowym w architekturze TCP/IP	271
6.4.5.2.5. Procedura sterowania przepływem na poziomie transportowym w architekturze SNA	272
6.4.5.2.6. Procedura sterowania przepływem na poziomie transportowym w architekturze APPN.....	273
6.4.6. Podsumowanie.....	275
6.4.6.1. Hierarchiczna struktura procedur sterowania przepływem w sieciach.....	275
6.4.6.2. Integracja i optymalizacja procedur sterowania przepływem.....	276
6.4.6.3. Sterowanie przepływem w sieciach z integracją trybów komutacji	277
6.4.6.4. Sterowanie przepływem w sieciach z integracją ruchu	277
6.4.6.5. Sterowanie przepływem i zarządzanie retransmisjami	278
6.4.6.6. Sterowanie przepływem i zadania wyznaczania tras	280
6.4.6.7. Sterowanie przepływem w systemach i sieciach rozgłoszeniowych ...	282
6.4.6.8. Sterowanie przepływem w systemach i sieciach heterogenicznych	282
Literatura	285
7. PRZECIWDZIAŁANIE PRZECIĄŻENIOM.....	287
7.1. Sterowanie prewencyjne i reakcyjne	287
7.1.1. Przeciwdziałanie przeciążeniom w sieci ATM.....	289
7.1.1.1. Zadania przeciwdziałania przeciążeniom w sieci ATM	289
7.1.1.2. Usługi, parametry ruchowe źródeł i parametry jakości usług w sieci ATM	290
7.1.1.3. Metody prewencyjnego i reakcyjnego przeciwdziałania przeciążeniom w sieci ATM	291
7.1.2. Przeciwdziałanie przeciążeniom w sieci FR.....	292
7.1.3. Przeciwdziałanie przeciążeniom w sieci TCP/IP.....	293
7.1.3.1. Przeciwdziałanie przeciążeniom w sieciach TCP z usługami bez gwarancji jakości	295
7.1.3.2. Koncepcje przeciwdziałania przeciążeniom w sieciach TCP/IP	299
7.1.3.3. Aktywne zarządzanie kolejkami.....	303
7.1.3.3.1. Mechanizm wczesnej losowej detekcji przeciążenia RED..	304
7.1.3.3.2. Mechanizm losowego wykładniczego znakowania REM...	309

7.2. Metody przeciwdziałania przeciążeniom	313
7.2.1. Planowanie i udostępnianie zasobów	314
7.2.2. Sterowanie przyjmowaniem wywołań	315
7.2.2.1. Zadania sterowania przyjmowaniem wywołań	315
7.2.2.2. Superpozycja strumieni zgłoszeń	315
7.2.2.3. Alokacja pojemności	316
7.2.2.4. Algorytmy przyjmowania wywołań	319
7.2.2.4.1. Algorytm GA	319
7.2.2.4.2. Algorytm FBR	322
7.2.2.4.3. Algorytm EC	328
7.2.2.4.4. Algorytm FA	334
7.2.2.4.5. Algorytm NA	339
7.2.2.4.6. Algorytm HTA	345
7.2.3. Kształtowanie ruchu	347
7.2.4. Utrzymywanie ruchu	348
7.2.4.1. Reglamentacja zasobów i zapotrzebowania na zasoby	348
7.2.4.2. Reglamentacja zasobów systemu	351
7.2.4.2.1. Algorytm ciekącego wiadra	351
7.2.4.2.2. Analiza działania algorytmu LB	358
7.2.4.2.3. Proces Bernoulliego zmodulowany procesem Markowa	359
7.2.4.2.4. Przydzielanie znaczników zgodnie z algorytmem LB(c)	367
7.2.4.3. Reglamentacja zapotrzebowania na zasoby systemu	371
7.2.4.3.1. Technika JW	372
7.2.4.3.2. Technika TJW	372
7.2.4.3.3. Technika MW	373
7.2.4.3.4. Technika EWMA	373
7.2.5. Odrzucanie selektywne	374
7.2.5.1. Mechanizm wypychania	374
7.2.5.2. Mechanizm progowy	375
7.3. Podsumowanie	376
Literatura	377
8. KONCEPCJE RÓŻNICOWANIA JAKOŚCI USŁUG SIECI TELEINFORMATYCZNYCH	381
8.1. Konceptje architektur jakości usług w systemach teleinformatycznych	382
8.1.1. Konceptja usług bez gwarancji jakości	383
8.1.2. Konceptja usług zintegrowanych	385
8.1.3. Konceptja usług zróżnicowanych	387
8.1.4. Konceptja względnego różnicowania jakości usług	390
8.1.5. Jakość usług i zarządzanie ruchem w sieci	391
8.2. Optymalizacja wykorzystania zasobów	394
8.3. Wyznaczanie tras i sterowanie przepływami	395
8.4. Hierarchia zarządzania przepływami w sieciach	397
8.5. Podsumowanie	399
Literatura	401
9. KONCEPCJA USŁUG ZINTEGROWANYCH	404
9.1. Podstawowe założenia	404
9.2. Model funkcjonalny obsługi wywołań	405

9.2.1. Charakterystyki wywołań	407
9.2.2. Sterowanie dostępem wywołań	408
9.2.3. Identyfikacja jednostek danych przepływu	410
9.2.4. Szeregowanie jednostek danych przepływów	410
9.3. Modele usług w koncepcji usług zintegrowanych	410
9.3.1. Model gwarantowanej usługi	411
9.3.1.1. Charakterystyka gwarancji opóźnienia	411
9.3.1.2. Sterowanie opóźnieniem w modelu gwarantowanej usługi	412
9.3.1.3. Ilość ruchu i opóźnienia w obsłudze przepływu	414
9.3.1.4. Formowanie przepływu w modelu gwarantowanej usługi	418
9.3.2. Model sterowanego obciążenia	420
9.3.2.1. Jakość usług	420
9.3.2.2. Opisowe charakterystyki jakości usług	421
9.3.2.3. Procedura obsługi wywołania	422
9.4. Rezerwacja zasobów	422
9.4.1. Właściwości protokołu rezerwacji zasobów	423
9.4.2. Ogólne zasady działania protokołu rezerwacji zasobów	424
9.4.3. Modele rezerwacji zasobów	426
9.4.3.1. Styl rezerwacji sumarycznej	428
9.4.3.2. Styl rezerwacji indywidualnej	430
9.4.3.3. Styl rezerwacji hybrydowej	432
9.5. Identyfikacja przepływu	434
9.5.1. Podstawowe wymagania w odniesieniu do mechanizmu identyfikacji przepływu	435
9.5.2. Technika funkcji skrótu	435
9.5.3. Funkcja skrótu w zadaniu identyfikacji jednostek danych przepływu	437
9.5.4. Suma kontrolna jako funkcja skrótu	439
9.5.5. Podwójna funkcja skrótu	440
9.6. Szeregowanie jednostek danych przepływów	441
9.6.1. Podstawowe wymagania dotyczące mechanizmu szeregowania jednostek danych	443
9.6.1.1. Separacja przepływów i współdzielenie zasobów	443
9.6.1.2. Ograniczenia opóźnień	444
9.6.1.3. Alokacja zasobów	444
9.6.1.4. Podstawowe zasady szeregowania jednostek danych przepływów	446
9.6.1.4.1. Szeregowanie zachowujące pracę i niezachowujące pracy	446
9.6.1.4.2. Priorytetowanie	447
9.6.1.5. Rodzaje algorytmów szeregowania jednostek danych	447
9.6.1.5.1. Algorytmy sprawiedliwego szeregowania	448
9.6.1.5.2. Algorytmy szeregowania z nieprzekraczalnym czasem obsługi	448
9.6.1.5.3. Algorytmy szeregowania warunkowego	449
9.7. Algorytmy ważonego sprawiedliwego szeregowania	450
9.7.1. Model przepływowy	450
9.7.2. Sprawiedliwy rozdział zasobów	452

9.7.2.1. Dyscyplina szeregowania z uogólnionym podziałem czasu procesora	453
9.7.2.2. Dyscyplina szeregowania z nieprzekraczalnym czasem obsługi	453
9.7.3. Miary jakości dyscyplin sprawiedliwego szeregowania	454
9.7.3.1. Sprawiedliwość dyscypliny szeregowania	454
9.7.3.2. Opóźnienie wnoszone przez dyscyplinę szeregowania	455
9.7.4. Dyscyplina szeregowania z uogólnionym podziałem czasu procesora	457
9.7.5. Emulacja dyscypliny szeregowania z uogólnionym podziałem czasu procesora	470
9.7.6. Porównanie działania dyscyplin szeregowania GPS i PGPS	475
9.7.7. Uproszczone wersje ważonych sprawiedliwych algorytmów szeregowania	480
9.7.7.1. Algorytm ważonego sprawiedliwego szeregowania dla najgorszego przypadku.....	481
9.7.7.2. Hierarchiczny algorytm ważonego sprawiedliwego szeregowania	481
9.7.7.3. Uproszczenia algorytmu ważonego sprawiedliwego szeregowania	484
9.7.8. Dyscypliny sprawiedliwego szeregowania z podziałem czasu	486
9.7.8.1. Algorytm WRR	486
9.7.8.2. Algorytm DRR	487
9.7.8.3. Algorytm CRR	487
9.7.9. Dyscypliny szeregowania w sieciach z jakością usług	488
9.8. Odwzorowanie jakości usług w sieciach z integracją usług	492
9.9. Podsumowanie	493
Literatura	495
10. Koncepcja usług zróżnicowanych	497
10.1. Podstawowe założenia koncepcji usług zróżnicowanych	497
10.2. Definiowanie usług w sieci usług zróżnicowanych	499
10.3. Ogólne zasady spedycji ruchu	501
10.4. Usługi sieci z różnicowaniem usług	502
10.5. Klasyfikowanie i kondycjonowanie ruchu	503
10.5.1. Klasyfikowanie ruchu	504
10.5.2. Kondycjonowanie ruchu	506
10.5.2.1. Moduł pomiarowy ruchu	507
10.5.2.2. Moduł znakowania ruchu	508
10.5.2.3. Moduł kształtowania ruchu	508
10.5.2.4. Moduł odrzucania ruchu	509
10.5.3. Lokalizacja modułów klasyfikowania i kondycjonowania ruchu	509
10.5.3.1. Lokalizacja w domenach źródłowych usług zróżnicowanych	509
10.5.3.2. Lokalizacja w węzłach brzegowych domen usług zróżnicowanych	510
10.5.3.3. Lokalizacja w węzłach wewnętrznych domen usług zróżnicowanych	511
10.6. Standardy spedycji ruchu	511
10.6.1. Grupy spedycji ruchu	511
10.6.1.1. Grupa AF PHB	511
10.6.1.2. Grupa EF PHB	513
10.6.2. Modele usług PHB	514
10.6.2.1. Usługa AS	514

10.6.2.2. Usługa Premium.....	515
10.6.2.3. Przykładowe realizacje usług Assured Service i Premium Service	517
10.6.2.3.1. Assured Service ze statycznym kontraktem SLA	518
10.6.3.2.2. Premium Service z dynamicznym kontraktem SLA.....	520
10.7. Wymagania sieci z koncepcją usług zróżnicowanych.....	523
10.7.1. Wymagania z zakresu różnicowania ruchu	523
10.7.2. Wymagania z zakresu obsługi ruchu w czasie rzeczywistym.....	524
10.7.3. Architektury routerów	525
10.7.3.1. Scentralizowane przetwarzanie i współdzielenie magistrali	525
10.7.3.2. Wieloprocessorowe przetwarzanie i współdzielenie magistrali.....	526
10.7.3.3. Wieloprocessorowe przetwarzanie i komutacja przestrzenna	526
10.7.3.4. Współdzielone przetwarzanie równoległe i komutacja przestrzenna.....	527
10.7.4. Optymalizacja przetwarzania pakietów w routerze	527
10.8. Klasyfikacja jednostek danych	528
10.8.1. Algorytmy klasyfikacji ruchu	530
10.8.2. Miary jakości algorytmów klasyfikacji	531
10.8.3. Ogólne sformułowanie zadania klasyfikacji.....	533
10.8.4. Algorytmy klasyfikacji.....	536
10.8.4.1. Dynamiczne buforowanie	536
10.8.4.2. Podejście geometryczne	537
10.8.4.3. Przestrzeń krotek	542
10.8.4.4. Algorytm RS	543
10.8.4.5. Algorytm BS	545
10.9. Wyznaczania adresu węzła trasy	548
10.9.1. Klasyfikacja w jednym wymiarze	548
10.9.2. Klasyfikacja w dwóch wymiarach.....	549
10.10. Modele różnicowania usług.....	549
10.10.1. Schematy względnego różnicowania usług.....	550
10.10.2. Model proporcjonalnego różnicowania usług	551
10.11. Zarządzanie zasobami	553
10.11.1. Zarządzanie pojemnością łączy wyjściowych.....	553
10.11.2. Przepływy nieadaptacyjne i adaptacyjne.....	554
10.11.3. Hierarchiczne zarządzanie pojemnością łączy	555
10.11.4. Skalowalność obsługi indywidualnych przepływów.....	556
10.11.4.1. Dyscypliny ważonego sprawiedliwego kolejkowania	556
10.11.4.1.1. Algorytmy idealnego szeregowania	556
10.11.4.1.2. Algorytmy dyskretnego szeregowania.....	556
10.11.4.2. Jakość obsługi separowanych przepływów.....	557
10.11.5. Zarządzanie usługami	560
10.12. Porównanie koncepcji sieci z integracją usług i sieci z różnicowaniem usług	561
10.13. Podsumowanie.....	564
Literatura	565
11. INŻYNIERIA RUCHU.....	568
11.1. Zadania inżynierii ruchu w sieciach	572
11.1.1. Mechanizmy jakości usług w sieciach z komutacją pakietów	572

11.1.2. Cele optymalizacji stopnia wykorzystania zasobów sieci	573
11.1.3. Komponenty systemu inżynierii ruchu w sieci	574
11.2. Monitorowanie stanów sieci	575
11.2.1. Algorytmy wektorów odległości	576
11.2.2. Algorytmy stanów łączy	576
11.3. Algorytmy wyznaczania tras w sieciach z jakością usług	577
11.3.1. Informacja i algorytmy wyznaczania tras	579
11.3.2. Algorytmy wyznaczania tras	579
11.3.3. Zadanie wyznaczania tras z ograniczeniami	580
11.3.3.1. Ogólne sformułowanie zadania wyznaczania tras z ograniczeniami ..	581
11.3.3.2. Złożoność obliczeniowa zadań wyznaczania tras z ograniczeniami ..	582
11.3.4. Wyznaczanie trasy z gwarancją jakości usług	583
11.3.4.1. Model sieci i trasy w sieci	584
11.3.4.2. Znajdowanie tras dopuszczalnych i optymalnych	585
11.3.4.3. Kryteria wyboru trasy optymalnej	586
11.4. Równoważenie obciążeń	587
11.4.1. Algorytmy rozdzielania przepływów	589
11.4.2. Algorytmy rozdziału o strukturze drzewa	590
11.4.3. Algorytmy rozdziału FCFS	592
11.4.4. Algorytmy rozdziału z funkcją skrótu	595
11.4.4.1. Podejście bezpośrednie	596
11.4.4.2. Podejście tablicowe	596
11.5. Technika komutacji etykiet	597
11.5.1. Wyznaczanie tras i przełączanie	597
11.5.2. Architektura MPLS	599
11.5.2.1. Etykieta	599
11.5.2.2. Klasy równoważności przekazywania ruchu FEC	600
11.5.2.3. Tablice przekazywania ruchu	601
11.5.2.4. Przydzielanie etykiet	601
11.5.2.5. Wyznaczanie tras w sieci	602
11.5.3. Modele zapewniania jakości usług sieci i komutacja etykiet	602
11.5.3.1. Model usług zintegrowanych i komutacja etykiet	602
11.5.3.2. Model usług zróżnicowanych i komutacja etykiet	603
11.5.4. Przykładowa realizacja usługi Premium Service w architekturze MPLS	603
11.6. Podsumowanie	606
Literatura	607
12. PODSUMOWANIE	610
SŁOWNIK	614
ZESTAWIENIE SKRÓTÓW	625
SKOROWIDZ	630

WPROWADZENIE

Przedmiotem monografii są wybrane zagadnienia sterowania ruchem w sieciach teleinformatycznych z komutacją pakietów, których zakres funkcjonalny i cele są ściśle związane z wymaganiami stawianymi przed sieciami teleinformatycznymi jako podsystemem rozproszonego systemu informatycznego.

Zakres zagadnień dotyczących sterowania ruchem w sieciach teleinformatycznych i omawianych w monografii jest w zasadzie ograniczony do funkcji tzw. części sieciowej architektury systemów teleinformatycznych. Zakres funkcjonalny części sieciowych różnych hierarchicznych, modułowych i warstwowych architektury systemów informatycznych dotyczy przekazu informacji poprzez sieć komunikacyjną. Funkcje realizowane przez części sieciowe, tzn. warstwy dolne logicznych architektury sieciowych, wspomagają realizację usług aplikacyjnych dostępnych dla użytkowników systemów teleinformatycznych w tzw. części użytkownika architektury sieciowych.

Podstawowym zadaniem sterowania ruchem w sieciach teleinformatycznych, niezależnie od realizowanej koncepcji dostarczania jakości usług użytkownikom sieci, jest optymalne wykorzystanie zasobów sieci. Realizowana w danej sieci koncepcja jakości usług definiuje zbiór ograniczeń zadania optymalnego wykorzystania zasobów. Do podstawowych zadań związanych ze sterowaniem ruchem należą zadania wyznaczania tras, przeciwdziałania przeciążeniom oraz sterowania przepływem. Złożoność wymienionych zadań optymalizacji rośnie wraz ze złożonością potrzeb różnicowania jakości usług sieciowych dostarczanych jednocześnie różnym grupom użytkowników tej samej sieci teleinformatycznej.

Omawiane w monografii zagadnienia dotyczą różnych sformułowań i rozwiązań zadań sterowania ruchem implementowanych w dolnych warstwach różnych architektury sieciowych, realizujących różne koncepcje jakości usług sieciowych. Prezentowane zagadnienia obejmują przykładowe rozwiązania zadań sterowania ruchem w dolnych warstwach, tzn.: fizycznej, łącza danych, sieciowej i transportowej architektury sieciowych realizujących różne koncepcje jakości dostarczanych usług sieciowych, tzn. bez gwarancji jakości, z integracją usług i z różnicowaniem usług.

Przedmiotem monografii jest przedstawienie ogólnych celów i zasad sterowania ruchem w sieciach teleinformatycznych, realizujących różne koncepcje dostarczania jakości usług sieciowych oraz przykładowych rozwiązań szczegółowych zadań związa-

nych ze sterowaniem ruchem, w tym m.in. przeciwdziałaniu przeciążeniom i sterowaniu przepływem.

Omawiane, w poszczególnych rozdziałach, rozwiązania z zakresu sterowania ruchem w sieciach teleinformatycznych dotyczą sformułowań zadań sterowania ruchem oraz przykładowych ich rozwiązań, charakterystycznych dla sieci realizujących różne koncepcje dostarczania jakości usług sieciowych. Analiza przytaczanych w opracowaniu rozwiązań ilustruje ewolucję zakresu funkcjonalnego zadań sterowania ruchem w sieciach teleinformatycznych oraz ewolucję koncepcji, metod i narzędzi stosowanych do rozwiązywania zadań dostarczania jakości usług sieciowych.

W rozdziale pierwszym przedstawiono ogólną charakterystykę zagadnień związanych z jakością usług sieci teleinformatycznych. Omówiono zależności pomiędzy jakością usług sieci teleinformatycznych a architekturą warstwową sieci. Zwrócono uwagę na – wynikającą z architektur warstwowych – hierarchię zadań komunikacyjnych i kryteriów jakości usług.

Rozdział drugi dotyczy powszechnie stosowanych miar jakości usług sieciowych zarówno do opisu jakościowych i ilościowych wymagań użytkowników sieci teleinformatycznych, jak i oceny rozwiązań teleinformatycznych. Wraz z rozwojem funkcjonalności sieci teleinformatycznych, związanych zwłaszcza z jednoczesną obsługą ruchu należącego do różnych klas ruchu, rośnie liczba równocześnie wykorzystywanych miar jakości.

W rozdziale trzecim zaprezentowano wybrane zadania odwzorowania opisów jakości usług w procesach dekompozycji i agregacji procesów komunikacyjnych, realizowanych w warstwowych sieciach teleinformatycznych. Odwzorowania wartości parametrów jakości usług w modelu warstwowym są o tyle istotne, że jakość usługi sieciowej – oceniana przez użytkownika – jest efektem działań wielu mechanizmów, lokalizowanych w różnych warstwach sieci. Zależności pomiędzy wynikową jakością usług a jakością usług procesów wspierających jest podstawą efektywnego projektowania i analizy sieci teleinformatycznych.

W rozdziale czwartym przedstawiono charakterystyki i sformułowano zadania komunikacji realizowane w sieciach teleinformatycznych o architekturze warstwowej. Omówiono zależności pomiędzy jakością usług dostarczanych w poszczególnych zagnieżdżonych warstwach hierarchicznych i modułarnych architektur warstwowych. Zależności te są o tyle istotne, że jakość realizacji zadania komunikacji w warstwie nadrzędnej jest zawsze determinowana jakością zadań komunikacji realizowanych w warstwach podrzędnych danej architektury sieci.

Rozdział piąty dotyczy zagadnień wyznaczania tras w sieciach, zapobiegania przeciążeniom sieci i sterowania przepływem w sieciach. Wymienione zadania są ze sobą ściśle powiązane, na ogół bowiem rozwiązanie jednego z nich albo determinuje, albo ogranicza rozwiązanie innego z wymienionych zadań. Podstawowe, klasyczne sformułowania zadań optymalizacji i sterowania w systemach teleinformatycznych, dostarczających usług bez gwarancji jakości, mogą być łatwo rozszerzane na sformułowania właściwe dla innych koncepcji dostarczania jakości usług.

W rozdziale szóstym scharakteryzowano metody przeciwdziałania przeciążeniom oraz sterowania przepływem stosowane w sieciach teleinformatycznych, zaprojektowanych do obsługi ruchu elastycznego, tzn. do obsługi wrażliwego ruchu na straty i niewrażliwych na opóźnienia jednostek danych. Opisywane metody, mechanizmy i procedury są charakterystyczne dla rozwiązań teleinformatycznych nie gwarantujących jakości usług. Znaczenie omawianych metod, mechanizmów i procedur wynika m.in. z tego, że rozwiązania te są stosowane, wprost lub po odpowiednich modyfikacjach, do realizacji różnych koncepcji sieci teleinformatycznych, zorientowanych na uwzględnianie rozmaitych wymagań dotyczących jakości usług systemu i efektywności wykorzystania jego zasobów. Charakterystyka metod sterowania przepływem ruchu elastycznego w systemach teleinformatycznych pokazuje zarówno uniwersalność, jak i skalowalność wybranych mechanizmów.

W rozdziale siódmym omówiono zagadnienia związane z przeciwdziałaniem przeciążeniom i sterowaniem przepływem, które łączy się z organizacją przepływów w sieciach teleinformatycznych o dużych szybkościach transmisji. Zwiększenie szybkości transmisji i wynikające z tego wymagania i ograniczenia sieci teleinformatycznych powodują wyraźny podział implementowanych mechanizmów na mechanizmy prewencyjne i reakcyjne (dotyczy to nie tylko szybkich systemów teleinformatycznych). Szybkość transmisji i obsługi jednostek danych uwypukliła różne znaczenie i przydatność mechanizmów, należących do omawianych grup, na różnych etapach obsługi zgłoszeń. Wzrost szybkości transmisji zwiększa znaczenie metod sterowania prewencyjnego oraz konieczność poszukiwania efektywnych obliczeniowo algorytmów heurystycznych rozwiązywania zadań sterowania reakcyjnego.

Rozdział ósmy zawiera ogólną charakterystykę dwóch koncepcji jakości usług w systemach teleinformatycznych z komutacją jednostek danych, tzn. koncepcji (modeli) usług zintegrowanych i usług zróżnicowanych. Koncepcje te są oparte na założeniu, że warunkiem spełnienia wymagań użytkowników i efektywnego wykorzystania zasobów jest różnicowanie ruchu i sposobów jego obsługi, tzn. że różne klasy ruchu w systemach teleinformatycznych muszą i mogą być obsługiwane w rozmaity sposób w celu zagwarantowania odpowiedniej jakości usług wyróżnionym klasom ruchu. Omawiane koncepcje, mające charakter uniwersalny, są oparte na tych samych założeniach, a podstawą różnic ich efektywności i skalowalności jest różny poziom granulacji obsługiwanego ruchu. Uniwersalność koncepcji dostarczania jakości usług polega na tym, że możliwe jest ich współistnienie w sieci teleinformatycznej. Daje to możliwość jednoczesnego organizowania usług o różnej jakości dla przepływów współdzielących zasoby systemu. Koncepcje zapewniania jakości usług w systemach teleinformatycznych są adresowane przede wszystkim do systemów wykorzystujących architekturę TCP/IP, ale ich znaczenie jest bardziej ogólne, co wynika z tego, że w koncepcjach tych zawarte są doświadczenia i wnioski z eksploatacji sieci teleinformatycznych o różnych rodowodach i przeznaczeniu oraz że są one przenaszalne.

W rozdziale dziewiątym scharakteryzowano podstawowe rozwiązania właściwe dla koncepcji usług zintegrowanych, zwłaszcza dla modelu usługi z gwarancją jej jakości. Koncepcja usług zintegrowanych jest związana z realizacją klasycznych zadań sterowania przepływem, z tym że gwarancja jakości usług wymaga ich uzupełnienia o mechanizmy wymuszania alokacji zasobów. Szczególną uwagę zwrócono na zagadnienia gwarantowania jakości usług oraz algorytmy sprawiedliwego szeregowania w wersji ciągłej i dyskretnej.

Rozdział dziesiąty dotyczy wybranych mechanizmów implementowanych w modelu usług zróżnicowanych. Podstawowa różnica między modelami usług zintegrowanych i zróżnicowanych wynika z podejścia do jakości transmisji ruchu. Koncepcja usług zintegrowanych jest zorientowana na przepływy elementarne; przedmiotem obsługi i sterowania w koncepcji usług zróżnicowanych jest natomiast przepływ zagregowany. Różne założenia sposoby obsługi przepływów elementarnych i zagregowanych decydują o zróżnicowaniu systemów realizujących wymienione dwie koncepcje, dlatego przy omawianiu koncepcji usług zróżnicowanych zwrócono uwagę na zagadnienia klasyfikacji jednostek danych, budowy i funkcjonalności węzłów sieci oraz zarządzanie zasobami.

W rozdziale jedenastym omówiono zagadnienia związane z inżynierią ruchu w sieciach teleinformatycznych realizujących różne koncepcje zapewnienia jakości dostarczanych usług. Podstawowym zadaniem metod inżynierii ruchu jest efektywność wykorzystania zasobów sieci teleinformatycznej przy ograniczeniach dotyczących jakości dostarczanych usług, uzyskiwana w szczególności przez równoważenie obciążeń systemu, co jest związane z formułowaniem i rozwiązywaniem zadań: wyznaczania tras z ograniczeniami dotyczącymi zasobów oraz rozdzielania ruchu i wymuszania przepływów w wyznaczonych trasach.

Mnogość wersji i modyfikacji różnych metod, mechanizmów i procedur sterowania przepływami w sieciach teleinformatycznych o różnych architekturach i przeznaczeniu pokazuje, że zbiór podstawowych koncepcji oraz rozwiązań stosowanych do zapewnienia jakości dostarczanych usług i efektywnego wykorzystania zasobów w zasadzie się nie zmienia. Efektywność tych koncepcji i rozwiązań wzrasta zwłaszcza w wyniku rozszerzania zakresu ich stosowania oraz opracowywania efektywnych obliczeniowo i pamięciowo algorytmów rozwiązywania zadań wyznaczania tras, alokacji zasobów, przeciwdziałania przeciążeniom i sterowania przepływami.

1. ARCHITEKTURA I PROTOKOŁY JAKOŚCI USŁUG SIECI TELEINFORMATYCZNYCH

Stale rosnące zapotrzebowanie na usługi multimedialne, aplikacje w architekturze klient–serwer, usługi komunikacyjne w standardach Internetu i intranetu czy usługi transportowe szerokopasmowych sieci wynika m.in. z tego, że transmisja danych i komunikacja pomiędzy komputerami jest integralną częścią każdego obszaru przedsiębiorczości. Co więcej, z każdą zmianą aktywności (ukierunkowanej na zwiększenie konkurencyjności, produktywności itd.) wiąże się wzrost zapotrzebowania na komunikację i wymianę różnego rodzaju danych. Z dowolną aktywnością wiążą się odpowiednie systemy przetwarzania wiedzy i informacji, które w uzasadnionych przypadkach są wspomagane przez różne rozproszone systemy automatycznego przetwarzania danych (systemy informatyczne i sieci komputerowe).

Obecny i przewidywany stan sieci teleinformatycznych jest konsekwencją wielu czynników, w tym m.in. zmian organizacji aktywności, zachowań i potrzeb użytkowników, technologii informatycznych i komunikacyjnych, infrastruktur sieci teleinformatycznych i stosowanych systemów informatycznych. Zmiany rozwiązań i zastosowań sieci komputerowych są następstwem ciągle wprowadzanych innowacji organizacyjnych, strukturalnych, technologicznych i aplikacyjnych, mających na celu nadążanie za zapotrzebowaniem na usługi informacyjne i komunikacyjne.

Metodologie, technologie i techniki informatyczne oraz telekomunikacyjne pełnią w rozwoju systemów informatycznych i sieci komputerowych podwójną rolę. Z jednej strony ich rozwój i doskonalenie jest stymulowane rosnącym zapotrzebowaniem na poprawę funkcjonalności i efektywności zautomatyzowanych systemów informacyjnych, z drugiej zaś ich rozwój oznacza powstawanie nowych usług informatycznych i komunikacyjnych, których implementacja wpływa na zakres i jakość usług informacyjnych dostarczanych użytkownikowi.

Inne postrzeżenie znaczenia informatyki we współczesnych organizacjach i instytucjach to akcentowanie jej roli sprawczej (ang. *enabler*) i implementacyjnej (ang. *implementer*). W pierwszej z wymienionych ról informatyka jest postrzegana jako skuteczne narzędzie do identyfikacji zdarzeń i procesów wpływających na efektywność systemów wytwarzania, zarządzania itd., a następnie do modelowania, symulacji,

weryfikacji i optymalizacji różnych koncepcji rozwiązań organizacyjnych, informacyjnych itd. Akcentowanie roli implementacyjnej oznacza podkreślanie znaczenia informatyki dla skutecznego i efektywnego wdrażania i eksploatacji przyjętych wcześniej koncepcji organizacyjnych, informacyjnych i innych.

Na obserwowane i przewidywane zmiany sieci komputerowych i ich podsystemów komunikacyjnych podstawowy wpływ mają:

- modele przetwarzania, realizowane w systemach informatycznych wspomaganych przez systemy komunikacji przestrzennej,
- wzrastające zapotrzebowanie aplikacji użytkowników na przepustowość, powodowane zmianami modeli przetwarzania i form prezentacji danych,
- zastępowanie i (lub) uzupełnianie systemów celowo zorientowanej dystrybucji danych aktywnym wyszukiwaniem danych (informacji) przez użytkownika,
- zastępowanie przeźroczystych dla aplikacji podsystemów komunikacyjnych rozwiązaniami stosującymi syntaktykę i semantykę przekazywanych danych.

1.1. APLIKACJE W SIECIOWYCH SYSTEMACH TELEINFORMATYCZNYCH

Sieciowe systemy teleinformatyczne stanowią atrakcyjną platformę implementacji różnych aplikacji, obejmujących systemy przetwarzania, pracy grupowej, interaktywnych symulacji czy rozproszonych baz danych. W węzłach sieciowych systemów informatycznych udostępniane są różne zasoby: pamięci, przetwarzania, dostępu do zbiorów danych oraz do różnych urządzeń wejściowych i wyjściowych itp.

Efektywna realizacja aplikacji w sieciowym systemie informatycznym zależy od właściwości sieci teleinformatycznej, zwłaszcza od jej szybkozmiennych charakterystyk istotnych dla danej aplikacji. Zakres zmienności charakterystyk ruchu sieciowego, generowanego przez różne aplikacje, może być bardzo duży. Znajomość tych charakterystyk jest warunkiem efektywnej alokacji zasobów dla aplikacji albo modyfikacji zapotrzebowania na zasoby. Dla aplikacji generujących wiele danych w krótkich przedziałach czasu duża chwilowa przepustowość sieci jest ważniejsza od średniej, natomiast dla generujących ruch ze stałą szybkością ważniejsza jest dostępność przepustowości uśrednianej w długich okresach.

Efektywne wykorzystanie dynamicznie zmieniających się ilości dostępnych zasobów dla różnych aplikacji w sieciowych systemach informatycznych, przy jednoczesnym spełnianiu wymagań użytkowników aplikacji (czas reakcji, współczynnik strat, pojemność pamięci buforowej, niezawodność, bezpieczeństwo, koszt itd.), jest jednym z kryteriów projektowania i oceny aplikacji.

Uzyskanie i utrzymanie wymaganej jakości aplikacji w sieciowym systemie informatycznym, cechującym się dynamicznymi zmianami ilości dostępnych zasobów, jest możliwe dzięki:

➤ Gwarancji jakości usług sieci dla aplikacji (ang. *quality of service assurance*) – gwarancja jakości usług sieci redukuje zmienność dostępnych zasobów z punktu widzenia aplikacji.

➤ Dopasowaniu aplikacji do zmienności dostępnych zasobów sieci (ang. *network-aware applications*) – zmiana jakości usług sieci powoduje zmianę zapotrzebowania aplikacji na zasoby.

Gwarantowanie jakości usług sieciowych polega na rezerwacji zasobów sieciowych dla aplikacji. Jakość aplikacji w rozwiązaniach, w których wykorzystują one gwarancję jakości usług sieci, jest ograniczona możliwościami: dokładnego prognozowania zapotrzebowania na zasoby oraz jego ograniczania do ilości dostarczanych w ramach gwarancji jakości usług. W ogólnym przypadku istotna dla jakości aplikacji jest zdolność do sterowania zapotrzebowaniem na zasoby, tak aby ilość użytkowanych zasobów nie przekraczała zarezerwowanych lub obserwowanych ilości dostępnych zasobów. Usługi sieciowe z gwarancją jakości usług są droższe od usług sieciowych dostarczanych bez gwarancji jakości. Korzystanie z gwarancji jakości usług sieciowych znacząco ogranicza elastyczność aplikacji.

Realizacja aplikacji adaptujących zapotrzebowanie na zasoby sieci, zależnie od ilości dostępnych w sieci zasobów, wymaga posiadania przez nią zdolności do pozyskiwania informacji o dostępności żądanych zasobów i stanie sieci. Warunkiem pozyskiwania takich informacji jest zdolność sieci do udostępniania ich węzłom, w których uruchamiane są aplikacje, lub aplikacjom. Zdolność do pozyskania informacji o dostępności zasobów sieci heterogenicznych za pomocą interfejsu, niezależnego od architektury sieci, oznacza możliwość wykonywania aplikacji w dowolnej sieci, pod warunkiem że udostępni ona informacje o stanie i dostępności zasobów.

Niezależnie od stosowanej strategii dostępu aplikacji do zasobów sieci, tzn. rezerwacji zasobów sieciowych lub adaptacji zapotrzebowania aplikacji na nie, jakość aplikacji w sieciowym systemie informatycznym zależy m.in. od następujących czynników:

➤ **Węzły i ich liczba** – realizacja aplikacji w systemie sieciowym wymaga wyboru węzłów, spełniających specyficzne wymagania, oraz liczby węzłów. Minimalna liczba węzłów zwykle jest określona przez sposoby i miejsca wprowadzania i wyprowadzania danych. Maksymalną natomiast liczbę węzłów zwykle determinują koszty komunikacji. Chociaż sporo aplikacji może korzystać z wielu węzłów, to zwiększenie ich liczby powoduje wzrost kosztów komunikacji międzywęzłowej.

➤ **Architektura sieci** – specyfika aplikacji może powodować, że różne rozwiązania sieciowe są w różnym stopniu przydatne do jej obsługi. Żądania dystrybucji danych w trybie rozgłoszeniowym, obsługi procesów czasu rzeczywistego, obsługi transakcji itp. dyskryminują jedno, a preferują inne architektury sieciowe.

➤ **Równoważenie obciążeń** – zmiany środowiska sieciowego mogą powodować konieczność zmiany początkowych warunków realizacji aplikacji. Jeżeli nie jest ona elastyczna ze względu na zróżnicowanie zasobów komunikacyjnych i przetwarzania, to jedynym rozwiązaniem jest zastosowanie innych warunków jej realizacji, tj. zmiany węzłów i ich liczby oraz wynikającej z tego reorganizacji połączeń komunikacyjnych.

➤ **Miary jakości aplikacji** – warunkiem realizacji różnych aplikacji jest spełnienie charakterystycznych dla nich wymagań (wrażliwość na zmienność opóźnienia w aplikacjach odtwarzania, synchronizacja przekazu w aplikacjach multimedialnych itd.). W danej sieci można wieloma różnymi sposobami spełniać specyficzne wymagania aplikacji, co może spowodować ograniczenie przetwarzania i ruchu generowanego przez aplikacje.

➤ **Lokalne i zdalne przetwarzanie** – ta sama funkcjonalność aplikacji może być uzyskana przy różnym udziale przetwarzania lokalnego i zdalnego. Wybór modelu przetwarzania zależy od dostępności usług sieci oraz mocy obliczeniowych dla ustalonego modelu kosztów.

1.2. JAKOŚĆ USŁUG SIECI TELEINFORMATYCZNYCH

1.2.1. JAKOŚĆ USŁUG

Jakość usług (ang. *Quality of Services – QoS*) w sieciach teleinformatycznych dotyczy klasyfikacji generowanego ruchu (ang. *traffic*) lub przepływów (ang. *flow*) w celu umożliwienia traktowania ich w szczególności – w porównaniu z innymi klasami ruchu – sposób. Jakość usług pozwala na różnicowanie klas ruchu i usług dostarczanych różnym klasom ruchu w sieci. Na ogół jakość usług to zdolność sieci (lub jej elementów) do gwarantowania konsekwentnego dostarczania usług dla wyróżnionych klas ruchu na określonym poziomie. Dostępność efektywnego różnicowania usług sieci dla wyróżnionych klas ruchu oznacza, że alokacja określonych zasobów sieciowych zależy od obsługiwanego w sieci ruchu zgodnie z jakościowymi i ilościowymi wymaganiami. Oznacza to także, że możliwość dostarczania różnej jakości usług różnym użytkownikom wymaga zarządzania zbiorem użytkowników sieci w celu legalizacji, weryfikacji i taryfikacji ich żądań jakości usług.

Zastosowanie koncepcji różnej jakości usług sieci dla różnych klas ruchu wymaga implementowania w sieci mechanizmów umożliwiających różnicowanie, identyfikowanie, zarządzanie i rozliczanie różnych klas ruchu i ich elementów. Zakres działania tych mechanizmów może być różny, może dotyczyć sieci, klasy ruchu, przepływu w sieci lub poszczególnych składowych przepływu. Zakres działania danego mechanizmu zależy od poziomu ogólności, na jakim zasoby danej sieci komputerowej są rozważane, alokowane i zarządzane.

Wymagane mechanizmy, implementowane w sieciach komputerowych w postaci protokołów, zalicza się do następujących grup mechanizmów:

- sygnalizacji dostępności różnej jakości usług sieciowych,
- alokacji ograniczonych zasobów sieciowych pomiędzy różne klasy ruchu i ich elementy,
- znakowania różnych klas ruchu ze względu na jakość dostarczanych i żądanych usług, pozwalającego na priorytetowanie i agregację ruchu,

➤ legalizacji wymagań różnych klas ruchu dotyczących zróżnicowanej jakości usług,

➤ weryfikacji źródeł ruchu i żądanej przez te źródła różnej jakości usług.

Dopóki w zmieniających się warunkach różnym klasom ruchu są oferowane i dostarczane usługi sieciowe o zróżnicowanej jakości, dopóty istnieje zapotrzebowanie na implementację w sieci:

➤ zasad przydzielania usług określonej jakości,

➤ reguł weryfikacji uprawnień do korzystania z usług i ich jakości,

➤ procedur nadzorowania przestrzegania obowiązujących zasad i reguł.

Zbiór takich obowiązujących w sieci zasad, reguł i procedur tworzy system jakości usług, który jest podstawowym składnikiem każdej sieci przygotowanej do różnicowania jakości oferowanych usług (ang. *QoS-enabled network*).

Znaczenie systemu jakości usług wzrasta wtedy, gdy zwiększa się granulacja zarządzanych w sieci elementów składowych ruchu oraz gdy rośnie liczba różnych, dostępnych w sieci usług i poziomów ich jakości. System jakości usług ma szczególne znaczenie w sieciach stosujących zasadę komutacji jednostek danych (wiadomości, pakietów, ramek, komórek). Sieci takie są bowiem niedeterministyczne w tym znaczeniu, że udostępnianie zasobów sieciowych klasom ruchu nie jest poprzedzone rezerwacją zasobów. Sieci, w których zasoby udostępniane klasom ruchu nie są predefiniowane są definiowane, projektowane i zarządzane jako systemy z opóźnieniami (ang. *delay-systems*). Mniejsze znaczenie systemu jakości usług w sieciach korzystających z komutacji łączy wynika z tego, że dostęp do zasobów sieci odbywa się na podstawie wcześniejszej rezerwacji zasobów. Sieci z komutacją łączy są definiowane, projektowane i zarządzane jako systemy ze stratami (ang. *loss-systems*). Brak zasobów sieciowych w chwili zgłoszenia zapotrzebowania na zasoby jest równoważny z odrzuceniem zgłoszenia.

W większości sieci komputerowych udostępnianie usług odbywa się zgodnie z zasadą najwyższej staranności, tzn. bez gwarancji jakości usług. Jest tak nawet w tych sieciach, w których część zasobów sieciowych jest dedykowana (wydzielona) na potrzeby zapewnienia usług o predefiniowanej jakości.

Stosowanie w sieciach obsługi ruchu zgodnie z zasadą największej staranności powoduje, że:

➤ całkowitą odpowiedzialność za jakość obsługi ruchu (i związaną z tym złożoność) ponosi użytkownik sieci komputerowej,

➤ sieć komputerowa jest (może być) względnie prosta,

➤ ewentualne przekroczenie zasobów sieci komputerowej przez zapotrzebowanie na jej usługi powoduje pogorszenie jakości dostarczanych usług, a nie odmowę ich dostarczenia.

1.2.2. RÓŻNICOWANIE JAKOŚCI USŁUG

Sieci teleinformatyczne są projektowane i budowane w celu wspierania możliwości korzystania z aplikacji. Zmiany i wzrost wymagań aplikacji prowadzą do nowych roz-

wiązań sieci teleinformatycznych; są one na ogół następstwem innych od dotychczasowych oczekiwań i wymagań użytkowników aplikacji, możliwych do spełnienia dzięki dostępnym technikom telekomunikacyjnym oraz informatycznym.

Wzrost zasobów sieci teleinformatycznych (zwłaszcza przepustowości) oraz zapotrzebowanie na różnicowanie jakości usług dostarczanych użytkownikom zmusza do wprowadzania mechanizmów różnicowania i zapewniania jakości usług (ang. *quality of services – QoS*). Rosnące możliwości sieci w zakresie transmisji, komunikacji oraz zarządzania ruchem pozwalają na jednoczesną, równoległą obsługę strumieni ruchu generowanych przez znacznie różniące się aplikacje, wymagające zróżnicowanych usług oferowanych przez sieci komputerowe. Celem systemów (polityk) jakości usług w sieciach jest wprowadzanie określonego poziomu sterowania ruchem i przewidywalności jakości usług w miejsce obsługi ruchu bez gwarancji jakości (ang. *best-effort basis*).

Implementacja mechanizmów sterowania i zarządzania ruchem w sieci, powodująca wzrost jakości obsługi ruchu, w porównaniu z obsługą bez gwarancji jakości, oznacza także zasadnicze zmiany z zasadach projektowania sieci teleinformatycznych. Zmiany te polegają na wprowadzeniu do praktycznie wszystkich elementów sieci funkcji sterowania i zarządzania ruchem i przepływami. Główną konsekwencją takich zmian w zadaniach projektowania jest odstępstwo od sprawdzonej wcześniej zasady, zgodnie z którą większość mechanizmów sterowania i zarządzania ruchem lokalizowana była na brzegu sieci, w celu zachowania możliwie największej jej prostoty.

Zapotrzebowanie na różnicowanie jakości usług ma charakter zarówno obiektywny, jak i subiektywny. Różne klasy ruchu, generowanego przez różne aplikacje i obsługiwanego w sieciach, ogólnie dzielone na dwie klasy:

➤ ruch elastyczny (ang. *elastic traffic*), obsługiwany zwykle w trybie bezpołączeniowym (ang. *connectionless*), na ogół tolerujący opóźnienia i fluktuację (wariancję) opóźnienia (ang. *delay tolerant*), co jest następstwem tolerowania zmian przepustowości w sieci, oraz wrażliwy na straty (ang. *loss sensitive*), co wymaga stosowania procedur zabezpieczania przed błędami i stratami,

➤ ruch strumieniowy (ang. *inelastic traffic*), obsługiwany zwykle w trybie połączeniowym (ang. *connection-oriented*), w ogólnym przypadku tolerujący straty (ang. *loss tolerant*), co wynika m.in. z redundancji informacyjnej ruchu, oraz wrażliwy na opóźnienia (ang. *delay sensitive*), fluktuację opóźnienia (ang. *delay jitter*) i przepustowość sieci (ang. *throughput sensitive*), co wynika z konieczności obsługi w czasie rzeczywistym (ang. *real-time sensitive*),

zarówno wymagają (oczekują), jak i akceptują zróżnicowaną jakość usług sieci, determinowaną różną ilością zaangażowanych zasobów sieci.

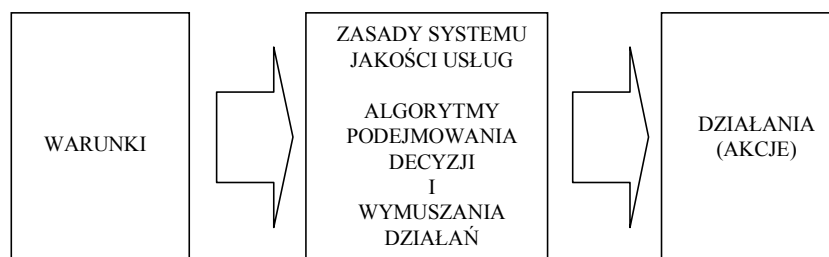
Ograniczone zasoby sieci oraz różne wymagania dotyczące jakości obsługi różnych klas ruchu są naturalną przyczyną potrzeby i konieczności różnicowania jakości możliwych, oferowanych usług. Jednocześnie akceptowanie przez użytkowników

różnej jakości usług dostarczanych przez sieci prowadzi do lepszego wykorzystania zasobów sieci i – w konsekwencji – do wzrostu poziomu zaspokojenia potrzeb użytkowników sieci.

1.2.3. SYSTEM JAKOŚCI USŁUG

W najprostszym znaczeniu system jakości usług (ang. *QoS system*), którego celem jest realizacja określonej polityki jakości usług (ang. *QoS policy*), jest zbiorem jednoznacznych i sprawdzalnych zasad opisujących działania (akcje), jakie są podejmowane wtedy, gdy w niedeterministycznym środowisku sieci teleinformatycznej spełnione są określone warunki.

W ogólnym przypadku opis dobrze zdefiniowanego systemu jakości usług, równoważnego z polityką jakości usług, jest charakteryzowany zbiorem definiowalnych warunków, działań oraz algorytmów sterujących procesami podejmowania decyzji o tym, które z warunków są w danym stanie sieci aktywne oraz jakie działania w takiej sytuacji należy podjąć (rysunek 1.1).



Rysunek 1.1. System jakości usług definiowany jest przez warunki, algorytmy i działania

Naturalne jest żądanie, aby każdemu określone mu zbiorowi warunków odpowiadało tylko jedno określone działanie – tak rozumiany system jakości usług jest automatem ze skończoną liczbą stanów.

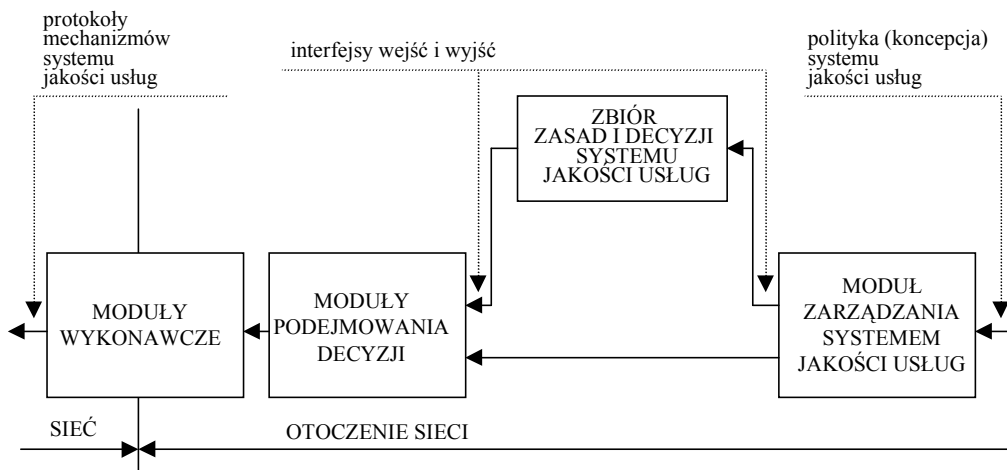
Zwykle polityki jakości usług (systemy jakości usług) są budowane jako systemy złożone z polityk prostszych. Modularność, hierarchiczność i zagnieżdżanie systemów jakości usług upraszcza zadanie zarządzania systemami realizującymi tak konstruowane polityki i prowadzi do systemów otwartych, modularnych, elastycznych i skalowalnych.

Konstrukcja systemu jakości usług jest ogólnym modelem, w którym wyróżnia się – na podstawie zdefiniowanych zakresów funkcjonalnych – dwie, logicznie separowane, podstawowe grupy modułów:

- moduły podejmowania decyzji (ang. *decision component*),
- moduły wykonawcze (ang. *enforcement component*).

Moduły wykonawcze umożliwiają realizację decyzji pobieranych ze zbioru decyzji (repozytorium) systemu. W najogólniejszym przypadku systemu jakości usług

o omawianej konstrukcji podstawą jego działania jest założenie, że efektem współpracy modułów podejmowania decyzji i modułów wykonawczych, organizowanej jako proces wymiany zapytań i decyzji, jest realizacja wcześniej zdefiniowanego celu (rysunek 1.2).



Rysunek 1.2. Elementy funkcjonalne systemu jakości usług

Współpraca pomiędzy modułami podejmowania decyzji oraz wykonawczymi przebiega w następujący sposób:

Moduły wykonawcze, lokalizowane na granicy sieci i wyposażone w jeden lub wiele interfejsów, przekazują predefiniowane elementy systemu wraz z odpowiednimi żądaniami, wykorzystując do tego celu określone protokoły. Otrzymanie żądania modułu wykonawczego przez moduł podejmowania decyzji oznacza uruchomienie predefiniowanego procesu (np. przeglądania, interpretacji, detekcji itp.) prowadzącego do wygenerowania decyzji; decyzja ta jest przekazywana do modułu wykonawczego za pomocą określonego protokołu.

W sieci może być implementowanych jednocześnie wiele modułów wykonawczych i podejmowania decyzji. Liczba modułów wykonawczych zależy od liczby wyróżnionych procesów sterowanych i zarządzanych w danej sieci, natomiast liczba modułów podejmowania decyzji zależy od struktury systemu jakości usług (scentralizowanej, hierarchicznej, rozproszonej itd.). Liczba modułów wykonawczych zwykle zależy od liczby punktów dostępu (ang. *access point*) do usług sieci, które mają różną lokalizację i są rozdzielone przestrzennie. Z naturalnych powodów liczba modułów podejmowania decyzji jest zwykle mniejsza od liczby modułów wykonawczych.

1.2.4. FUNKCJE SYSTEMU JAKOŚCI USŁUG

Zbiór funkcji realizowanych przez moduły wykonawcze i podejmowania decyzji dzieli się na trzy podzbiory:

- podejmowania decyzji (ang. *decision making*),
- wykonywania decyzji (ang. *enforcement*),
- monitorowania (ang. *metering*).

Celem działania modułów podejmowania decyzji jest realizacja zbioru funkcji niezbędnych do przygotowania decyzji adekwatnej do zidentyfikowanych i raportowanych warunków (stanów) w sieci. Oznacza to, że w module podejmowania decyzji – po otrzymaniu żądania modułu wykonawczego – następuje sprawdzanie zasad, porównanie opisów interfejsów, uruchomienie algorytmów przeglądania i interpretowania, detekcja potencjalnych konfliktów działań, ustalanie działań odpowiednich do warunków i stanu sieci, uruchomienie procedur wyznaczania działań i przekazanie wyników do modułów wykonawczych.

Wykonywanie decyzji oznacza implementację działań jednoznacznie odpowiadających przygotowanym decyzjom, zgodnych z systemem jakości usług i adekwatnych do aktualnego stanu sieci. Warunki, które są wymuszane przygotowywaniem i implementacją decyzji, mogą być statyczne (np. adresy źródeł i ujść) lub dynamiczne (np. chwilowa dostępność kanałów transmisyjnych).

Mechanizmy podejmowania i implementacji decyzji wymagają pasywnego i (lub) aktywnego monitorowania środowiska systemu jakości usług, tzn. sieci teleinformatycznej i jej użytkowników. Celem monitorowania użytkowników sieci jest gromadzenie informacji dotyczących jakości otrzymywanych przez nich usług oraz ich zachowań w reakcji na określoną jakość usług, zwłaszcza wtedy, gdy jakość tych usług jest niższa od wymaganej oraz gdy użytkownicy korzystają z usług, do których nie są uprawnieni.

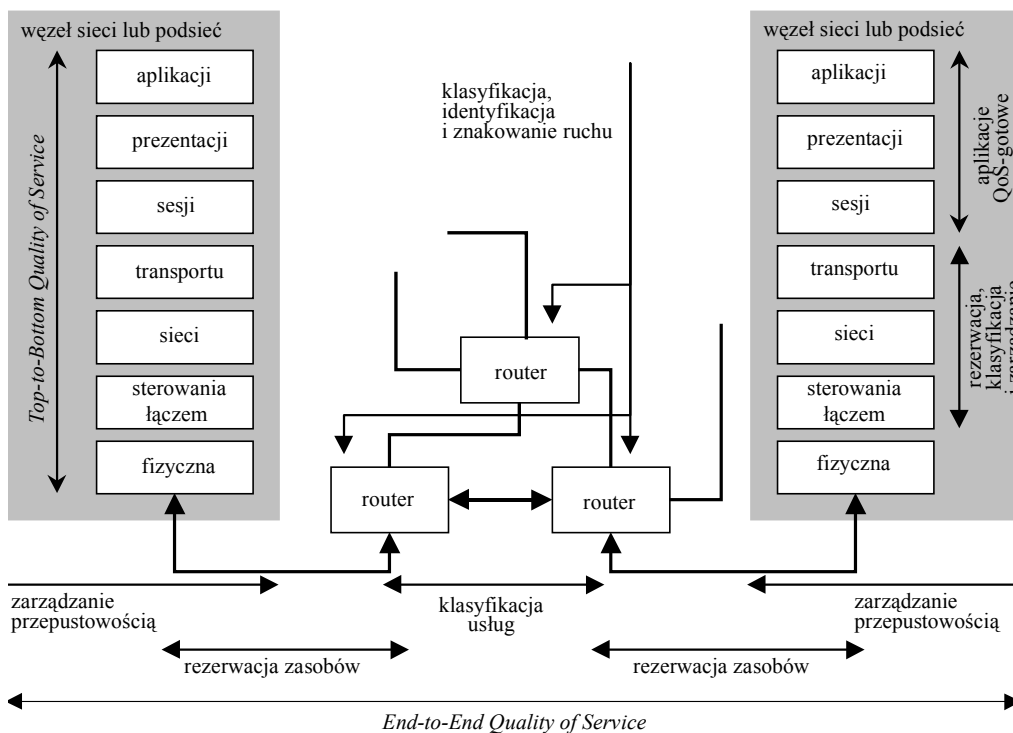
Monitorowanie sieci komputerowej i jej elementów składowych ma na celu sprawdzanie możliwości dostępu do zasobów niezbędnych do spełniania ilościowych i jakościowych wymagań użytkowników. Monitorowanie może być podzielone na zadania realizowane przez moduły podejmowania decyzji i moduły wykonawcze. Na ogół celem monitorowania realizowanego przez moduły podejmowania decyzji jest sprawdzanie, czy jakość oferowanych usług spełnia wymagania użytkowników, oraz – jeżeli nie spełnia – ustalanie działań, jakie należy podjąć, aby wymagania były spełniane.

Celem monitorowania środowiska systemu jakości usług przez moduły wykonawcze jest nadzór nad przestrzeganiem reguł korzystania z systemu jakości usług przez jego użytkowników w danym środowisku. Wynikiem monitorowania jest aktywna rola modułów wykonawczych w zadaniach sterowania dostępem, klasyfikacji i identyfikacji przepływów oraz kształtowania charakterystyk ruchu wprowadzanego do sieci komputerowej (odrzućanie żądań dostępu, wygładzanie charakterystyk, spowalnianie napływu itd.) i ruchu transferowanego przez sieć komputerową (spowalnianie przepływu, zmiana sposobów obsługi wydzielonych części ruchu, odrzućanie przepływów itp.).

1.3. ARCHITEKTURY WARSTWOWE SYSTEMÓW JAKOŚCI USŁUG

1.3.1. ŁAŃCUCHY JAKOŚCI USŁUG

Funkcjonalność systemów jakości usług w sieciach teleinformatycznych jest zawarta i organizowana w strukturach hierarchicznych oraz szeregowych, łączących powszechnie stosowane w architekturach sieci komputerowych koncepcje hierarchicznego zagnieżdżenia usług (ang. *top-to-bottom*) oraz szeregowej ich organizacji (ang. *end-to-end*) (rysunek 1.3).



Rysunek 1.3. Funkcjonalność hierarchicznej (*top-to-bottom*) i szeregowej (*end-to-end*) organizacji jakości usług systemów teleinformatycznych

Uzupełniającym się – hierarchicznym i szeregowym – koncepcjom organizacji usług (łańcuchom jakości usług) sieciowych oraz zapewniania jakości usług odpowiadają hierarchie i ciągi implementowanych mechanizmów i technik teleinformatycznych. Hierarchiom i ciągom mechanizmów odpowiadają kryteria oceny jakości usług o strukturze hierarchicznej (np. maksimum opóźnień) i szeregowej (np. suma opóźnień).

Omawiane dwa łańcuchy jakości usług (ang. *QoS chains*) są wprowadzane w celu zapewnienia pożądanej jakości usług w połączeniu pomiędzy aplikacjami, z tym że połączenie takie podzielone jest na trzy części (etapy): pomiędzy aplikacją a siecią, pomiędzy węzłami końcowymi sieci oraz pomiędzy siecią a aplikacją.

Zadaniem hierarchii usług, a tym samym hierarchii jakości usług (ang. *top-to-bottom QoS*), jest zapewnienie nadawanemu i odbieranemu strumieniowi danych odpowiedniej jakości obsługi (np. obsługa priorytetów) na pierwszym i trzecim etapie trasy, tj. w czasie transferu strumienia danych (ruchu informatycznego) od aplikacji do sieci oraz od sieci do aplikacji, odpowiednio w węźle źródłowym oraz w węźle ujściowym o warstwowych architekturach.

Ciąg usług, a tym samym ciąg jakości usług (ang. *end-to-end QoS*), jest realizowany na drugim (środkowym) etapie połączenia pomiędzy aplikacjami i ma na celu zapewnienie jakości usług dla aplikacji w dwojaki sposób: albo tak, że aplikacja zapewnia żadaną przez siebie jakość obsługi bezpośrednio (ang. *explicitly*), albo tak, że sieć zapewnia żadaną przez aplikację jakość obsługi pośrednio (ang. *implicitly*) – w imieniu aplikacji. Jakość obsługi aplikacji realizowana w procesie, na który składają się uzupełniające się hierarchie i ciąg usług, jest determinowana jakością usługi składowej o najniższej jakości.

Hierarchiczna struktura wybranych, niezależnie implementowanych i współpracujących mechanizmów jest stosowana w elementach składowych sieci komputerowej (źródłach, ujściach, węzłach pośrednich, węzłach komutacji itd.), ciągi mechanizmów są natomiast wykorzystywane na wskroś sieci komputerowej. Zadaniem struktur hierarchicznych jest dostarczenie warstwowej kompozycji mechanizmów zapewniających translację wymagań jakościowych aplikacji na wymagania dotyczące jakości obsługi generowanego ruchu, podczas gdy celem struktur szeregowych jest zapewnienie pożądanej jakości obsługi ruchu w czasie jego transferu poprzez sieć. Szeregową strukturą mechanizmów, budowaną w celu zapewnienia założonej jakości obsługi transferu ruchu poprzez sieć, jest zgodna z ogólną ideą organizacji kanałów komunikacyjnych; są budowane z różnych elementów (z dobrze zdefiniowanymi interfejsami) wymagających różnorodnych metod sterowania i zarządzania.

Istotne jest to, że zbiory procedur, zarówno o strukturach hierarchicznych, jak i szeregowych, są zwykle realizowane w heterogenicznych sieciach komputerowych. W sieciach takich, do połączenia współpracujących systemów o różnych architekturach, zlokalizowanych w różnych węzłach sieci stosowane są różne – koncepcyjnie i technologicznie – rozwiązania komunikacyjne i transmisyjne. Oznacza to, że implementacja rozmaitych koncepcji oraz mechanizmów zapewniania jakości w poszczególnych częściach sieci w znaczący sposób zależy od architektur i technologii jej części składowych. Oznacza to dalej, że różne rozwiązania zapewniania jakości usług, dostępne w określonych architekturach i technologiach, nie są przenaszalne (lub są trudno przenaszalne) do rozwiązań sieciowych o innych architekturach i technologiach. W takich przypadkach realizacja określonych koncepcji jakości usług może nie być możliwa wprost w sieci o innej architekturze. Realizacja określonych usług sie-

ciowych, o zadanej jakości, może być związana ze znacznym wzrostem złożoności rozwiązań sieciowych i zwiększeniem ilości niezbędnych zasobów.

1.3.2. KONCEPCJE JAKOŚCI USŁUG W SIECIACH

Potrzeba wspierania rozmaitych wymagań aplikacji dotyczących jakości usług sieci uzasadnia istnienie różnych protokołów umożliwiających dostarczanie zróżnicowanej jakości usług. Różnorodność wymagań aplikacji dotyczących jakości usług jest jednym z powodów wyróżniania dwóch klas systemów jakości usług, dla których charakterystyczne są mechanizmy:

- rezerwacji zasobów (usługi zintegrowane) – dostępne zasoby sieci komputerowej są dzielone pomiędzy aplikacje (i ruch generowany przez nie) żądające usług o określonej jakości; zasoby sieci są rozdzielane pomiędzy źródła ruchu aż do chwili wykorzystania wszystkich dostępnych zasobów (w ramach ograniczeń wynikających z procedur zarządzania przepustowością),

- priorytetowania ruchu (usługi zróżnicowane) – ruch generowany i przekazywany w sieci komputerowej jest klasyfikowany zgodnie z przyjętą (określoną) w sieci hierarchią ruchu; przydział zasobów odbywa się zgodnie z przyjętymi procedurami zarządzania przepustowością, pozwalającymi na preferowanie klas ruchu charakteryzowanych wyższymi wymaganiami co do jakości usług.

Wymienione mechanizmy, a tym samym koncepcje (systemy, polityki) jakości usług, mogą być stosowane w odniesieniu zarówno do pojedynczych strumieni ruchu oraz przepływów, jak i zagregowanych strumieni ruchu oraz przepływów. Podstawą podziału strumieni ruchu lub przepływów w sieci na pojedyncze i zagregowane jest sposób opisu i wartości parametrów opisu ruchu:

- termin pojedynczy odnosi się do jednokierunkowego strumienia danych generowanych przez źródło (strumień ruchu) lub przekazywanych przez element sieci (przepływ), niepodzielnego ze względu na przyjęty sposób jego charakteryzowania, np. z uwagi na stosowany protokół komunikacyjny, adresy źródła i portu źródłowego (aplikacji źródłowej) oraz adresy ujścia i portu ujścia (aplikacji ujściowej),

- w przypadku zagregowanego strumienia ruchu lub przepływu istnieje możliwość wydzielenia części składowych, w opisie których występują te same wartości tych samych parametrów opisu, istotnych z punktu widzenia sterowania i zarządzania, np. przynależność do wyróżnionej klasy ruchu, priorytet obsługi.

Względność pojęć „pojedynczy” lub „zagregowany” wynika z tego, że odpowiednia rozbudowa lub redukcja opisu danego strumienia, na potrzeby sterowania i zarządzania ruchem, może prowadzić do podziału lub scalenia strumieni ruchu.

Zastosowanie określonej koncepcji realizacji jakości usług zależy od wielu czynników: w szczególności od obsługiwanych przez sieć komputerową aplikacji i architektury sieci komputerowej. Koncepcje te mogą być implementowane jako uzupełniające się lub zamienne na różnym poziomie granulacji ruchu generowanego i obsługiwanego w sieci oraz w różnych elementach składowych sieci.

Mechanizmy rezerwacji zasobów i priorytetowania ruchu, będące podstawą wyróżniania systemów jakości usług, są związane z dwiema kategoriami sterowania ruchem: prewencyjny (ang. *preventive control*) i reakcyjny (ang. *reactive control*). W systemach jakości usług opartych na mechanizmach rezerwacji zasobów stosowane są długoterminowe procedury sterowania napływem ruchu, zawierające metody sterowania zarówno prewencyjnego, jak i reakcyjnego. W systemach jakości usług opartych na priorytetowaniu ruchu główne zastosowanie mają krótkoterminowe mechanizmy kształtowania ruchu, które zalicza się do metod sterowania reakcyjnego.

1.3.3. STEROWANIE RUCHEM W SIECIACH

Zadaniem sterowania ruchem w sieciach teleinformatycznych (zwłaszcza w sieciach z komutacją jednostek danych) jest efektywne zarządzanie wykorzystaniem zasobów sieciowych w celu zabezpieczenia sieci przed przeciążeniami i blokadami. Ponieważ zasoby sieci przydzielane są jednocześnie dla wielu strumieni ruchu, konieczne jest więc zarządzanie stopniem wykorzystania zasobów i sterowanie ruchem tak, aby działanie sieci było możliwe także wtedy, gdy żądane zapotrzebowanie na usługi sieci jest większe od aktualnych możliwości sieci.

Procedury przeciwdziałania przeciążeniom i sterowania ruchem są stosowane zarówno w sieciach wykorzystujących technikę komutacji łączy, jak i technikę komutacji jednostek danych (wiadomości, pakietów, ramek, komórek).

W sieciach z komutacją łączy (ang. *circuit-switched*) niezbędne procedury sterowania napływem ruchu są względnie proste. Przydział wymaganej i ustalonej ilości zasobów (zwłaszcza pojemności) na czas trwania połączenia odbywa się jednorazowo i poprzedza decyzję o przyjęciu żądania połączenia (wywołania) do obsługi. Rezerwacja zasobów wyklucza rywalizację o zasoby.

W przypadku sieci z komutacją jednostek danych (ang. *packet-switched*) sterowanie ruchem jest zadaniem znacznie bardziej złożonym ze względu na losowy charakter napływającego ruchu i rywalizację o dostęp do zasobów. Oznacza to konieczność implementacji mechanizmów optymalizacji stopnia wykorzystania zasobów sieci poprzez sterowanie napływającym do sieci ruchem i przepływami w sieci. Celem sterowania jest efektywne wykorzystanie zasobów sieci oraz zapobieganie jej przeciążeniom i blokadom, których powstanie prowadzi do degradacji jakości usług.

W sieciach pakietowych wyróżnia się dwa rodzaje sterowania ruchem: sterowanie przepływem (ang. *flow control*) i przeciwdziałanie przeciążeniom (ang. *congestion control*). Obydwa rodzaje sterowania są traktowane i stosowane jako wzajemnie się uzupełniające.

Sterowanie przepływem dotyczy zadań regulacji szybkości nadawania (szybkości, z jaką źródło transmituje jednostki danych) w celu jej dopasowania do możliwości obsługi w sieci i szybkości odbioru jednostek danych w ujściu. W większości praktycznych realizacji do sterowania przepływem stosuje się różne odmiany metody okna

(ang. *window method*) lub metody kredytów (ang. *credit*), implementowane na różnych poziomach agregacji ruchu (warstwa transportowa, warstwa sieciowa, trasa pomiędzy węzłami źródłowym i ujściowym, łącze danych). Metody sterowania przepływem są adresowane do przepływów (ang. *flow-oriented*) lub do połączeń (ang. *connection-oriented*) i w węzłach sieci stosuje się je niezależnie w celu sterowania komutacją (przełączaniem) ruchu na wybranych poziomach granulacji ruchu.

Zadaniem mechanizmów przeciwdziałania przeciążeniom jest zabezpieczenie sieci lub jej elementów przed nadmiernym ruchem, tzn. utrzymywanie liczby jednostek danych w sieci i (lub) wybranych jej elementach na poziomie, którego przekroczenie powoduje degradację jakości usług. Mechanizmy sterowania przeciążeniami są z reguły adresowane do sieci (ang. *network-oriented*), tzn. są stosowane na potrzeby zarządzania ruchem w danej sieci, traktowanej jako całość.

1.4. PODSUMOWANIE

Architektura i usługi sieci teleinformatycznej są definiowane za pomocą aplikacji wspieranych przez sieć teleinformatyczną. Nadrzędność systemu informacyjnego (aplikacji) nad systemem informatycznym (w tym sieci teleinformatycznej) oznacza, że jakość usług sieci teleinformatycznej jest weryfikowana na poziomie aplikacji. Współistnienie wielu aplikacji o różnych wymaganiach jakościowych i ilościowych, wspieranych przez jedną sieć teleinformatyczną, powoduje, że jednakowy sposób obsługi całości generowanego ruchu jest nie do przyjęcia. Taki sposób obsługi wymaga gwarancji jakości usług dla tzw. najgorszego przypadku, tzn. jakość usług sieci musi być jakością usług dla najbardziej wymagającej klasy ruchu. Takie rozwiązanie, jeżeli w ogóle jest możliwe, wyróżnia się niewielką efektywnością wykorzystania zasobów sieci.

Zróznicowane wymagania różnych aplikacji pod adresem tej samej sieci teleinformatycznej mogą być spełniane albo przez wydzielanie zasobów i kierowanie ich do obsługi ruchu generowanego przez poszczególne aplikacje, albo przez współdzielenie zasobów, albo przez różnicowanie sposobów obsługi różnych klas ruchu. Pierwsze z wymienionych rozwiązań charakteryzuje względnie mała efektywność wykorzystania zasobów i ograniczona skalowalność, drugie natomiast cechuje konieczność wbudowania złożonych mechanizmów i procedur różnicowania, klasyfikacji i monitorowania ruchu.

LITERATURA

- [1] ADAS A., *Traffic models in broadband telecommunications networks*, IEEE Communications Magazine, July 1997 s. 82–89.
- [2] AKIMURU H., KAWASHIMA K., *Teletraffic theory and applications*, Springer Verlag, 1993.

- [3] ATKINS J., NORRIS M., *Total Area Networking*, Wiley, Chichester 1995.
- [4] BAUER F.L., GOOS G., *Informatyka*, WNT, Warszawa 1977.
- [5] BAZEWICZ M. (red.), *Własności i funkcje sieci komputerowych*, Wydawnictwo Politechniki Wrocławskiej, Wrocław 1979.
- [6] BEM D., *Sieci metropolitarne na przykładzie Wrocławskiej Akademickiej Sieci Komputerowej*, materiały konferencyjne „Komputerowe wspomaganie badań naukowych”, Wrocław 1994, s. 11–33.
- [7] BERTSEKAS D., GALLAGER R., *Data networks*, Prentice-Hall International, Englewood Cliffs 1987.
- [8] BONATTI M., CASALI F., POPPLE G., *Integrated broadband communications: network and engineering aspects*, North-Holland, Amsterdam 1991.
- [9] BROEN A.W., JEAGER K., *The future of Enterprise Application Development with components and patterns*, Components & Partners White Paper, 2000.
- [10] BUBNICKI Z., *Podstawy informatycznych systemów zarządzania*, Wydawnictwo Politechniki Wrocławskiej, Wrocław 1993.
- [11] BURAKOWSKI W., *Sieć B-ISDN ATM*, Przegląd Telekomunikacyjny, nr 5, 1993.
- [12] BUTRIMENKO A.W., *Projektowanie i eksploatacja sieci komputerowych*, PWN, Warszawa 1983.
- [13] CAMPBELL A., COULSON G., HUTCHISON D., *A Quality of Service architecture*, Lancaster University (www.itpapers.com).
- [14] CAMPBELL A., COULSON G., GARCIA F., HUTCHISON D., LEOPOLD H., *Integrated quality of service for multimedia communications*, Lancaster University (www.itpapers.com).
- [15] CHAO H.J., GUO X., *Quality of Service control in High-Speed Networks*, Wiley, 2002.
- [16] CYPSEK R.J., *Communications for co-operating systems OSI, SNA and TCP/IP*, Addison-Wesley, Reading 1991.
- [17] DOVROLIS C., RAMANATHAN P., *A case for Relative Differentiated Services and the Proportional Differentiation Model*, IEEE Network, September/October 1999, s. 26–34.
- [18] DAVIES D.W., BARBER D.L.A., *Sieci teleinformatyczne*, WNT, Warszawa 1979.
- [19] DeWITT T., GROSS T., LOWEKAMP B., MILLER N., STEENKISTE P., SUBHLOK J., SUTHERLAND D., *ReMoS: A resource monitoring system for network-aware applications*, CMU-CS-97-194, Carnegie Mellon University, December 1997.
- [20] ELBAUM R., SIDI M., *Topological design of local-area networks using genetic algorithms*, IEEE Transactions on Networking, Vol. 4, No. 5, October 1996, s. 766–778.
- [21] ENCK J., BECKMAN M., *LAN to WAN interconnections*, McGraw-Hill, New York 1994.
- [22] FILIPIAK J., *Real time network management*, North-Holland, Amsterdam 1991.
- [23] FILIPIAK J. (red.), *Sieci dostępowe dla usług szerokopasmowych*, t. I, Wydawnictwo Fundacji Postępu Telekomunikacji, Kraków 1997.
- [24] FLOD J.E., *Telecommunications, switching, traffic and networks*, Prentice Hall, New York 1995.
- [25] FRANK H., FRISCH I.T., *Communication, transmission and transportation networks*, Addison-Wesley, Reading 1971.
- [26] FROST V.S., MELAMED B., *Traffic modelling for telecommunications networks*, IEEE Communications Magazine, Vol. 32, No. 3, March 1994, s. 70–81.
- [27] GLOVER I. A., GRANT P. M., *Digital Communications*, Prentice Hall, London 1998.

- [28] GRIFFITHS J.M., *ISDN explained*, Wiley, Chichester 1992.
- [29] GRZECH A., *Problemy wyboru podsystemów komunikacyjnych lokalnych sieci komputerowych*, seria Monografie, nr 5, Wydawnictwo Politechniki Wrocławskiej, Wrocław 1989.
- [30] GRZECH A., *Integrated communication subsystem of local area network*, Proceedings of the Tenth European Meeting on Cybernetics and Systems Research, World Scientific, Singapore 1990, s. 717–724.
- [31] GRZECH A., *Zbiór rozłączonych prostych podsystemów komunikacyjnych jako złożony podsystem komunikacyjny lokalnej sieci komputerowej*, Archiwum Automatyki i Telemechaniki, t. 35, z. 3, 4, 1990, s. 277–296.
- [32] GRZECH A., *Procedures of designing enterprise network communication subsystems*, Cybernetics and Systems, Vol. 31, No. 5, 2000, s. 531–545.
- [33] GUERIN R.A., PERIS V., *Quality-of-Service in packet networks basic mechanisms and directions*, Computer Networks, Vol. 31, No. 3, February 1999, s. 169–179.
- [34] HAYKIN S., *Systemy telekomunikacyjne*, WKŁ, Warszawa 1998.
- [35] HALSALL F., *Data communications, computer networks and open systems*, Addison-Wesley, Wokingham 1989.
- [36] HALSALL F., *Multimedia communications: applications, networks, protocols and standards*, Addison-Wesley Publishing, 2000.
- [37] HEGERING H.G., ABECK S., *Integrated network and system management*, Addison-Wesley Publishing, Wokingham 1994.
- [38] HUGHES L.J., *Internet security techniques*, New Riders Publishing, Indianapolis 1995.
- [39] INOSE H., *Digital integrated communication systems*, University of Tokyo Press, Tokyo 1979.
- [40] JACKSON M.C., *Systems approaches to management*, Kluwer Academic/Plenum Publishers, New York 2000.
- [41] JAJSZCZYK A., *Wstęp do telekomunikacji*, WNT, Warszawa 2000.
- [42] KASPRZAK A., *Rozległe sieci komputerowe z komutacją pakietów*, Oficyna Wydawnicza Politechniki Wrocławskiej, Wrocław 1997.
- [43] KHASNABISH B, SARACCO R., *Intranets: Technologies, services and management*, IEEE Communication Magazine, October 1997, s. 84–91.
- [44] KLEIN M.R., METHLIE L.B., *Knowledge-based support systems*, Wiley, Chichester 1995.
- [45] KLEINROCK L., *Communication nets*, Volume I: *Stochastic message flow and delay*, Volume II: *Computer Applications*, Wiley, New York 1976.
- [46] KUMMAR V.P., LAKSHMAN T.V., STILIADIS D., *Beyond Best Effort: router architectures for the differentiated services of tomorrow's Internet*, IEEE Communications Magazine, May 1988, s. 152–164.
- [47] LAKSHMAN T.V., MADHOW U., *The performance of TCP/IP for networks with highbandwidth-delay products and random loss*, IEEE/ACM Transactions on Networking, Vol. 5, No. 3, June 1977, s. 336–350.
- [48] LEE B.G., KANG M., LEE J., *Broadband telecommunications technology*, Artech House, Boston 1993.
- [49] LORENZ D.H., *QoS routing in networks with uncertain parameters*, University of Haifa, 1998.

- [50] LOSHIN P., *Extranet design and implementation*, Network Press, San Francisco 1997.
- [51] LEUE S., OECHSLIN P.A., *On parallelizing and optimising the implementation of communication protocols*, IEEE Transactions on Networking, Vol. 4, No. 1, February 1996, s. 55–70.
- [52] MIDWINTER J.E., *Światłowody telekomunikacyjne*, WNT, Warszawa 1983.
- [53] MULLER N.J., DAVIDSON R.P., *LANs to WANs*, Artech House, Boston 1990.
- [54] ONVURAL R.O., *Asynchronous Transfer Mode; Performance issues*, Artech House, Boston 1995.
- [55] PACH A. (red.), *Sieci dostępne dla usług szerokopasmowych*, t. II, Wydawnictwo Fundacji Postępu Telekomunikacji, Kraków 1997.
- [56] PACH A., LASOŃ A., WAJDA K., *Współpraca sieci ATM z innymi systemami telekomunikacyjnymi*, Wydawnictwo Fundacji Postępu Telekomunikacji, Kraków 1995.
- [57] PAPIR Z. (red.), *Sieci z komutacją pakietów – od X.25 do Frame Relay i ATM*, Wydawnictwo Fundacji Postępu Telekomunikacji, Kraków 1997.
- [58] PAPIR Z. (red.), *Sieci dostępne dla usług szerokopasmowych*, t. III, Wydawnictwo Fundacji Postępu Telekomunikacji, Kraków 1997.
- [59] PAPIR Z., *Ruch telekomunikacyjny i przeciążenia sieci pakietowych*, WKŁ, Warszawa 2001.
- [60] PAXSON V., *End-to-end Internet packet dynamics*, IEEE/ACM Transactions on Networking, Vol. 7, No. 3, June 1999, s. 277–292.
- [61] POJOLLE G., SERET D., DROMARD D., HORLAIT E., *Integrated digital communications networks*, Wiley, Chichester 1988.
- [62] PRAHALAD C.K., KRISHNAN M.S., *The new meaning of quality in the information age*, Harvard Business Review, September 1999.
- [63] RODEN M.S., *Analog and digital communication systems*, Prentice Hall, New Jersey 1996.
- [64] SALTZER J.H., REED D.P., CLARK D.D., *End-to-end arguments in system design*, 2nd International Conference on Distributed Computing Systems, Paris, April 1981, s. 509–512.
- [65] SEIDLER J., *Nauka o informacji*, WNT, Warszawa 1983.
- [66] SEIDLER J., *Analiza i synteza sieci łączności dla systemów teleinformatycznych*, PWN, Warszawa 1979.
- [67] SIEGEL E., PASSMORE L. D., *Network Quality of Service – What Good Enough?*, July 1997 (www.netrefence.com).
- [68] SMITH M.D., BAILEY J., BRYNJOLFSSON E., *Understanding digital markets: review and assessment*, MIT Press, 1999.
- [69] SOBCZAK W., MALINA W., *Metody selekcji informacji*, WNT, Warszawa 1978.
- [70] SOBCZAK W. (red.), *Wprowadzenie do teleinformatyki*, WKŁ, Warszawa 1979.
- [71] SOBCZAK W. (red.), *Problemy teleinformatyki*, WKŁ, Warszawa 1984.
- [72] SPOHN D.L., *Data Network Design*, McGraw-Hill on Computer Communications, New York, 1997.
- [73] STALLINGS W., *Data and computer communications*, Prentice-Hall, Upper Saddle River 1997.
- [74] STALLINGS W., *High-speed networks: TCP-IP and ATM design principle*, Prentice-Hall, New York 1998.

-
- [75] STOKŁOSA J., BILSKI T., PANKOWSKI T., *Bezpieczeństwo danych w systemach informatycznych*, PWN, Warszawa 2001.
- [76] TADEUSIEWICZ R., *Sieci neuronowe*, Akademicka Oficyna Wydawnicza PLJ, Warszawa 1998.
- [77] TAKAGI H. (red.), *Stochastic analysis of computer and communication systems*, North-Holland, Amsterdam 1990.
- [78] TANNENBAUM A.S., *Computer networks*, Englewood Cliffs 1989.
- [79] TABATABAE V., GEORGIADIS L., TASSIULAS L., *QoS provisioning and tracking fluid policies in input queueing switches*, IEEE/ACM Transactions on Networking, Vol. 9, No. 5, October 2001, s. 605–617.
- [80] TIEMNIKOW F.E., AFONIN W.A., DMITRIJEW, *Podstawy techniki informacyjnej*, WNT, Warszawa 1974.
- [81] TOMASI W., *Electronic Communications Systems; Fundamentals Through Advanced*, New Jersey 1998.
- [82] TURSki W.M., *Propedeutyka informatyki*, PWN, Warszawa 1985.
- [83] WAJDA K., *Sieci szerokopasmowe*, Wydawnictwo Fundacji Postępu Telekomunikacji, Kraków 1995.
- [84] WAJDA K. (red.), *Budowa sieci komputerowych w technologii ATM*, Wydawnictwo Fundacji Postępu Telekomunikacji, Kraków 1996.
- [85] WAJDA K. (red.), *Wybrane zagadnienia budowy i eksploatacji sieci korporacyjnych*, Wydawnictwo Fundacji Postępu Telekomunikacji, Kraków 1999.
- [86] WANG Z., *Internet QoS: Architectures and mechanisms for Quality of Service*, Academic Press, London 2001.
- [87] WĘGRZYN S., *Podstawy informatyki*, PWN, Warszawa 1982.
- [88] WILSON R.M., *The quantitative impact of survivable network architectures on service availability*, IEEE Communications Magazine, May 1998, s. 122–126.
- [89] WINCH R.G., *Telecommunication Transmission Systems*, McGraw-Hill Telecommunications, New York 1998.
- [90] WOLISZ A., *Podstawy lokalnych sieci komputerowych*, WNT, Warszawa 1992.
- [91] WOŹNIAK J., *Analiza i projektowanie protokołów komunikacyjnych dla radiowych sieci teleinformatycznych*, Wydawnictwo PG, Gdańsk 1991.
- [92] WOŹNIAK J., NOWICKI K., *Sieci LAN, MAN i WAN – protokoły komunikacyjne*, Wydawnictwo Fundacji Postępu Telekomunikacji, Kraków 1998.

2. MIARY JAKOŚCI USŁUG SIECI TELEINFORMATYCZNYCH

Analiza wymagań jest pierwszym i najważniejszym etapem projektowania sieci teleinformatycznej. Dwa istotne punkty widzenia na wymagania są punktami widzenia użytkownika i projektanta (operatora), których efektem są dwa, często odmienne, zbiory wymagań i ograniczeń. Uwzględnienie tych zbiorów jest warunkiem budowy rozwiązań teleinformatycznych, spełniających wymagania użytkownika oraz uwzględniających różne ograniczenia technologiczne, techniczne i organizacyjne.

Wymagania i ograniczenia są formułowane w różny sposób. Względnie łatwe jest sformułowanie wymagań ilościowych. Sformułowanie wymagań jakościowych jest trudniejsze, zwykle bowiem wiąże się z koniecznością znajomości możliwości i ograniczeń różnych technologii oraz technik informatycznych i telekomunikacyjnych. Konsekwencją formułowania wymagań ilościowych i jakościowych jest wbudowanie do sieci narzędzi pozwalających wymuszać spełnienie wymagań oraz monitorować zmiany wartości parametrów opisujących te wymagania, tzn. zmiany wartości miar jakości.

Zwiększenie liczby i rodzaju aplikacji jednocześnie obsługiwanych w sieci oraz wzrost złożoności sieci związane są m.in. ze zwiększeniem liczby miar jakości stosowanych przez użytkowników sieci i dostawców usług do formułowania wymagań i ograniczeń dotyczących działania sieci oraz pomiaru stopnia spełniania przez nie wymagań ilościowych i jakościowych.

2.1. KLASY RUCHU

Podstawowe kryterium podziału ruchu generowanego przez użytkowników sieci komputerowych to:

- wrażliwość ruchu na opóźnienie,
- wrażliwość ruchu na straty

wnoszone przez system obsługi ruchu jakim jest sieć komputerowa. Z tego punktu widzenia aplikacje obsługiwane w sieci i ruch generowany przez aplikacje sieciowe dzieli się na dwie klasy:

- aplikacje wrażliwe na obsługę w czasie rzeczywistym (ang. *real-time sensitive*),
- aplikacje tolerujące opóźnienia (ang. *delay tolerant*).

Ze względu na istnienie jednoznacznej zależności pomiędzy aplikacjami obsługiwanymi przez sieci komputerowe a ruchem generowanym przez te aplikacje, wprowadzenie podziału ruchu na klasy jest tożsame z wyróżnieniem klas aplikacji i odwrotnie.

Ruch wrażliwy na obsługę w czasie rzeczywistym jest generowany przez grupę aplikacji, charakteryzujących się rygorystycznymi wymaganiami dotyczącymi opóźnień i – w większości przypadków – tolerowaniem strat. Typową konsekwencją górnego ograniczenia czasu dostarczenia jednostki danych (opóźnienie jednostki danych) jest gwałtowny spadek przydatności przekazywanych (obsługiwanych) jednostek danych dla aplikacji po przekroczeniu zadanego opóźnienia. Przykładem aplikacji generujących ruch wrażliwy na opóźnienia są m.in. systemy przekazu głosu i obrazu (np. systemy wideokonferencji), systemy automatyki itp. obsługiwane przez sieci z komutacją jednostek danych.

Ruch tolerujący opóźnienia zwykle jest generowany przez aplikacje charakteryzujące się istnieniem rygorystycznych wymagań dotyczących bezbłędności i – w większości przypadków – tolerowaniem opóźnień. Tolerowanie opóźnień nie należy rozumieć jako akceptację dowolnych opóźnień wnoszonych przez sieć. Tolerowanie opóźnień przez aplikację należy rozumieć jako akceptację zasady, zgodnie z którą być może nieuniknionym kosztem bezbłędnej obsługi ruchu jest powstawanie różnych opóźnień różnych jednostek danych. Przykładem aplikacji generujących ruch tolerujący opóźnienia są m.in. systemy przetwarzania danych.

2.2. MIARY JAKOŚCI USŁUG W SIECIACH KOMPUTEROWYCH

Miary jakości usług (ang. *QoS metrics*) w sieciach komputerowych dzieli się na dwie klasy:

- miary jakości połączeń (ang. *call control parameters*),
- miary jakości transferu (ang. *information transfer parameters*).

Pierwsza z wymienionych klas miar jakości ma zastosowanie zarówno w sieciach, w których transfer ruchu generowanego przez źródła jest poprzedzany ustanawianiem fizycznego lub logicznego połączenia pomiędzy źródłem i ujściem, tzn. w sieciach z komutacją łączy (ang. *circuit-switched*), jak i w sieciach z komutacją wiadomości (ogólnie jednostek danych) zorientowanych połączeniowo (ang. *connection-oriented*).

Miary jakości zaliczane do drugiej z wymienionych klas są stosowane w różnych sieciach, niezależnie od sposobów udostępniania ich zasobów do obsługi generowanego ruchu, tzn. zarówno w sieciach zorientowanych połączeniowo, jak i w sieciach datagramowych. Różne miary jakości transferu stosuje się w zależności od klasy transferowanego w sieci ruchu. Na ogół przyjmuje się podział miar jakości na miary

jakości obsługi ruchu wrażliwego na straty (ang. *loss-sensitive*) i ruchu wrażliwego na opóźnienia (ang. *delay-sensitive*).

2.2.1. MIARY JAKOŚCI POŁĄCZEŃ

Naturalnymi miarami jakości połączeń w sieciach zorientowanych połączeniowo są wartości parametrów charakteryzujących ustanawianie, utrzymywanie i znoszenie połączeń oraz dostępność połączeń dla żądań ich ustanowienia.

2.2.1.1. CZAS USTANOWIENIA POŁĄCZENIA

Czas (opóźnienie) ustanowienia połączenia (ang. *setup connection delay*) jest długością przedziału czasu upływającego od chwili wysłania wiadomości z żądaniem ustanowienia połączenia do chwili otrzymania potwierdzenia połączenia. Wartość tego czasu zależy od czasów przetwarzania żądania w węzłach (punktach) sieci.

Przykładem wartości czasu ustanawiania połączenia jest wartość definiowana w standardzie N-ISDN (ang. *Narrowband Integrated Services Digital Network*) dla szybkości 64 kbps na odległość nie większą niż 27 500 km. W tym przypadku wartość średnia czasu ustanawiania połączenia powinna być mniejsza niż 4,5 s, a 95% wartości opóźnień powinno być mniejszych niż 8,35 s.

2.2.1.2. CZAS ROZŁĄCZANIA POŁĄCZENIA

Czas (opóźnienie) rozłączania połączenia (ang. *connection release delay*) jest długością przedziału czasu upływającego pomiędzy chwilą generacji żądania rozłączenia połączenia i chwilą otrzymania potwierdzenia rozłączenia. W tym przypadku czasy przetwarzania w węzłach (punktach) sieci nie mają wpływu na opóźnienie.

Przykładem wartości czasu rozłączania połączenia jest jego wartość definiowana w standardzie ISDN. Wartość średnia opóźnienia rozłączania powinna być mniejsza niż 300 ms, a 95% wartości opóźnień powinno być mniejszych niż 850 ms.

2.2.1.3. PRAWDOPODOBIENSTWO AKCEPTACJI POŁĄCZENIA

Prawdopodobieństwo akceptacji połączenia (ang. *connection acceptance probability*) definiuje się jako iloraz liczby ustanowionych połączeń i wszystkich żądań ich ustanowienia w dostatecznie długim przedziale czasu. Wartość tego parametru może być stosowana do szacowania prawdopodobieństwa blokady (ang. *blocking probability*) żądań połączeń.

Prawdopodobieństwo blokady jest jedną z prostszych i ważniejszych miar stosowanych w zadaniach wymiarowania sieci, tzn. w zadaniach projektowania i alokacji zasobów sieci. Jedną z częściej używanych wersji tej miary jakości sieci z komutacją

łączy i komutacją pakietów jest formuła B-Erlanga, w której prawdopodobieństwo blokady wyznacza się dla wykładniczego modelu ruchu i wykładniczego modelu obsługi (modelu $M/M/n$).

Wyznaczanie prawdopodobieństwa blokady w sieciach z integracją usług, tzn. sieciach B-ISDN (ang. *Broadband Integrated Services Digital Network*) – ze względu na współistnienie różnych klas ruchu, złożone charakterystyki ruchu w poszczególnych klasach i ograniczone możliwości predykcji – jest zadaniem trudnym.

2.2.2. MIARY JAKOŚCI TRANSFERU

Do charakteryzowania wymagań użytkowników, opisu usług dostarczanych przez sieci, podejmowania decyzji o akceptacji lub odrzuceniu żądań dostępu, podejmowania decyzji o rozdziale zasobów itp. stosuje się różne miary jakości transferu.

Nie wszystkie możliwe miary jakości transferu są stosowane i jednakowo przydatne do formułowania wymagań wszystkich możliwych aplikacji i szacowania jakości usług dostarczanych przez różne sieci.

Od zbiorów miar jakości, definiowanych dla poszczególnych aplikacji i sieci, wymaga się: ich dopasowania do specyfiki aplikacji i sieci oraz zdolności do transformacji odpowiednich charakterystyk aplikacji na odpowiednie charakterystyki sieci i odwrotnie. Miary jakości mogą mieć różny zakres zastosowania i inną interpretację. Zastosowanie i interpretacja takiej samej miary jakości transferu zależą od aplikacji, klas ruchu generowanego przez te aplikacje oraz typu sieci obsługującej generowane klasy ruchu.

Przykładem miary jakości, której przydatność zależy od klasy aplikacji, jest prawdopodobieństwo strat jednostek danych w czasie ich transmisji w sieci. Omawiana miara jakości jest stosowana do aplikacji niewrażliwych na straty, tj. dopuszczających straty generowanych jednostek danych i wrażliwych na opóźnienia (np. transmisja głosu), nie jest natomiast przydatna do charakteryzowania aplikacji niewrażliwych na opóźnienia i wrażliwych na straty (np. transmisja danych). Innym przykładem, rozmaicie interpretowanej miary jakości, jest opóźnienie jednostek danych – gdy jest większe od założonej wartości, może być interpretowane jako błąd w obsłudze aplikacji czasu rzeczywistego, nie jest natomiast częścią definicji błędu w obsłudze aplikacji niewrażliwych na opóźnienia.

Definiowanie zbioru miar jakości transferu dla danej aplikacji i danej sieci nie jest zadaniem trywialnym. Wymagania stawiane definiowanym zbiorom miar jakości są następujące – miary jakości powinny być:

- dopasowane do aplikacji (tzn. gwarantować możliwość charakteryzacji klasy aplikacji i dokładnego opisu jej wymagań),
- dopasowane do procedur i mechanizmów stosowanych w sieci (tzn. gwarantować możliwość opisu tych procedur i mechanizmów na potrzeby szacowania ich wpływu na jakość obsługi ruchu w sieci),

- skalowalne przestrzennie (tzn. gwarantować możliwość opisu zarówno lokalnych, jak i globalnych charakterystyk sieci, bez konieczności specyfikacji architektury sieci i jej fragmentów),
- elastyczne (tzn. gwarantować możliwość uzyskania różnej precyzji i dokładności dla różnej złożoności obliczeniowej).

2.2.2.1. STOPA BŁĘDÓW

Elementarną stopę błędów BER (ang. *Bit Error Ratio*) definiuje się jako wartość średnią ilorazu liczby błędów elementarnych w polu informacyjnym jednostki danych i całkowitej liczby bitów w nim transmitowanych. Wartość tej miary jakości w decydującym stopniu zależy od jakości systemu transmisyjnego i stosowanego medium transmisyjnego – może się znacznie różnić np. dla kabla koncentrycznego i światłowodu.

Ze względu na to, że nie ma systemu transmisyjnego gwarantującego bezbłędną transmisję, stopa błędów jest zawsze różna od zera. Wzrost jakości systemów transmisyjnych prowadzi do obniżenia stopy błędów oraz redukcji skali i złożoności metod kompensacji błędów.

Stosowane metody kompensacji błędów zależą od wrażliwości na błędy aplikacji obsługiwanych przez dany system transmisyjny oraz architektury sieci. Na ogół stosowane są trzy klasy rozwiązań:

- detekcja i korekcja błędów są realizowane przez protokół łącza transmisji danych, tzn. w protokole warstwy transmisji danych implementowane są mechanizmy detekcji i korekcji błędów; protokół taki gwarantuje bezbłędną transmisję,
- detekcja i korekcja błędów są realizowane w jednej z warstw nadrzędnych warstwy transmisji danych, tzn. w protokole warstwy transmisji danych nie ma mechanizmów detekcji i kompensacji błędów; protokół warstwy transmisji danych nie gwarantuje bezbłędnej transmisji, a gwarancje bezbłędnego transferu danych realizowane są w jednej z warstw wyższych,
- błędy transmisji nie są wykrywane i korygowane ani w warstwie transmisji danych, ani w żadnej innej warstwie architektury sieci; zadania detekcji i korekcji błędów są zadaniami użytkownika sieci.

2.2.2.2. WSPÓŁCZYNNIK STRAT JEDNOSTEK DANYCH

Współczynnik strat jednostek danych (ang. *loss ratio*) definiuje się jako wartość średnią ilorazu liczby jednostek danych traconych w sieci i liczby wszystkich jednostek wysyłanych przez źródło (w ramach połączenia, przepływu, w sieci itd.). Omawiana miara jakości ma różne zastosowanie w zależności od architektury sieci; im mniejsza granulacja jednostek danych transferowanych w sieci, tym większy jest zakres jej stosowania. Przykładem są sieci ATM (ang. *Asynchronous Transfer Mode*), w których względnie mała długość komórki ATM i połączeniowy tryb transmisji w tej

sieci powodują, że stopa strat komórek CLR (ang. *Cell Loss Rate*) jest specyficzną miarą jakości sieci ATM.

Wyróżnia się dwie przyczyny strat jednostek danych w sieci:

- przepełnienie buforów odbiorczych w węzłach komutacji ruchu, powodujące odrzucanie nadmiarowych jednostek danych,
- występowanie niewykrywalnych i niekorygowalnych błędów elementarnych w nagłówkach jednostek danych, powodujące odrzucanie niepoprawnie odebranych jednostek danych.

Efekty strat jednostek danych i działania podejmowane w razie wystąpienia straty jednostki danych są różne i zależą od klasy aplikacji obsługiwanych w sieci. Na przykład aplikacje wideo są wrażliwe na straty jednostek danych (np. ramek ATM); strata jednostki danych może oznaczać stratę fragmentu obrazu, a w konsekwencji utratę integralności transmisji. W systemach transmisji głosu, tzn. aplikacjach tolerujących straty jednostek danych, dopuszczalny jest pewien poziom strat. Zależy on od warstwy, w której następuje strata, i długości traconych jednostek danych. W przypadku transmisji danych każda utrata jednostki danych wymaga jej retransmisji.

Poziom strat jednostek danych na ogół rośnie wraz ze zwiększeniem szybkości transmisji i komutacji, wymuszając stosowanie rozwiązań zapobiegających stratom lub niwelującym konsekwencje strat dla aplikacji, zwłaszcza wrażliwych na straty.

Ze względu na klasę aplikacji oraz architekturę sieci można wyróżniać mniej lub bardziej korzystne rozkłady strat jednostek danych. Na przykład w transmisji wideo korzystniejsze są pojedyncze straty jednostek danych (np. komórek ATM) od takiej samej liczby straconych jednostek, ale występujących w postaci „paczek”. W tym przypadku zasadne jest stosowanie systemu transmisyjnego, w którym rozkład strat jest równomierny. Jeśli natomiast w systemach transmisji danych występują straty jednostek danych, to korzystniejsze jest ich występowanie w postaci „paczek”, niezależnie od liczby traconych jednostek danych (np. komórek ATM) zawartych w jednej nadrzędnej jednostce danych (np. ramce warstwy łącza danych); nadrzędna jednostka danych będzie w całości retransmitowana. W takim przypadku równomierny rozkład strat jednostek danych nie jest korzystny ze względu na zwiększenie liczby nadrzędnych jednostek danych wymagających korekcy np. przez retransmisję.

2.2.2.3. WSPÓŁCZYNNIK „WTRĄCANIA” JEDNOSTEK DANYCH

Współczynnik wtrącania jednostek danych (ang. *insertion ratio*) definiuje się jako wartość ilorazu liczby dostarczanych jednostek danych do liczby wszystkich wysłanych jednostek danych (w połączeniu, w przepływie, w sieci itd.).

W sieciach, wraz ze zwiększeniem szybkości transmisji i komutacji, rośnie prawdopodobieństwo wystąpienia błędów elementarnych, powodujących niewykrywalne i niekorygowalne zmiany w nagłówkach jednostek danych. Rezultatem takich zmian

jest dostarczanie jednostek danych pod niewłaściwy adres, tzn. do niewłaściwego węzła ujściowego.

Przykładem znaczenia omawianej miary jakości są sieci ATM, gdzie współczynnik „wtrącenia” komórki CIR (ang. *Cell Insertion Rate*) jest specyficzną miarą jakości. Złożona, hierarchiczna struktura organizacji i sterowania przepływem w sieciach ATM powoduje, że błąd w nagłówku komórki ATM może powodować zmianę adresów połączenia wirtualnego komórki, a zatem skierowanie jej na trasę wirtualną i (lub) do kanału wirtualnego, prowadzących do innego, niż pożądane, ujścia.

Inną wyróżnianą przyczyną „wtrącania” jednostek danych jest identyfikacja jednostek danych przepływów na podstawie zredukowanej reprezentacji nagłówka jednostki danych. Konieczność przyspieszenia klasyfikacji (klasyfikacji jednostek danych do wyróżnianych klas ruchu) i identyfikacji jednostek danych (identyfikacji przynależności jednostki danych do określonych przepływów lub klas ruchu) oraz wyboru trasy w węzłach komutacyjnych sieci prowadzi do rozwiązań, w których nagłówki jednostek danych (np. pakietów) są reprezentowane przez krótsze sekwencje bitów. Niejednoznaczność skróconych reprezentacji nagłówków może być przyczyną błędnej identyfikacji jednostek danych i wyboru błędnej trasy ich transferu w sieci, a tym samym powstania zdarzenia polegającego na „wtrąceniu” jednostki danych.

Wykrywanie, zapobieganie i kompensacja błędów, jakimi są „wtrącenia” jednostek danych, są zadaniami trudniejszymi od analogicznych zadań w przypadku błędu polegającego na stracie jednostki danych w sieci.

„Wtrącanie” jednostek danych może być źródłem zakłóceń obsługi aplikacji i powodem degradacji jakości usług sieci.

W zależności od klasy aplikacji odebranie nieoczekiwanej, „wtrąconej” jednostki danych przez ujście może powodować zaburzenia lub utratę synchronizacji terminali. Co więcej, „wtrącanie” jednostek danych, poprzedzane błędnym wyborem połączenia, oznacza, że wzrasta ilość ruchu obsługiwanego przez połączenia inne od połączeń, do których „wtrącana” jednostka danych należy. Jeżeli omawiane inne połączenia są znacznie obciążone, to dodatkowy przepływ w nich może powodować przeciążenie, będące źródłem degradacji jakości usług innych przepływów lub klas ruchu obsługiwanych przez to połączenie.

2.2.2.4. OPÓŹNIENIE TRANSFERU JEDNOSTEK DANYCH

Opóźnienie transferu jednostek danych (ang. *transfer delay*) pomiędzy dwoma punktami (początkowym i końcowym) w sieci definiuje się jako przedział czasu, którego długość jest różnicą czasu pomiędzy chwilą wypływu pierwszego bitu jednostki danych z punktu początkowego i chwilą napływu ostatniego bitu jednostki danych do punktu końcowego. Jeżeli punkty początkowy i końcowy są interfejsami sieci, to opóźnienie wnoszone przez sieć na trasie pomiędzy węzłami źródłowym i ujściowym jest opóźnieniem typu koniec–koniec (ang. *end-to-end*).

Całkowite opóźnienie transferu danych wnoszone przez sieć jest sumą różnych opóźnień składowych, wnoszonych przez różne elementy sieci i wykonywane w tych elementach sieci procedury. Występowanie poszczególnych składowych opóźnień całkowitego zależy zarówno od klasy obsługiwanych aplikacji, jak i architektury sieci.

Opóźnienie kodowania (ang. *coding delay*) jest czasem niezbędnym do przekształcenia sygnału generowanego przez źródło w sygnał przekazywany w sieci. Zgodnie z tym opóźnienie kodowania występuje zarówno wtedy, gdy realizowana jest konwersja źródłowego sygnału analogowego w sygnał cyfrowy, jak i wtedy, gdy realizowane są kodowanie i (lub) kompresja cyfrowego sygnału źródłowego.

Przekształcanie sygnału analogowego w cyfrowy jest charakterystyczne dla różnych zadań obsługi głosu i (lub) obrazu. Konwersja sygnałów występuje także w zastosowaniach sieci w zadaniach metrologii, automatyki itp.

W przypadku przekształcania sygnału analogowego w sygnał cyfrowy wartość opóźnienia kodowania zależy od algorytmów i sprzętowo-programowych sposobów realizacji trzech – charakterystycznych dla konwersji sygnału analogowego w sygnał cyfrowy – procesów: próbkowania, kwantowania i kodowania.

Opóźnienie kodowania wnoszone przez kodowanie detekcyjne i (lub) korekcyjne zależy od stosowanego kodu; miarą tego opóźnienia jest liczba i rozmieszczenie pozycji kontrolnych oraz długość sekwencji bitów stosowanych do wyznaczania wartości pozycji kontrolnych.

Opóźnienie kodowania wnoszone w czasie kompresji sygnału zależy od stosowanego standardu kompresji i jego realizacji; górnym ograniczeniem opóźnienia kodowania jest czas wyznaczania ramki interpolowanej, której przygotowanie wymaga uwzględnienia wcześniejszego i późniejszego obrazu odniesienia.

Opóźnienie kodowania – rosnące wraz z żądaną dokładnością konwersji sygnałów, zdolnością detekcyjną i korekcyjną kodów oraz wartością współczynnika kompresji – decyduje także o opóźnieniu wnoszonym przez odpowiednie procesy dekodowania, realizowane w węzle ujściowym: odtwarzania sygnału analogowego, dekodowania i dekompresji.

Opóźnienie pakietyzacji (ang. *packetization delay*) jest czasem niezbędnym do utworzenia jednostki danych, tj. opóźnieniem powstającym w wyniku konieczności oczekiwania z rozpoczęciem obsługi jednostki danych do chwili jej skompletowania, tzn. do chwili odbioru ostatniego bitu jednostki danych (np. pakietu).

W najprostszym przypadku jest to czas potrzebny na skompletowanie liczby bitów wymaganych do formowania pola danych jednostki danych. Kompletacja jednostki danych, jednoznacznie definiowanej przez protokół danej warstwy, polega na łączeniu lub podziale jednostek danych warstwy nadrzędnej. Gdy utworzenie jednostki danych wymaga dodatkowego przetwarzania sekwencji bitów tworzących pole danych (np. wyliczanie sumy kontrolnej), opóźnienie pakietyzacji jest sumą czasów potrzebnych na zebranie wymaganej liczby bitów i przygotowanie zawartości różnych pól nagłówka jednostki danych.

Opóźnienie propagacji (ang. *propagation delay*) zależy od rodzaju i długości (odległości pomiędzy źródłem i ujściem) użytego medium transmisyjnego.

Opóźnienie transmisji (ang. *transmission delay*) jest czasem transmisji wszystkich bitów jednostki danych. Jest ono zwykle definiowane jako czas potrzebny na skompletowanie jednostki danych lub takiego jej fragmentu, który umożliwia rozpoczęcie przetwarzania jednostki danych.

Czasami opóźnienie transmisji, malejące wraz ze wzrostem szybkości transmisji i zmniejszaniem się odległości pomiędzy węzłami, może być pomijalnie małe. W systemach transmisji satelitarnej względnie duże opóźnienia transmisji nie zależą od odległości pomiędzy węzłami sieci, lecz od wysokości orbity.

W klasycznych rozwiązaniach sieci opartych na zasadzie „zapamiętaj i prześlij dalej” (ang. *store-and-forward*) przetwarzanie jednostki danych jest możliwe pod warunkiem odebrania wszystkich bitów jednostki danych. Opóźnienie transmisji jest czasem transmisji całej jednostki danych. W rozwiązaniach, w których przetwarzanie jednostki danych rozpoczyna się po otrzymaniu jej fragmentu, opóźnienie transmisji zależy od długości fragmentu niezbędnego do rozpoczęcia przetwarzania.

Opóźnienie komutacji (przełączania) (ang. *switching delay*) jest całkowitym czasem przejścia jednostki danych przez przełącznik. Wartość omawianego opóźnienia zależy od architektury i konstrukcji przełącznika, szybkości przełączania oraz długości sekwencji bitów w jednostce danych stosowanych do przełączania jednostek danych.

Na ogół opóźnienie komutacji jest sumą czasów: przygotowania decyzji o wyborze portu wyjściowego przełącznika i transmisji jednostki danych z wejścia do wyjścia przełącznika. Czas transmisji jednostki danych pomiędzy portami zależy od wewnętrznej szybkości transmisji przełącznika oraz stosowanych w przełączniku technik rozstrzygnięcia konfliktów w dostępie do zasobów przełącznika.

Wartości składowych opóźnienia komutacji są wartościami charakterystycznymi przełączników i decydują o ich jakości.

Opóźnienie kolejkowania (ang. *queuing delay*) występuje w rozwiązaniach sieciowych, w węzłach których przekazywane jednostki danych są buforowane. Mogą one być przechowywane w buforach organizowanych na portach wejściowych, wewnątrz węzłów (przełączników) lub na portach wyjściowych. W zależności od rozwiązania do buforowania jednostek danych mogą być stosowane wybrane lub wszystkie z wymienionych sposoby buforowania. Czas oczekiwania na obsługę w węzłach sieci zależy od ilości zasobów udostępnianych w jej węzłach, algorytmów dostępu do zasobów i algorytmów szeregowania, tj. procedur wymuszania dostępu jednostek danych do zasobów (np. strategii obsługi kolejek).

Celem buforowania jednostek danych na portach wejściowych jest wygładzanie zmiennych intensywności napływu strumienia jednostek danych oraz kompensacja różnicy szybkości napływu jednostek danych do węzła i szybkości w nim ich obsługi. Bufory wejściowe stosuje się także do chwilowego kumulowania nadmiarowych, względem ograniczonych zasobów węzłów, ilości napływających jednostek danych.

Buforowanie wewnętrzne jednostek danych w węźle stosuje się do rozwiązywania konfliktów w dostępie do współdzielonych zasobów wewnętrznych węzła, natomiast buforowanie na portach wyjściowych – do rozwiązywania konfliktów w dostępie jednostek danych do współdzielonych wyjść węzła.

Jednoczesna obsługa jednostek danych, należących do różnych klas ruchu, praktycznie dyskwalifikuje stosowanie buforów na portach wejściowych. Efektywna obsługa jednostek danych, należących do różnych klas ruchu, wymaga, aby wszystkie napływające do węzła były obsługiwane na wejściu z szybkością żadaną dla najbardziej wymagającej klasy ruchu. Oznacza to dalej, że przetwarzanie nagłówka jednostki danych w węźle nie może być poprzedzone czasem oczekiwania w kolejce na przetwarzanie, tzn. jednostki danych powinny być obsługiwane z szybkością nie mniejszą niż szybkość napływu do węzła.

Opóźnienie przetwarzania (ang. *processing delay*) jest wnoszone przez implementowane w węzłach sieci różne procedury, których celem jest stwierdzenie poprawności jednostki danych, detekcja i korekcja błędów, sprawdzenie uprawnień do korzystania z zasobów, wymuszenie ich alokacji, wyznaczenie kolejnych węzłów trasy, przygotowanie do wyprowadzenia z sieci itp.

W sieciach dostarczających usługi o żądanej jakości opóźnienie przetwarzania jest konsekwencją stosowania złożonych obliczeniowo procedur klasyfikacji jednostek danych, algorytmów ich identyfikacji, procedur rezerwacji i alokacji zasobów, algorytmów wyznaczania tras, procedur kształtowania (wygładzania) ruchu, algorytmów szeregowania jednostek danych itp.

Opóźnienie odtwarzania (ang. *reassembly delay*) jest sumą czasów niezbędnych do wykonania w węźle ujściowym procesów odwrotnych do procesów pakietyzacji, realizowanych w węźle źródłowym, tj. procesów konwersji sygnałów, łączenia lub podziału jednostek danych, dekodowania, dekompresji itd.

Celem procesów odtwarzania, realizowanego w danej warstwie, jest przetworzenie odebranej jednostki danych do postaci wymaganej w warstwie nadrzędnej lub do wprowadzania danych.

2.2.2.5. MODEL OPÓŹNIENIA PRZEPLYWÓW

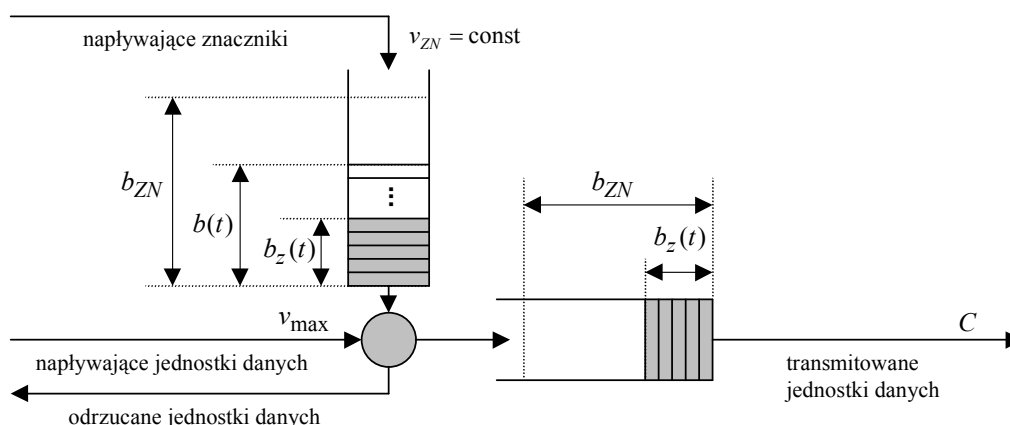
Jednym z prostszych modeli przepływu, pozwalającym na szacowanie wartości opóźnienia jednostek danych (czasu oczekiwania na obsługę) wnoszonego przez kanał transmisyjny, jest model przepływowy (ang. *fluid model*), w którym do sterowania szybkością przepływu (ograniczania szybkości przepływu) stosowany jest algorytm ciekącego wiadra (ang. *leaky bucket*).

Do sterowania przepływem stosuje się różne wersje algorytmu ciekącego wiadra. Algorytm ten w najprostszej wersji, tzn. bez bufora jednostek danych oczekujących na znaczniki, ma dwa parametry: stałą szybkość napływu znaczników (ang. *token arrival rate*) – v_{ZN} ($v_{ZN} = \text{const}$) i pojemność (głębokość) bufora znaczników (ang. *bucket depth*) – b_{ZN} (rysunek 2.1). Oprócz tego warunki przyjmowania jednostek danych

w danej chwili t zależą od dwóch zmiennych wartości: liczby znaczników w buforze znaczników $b(t)$, gdzie $b(t) \leq b_{ZN}$ oraz liczby znaczników zajętych $b_z(t)$, gdzie $b_z(t) \leq b(t)$.

Znaczniki napływają do bufora znaczników ze stałą szybkością v_{ZN} i są pobierane przez nadchodzące jednostki danych (w wersji uproszczonej można przyjąć, że jednostki danych mają jednakową długość i pobierają jeden znacznik). W chwili nadejścia jednostki danych sterownik przepływu, działający zgodnie z omawianym algorytmem, wysyła jednostkę danych do kolejki jednostek oczekujących na transmisję, pod warunkiem, że bufor znaczników nie jest pusty. W chwili zakończenia transmisji jednostki danych sterownik usuwa znacznik z bufora znaczników. W razie nadejścia jednostki danych do sterownika przepływu i braku znacznika w buforze znaczników jednostka danych jest odrzucana.

Pojemność bufora znaczników jest górną granicą liczby znaczników gromadzonych w sterowniku; z chwilą gdy bufor znaczników zostanie wypełniony, sterownik odrzuca nadchodzące znaczniki do chwili, gdy liczba znaczników w buforze jest mniejsza niż jego pojemność.



Rysunek 2.1. Schemat funkcjonalny sterownika przepływu z uproszczonym algorytmem ciekącego wiadra

Algorytm ciekącego wiadra charakteryzują:

- całkowita liczba jednostek danych $n(t)$, które mogą być wysłane przez źródło w przedziale czasu o długości T , jest ograniczona liniowo: $n(T) \leq v_{ZN} + b_{ZN}$,
- szybkość napływu znaczników (v_{ZN}) odpowiada średniej szybkości transferu jednostek danych,
- możliwe jest wysłanie paczki jednostek danych przez źródło, ale maksymalny rozmiar paczki nie może być większy od pojemności bufora znaczników (b_{ZN}),
- jeżeli źródło wysyła jednostki danych z szybkością mniejszą od szybkości nadchodzenia znaczników (v_{ZN}), to liczba znaczników w buforze znaczników rośnie; w granicznym przypadku liczba znaczników w buforze znaczników może być równa jego pojemności (b_{ZN}).

Przy założeniu, że:

- opóźnienia jednostek danych są czasami oczekiwania na obsługę (pozostałe składowe opóźnienia są pomijalnie małe),
- kanał transmisyjny charakteryzowany jest pojemnością C ,
- źródło przepływu charakteryzowane jest szybkością szczytową v_{\max} ,
- algorytm sterowania przepływem jest charakteryzowany wartościami parametrów: szybkością napływu znaczników v_{ZN} , całkowitą pojemnością bufora znaczników (wiadra) b_{ZN} oraz liczbą zajętych znaczników b_z ,
- w chwili rozpoczęcia nadawania jednostek danych liczba znaczników wolnych (dostępnych dla jednostek danych) jest równa pojemności bufora znaczników – liczba jednostek danych ze znacznikami oczekujących na transmisję jest równa zero ($b_z = 0$),

wartość maksymalnego opóźnienia jest określona wyrażeniem:

$$d_{\max} = \begin{cases} \frac{b_{ZN}}{C} & \text{gdy } v_{\max} \rightarrow \infty, \quad C \geq v_{ZN} \\ \frac{[b_{ZN}(v_{\max} - C)]}{[C(v_{\max} - v_{ZN})]} & \text{gdy } v_{\max} > C \geq v_{ZN} \\ 0 & \text{gdy } v_{\max} \leq C \\ \frac{b_{ZN}}{C} & \text{gdy } v_{ZN} > C \end{cases}$$

Interpretacja składników tego wyrażenia jest następująca:

➤ Wartość maksymalnego opóźnienia jednostek danych (d_{\max}) jest czasem obsługi ostatniej jednostki danych w kolejce jednostek mających znacznik i oczekujących na transmisję; liczba jednostek danych oczekujących w kolejce na transmisję nie jest większa od liczby maksymalnej liczby znaczników w buforze znaczników (b_{ZN}).

➤ Jeżeli szybkość szczytowa napływu jednostek danych jest nieograniczona ($v_{\max} \rightarrow \infty$), to źródło praktycznie w nieskończenie krótkim czasie wykorzystuje wszystkie znaczniki znajdujące się w buforze znaczników. W najkorzystniejszym przypadku (w chwili rozpoczęcia napływu jednostek danych bufor znaczników jest wypełniony – zawiera b_{ZN} znaczników) maksymalna długość kolejki jednostek danych oczekujących na transmisję wynosi b_{ZN} , a maksymalne opóźnienie jednostki danych jest opóźnieniem jednostki przejmującego b_{ZN} -ty znacznik, tj. czasem obsługi b_{ZN} jednostek danych z szybkością C równym $d_{\max} = b_{ZN}/C$ (w tym przypadku czas oczekiwania na transmisję zależy od pojemności bufora znaczników). Maksymalny rozmiar paczki jednostek danych przyjmowanej do obsługi jest równy pojemności bufora znaczników b_{ZN} .

➤ Jeżeli szybkość szczytowa napływu jednostek danych jest porównywalna z szybkością napływu znaczników, a szybkość transmisji jest większa od szybkości

napływu znaczników ($v_{\max} > C \geq v_{ZN}$), to w najgorszym razie maksymalne opóźnienie ostatniej jednostki danych, przyjmującej ostatni znacznik i zajmującego ostatnie miejsce w kolejce jednostek danych oczekujących na transmisję, jest pomniejszone (w porównaniu z opóźnieniem gdy $v_{\max} \rightarrow \infty$ i $C \geq v_{ZN}$). Pomniejszenie to wynika z tego, że w czasie zajmowania ostatniego miejsca w kolejce jedna z jednostek danych jest w trakcie transmisji. Liczba znaczników (przy założeniu, że w chwili rozpoczęcia napływu jednostek danych bufor znaczników jest wypełniony) wykorzystanych przez źródło ruchu (aktualnie zajmowanych i zwolnionych przez sterownik) do chwili zajęcia liczby znaczników równej pojemności bufora znaczników wynosi $b_{ZN}(v_{\max} - C)/(v_{\max} - v_{ZN})$ i jest większa od pojemności bufora znaczników b_{ZN} . W czasie potrzebnym na przejście jednego znacznika obsługiwanego jest $(v_{\max} - v_{ZN})/(v_{\max} - C)$ jednostek danych, tj. czas potrzebny na zajęcie ostatniego znacznika jest równy zaawansowaniu czasowemu transmisji pierwszej jednostki danych w kolejce.

➤ Jeżeli szybkość szczytowa napływu jednostek danych (v_{\max}) jest mniejsza od pojemności (C) i szybkość napływu znaczników (v_{ZN}) jest mniejsza od pojemności (C), to opóźnienie wnoszone przez sterownik jest równe zero. Jednostki danych są obsługiwane z szybkością większą niż szybkość generowania jednostek danych przez źródło (jeżeli $v_{ZN} \geq v_{\max}$) lub z szybkością większą niż szybkość napływu znaczników (jeżeli $v_{ZN} < v_{\max}$).

➤ Jeżeli szybkość napływu znaczników (v_{ZN}) jest większa od pojemności (C), to ograniczeniem szybkości obsługi jest pojemność bufora znaczników; w najgorszym razie tylko b_{ZN} jednostek danych może znajdować się w kolejce, a przychodzące znaczniki będą odrzucane.

2.2.2.6. ZMIENNOŚĆ OPÓŹNIENIA

Całkowite opóźnienie jednostki danych (opóźnienie pomiędzy węzłem źródłowym i ujściowym, tj. opóźnienie typu koniec–koniec), wnoszone przez sieć w czasie jej transferu pomiędzy źródłem i ujściem, jest sumą składowych o stałych i zmiennych wartościach. Składowe o wartościach stałych charakteryzują elementy sieci stosowane do transferu jednostek danych, podczas gdy składowe o wartościach zmiennych charakteryzują warunki, w jakich odbywa się transfer przez wykorzystywane elementy sieci.

Do składowych opóźnienia całkowitego o wartościach stałych zalicza się opóźnienia: paketyzacji, propagacji, komutacji i odtwarzania. Wartości tych opóźnień zależą od generowanych jednostek danych oraz od właściwości stosowanych procedur przetwarzania i od właściwości elementów sieci.

Do składowych opóźnienia całkowitego o wartościach zmiennych zalicza się opóźnienia buforowania. Opóźnienia te są rezultatem i jednocześnie miarą sumarycznego obciążenia sieci, czyli warunków, w jakich odbywa się obsługa jednostek danych przez współdzielone zasoby sieci.

Po oznaczeniu, dla k -tej ($k = 1, 2, \dots$) jednostki danych, przez:

d_k – całkowite opóźnienie,

c – suma składowych opóźnień całkowitego o wartościach stałych,

z_k – suma składowych opóźnień całkowitego o wartościach zmiennych,

różnica czasów międzyzłogoseniowych k -tej i l -tej jednostek danych w węźle ujściowym jest określona równaniem:

$$\delta_{lk} = d_k - d_l = (c + z_k) - (c + z_l) = z_k - z_l.$$

Jeżeli opóźnienia całkowite różnych jednostek danych byłyby równe, to czasy międzyzłogoseniowe jednostek danych w węźle ujściowym byłyby takie same jak czasy międzyzłogoseniowe jednostek danych w węźle źródłowym. Ze względu na to, że zmienne z_k ($k = 1, 2, \dots$) są zmiennymi losowymi, różnice czasów międzyzłogoseniowych są również zmiennymi losowymi.

Zmienność (fluktuacja) opóźnień (ang. *delay variation – jitter*) definiuje się w różny sposób. Do najważniejszych definicji zmienności opóźnień należą następujące:

➤ wariancja opóźnień jednostek danych:

$$j_k = E\{(z_k - E[z_k])^2\},$$

➤ średnie odchylenie opóźnień jednostek danych:

$$\bar{j}_k = P\{z_k - E[z_k] > z\},$$

➤ prawdopodobieństwo zdarzenia, że opóźnienia całkowite kolejnych jednostek danych nie są większe od zadanej wartości:

$$j_{k,k+1} = P\{z_k - z_{k+1} > z\},$$

➤ maksymalna różnica pomiędzy całkowitymi opóźnieniami jednostek danych:

$$j_{\max} = \max_{k,l=1,2,\dots,k \neq l} (z_k - z_l).$$

Innymi przykładowymi definicjami zmienności opóźnień są definicje dla sieci ATM: jednopunktowa zmienność opóźnień komórki (ang. *1-point Cell Delay Variation*) oraz dwupunktowa zmienność opóźnień komórki (ang. *2-point Cell Delay Variation*).

W pierwszej definicji zmienności opóźnień komórek ATM, tzn. *1-point CDV*, podstawą szacowania wartości zmienności opóźnień jest długość przedziału czasu T pomiędzy generacją kolejnych komórek, odpowiadająca szczytowej szybkości generowania komórek: $v_{\max} = 1/T$; zmienność opóźnień jest wyznaczana na podstawie porównania rzeczywistego czasu napływu komórki i oczekiwanego czasu napływu tej komórki na wyjściu danego połączenia (jeden punkt odniesienia). Zmienność opóź-

nienia k -tej komórki y_k jest definiowana jako różnica pomiędzy rzeczywistym czasem napływu k -tej komórki a_k a czasem odniesienia dla k -tej komórki c_k :

$$y_k = c_k - a_k,$$

gdzie:

$$c_0 = a_0 = 0,$$

$$c_{k+1} = \begin{cases} c_k + T & \text{gdy } c_k \geq a_k \\ a_k + T & \text{gdy } c_k < a_k \end{cases}.$$

Dodatnia wartość y_k oznacza zmniejszanie się odległości pomiędzy kolejnymi komórkami w porównaniu do wzorcowej odległości (skupianie się komórek), ujemna wartość y_k jest natomiast równoważna ze zwiększaniem się odległości pomiędzy kolejnymi komórkami w stosunku do wzorcowej odległości.

W drugiej definicji zmienności opóźnienia komórek ATM, tzn. *2-point CDV*, podstawą szacowania wartości zmienności opóźnienia komórek jest porównywanie czasów napływu komórek obserwowanych na wyjściu połączenia oraz czasów generowania komórek obserwowanych na wejściu połączenia (dwa punkty odniesienia). Wartość zmienności opóźnienia k -tej komórki u_k jest różnicą pomiędzy całkowitym opóźnieniem komórki x_k , rejestrowanym na końcu połączenia, a definiowanym dla połączenia opóźnieniem komórki d :

$$u_k = x_k - d.$$

Sterowanie zmiennością opóźnienia, polegające na ograniczaniu lub kompensowaniu zmienności opóźnienia pomiędzy wyprowadzanymi jednostkami danych w węźle ujściowym, jest realizowane przez stosowanie buforowania jednostek danych i odbywa się kosztem zwiększania ich opóźnienia.

Opóźnianie lub kompensacja zmienności opóźnienia polega na tym, że napływające jednostki danych są gromadzone i opóźniane w wyrównawczej pamięci buforowej (ang. *jitter removal buffer*) o przedział czasu pozwalający na wyprowadzanie jednostek danych z szybkością równą szybkości ich generowania oraz w odstępach równych czasom międzyzgłoszeniowym w węźle źródłowym. Całkowita kompensacja zmienności opóźnienia jest możliwa pod warunkiem, że pojemność pamięci buforowej w węźle ujściowym umożliwia dodatkowe opóźnianie jednostek danych, równe maksymalnemu opóźnieniu jednostek danych wnoszonych przez sieć. Całkowita kompensacja zmienności opóźnienia oznacza, że czasy międzyzgłoszeniowe jednostek danych, wyprowadzanych z sieci w węźle ujściowym, są dokładnie takie same jak czasy międzyzgłoszeniowe jednostek wprowadzanych do sieci w węźle źródłowym.

Zakres ograniczania lub kompensacji zmienności opóźnienia jest na ogół ograniczony wymaganiami aplikacji, formułowanymi w postaci górnego ograniczenia opóźnienia jednostek danych pomiędzy źródłem i ujściem, tj. górne ograniczenie sumy opóźnień wnoszonych przez sieć i wyrównawczą pamięć buforową. W takich przypadkach pojemność pamięci buforowej i wnoszone przez nią opóźnienie jest kompro-

misem pomiędzy dwoma sprzecznymi wymaganiami aplikacji: wyprowadzanie jednostek danych z sieci w sposób będący powtórzeniem procesu wprowadzania ich do sieci oraz opóźnienia jednostek danych nie większego od danego, górnego ograniczenia opóźnienia. Efektem rozwiązania realizującego ten kompromis jest odrzucanie w węzle ujściowym połączenia tych jednostek danych, które nie spełniają jednego i (lub) drugiego wymagania.

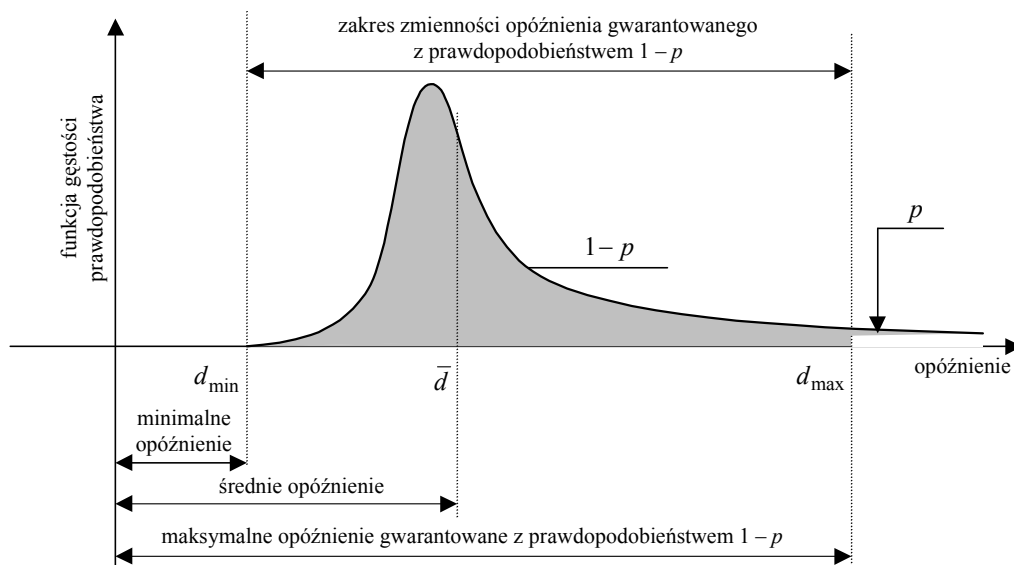
Klasycznym przykładem ruchu wrażliwego na zmienność (fluktuację) opóźnienia jest ruch generowany przez aplikacje czasu rzeczywistego (obsługa głosu, obrazu itd.) i obsługiwany w sieciach z komutacją jednostek danych. Przyczyną powstawania różnych opóźnień poszczególnych jednostek danych, należących do generowanego strumienia, jest indywidualny tryb ich obsługi. Obsługa w sieci jednostek danych, zawierających próbki głosu lub obrazu przygotowanych w źródle i wprowadzonych do sieci, polega na traktowaniu ich jako niezależnych fragmentów. Indywidualizacja obsługi jednostek danych w sieci powoduje, że oryginalne, powstałe w źródle strumienia jednostek danych, kolejność i odstępy czasowe pomiędzy nimi nie są zachowywane przez sieć. Oznacza to, że kolejność jednostek danych oraz odstępy czasowe pomiędzy odbieranymi jednostkami danych w odbiorniku mogą się różnić od kolejności jednostek i odstępów pomiędzy nimi w źródle. Wariancja opóźnienia, tj. różnica pomiędzy maksymalną i minimalną wartością opóźnienia, jest charakteryzowana jako zmienność (fluktuacja) opóźnienia. Minimalna wartość opóźnienia jednostki danych jest tożsama ze stałym opóźnieniem, wnoszonym przez system obsługi, którego wartość nie zależy od obciążenia systemu.

Maksymalne opóźnienie jednostki danych, wnoszone przez system obsługi, jest definiowane dwojako: jako maksymalne opóźnienie jednostki danych z prawdopodobieństwem $1 - p$ lub jako maksymalne opóźnienie jednostki danych z prawdopodobieństwem 1 .

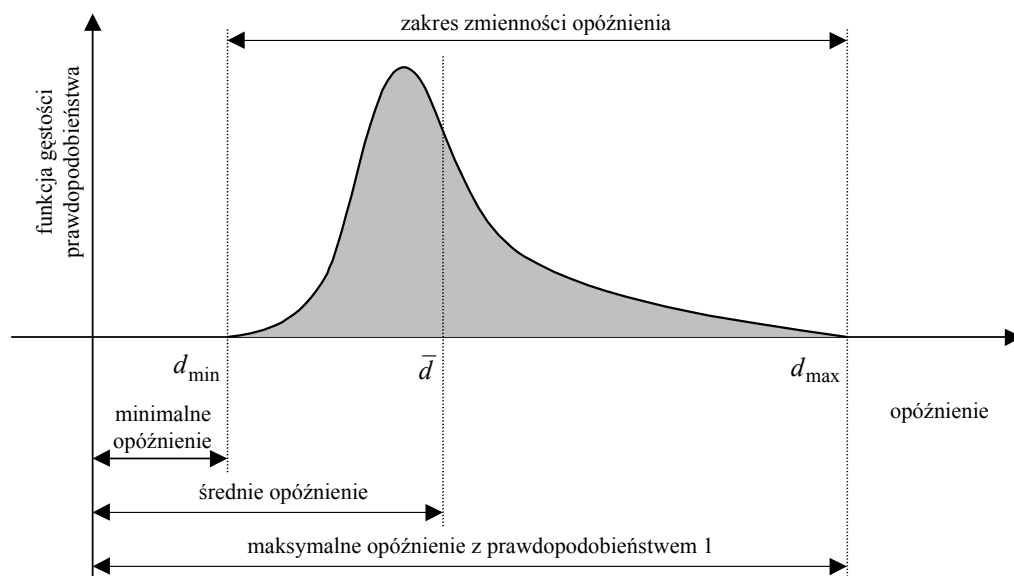
W pierwszym z wymienionych przypadków opóźnienie jednostek danych jest z prawdopodobieństwem $1 - p$ nie większe od zadanej maksymalnej wartości d_{\max} , z prawdopodobieństwem p opóźnienie jednostek danych jest większe od wartości d_{\max} . Jeżeli jednostki danych są generowane przez aplikację wrażliwą na opóźnienia, a d_{\max} jest górnym ograniczeniem wartości opóźnienia tolerowanego przez aplikację opóźnienia, to wartość współczynnika strat jednostek danych jest równa wartości prawdopodobieństwa p . Zmiana wartości współczynnika strat jest możliwa przez poprawę jakości usług, tzn. zmniejszenie wartości prawdopodobieństwa p lub zwiększenie górnego ograniczenia wartości opóźnienia tolerowanego przez aplikację (rysunek 2.2).

W drugim z wymienionych przypadków opóźnienie wszystkich jednostek danych jest zawsze nie większe od wartości d_{\max} , tzn. że system obsługi gwarantuje z prawdopodobieństwem 1 opóźnienie jednostek danych nie większe niż zadana wartość d_{\max} . Jeżeli aplikacja generująca jednostki danych jest wrażliwa na opóźnienia, to gwarancja opóźnienia d_{\max} oznacza brak strat jednostek danych (rysunek 2.3).

Zakłócenia regularności odbioru jednostek danych, powodowane zmiennością opóźnienia, mogą powodować spadek jakości sygnału wyjściowego w odbiorniku, jeżeli odbierane jednostki danych będą wyprowadzane bezpośrednio po odbiorze.



Rysunek 2.2. Rozkład opóźnienia i zmienności opóźnienia w przypadku gwarancji maksymalnego opóźnienia z prawdopodobieństwem $1-p$



Rysunek 2.3. Rozkład opóźnienia i zmienności opóźnienia w przypadku gwarancji górnego ograniczenia opóźnienia – gwarancji maksymalnego opóźnienia z prawdopodobieństwem 1

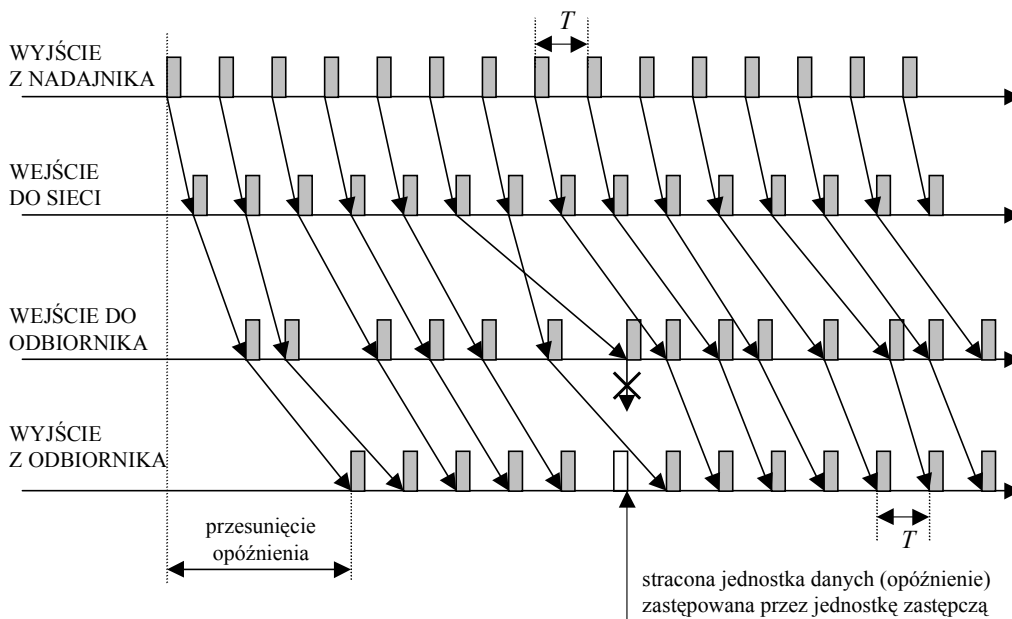
Eliminacja degradacji jakości sygnału wyjściowego powodowanego zmiennością opóźnienia w aplikacjach odtwarzania jest możliwa (w określonych granicach) dzięki zastosowaniu odbiorczego bufora wyrównawczego (ang. *play-out buffer*). Możliwość

gromadzenia i przechowywania jednostek danych w buforze odbiorczym pozwala na kompensowanie różnic opóźnień i uzyskanie efektu wygładzenia ruchu (ang. *smoothing*). Wygładzenie ruchu odbywa się kosztem wprowadzenia dodatkowego opóźnienia w odbiorniku, tj. przesunięcia (uskoku) opóźnienia (ang. *delay offset*).

Jednostki danych wyprowadzane z odbiornika (bufora odbiorczego) charakteryzuje stałe przesunięcie opóźnienia (czas oczekiwania na odtwarzanie) względem źródła jednostek danych (opóźnienie wyprowadzania) i częstotliwość wyprowadzania równa częstotliwości generowania jednostek danych w źródle ($f = 1/T$); wartość przesunięcia opóźnienia jest równoważna znajomości czasu odtwarzania (ang. *play-out point*).

Możliwe są dwa scenariusze obsługi jednostek danych, których opóźnienie – wnoszone przez sieć – jest większe od ustalonej wartości przesunięcia opóźnienia w odbiorniku.

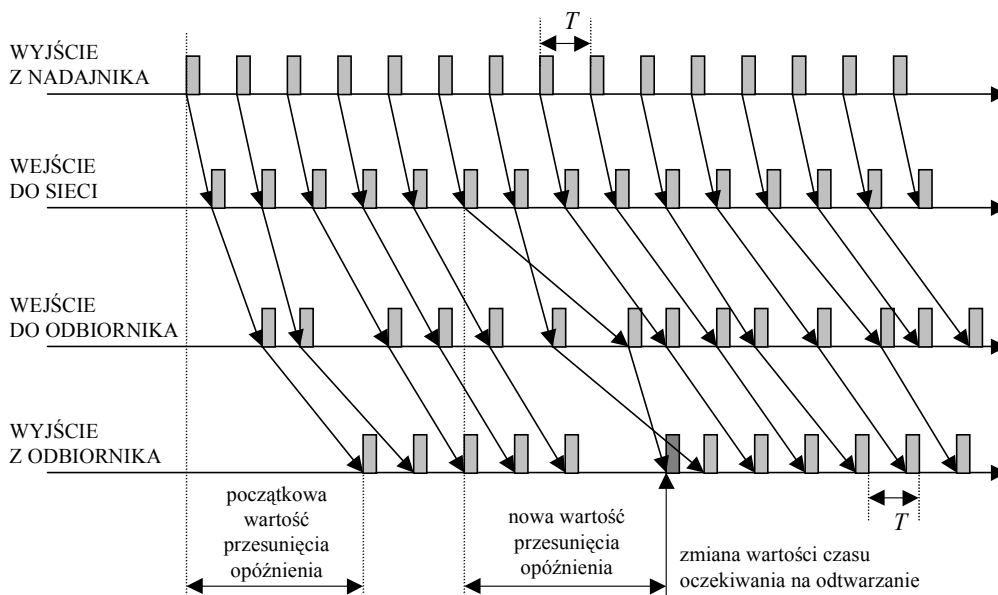
Pierwsze z rozwiązań polega na tym, że takie jednostki danych są traktowane jako bezużyteczne – są odrzucane i tracone. W razie takiego sposobu postępowania istnieje konieczność zastąpienia ich zastępczymi jednostkami danych, tzn. jednostkami danych generowanymi w odbiorniku (rysunek 2.4). Drugie z rozwiązań polega na tym, że odebranie jednostki danych z opóźnieniem większym niż przyjęta wartość przesunięcia opóźnienia powoduje zwiększenie wartości przesunięcia opóźnienia (rysunek 2.5).



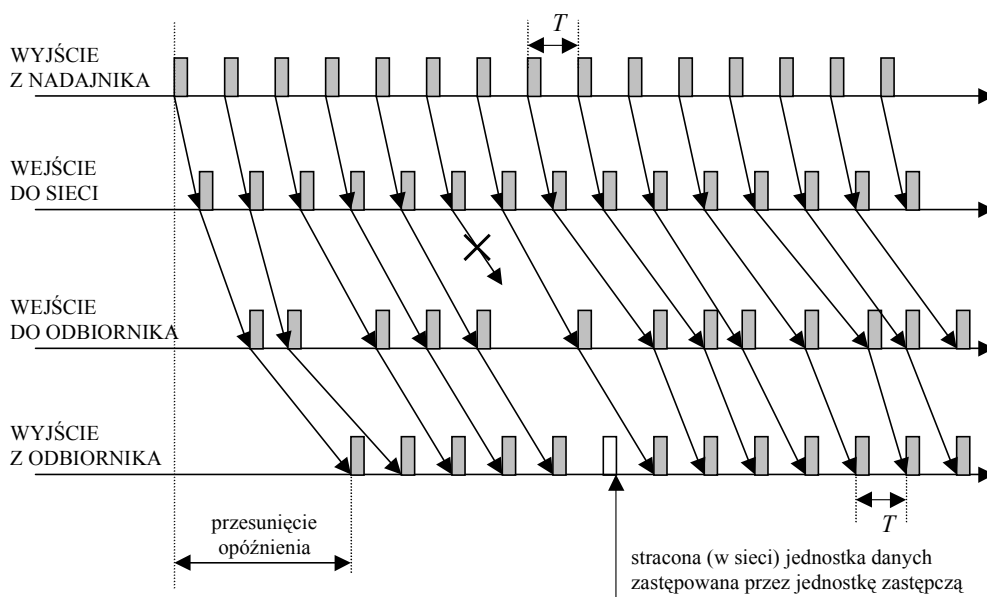
Rysunek 2.4. Eliminowanie zmienności opóźnienia – strata jednostki danych o opóźnieniu większym od ustalonego przesunięcia opóźnienia

Zwiększanie przesunięcia opóźnienia jest związane praktycznie z wyznaczeniem jego maksymalnej wartości. Jednostka danych może być stracona podczas przechodzenia przez sieć i oczekiwanie na jej nadejście jest nieuzasadnione. W razie wyczer-

pania maksymalnego czasu przesunięcia opóźnienia jednostka danych, która nie nadeszła w ustalonym przedziale czasu, musi być zastąpiona przez zastępczą jednostkę danych (rysunek 2.6).



Rysunek 2.5. Eliminowanie zmienności opóźnienia – zmiana wartości przesunięcia opóźnienia



Rysunek 2.6. Eliminowanie zmienności opóźnienia – strata jednostki danych w sieci

Istnieją różne metody generowania zastępczych jednostek danych na podstawie znajomości charakterystyk strumienia i wcześniej odebranych jednostek danych.

2.2.2.7. ASYMETRIA

Asymetrię (ang. *skew*) definiuje się jako różnicę czasów prezentacji dwóch wzajemnie związanych strumieni jednostek danych, np. obrazu i głosu. Dla aplikacji multimedialnych wymagania w omawianym zakresie są formułowane w postaci asymetrii zgrubej (ang. *coarse skew*) i asymetrii dokładnej (ang. *fine skew*).

2.2.2.8. CZAS OCZEKIWANIA NA ODTWARZANIE

Czas oczekiwania na odtwarzanie (ang. *latency*) definiuje się jako różnicę pomiędzy czasem generowania sygnału przez nadajnik i czasem odtwarzania go w odbiorniku. W aplikacjach odtwarzania dźwięku (ang. *playback*) czas oczekiwania na odtwarzanie jest równy przesunięciu opóźnienia.

Czas oczekiwania na odtworzenie jest sumą czasów: oczekiwania na transfer, transferu przez sieć i oczekiwania na wyprowadzenie z pamięci buforowej odbiornika.

Różne aplikacje charakteryzują się różną wrażliwością na długość czasu oczekiwania na odtwarzanie; niektóre z nich (np. transmisja głosu) są bardzo wrażliwe na czas oczekiwania na odtwarzanie nadanego sygnału. Wartość górnej granicy czasu oczekiwania na odtwarzanie, po przekroczeniu której jakość aplikacji gwałtownie spada, jest pojęciem subiektywnym lub obiektywnym. W przypadku aplikacji polegających na transmisji głosu wartość górnej granicy czasu oczekiwania na odtwarzanie jest determinowana fizjologią słuchu i zawiera się w przedziale 100÷300 ms.

2.2.2.9. WIERNOŚĆ ODTWARZANIA

Straty jednostek danych (powodowane stratami w sieci, przekroczeniem wartości górnej granicy przesunięcia opóźnienia i odkształceniem podstawy czasu) powodują degradację wierności odtwarzania (ang. *fidelity*). Niezależnie od tego, czy wzrost opóźnienia jednostek danych powoduje zwiększanie przesunięcia opóźnienia, czy też utratę jednostki danych i zastępowanie jej jednostką zastępczą, wierność odtwarzania spada. W pierwszym z omawianych przypadków spadek wierności jest konsekwencją zmiany podstawy czasu odtwarzania, w drugim natomiast jest powodowany utratą jednostki danych (wprowadzanie jednostki zastępczej ma na celu minimalizację spadku wierności odtwarzania).

Zależność pomiędzy przesunięciem opóźnienia a wiernością odtwarzania ma charakter zamienny (ang. *trade-off*). Zwiększanie wierności odtwarzania, które powoduje wzrost jakości odbioru, jest możliwe przez zwiększanie wartości (w najgorszym razie do maksymalnej, dopuszczalnej wartości) przesunięcia opóźnienia (minimalizującego liczbę traconych jednostek danych), powodującego spadek jakości odbioru. Z kolei zmniejszanie przesunięcia opóźnienia, które powoduje wzrost jakości odbioru, odbywa się za cenę zwiększenia liczby traconych jednostek danych, powodującego spadek

wierności odtwarzania. Konsekwencją tych zależności jest to, że jakość odbioru jest następstwem kompromisu pomiędzy przesunięciem opóźnienia i wiernością odtwarzania.

Aplikacje generujące ruch w czasie rzeczywistym i wymagające odtwarzania w czasie rzeczywistym są klasyfikowane, ze względu na wymagania dotyczące wierności odtwarzania, na dwie grupy: wrażliwych i niewrażliwych na wierność odtwarzania.

W przypadku aplikacji wrażliwych na wierność odtwarzania wymagane jest bardzo dokładne odtwarzanie. Wymaganie to wynika bądź z natury aplikacji, bądź z braku możliwości obsługi strat jednostek danych i obsługi zmian w podstawie ich odtwarzania. Wówczas jedynym rozwiązaniem jest wykorzystywanie stałego przesunięcia opóźnienia, którego wartość musi być większa od największego możliwego opóźnienia jednostki danych w sieci, tak aby wartość opóźnienia żądanej jednostki danych nie była większa od ustalonego czasu odtwarzania (ang. *play-out point*). Spełnienie podanych warunków oznacza, że konieczne jest gwarantowanie przez sieć zachowania górnego ograniczenia opóźnienia oraz że wartość tego ograniczenia musi być na tyle mała, aby spełnić wymagania nakładane przez aplikację na czas odtwarzania.

Wiele z aplikacji odtwarzania dźwięku projektuje się tak, aby osłabić ich wrażliwość na straty jednostek danych i wyposażyć w mechanizmy adaptacyjne, pozwalające na obsługę zmiennych opóźnień przez zmianę położenia punktu odtwarzania. Jedno z podstawowych rozwiązań polega na wykorzystaniu informacji z jednostek danych poprzedzających traconą lub z jednostek danych o zbyt dużym opóźnieniu.

Niezależnie od mechanizmów adaptacyjnych, wprowadzanych w aplikacjach odtwarzania dźwięku w celu zwiększenia ich elastyczności, istnieje potrzeba sterowania opóźnieniem wnoszonym przez sieć. Jakość omawianych aplikacji jest determinowana niewielką liczbą jednostek danych, których opóźnienie jest większe od ustalonych przesunięć opóźnienia. Redukcja liczby takich jednostek danych, równoważna zmniejszeniu wartości maksymalnego opóźnienia w sieci, znacząco wpływa na poprawę jakości aplikacji.

Jakość obsługi aplikacji odtwarzania w sieciach datagramowych jest determinowana właściwymi dla tych sieci: zmiennością opóźnień wnoszonych przez kolejki pakietów (jednostki danych) w węzłach sieci i stratami pakietów powodowanych przeciążeniami. Wartości wymienionych miar jakości działania sieci zależą od jej obciążenia – rosnące obciążenie powoduje szybki wzrost stopnia degradacji jakości usług mierzonych m.in. dwoma wymienionymi wskaźnikami jakości działania.

2.3. PODSUMOWANIE

Wybór miar jakości usług dostarczanych w sieciach komputerowych zależy od celu gromadzenia informacji dotyczących jakości działania sieci.

Z punktu widzenia użytkownika sieci istotne są tylko te miary jakości usług, które charakteryzują procesy wymiany danych pomiędzy systemami końcowymi i których

wartość wpływa na jakość użytkowania aplikacji, tzn. miary jakości transferu danych w sieci. Istotne jest przy tym, aby stosowane miary jakości były dopasowane do aplikacji implementowanych w systemach końcowych. Celem wyboru miar istotnych dla użytkownika jest monitorowanie stopnia realizacji wymagań ilościowych i jakościowych w obsłudze ruchu generowanego przez aplikacje.

Z punktu widzenia operatora sieci wartości miar jakości istotnych dla użytkownika są ograniczeniami, które muszą być spełnione w zadaniach maksymalizacji stopnia wykorzystania zasobów. Ze względu na złożoność zadań efektywnego wykorzystania zasobów, liczba miar jakości stosowanych w systemach zarządzania ruchem i monitorowania ruchu w sieciach jest znacznie większa od liczby miar jakości transferu danych.

LITERATURA

- [1] ADAS A., *Traffic models in broadband telecommunications networks*, IEEE Communications Magazine, July 1997.
- [2] ATKINS J., NORRIS M., *Total Area Networking*, Wiley, Chichester 1995.
- [3] ANDRADE J. BURAKOWSKI W., VILLEN-ALTAMIRANO M., *Characterization of cell traffic generated by ATM source*, ITC-13, s. 545–560, 1991.
- [4] BERTSEKAS D., GALLAGER R., *Data networks*, Prentice-Hall International, Englewood Cliffs 1987.
- [5] BUTRIMENKO A.W., *Projektowanie i eksploatacja sieci komputerowych*, PWN, Warszawa 1983.
- [6] ELBAUM R., SIDI M., *Topological design of local-area networks using genetic algorithms*, IEEE Transactions on Networking, Vol. 4, No. 5, October 1996, s. 766–778.
- [7] FLOD J.E., *Telecommunications, switching, traffic and networks*, Prentice Hall, New York 1995.
- [8] FRANK H., FRISCH I.T., *Communication, transmission and transportation networks*, Addison-Wesley, Reading 1971.
- [9] FROST V.S., MELAMED B., *Traffic modelling for telecommunications networks*, IEEE Communications Magazine, Vol. 32, No. 3, March 1994, s. 70–81.
- [10] GERLA M., KLEINROCK L., *On the topological design of distributed computer networks*, IEEE Transactions on Communications, Vol. COM-25, 1977, s. 48–60.
- [11] GRZECH A., *Structures of services delivered by intelligent networks*, Proceedings of IEEE Singapore International Conference on Networks, IEEE 1995, s. 388–392.
- [12] GRZECH A., STOPA E., *Description of services delivered by Intelligent Networks*, Systems Science, Vol. 21, No. 3, 1995, s. 43–55.
- [13] HARMAS D.D., KRAETZL M., COLBOURN C.J., DEVITT J.S., *Network reliability*, CRC Press, Boca Raton 1995.
- [14] HUANG Y.M., GUAN S.U., *A refined cut-through buffer management scheme for layered protocol stacks*, IEEE Communications Magazine, Vol. 32, No. 3, March 1994, s. 82–86.
- [15] JAFFE J.M., *Flow control power is nondcentralizable*, IEEE Transactions on Communications, Vol. COM-29, No. 9, September 1981, s. 1301–1306.
- [16] IEEE Journal of Selected Areas in Communications – special issue on network performance evaluation, Vol. SAC-4, No. 6, September 1986.

- [17] KLEINROCK L., *Communication nets*, Volume I: *Stochastic message flow and delay*, Volume II: *Computer Applications*, Wiley, New York 1976.
- [18] KNOCHE H., MEER H., *Quantitative QoS-mapping: a unified approach*, University of Hamburg, 2001.
- [19] KULKARNI A., LI A.Q., *Performance analysis of a rate-based feedback control scheme*, IEEE/ACM Transactions on Networking, Vol. 6, No. 6, December 1998, s. 797–810.
- [20] KUMMERLE K., LIMB J.O., TOBAGI F.A., *Advances in Local Area Networks*, IEEE Press, New York 1986.
- [21] LORENZ D.H., *QoS routing in networks with uncertain parameters*, University of Haifa, 1998.
- [22] MAGLARIS B., ANASTASSIOU D., SEN P., KARLSSON G., ROBINS J., *Performance models of statistical multiplexing in packet video communications*, IEEE Transactions on Communications, Vol. COM-36, 1988, s. 834–844.
- [23] ONVURAL R.O., *Asynchronous Transfer Mode; Performance issues*, Artech House, Boston 1995.
- [24] PAPIR Z. (red.), *Sieci z komutacją pakietów – od X.25 do Frame Relay i ATM*, Wydawnictwo Fundacji Postępu Telekomunikacji, Kraków 1997.
- [25] PENNOTTI M.C., SCHWARTZ M., *Congestion control in store-and-forward tandem links*, IEEE Transactions on Communications, Vol. COM-23, No. 12, December 1975, s. 1434–1443.
- [26] POJOLLE G., SERET D., DROMARD D., HORLAI E., *Integrated digital communications networks*, Wiley, Chichester 1988.
- [27] RODEN M.S., *Analog and digital communication systems*, Prentice Hall, New Jersey 1996.
- [28] ROSENBERG S., AISSAOUI M., GALWAY K., GIROUX N., *Functionality at the edge: designing scalable multiservice ATM networks*, IEEE Communications Magazine, May 1998, s. 88–99.
- [29] SAHINOGLU Z., TEKINAY S., *On multimedia networks: self-similar traffic and network performance*, IEEE Communications Magazine, January 1999, s. 48–56.
- [30] SAITO H., *Call Admission Control in an ATM network using upper bound of Cell Loss Probability*, IEEE Transactions on Communications, Vol. 40, No. 9, September 1992, s. 1512–1521.
- [31] SEIDLER J., *Analiza i synteza sieci łączności dla systemów teleinformatycznych*, PWN, Warszawa 1979.
- [32] TAKAGI H. (red.), *Stochastic analysis of computer and communication systems*, North-Holland, Amsterdam 1990.
- [33] VANDALORE B., FAHMY S., JAIN R., GOYAL R., GOYAL M., *QoS and multipoint support for multimedia applications over the ATM ABR service*, IEEE Communications Magazine, January 1999, s. 53–57.
- [34] WINCH R.G., *Telecommunication Transmission Systems*, McGraw-Hill Telecommunications, New York 1998.
- [35] WOLFINGER B.E., *Characterization of mixed traffic load in service integrated networks*, Proceedings 6th Polish Teletraffic Symposium, Wrocław 1999, s. 11–28.
- [36] WOŹNIAK J., *Analiza i projektowanie protokołów komunikacyjnych dla radiowych sieci teleinformatycznych*, Wydawnictwo PG, Gdańsk 1991.

3. ODWZOROWANIE WARTOŚCI PARAMETRÓW JAKOŚCI USŁUG W MODELU WARSTWOWYM

Warunkiem realizacji i eksploatacji rozwiązania teleinformatycznego o zadanej funkcjonalności i jakości usług jest dobra znajomość związków zachodzących pomiędzy wartościami parametrów jakości usług dostarczanych przez różne moduły funkcjonalne, zlokalizowane w różnych warstwach architektury sieci. Znajomość tych zależności jest także warunkiem optymalnego wykorzystania zasobów, tzn. osiągnięcia stanu, kiedy zasoby sieci są sterowalne i sterowane, a celem jest zabezpieczenie sieci przed przeciążeniami powodującymi degradację jakości usług i stopnia wykorzystania zasobów.

Związki pomiędzy jakością usług w różnych warstwach architektury sieci mają także istotne znaczenie dla predykcji zmian jakości usług dostarczanych użytkownikom sieci w warstwie aplikacji, w następstwie celowych i losowych zmian zachodzących w warstwach niższych architektury sieci oraz zmian w warunkach eksploatacji sieci.

3.1. MODEL ŁĄCZENIA SYSTEMÓW OTWARTYCH

3.1.1. WARSTWOWOŚĆ

Warstwowość (architektura warstwowa), będąca formą funkcjonalnej modularności, jest podstawowym pojęciem w zadaniach projektowania sieci komputerowych. Istotą koncepcji funkcjonalnej modularności jest możliwość dwójakiego rozumienia i analizowania każdego modułu. Z jednej strony przedmiotem zainteresowań może być jego wewnętrzna struktura i wykonywane wewnątrz operacje, z drugiej zaś wejścia i wyjścia modułu oraz relacje funkcjonalne pomiędzy nimi. To drugie rozumienie modułu pozwala traktować go – zwłaszcza wtedy, gdy jest komponentem większej całości – jako czarną skrzynkę (ang. *black box*). Tak rozumiany i opisany moduł może być, wraz z innymi modułami, wykorzystany do konstrukcji bardziej złożonych modułów, które z kolei można – na wyższym (innym) poziomie ogólności (abstrakcji) – rozumieć jako jeden moduł itd.

Przedstawione podejście prowadzi w sposób naturalny do strukturalnego projektowania i budowy systemów złożonych, implementowanych jako zbiory połączonych modułów. W razie zastosowania do budowy systemów złożonych modułów standardowych, mogą one być zastępowane przez nowe, funkcjonalnie równoważne (tańsze, bardziej niezawodne, realizowane w różnych technologiach, uzupełnione o różne wartości dodane itd.). O jakości i ilości usług dostarczanych przez takie systemy decydują rodzaj i liczba użytych modułów oraz sposób ich połączenia.

Korzyści wynikające z funkcjonalnej modularności dla projektowania, implementacji, utrzymania i modernizacji (m.in. dekompozycji funkcji, wymienności, standardowości) uzasadniają popularność i zakres stosowalności architektur warstwowych sieci (w tym sieci komputerowych), w których stosowane są najprostsze schematy łączenia modułów – hierarchia zagnieżdżonych modułów (ang. *nested modules*).

W architekturach warstwowych istotne są dwa odmienne rodzaje komunikacji pomiędzy modułami tej samej i różnych warstw. Pierwszy z nich to protokół (algorytm) warstwy (protokół komunikacyjny) wykorzystywany przez równoważne moduły (z reguły rozproszone przestrzennie) do wymiany wiadomości (w różnych formach) w celu dostarczenia modułom warstwy wyższej usługi o wymaganych cechach jakościowych i ilościowych. Druga to precyzyjna specyfikacja interfejsu (protokół transmisyjny) pomiędzy danym modułem a modułem warstwy niższej, poprzez który odbywa się wymiana wspomnianych wiadomości. Obydwa rodzaje komunikacji mają podstawowe znaczenie dla zrozumienia koncepcji projektowania, działania i standaryzacji architektur warstwowych. Są jednocześnie jednym ze źródeł ich ograniczeń koncepcyjnych i funkcjonalnych.

Podstawowe znaczenie, ze względu na uniwersalność i rozpowszechnienie, ma architektura warstwowa modelu odniesienia dla łączenia systemów otwartych RM OSI (ang. *Reference Model of Open Systems Interconnections*) w postaci zaproponowanej jako standard dla sieci komputerowych przez ISO (ang. *International Standard Organisation*) – model odniesienia ISO/OSI. Wiele z istniejących sieci komputerowych zaprojektowano i zrealizowano zgodnie z innymi architekturami warstwowymi, spośród których podstawowe znaczenie (ze względu na ogólność) mają: SNA (ang. *Systems Network Architecture*), DNA (ang. *Digital Network Architecture*), TCP/IP (ang. *Transport Control Protocol/Internet Protocol*). W architekturach tych różna jest liczba warstw, różne są ich nazwy oraz zakresy funkcji przez te warstwy realizowane. Wspólna dla różnych architektur warstwowych jest koncepcja współpracy warstw i wymiany usług pomiędzy nimi. O różnicach pomiędzy wymienionymi architekturami zdecydowały odmienne koncepcje budowy systemów rozproszonych, różne sformułowania celów stawianych zbiorom protokołów oraz akcentowanie znaczenia różnych aspektów komunikacji w środowisku rozproszonym. Mimo stałego wzrostu znaczenia architektury TCP/IP, ograniczającego znaczenie modelu ISO/OSI, ten ostatni ma – ze względu na czas jego opracowania i oczekiwania z nim związane – w dalszym ciągu szeroki zakres zastosowań oraz niewątpliwe walory poznawcze i dydaktyczne, istotne dla rozumienia otwartych i modularnych architektur sieciowych.

Zasada modularności i zagnieżdżenia warstw była i jest stosowana do budowy modeli architektur elementów składowych sieci teleinformatycznych, w szczególności obejmujących funkcjonalność niższych warstw modelu ISO/OSI, realizowanych z użyciem różnych technologii i standardów, jak np. LAN (ang. *Local Area Network*), N-ISDN (ang. *Narrowband Integrated Services Digital Network*), B-ISDN (ang. *Broadband Integrated Services Digital Network*), ATM (ang. *Asynchronous Transfer Mode*).

3.1.2. MODEL WARSTWOWY ŁĄCZENIA SYSTEMÓW OTWARTYCH ISO/OSI

Model warstwowy ISO/OSI to zbiór funkcji i operacji międzymodułowych oraz reguł ich rozmieszczania i strukturalizacji w siedmiu warstwach współpracujących w ramach hierarchii i zagnieżdżenia. Na funkcjonalność i złożoność tego modelu istotny wpływ wywarły: doświadczenia zgromadzone w czasie eksploatacji ograniczonych funkcjonalnie hierarchicznych struktur sieci SNA, wola i przekonanie o możliwości zbudowania uniwersalnego modelu łączenia dowolnych systemów otwartych oraz dostępne technologie komunikacyjne i transmisyjne. Parametry modelu ISO/OSI są wynikiem kompromisu pomiędzy dążeniem do wyodrębnienia w warstwach jednoznacznie zdefiniowanych i separowalnych funkcji a kosztem realizacji sieci zgodnej z takim modelem. Model ISO/OSI to także zbiór odpowiednich protokołów opracowanych z myślą o bezproblemowym łączeniu sieci (ang. *internetworking utopia*) – jeżeli używałoby się tych i tylko tych protokołów, wszystkie problemy związane z łączeniem sieci byłyby rozwiązane.

Dzisiejsza krytyka tego modelu odwołuje się przede wszystkim do jego złożoności, sztywności, przedefiniowania, dużej nadmiarowości funkcjonalnej i organizacyjnej, centralizacji sterowania oraz niewielkiej adaptowalności do coraz powszechniej stosowanych koncepcji rozpraszania inteligencji w sieciach komputerowych.

Niezależnie jednak od krytyki, z jaką spotyka się model ISO/OSI, pozostanie on pewnym wzorem dla istniejących i stosowanych oraz proponowanych zbiorów reguł współpracy elementów rozproszonych przestrzennie i funkcjonalnie systemów komputerowych. Rola modelu odniesienia w porządkowaniu bogatej i złożonej funkcjonalnie sfery protokołów komunikacyjnych jest trudna do przecenienia. Bez tego modelu trudno sobie wyobrazić rozwój infrastruktury sieci teleinformatycznych.

Przetrawanie i aktualność modelu ISO/OSI, pomimo wielu protokołów komunikacyjnych opracowanych w ostatnich latach i tym samym okazji do weryfikacji jego przydatności, dowodzi, że przyjęte sposoby modelowania złożonych rozwiązań komunikacyjnych mają charakter uniwersalny.

3.1.2.1. WARSTWA FIZYCZNA

Podstawową funkcją warstwy fizycznej (ang. *physical layer*) jest zapewnienie kanału wirtualnego na potrzeby transmisji sekwencji bitów pomiędzy parą węzłów

(ang. *nodes*) połączonych fizycznym kanałem transmisyjnym. Realizacja tej funkcji oznacza istnienie po obydwu stronach modułów fizycznych interfejsów (ang. *physical interface module*), których zadaniem jest – po stronie nadawczej – odzworowanie napływających bitów w sygnały właściwe dla danego kanału transmisyjnego oraz – po stronie odbiorczej – przekształcenie przekazywanych sygnałów w sekwencję bitów. Moduły te są zwykle nazywane modemami, przy czym termin ten odnosi się do dowolnego modułu pełniącego wyżej wymienione funkcje modulacji i demodulacji (nakładania danych na sygnał nośny i odtwarzania danych z sygnału nośnego). Funkcje i charakterystyki modemów są determinowane przez rodzaje i sposoby organizacji transmisji w kanale. Dla warstw wyższych czarna skrzynka – formowana przez zespół: modem, kanał transmisyjny i modem – oznacza formowanie ciągu bitów (ang. *bit pipe*) z ukrytą wewnątrz złożonością fizycznego kanału transmisyjnego.

W kanałach transmisyjnych typu punkt–punkt (ang. *point-to-point*) odbierane na końcu kanału sygnały są „zazumioną” repliką sygnałów generowanych po stronie nadawczej. W niektórych sieciach do transmisji wybranych lub wszystkich sygnałów stosowane są kanały wielodostępne – odbierane w węzłach sygnały są sumą sygnałów, a sygnały transmitowane z jednego węzła są odbieralne w wielu węzłach.

Dla konstrukcji i organizacji modemów istotne znaczenie mają charakterystyki strumieni podawanych bitów – cztery podstawowe tryby transmisji to: synchroniczny (ang. *synchronous*), przerywany synchroniczny (ang. *intermittent synchronous*), asynchroniczny (ang. *asynchronous*) i izochroniczny (ang. *isochronous*).

3.1.2.2. WARSTWA ŁĄCZA DANYCH

Przeznaczenie warstwy łącza danych (ang. *data link control layer*) to konwersja usług modułów warstwy fizycznej (warstwy niższej) w wirtualne połączenie komunikacyjne dla wolnego od błędów i asynchronicznego przekazywania pakietów w obydwu kierunkach (z punktu widzenia warstwy łącza danych pakiet jest strumieniem bitów otrzymywanym z modułu warstwy wyższej) pomiędzy dwoma sąsiednimi węzłami. Połączenie komunikacyjne jest ustanawiane i utrzymywane w celu inicjowania, wspierania, koordynacji i kończenia dwóch współbieżnych i równoważnych procesów (ang. *peer processes*) obsługi w dwóch sąsiednich komunikujących się węzłach.

Komunikacja w omawianej warstwie jest asynchroniczna w dwojakim rozumieniu. Po pierwsze – istnieje zmienne opóźnienie pomiędzy wprowadzeniem pakietu do modułu warstwy z jednego końca kanału i jego wyprowadzeniem po drugiej stronie; zmienność tego opóźnienia jest konsekwencją konieczności korekcji błędów powstających w kanale oraz zmienną długością obsługiwanego pakietów. Po drugie – zmiennej czas pomiędzy pojawianiem się kolejnych pakietów w modułach warstwy jest wynikiem braku pakietów do przekazania w warstwach wyższych i (lub) braku możliwości akceptacji podawanych pakietów w warstwie łącza danych (np. ze względu na konieczność retransmitowania wcześniej obsługiwanego pakietów).

Obsługa pakietów przez moduły warstwy łącza danych wiąże się z generowaniem ramki (ang. *frame*), powstającej przez uzupełnienie pakietu o nadmiarowe bity sterowania. Ich ilość i struktura zależą od zakresu funkcjonalnego protokołu komunikacyjnego warstwy, którego realizację wspierają. Podstawowe zadania tych nadmiarowych bitów są związane z identyfikacją początku i końca ramki jako jednostki danych, identyfikacją typu ramki, detekcją i (lub) korekcją błędów wnoszonych przez kanał transmisyjny, retransmisją ramek.

Zakres funkcjonalny modułów warstwy łącza danych zależy od tego, czy protokół tej warstwy realizuje komunikację typu punkt–punkt czy wielodostęp (ang. *multi-access*). W drugim przypadku niezbędny jest moduł pośredniczący pomiędzy modułami warstwy łącza danych i warstwy fizycznej. Zadaniem modułu sterowania dostępem do medium MAC (ang. *Medium Access Control*) jest zarządzanie i rozwiązywanie konfliktów powstających w następstwie jednoczesnych żądań dostępu modułów warstwy łącza danych do usług dostarczanych przez moduły warstwy fizycznej.

3.1.2.3. WARSTWA SIECIOWA

W warstwie sieciowej (ang. *network layer*) – jedynej w architekturze warstwowej – realizowane są procesy związane ze wszystkimi węzłami sieci oraz każdą zewnętrzną lokalizacją. Wszystkie te procesy są równoważne i współpracują w zadaniach implementacji algorytmów ustalania tras (ang. *routing*), komutacji (ang. *switching*) i sterowania przepływem (ang. *flow control*).

Procesy realizowane w warstwie sieciowej są odpowiedzialne za przyjmowanie jednostek danych z warstw wyższych (najczęściej transportowej) oraz podejmowanie decyzji o tym, czy i kiedy pakiety zostaną przekazane do warstwy łącza danych. Na decyzje te ma wpływ między innymi analiza przeciążeń w sieci. Opóźnienia wynikające z konieczności unikania przeciążeń oraz wnoszone przez procesy realizowane w warstwie łącza danych wymagają implementacji w warstwie sieciowej funkcji buforowania.

Ustalanie tras i sterowania przepływem wymaga wymiany informacji pomiędzy równoważnymi procesami warstwy sieciowej. Część z tych informacji jest przekazywana w postaci bitów sterowania dodawanych do pakietów danych przekazywanych pomiędzy węzłami. Pozostałe informacje (np. do ustanowienia nowych sesji, aktualizacji tablic kierunków) są przekazywane w postaci pakietów sterowania, które nie zawierają danych.

Jeżeli przyjąć, że moduły warstwy sieciowej i niższych we wszystkich węzłach sieci są postrzegane jako czarna skrzynka, to warstwę sieciową można interpretować jako wirtualne połączenie typu koniec–koniec (ang. *end-to-end*), przenoszące pakiety pomiędzy źródłem i ujściem. W zależności od zbioru użytych w warstwie sieciowej mechanizmów oraz wartości ich parametrów połączenie to może dostarczać usługi różnej jakości. W większości przypadków jest to połączenie dostarczające każdy pakiet niezawodnie, bezbłędnie i w jednej kopii. W razie rezygnacji w warstwie sieciowej z mechanizmu gwarantującego wymaganą jakość określonej cechy transferu pakietów, zadanie zagwarantowania oczekiwanej jakości przechodzi do jednej z warstw wyższych.

Warstwa sieciowa jest najbardziej złożona koncepcyjnie, dlatego że warunkiem realizacji funkcji warstwy jest współpraca wszystkich równoważnych, przestrzennie rozproszonych procesów. Istotą tej współpracy jest wspólna realizacja rozproszonych algorytmów. W przeciwieństwie do warstwy sieciowej, w warstwie łącza danych (z wyłączeniem podwarstwy sterowania łączem danych obsługującej wielodostęp) oraz warstwach wyższych komunikacja odbywa się tylko pomiędzy parami równoważnych procesów.

3.1.2.4. WARSTWA TRANSPORTOWA

Zadaniem warstwy transportowej (ang. *transport layer*) jest niezawodne dostarczanie wiadomości ze źródła do ujścia (ang. *entry-to-exit*). Dla każdego wirtualnego połączenia dostarczanego przez warstwę sieciową warstwie transportowej uaktywniane są dwa równoważne procesy – po jednym na każdym z końców (źródło i ujście) omawianego połączenia transportowego. Zadania tych procesów wynikają z charakterystycznego dla danej sieci zakresu niezbędnego dopasowania strumienia danych dostarczanych przez warstwy wyższe do warunków obsługi tych strumieni przez połączenie wirtualne realizowane w warstwie sieciowej i odwrotnie.

W warstwie transportowej realizowane są segmentacja i desegmentacja wiadomości (odtworzenie wiadomości). Złożoność tego zadania oraz warunków jego realizacji zależy od wartości parametrów modułów i mechanizmów warstwy transportowej (np. pojemność pamięci buforowej i zasady gospodarki tą pamięcią) oraz jakości usług dostarczanych przez warstwę sieciową (np. transfer pakietów zgodnie z kolejnością nadawania).

W warstwie transportowej możliwa jest implementacja funkcji zwielokrotniania (ang. *multiplexing*) i (lub) rozdziału (ang. *splitting*) strumieni danych podawanych z warstw wyższych. W pierwszym z wymienionych przypadków scalenie strumieni danych generowanych przez różne sesje może prowadzić do redukcji narzutów organizacyjnych na obsługę transferu danych przez warstwę sieciową, a tym samym do poprawy jakości obsługi. W drugim przypadku brak możliwości obsługi przez warstwę sieciową strumienia danych generowanego w warstwie sesji wymaga jego podziału; podział ten dla warstwy sieciowej jest równoważny z obsługą strumieni danych wielu sesji.

Mechanizmy implementowane w warstwie transportowej mogą być uzupełniającymi lub zastępującymi mechanizmy stosowane w warstwie sieciowej. Klasycznym przykładem jest uzupełnienie bezpołączeniowego (ang. *connectionless*) protokołu warstwy sieciowej protokołem połączeniowym (ang. *connection-oriented*) warstwy transportowej.

W sieciach utworzonych przez połączenie różnych sieci o niekompatybilnych warstwach sieciowych w warstwie transportowej odbywa się konwersja strumieni danych oraz warunków ich obsługi właściwych dla połączonych podsieci. Przykładem takich zadań jest konwersja długości pakietów.

3.1.2.5. WARSTWA SESJI

Realizacja usług warstwy sesji (ang. *session layer*) wiąże się z zarządzaniem dwoma równoważnymi procesami obsługiwany przez połączenie komunikacyjne warstwy transportowej. Zadaniem modułów tej warstwy jest ustanawianie, nadzorowanie i zawieszanie połączenia pomiędzy procesami użytkowymi.

3.1.2.6. WARSTWA PREZENTACJI

Główne zadania modułów funkcjonalnych warstwy prezentacji (ang. *presentation layer*) związane są z szyfrowaniem danych i ich zabezpieczeniem kryptograficznym, kompresją danych oraz konwersją kodów, formatów i typów danych po stronie nadawczej i realizacją procesów odwrotnych po stronie odbiorczej.

Zadania te, będące zadaniami przetwarzania, a nie komunikacji, dotyczą trzech ważnych zagadnień organizacyjnych w sieciach: oszczędzania zasobów komunikacyjnych i transmisyjnych sieci (kompresja), bezpieczeństwa danych (kryptografia) oraz eliminacji niekompatybilności sprzętowej i programowej (konwersja). Chociaż zadania te mogą być realizowane także w innych warstwach (w tym wielokrotnie), to celowe i zasadne jest ich wykonanie możliwie najwcześniej, tzn. w modułach warstwy pierwszej na drodze strumieni danych pomiędzy aplikacjami. Kompresja w warstwie prezentacji oznacza największą z możliwych redukcję informacji organizacyjnych, niezbędnych do realizacji transferu wiadomości. Wykonanie zabezpieczeń kryptograficznych w warstwie prezentacji oznacza względnie największe bezpieczeństwo transferowanych danych. Przekształcenie kodów, formatów i typów w warstwie prezentacji oznacza, że wszystkie strumienie danych w warstwach niższych mogą być obsługiwane według tych samych zasad.

3.1.2.7. WARSTWA APLIKACJI

Zadaniem warstwy aplikacji (ang. *application layer*) jest udostępnianie użytkownikom (procesom aplikacyjnym) usług oferowanych przez pozostałe warstwy (środowisko OSI). W przeciwieństwie do warstw niższych, dostarczających usługi wielu różnym aplikacjom, w warstwie aplikacji realizowane są procesy specyficzne dla poszczególnych aplikacji.

3.1.3. ALGORYTMY

Zbiór funkcji, objętych modelem odniesienia, dzieli się na dwa podzbiory:

➤ pierwszy podzbiór, zawierający funkcje trzech najniższych warstw (fizycznej, łącza danych i sieciowej), jest określany mianem podsieci komunikacyjnej (ang. *communication subnetwork*), podsieci (ang. *subnet*), podsystemu komunikacyjnego (ang. *communication subsystem*) lub częścią sieciową (ang. *carrier portion*),

➤ drugi podzbiór, zawierający funkcje trzech najwyższych warstw (sesji, prezentacji i aplikacji), jest określany mianem procesu użytkownika (ang. *host process*) lub części użytkownika (ang. *customer portion*).

Wymienione podzbiory funkcji (warstw) łączy zestaw procedur (funkcji) obsługujących połączenia pomiędzy źródłem a ujściem, zawarty w warstwie transportowej (pierwsza, licząc od najniższej warstwy, warstwa zapewniająca połączenia pomiędzy procesami końcowymi), który zwykle jest włączany do procesu użytkownika.

Wszystkie omawiane i wyróżnione w architekturze ISO/OSI warstwy mają wspólne cechy, będące konsekwencją stosowania tej samej koncepcji organizacji usług i wspierających je zasad komunikacji. Każdą z warstw można scharakteryzować istnieniem równoważnych procesów w każdym z co najmniej dwóch przestrzennie rozproszonych punktów, które komunikują się, wykorzystując do tego celu usługi dostarczane przez warstwy niższe. Procesy realizowane w różnych punktach i należące do tej samej warstwy mają wspólny cel, którego warunkiem jest łączne i współbieżne wykonywanie tych procesów, polegających na rozproszonym przetwarzaniu i wzajemnym przekazywaniu informacji. Wykonywanie tych procesów oznacza istnienie i realizację rozproszonych algorytmów (ang. *distributed algorithm*) przetwarzania.

Każdy z rozproszonych algorytmów (protokołów) jest zbiorem algorytmów lokalnych realizowanych przez każdy z równoważnych procesów. Realizacja określonego algorytmu lokalnego oznacza wykonywanie operacji przetwarzania dostępnych danych, wysyłanie ich do jednego lub więcej równoważnych procesów oraz odbieranie danych wysłanych przez inne procesy. Efektywność i funkcjonalność każdego z rozproszonych algorytmów realizowanych w warstwach lub na styku warstw zależy od tego, w jaki sposób wykonywane i koordynowane są operacje algorytmów lokalnych.

W najprostszym przypadku kolejność operacji wykonywanych przez poszczególne algorytmy lokalne jest zdeterminowana, a w danej chwili czasu wykonywane są operacje tylko jednego algorytmu. Wykonywanie operacji danego algorytmu związane jest z pobieraniem i zwracaniem danych odpowiednio z i do jednego lub wielu algorytmów. Inicjatorem rozpoczęcia działania przez kolejny algorytm lokalny może być algorytm nadrzędny (wersja scentralizowana) lub bezpośrednio poprzedzający aktywny algorytm lokalny (wersja zdecentralizowana).

W bardziej złożonych przypadkach liczba jednocześnie wykonywanych algorytmów lokalnych może być większa niż jeden, z tym że w każdym z algorytmów lokalnych są predefiniowane warunki ich zatrzymywania oraz zbiory danych, których odebranie umożliwia kontynuowanie działania algorytmu.

W najbardziej złożonych przypadkach kolejność wykonywania poszczególnych algorytmów wyznaczana jest przez otrzymywane dane i ich semantykę.

3.2. HIERARCHIA JAKOŚCI USŁUG W MODELU WARSTWOWYM

Jakościowe i ilościowe wymagania różnych aplikacji sieciowych mogą być wyrażane za pomocą wartości parametrów jakości usług dostarczanych w sieciach. Speł-

nienie wymagań aplikacji, zgodnie z paradygmatem jakości usług, jest podstawowym kryterium oceny jakości dowolnej sieci i jej warstw oraz usług dostarczanych w tej sieci i jej warstwach.

Zasada obsługi aplikacji w systemie teleinformatycznym o architekturze warstwowej oznacza, że jakość usług dostarczanych aplikacji jest hierarchicznym złożeniem jakości usług dostarczanych w różnych warstwach sieci. Zadania projektowania i zarządzania sieci spełniających wymagania jakościowe i ilościowe aplikacji można rozumieć jako zadania odwzorowywania (translacji) wartości parametrów jakości usług (ang. *QoS parameter mapping*) pomiędzy wszystkimi sąsiednimi warstwami modelu sieci; w architekturze warstwowej ISO/OSI jest to odwzorowywanie jakości usług we wszystkich warstwach architektury (od warstwy aplikacji do warstwy fizycznej) i pomiędzy wszystkimi sąsiednimi warstwami.

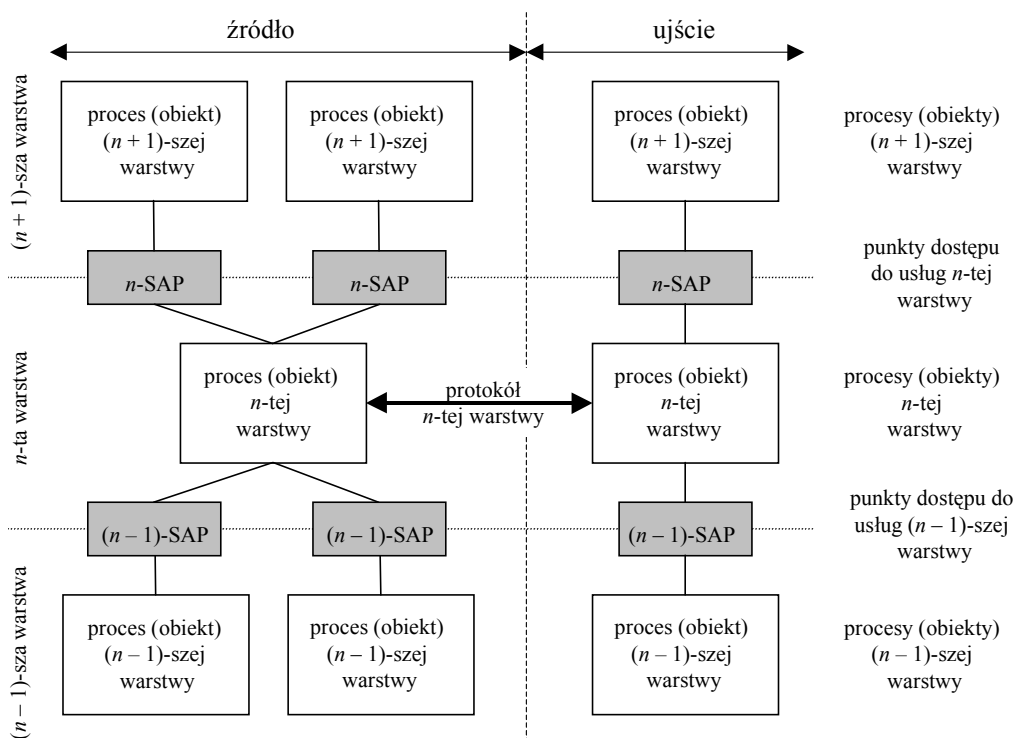
Odwzorowanie wartości parametrów jakości usług jest zadaniem wykonywanym na styku procesów realizowanych w dwóch współpracujących warstwach architektury (najczęściej sąsiednich), tzn. w punktach dostępu do usług warstw SAP (ang. *Service Access Point*). Odwzorowanie jest niezbędne w procesie przekształcania zadania transferu informacji w warstwie aplikacji w zadania transmisji, realizowanym w węźle źródłowym oraz w procesie odwrotnym, realizowanym w węźle odbiorczym (rysunek 3.1). Zadania takie wykonywane są także w węzłach pośrednich, zlokalizowanych na trasach pomiędzy węzłami źródłowymi i odbiorczymi. Zakres odwzorowania, realizowany w węzłach pośrednich, zależy od funkcji pełnionych przez węzeł pośredni. Na przykład w przypadku routera zakres odwzorowania obejmuje zwykle trzy najniższe warstwy sieci – jest determinowany zakresem funkcjonalnym routera.

Odwzorowanie wartości parametrów jakości usług w architekturze warstwowej jest zadaniem, którego celem jest znalezienie zależności pomiędzy:

- wartościami parametrów jakości usług w warstwie aplikacji a wartościami parametrów jakości usług w warstwach dostarczających usługi warstwie aplikacji, wynikającymi ze sposobów i z ilości zasobów udostępnianych w tych warstwach,
- narzutami na wartości parametrów jakości usług wnoszonymi przez procesy translacji usług z warstwy aplikacji do warstwy zasobów (warstwy fizycznej) oraz z warstwy zasobów do warstwy aplikacji.

Jedynym istotnym kryterium jakości usług sieci jest jakość usług dostarczanych w warstwie aplikacji, która może być złożeniem jakości usług dostarczanych warstwie aplikacji przez warstwy niższe w modelu hierarchicznym. Wartości parametrów jakości usług w warstwie aplikacji są sumą wartości parametrów jakości usług warstw niższych, które mogą być traktowane jako efekt różnego rozdziału zasobów (jedno- lub wielokrotnego) wykorzystywanych w różnych warstwach sieci. Wartości parametrów jakości usług w poszczególnych warstwach sieci są konsekwencją różnego rozdziału zasobów. Zadanie projektowania sieci, dla tak rozumianych wartości parametrów jakości usług, polega na znalezieniu dekompozycji wartości danych parametrów jakości usług warstwy pomiędzy wartości parametrów jakości usług warstw niższych. Tak sformułowane zadanie oznacza m.in., że dana wartość parametru jakości usług w danej warstwie sieci może

być różnie dekomponowana. Do wyboru najlepszej (optymalnej) dekompozycji spośród możliwych można stosować różne kryteria, np. efektywność wykorzystania zasobów.



Rysunek 3.1. Zadania odwzorowania wartości parametrów jakości usług w modelu warstwowym sieci

Zanim dane będą transferowane w podsystemie komunikacyjnym sieci teleinformatycznej konieczne jest poddanie ich różnym procedurom, takim jak segmentacja, łączenie, kodowanie, kompresja. Każda z tych procedur, we właściwy dla siebie sposób i w charakterystycznych ilościach, wykorzystuje części wartości parametrów jakości usług zadanych w warstwie aplikacji. Klasycznym przykładem wykorzystywania części wartości parametrów jakości usług, zdefiniowanych w warstwie aplikacji, jest opóźnienie wnoszone przez praktycznie każdą procedurę wykonywaną podczas przygotowywania danych do transferu i w czasie ich transferu.

Każda z omawianych procedur, którą można interpretować jako procedurę odwzorowywania usług pomiędzy warstwami sieci, pochłania część budżetu jakości usług (ang. *QoS budget*) definiowanego w warstwie aplikacji. Oznacza to dalej, że zmiana procedur jest równoważna ze zmianą pochłanianej części budżetu jakości usług. Dlatego też każdy algorytm odwzorowywania usług pomiędzy warstwami sieci oraz każdy protokół warstwy można interpretować jako sposoby ustalania i kontrolowania udziału poszczególnych warstw w współdzieleniu danego budżetu jakości usług.

3.3. PARAMETRY JAKOŚCI USŁUG

Do powszechnie stosowanych parametrów jakości usług, dostarczanych jednostkom danych SDU (ang. *Service Data Unit*) poszczególnych warstw w modelu warstwowym sieci, stosowane są m.in. następujące miary jakości usług: opóźnienie jednostki danych (ang. *delay*), zmienność opóźnienia jednostki danych (ang. *jitter*), zmienność czasu transmisji jednostki danych (ang. *deterministic jitter*), niezawodność (ang. *reliability*) oraz przepustowość (ang. *throughput*).

3.3.1. MAKSYMALNE OPÓŹNIENIE JEDNOSTEK DANYCH

Opóźnienie i -tej jednostki n -SDU w n -tej warstwie, oznaczone przez d_{ni} , jest długością przedziału czasu pomiędzy czasem generacji żądania dostępu dla jednostki n -SDU w punkcie n -SAP dostępu do usług n -tej warstwy w węźle nadawczym i czasem udostępnienia jednostki n -SDU w punkcie n -SAP dostępu do usług n -tej warstwy w węźle odbiorczym (rysunek 3.1).

Jednym ze podstawowych wskaźników jakości usługi n -tej warstwy jest maksymalne opóźnienie wnoszone przez protokół n -tej warstwy, oznaczane przez $d_{n \max}$, takie że:

$$d_{ni} \leq d_{n \max},$$

gdzie $i \in I$, a I jest danym zbiorem indeksów wszystkich jednostek danych (strumienia jednostek danych) n -SDU w n -tej warstwie.

3.3.2. ZMIENNOŚĆ OPÓŹNIENIA JEDNOSTEK DANYCH

Niezależnie od różnych definicji zmienności opóźnienia jest to miara chwilowych zmian opóźnienia jednostek n -SDU obsługiwanych w n -tej warstwie.

Jedną z definicji zmienności opóźnienia i -tej jednostki n -SDU w n -tej warstwie, oznaczonej przez j_{ni} , jest różnica maksymalnego opóźnienia $d_{n \max}$ jednostek n -SDU oraz opóźnienia danej i -tej jednostki n -SDU w n -tej warstwie:

$$j_{ni} = d_{n \max} - d_{ni}.$$

Zgodnie z tą definicją, maksymalna wartość zmienności opóźnienia jednostek n -SDU w n -tej warstwie spełnia warunek:

$$j_{ni} \leq j_{n \max} = \max_i j_{ni}.$$

Istnienie maksymalnej wartości zmienności opóźnienia $j_{n \max}$ jest równoważne istnieniu minimalnej wartości opóźnienia $d_{n \min}$ jednostek n -SDU w n -tej warstwie:

$$d_{n \min} = d_{n \max} - j_{n \max}.$$

3.3.3. ZMIENNOŚĆ CZASU TRANSMISJI JEDNOSTEK DANYCH

Opóźnienie jednostek danych n -SDU, wnoszone dynamicznie przez protokół n -tej warstwy, zależy od zmian w obciążeniu warstwy i czasów oczekiwania na transfer jednostek w kolejkach węzłów pośrednich.

Opóźnienie jednostek n -SDU jest przykładem parametru jakości usług, którego wartość nie jest znana a priori.

Przykładem parametru o znanej a priori wartości jest zmienność czasu transmisji jednostek n -SDU. Przy założeniu, że pojemność n -tej warstwy jest stała i wynosi C_n oraz że maksymalna długość jednostki n -SDU w tej warstwie wynosi $S_{n \max}$, maksymalny czas transferu jednostki n -SDU jest równy:

$$t_{n \max} = \frac{S_{n \max}}{C_n}.$$

Znajomość czasu transferu $t_{n \max}$ jednostki n -SDU o maksymalnej długości równej $S_{n \max}$ oraz czasu transferu i -tej jednostki n -SDU równego t_{ni} pozwala na definiowanie zmienności czasu transferu jednostek n -SDU w n -tej warstwie, oznaczonego przez T_{ni} , jako różnicy maksymalnego czasu transferu jednostki n -SDU i czasu transferu i -tej jednostki n -SDU w n -tej warstwie:

$$T_{ni} = t_{n \max} - t_{ni}.$$

3.3.4. NIEZAWODNOŚĆ

Z punktu widzenia aplikacji termin niezawodność określa skutki różnego rodzaju niepożądanych zdarzeń. Takie rozumienie niezawodności obejmuje m.in. konsekwencje dostarczenia uszkodzonych jednostek SDU lub niedostarczenia jednostek SDU.

Jeżeli niezawodność transferu jednostek n -SDU w n -tej warstwie utożsamiać z prawdopodobieństwem poprawnego transferu danych, to żądana niezawodność n -tej warstwy, oznaczana przez r_n , jest określona warunkiem:

$$r_n \geq p_{n \min},$$

w którym $p_{n \min}$ jest minimalną wartością prawdopodobieństwa poprawnego transferu jednostki danych n -SDU.

Poprawny transfer jednostki n -SDU oznacza brak straty danych (bitów) lub wystąpienie błędów danych (bitów) w jednostce n -SDU, tzn. strata co najmniej jednego bitu danych lub wystąpienie co najmniej jednego błędu elementarnego oznacza błędny transfer jednostki n -SDU.

Z powodów interpretacyjnych i obliczeniowych tak definiowaną niezawodność niekiedy wygodniej jest zastąpić górnym ograniczeniem prawdopodobieństwa błędnego transferu jednostki n -SDU – oznaczonym przez $\bar{r}_{n \max}$ – i równym:

$$\bar{r}_{n \max} = 1 - r_n = 1 - p_{n \min}.$$

3.3.5. PRZEPUSTOWOŚĆ

Ustalanie (negocjowanie) przepustowości (ang. *throughput*), pomiędzy n -tą a $(n - 1)$ -szą warstwą, polega na ustaleniu maksymalnej długości jednostki n -SDU w n -tej warstwie oraz minimalnej różnicy czasu pomiędzy dwoma kolejnymi żądaniami dostępu jednostek n -SDU do $(n - 1)$ -szej warstwy. Zgodnie z tym przepustowość $(n - 1)$ -szej warstwy definiuje się przez podanie maksymalnej długości jednostek danych warstwy nadrzędnej, tzn. warstwy n -SDU – $S_{n \max}$ oraz minimalnego czasu pomiędzy kolejnymi żądaniami obsługi jednostek – τ_n . Żądana przepustowość $(n - 1)$ -szej warstwy jest definiowana jako liczba żądań obsługi jednostek n -SDU o ustalonej długości w danym przedziale czasu.

Pojemność (ang. *bandwidth*) $(n - 1)$ -szej warstwy jest definiowana jako liczba jednostek $(n - 1)$ -SDU o ustalonej długości S_{n-1} , którą świadcząca usługi $(n - 1)$ -sza warstwa jest w stanie przyjąć w przedziale czasu o ustalonej długości.

Znajomość długości przedziału czasu obsługi jednostek SDU, długości jednostki SDU oraz minimalnej różnicy czasu pomiędzy dwiema kolejnymi jednostkami SDU jest tożsama ze znajomością maksymalnej liczby jednostek SDU obsługiwanych przez daną warstwę w danym przedziale czasu.

Strumienie CBR (ang. *Constant Bit Rate*) charakteryzowane są stałą długością jednostek i stałym czasem międzyzgłoszeniowym, charakterystyka strumieni VBR (ang. *Variable Bit Rate*) wymaga natomiast podania minimalnych oraz maksymalnych długości jednostek, których znajomość jest istotna dla prawidłowego odwzorowania wymagań pomiędzy warstwami.

3.4. ODWZOROWANIE PARAMETRÓW JAKOŚCI USŁUG

3.4.1. ODWZOROWANIE NIEZAWODNOŚCI

Sposób odwzorowania niezawodności jednostek danych $(n - 1)$ -SDU w $(n - 1)$ -szej warstwie w niezawodność jednostek danych n -SDU w n -tej warstwie zależy od sposobu, w jaki niezawodność jednostek danych $(n - 1)$ -SDU wpływa na niezawodność jednostek danych n -SDU, dlatego sposoby odwzorowania niezawodności mogą być różne.

Jeżeli prawdopodobieństwo błędnego transferu jednostek $(n - 1)$ -SDU w $(n - 1)$ -szej warstwie jest dane wartościami górnych ograniczeń prawdopodobieństwa straty danych oraz prawdopodobieństwa wystąpienia błędu danych w jednostce $(n - 1)$ -SDU w $(n - 1)$ -szej warstwie, to na podstawie tych wartości możliwe jest wyznaczenie wartości górnego ograniczenia prawdopodobieństwa błędnego transferu jednostki n -SDU w n -tej warstwie – $\bar{r}_{n \max}$.

W zależności od tego, jak straty jednostek $(n - 1)$ -SDU wpływają na niezawodność w n -tej warstwie, zasadne są różne sposoby odwzorowania.

Jeżeli strata danych w jednostce $(n - 1)$ -SDU ma taki sam wpływ na jakość usług, jak wystąpienie w jednostce $(n - 1)$ -SDU jedno- lub wielokrotnego błędu elementarnego, tzn. jeżeli w przypadku wystąpienia strat lub błędów jednostki $(n - 1)$ -SDU nie są przydatne, to może mieć zastosowanie następujące odzworowanie:

$$\bar{r}_{n \max} = \bar{r}_{(n-1)\max} = \bar{s}_{(n-1)\max} + \bar{b}_{(n-1)\max},$$

w którym:

$\bar{s}_{(n-1)\max}$ – górne ograniczenie prawdopodobieństwa straty co najmniej jednego bitu w jednostce danych $(n - 1)$ -SDU długości S_{n-1} ,

$$\bar{s}_{(n-1)\max} = 1 - (1 - p_{(n-1)s})^{S_{n-1}},$$

$\bar{b}_{(n-1)\max}$ – górne ograniczenie prawdopodobieństwa wystąpienia co najmniej jednego błędu elementarnego w jednostce danych $(n - 1)$ -SDU długości S_{n-1} ,

$$\bar{b}_{(n-1)\max} = 1 - (1 - p_{(n-1)e})^{S_{n-1}},$$

$p_{(n-1)s}$ – prawdopodobieństwo straty bitu w jednostce $(n - 1)$ -SDU,

$p_{(n-1)e}$ – prawdopodobieństwo błędu elementarnego w jednostce $(n - 1)$ -SDU.

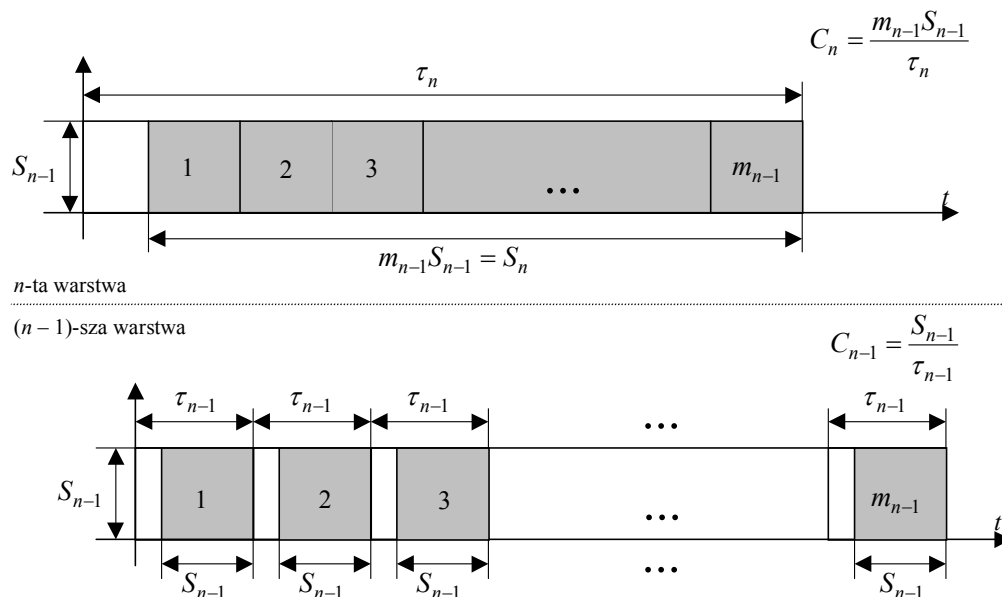
Inne odzworowanie można stosować wtedy, gdy strata jednostki $(n - 1)$ -SDU w $(n - 1)$ -szej warstwie jest kompensowana w n -tej warstwie przez wygenerowanie zastępczej jednostki $(n - 1)$ -SDU w celu zachowania integralności strumienia bitów. Jeżeli, w omawianym przypadku, stratę jednostki $(n - 1)$ -SDU można utożsamiać z wystąpieniem l ($1 \leq l \leq S_{n-1}$) błędów elementarnych, to może być stosowane następujące odzworowanie:

$$\bar{r}_{n \max} = l p_{(n-1)e} + \bar{b}_{(n-1)\max} \cong l p_{(n-1)e} + S_{n-1} p_{(n-1)e} = (l + S_{n-1}) p_{(n-1)e},$$

w którym: $\bar{b}_{(n-1)\max} = 1 - (1 - p_{(n-1)e})^{S_{n-1}} \cong S_{n-1} p_{(n-1)e}$, jeżeli $p_{(n-1)e} \ll 1$.

3.4.2. ODWZOROWANIA MIAR JAKOŚCI USŁUG W PROCESACH SEGMENTACJI I DESEGMENTACJI

Długość jednostek n -SDU w n -tej warstwie (warstwie nadrzędnej) może być większa od długości jednostek $(n - 1)$ -SDU w $(n - 1)$ -szej warstwie (warstwie podrzędnej), bezpośrednio świadczącej usługi dla n -tej warstwy. Wówczas w węzłach źródłowych konieczna jest segmentacja (ang. *segmentation*) jednostek n -SDU na jednostki $(n - 1)$ -SDU $(n - 1)$ -szej warstwy oraz odpowiednia rekonstrukcja (ang. *reassembly*) w węzłach odbiorczych jednostek n -SDU n -tej warstwy z jednostek $(n - 1)$ -SDU warstwy $(n - 1)$ -szej (rysunek 3.2).

Rysunek 3.2. Ogólny schemat podziału jednostki n -SDU na jednostki $(n-1)$ -SDU

Liczba m_{n-1} jednostek $(n-1)$ -SDU $(n-1)$ -szej warstwy, o ustalonej długości S_{n-1} , niezbędnych do transferu jednej jednostki n -SDU n -tej warstwy o stałej długości S_n , może być wyznaczona z zależności:

$$m_{n-1} = \left\lceil \frac{S_n}{S_{n-1}} \right\rceil.$$

Jeżeli długość jednostek n -SDU jest zmienna, to do wyznaczania liczby m_{n-1} stosowana jest maksymalna długość jednostek n -SDU – $S_{n \max}$:

$$m_{n-1} = \left\lceil \frac{S_{n \max}}{S_{n-1}} \right\rceil.$$

Zamiana jednostek n -SDU na jednostki $(n-1)$ -SDU podczas przechodzenia z n -tej warstwy do $(n-1)$ -szej warstwy i związana z tym zmiana liczby jednostek SDU ma wpływ na sposoby obliczania i wartości wskaźników jakości usług.

3.4.2.1. OPÓŹNIENIA

Jeżeli $d_{(n-1) \max}$ i $d_{n \max}$ oznaczają, odpowiednio, maksymalne opóźnienia wnoszone przez protokoły:

- $(n-1)$ -szej warstwy dla jednostek danych n -SDU,

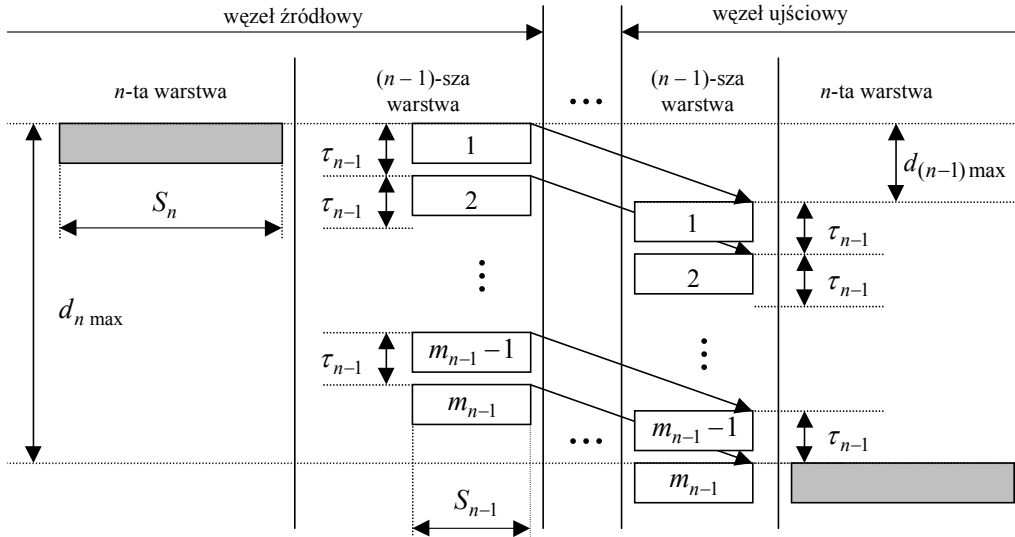
➤ n -tej warstwy dla jednostek danych $(n - 1)$ -SDU otrzymanych w wyniku segmentacji jednostek danych n -SDU,

to zależności pomiędzy opóźnieniami jednostek danych w omawianych warstwach dane są wyrażeniem (rysunek 3.3)

$$d_{n \max} = d_{(n-1) \max} + (m_{n-1} - 1)\tau_{n-1},$$

w którym:

τ_{n-1} – różnica czasu pomiędzy zgłoszeniami kolejnych jednostek $(n - 1)$ -SDU,
 $(m_{n-1} - 1)\tau_{n-1}$ – czas trwania procesu segmentacji jednej jednostki n -SDU w n -tej warstwie, dzielonej na m_{n-1} jednostek $(n - 1)$ -SDU w $(n - 1)$ -szej warstwie.



Rysunek 3.3. Zależności pomiędzy opóźnieniami jednostek n -SDU i $(n - 1)$ -SDU w procesie segmentacji

3.4.2.2. ZMIENNOŚĆ OPÓŹNIENIA

Przyjęcie założenia o stałej długości jednostek n -SDU i $(n - 1)$ -SDU oznacza, że czas oczekiwania na akceptację żądań obsługi m_{n-1} jednostek $(n - 1)$ -SDU w $(n - 1)$ -szej warstwie jest stały, tzn. opóźnienie wprowadzane przez segmentację jest stałe. W takim przypadku maksymalne wartości zmienności opóźnienia w warstwach n -tej i $(n - 1)$ -szej spełniają warunki:

$$\begin{aligned} j_{n \max} &= \max_i j_{ni} = \max_i (d_{n \max} - d_{ni}) = \\ &= \max_i ((d_{(n-1) \max} + (m_{n-1} - 1)\tau_{n-1}) - (d_{(n-1)i} + (m_{n-1} - 1)\tau_{n-1})) = j_{(n-1) \max}, \end{aligned}$$

tzn. że segmentacja, wprowadzająca opóźnienie o stałej wartości, nie ma wpływu na wartości zmienności opóźnienia.

Przyjęcie założenia, że jednostki n -SDU mają zmienną długość oznacza, że segmentacja powoduje zmienne opóźnienie, a tym samym zmienne maksymalne wartości zmienności opóźnienia. Jeżeli natomiast maksymalna i minimalna długości jednostek n -SDU wynoszą odpowiednio $S_{n \max}$ i $S_{n \min}$ oraz długość jednostki $(n - 1)$ -SDU jest stała i wynosi S_{n-1} , to zachodzą zależności:

$$\begin{aligned} j_{n \max} &= d_{n \max} - d_{n \min} = \\ &= (d_{(n-1) \max} - d_{(n-1) \min}) + (m_{(n-1) \max} - m_{(n-1) \min}) \tau_{n-1} = \\ &= j_{(n-1) \max} + \left(\left\lceil \frac{S_{n \max}}{S_{n-1}} \right\rceil - \left\lfloor \frac{S_{n \min}}{S_{n-1}} \right\rfloor \right) \tau_{n-1}. \end{aligned}$$

3.4.2.3. PRZEPUSTOWOŚĆ

W zależności od długości jednostek n -SDU i $(n - 1)$ -SDU segmentacja jednostek n -SDU może być przyczyną niepełnego wykorzystania jednostek $(n - 1)$ -SDU w $(n - 1)$ -szej warstwie. Jeżeli długość jednostki n -SDU – S_n – nie jest całkowitą wielokrotnością długości jednostki $(n - 1)$ -SDU – S_{n-1} , to rezultatem segmentacji jednostki n -SDU jest pełne wykorzystanie $m_{n-1} - 1$ jednostek $(n - 1)$ -SDU i częściowe wykorzystanie jednej jednostki $(n - 1)$ -SDU.

Przy założeniu dla przepływowego modelu ruchu, że:

➤ długość jednostki S_n jest całkowitą wielokrotnością długości jednostki S_{n-1} , tzn. $S_n = m_{n-1} S_{n-1}$,

➤ τ_n – różnica czasu pomiędzy kolejnymi zgłoszeniami jednostek n -SDU o długości S_n w n -tej warstwie (czas pomiędzy żądaniem obsługi dwóch kolejnych jednostek n -SDU),

➤ τ_{n-1} – różnica czasu pomiędzy kolejnymi zgłoszeniami jednostek $(n - 1)$ -SDU o długości S_{n-1} w $(n - 1)$ -szej warstwie,

z warunku równości żądanych przepustowości warstw:

$$\frac{S_n}{\tau_n} = \frac{m_{n-1} S_{n-1}}{\tau_n} = \frac{S_{n-1}}{\tau_{n-1}}$$

wynika, że zależności pomiędzy czasami międzyzgłoszeniowymi jednostek w omawianych warstwach są określone zależnością:

$$\tau_n = m_{n-1} \tau_{n-1}.$$

Jeżeli:

➤ $\tau_n < m_{n-1} \tau_{n-1}$, to przepustowość $(n-1)$ -szej warstwy jest mniejsza od żądanej; $(n-1)$ -sza warstwa wnosi dodatkowe opóźnienie jednostek n -SDU, spowalniając obsługę jednostek n -SDU,

➤ $\tau_n > m_{n-1} \tau_{n-1}$, to przepustowość $(n-1)$ -szej warstwy nie jest w pełni wykorzystana; wykorzystanie przepustowości $(n-1)$ -szej warstwy jest możliwe po zwiększeniu szybkości napływu jednostek n -SDU.

3.4.2.4. NIEZAWODNOŚĆ

Podział jednej jednostki n -SDU na wiele jednostek $(n-1)$ -SDU w procesie segmentacji wymusza odpowiednie odzworowanie wymagań niezawodnościowych.

Jeżeli jednostka n -SDU jest dzielona na m_{n-1} jednostek $(n-1)$ -SDU, to wartość górnego ograniczenia żądanej niezawodności w n -tej warstwie zależy od wartości górnego ograniczenia niezawodności w $(n-1)$ -szej warstwie. Celem zadania odzworowania niezawodności jest ustalenie wartości parametrów niezawodności w $(n-1)$ -szej warstwie, gwarantujących osiągnięcie zadanej niezawodności w n -tej warstwie.

Jeżeli przyjąć, że jednostka n -SDU jest obsługiwana bezbłędnie wtedy i tylko wtedy, gdy bezbłędnie obsługiwane są wszystkie jednostki $(n-1)$ -SDU, otrzymane z segmentacji jednostki n -SDU, to zależność pomiędzy górnymi ograniczeniami prawdopodobieństwa błędnego transferu jednostki n -SDU – $\bar{r}_{n \max}$ – oraz jednostki $(n-1)$ -SDU – $\bar{r}_{(n-1) \max}$ – jest określona wyrażeniem:

$$\bar{r}_{n \max} = 1 - (1 - \bar{r}_{(n-1) \max})^{m_{n-1}}.$$

Jeżeli założyć dalej, że:

➤ prawdopodobieństwo wystąpienia błędu elementarnego $p_{(n-1)e}$ w jednostce $(n-1)$ -SDU spełnia warunek: $p_{(n-1)e} \ll 1$,

➤ górne ograniczenie prawdopodobieństwa $\bar{r}_{(n-1) \max}$ błędnego transferu jednostki $(n-1)$ -SDU spełnia warunek: $\bar{r}_{(n-1) \max} \ll 1$,

to w najgorszym przypadku, kiedy wystąpienie przynajmniej jednego błędu elementarnego w jednej z m_{n-1} jednostek $(n-1)$ -SDU powoduje błąd jednostki n -SDU, zachodzą zależności:

$$\bar{r}_{(n-1) \max} = 1 - (1 - p_{(n-1)e})^{S_{n-1}} \cong S_{n-1} p_{(n-1)e},$$

$$\bar{r}_{n \max} = 1 - (1 - \bar{r}_{(n-1) \max})^{m_{n-1}} \cong m_{n-1} \bar{r}_{(n-1) \max} \cong m_{n-1} S_{n-1} p_{(n-1)e} \leq S_n p_{(n-1)e}.$$

Innym przykładem odzworowania niezawodności transferu jednostek n -SDU i $(n-1)$ -SDU, odpowiednio w warstwach n -tej i $(n-1)$ -szej jest przyjęcie jako miary niezawodności transferu prawdopodobieństwa straty jednostki (ang. SDU *loss rate*).

Zachowanie tej samej wartości prawdopodobieństwa strat jednostek $(n - 1)$ -SDU – lr_{n-1} oraz jednostek n -SDU – lr_n , tzn. $lr_{n-1} = lr_n$ oznacza, że żądana niezawodność w warstwie $(n - 1)$ -szej jest m_{n-1} większa od żądanej niezawodności w warstwie n -tej.

3.4.3. ODWZOROWANIA MIAR JAKOŚCI USŁUG W PROCESACH ŁĄCZENIA JEDNOSTEK DANYCH

Zadaniem przeciwnym do zadania segmentacji jest łączenie (ang. *blocking, concatenation*) wielu jednostek danych n -SDU n -tej warstwy (nadrzędnej) w jedną jednostkę danych $(n - 1)$ -SDU warstwy $(n - 1)$ -szej (podrzędnej). Przykładem zadania łączenia jednostek danych jest pakietyzacja próbek głosu. Pole danych pakietu zawiera wiele próbek głosu generowanych w stałych odstępach czasu. Łączenie próbek w pakiety jest przykładem agregacji ruchu, która wnosi dodatkowe i różne opóźnienia dla poszczególnych próbek wchodzących w skład pakietu.

Konsekwencje realizacji procesu łączenia kilku jednostek n -SDU w jednostki $(n - 1)$ -SDU są podobne do konsekwencji realizacji procesu segmentacji jednostek, z tym że o przeciwnych następstwach.

Przy założeniu, że:

- długość jednostek n -SDU w n -tej warstwie jest stała i równa S_n ,
- długość przedziałów czasu pomiędzy kolejnymi żądaniami dostępu jednostek n -SDU do $(n - 1)$ -szej warstwy jest stała i równa τ_n ,

m_n jednostek n -SDU jest łączonych w jedną jednostkę $(n - 1)$ -SDU o długości równej S_{n-1} , przy czym:

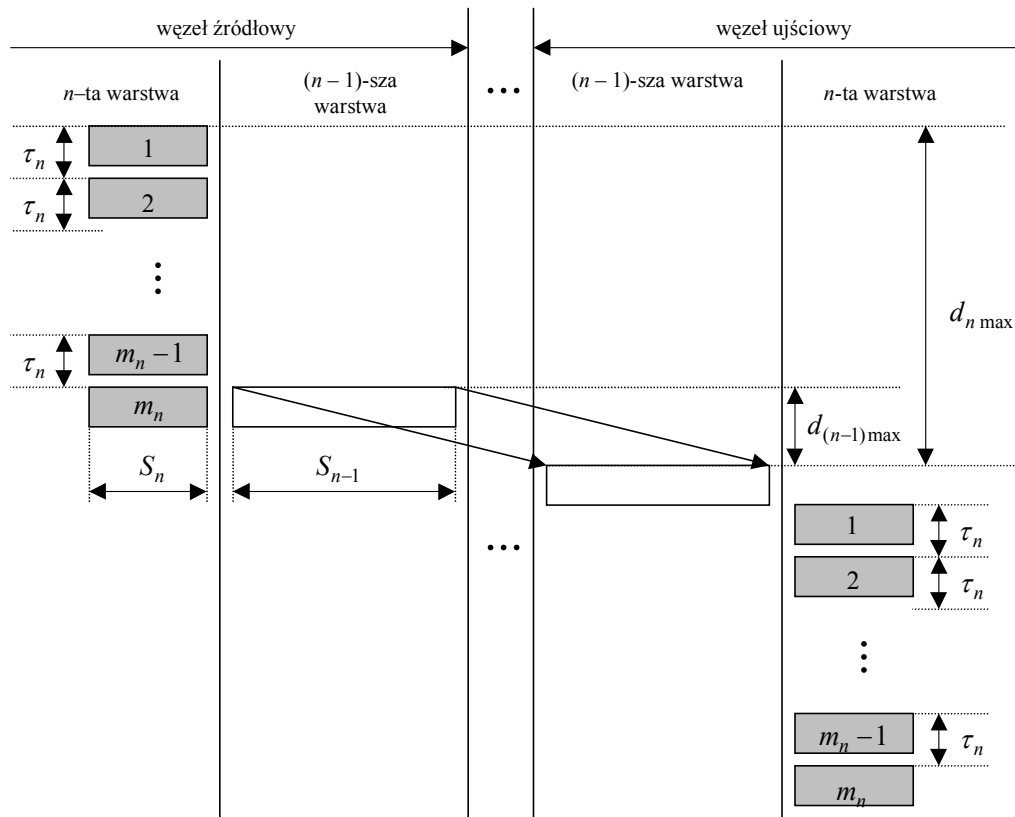
$$m_n = \left\lceil \frac{S_{n-1}}{S_n} \right\rceil.$$

Łączenie wielu jednostek n -SDU w jedną jednostkę $(n - 1)$ -SDU powoduje, że opóźnienie poszczególnych jednostek n -SDU, wchodzących w skład jednostki $(n - 1)$ -SDU, jest różne i zależy od miejsca zajmowanego w jednostce $(n - 1)$ -SDU.

3.4.3.1. OPÓŹNIENIA

Powodem różnego opóźnienia jednostek n -SDU w razie łączenia ich w jednostki $(n - 1)$ -SDU jest różny czas ich oczekiwania na skompletowanie jednostki $(n - 1)$ -SDU. W procesie łączenia jednostek maksymalne opóźnienie jest udziałem jednostki n -SDU, rozpoczynającej jednostkę $(n - 1)$ -SDU, która oczekuje na kolejne $(m_n - 1)$ jednostek n -SDU, niezbędnych do skompletowania jednostki $(n - 1)$ -SDU (rysunek 3.4). Wartość maksymalnego opóźnienia jednostki n -SDU, wnoszonego przez proces łączenia w $(n - 1)$ -szej warstwie, jest określona równaniem:

$$d_{n \max} = d_{(n-1) \max} + (m_n - 1)\tau_n.$$

Rysunek 3.4. Zależności pomiędzy opóźnieniami jednostek n -SDU i $(n-1)$ -SDU w procesie łączenia

3.4.3.2. ZMIENNOŚĆ OPÓŹNIENIA

Jeżeli wszystkie jednostki n -SDU mają jednakową długość, to różnice ich opóźnienia, wnoszone przez procedurę łączenia jednostek, wynikają z miejsca zajmowanego w jednostkach $(n-1)$ -SDU w czasie procesu łączenia jednostek n -SDU. Po oznaczeniu przez:

$d_{n \min}$ – opóźnienie jednostki n -SDU zajmującej ostatnią pozycję w jednostce $(n-1)$ -SDU (opóźnienie nie większe od maksymalnego opóźnienia jednostki $(n-1)$ -SDU),

$d_{n \max}$ – opóźnienie jednostki n -SDU zajmującej pierwszą pozycję w jednostce $(n-1)$ -SDU (opóźnienie nie większe od sumy maksymalnego opóźnienia jednostki $(n-1)$ -SDU i czasu oczekiwania na transmisję jednostki $(n-1)$ -SDU),

maksymalna zmienność opóźnienia jednostek n -SDU $j_{n \max}$ jest określona zależnością:

$$j_{n \max} = d_{n \max} - d_{n \min} = [d_{(n-1) \max} + (m_n - 1)\tau_n] - d_{(n-1) \max} = (m_n - 1)\tau_n.$$

3.4.3.3. PRZEPUSTOWOŚĆ

W przypadku łączenia jednostek różnica pomiędzy żądanymi: przepustowością n -tej warstwy oraz przepustowością $(n - 1)$ -szej warstwy wynika z tego, że wielokrotność długości S_n jednostek n -SDU może nie w pełni wykorzystywać długości S_{n-1} jednostki $(n - 1)$ -SDU, wówczas przepustowość $(n - 1)$ -szej warstwy musi być większa od przepustowości n -tej warstwy.

Jeżeli w przepływowym modelu ruchu:

➤ długość S_{n-1} jednostki $(n - 1)$ -SDU jest całkowitą wielokrotnością długości S_n jednostki n -SDU, tzn. $S_{n-1} = m_n S_n$,

➤ τ_n – różnica czasu pomiędzy kolejnymi zgłoszeniami jednostek n -SDU długości S_n w n -tej warstwie (czas pomiędzy żądaniem obsługi dwóch kolejnych jednostek n -SDU),

➤ τ_{n-1} – różnica czasu pomiędzy kolejnymi zgłoszeniami jednostek $(n - 1)$ -SDU długości S_{n-1} w $(n - 1)$ -szej warstwie,

to z warunku równości żądanych przepustowości warstw:

$$\frac{S_{n-1}}{\tau_{n-1}} = \frac{m_n S_n}{\tau_{n-1}} = \frac{S_n}{\tau_n}$$

wynika, że zależności pomiędzy czasami międzyzgłoszeniowymi jednostek w omawianych warstwach są określone zależnością:

$$\tau_{n-1} = m_n \tau_n.$$

Jeżeli:

➤ $\tau_{n-1} < m_n \tau_n$, to przepustowość $(n - 1)$ -szej warstwy jest nie jest w pełni wykorzystana,

➤ $\tau_{n-1} > m_n \tau_n$, to przepustowość n -tej warstwy nie jest dopasowana do przepustowości $(n - 1)$ -szej warstwy; $(n - 1)$ -sza warstwa wnosi dodatkowe opóźnienie.

3.4.3.4. NIEZAWODNOŚĆ

Jeżeli założyć, że utrata jednej jednostki $(n - 1)$ -SDU, na którą składa się m_n jednostek n -SDU, oznacza utratę m_n kolejnych jednostek n -SDU, to agregacja jednostek przenosi miary niezawodności pomiędzy warstwami. W takim przypadku jednostka n -SDU jest obsłużona bezbłędnie wtedy i tylko wtedy, gdy bezbłędnie obsłużona jest jednostka $(n - 1)$ -SDU; wartości górnych ograniczeń prawdopodobieństwa błędnego transferu jednostki n -SDU – $\bar{r}_{n \max}$ – oraz jednostki $(n - 1)$ -SDU – $\bar{r}_{(n-1) \max}$ – są równe:

$$\bar{r}_{n \max} = \bar{r}_{(n-1) \max}.$$

Innym przykładem odwzorowania niezawodności transferu jednostek n -SDU oraz $(n - 1)$ -SDU, odpowiednio w warstwach n -tej i $(n - 1)$ -szej, jest przyjęcie jako miary niezawodności transferu prawdopodobieństwa straty jednostki. Zachowanie tej samej

wartości prawdopodobieństwa utraty jednostek w warstwach n -tej i $(n - 1)$ -szej oznacza, że w razie agregacji jednostek n -SDU w jednostki $(n - 1)$ -SDU żądana niezawodność w n -tej warstwie lr_n jest m_n razy większa od niezawodności w $(n - 1)$ -szej warstwie lr_{n-1} .

Jeżeli założyć, że wystąpienie przynajmniej jednego błędu elementarnego w jednostce $(n - 1)$ -SDU powoduje błąd m_n jednostek n -SDU, to zadaniem odwzorowania niezawodności jest ustalenie takich wartości parametrów niezawodności w $(n - 1)$ -szej warstwie, które gwarantują osiągnięcie zadanej niezawodności w n -tej warstwie.

Jeżeli wystąpienie dokładnie jednego błędu elementarnego w jednostce $(n - 1)$ -SDU jest równoznaczne z wystąpieniem dokładnie jednego błędu elementarnego w jednej z m_n jednostek n -SDU, to zachowanie niezawodności w procesie łączenia jednostek wymaga spełnienia warunku:

$$S_{n-1} P_{(n-1)e} (1 - p_{(n-1)e})^{S_{n-1}-1} = m_n \left[S_n P_n (1 - p_{ne})^{S_n-1} \right] \left[(1 - p_{ne})^{S_n} \right]^{m_n-1},$$

z którego można wyznaczyć zależności pomiędzy wartościami prawdopodobieństw wystąpienia błędów elementarnych p_{ne} i $p_{(n-1)e}$.

Jeżeli wystąpienie co najmniej jednego błędu elementarnego w jednostce $(n - 1)$ -SDU jest równoznaczne z wystąpieniem co najmniej jednego błędu elementarnego w jednej z m_n jednostek n -SDU, to zachowanie niezawodności w procesie łączenia jednostek wymaga spełnienia warunku:

$$\bar{r}_{n \max} = \bar{r}_{(n-1) \max},$$

w którym:

$$\bar{r}_{(n-1) \max} = 1 - (1 - p_{(n-1)e})^{S_{n-1}} \cong S_{n-1} P_{(n-1)e},$$

$$\bar{r}_{n \max} = m_n \left[1 - (1 - p_{ne})^{S_n} \right] (1 - p_{ne})^{S_n-1} \cong m_n S_n (1 - p_{ne})^{S_n-1}.$$

3.4.3.5. KOMPENSOWANIE OPÓŹNIENIA

Jednym ze sposobów zmniejszania wartości zmienności opóźnienia jest zastosowanie buforów odtwarzania (ang. *playout buffer*). Jednostki $(n - 1)$ -SDU wyprowadzane są periodycznie z bufora odtwarzania do n -tej warstwy ze stałą szybkością. Długość przedziału czasu pomiędzy kolejnymi transferami jednostek $(n - 1)$ -SDU do n -tej warstwy jest stała i wynosi τ_n . Konsekwencją stosowania dowolnego mechanizmu zmniejszania wartości zmienności opóźnienia $j_{(n-1) \max}$ jest wzrost wartości średniego opóźnienia jednostek $(n - 1)$ -SDU $d_{(n-1)av}$, a w szczególnych przypadkach także wzrost wartości maksymalnego opóźnienia $d_{(n-1) \max}$.

Jednym z prostszych sposobów kompensacji zmienności opóźnienia jest rozwiązanie polegające na opóźnieniu pierwszej jednostki $(n - 1)$ -SDU o maksymalną wartość opóźnienia bufora odtwarzania $d_{(n-1)B \max}$ i wyprowadzaniu z bufora kolejnych

jednostek $(n - 1)$ -SDU do n -tej warstwy po upływie kolejnych przedziałów czasu długości τ_n . Procedura taka wymusza zwiększenie wartości maksymalnego opóźnienia $d_{n \max}$ i zmniejszenie wartości maksymalnej zmienności opóźnienia $j_{n \max}$ jednostek n -SDU w porównaniu z wartościami maksymalnego opóźnienia i maksymalnej zmienności opóźnienia w $(n - 1)$ -szej warstwie:

$$d_{n \max} = d_{(n-1) \max} + d_{(n-1)B \max},$$

$$j_{n \max} = j_{(n-1) \max} - d_{(n-1)B \max}.$$

W ogólnym przypadku pojemność bufora odtwarzania B_{n-1} zależy od długości jednostki $(n - 1)$ -SDU S_{n-1} , wartości kompensowanej zmienności opóźnienia \bar{j}_{n-1} oraz czasu międzyzłogłoszeniowego τ_{n-1} :

$$B_{n-1} = \frac{2\bar{j}_{n-1}}{\tau_{n-1}} S_{(n-1)}.$$

Ograniczeniem stosowania bufora odtwarzania do ograniczania wartości zmienności opóźnienia jest dodatkowe opóźnienie jednostek wnoszone przez bufor.

Istotne znaczenie ma sposób obsługi pierwszej jednostki SDU przyjmowanej do bufora odtwarzania – sposób obsługi pierwszej jednostki zależy od charakteru zmienności opóźnienia.

Gdy zmienność opóźnienia jest deterministyczna, wówczas znane są opóźnienia pierwszych napływających do bufora odtwarzania jednostek SDU. Jeżeli pierwsza napływająca jednostka SDU ma opóźnienie minimalne, to dodatkowe jej opóźnienie w buforze odtwarzania jest równe maksymalnemu opóźnieniu tego bufora. Jeżeli pierwsza napływająca jednostka SDU ma opóźnienie maksymalne, to jednostka taka nie jest dodatkowo opóźniana w buforze odtwarzania. W przeciwnym razie, tzn. wtedy, gdy opóźnienie pierwszej napływającej jednostki SDU nie jest znane, jedynym rozwiązaniem jest jej maksymalne opóźnianie w buforze odtwarzania.

Kompensacja zmienności opóźnienia statystycznego jest zadaniem trudniejszym od kompensacji zmienności opóźnienia deterministycznego, w którym opóźnienie pierwszych napływających jednostek SDU jest znane.

W przypadku kompensowania deterministycznej zmienności opóźnienia jednostek n -SDU, wnoszonej przez procedurę łączenia tych jednostek w jednostki $(n - 1)$ -SDU, pojemność bufora odtwarzania w $(n - 1)$ -szej warstwie B_{n-1} zależy od wartości: długości jednostki $(n - 1)$ -SDU S_{n-1} , wyrażonej wielokrotnością długości jednostek n -SDU n -tej warstwy S_n , kompensowanej zmienności opóźnienia \bar{j}_n ($\bar{j}_n \leq j_{n \max}$) oraz czasu międzyzłogłoszeniowego τ_n :

$$B_{n-1} = \frac{\bar{j}_n}{\tau_n} S_n.$$

Jednym z rozwiązań zadania kompensacji zmienności opóźnień w omawianym przypadku jest procedura, zgodnie z którą kolejna i -ta jednostka n -SDU ($1 \leq i \leq m_n$) o opóźnieniu d_{ni} ($d_{n \min} \leq d_{ni} \leq d_{n \max}$), przekazywana w jednostce $(n - 1)$ -SDU, jest opóźniana dodatkowo w buforze odtwarzania zgodnie z regułą:

$$\bar{J}_{ni} = d_{n \max} - d_{ni}.$$

3.4.4. ODWZOROWANIA MIAR JAKOŚCI USŁUG W PROCESACH UZUPEŁNIANIA JEDNOSTEK DANYCH

Proces przekazywania danych z warstwy nadrzędnej do warstwy podrzędnej jest zawsze związany z wykorzystaniem kolejnego protokołu, a tym samym z wprowadzaniem dodatkowego narzutu w postaci nagłówków, pól adresowych, pól sterowania, pól kontrolnych itp.

Konsekwencją dodatkowego narzutu jest zwiększenie długości jednostek danych w procesie przechodzenia z n -tej warstwy do $(n - 1)$ -szej warstwy. Przyrost długości jednostek powoduje zmiany w opóźnieniach jednostek oraz wymagań dotyczących przepustowości i niezawodności warstw.

3.4.4.1. OPÓŹNIENIA

Konwersja jednostek n -SDU w jednostki $(n - 1)$ -SDU i odwrotnie wnosi dodatkowe opóźnienie. W ogólnym przypadku czas konwersji jednostek jest dla danych warstw stały. Czas konwersji jednostki n -SDU jest składową opóźnienia wnoszonego przez daną warstwę $d_{n \max}$.

3.4.4.2. PRZEPUSTOWOŚĆ

Uzupełnianie jednostek n -SDU o narzut, determinowany protokołem $(n - 1)$ -szej warstwy, oznacza, że długość jednostki $(n - 1)$ -SDU, oznaczona przez S_{n-1} , jest sumą długości jednostki n -SDU S_n oraz długości narzutu O_{n-1} :

$$S_{n-1} = S_n + O_{n-1}.$$

Zmiana długości jednostki n -SDU przy jej zamianie na jednostkę $(n - 1)$ -SDU wymaga dopasowania przepustowości warstw. W najprostszym przypadku, kiedy jedna jednostka n -SDU jest zamieniana w jedną jednostkę $(n - 1)$ -SDU, przyrost żądanej przepustowości jest wprost proporcjonalny do wartości ilorazu:

$$\frac{O_{n-1} + S_n}{S_n}.$$

3.4.4.3. NIEZAWODNOŚĆ

Do konwersji jednostek n -SDU w jednostki $(n - 1)$ -SDU konieczna jest translacja wymagań niezawodnościowych. Translacja ta zależy od tego, jak protokół $(n - 1)$ -szej

warstwy jest wrażliwy na straty i błędy w porównaniu z wrażliwością na straty i błędy protokołu n -tej warstwy.

Jednym z możliwych odwzorowań niezawodności jest odwzorowanie po przyjęciu założenia, że protokoły warstw są jednakowo wrażliwe na występowanie błędów elementarnych. Jeżeli $p_{(n-1)e}$ i p_{ne} są prawdopodobieństwami wystąpienia błędów elementarnych, spełniającymi warunki $p_{(n-1)e} \ll 1$ i $p_{ne} \ll 1$, to warunkiem równości górnych ograniczeń prawdopodobieństw błędnego transferu jednostek n -SDU oraz $(n-1)$ -SDU, oznaczonych odpowiednio przez $\bar{r}_{n \max}$ i $\bar{r}_{(n-1) \max}$, określonych wyrażeniami:

$$\bar{r}_{n \max} = 1 - (1 - p_{ne})^{S_n} \cong p_{ne} S_n,$$

$$\bar{r}_{(n-1) \max} = 1 - (1 - p_{(n-1)e})^{(S_n + O_{n-1})} \cong p_{(n-1)e} (S_n + O_{n-1}),$$

jest następująca zależność pomiędzy prawdopodobieństwami wystąpienia błędów elementarnych:

$$p_{(n-1)e} = p_{ne} \frac{S_n}{S_n + O_{n-1}}.$$

3.4.5. ODWZOROWANIA MIAR JAKOŚCI USŁUG W PROCESACH KODOWANIA I KOMPRESJI

Podstawą różnych standardów kodowania i kompresji sygnału są rodzaje i ilości nadmiarowości występujących w danym sygnale. Na przykład w sygnale wideo wydrebniane są cztery rodzaje nadmiarowości: przestrzenna (wartości pikseli sąsiadujących ze sobą w jednym obrazie nie są niezależne), czasowa (istnieje korelacja pomiędzy wartościami pikseli o tym samym położeniu przestrzennym w sąsiednich obrazach), statystyczna (istnieją w obrazie wartości pikseli występujące częściej niż inne) i subiektywna (ograniczona zdolność oka do postrzegania szczegółów i śledzenia ruchu).

Różne algorytmy kompresji pozwalają na ograniczenie lub usunięcie wszystkich czterech rodzajów nadmiarowości. Powszechnie stosowanym standardem kodowania obrazów ruchomych jest standard MPEG-2 (ang. *Moving Picture Expert Group*), definiujący cyfrową transmisję sygnału wizyjnego dla przepustowości łącza w zakresie $4 \div 60$ Mb/s. Standard pozwala na uzyskanie współczynników kompresji rzędu 40:1.

Zgodnie ze standardem MPEG-2 obraz po kompresji może być reprezentowany przez jedną z trzech ramek:

➤ ramkę interkodowaną – typu I (ang. *interframe coding*), charakteryzującą się minimalną kompresją; ramki te nie mają powiązań z żadną z ramek,

➤ ramkę z predykcją ruchu – typu P (ang. *predictive coding*); algorytm kompresji tych ramek opiera się na ortonormalnych dyskretnych transformatach kosinusowych DCT (ang. *Discrete Cosine Transform*) i tzw. kompensacji ruchu, co sprawia, iż ramki typu P są powiązane z wcześniejszymi ramkami,

➤ ramkę interpolowaną – typu *B* (ang. *bidirectional coding*), charakteryzującą się największym stopniem kompresji. Kompresja jest oparta na DCT i dwukierunkowej kompensacji ruchu wykorzystującej podobieństwo bloków w dwóch obrazach odniesienia: wcześniejszym i późniejszym.

Ramki typu *I* są kodowane bez odniesienia do innych ramek, ramki typu *P* są kodowane jako różnica obrazów z ostatnich ramek typu *I* oraz *P*, natomiast ramki typu *B* są kodowane jako różnice interpolacji poprzedzających i późniejszych ramek typu *I* lub *P*. Procedura kodowania ramki typu *B* wnosi dodatkowe opóźnienie – wymaga oczekiwania na przyszłe ramki typu *I* lub *P*.

Kodowanie w standardzie MPEG (podobnie jak w każdym innym standardzie), realizowane w jednej z warstw, wnosi dodatkowe opóźnienie i dodatkową zmienność opóźnienia. Dodatkowe opóźnienie wynika ze sposobu wyznaczania ramek różnego typu, a dodatkowa zmienność opóźnienia jest powodowana czasem oczekiwania na ramki odniesienia w przypadku wyznaczania ramek typu *B*.

3.4.5.1. OPÓŹNIENIA

Opóźnienia wnoszone przez procedurę kodowania w standardzie MPEG są sumą czasu oczekiwania na ramki odniesienia i czasu przetwarzania:

➤ wartość maksymalnego opóźnienia $d_{\text{MPEG max}}$ jest opóźnieniem wnoszonym przez kodowanie ramki typu *B*; zależy od maksymalnego czasu przetwarzania niezbędnego do kodowania ramki interpolowanej $t_{f \text{ max}}$, maksymalnej odległości kodowanej ramki typu *B* od ramek odniesienia l_f oraz szybkości generowania ramek v_f :

$$d_{\text{MPEG max}} = \frac{l_f}{v_f + t_{f \text{ max}}},$$

➤ wartość minimalnego opóźnienia $d_{\text{MPEG min}}$ jest opóźnieniem wnoszonym przez kodowanie ramki typu *P*; w tym przypadku czas oczekiwania na ramki odniesienia jest równy zeru:

$$d_{\text{MPEG min}} = t_{f \text{ min}}.$$

3.4.5.2. ZMIENNOŚĆ OPÓŹNIENIA

Maksymalna wartość zmienności opóźnienia, wnoszonego przez kodowanie w standardzie MPEG, jest różnicą pomiędzy maksymalnym opóźnieniem, wnoszonym przez kodowanie ramki typu *B*, i minimalnym opóźnieniem, będącym następstwem kodowania ramki typu *P*:

$$j_{\text{MPEG max}} = d_{\text{MPEG max}} - d_{\text{MPEG min}} = \frac{l_f}{v_f} + t_{f \text{ max}} - t_{f \text{ min}}.$$

Dla stałych wartości czasu wyznaczania ramek i szybkości ich generowania wartość maksymalnej zmienności opóźnienia rośnie wraz ze wzrostem odległości ramek typu *B* i ramek odniesienia, na podstawie których wyznaczana jest ramka typu *B*.

3.4.5.3. PRZEPUSTOWOŚĆ

Podstawowym celem kompresji jest redukcja zapotrzebowania na przepustowość; stopień redukcji zapotrzebowania jest dany wartością współczynnika kompresji.

Redukcja zapotrzebowania na przepustowość w wyniku kompresji odbywa się kosztem zwiększenia opóźnienia i wzrostu jego zmienności. Stopień redukcji zapotrzebowania na przepustowość zależy od transmitowanego sygnału i użytych wzorców GOP (ang. *Group of Pictures*) – deterministycznych struktur, w jakie grupowane są skompresowane ramki.

Jeżeli procedura kompresji nie gwarantuje minimalnej wartości współczynnika kompresji, to stopień redukcji poszczególnych jednostek SDU nie może być stosowany do wyznaczenia deterministycznej wartości żądanej przepustowości.

Jeżeli wynikiem kompresji kolejnych jednostek n -SDU o długości S_{ni} ($i = 1, 2, \dots$) są jednostki $(n - 1)$ -SDU o długościach $S_{(n-1)i}$, to zależność długości tych jednostek, stosowana do szacowania żądanej przepustowości $(n - 1)$ -szej warstwy, jest dana zależnością:

$$S_{n-1} = S_n \max_i \frac{S_{(n-1)i}}{S_{ni}} = S_n c_{\min},$$

w której c_{\min} jest minimalną wartością współczynnika kompresji jednostek n -SDU w n -tej warstwie.

3.4.5.4. NIEZAWODNOŚĆ

Zastosowanie kompresji jednostek n -SDU w jednostki $(n - 1)$ -SDU wymaga konwersji wymagań niezawodnościowych. Niezawodność wymagana do przekazywania skompresowanych ramek w $(n - 1)$ -szej warstwie jest na ogół wyższa od niezawodności n -tej warstwy: strumień ramek MPEG jest bardziej wrażliwy na występowanie błędów i strat niż strumień oryginalnych ramek.

Jednym z możliwych sposobów konwersji wymagań niezawodnościowych jest wykorzystanie minimalnej wartości współczynnika kompresji do wyznaczania żądanej niezawodności w $(n - 1)$ -szej warstwie, gdy dana jest niezawodność w n -tej warstwie.

3.4.6. ODWZOROWANIA MIAR JAKOŚCI USŁUG W PROCESACH WYGŁADZANIA

Duże zapotrzebowanie na pojemność, charakterystyczne dla ruchu VBR (ang. *Variable Bit Rate*), można redukować przez stosowanie procedur wygładzania (ang. *pik smoothing*). Wygładzanie jednostek n -SDU oznacza, że jednostka n -SDU nie jest transferowana w całości w żądanym przedziale czasu τ_n , ale w czasie będącym wielokrotnością τ_n .

Wynikiem wygładzania jednostek n -SDU o maksymalnej długości $S_{n \max}$ są jednostki $(n - 1)$ -SDU o maksymalnej długości $S_{(n-1) \max}$. Jeżeli czasy międzyzgłoszenio-

we jednostek n -SDU i $(n - 1)$ -SDU są takie same, tzn. $\tau_n = \tau_{(n-1)}$, to w zależności od „wybuchowości” źródła jednostek n -SDU i wielokrotności przedziału czasu τ_n przeznaczonego na wygładzanie, maksymalna długość jednostki $(n - 1)$ -SDU $S_{(n-1) \max}$ może być zredukowana do średniej długości jednostek n -SDU $S_{n \text{ av}}$.

3.4.6.1. OPÓŹNIENIA

Wpływ procesu wygładzania jednostek n -SDU na opóźnienie jest podobny do wpływu procesu łączenia wielu jednostek n -SDU w jedną jednostkę $(n - 1)$ -SDU o długości będącej wielokrotnością długości jednostki n -SDU.

Zakładając, że jednostka n -SDU, o maksymalnej długości $S_{n \max}$, jest transferowana w $(n - 1)$ -szej warstwie w postaci jednostek $(n - 1)$ -SDU o maksymalnej długości $S_{(n-1) \max}$ ($S_{(n-1) \max} = S_{n \text{ av}}$), maksymalne opóźnienie wnoszone przez procedurę wygładzania jest określone wzorem:

$$d_{n \max} = d_{(n-1) \max} + (m_n - 1) \tau_n,$$

w którym $m_n = \frac{S_{n \max}}{S_{n \text{ av}}}$.

3.5. PODSUMOWANIE

Hierarchiczna i warstwowa architektura sieci teleinformatycznej jednoznacznie definiuje sposoby dostarczania i zapewniania zasady odzworowania jakości usług. Jednoznaczność oznacza tu m.in. możliwość dokładnego odzworowania wartości parametrów jakości usług pomiędzy dwiema dowolnymi warstwami modelu.

Z realizacją zadań wykonywanych w poszczególnych zagnieżdżonych warstwach modelu sieci teleinformatycznej związane są zmiany wartości różnych parametrów charakteryzujących przepływy i jakość ich obsługi. Uwzględnienie zmian, wprowadzanych przez poszczególne warstwy, polega albo na kompensowaniu zmian przez zwiększanie zasobów niwelujących ich skutki, albo na takim formułowaniu, które zakłada wprowadzanie zmian wnoszonych przez poszczególne warstwy.

Celem odzworowywania wartości parametrów jakości usług warstw w architekturach warstwowych jest poszukiwanie zależności pomiędzy jakością usług różnych warstw i protokołów implementowanych w różnych warstwach oraz transformacja wymagań, formułowanych w warstwie aplikacji, na wymagania w różnych warstwach wspierających funkcje warstwy aplikacji.

Odzworowanie wartości parametrów jakości usług jest stosowane m.in. w następujących zadaniach:

- formułowania wymagań z zakresu jakości usług dla warstw niższych na podstawie znajomości wymagań z zakresu jakości usług w warstwach wyższych, zwłaszcza w warstwie aplikacji (zadanie dekompozycji jakości usług),

- predykcji jakości usług dostarczanych w warstwach wyższych na podstawie jakości usług dostarczanych w warstwach niższych (zadanie syntezy jakości usług),
- projektowania sposobów dostarczania zadanej jakości usług, tzn. optymalizacji homogenicznych i heterogenicznych hierarchii protokołów dla zadanych wskaźników jakości usług w architekturach warstwowych (zadanie optymalizacji).

Ze względu na ogólne właściwości systemów teleinformatycznych o architekturach warstwowych, charakteryzujących się m.in. tym, że:

- nadrzędnym kryterium oceny jakości usług dostarczanych w sieci jest jakość usług dostarczanych w warstwie aplikacji,
- jakość usług dostarczanych w warstwie aplikacji jest rezultatem złożenia jakości usług dostarczanych w różnych niższych warstwach architektury,
- ta sama żądana jakość usług w warstwie wyższej może być uzyskana przez złożenie usług różnej jakości, dostarczanych przez różne niższe warstwy,
- koszt uzyskania usług o żądanej jakości w warstwie aplikacji zależy od liczby i lokalizacji warstw, w których udostępniane są usługi składowe,

zadanie odwzorowania wartości parametrów jakości usług jest niezbędnym elementem każdego zadania projektowania i analizy systemu teleinformatycznego.

LITERATURA

- [1] ATKINS J., NORRIS M., *Total Area Networking*, Wiley, Chichester 1995.
- [2] BATYCKI T., GRZECH A., *A stochastic models of network's mode performance and transmission demand in computer communication network*, Proceedings of the 12th Hawaii International Conference on System Sciences, Vol. 2, Honolulu 1979, s. 280–288.
- [3] BERTSEKAS D., GALLAGER R., *Data networks*, Prentice-Hall International, Englewood Cliffs 1987.
- [4] BŁAŻEWICZ J., CELLARY W., SŁOWIŃSKI R., WĘGLARZ J., *Badania operacyjne dla informatyków*, WNT, Warszawa 1983.
- [5] CAMPBELL A., COULSON G., GARCIA F., HUTCHISON D., LEOPOLD H., *Integrated quality of service for multimedia communications*, Lancaster University (www.itpapers.com).
- [6] CHODOREK A., *Metoda predykcji ruchu wideo MPEG-2 z wykorzystaniem linearyzowanej dekompozycji w przestrzeni fazowej*, (rozprawa doktorska), Kraków 2001.
- [7] CZACHÓRSKI T., *Modele kolejkowe w ocenie efektywności sieci i systemów komputerowych*, Pracownia Komputerowa Jacka Skalmierskiego, Gliwice 1999.
- [8] ELBAUM R., SIDI M., *Topological design of local-area networks using genetic algorithms*, IEEE Transactions on Networking, Vol. 4, No. 5, s. 766–778, October 1996.
- [9] FRANK H., FRISCH I.T., *Communication, transmission and transportation networks*, Addison-Wesley, Reading 1971.
- [10] FROST V.S., MELAMED B., *Traffic modelling for telecommunications networks*, IEEE Communications Magazine, Vol. 32, No. 3, March 1994, s. 70–81.

- [11] GERLA M., KLEINROCK L., *On the topological design of distributed computer networks*, IEEE Transactions on Communications, Vol. COM-25, 1977, s. 48–60.
- [12] GŁOWACKI M., *Measurement of QoS parameters of packet-switched Wide Area Networks*, Proceedings of 6th Polish Teletraffic Symposium, Szklarska Poręba 1999, s. 128–142.
- [13] GRZECH A., *Performance comparison of single and integrated local area networks*, Systems Science, Vol. 14, No. 4, 1988, s. 53–68.
- [14] GRZECH A., *Zbiór rozłączonych prostych podsystemów komunikacyjnych jako złożony podsystem komunikacyjny lokalnej sieci komputerowej*, Archiwum Automatyki i Telemechaniki, t. 35, z. 3, 4, 1990, s. 277–296.
- [15] GRZECH A., *Group hybrid access-control discipline for multi-access channels. A throughput analysis*, Systems Anal., Vol. 7, No. 4, 1990, s. 267–287.
- [16] GRZECH A., *Procedures of designing enterprise network communication subsystems*, Cybernetics and Systems, Vol. 31, No. 5, 2000, s. 531–545.
- [17] HARMAS D.D., KRAETZL M., COLBOURN C.J., DEVITT J.S., *Network reliability*, CRC Press, Boca Raton 1995.
- [18] JAFFE J.M., *Flow control power is nondecentralizable*, IEEE Transactions on Communications, Vol. COM-29, No. 9, September 1981, s. 1301–1306.
- [19] KLEINROCK L., *Communication nets*, Volume I: *Stochastic message flow and delay*, Volume II: *Computer Applications*, Wiley, New York 1976.
- [20] KNOCHE H., MEER H., *Quantitative QoS-mapping: a unified approach*, University of Hamburg 2001.
- [21] KULKARNIL A., LI A.Q., *Performance analysis of a rate-based feedback control scheme*, IEEE/ACM Transactions on Networking, Vol. 6, No. 6, December 1998, s. 797–810.
- [22] KUMMERLE K., LIMB J.O., TOBAGI F.A., *Advances in Local Area Networks*, IEEE Press, New York 1986.
- [23] LEUE S., OECHSLIN P.A., *On parallelizing and optimising the implementation of communication protocols*, IEEE Transactions on Networking, Vol. 4, No. 1, February 1996, s. 55–70.
- [24] PAXSON V., *End-to-end Internet packet dynamics*, IEEE/ACM Transactions on Networking, Vol. 7, No. 3, June 1999, s. 277–292.
- [25] POJOLLE G., SERET D., DROMARD D., HORLAI E., *Integrated digital communications networks*, Wiley, Chichester 1988.
- [26] SEIDLER J., *Analiza i synteza sieci łączności dla systemów teleinformatycznych*, PWN, Warszawa 1979.
- [27] TAKAGI H. (red.), *Stochastic analysis of computer and communication systems*, North-Holland, Amsterdam 1990.
- [28] WANG Z., *Internet QoS: Architectures and mechanisms for Quality of Service*, Academic Press, London, 2001.
- [29] WINCH R.G., *Telecommunication Transmission Systems*, McGraw-Hill Telecommunications, New York 1998.
- [30] WOŹNIAK J., *Analiza i projektowanie protokołów komunikacyjnych dla radiowych sieci teleinformatycznych*, Wydawnictwo PG, Gdańsk 1991.

4. ZADANIA KOMUNIKACJI W SIECIACH O ARCHITEKTURACH WARSTWOWYCH

Zasada warstwowości i zagnieżdżenia funkcji, powszechnie wykorzystywana w architekturach rozproszonych systemów teleinformatycznych, jest stosowana we wszystkich architekturach i scenariuszach dostarczania jakości usług. Oznacza to, że tę samą – z punktu widzenia aplikacji – jakość usług można uzyskać przez implementację różnych zbiorów współpracujących i uzupełniających się procedur.

4.1. ZADANIA KOMUNIKACJI W ARCHITEKTURZE WARSTWOWEJ

Lokalizacja różnych procedur, zapewniających jakość usług w warstwie aplikacji, w warstwach niższych architektury systemu informatycznego (zwłaszcza definiujących funkcjonalność podsystemu komunikacyjnego sieci) wymaga szczególnej staranności. Tę samą lub wyższą wartość wskaźników jakości usług można uzyskać przez lokalizację określonej procedury w różnych warstwach architektury sieci teleinformatycznej.

Realizacja określonej funkcji w warstwach niższych może być efektywna, jeżeli minimalnie ingeruje się w działanie innych mechanizmów występujących w tych warstwach. Lokalizacja procedur zapewniania jakości usług w warstwach wyższych jest uzasadniona wtedy, gdy jakość usług – zapewniana określonymi procedurami –

- jest wymagana tylko przez niektóre spośród wszystkich aplikacji wspieranych funkcjonalnością warstw niższych,
- nie może być uzyskana efektywnie bez dostępu do informacji niedostępnych w warstwach niższych.

4.2. HIERARCHIA ZADAŃ KOMUNIKACJI W ARCHITEKTURACH WARSTWOWYCH

W różnych architekturach rozproszonych systemów komputerowych powszechnie jest stosowana koncepcja warstwowości i zagnieżdżenia funkcji, zgodnie z którą reali-

zacja zadań komunikacji wiąże się z dekompozycją pierwotnego oraz przekształceniem prostego zadania komunikacji w realizowalne zadania transmisyjne.

Celem dekompozycji pierwotnego zadania komunikacji (najczęściej dwukierunkowego) jest otrzymanie sekwencji prostych zadań komunikacji (najczęściej jednokierunkowych), których wykonanie polega na jedno- lub wielokrotnym odwołaniu się do systemu transmisji danych za pośrednictwem systemu komunikacji danych.

Przekształcenie prostego zadania komunikacji w realizowalne zadania transmisji danych związane jest z przypisaniem zadaniu komunikacji odpowiedniego zbioru funkcji, których celem jest realizacja zadania komunikacyjnego za pośrednictwem systemu transmisyjnego z zachowaniem właściwości zadania. Przekształcenie musi być odwracalne.

Ze względu na wymogi wynikające ze standaryzacji, otwartości i uniwersalności architektur systemów rozproszonych poszczególne funkcje należące do zbioru funkcji przypisywanych prostym zadaniom komunikacji tworzą hierarchię, która jest w naturalny sposób związana z koncepcją warstwowości. Hierarchia ta zarówno generuje, jak i wykorzystuje warstwowość. Generuje dlatego, że hierarchia funkcji pozwala na wyróżnienie rozróżnialnych zakresów funkcjonalnych i definiowanie interfejsów pomiędzy tymi zakresami. Wykorzystuje dlatego, że komplementarność funkcjonalna warstw pozwala na tworzenie ciągów funkcji komplementarnych o różnych właściwościach.

4.3. PODZIAŁ FUNKCJONALNY ZADANIA KOMUNIKACJI

Realizacja zadania komunikacji w rozproszonym systemie informacyjnym jest uwarunkowana – w ogólnym przypadku – przekazywaniem informacji pomiędzy komunikującymi się procesami, które odbywa się, zgodnie z wcześniej zdefiniowanymi ilościowymi i jakościowymi wymaganiami, w sposób i w formie determinowanej dostępnym systemem transmisji danych.

Wykonanie każdego zadania komunikacji na poziomie rozproszonego systemu informacyjnego (dalej zadania komunikacji) oznacza wykonanie zadania (zadań) komunikacji informacji oraz zadania (zadań) transferu informacji, co uzasadnia podział systemu komunikacji na dwie komplementarne części:

- podsystem komunikacji informacji, którego istotą jest znaczenie informacji (semantyka),

- podsystem transferu informacji, którego istotą jest transfer informacji (syntaktyka), realizowany w postaci transmisji danych w rzeczywistym systemie transmisyjnym; celem jest przeźroczysta organizacja transferu informacji na potrzeby systemu informacyjnego.

Wykonanie zadania transferu informacji jako transferu danych za pomocą systemu transmisji danych związane jest z zadaniem (zadaniami) przygotowania danych do

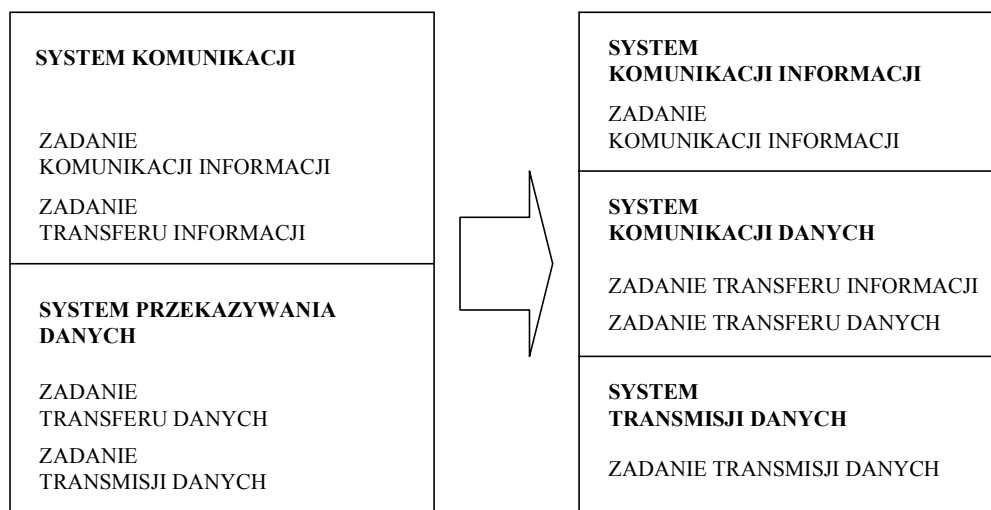
transmisji oraz zadaniem (zadaniami) transmisji, co z kolei uzasadnia podział systemu przekazywania danych na dwie komplementarne części:

- podsystem transferu danych, którego celem jest organizacja i sterowanie ich przepływem w określonym systemie transmisji danych,
- podsystem transmisji danych, którego celem jest ich transmisja w rzeczywistych kanałach transmisji danych.

Wyróżnienie czterech zadań oznacza jednoznaczne zdefiniowanie punktów końcowych procesów realizujących te zadania, hierarchicznej lokalizacji tych procesów oraz interfejsów pomiędzy nimi.

Taki podział systemów i zadań związanych z komunikacją i transmisją prowadzi do wydzielenia systemu komunikacji danych jako pośredniczącego pomiędzy systemem komunikacji informacji a systemem transmisji danych. W tak określonym systemie komunikacji danych realizowane są funkcje właściwe dla:

- podsystemu transferu informacji (organizacja transferu informacji), którego zadaniem jest przejęcie od systemu informacyjnego funkcji związanych z transferem informacji; większość funkcji implementowanych w tym podsystemie ma na celu eliminację konsekwencji technologii transferu i transmisji danych dla systemu komunikacji (informacyjnego) – transfer informacji jest przezroczysty dla zadania komunikacji informacji,
- podsystemu transferu danych (przygotowania i sterowania przepływem danych), którego zadaniem jest efektywne wykorzystanie systemu transmisji danych; implementacja większości funkcji tego podsystemu ma na celu eliminację konsekwencji ograniczonej i zawodności rzeczywistych systemów transmisji danych (rysunek 4.1).



Rysunek 4.1. Podział funkcjonalny systemu komunikacji informacji

Wydzielenie systemu komunikacji danych w ogólnym zadaniu komunikacji informacji i poszerzenie jego zakresu funkcjonalnego jest naturalną konsekwencją poszukiwania uniwersalnego, elastycznego systemu komunikacji danych:

➤ pośredniczącego pomiędzy różnorodnością rzeczywistych zadań komunikacji informacji a względnie niewielką liczbą różnych możliwych systemów transmisji danych (transformacja wymagań zadania komunikacji informacji w zadania transmisji danych i zadania sterowania transmisją danych),

➤ redukującego listę funkcji systemu informacyjnego niezwiązanych ze znaczeniem generowanych, przetwarzanych i odbieranych informacji (zadanie uniwersalnego interfejsu pomiędzy zadaniami komunikacji i transferu),

➤ efektywnie wykorzystującego różne technologie oraz techniki transferu i transmisji danych, zwłaszcza w warunkach współdzielenia systemu transmisji (zadanie optymalnego wykorzystania współdzielonych zasobów systemu transmisji danych).

Wykonanie zadania komunikacji, w tak zdefiniowanym systemie, zgodnie z ilościowymi i jakościowymi wymaganiami aplikacji, wiąże się z określeniem zbiorów funkcji, jakie będą realizowane w każdym z trzech wyróżnionych systemów: komunikacji informacji, komunikacji danych i transmisji danych.

Przy założeniu, że w systemie komunikacji danych są dostępne wszystkie funkcje niezbędne do wykonania transferu informacji oraz rozmaite rozwiązania systemu transmisji danych, możliwe są różne warianty realizacji zadania komunikacji oznaczające różne definiowanie interfejsów pomiędzy systemami:

<p>SYSTEM KOMUNIKACJI</p> <p>ZADANIA:</p> <ul style="list-style-type: none"> - KOMUNIKACJI INFORMACJI - TRANSFERU INFORMACJI - TRANSFERU DANYCH 	<p>SYSTEM KOMUNIKACJI INFORMACJI</p> <p>ZADANIE:</p> <ul style="list-style-type: none"> - KOMUNIKACJI INFORMACJI 	<p>SYSTEM KOMUNIKACJI INFORMACJI</p> <p>ZADANIA:</p> <ul style="list-style-type: none"> - KOMUNIKACJI INFORMACJI - TRANSFERU INFORMACJI
<p>SYSTEM TRANSMISJI DANYCH</p> <p>ZADANIE</p> <ul style="list-style-type: none"> - TRANSMISJI DANYCH 	<p>SYSTEM KOMUNIKACJI DANYCH</p> <p>ZADANIA:</p> <ul style="list-style-type: none"> - TRANSFERU INFORMACJI - TRANSFERU DANYCH 	<p>SYSTEM KOMUNIKACJI DANYCH</p> <p>ZADANIA:</p> <ul style="list-style-type: none"> - TRANSFERU INFORMACJI - TRANSFERU DANYCH
<p>SYSTEM TRANSMISJI DANYCH</p> <p>ZADANIE</p> <ul style="list-style-type: none"> - TRANSMISJI DANYCH 	<p>SYSTEM TRANSMISJI DANYCH</p> <p>ZADANIE</p> <ul style="list-style-type: none"> - TRANSMISJI DANYCH 	<p>SYSTEM TRANSMISJI DANYCH</p> <p>ZADANIE</p> <ul style="list-style-type: none"> - TRANSMISJI DANYCH

a) funkcje systemu komunikacji danych pełni system komunikacji informacji

b) funkcje systemu komunikacji danych oraz systemu komunikacji informacji są rozłączne

c) funkcje systemu komunikacji danych oraz system komunikacji informacji są redundantne

Rysunek 4.2. Warianty realizacji zadania komunikacji

➤ zadanie transferu informacji jest realizowane z pominięciem systemu komunikacji danych; wszystkie funkcje – niezbędne do realizacji zadania transferu informacji – są implementowane na poziomie systemu informacyjnego i przekazywane bezpośrednio do systemu transmisji danych (rysunek 4.2a),

➤ zadanie transferu informacji jest realizowane przez systemy: informacyjny, komunikacji danych i transmisji danych, tak że zakresy funkcjonalne systemów komunikacji informacji i transmisji danych są rozłączne (rysunek 4.2b),

➤ zadanie transferu informacji jest realizowane przez systemy: informacyjny, system komunikacji danych i system transmisji danych, tak że zakresy funkcjonalne systemów komunikacji informacji i komunikacji danych nie są rozłączne; w pewnym zakresie funkcjonalnym istnieje redundancja, a każdy z systemów wykonuje własną wersję określonych funkcji (rysunek 4.2c).

4.4. WARSTWY SYSTEMU KOMUNIKACJI DANYCH

Zbiór funkcji zlokalizowanych w systemie komunikacji danych, których wykonanie pozwala na transformację:

- wcześniej zdefiniowanego zadania transferu informacji w zadanie transferu danych,
- zadania transferu danych w zadanie transmisji danych,

jest determinowany zarówno przez rodzaj zadania komunikacyjnego, jak i dostępny system transmisji.

Wymagania stawiane systemom komunikacji danych, dotyczące ich otwartości, modularności, skalowalności oraz zgodności ze standardami, uzasadniają ich architekturę opartą na koncepcji warstwowości i zagnieżdżenia funkcji.

Koncepcja warstwowości w systemie komunikacji danych opiera się na następujących założeniach:

➤ proces transformacji zadania transferu informacji w zadanie transferu danych jest sekwencją dobrze zdefiniowanych i rozłącznych czynności (procesów), niezależną od zadania komunikacji informacji i zadania transmisji danych,

➤ realizacja każdej wyróżnionej czynności w procesie transformacji oznacza wykonanie funkcji zdeterminowanych zarówno zadaniem komunikacji informacji, jak i wykorzystywanym rozwiązaniem transmisyjnym,

➤ możliwe jest wydzielenie hierarchii zbiorów funkcji (usług) – warstw grupujących funkcje o tych samych zakresach funkcjonalnych umożliwiające realizację wyróżnionych czynności,

➤ realizacja transformacji zadania transferu informacji w zadanie transmisji danych może oznaczać potrzebę wykonania funkcji należących do wszystkich lub tylko wybranych warstw,

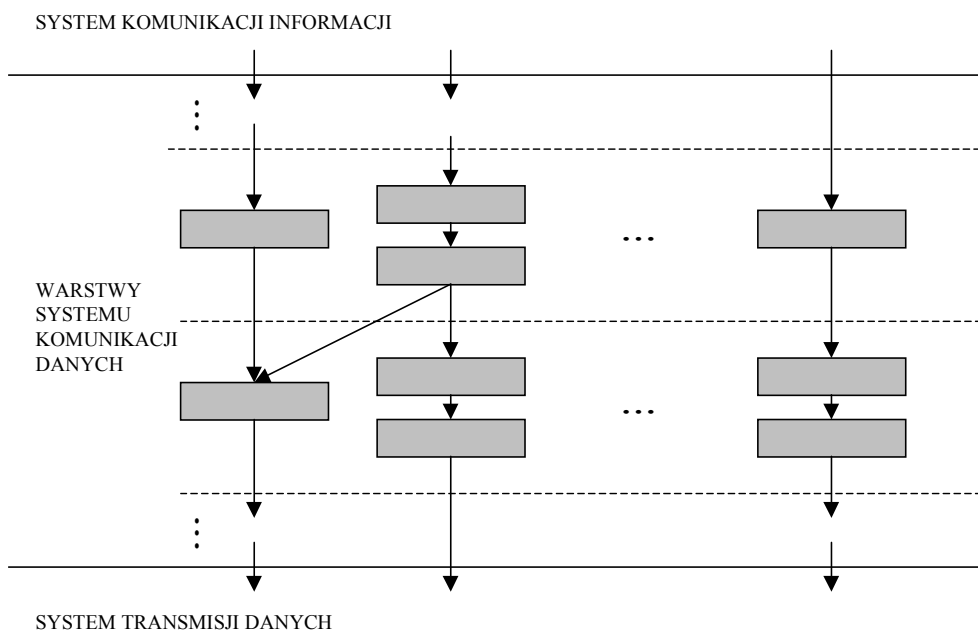
➤ funkcje dostępne w ramach poszczególnych warstw mogą być funkcjami złożonymi z funkcji o zakresach funkcjonalnych, będących podzbiorem zakresu funkcjonalności warstwy – odpowiada to możliwości tworzenia podwarstw danej warstwy,

➤ różnice pomiędzy różnymi funkcjami o tych samych zakresach funkcjonalnych i należących do danej warstwy wynikają z:

- dostosowania poszczególnych funkcji do obsługi różnych zadań transferu z wykorzystaniem różnych rozwiązań transmisyjnych,
- możliwości udostępniania funkcji na potrzeby różnej – ilościowo i jakościowo – obsługi tych samych zadań transferu z wykorzystaniem tych samych rozwiązań transmisyjnych.

Przyjęcie koncepcji warstwowości oznacza, że:

- istnieje hierarchia warstw zawierających różne funkcje o zdefiniowanym dla warstwy zakresie funkcjonalnym,
- istnieje procedura wywoływania przez warstwy i udostępniania przez nie funkcji w ramach transformacji zadania komunikacji w zadanie transmisji,
- istnieją interfejsy, przez które realizowana jest komunikacja pomiędzy sąsiadującymi warstwami – poprzez które są wywoływane funkcje i zwracane wyniki wywołania funkcji (rysunek 4.3).



Rysunek 4.3. Hierarchia redundantnych warstw architektury

4.5. REDUNDANCJA FUNKCJI W SYSTEMIE KOMUNIKACJI DANYCH

Spośród wymienionych możliwości podziału funkcji pomiędzy systemy komunikacji informacji i komunikacji danych praktyczne znaczenie ma wariant, w którym

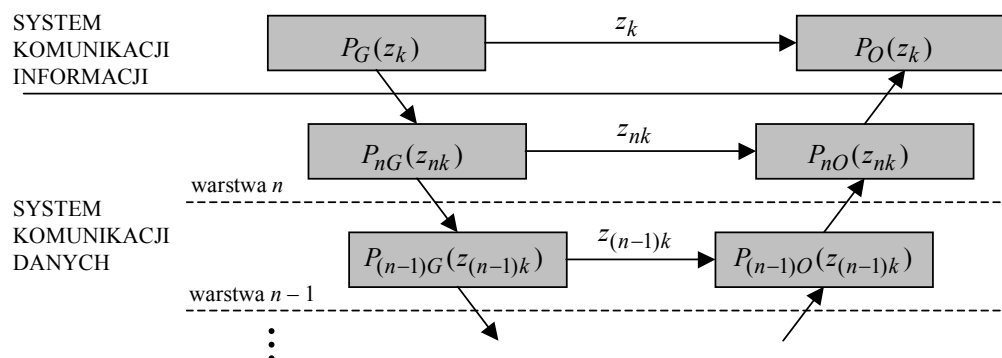
funkcje, niezbędne do realizacji zadania komunikacji zgodnie z wcześniej określonymi wymaganiami ilościowymi i jakościowymi, są implementowane zarówno w systemie komunikacji informacji, jak i w systemie komunikacji danych.

Implementacja tych samych funkcji zarówno w systemie komunikacji informacji, jak i w systemie komunikacji danych związana jest z realizacją dwóch ogólnych zasad projektowania rozwiązań komunikacyjnych stosujących architektury warstwowe:

- wszystkie funkcje decydujące o jakości komunikacji informacji muszą być implementowane w podsystemie komunikacji informacji,
- wszystkie funkcje decydujące o jakości komunikacji informacji mogą być równocześnie implementowane w różnych warstwach podsystemu komunikacji danych,
- o lokalizacji danej funkcji w danej warstwie decydują koszty implementacji danej funkcji przy założeniu tego samego efektu dla podsystemu komunikacji informacji wywołującego (bezpośrednio lub pośrednio) daną funkcję.

4.6. OBSŁUGA ZADAŃ KOMUNIKACJI W MODELU WARSTWOWYM

Zadanie (pierwotne) komunikacji z_k , generowane przez proces źródłowy $P_G(z_k)$ i adresowane do procesu ujściowego $P_O(z_k)$, zlokalizowane w systemie komunikacji informacji, jest wykonywane z wykorzystaniem funkcji dostępnych w systemie komunikacji danych i zlokalizowanych w warstwach n -tej i $(n-1)$ -szej; procesy $P_G(z_k)$ i $P_O(z_k)$ są procesami końcowymi zadania komunikacji z_k (rysunek 4.4). Zadanie komunikacji z_k oraz zadania transferu danych z_{nk} , $z_{(n-1)k}$, ..., formułowane w warstwach n -tej i $(n-1)$ -szej, ... tworzą hierarchię zadań.



Rysunek 4.4. Dekompozycja zadania komunikacji

W pierwszym etapie realizacji zadania komunikacji z_k wykonywane jest zadanie transferu danych z_{nk} zlokalizowane w warstwie n -tej. Zadanie z_{nk} jest wykonywane

między procesami: źródłowym $P_{nG}(z_{nk})$ i ujściowym $P_{nO}(z_{nk})$; procesy te są procesami końcowymi zadania z_{nk} .

Procesy $P_{nG}(z_{nk})$ i $P_{nO}(z_{nk})$, formułowane w najwyższej warstwie systemu komunikacji danych, są odpowiednikami procesów $P_G(z_k)$ i $P_O(z_k)$ w systemie komunikacji informacji.

Wykonanie zadania transferu danych z_{nk} wymaga wywołania funkcji dostępnych w warstwie niższej omawianej struktury hierarchicznej zadań – w warstwie $(n - 1)$ -szej. Jest to równoważne ze sformułowaniem nowego, kolejnego zadania transferu danych – zadania $z_{(n-1)k}$ – wykonywanego pomiędzy procesami: źródłowym $P_{(n-1)G}(z_{(n-1)k})$ oraz ujściowym $P_{(n-1)O}(z_{(n-1)k})$.

Procedura formułowania i zagnieżdżania kolejnych zadań transferu danych w kolejnych warstwach jest powtarzana aż do wyczerpania listy warstw dostępnych w danym systemie komunikacji danych i przeznaczonych do obsługi określonego zadania komunikacji informacji. Ostatnim zadaniem jest zwykle zadanie transmisji danych formułowane w warstwie najniższej modelu hierarchicznego, tzn. w warstwie fizycznej architektury sieci teleinformatycznej.

Miary jakości usług (np. niezawodność, średnie opóźnienie, średnia prędkość transferu itp.) dla zadania komunikacyjnego z_k oraz – odpowiadających mu – zadań transferu danych z_{nk} , $z_{(n-1)k}$, ... oznaczone, odpowiednio, przez $Q(z_k)$ oraz $Q(z_{nk})$, $Q(z_{(n-1)k})$, ... spełniają następujący warunek:

$$Q(z_k) \leq Q(z_{nk}) \leq Q(z_{(n-1)k}) \leq \dots$$

Zgodnie z powyższym wyrażeniem, odzwierciedlającym naturalną zależność pomiędzy wartościami wskaźników jakości usług w hierarchicznej strukturze zagnieżdżanych zadań, wartość wskaźnika jakości usług w nadrzędnym zadaniu komunikacji $Q(z_k)$ jest zawsze nie większa od wartości wskaźnika jakości usług możliwej do uzyskania w podrzędnym zadaniu transferu danych $Q(z_{nk})$.

Podobnie wartość wskaźnika jakości usług w zadaniu transferu danych formułowanym w n -tej warstwie $Q(z_{nk})$ jest mniejsza (lub równa) od wartości wskaźnika jakości usług $Q(z_{(n-1)k})$ dla podrzędnego zadania transferu danych formułowanego w $(n - 1)$ -szej warstwie. Zależności te wynikają z faktu, że jakość obsługi zadania nadrzędnego zależy całkowicie od jakości obsługi zadania podrzędnego (rysunek 4.5).

Na wartość wskaźnika jakości $Q(z_k)$ zadania komunikacji, nadrzędnego wobec zadań transferu danych, mają wpływ:

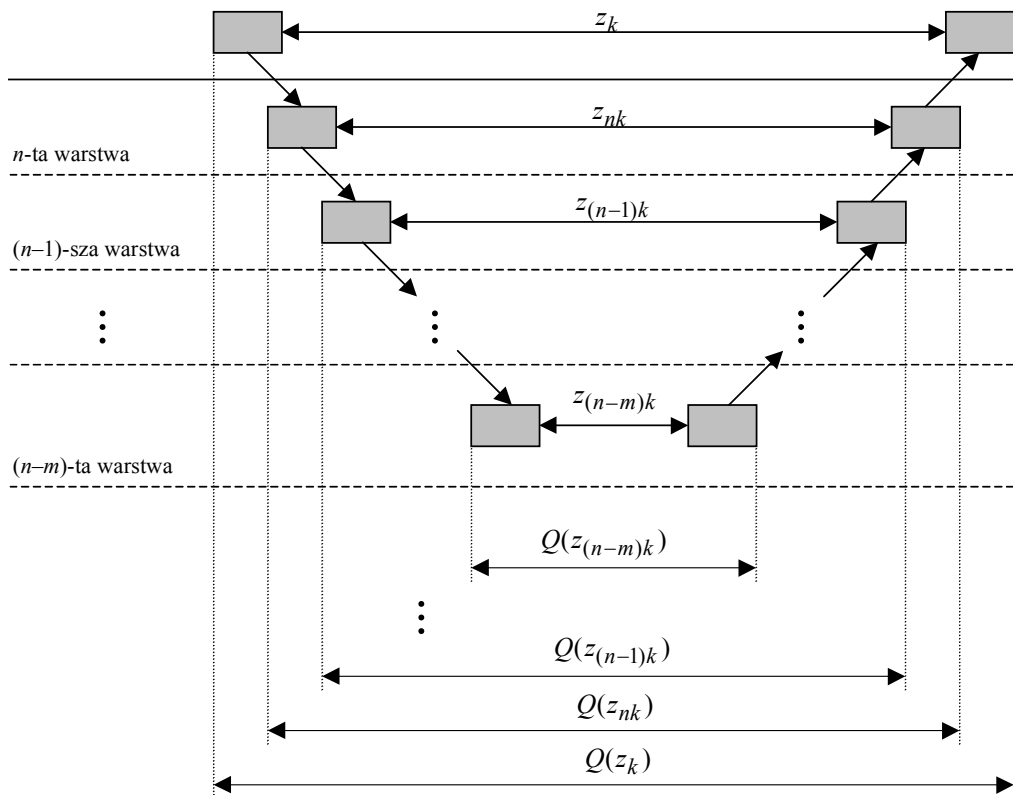
- jakość transformacji zadania komunikacyjnego z_k w zadanie transferu danych z_{nk} $T(z_k, z_{nk})$, realizowanej w węźle wejściowym pomiędzy procesami $P_G(z_k)$ oraz $P_{nG}(z_{nk})$ i oznaczanej przez $Q(T(z_k, z_{nk}))$; jakość pomiędzy procesami generowania zadań (procedura zapewniania jakości ze zbioru procedur *top-to-down*),

- jakość zadania transferu danych z_{nk} w n -tej warstwie $Q(z_{nk})$ realizowanego pomiędzy procesami: źródłowym $P_{nG}(z_{nk})$ i ujściowym $P_{nO}(z_{nk})$ (procedury zapewniania jakości komunikacji ze zbioru procedur *end-to-end*),

➤ jakość transformacji $T(z_{nk}, z_k)$ (transformacji odwrotnej do transformacji $T(z_k, z_{nk})$) zadania transferu danych z_{nk} w pierwotne zadanie komunikacji z_k , realizowanej w węźle ujściowym pomiędzy procesami $P_{nO}(z_{nk})$ oraz $P_O(z_k)$ i oznaczanej przez $Q(T(z_{nk}, z_k))$; jakość pomiędzy procesami końcowymi zadań (procedura zapewniania jakości ze zbioru procedur *down-to-top*):

$$Q(z_k) = f_n[Q(T(z_k, z_{nk})), Q(z_{nk}), Q(T(z_{nk}, z_k))],$$

gdzie $f_n(\dots)$ jest funkcją niemalejącą.



Rysunek 4.5. Hierarchia miar jakości usług komunikacji w modelu warstwowym

Analogiczna zależność dotyczy wartości wskaźników jakości dla zadania transferu danych z_{nk} , formułowanego w n -tej warstwie, który jest realizowany z wykorzystaniem zadania transferu danych $z_{(n-1)k}$ formułowanego w $(n-1)$ -szej warstwie:

$$Q(z_{nk}) = f_{n-1}[Q(T(z_{nk}, z_{(n-1)k})), Q(z_{(n-1)k}), Q(T(z_{(n-1)k}, z_{nk}))].$$

Podstawienie wyrażenia określającego $Q(z_{nk})$ do wyrażenia na $Q(z_k)$ prowadzi do równania:

$$\begin{aligned} Q(z_k) &= f_n [Q(T(z_k, z_{nk})), Q(z_{nk}), Q(T(z_{nk}, z_k))] = \\ &= f_n \{ Q(T(z_k, z_{nk})), f_{n-1} [Q(T(z_{nk}, z_{(n-1)k})), Q(z_{(n-1)k}), Q(T(z_{(n-1)k}, z_{nk}))], Q(T(z_{nk}, z_k)) \}, \end{aligned}$$

w którym:

$$Q(z_{(n-1)k}) = f_{n-2} [Q(T(z_{(n-1)k}, z_{(n-2)k})), Q(z_{(n-2)k}), Q(T(z_{(n-2)k}, z_{(n-1)k}))].$$

Jeżeli system komunikacji danych jest złożony z m warstw, to wartość wskaźnika jakości komunikacji $Q(z_k)$ dla zadania komunikacji z_k jest określona wyrażeniem:

$$\begin{aligned} Q(z_k) &= f_n \left\{ Q(T(z_k, z_{nk})), f_{n-1} [Q(T(z_{nk}, z_{(n-1)k})), \right. \\ &\quad \dots, f_{n-m} [Q(T(z_{(n-m+1)k}, z_{(n-m)k}), Q(z_{(n-m)k}), Q(T(z_{(n-m)k}, z_{(n-m+1)k}))], \\ &\quad \left. \dots, Q(T(z_{(n-1)k}, z_{nk}))], Q(T(z_{nk}, z_k)) \right\}. \end{aligned}$$

Praktyczne znaczenie powyższego wyrażenia jest takie, że szacowanie wartości wskaźnika jakości dla zadania komunikacji z_k opiera się na wartości wskaźników jakości dla zadań transferu danych w poszczególnych warstwach określonego systemu komunikacji danych $z_{nk}, z_{(n-1)k}, \dots, z_{(n-m)k}$. Co więcej, otrzymanie założonej wartości wskaźnika jakości dla jednego z wyróżnionych zadań transferu danych jest warunkiem koniecznym, ale nie jest warunkiem wystarczającym, otrzymania tej samej założonej wartości wskaźnika jakości dla pierwotnego zadania komunikacji z_k , tzn. do zapewnienia żądanej jakości usług komunikacyjnych.

4.7. PRZYKŁAD DEKOMPOZYCJI ZADANIA KOMUNIKACJI

Założmy, na potrzeby przykładu ilustrującego wpływ różnych mechanizmów, implementowanych w warstwie podrzędnej zadania komunikacji na jakość usług, że (rysunek 4.6):

➤ zadanie komunikacji informacji z_k , generowane przez proces $P_G(z_k)$ i adresowane do procesu $P_O(z_k)$, jest realizowane przez bezpośrednie odwołanie się do usług warstwy podrzędnej, tzn. j -tej warstwy systemu komunikacji danych,

➤ realizacja zadania komunikacji danych w warstwie j -tej oznacza – alternatywnie – realizację trzech wersji zadania komunikacji danych, które różnią się zastosowaną

procedurą korekcji błędów przez retransmisję ARQ (ang. *Automatic Retransmission Request*) (rys. 4.6):

➤ $z_{jk}(SW)$ pomiędzy procesami $P_{jG}(z_{jk}(SW))$ i $P_{jO}(z_{jk}(SW))$, gdzie SW oznacza procedurę korekcji przez pojedyncze retransmisje (ang. *stop-and-wait ARQ*),

➤ $z_{jk}(C)$ pomiędzy procesami $P_{jG}(z_{jk}(C))$ i $P_{jO}(z_{jk}(C))$, gdzie C oznacza procedurę korekcji przez grupowe retransmisje (ang. *continuous go-back-N ARQ*),

➤ $z_{jk}(S)$ pomiędzy procesami $P_{jG}(z_{jk}(S))$ i $P_{jO}(z_{jk}(S))$, gdzie S oznacza procedurę korekcji przez selektywne wybiórcze retransmisje (ang. *selective-repeat ARQ*),

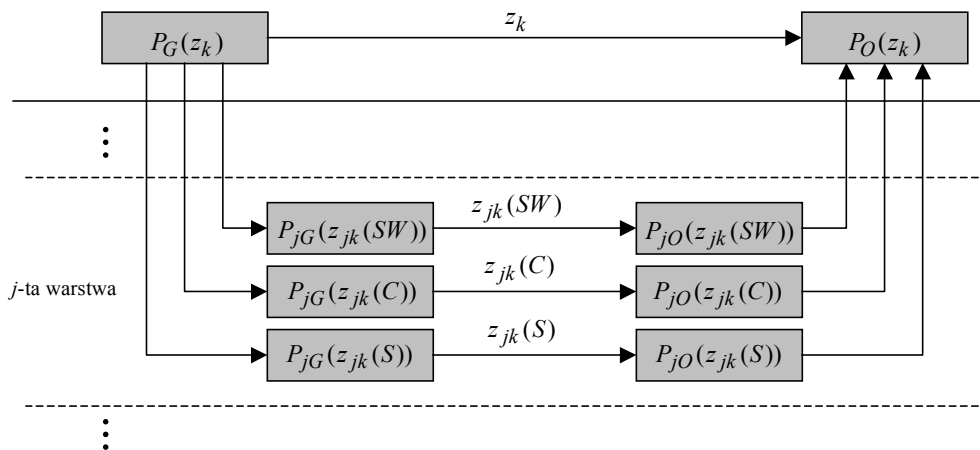
➤ miarą jakości usług wyróżnionych wersji zadania komunikacji danych z_{jk} realizowanego w j -tej warstwie, tzn. $z_{jk}(SW)$, $z_{jk}(C)$ oraz $z_{jk}(S)$, są średnie szybkości transferu pakietów: $\bar{v}_{jk}(SW)$, $\bar{v}_{jk}(C)$ oraz $\bar{v}_{jk}(S)$,

➤ miarą jakości usług w zadaniu komunikacji informacji z_k , dla różnych procedur korekcji, są średnie szybkości transferu wiadomości złożonych z N pakietów – tzn. $\bar{w}_k(SW)$, $\bar{w}_k(C)$ oraz $\bar{w}_k(S)$ – przez proces $P_O(z_k)$,

➤ średnie szybkości transferu wiadomości przez proces $P_O(z_k)$ dla różnych procedur retransmisji, tj. $\bar{w}_k(SW)$, $\bar{w}_k(C)$ oraz $\bar{w}_k(S)$, zależą od średniej szybkości transferu pakietów przez odpowiednie procesy $P_{jO}(z_{jk}(SW))$, $P_{jO}(z_{jk}(C))$ oraz $P_{jO}(z_{jk}(S))$ oraz konieczności lub jej braku odtwarzania kolejności pakietów w procesie $P_O(z_k)$,

➤ średnie szybkości transferu pakietów do j -tej warstwy, dla omawianych wersji procedur korekcji ARQ, $\bar{v}[P_G(z_k), P_{jG}(z_{jk}(SW))]$, $\bar{v}[P_G(z_k), P_{jG}(z_{jk}(C))]$ oraz $\bar{v}[P_G(z_k), P_{jG}(z_{jk}(S))]$, tzn. odpowiednio szybkości transferu pakietów pomiędzy procesami $P_G(z_k)$ oraz $P_{jG}(z_{jk}(SW))$, $P_{jG}(z_{jk}(C))$ i $P_{jG}(z_{jk}(S))$ są takie same.

PODSYSTEM KOMUNIKACJI INFORMACJI



PODSYSTEM KOMUNIKACJI DANYCH

Rysunek 4.6. Wersje zadania komunikacji danych dla różnych procedur korekcji

4.7.1. PROCEDURA RETRANSMISJI POJEDYNCZYCH

Nadajnik transmituje n -ty pakiet i oczekuje na powiadomienie – pozytywne lub negatywne. Jeżeli powiadomienie jest negatywne, to n -ty pakiet jest retransmitowany; w przeciwnym razie nadawany jest kolejny $(n + 1)$ -szy pakiet. Realizacja procedury retransmisji pakietu wnosi opóźnienie będące sumą czasów propagacji sygnału w kanałach transmisyjnych oraz czasów przetwarzania pakietu i potwierżeń.

Po przyjęciu następujących oznaczeń:

t_p – czas propagacji sygnału,

t_{1d} – czas przetwarzania pakietu (czas generowania powiadomienia),

t_{2d} – czas przetwarzania powiadomienia,

l – długość pakietu (w bitach),

l_{ACK} – długość potwierżenia (w bitach),

d – czas transmisji pakietu równy czasowi upływającemu od chwili rozpoczęcia transmisji pakietu do chwili rozpoczęcia transmisji następnego pakietu,

\bar{d} – średni czas transmisji pakietu,

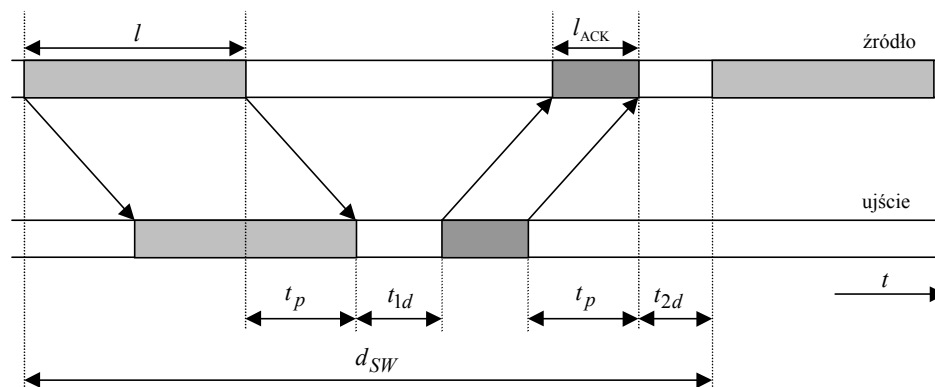
v – szybkość transmisji (w bitach/sekundę),

p_e – prawdopodobieństwo wystąpienia błędu elementarnego,

czas transmisji pakietu d_{SW} jest określony wyrażeniem (rysunek 4.7):

$$d_{SW} = \frac{l + l_{ACK}}{v} + 2t_p + t_{1d} + t_{2d} = \left(\frac{l}{v} \right) + \left(\frac{l_{ACK}}{v} + 2t_p + t_{1d} + t_{2d} \right),$$

gdzie pierwszy składnik jest równy czasowi transferu pakietu o długości l w łączu danych o szybkości transmisji v , a drugi jest sumą czasów „obsługi” tego transferu danych.



Rysunek 4.7. Czas transmisji pakietu dla procedury retransmisji pojedynczych

Po przyjęciu, dla uproszczenia, że czasy przetwarzania pakietu i powiadomień są pomijalnie małe oraz że czas transmisji powiadomienia jest, w porównaniu z czasem transmisji pakietu T ($T = l/v$), pomijalny, otrzymuje się:

$$d_{SW} \approx T + 2t_p.$$

Prawdopodobieństwo zdarzenia, że transmisja pakietu jest bezbłędna wynosi $p_c = (1 - p_e)^l$, gdzie p_e jest prawdopodobieństwem błędu elementarnego. Jeżeli i -ta ($i = 1, 2, \dots$) próba transmisji bloku danych zakończona jest powodzeniem, tzn. że wcześniejszych ($i - 1$) kolejnych prób kończy się powtórzeniami transmisji, średni czas transmisji pakietu:

$$\bar{d}_{SW} = \sum_{i=1}^{+\infty} i d_{SW} p_c (1 - p_c)^{i-1} = \frac{d_{SW}}{p_c},$$

a efektywna szybkość transferu pakietów \bar{v}_{SW} w zadaniu $z_{jk}(SW)$ j -tej warstwy:

$$\bar{v}_{SW} = \frac{1}{\bar{d}_{SW}} = \frac{p_c}{d_{SW}}.$$

Ze względu na to, że:

➤ kolejność transferu pakietów pomiędzy procesami $P_{jG}(z_{jk}(SW))$ i $P_{jO}(z_{jk}(SW))$ dla omawianej procedury retransmisji, realizowanej w zadaniu $z_{jk}(SW)$ podrzędnej j -tej warstwy, jest zgodna z kolejnością transferu pakietów przez proces $P_G(z_k)$,

➤ średnie szybkości transferu pakietów w zadaniu komunikacji danych j -tej warstwy $z_{jk}(SW)$ oraz pomiędzy procesami $P_{jO}(z_{jk}(SW))$ i $P_O(z_k)$ spełniają warunek:

$$\bar{v}_{jk}(SW) = \bar{v}[P_{jO}(z_{jk}(SW)), P_O(z_k)] = \bar{v}_{SW},$$

średnia szybkość transferu wiadomości, złożonej z N pakietów, w zadaniu komunikacji informacji z_k jest równa:

$$\bar{w}_k(SW) = \frac{\bar{v}_{SW}}{N} = \frac{p_c}{Nd_{SW}}.$$

4.7.2. PROCEDURA RETRANSMISJI GRUPOWEJ

Nadajnik transmituje pakiety bez przerywania transmisji w oczekiwaniu na powiadomienia dla poszczególnych pakietów. Proces nadawania pakietów jest przerywany w razie otrzymania negatywnego powiadomienia wskazującego pakiet, w którym wykryto błąd. W takim przypadku nadajnik retransmituje wskazany pakiet i wszystkie pakiety następujące po nim.

Analiza opóźnienia wnoszonego przez procedurę retransmisji ciągłej wymaga określenia wartości tzw. współczynnika wyprzedzania m (ang. *degree of anticipation*). Kolejne pakiety – $(n + 1)$, $(n + 2)$, ..., $(n + m)$ – są transmitowane zgodnie z ustalonym porządkiem. Wartość współczynnika wyprzedzania m jest równa maksymalnej liczbie pakietów, które mogą być wysłane do chwili otrzymania powiadomienia dla n -tego pakietu.

Jeżeli transmisja pakietów przebiega bezbłędnie, to czas transmisji pakietu w obecności ciągłej procedury powtarzania $d_C = l/v$, a czas oczekiwania na powiadomienie $d_{ACK} = (l_{ACK}/v) + 2t_p$.

Transmisja kolejnych pakietów – w tym przypadku – jest transmisją bez przerw, jeżeli spełniony jest warunek $d_{ACK} \leq (m - 1)d_C$.

Przyjmując, że wartość współczynnika wyprzedzania m została wybrana tak, że powiadomienie dla $(n + 1)$ -szego pakietu nadchodzi w czasie, gdy transmitowany jest $(n + m)$ -ty pakiet, wówczas – jeżeli odebrane powiadomienia jest:

- pozytywne – nie ma przerw w nadawaniu pakietów,
- negatywne – nadajnik rozpoczyna retransmisję m pakietów, rozpoczynając od pakietu $(n + 1)$ -szego.

Jeśli kolejna i -ta ($i = 1, 2, \dots$) próba transmisji pakietu jest zakończona powodzeniem, tzn. że kolejnych wcześniejszych $(i - 1)$ prób kończy się retransmisją danego pakietu oraz dodatkowo m pakietów (w najgorszym przypadku), to średni czas transmisji pakietu:

$$\bar{d}_C = \sum_{i=0}^{+\infty} (im + 1) d_C p_c (1 - p_c)^i = d_C \left(1 + m \frac{1 - p_c}{p_c} \right),$$

a efektywna szybkość transferu pakietów \bar{v}_C w zadaniu $z_{jk}(C)$ j -tej warstwy:

$$\bar{v}_C = \frac{1}{\bar{d}_C} = \left[d_C \left(1 + M \frac{1 - p_c}{p_c} \right) \right]^{-1}.$$

Ze względu na to, że:

➤ kolejność transferu pakietów pomiędzy procesami $P_{jG}(z_{jk}(C))$ i $P_{jO}(z_{jk}(C))$ dla omawianej procedury retransmisji, realizowanej w zadaniu $z_{jk}(C)$ podrzędnej j -tej warstwy, jest zgodna z kolejnością transferu pakietów przez proces $P_G(z_k)$,

➤ średnie szybkości transferu pakietów w zadaniu komunikacji danych j -tej warstwy $z_{jk}(C)$ oraz pomiędzy procesami $P_{jO}(z_{jk}(C))$ i $P_O(z_j)$ spełniają warunek:

$$\bar{v}_{jk}(C) = \bar{v} [P_{jO}(z_{jk}(C)), P_O(z_k)] = \bar{v}_C,$$

średnia szybkość transferu wiadomości, złożonej z N pakietów, w zadaniu komunikacji informacji z_k jest równa:

$$\bar{w}_k(C) = \frac{\bar{v}_C}{N} = \left[Nd_C \left(1 + m \frac{1 - p_c}{p_c} \right) \right]^{-1}.$$

4.7.3. PROCEDURA RETRANSMISJI SELEKTYWNEJ

Procedura nadawania w tym przypadku jest identyczna z retransmisją typu ciągłego z powtarzaniem grupowym, przy czym retransmitowany jest tylko pakiet wskazany w negatywnym powiadomieniu.

Jeżeli transmisja pakietów przebiega bezbłędnie, to czas transmisji pakietu o długości l w systemie z selektywną procedurą retransmisji $d_S = l/v$. W razie wykrycia błędu generowane jest powiadomienie negatywne, którego odbiór w nadajniku skutkuje powtórzeniem transmisji wskazanego pakietu.

Przyjmując, że kolejna i -ta ($i = 1, 2, \dots$) próba transmisji pakietu jest zakończona powodzeniem, tzn. że kolejnych wcześniejszych ($i - 1$) prób kończy się retransmisją danego pakietu, średni czas transmisji pakietu:

$$\bar{d}_S = \sum_{i=0}^{+\infty} (i+1) d_S p_c (1-p_c)^i = \frac{d_S}{p_c},$$

a efektywna szybkość transferu pakietów \bar{v}_S w zadaniu $z_{jk}(S)$ j -tej warstwy:

$$\bar{v}_S = \frac{1}{\bar{d}_S} = \frac{p_c}{d_S}.$$

Ze względu na to, że:

➤ kolejność transferu pakietów pomiędzy procesami $P_{jG}(z_{jk}(S))$ i $P_{jO}(z_{jk}(S))$ dla omawianej procedury retransmisji, realizowanej w zadaniu $z_{jk}(S)$ podrzędnej j -tej warstwy, jest różna od kolejności transferu pakietów przez proces $P_G(z_k)$,

➤ kolejność pakietów, zgodna z kolejnością transferu pakietów przez proces $P_G(z_k)$, jest odtwarzana w procesie $P_{jO}(z_{jk}(S))$,

średnie szybkości transferu pakietów w zadaniu $z_{jk}(S)$ podrzędnej j -tej warstwy oraz pomiędzy procesami $P_{jO}(z_{jk}(S))$ i $P_O(z_k)$ są różne, tzn.:

$$\bar{v}_{jk}(S) = \bar{v}_S \neq \bar{v} [P_{jO}(z_{jk}(S)), P_O(z_k)].$$

Konieczność odtworzenia kolejności pakietów w procesie $P_{jO}(z_{jk}(S))$ oznacza, że proces odtwarzania kolejności wnosi dodatkowe opóźnienie $\bar{d}(N)$. Opóźnienie odtworzenia $\bar{d}(N)$ jest równe średniemu czasowi oczekiwania pakietu na odtworzenie kolejności pakietów w wiadomości złożonej z N pakietów:

$$\bar{d}(N) = \bar{d}_S \sum_{j=1}^{N-1} j(N-j) (1-p_c)^j p_c = d_S \sum_{j=1}^{N-1} j(N-j) (1-p_c)^j.$$

Średnia szybkość transferu pakietów wiadomości N -pakietowej z procesu $P_{jO}(z_{jk}(S))$ do procesu $P_O(z_k)$, zachowującego kolejność generowania pakietów przez proces $P_G(z_k)$:

$$\bar{v} [P_{jO}(z_{jk}(S)), P_O(z_k)] = \frac{1}{\bar{d}(N)} = \left[d_S \sum_{j=1}^{N-1} j(N-j) (1-p_c)^j \right]^{-1}$$

jest mniejsza od szybkości transferu pakietów w zadaniu $z_{jk}(S)$.

W procedurze retransmisji selektywnej w j -tej warstwie średnia szybkość transferu wiadomości, złożonej z N pakietów, w zadaniu komunikacji informacji z_k jest równa:

$$\bar{w}_k(S) = \left[N d_S \sum_{j=1}^{N-1} j(N-j) (1-p_c)^j \right]^{-1}.$$

Średnia szybkość transferu wiadomości $\bar{w}_k(S)$ jest mniejsza od ilorazu średniej szybkości transferu pakietów w zadaniu $z_{jk}(S)$ i liczby pakietów, tzn. średnia szybkość transferu pakietów w kolejności zgodnej z kolejnością generowania jest mniejsza od średniej szybkości transferu pakietów dla procedury retransmisji selektywnej:

$$\bar{w}_k(S) \leq \frac{\bar{v}_S}{N}.$$

Podczas stosowania procedury retransmisji selektywnej w warstwie podrzędnej zadanie odtwarzania kolejności może być realizowane zarówno w warstwie podrzędnej, jak i nadrzędnej. W pierwszym przypadku jakość usług w warstwie nadrzędnej jest równa jakości usług w warstwie podrzędnej, w drugim natomiast jakość usług warstwy nadrzędnej jest niższa od jakości usług w warstwie podrzędnej.

4.8. PODSUMOWANIE

Realizacja każdego zadania komunikacji, formułowanego w warstwie aplikacji architektury warstwowej, związana jest z jego wielokrotną dekompozycją na zadania realizowane w warstwach niższych architektury. Celem dekompozycji jest transformacja zadania z warstwy aplikacji w ciąg zadań transmisji w określonych mediach transmisyjnych. Ze względu na wielość mechanizmów i procedur o podobnym zakresie funkcjonalnym w różnych warstwach architektury, wielokrotna dekompozycja zadania może być prowadzona na różne sposoby. W każdym przypadku jakość obsługi zadania komunikacji w warstwie nadrzędnej jest nie wyższa od jakości obsługi zadania komunikacji w warstwie podrzędnej.

Proces konwersji zadania komunikacji z warstwy aplikacji w kolejne zadania, zagnieżdżane w warstwach niższych, oznacza za każdym razem wprowadzanie dodatkowego nadmiaru, do którego obsługi potrzebne są dodatkowe zasoby, a skutkiem jest wzrost wolumenu ruchu transmitowanego w sieci.

Jakość każdej z możliwych dekompozycji oryginalnego zadania komunikacji jest określona miarami jakości stosowanymi w warstwie aplikacji, tzn. ostateczna ocena jakości usług dostarczanych w dowolnej warstwie architektury odbywa się w warstwie aplikacji.

LITERATURA

- [1] ANAGNOSTOU M, PROTONTARIOS E, *Performance analysis of the Selective Repeat ARQ protocol*, IEEE Transactions on Communications, Vol. COM-34, No. 2, February 1986, s. 127–135.
- [2] BERTSEKAS D., GALLAGER R., *Data networks*, Prentice-Hall International, Englewood Cliffs 1987.

- [3] BONATTI M., CASALI F., POPPLE G., *Integrated broadband communications: network and engineering aspects*, North-Holland, Amsterdam 1991.
- [4] BORZEMSKI L., GRZECH A., KASPRZAK A., *NETEX local area network architecture and application*, Systems Science, Vol. 12, No. 1/2, 1986, s. 93–108.
- [5] CZACHÓRSKI T., *Modele kolejkowe w ocenie efektywności sieci i systemów komputerowych*, Pracownia Komputerowa Jacka Skalmierskiego, Gliwice 1999.
- [6] ELBAUM R., SIDI M., *Topological design of local-area networks using genetic algorithms*, IEEE Transactions on Networking, Vol. 4, No. 5, October 1996, s. 766–778.
- [7] FRANK H., FRISCH I.T., *Communication, transmission and transportation networks*, Addison-Wesley, Reading 1971.
- [8] FROST V.S., MELAMED B., *Traffic modelling for telecommunications networks*, IEEE Communications Magazine, Vol. 32, No. 3, March 1994, s. 70–81.
- [9] GŁOWACKI M., *Measurement of QoS parameters of packet-switched Wide Area Networks*, Proceedings of 6th Polish Teletraffic Symposium, Szklarska Poręba 1999, s. 128–142.
- [10] GRZECH A., *Performance comparison of single and integrated local area networks*, Systems Science, Vol. 14, No. 4, 1988, s. 53–68.
- [11] GRZECH A., KASPRZAK A., *Switching control in integrated communication subsystems of local area networks*, Proceedings of 17th Biennial Symposium on Communications, Ontario 1994, s. 47–50.
- [12] GRZECH A., *Traffic flow at network node with protocols redundancy*, Proceedings 6th Polish Teletraffic Symposium, Wrocław 1999, s. 273–284.
- [13] GRZECH A., *Analysis and design of protocol stacks at node of active network*, Systems Science, Vol. 25, No. 2, 1999 s. 51–64.
- [14] GRZECH A., *Quality of service at layered architecture of data communication systems*, Systems, Vol. 5, No. 1/2, 2000, s. 113–130.
- [15] GRZECH A., *Procedures of designing enterprise network communication subsystems*, Cybernetics and Systems, Vol. 31, No. 5, 2000, s. 531–545.
- [16] GRZECH A., *Performance measures of intelligent computer network nodes*, Proceedings 16th World Computer Congress, Vol. 1, IEEE Press 2000, s. 454–462.
- [17] KLEINROCK L., *Communication nets*, Volume I: *Stochastic message flow and delay*, Volume II: *Computer Applications*, Wiley, New York 1976.
- [18] LUBACZ J., TOMASZEWSKI A., *Internet a sieci nowej generacji*, Przegląd Telekomunikacyjny, rocznik LXXIV, nr 5–6, 2001, s. 326–330.
- [19] LUBACZ J., *Modele odniesienia w rozwoju infrastruktury informacyjnej*, Przegląd Telekomunikacyjny, rocznik LXXIV, nr 1, 2001, s. 48–52.
- [20] MELSA J.L., SAGE A.P., *An introduction to probability and stochastic processes*, Prentice Hall, Englewood Cliffs 1973.
- [21] PAPIR Z., *Ruch telekomunikacyjny i przeciążenia sieci pakietowych*, WKŁ, Warszawa 2001.
- [22] POJOLLE G., SERET D., DROMARD D., HORLAIT E., *Integrated digital communications networks*, Wiley, Chichester 1988.
- [23] STALLINGS W., *High-speed networks: TCP-IP and ATM design principle*, Prentice-Hall, New York 1998.

5. WYZNACZANIE TRAS I STEROWANIE PRZEPIYWEM

Ograniczoność zasobów każdego systemu teleinformatycznego uzasadnia i wymusza konieczność stosowania mechanizmów, których zadaniem jest zarówno optymalne wykorzystanie zasobów systemu przez ruch wprowadzany do sieci, jak i zapobieganie przeciążeniom dostępnych zasobów przez ograniczanie ruchu przyjmowanego i obsługiwanego w systemie. Celem stosowania omawianych mechanizmów jest uzyskanie lub utrzymanie zadanej jakości usług dostarczanych użytkownikom systemu teleinformatycznego. Zadania te są realizowane albo przez rozdział zasobów, powodujący wzrost jakości usług, albo przez rozdział zasobów zapobiegający degradacji jakości usług.

Zadania wyznaczania tras i sterowania przepływem na ogół są rozłączne w tym znaczeniu, że wyznaczanie tras zwykle dotyczy wymiarowania sieci i udostępniania jej zasobów, natomiast zadanie sterowania przepływem jest związane z efektywnym wykorzystaniem zasobów udostępnianych w sieci.

Rozłączność omawianych zadań jest wyrazista w sieciach, w których jako kryteria wyboru tras stosowane są proste miary odległości węzłów sieci. W takich przypadkach zmiany optymalnych tras w sieci są powodowane zmianami struktury topologicznej sieci albo w wyniku awarii elementów sieci, albo zmian w niej ilości zasobów. Z tego punktu widzenia częstość wywoływania algorytmów wyznaczania tras i sterowania przepływami jest bardzo różna.

Zadania wyznaczania optymalnych tras są formułowane zarówno jako zadania projektowania sieci, jak i zadania inżynierii ruchu. Jakkolwiek sformułowania tych zadań są podobne, są to jednak zadania różne. Główna różnica polega na tym, że zadania wyznaczania tras w procesie projektowania dotyczą wymiarowania sieci, zadanie wyznaczania tras na potrzeby zarządzania ruchem w sieci dotyczy natomiast takiego wyboru tras, których zastosowanie do przekazywania ruchu w sieci prowadzi do zwiększenia stopnia wykorzystania zasobów.

W sieciach, w których są realizowane różne koncepcje dostarczania jakości usług, procedury wyboru tras i wyznaczania na nich przepływów są wywoływane jednakowo często. W takich przypadkach rozdzielanie omawianych zadań nie jest ani oczywiste, ani zasadne.

5.1. ZADANIA WYZNACZANIA TRAS I PRZEPLYWÓW

Mechanizmy odpowiedzialne w systemie teleinformatycznym za rozdział oraz wykorzystanie współdzielonych i ograniczonych zasobów systemu zalicza się do trzech grup mechanizmów: wyznaczania tras, przeciwdziałania przeciążeniom i sterowania przepływem. Mechanizmy należące do wymienionych trzech grup charakteryzuje różna funkcjonalność oraz komplementarność. Różna funkcjonalność wynika z tego, że uaktywnianie poszczególnych mechanizmów zależy zarówno od sposobu zarządzania jakością usług w systemie, jak i od stanu sieci oraz stopnia wykorzystania jej zasobów. Komplementarność omawianych mechanizmów oznacza, że warunkiem skutecznego zarządzania jakością usług w systemie teleinformatycznym jest współistnienie i współdziałanie mechanizmów z omawianych trzech grup.

Zadania przeciwdziałania przeciążeniom i sterowania przepływem są często łączone, to znaczy że rozwiązanie zadania przeciwdziałania przeciążeniom definiuje warunki realizacji zadania sterowania przepływem. Sterowanie przepływem, w szerszym rozumieniu, obejmuje zagadnienia definiowania warunków powstawania przeciążeń oraz ograniczania ilości ruchu przyjmowanego do obsługi w sieci lub wybranym jej fragmencie. Dlatego też często spotyka się rozwiązania, w których np. mechanizmy sterowania przepływem pośrednio przeciwdziałają powstawaniu przeciążeń lub mechanizmy przeciwdziałania przeciążeniom pośrednio spełniają funkcje mechanizmów sterowania przepływem. Podobnie pośrednim efektem stosowania mechanizmów wyznaczania tras jest zwiększenie odporności sieci na przeciążenia.

Sposoby formułowania zadań wyznaczania optymalnych tras oraz sterowania przepływem (wyznaczania przepływu), mimo różnej funkcjonalności, są zbliżone. Podobieństwo sformułowań wynika z faktu, że zadania te dotyczą alokacji zasobów dla przepływów przyjmowanych do obsługi lub obsługiwanych w systemie. Różnice sformułowań wynikają z tego, że:

- celem zadania wyznaczania tras jest optymalna alokacja zasobów dla znanego ruchu generowanego przez źródła,
- celem zadania sterowania przepływem oraz przeciwdziałania przeciążeniom jest definiowanie warunków, w których konieczne jest ograniczanie ilości ruchu, a w razie konieczności ograniczenia (dławienia) ilości wprowadzanego i obsługiwanego ruchu taki sposób alokacji ograniczonych zasobów, który gwarantuje zadany poziom jakości obsługi ruchu.

Zadania wyznaczania tras formułuje się i rozwiązuje na etapie projektowania sieci, tzn. na etapie zabezpieczania zasobów – w ogólnym przypadku są zadaniami sterowania prewencyjnego. Zadania przeciwdziałania przeciążeniom i sterowania przepływem są formułowane i rozwiązywane na potrzeby zarządzania dostępem do sieci i ruchem w sieci, tzn. na etapie alokacji zasobów. Przeciwdziałanie przeciążeniom jest działaniem o charakterze prewencyjnym, sterowanie przepływem natomiast jest działaniem o charakterze reaktywnym.

5.2. ZADANIE WYZNACZANIA TRAS

Ocena jakości działania sieci z komutacją pakietów, i tym samym jakości usług sieci, otrzymywanych w wyniku stosowania różnych algorytmów wyznaczania tras w sieci i różnych procedur sterowania przepływem, wymaga zdefiniowania przeciążenia.

Przeciążenia (ang. *congestion*) w sieci mogą być mierzone zależnościami wartości parametrów procesów napływu pakietów do sieci oraz wartości parametrów kolejek pakietów oczekujących na obsługę w każdym z węzłów sieci. Działanie dwóch różnych algorytmów wyznaczania tras w sieci można oceniać przez porównanie m.in. wartości średnich i wariancji opóźnień pakietów. Algorytm zapewniający mniejszą wartość średnią i mniejszą wariancję opóźnienia pakietów – dla tej samej macierzy generowanego ruchu – jest lepszy.

Dogodnym, aczkolwiek niedoskonałym, sposobem mierzenia przeciążenia łącza jest wartość średnia ruchu przenoszonego przez to łącze. Jeżeli przyjąć, że zmiany statystyk procesu napływu pakietów dla każdego łącza (i, j) są konsekwencją działania algorytmu wyznaczania tras, to jedną z miar przeciążeń jest szybkość (intensywność) napływu pakietów (przepływ pakietów) łącza $(i, j) - f_{ij}$, gdzie $(i, j) \in \Gamma$, a Γ jest zbiorem wszystkich łuków grafu $G = (N, \Gamma)$.

Jednym z podstawowych założeń, formułowanych dla różnych modeli przepływów w sieciach, jest założenie o niezmienności statystyk ruchu generowanego na wejściach sieci. Nie jest ono ograniczeniem, jeśli zmiany statystyk są powolne względem średniego czasu obsługi kolejek pakietów w węzłach sieci. Nie jest także ograniczeniem w sieciach, w których liczba jednocześnie obsługiwanych źródeł ruchu jest bardzo duża, a przepływy generowane przez źródła są małe w porównaniu z sumarycznym przepływem w sieci.

Większość klasycznych, jednokryterialnych zadań optymalizacji w sieciach formułuje się dla kryterium postaci

$$d = \sum_{(i,j) \in \tilde{\Gamma}} d_{ij}(f_{ij}) = \sum_{(i,j) \in \tilde{\Gamma}} \frac{f_{ij}}{C_{ij} - f_{ij}} + \alpha_{ij} f_{ij},$$

w którym:

d_{ij} – monotonicznie malejąca funkcja przepływu f_{ij} ,

C_{ij} – pojemność łącza (i, j) wyrażona w jednostkach przepływu,

$\alpha_{ij} f_{ij}$ – opóźnienie przetwarzania i propagacji wnoszone przez łącze (i, j) dla przepływu f_{ij} .

Zgodnie z podanym wyrażeniem wartość średniego opóźnienia d zależy od wszystkich przepływów w łączach f_{ij} i pojemności wszystkich łączy C_{ij} w sieci. Średnie opóźnienie można interpretować jako sumę średnich czasów oczekiwania wszystkich jednostek danych (pakietów) na obsługę w kolejkach, przy założeniu, że kolejki w węzłach są typu $M/M/1$, gdzie f_{ij} jest intensywnością napływu zgłoszeń, a C_{ij} – intensywnością obsługi zgłoszeń. Omawiane wyrażenie jest dogodną i praktyczną miarą

przeciążenia, ze względu na to, że definiuje przeciążenia łączy sieci jako stany, w których wartości przepływów f_{ij} rosną do wartości bliskich pojemnościom łączy C_{ij} : $d_{ij}(f_{ij}) \rightarrow \infty$, jeżeli $f_{ij} \rightarrow C_{ij}$.

Wartość d może być stosowana do wyznaczenia średniego opóźnienia jednostek danych w sieci \tilde{d} , zgodnie z następującą zależnością:

$$\tilde{d} = \frac{1}{\gamma} d = \frac{1}{\gamma} \sum_{(i,j) \in \tilde{A}} d_{ij}(f_{ij}) = \frac{1}{\gamma} \sum_{(i,j) \in \tilde{A}} \frac{f_{ij}}{C_{ij} - f_{ij}} + \alpha_{ij} f_{ij},$$

gdzie γ jest sumaryczną intensywnością napływu jednostek danych.

Wartość γ jest sumą wszystkich elementów macierzy ruchu (macierzy wymagań) (ang. *requirement matrix*), którymi są intensywności strumieni jednostek danych wprowadzanych do sieci we wszystkich węzłach wejściowych sieci, tj. intensywności przepływów v_{ij} pomiędzy wszystkimi parami węzłów źródło–ujście:

$$\gamma = \sum_{i \in \tilde{I}} \sum_{j \in \tilde{I}, i \neq j} v_{ij}.$$

Inną miarą jakości, o podobnych właściwościach wyrażania przeciążenia łączy, jest współczynnik maksymalnego wykorzystania łączy:

$$u_{ij} = \max_{(i,j)} \left\{ \frac{f_{ij}}{C_{ij}} \right\}.$$

Porównanie wyników stosowania wymienionych dwóch funkcji kryterialnych w zadaniach wyznaczania tras w sieciach pakietowych pokazuje, że otrzymane rozwiązania są zbliżone.

5.3. ZADANIE WYZNACZANIA OPTYMALNYCH TRAS

Zadanie wyznaczania optymalnych tras w sieci polega na takim podziale ruchu, generowanego przez źródła, pomiędzy wiele tras prowadzących ze źródeł do ujść, który minimalizuje funkcję kryterialną.

Po przyjęciu oznaczeń:

- $w = (s, t)$ – para źródło–ujście,
- v_w – stała szybkość napływu pakietów do sieci w s -tym węźle i wyprowadzanych z sieci w t -tym węźle, charakteryzująca parę $w = (i, j)$,
- W – zbiór wszystkich par źródło–ujście w sieci,
- P_w – zbiór wszystkich tras skierowanych dla pary $w = (s, t)$, tj. zbiór wszystkich tras łączących s -ty węzeł źródłowy z t -tym węzłem ujściowym,
- x_p – przepływ na trasie p ,

zbiór wszystkich przepływów na trasach pomiędzy wszystkimi parami węzłów sieci spełnia ograniczenia:

$$\sum_{p \in P_w} x_p = v_w \text{ dla każdego } w \in W \text{ oraz } x_p \geq 0 \text{ dla każdego } p \in P_w \text{ i } w \in W.$$

Sumaryczny przepływ w łączy (i, j) jest sumą przepływów po wszystkich trasach, które zawierają łączy (i, j) :

$$f_{ij} = \sum_{w \in W} \sum_{p \in P_w(i, j)} x_p,$$

gdzie $P_w(i, j)$ jest zbiorem wszystkich tras w zbiorze P_w zawierających łączy (i, j) .

Zadanie wyboru optymalnych tras, polegające na wyznaczeniu zbioru przepływów na trasach $\{x_p\}$, minimalizujących wartość wybranej funkcji kryterialnej, jest sformułowane następująco:

$$d^* = \min_{\{x_p\}} \sum_{(i, j) \in \tilde{A}} d_{ij} \left(\sum_{w \in W} \sum_{p \in P_w(i, j)} x_p \right),$$

przy ograniczeniach:

$$\sum_{p \in P_w} x_p = v_w \text{ dla każdego } w \in W,$$

$$x_p \geq 0 \text{ dla każdego } p \in P_w \text{ i } w \in W.$$

Zadanie wyznaczania optymalnych tras, w podanym sformułowaniu, ma postać umożliwiającą otrzymanie rozwiązań analitycznych w wyniku działania rozproszonych algorytmów. Źródłem podstawowego ograniczenia tak sformułowanego zadania optymalizacji jest funkcja kryterialna, której wybór jest oparty na założeniu, że optymalne trasy mogą być wyznaczone przez optymalizację średnich wartości przepływów, bez brania pod uwagę innych charakterystyk przepływów. Tak wybrana funkcja kryterialna nie jest wrażliwa na wariancje, korelacje czasów pomiędzy zgłoszeniami kolejnych pakietów i czasy ich transmisji.

Rozwiązanie zadania wyznaczania optymalnych tras w sieci, dla danego zbioru przepływów generowanych dla wszystkich par źródło–ujście, jest najczęściej wyrażane z uwzględnieniem pojęcia najkrótszej trasy. Rozwiązanie to jest charakteryzowane za pomocą właściwości funkcji d_{ij} oraz pochodnych funkcji d_{ij} , spełniających następujące warunki:

- d_{ij} jest podwójnie różniczkowalną funkcją sumarycznego przepływu w łączy f_{ij} ,
- d_{ij} jest funkcją zdefiniowaną w przedziale $[0, C_{ij})$, gdzie C_{ij} jest liczbą dodatnią (zwykle określającą pojemność łączy) lub liczbą równą nieskończoności,
- $d_{ij}(f_{ij}) \rightarrow \infty$ wtedy, gdy $f_{ij} \rightarrow C_{ij}$,
- pierwsza i druga pochodne funkcji d_{ij} , oznaczone odpowiednio przez d'_{ij} oraz d''_{ij} , są dodatnie dla wszystkich wartości f_{ij} z przedziału $[0, C_{ij})$.

Spełniająca podane warunki funkcja d_{ij} jest wypukłą i monotonicznie rosnącą funkcją f_{ij} w przedziale $[0, C_{ij})$. Ponieważ sumaryczny przepływ f_{ij} w łączy jest funk-

cją liniową przepływów na trasach x_p , funkcja kryterialna zadania optymalizacji ze względu na przepływy na trasach $\{x_p\}$ jest więc funkcją wypukłą, podwójnie różniczkowalną, a zbiór ograniczeń jest zbiorem wypukłym.

Charakterystyka optymalnego rozwiązania zadania wyznaczania tras w sieci jest podawana w postaci ogólnych warunków koniecznych i wystarczających optymalności, wynikających z następującego lematu:

Jeżeli f jest różniczkowalną i wypukłą funkcją n -wymiarowego wektora $\mathbf{x} = [x_1, x_2, \dots, x_n]$, których zbiór X jest zbiorem wypukłym, to wektor \mathbf{x}^* ($\mathbf{x}^* \in X$) jest optymalnym rozwiązaniem zadania minimalizacji funkcji $f(x)$ dla $\mathbf{x} \in X$ wtedy i tylko wtedy, gdy:

$$\sum_{i=1}^n \frac{\partial f(\mathbf{x}^*)}{\partial x_i} (x_i - x_i^*) \geq 0 \text{ dla każdego } \mathbf{x} \in X,$$

gdzie $\partial f(\mathbf{x}^*)/\partial x_i$ jest wartością pierwszej pochodnej funkcji f ze względu na współrzędną x_i , obliczaną dla wektora \mathbf{x}^* .

Po wprowadzeniu oznaczeń:

$$D(\mathbf{x}) = \sum_{(i,j) \in \bar{A}} d_{ij} \left(\sum_{w \in W} \sum_{p \in P_w(i,j)} x_p \right),$$

$$\frac{\partial D(\mathbf{x})}{\partial x_p} = \sum_{\{(k,l)\}_p} d'_{kl},$$

- | | |
|---------------------------------------|---|
| $D(\mathbf{x})$ | – funkcja kryterialna, |
| \mathbf{x} | – wektor, którego elementy są przepływami na wszystkich trasach należących do zbioru P_w dla wszystkich par źródło–ujście ze zbioru par W , |
| $\{(k, l)\}_p$ | – zbiór wszystkich łączy należących do trasy p , |
| $\partial D(\mathbf{x})/\partial x_p$ | – wartość pochodnej cząstkowej $D(\mathbf{x})$ ze względu na x_p obliczana dla przepływów odpowiadających wektorowi \mathbf{x} , |

wartość pochodnej cząstkowej $\partial D(\mathbf{x})/\partial x_p$ jest długością trasy p wtedy, gdy długość każdego łącza (k, l) , należącego do zbioru $\{(k, l)\}_p$, jest wartością pierwszej pochodnej d'_{kl} obliczonej dla wektora przepływów \mathbf{x} .

Zgodnie z lematem, wektor przepływów $\mathbf{x}^* = \{x_p^*\}$ jest optymalny, gdy spełnia ograniczenia zadania wyznaczania optymalnych tras oraz warunek konieczny i wystarczający optymalności (wynikający z lematu):

$$\sum_{w \in W} \sum_{p \in P_w} \frac{\partial D(\mathbf{x}^*)}{\partial x_p} (x_p - x_p^*) \geq 0,$$

dla wszystkich x_p , spełniających następujące ograniczenia:

$$\sum_{p \in P_w} x_p = v_w \text{ oraz } x_p \geq 0 \text{ dla wszystkich } p \in P_w \text{ i } w \in W.$$

Po przyjęciu, dla ustalonej pary źródło–ujście w ($w \in W$), że równość $x_p = x_p^*$ zachodzi dla wszystkich pozostałych par źródło–ujście u ($u \in W - \{w\}$), tzn.:

$$\sum_{w \in W} \sum_{p \in P_w} \frac{\partial D(\mathbf{x}^*)}{\partial x_p} (x_p - x_p^*) = \sum_{p \in P_w} \frac{\partial D(\mathbf{x}^*)}{\partial x_p} (x_p - x_p^*) \geq 0,$$

warunki optymalności wektora $\mathbf{x}^* = \{x_p^*\}$ dla każdej pary źródło–ujście w ($w \in W$) mogą być przedstawione następująco:

$$\sum_{p \in P_w} \frac{\partial D(\mathbf{x}^*)}{\partial x_p} (x_p - x_p^*) \geq 0$$

dla wszystkich $x_p \geq 0$ takich, że:

$$\sum_{p \in P_w} x_p = v_w.$$

Warunek ten jest równoważny z warunkiem spełnianym dla każdej pary źródło–ujście w ($w \in W$):

$$x_p^* > 0 \text{ tylko wtedy, gdy } \frac{\partial D(\mathbf{x}^*)}{\partial x_{p'}} \geq \frac{\partial D(\mathbf{x}^*)}{\partial x_p} \text{ dla wszystkich } p' \in P_w.$$

Warunek ten można interpretować w następujący sposób: zbiór przepływów na trasach jest optymalny wtedy i tylko wtedy, gdy przepływy są dodatnie tylko na trasach o minimalnej wartości pierwszych pochodnych.

5.4. MOC SIECI

Powszechnie stosowanym kryterium jakości do znajdowania kompromisu pomiędzy opóźnieniem transferu jednostek danych wnoszonym przez sieć a przepustowością sieci jest moc sieci (ang. *network power*). Do definiowania mocy sieci stosuje się średnie wartości szybkości przepływów i średnie opóźnienia jednostek danych wnoszone przez sieć.

Jeżeli sumaryczny przepływ w łączy (i, j) sieci, danej grafem $G = (N, I)$, jest sumą przepływów po wszystkich trasach, które zawierają łączy (i, j) :

$$f_{ij} = \sum_{w \in W} \sum_{p \in P_{w(i,j)}} x_p,$$

gdzie $P_{w(i,j)}$ jest zbiorem wszystkich tras w zbiorze P_w zawierających łącze (i, j) , to średnie opóźnienie jednostek danych sumarycznego przepływu f_{ij} , wnoszone przez łącze (i, j) , jest określone wyrażeniem:

$$\tilde{d}_{ij} = \frac{1}{C_{ij} - f_{ij}}.$$

Średnie opóźnienie jednostek danych \tilde{d}_w dla każdej pary źródło–ujście $w = (s, t)$ ($w \in W$), generowanych ze średnią szybkością v_w , jest sumą średnich opóźnień wnoszonych przez poszczególne łącza na trasach łączących źródło i ujście danej pary, tj. trasach należących do zbioru P_w . W omawianym przypadku średnia szybkość generowania jednostek danych jest tożsama z przepustowością sieci dla każdej z par węzłów.

Sieć, dana grafem $G = (N, \Gamma)$ i macierzą wymagań, tj. macierzą szybkości napływu jednostek danych v_w ($w \in W$) dla każdej pary węzłów $w = (i, j)$, może być charakteryzowana m.in. $2W$ -elementowym wektorem x ($x \in R^{2W}$) średnich szybkości generowania jednostek danych v_w i średniego opóźnienia jednostek danych \tilde{d}_w wnoszonych przez sieć:

$$x = [x_1, x_2, \dots, x_{2W}] = [v, \tilde{d}] = [v_1, v_2, \dots, v_w, \dots, v_W, \tilde{d}_1, \tilde{d}_2, \dots, \tilde{d}_w, \dots, \tilde{d}_W],$$

gdzie W jest liczbą wszystkich par źródło–ujście.

Z każdymi dwoma różnymi zbiorami szybkości generowania jednostek danych przez źródła przepływów i odpowiadających im średnich opóźnień jednostek danych wnoszonych przez sieć, tzn. x ($x = [v_1, v_2, \dots, v_w, \dots, v_W, \tilde{d}_1, \tilde{d}_2, \dots, \tilde{d}_w, \dots, \tilde{d}_W]$) i x^* ($x^* = [v_1^*, v_2^*, \dots, v_w^*, \dots, v_W^*, \tilde{d}_1^*, \tilde{d}_2^*, \dots, \tilde{d}_w^*, \dots, \tilde{d}_W^*]$) ($x, x^* \in R^{2W}$) można związać miarę jakości G , która porządkuje $2W$ -elementowe wektory. Wartość wyrażenia $x G x^*$ wskazuje na ile wektor szybkości przepływu $v = [v_1^*, v_2^*, \dots, v_w^*, \dots, v_W^*]$ powodujących opóźnienia $\tilde{d} = [\tilde{d}_1^*, \tilde{d}_2^*, \dots, \tilde{d}_w^*, \dots, \tilde{d}_W^*]$ jest, w sensie przyjętego kryterium jakości, korzystniejszy od wektora szybkości przepływu $v = [v_1, v_2, \dots, v_w, \dots, v_W]$ powodujących opóźnienia $\tilde{d} = [\tilde{d}_1, \tilde{d}_2, \dots, \tilde{d}_w, \dots, \tilde{d}_W]$.

Moc sieci jest definiowana jako iloraz przepustowości i opóźnienia wnoszonego przez sieć. Dany zbiór średnich szybkości generowania jednostek danych jest lepszy, gdy jego konsekwencją jest większa wartość mocy sieci.

Moc sieci P , definiowanej W -elementowym wektorem średnich szybkości przepływu $v = [v_1, v_2, \dots, v_w, \dots, v_W]$ i w -elementowym wektorem średnich opóźnień $\tilde{d} = [\tilde{d}_1, \tilde{d}_2, \dots, \tilde{d}_w, \dots, \tilde{d}_W]$, dla której:

➤ całkowita przepustowość sieci jest sumą średnich szybkości przepływu:

$$\gamma = \sum_{w=1}^W v_w,$$

➤ całkowite średnie opóźnienie jednostek, wyrażonych przez ważoną sumę opóźnień poszczególnych przepływów:

$$D = \frac{1}{\sum_{w=1}^W v_w} \sum_{w=1}^W v_w \tilde{d}_w = \frac{1}{\gamma} \sum_{w=1}^W v_w \tilde{d}_w,$$

jest wartością wyrażenia $P = \gamma/D$.

5.5. ZADANIE STEROWANIA PRZEPLÝWEM

Rozwiązywanie zadania sterowania przepływem na ogół jest konieczne wtedy, gdy zachodzi potrzeba ograniczania szybkości generowania strumienia zgłoszeń przez źródło, powodowana ograniczonymi zasobami komunikacyjnymi pomiędzy węzłami sieci lub zasobami przetwarzania w węzłach sieci.

Procedury sterowania przepływem mogą być implementowane na różnych poziomach agregacji przepływów i różnych poziomach ogólności. Każda z procedur sterowania przepływem zawiera dwa moduły funkcjonalne:

- identyfikacji aktualnego stanu tras (stopnia wykorzystania zasobów) w sieci,
- aktualizacji wartości parametrów przepływów na trasach sieci jako funkcji aktualnego stopnia wykorzystania zasobów sieci.

Procedury sterowania przepływem mogą być implementowane w wersji scentralizowanej (ang. *centralized flow control*) lub zdecentralizowanej (ang. *decentralized flow control*). Implementacja procedury zdecentralizowanej oznacza, że każdy z użytkowników sieci wyznacza wartości parametrów przepływu niezależnie od innych.

Cele stawiane procedurom sterowania przepływem dotyczą:

- znalezienia rozwiązania będącego kompromisem pomiędzy poziomem dławienia (blokowania) szybkości generowania zgłoszeń przez użytkowników sieci oraz utrzymywania wartości średnich opóźnień zgłoszeń na pewnym, możliwym do zaakceptowania przez użytkowników sieci, poziomie,
- sprawiedliwego podziału zasobów, polegającego na umożliwianiu dostępu do zasobów sieci każdemu użytkownikowi, niezależnie od różnicy całkowitego ruchu w sieci generowanego przez źródła i ruchu nieprzyjmowanego do obsługi w sieci ze względu na jej ograniczone zasoby, tzn. równomiernego rozkładu ruchu przyjmowanego do obsługi w sieci pomiędzy wszystkich jej użytkowników,
- przeciwdziałania degradacji przepustowości i powstawaniu wąskich gardeł powodowanych przepełnieniem pamięci buforowych.

Podstawowym kryterium jakości usług w sieci komputerowej jest opóźnienie jednostki danych właściwej dla poziomu (warstwy) architektury sieci, w której implementowany jest mechanizm sterowania przepływem. Małe wartości średniego opóźnienia są oczekiwane i pożądane przez praktycznie każdą aplikację obsługiwaną w sieci. Uzupełnianie zbioru kryteriów jakości usług o inne, oprócz opóźnienia, kryteria jest najczęściej powodowane tym, że pozwalają one wyróżniać klasy ruchu o różnej wrażliwości na opóźnienia i jego zakresy zmienności.

Implementacja dowolnej procedury sterowania przepływem praktycznie w żadnym przypadku nie powoduje zmniejszenia opóźnienia jednostek danych użytkowników sieci. Procedura sterowania przepływem przesuwając opóźnienie z poziomu (warstwy), na którym jest implementowana, na poziom wyższy, nadrzędny dla poziomu implementacji procedury. Na przykład ograniczanie napływu pakietów do sieci przez mechanizm sterowania dostępem wymusza wydłużenie czasu oczekiwania pakietów w kolejkach organizowanych na wejściach do sieci (w węzłach brzegowych), czyli oczekiwania poza siecią. Jeśli założymy, że czas obsługi segmentów w warstwie transportowej jest stały, to zastosowanie sterowania przepływem w warstwie sieciowej powoduje ograniczenie liczby pakietów jednocześnie obsługiwanych w warstwie sieciowej. Pakiety oczekują na dostęp do warstwy sieciowej w warstwie transportowej. Zmniejszenie średniego opóźnienia pakietów w warstwie sieciowej odbywa się kosztem wydłużenia czasu oczekiwania w warstwie transportowej (rysunek 5.1).

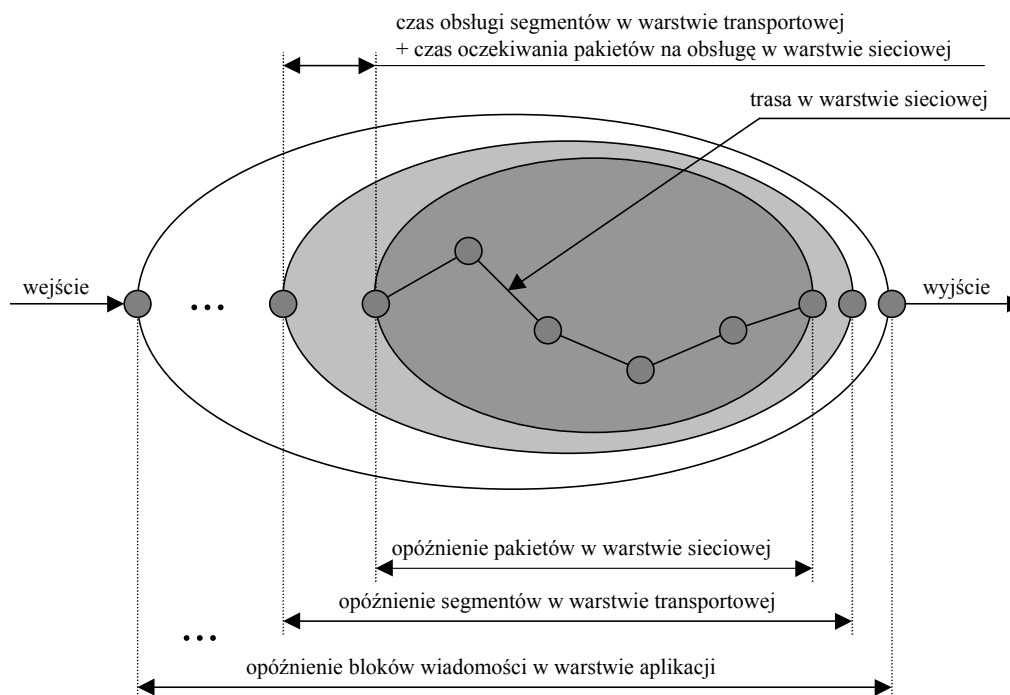
Głównym zadaniem sterowania przepływem jest przeciwdziałanie natłokom jednostek danych w sieci lub jej wybranych fragmentach, zagrażających jakości usług świadczonych wszystkim użytkownikom. Sterowanie przepływem jest sposobem reglamentacji zasobów sieci, zwłaszcza wtedy, gdy chwilowe zapotrzebowanie na zasoby jest większe od ilości dostępnych zasobów. Jedynym sposobem poprawy jakości usług dla wszystkich użytkowników sieci jest zwiększanie jej zasobów oraz poprawa algorytmów ich udostępniania. Innym sposobem utrzymywania założonego poziomu jakości usług jest ograniczanie liczby użytkowników sieci.

Zasadniczym powodem, dla którego celem mechanizmów sterowania przepływem jest utrzymywanie wartości opóźnień na niskim poziomie raczej w sieci niż poza nią, jest ochrona zasobów przed ich marnotrawieniem, powodowanym retransmisjami pakietów. Przyczyny retransmisji pakietów są dwojakie:

- zwiększony napływ pakietów do sieci oznacza wzrost zajętości pamięci buforowych i wartości prawdopodobieństwa przepełnienia, powodujących wzrost liczby odrzucanych pakietów wymagających następnie retransmisji,
- zwiększenie opóźnień pakietów oznacza wzrost opóźnień potwierdzeń pakietów i tym samym wartości prawdopodobieństwa przekroczenia limitu czasu oczekiwania na potwierdzenie pakietu; konsekwencją jest zwiększenie liczby retransmisji pakietów.

Każda retransmisja pakietu, powodowana utratą pakietu lub jego potwierdzeniem, oznacza, że zasoby przeznaczone na nieudaną transmisję pakietu są tracone; maleje

efektywna przepustowość sieci i powstają warunki rozprzestrzeniania się stanów przeciążenia w sieci.



Rysunek 5.1. Opóźnienia wnoszone przez warstwy architektury warstwowej sieci

Utrzymywanie małych opóźnień pakietów w sieci wtedy, gdy ilość sumarycznego ruchu rośnie, jest możliwe tylko przez obniżenie efektywnej przepustowości sieci. Stąd też wynika naturalny wybór: albo nieskrępowany dostęp do sieci ruchu generowanego przez użytkowników sieci, albo ograniczenie dostępu do sieci i utrzymywanie małych wartości opóźnień, które powodują, że retransmisje nie degradują jakości działania sieci.

Ogólne zadanie sterowania przepływem i warunki jego stosowania w sieci można scharakteryzować następująco: jeżeli opóźnienie pakietów jest mniejsze od pewnej wartości krytycznej, to sterowanie przepływem jest zbędne; w przeciwnym razie, tj. w przypadku dużego obciążenia sieci, celem sterowania przepływem jest takie ograniczenie ilości ruchu wprowadzanego do sieci, którego efektem jest opóźnienie pakietów nie większe od ustalonej wartości krytycznej. Tak sformułowane zadanie sterowania przepływem nie jest zadaniem trywialnym ze względu na to, że zależności opóźnienia pakietów i przepustowości sieci od procedur sterowania przepływem nie są jednoznaczne.

5.6. ŁĄCZNE ZADANIE WYZNACZANIA TRAS I STEROWANIA PRZEPIYWEM

Wyznaczanie optymalnych tras i sterowanie przepływem mogą być łączone w zadanie jednoczesnego wyznaczania tras i szybkości przepływów dla par źródło–ujście. Specjalnym przypadkiem jest zadanie, w którym przyjmuje się założenie o istnieniu stałych tras; wtedy sprowadza się to do zadania sterowania przepływem.

Łączenie zadań wyznaczania tras i sterowania przepływem jest charakterystyczne dla sieci, w których obsługa ruchu odbywa się bez gwarancji jakości (ang. *best-effort*). Łączenie zadań jest możliwe dlatego, że sposoby obsługi wszystkich składowych zagregowanego strumienia ruchu wprowadzonego do sieci są jednako- we. W sieciach, w których różnym klasom ruchu dostarczane są usługi o różnej jakości, zasadne jest rozdzielanie zadań wyznaczania tras i efektywnego wykorzystania ich do obsługi ruchu. Rozdzielenie omawianych zadań pozwala na budowę hierarchicznych rozwiązań, w których wyniki rozwiązania podrzędnego zadania wyznaczania tras są stosowane do rozwiązywania nadrzędnego zadania sterowania przepływami, współzawodniczącymi o pojemności tras w sieci.

Do sformułowania zadania jednoczesnego wyznaczania tras i sterowania przepływem w sieci można użyć modelu przepływów w sieci, zbliżonego do modelu stosowanego w zadaniu wyznaczania optymalnych tras w sieci.

W modelu przepływów w sieci v_w jest szybkością przepływu pakietów pomiędzy parą węzłów $w = (s, t)$, gdzie s -ty węzeł jest węzłem źródłowym, a t -ty – ujściowym. Interpretacja optymalnej wartości szybkości przepływu v_w może być różna i zależy od aplikacji generującej przepływ. Na przykład podczas transmisji:

- pojedynczych jednostek danych optymalna wartość v_w jest średnią szybkością przepływu, mierzoną we względnie długim przedziale czasu; celem sterowania jest transmitowanie nie więcej (nie mniej) niż $v_w T$ jednostek danych w przedziale czasu T ,

- w łączy wirtualnym optymalna wartość v_w jest stałą szybkością przepływu; celem sterowania jest utrzymanie stałej szybkości przez blokowanie lub akceptację nowych żądań przepływów,

- ruchu generowanego przez aplikacje czasu rzeczywistego optymalna wartość v_w jest chwilową szybkością przepływu; celem sterowania jest transmitowanie jednostek danych zgodnie z szybkością ich generowania przez źródło.

Jednoczesne wyznaczanie tras i szybkości przepływów dla par źródło–ujście wymaga modyfikacji zadania wyznaczania tras, jednoczesna bowiem minimalizacja funkcji kryterialnej w zadaniu wyznaczania optymalnych tras, ze względu na szybkości przepływów na trasach $\{\bar{x}_p\}$ oraz szybkości napływu do sieci $\{v_w\}$, tzn.:

$$\min_{\{\bar{x}_p\}, \{v_w\}} \sum_{(i,j) \in \tilde{A}} d_{ij} \left(\sum_{w \in W} \sum_{p \in P_w(i,j)} \bar{x}_p \right),$$

przy ograniczeniach:

$$\sum_{p \in P_w} \bar{x}_p = v_w \text{ dla każdego } w \in W,$$

$$\bar{x}_p \geq 0 \text{ dla każdego } p \in P_w \text{ i } w \in W$$

prowadzi do rozwiązania: $\bar{x}_p = 0$ oraz $v_w = 0$ dla każdego $p \in P_w$ i $w \in W$.

Modyfikacja zadania wyznaczania tras polega na modyfikacji funkcji kryterialnej przez wprowadzenie do niej wyrażenia określającego karę za ograniczanie (zmniejszanie) przez procedurę sterowania szybkości przepływu v_w oferowanej przez źródło, tzn. funkcji kary $h_w(\bar{v}_w)$, gdzie \bar{v}_w jest rzeczywistą (zaakceptowaną) szybkością przepływu. Wprowadzenie funkcji kary prowadzi do następującego sformułowania zadania jednoczesnego wyznaczania tras i przepływów:

$$\bar{d}^* = \min_{\{\bar{x}_p\}, \{\bar{v}_w\}} \left[\sum_{(i,j) \in \bar{A}} d_{ij} \left(\sum_{w \in W} \sum_{p \in P_w(i,j)} \bar{x}_p \right) + \sum_{w \in W} h_w(\bar{v}_w) \right],$$

przy ograniczeniach:

$$\sum_{p \in P_w} \bar{x}_p = \bar{v}_w \text{ dla każdego } w \in W,$$

$$\bar{x}_p \geq 0 \text{ dla każdego } p \in P_w \text{ i } w \in W,$$

$$0 \leq \bar{v}_w \leq v_w \text{ dla każdego } w \in W$$

gdzie $h_w(\bar{v}_w)$ jest funkcją kary za zwiększenie różnicy oferowanej i akceptowanej szybkości przepływu (za odrzucanie jednostek danych przepływu).

Funkcja kary, stosowana w omawianych zadaniach, jest funkcją wypukłą, monotonicznie malejącą w przedziale $(0, \infty)$ oraz spełniającą warunki:

$$h_w(\bar{v}_w) \rightarrow \infty \text{ gdy } \bar{v}_w \rightarrow 0,$$

$$h_w(\bar{v}_w) \rightarrow 0 \text{ gdy } \bar{v}_w \rightarrow \infty,$$

pierwsza – h'_w i druga – h''_w pochodne istnieją w przedziale $(0, \infty)$, a ich wartości zawsze są – odpowiednio – ujemne i dodatnie.

Szczególne właściwości ma klasa funkcji kary, których pierwsza pochodna ma postać:

$$h'_w(\bar{v}_w) = - \left(\frac{a_w}{\bar{v}_w} \right)^{b_w},$$

gdzie a_w i b_w są danymi wartościami stałymi.

Wybór wartości stałych a_w i b_w ma wpływ na optymalną szybkość przepływu \bar{v}_w oraz priorytet przepływu pomiędzy parą węzłów $w = (s, t)$.

Najprostszym przypadkiem sformułowanego zadania jest takie, w którym zbiory P_w są jednoelementowe – dla każdej pary $w = (s, t)$ istnieje tylko jedna trasa. Wówczas

jednoczesne wyznaczenie tras i przepływów redukuje się do zadania wyznaczenia przepływów, tzn. optymalnych wartości współczynników będących ilorzem szybkości generowanych przepływów (szybkości przepływów oferowanych przez źródło) v_w oraz szybkości przepływów przyjmowanych do sieci \bar{v}_w .

Podane sformułowanie zadania jednoczesnego wyznaczenia tras i przepływów może być rozszerzone na przypadek, gdy parę $w = (s, t)$ interpretuje się jako klasę użytkowników sieci współdzielących ten sam zbiór tras P_w . Taka interpretacja przepływu dla pary $w = (s, t)$ pozwala na ustanawianie różnych priorytetów (przez wprowadzanie różnych funkcji kar h_w) dla różnych klas użytkowników, w tym dla użytkowników współdzielących tę samą trasę. Formalnie zadanie łącznego wyznaczenia tras i przepływów w sieci jest równoważne zadaniu wyznaczenia tras w sieci.

W celu wykazania równoważności zadania wyznaczenia optymalnych tras oraz zadania jednoczesnego wyznaczenia optymalnych tras i przepływów wprowadza się dodatkową zmienną y_w , która – dla każdej pary węzłów $w \in W$ – jest wartością różnicy oferowanej i rzeczywistej szybkości przepływów, tzn.

$$y_w = v_w - \bar{v}_w = v_w - (\bar{x}_{p_1} + \bar{x}_{p_2} + \dots + \bar{x}_{p_{m_w}}),$$

gdzie:

\bar{x}_{p_j} – szybkość przepływu na trasie p_j ($p_j \in \{p_1, p_2, \dots, p_{m_w}\} = P_w$) pomiędzy parą węzłów $w = (s, t)$,
 m_w – liczba wszystkich różnych tras dla pary źródło–ujście w .

Wartość y_w jest tą częścią szybkości przepływu v_w , która jest blokowana na wejściu do sieci. Z punktu widzenia sieci wartość zmiennej y_w można interpretować jako przepełnienie, czyli przepływ w dodatkowo wprowadzanym łączu przepełnienia (rysunek 5.2).

Po zdefiniowaniu nowej funkcji kary dla zmiennej y_w :

$$H_w(y_w) = h_w(v_w - \bar{v}_w),$$

zadanie jednoczesnego wyznaczenia tras i przepływów ma postać:

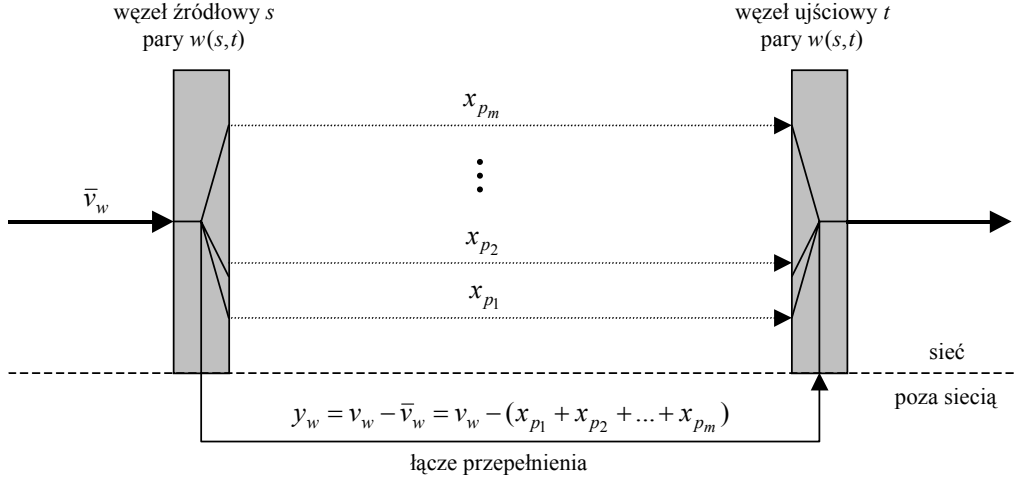
$$\tilde{d}^* = \min_{\{\bar{x}_p\}, \{y_w\}} \left[\sum_{(i,j) \in \tilde{A}} d_{ij} \left(\sum_{w \in W} \sum_{p \in P_w(i,j)} \bar{x}_p \right) + \sum_{w \in W} H_w(y_w) \right],$$

przy ograniczeniach:

$$\sum_{p \in P_w} \bar{x}_p + y_w = v_w \quad \text{dla każdego } w \in W,$$

$$\bar{x}_p \geq 0 \quad \text{dla każdego } p \in P_w \text{ i } w \in W,$$

$$y_w \geq 0 \quad \text{dla każdego } w \in W.$$



Rysunek 5.2. Równoważności zadania wyznaczania optymalnych tras oraz zadania jednoczesnego wyznaczania optymalnych tras i przepływów

Przyjęta zależność pomiędzy funkcjami kar $H_w(y_w) = h_w(v_w - \bar{v}_w)$ oraz $h_w(\bar{v}_w)$ oznacza, że jeżeli $h_w(\bar{v}_w) \rightarrow \infty$ wtedy, gdy $\bar{v}_w \rightarrow 0$ (wartość funkcji kary rośnie do nieskończoności wtedy, gdy przepływ klasy użytkowników w jest w całości odrzucony), to $H_w(y_w) \rightarrow \infty$. Przepełnienie y_w osiąga maksymalną wartość równą szybkości przepływu oferowanej przez źródło v_w , tzn. $y_w = v_w$.

Ponieważ dla ustalonej wartości szybkości oferowanego przepływu v_w spełnione są warunki:

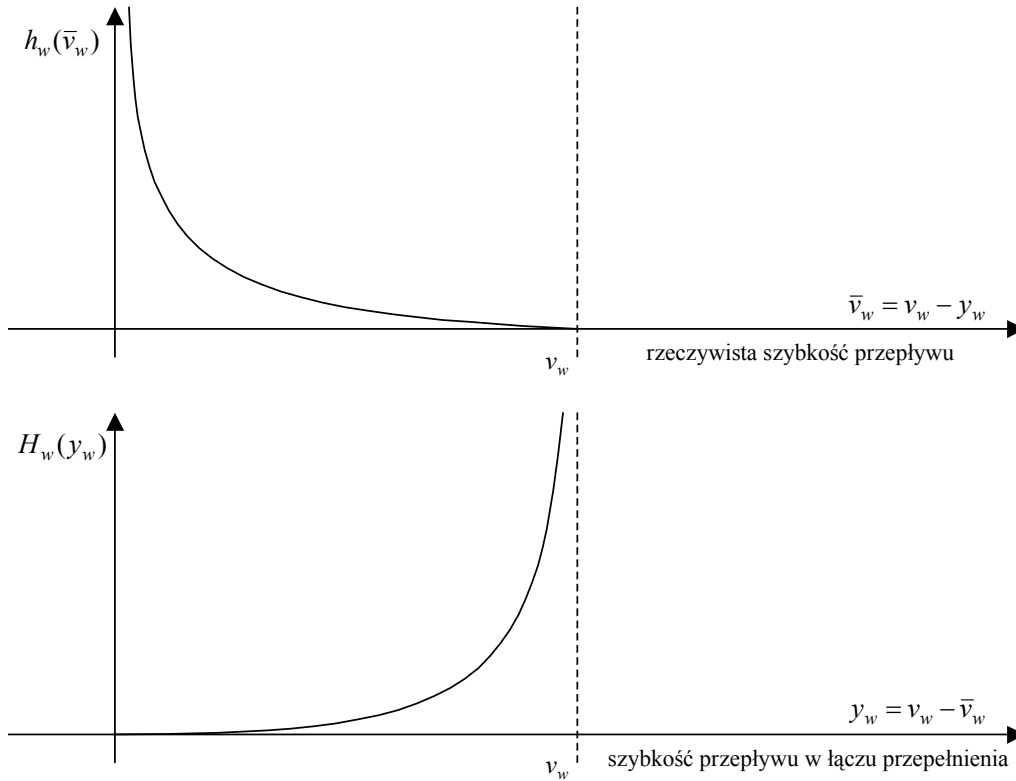
$$H_w(y_w) \rightarrow 0 \text{ wtedy, gdy } y_w = (v_w - \bar{v}_w) \rightarrow 0,$$

$$H_w(y_w) \rightarrow \infty \text{ wtedy, gdy } y_w = (v_w - \bar{v}_w) \rightarrow v_w,$$

wartość funkcji kary $H_w(y_w)$ może więc być interpretowana jako opóźnienie wnoszone przez łącze przepełnienia, a v_w jako pojemność łącza przepełnienia (rysunek 5.3).

Interpretacja funkcji kary $H_w(y_w)$ jako opóźnienia wnoszonego przez dodatkowo wprowadzoną trasę (łącze) przepełnienia oznacza, że zadanie jednoczesnego wyznaczania tras i przepływów w sieci jest matematycznie równoważne zadaniu wyznaczania optymalnych tras. W zadaniu tym każdy zbiór wszystkich tras P_w , dla każdej pary źródło–ujście w ($w \in W$), jest uzupełniany o jedną dodatkową trasę p_{m_w+1} (trasa jest łączem przepełnienia o pojemności równej szybkości przepływu generowanego przez źródło v_w); zbiór wszystkich tras dla każdej pary źródło – ujście w ($w \in W$) jest zbiorem \tilde{P}_w :

$$\tilde{P}_w = P_w \cup p_{m_w+1} = \{p_1, p_2, \dots, p_{m_w}\} \cup p_{m_w+1} = \{p_1, p_2, \dots, p_{m_w}, p_{m_w+1}\}.$$



Rysunek 5.3. Funkcje kar oraz ich interpretacje jako opóźnienie i pojemność łącza przepelnienia

Różnica pomiędzy zadaniem wyznaczania tras oraz zadaniem jednoczesnego wyznaczania tras i przepływu polega na tym, że w tym ostatnim, dla każdej pary źródło–ujście, ustanawiana jest dodatkowa trasa przejmująca odrzucaną część oferowanego przepływu.

Równoważność matematyczna omawianych zadań oznacza, że zarówno warunki optymalności rozwiązania zadań, jak i algorytmy rozwiązywania omawianych zadań są identyczne.

Po wprowadzeniu oznaczeń:

$$D(\tilde{\mathbf{x}}) = \sum_{(i,j) \in \tilde{\mathcal{A}}} d_{ij} \left(\sum_{w \in W} \sum_{p \in P_w(i,j)} \bar{x}_p \right) + \sum_{w \in W} H_w(y_w),$$

$$\frac{\partial D(\tilde{\mathbf{x}})}{\partial \bar{x}_p} = \sum_{\{(k,l)\}_p} d'_{kl}(\bar{f}_{kl}),$$

$$\frac{\partial D(\tilde{\mathbf{x}})}{\partial y_w} = H'_w(y_w) = -h'_w(v_w - \bar{v}_w),$$

gdzie:

- $D(\tilde{\mathbf{x}})$ – funkcja kryterialna,
- $\tilde{\mathbf{x}} = \{\tilde{x}_s\} = \{\{\bar{x}_p\}, \{y_w\}\}$ – wektor, którego elementy są przepływami na wszystkich trasach należących do zbioru \tilde{P}_w dla par źródło–ujście ze zbioru wszystkich par W ,
- $\partial D(\tilde{\mathbf{x}})/\partial x_p$ – wartość pochodnej cząstkowej $D(\tilde{\mathbf{x}})$ ze względu na przepływy \bar{x}_p (na wszystkich trasach dla pary źródło–ujście w z wyłączeniem trasy przepelnienia, tj. dla wszystkich tras ze zbioru P_w) obliczana dla przepływów danego wektora $\tilde{\mathbf{x}}$,
- $\partial D(\tilde{\mathbf{x}})/\partial y_w$ – wartość pochodnej cząstkowej $D(\tilde{\mathbf{x}})$ ze względu na przepływy y_w (trasy przepelnienia) obliczana dla przepływów danego wektora $\tilde{\mathbf{x}}$,
- \tilde{f}_{kl} – sumaryczny przepływ w łączu (k, l) , odpowiadający wektorowi przepływów \bar{x}_p

warunki konieczne i wystarczające optymalności wektora przepływów $\tilde{\mathbf{x}}$ mogą być formułowane analogicznie do warunków optymalności zadania wyznaczania optymalnych tras.

Wektor przepływów $\tilde{\mathbf{x}}^* = \{\tilde{x}_s^*\}$ jest optymalny, gdy spełnia ograniczenia zadania jednoczesnego wyznaczania optymalnych tras i przepływów oraz warunek konieczny i wystarczający optymalności, określony zależnością:

$$\sum_{w \in W} \sum_{s \in \tilde{P}_w} \frac{\partial D(\tilde{\mathbf{x}}^*)}{\partial \tilde{x}_s} (\tilde{x}_s - \tilde{x}_s^*) \geq 0,$$

dla wszystkich \tilde{x}_s spełniających następujące ograniczenia:

$$\sum_{s \in \tilde{P}_w} \tilde{x}_s = v_w \text{ oraz } \tilde{x}_s \geq 0 \text{ dla wszystkich } s \in \tilde{P}_w \text{ i } w \in W.$$

Warunki optymalności przepływów, formułowane dla każdej z par źródło–ujście w ($w \in W$), są następujące:

$$\sum_{s \in \tilde{P}_w} \frac{\partial D(\tilde{\mathbf{x}}^*)}{\partial \tilde{x}_s} (\tilde{x}_s - \tilde{x}_s^*) \geq 0,$$

jeżeli przepływy \tilde{x}_s są nieujemne ($\tilde{x}_s \geq 0$), a suma szybkości przepływów na wszystkich trasach ze zbioru \tilde{P}_w dla każdej pary w jest równa szybkości generowania przepływu w węźle źródłowym pary źródło–ujście w ($w \in W$):

$$\tilde{x}_{p_1} + \tilde{x}_{p_2} + \dots + \tilde{x}_{p_{m_w}} + \tilde{x}_{p_{m_w+1}} = \bar{x}_{p_1} + \bar{x}_{p_2} + \dots + \bar{x}_{p_{m_w}} + y_w = \bar{v}_w + y_w = v_w.$$

Wyznaczenie wektora przepływów

$$\tilde{\mathbf{x}}^* = \left\{ \{\bar{x}_p^*\}, \{y_w^*\} \right\},$$

tzn. przepływów w rzeczywistych i dodanych łączach sieci, jest równoważne wyznaczeniu optymalnych tras przepływów, wartości optymalnych przepływów na trasach oraz wartości optymalnych szybkości przepływów przyjmowanych do obsługi w sieci $\{\bar{v}_w^*\} = \{v_w - y_w^*\}$.

Zbiory dopuszczalnych wartości przepływów na trasach $\{\bar{x}_p^*\}$ oraz wartości przepływów akceptowanych do obsługi $\{\bar{v}_w^*\}$, będące rozwiązaniem zadania jednoczesnego wyznaczania tras i przepływów, są optymalne wtedy i tylko wtedy, gdy dla każdej pary źródło–ujście w ($w \in W$) i dla każdej trasy należącej do zbioru P_w spełnione są warunki:

$$\bar{x}_p^* > 0 \text{ tylko wtedy, gdy } \frac{\partial D(\tilde{\mathbf{x}}^*)}{\partial \bar{x}_p} \leq \frac{\partial D(\tilde{\mathbf{x}}^*)}{\partial \bar{x}_{p'}} \text{ dla każdej trasy } p' \in P_w \text{ oraz}$$

$$\frac{\partial D(\tilde{\mathbf{x}}^*)}{\partial \bar{x}_p} \leq H'_w(y_w^*) = -h'_w(v_w - y_w^*) = -h'_w(\bar{v}_w^*),$$

$$\bar{v}_w^* < v_w \text{ tylko wtedy, gdy } H'_w(y_w^*) = -h'_w(\bar{v}_w^*) \leq \frac{\partial D(\tilde{\mathbf{x}}^*)}{\partial \bar{x}_p} \text{ dla każdej trasy } p \in P_w.$$

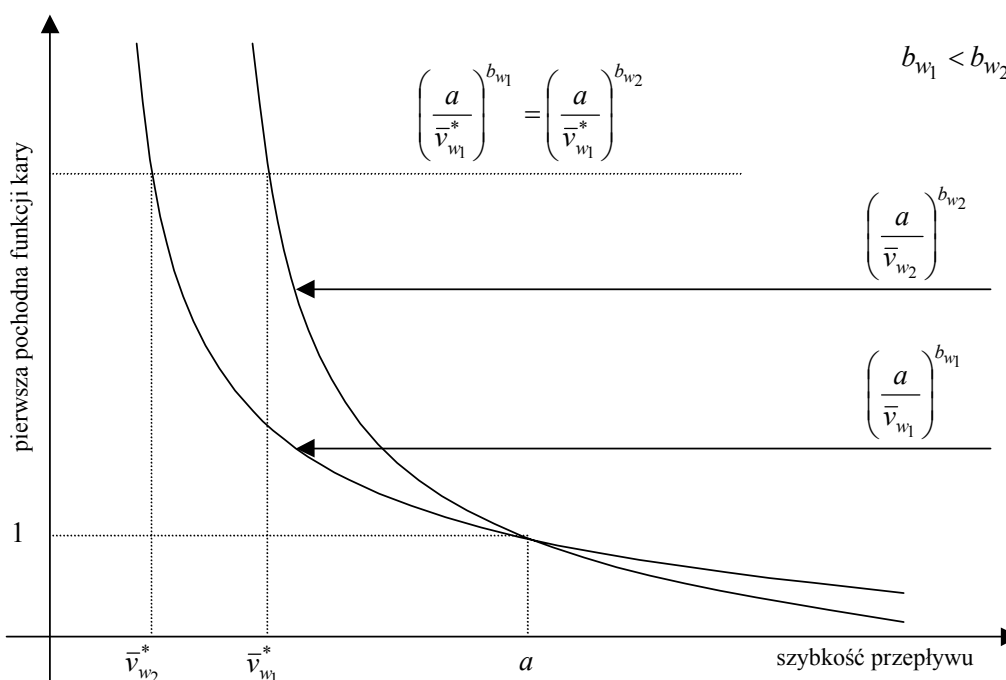
Podane warunki optymalności rozwiązania zadania zależą tylko od pochodnych funkcji d_{ij} oraz h_w . Dodanie arbitralnie wybranych stałych do funkcji d_{ij} oraz h_w nie wpływa na zmianę rozwiązania. Wynika z tego także, że dla przepływów spełniających warunek $\bar{v}_w^* > v_w$ (szybkość oferowanego przepływu jest mniejsza od wartości optymalnego przepływu w danej sieci), wartości optymalnego przepływu $\{\bar{v}_w^*\}$ nie zależą od oferowanego przepływu v_w . Zależność ta charakteryzuje pożądaną cechę mechanizmu sterowania przepływem, które jest uaktywniane wtedy, gdy oferowany przepływ jest większy od optymalnego przepływu w danej sieci. Po uaktywnieniu mechanizmu sterowania przepływem następuje odrzucanie nadmiarowej części przepływu.

Jeżeli kilka różnych par źródło–ujście w_j ($w_j \in \{w_1, w_2, \dots, w_n\}$) korzysta z tych samych tras w sieci, tzn. $P_{w_1} = P_{w_2} = \dots = P_{w_n}$, to z warunków optymalności wynika, że oferowane przepływy są ograniczane przez procedurę sterowania przepływem $(\bar{v}_{w_1}^* < v_{w_1}, \bar{v}_{w_2}^* < v_{w_2}, \dots, \bar{v}_{w_n}^* < v_{w_n})$ wtedy, gdy:

$$-h'_{w_1}(\bar{v}_{w_1}^*) = \dots = -h'_{w_n}(\bar{v}_{w_n}^*) = \min_{p \in P_{w_1}} \left\{ \frac{\partial D(\tilde{\mathbf{x}}^*)}{\partial \bar{x}_p} \right\} = \dots = \min_{p \in P_{w_n}} \left\{ \frac{\partial D(\tilde{\mathbf{x}}^*)}{\partial \bar{x}_p} \right\}.$$

Przykład 5.1

Dla funkcji kary, których pierwsze pochodne mają postać $h'_w(\bar{v}_w) = -(a_w/\bar{v}_w)^{b_w}$, tzn. są definiowane wartościami stałych a_w i b_w , odpowiedni dobór wartości tych liczb pozwala m.in. na wprowadzanie innych priorytetów dla różnych par źródło–ujście (dla różnych klas użytkowników).



Rysunek 5.4. Wpływ wartości parametrów funkcji kary na jakość obsługi jednocześnie obsługiwanych przepływów

Jeżeli $n = 2$, to możliwe są m.in. następujące przypadki (rysunek 5.4):

➤ jeżeli $a_{w_1} \neq a_{w_2}$ i $b_{w_1} = b_{w_2}$, to z równości wartości pierwszych pochodnych funkcji kar: $-(a_{w_1}/\bar{v}_{w_1}^*)^{b_{w_1}} = -(a_{w_2}/\bar{v}_{w_2}^*)^{b_{w_2}}$ wynika, że $\bar{v}_{w_1}^*/\bar{v}_{w_2}^* = a_{w_1}/a_{w_2}$. Względne wartości parametrów a_w , przypisanych poszczególnym parom węzłów, decydują o względnym udziale optymalnych przepływów różnych par węzłów w optymalnym sumarycznym przepływie:

➤ jeżeli $a_{w_1} = a_{w_2} = a$, $b_{w_1} > b_{w_2}$ oraz $\bar{v}_{w_1}^*, \bar{v}_{w_2}^* < a$, to z równości wartości pierwszych pochodnych funkcji kar: $-(a/\bar{v}_{w_1}^*)^{b_{w_1}} = -(a/\bar{v}_{w_2}^*)^{b_{w_2}}$ wynika, że przepływ pierwszej pary węzłów $\bar{v}_{w_1}^*$ (dla której wartość parametru b_{w_1} jest większa od b_{w_2}

funkcji kary dla drugiego przepływu) jest większy od przepływu drugiej pary węzłów $\bar{v}_{w_2}^* : \bar{v}_{w_1}^* > \bar{v}_{w_2}^*$,

➤ jeżeli $a_{w_1} = a_{w_2} = a$, $b_{w_1} > b_{w_2}$ oraz $\bar{v}_{w_1}^*, \bar{v}_{w_2}^* > a$, to przepływ pierwszej pary węzłów $\bar{v}_{w_1}^*$ (dla której wartość parametru b_{w_1} jest większa od b_{w_2} funkcji kary dla drugiego przepływu) jest mniejszy od przepływu drugiej pary węzłów $\bar{v}_{w_2}^* : \bar{v}_{w_1}^* < \bar{v}_{w_2}^*$, tzn. wartości parametrów b_w decydują o względnych priorytetach optymalnych przepływów różnych par węzłów w optymalnym sumarycznym przepływie.

W warunkach względnie większego obciążenia systemu obsługi ($\bar{v}_{w_1}^* > a, \bar{v}_{w_2}^* > a$) oraz dla jednakowej i ustalonej wartości parametru a_w przepływ, dla którego wartość parametru b_w jest większa ($b_{w_1} > b_{w_2}$), jest w mniejszym stopniu dyskryminowany niż wtedy, gdy obciążenie systemu obsługi jest względnie mniejsze ($\bar{v}_{w_1}^* < a, \bar{v}_{w_2}^* < a$).

Względne wartości parametrów a_w definiują względne udziały jednocześnie obsługiwanych przepływów w sumarycznym przepływie, względne wartości parametrów b_w decydują natomiast o udziale odrzucanych ilości poszczególnych przepływów w sumarycznej ilości odrzucanych przepływów.

Przykład 5.2

Zadanie jednoczesnego wyznaczania tras i przepływów, gdy przepływ pomiędzy dwoma węzłami jest obsługiwany przez łącze o pojemności C , jest sformułowane następująco:

$$d^*(\bar{v}) = \min_{\bar{v}} [f(\bar{v}) + H(y)] = \min_{\bar{v}} [f(\bar{v}) + h(v - y)] = \min_{\bar{v}} \left[\frac{\bar{v}}{C - \bar{v}} + \frac{a}{\bar{v}} \right],$$

gdzie:

- v – oferowana szybkość przepływu, spełniająca warunek:
 $v = \bar{v} + y$,
- \bar{v} ($\bar{v} \geq 0$) – szybkość przepływu przyjmowanego do obsługi, tzn. kierowanego do obsługi przez dwa równoległe łącza o pojemności C ,
- y ($y \geq 0$) – szybkość przepływu odrzucanego przez mechanizm sterowania przepływem (część oferowanej szybkości przepływu), tzn. kierowanego do łącza przepelnienia,
- $f(\bar{v}) = \bar{v}/(C - \bar{v})$ – opóźnienie wnoszone przez łącze o pojemności C dla szybkości przepływu \bar{v} ,
- $H(y)$ i $h(\bar{v})$ – funkcje kary spełniające warunek: $H(y) = h(\bar{v}) = h(v - y)$,
- $h(\bar{v}) = a/\bar{v}$ – funkcja klasy $(a/\bar{v})^b$, gdzie $b = 1$,
- a – stała.

Zgodnie z warunkami optymalności przepływu brak konieczności sterowania przepływem, tzn. brak przepływu w łączu przepelnienia ($\bar{v} = v, y = 0$) występuje wtedy, gdy pierwsza pochodna długości łącza rzeczywistego jest mniejsza od pierwszej pochodnej długości łącza przepelnienia:

$$\frac{C}{(C-v)^2} < \frac{a}{v^2}.$$

Rozwiązanie powyższej nierówności:

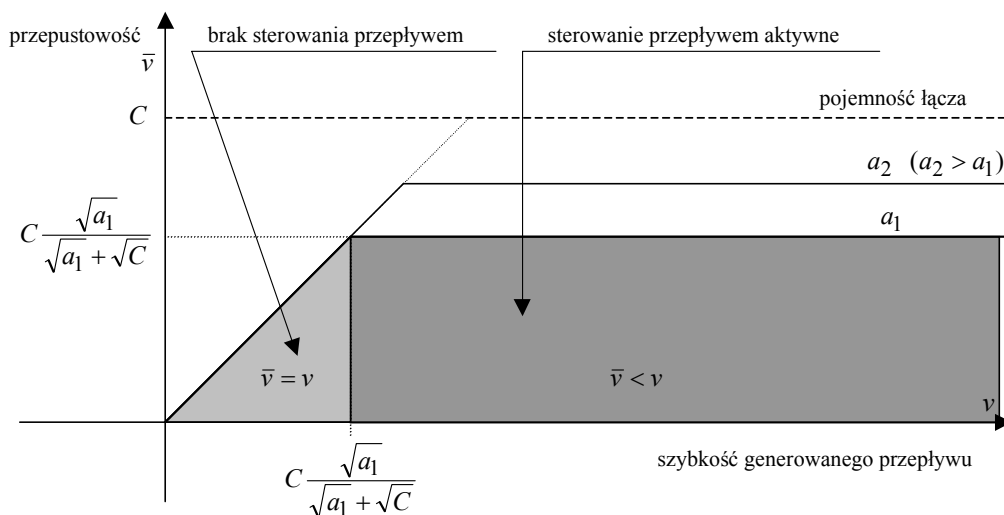
$$v < \bar{v}_g = C \frac{\sqrt{a}}{\sqrt{a} + \sqrt{C}},$$

określa zakres wartości szybkości oferowanego przepływu, dla których przepływ jest w całości obsługiwany przez łącze pomiędzy źródłem i ujściem.

Warunkiem konieczności sterowania przepływem, polegającego na odrzucaniu części przepływu ($\bar{v} < v, y > 0$), jest równość pierwszych pochodnych długości łączy: rzeczywistego i przepelnienia, tzn. wtedy, gdy $v > v_g$. Jeżeli spełniony jest warunek $v > v_g$, to przepływ w łączu rzeczywistym $\bar{v} = C\sqrt{a}/(\sqrt{a} + \sqrt{C})$, niezależnie od szybkości oferowanego przepływu v (rysunek 5.5).

Szybkość przepływu zaakceptowanego do obsługi \bar{v} jest równa przepustowości omawianego systemu obsługi:

$$\bar{v} = \begin{cases} v & \text{gdy } v \leq C\sqrt{a}(\sqrt{a} + \sqrt{C}) \\ C\sqrt{a}(\sqrt{a} + \sqrt{C}) & \text{gdy } v > C\sqrt{a}(\sqrt{a} + \sqrt{C}) \end{cases}.$$



Rysunek 5.5. Obszary braku aktywności i aktywności sterowania przepływem

Wartość przepustowości \bar{v} dla oferowanego przepływu v , spełniającego warunek:

$$v > \frac{C\sqrt{a}}{\sqrt{a} + \sqrt{C}},$$

nie zależy od szybkości oferowanego przepływu, zależy natomiast od pojemności łącza C i wartości stałej a – parametru funkcji kary. Zmiana maksymalnej przepustowości jest możliwa przez zmianę wartości a : jeżeli $a \rightarrow \infty$, to $\bar{v} \rightarrow C$. Wzrost wartości maksymalnej przepustowości powoduje zwiększenie wartości opóźnienia: jeżeli $\bar{v} \rightarrow C$, to $d^*(\bar{v}) \rightarrow \infty$.

5.7. SPRAWIEDLIWY ROZDZIAŁ ZASOBÓW W SIECI W ZADANIACH STEROWANIA PRZEPLÝWEM

Związana z procedurami sterowania przepływem konieczność ograniczania ilości ruchu przyjmowanego do obsługi w sieci lub jej fragmencie oznacza, że część ruchu generowanego przez źródła jest dodatkowo opóźniana lub odrzucana. Od procedur sterowania przepływem (w zasadzie procedur szeregowania pakietów przepływów) wymaga się, aby odrzucanie lub opóźnianie pakietów dotyczyło nie tylko wybranych, ale wszystkich obsługiwanych przepływów. Ta zasada dyskryminacji ruchu związana jest z pojęciem sprawiedliwego rozdziału zasobów sieci (ang. *fairness*).

Realizacja zasad sprawiedliwego rozdziału zasobów sieci powoduje dodatkową złożoność procedur sterowania przepływem ze względu na to, że maksymalizacja przepustowości sieci jest często sprzeczna z zasadą sprawiedliwego rozdziału zasobów.

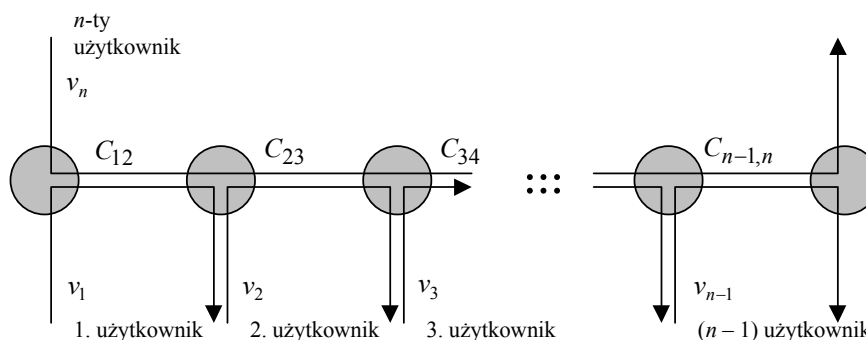
Zasadę sprawiedliwego rozdziału zasobów, według której ograniczanie szybkości (intensywności) napływu pakietów przez procedurę sterowania przepływem dotyczy wszystkich przepływów lub klas ruchu obsługiwanych przez sieć, a zwłaszcza tych, które ponoszą największą odpowiedzialność za przeciążenie sieci, zilustrowano na rysunku 5.6.

Zakładając, że n użytkowników, z których każdy generuje ruch o intensywności $v_i = v$ [jednostka danych na jednostkę czasu] ($i = 1, 2, \dots, n$), obsługuje zbiór szeregowo połączonych łączy o jednakowej pojemności $C_{ij} = C = v$ ($i, j = 1, 2, \dots, n, i \neq j$) oraz przyjmując dodatkowo, że ruch generowany przez $n - 1$ użytkowników obsługują tylko pojedyncze łącza, a ruch generowany przez jednego n -tego użytkownika obsługują $n - 1$ szeregowo połączone łącza, możliwe są m.in. następujące rozwiązania zadania rozdziału zasobów:

➤ **maksymalizujące przepustowość** – ruch generowany przez $n - 1$ użytkowników żądających dostępu do pojedynczych łączy jest przyjmowany do obsługi ($\bar{v}_1 = \bar{v}_2 = \dots = \bar{v}_{n-1} = v$), natomiast ruch generowany przez użytkownika żądającego dostępu jednocześnie do $n - 1$ łączy jest odrzucany ($\bar{v}_n = 0$); w tym przypadku mak-

symalna przepustowość systemu obsługi jest równa pojemności systemu obsługi i wynosi $(n-1)C$,

➤ **jednakowe dla wszystkich użytkowników** – ilość odrzucanego ruchu dla każdego z użytkowników jest proporcjonalna do ilości ruchu generowanego, tzn. ograniczenia ruchu są jednakowe dla wszystkich użytkowników ($\bar{v}_1 = \bar{v}_2 = \dots = \bar{v}_{n-1} = v_n = v/2$); w tym przypadku maksymalna przepustowość systemu obsługi wynosi $nC/2$ i jest mniejsza od pojemności systemu obsługi równej $(n-1)C$: $nC/2 \leq (n-1)C$ dla $n \geq 2$.



$$C_{12} = C_{21} = \dots = C_{n-1,n} = C = v \quad \text{maksymalna przepustowość: } \bar{v}_1 = \bar{v}_2 = \dots = \bar{v}_{n-1} = v$$

$$v_1 = v_2 = \dots = v_n = v \quad \bar{v}_n = 0$$

$$\text{sprawiedliwy rozdział: } \bar{v}_1 = \bar{v}_2 = \dots = \bar{v}_{n-1} = \bar{v}_n = v/2$$

Rysunek 5.6. Schemat systemu obsługi z rozdziałem zasobów maksymalizującym przepustowość i ze sprawiedliwym (jednakowym) rozdziałem zasobów

Rozdział zasobów, proporcjonalny do żądań poszczególnych użytkowników (w szczególności jednakowy dla wszystkich użytkowników), jest jednym z przykładów sprawiedliwego rozdziału zasobów.

Inny przykład sprawiedliwego rozdziału zasobów jest pokazany na rysunku 5.7. W przykładzie tym użytkownicy są obsługiwani przez zbiór szeregowo połączonych łączy, z których $n-1$ ma pojemność $C_{ij} = C$ ($i, j = 1, 2, \dots, n, i \neq j$), a łącze $(n-1, n)$ ma pojemność $C_{n-1,n} = 2C$.

Przyjmując, że ruch generowany przez $n-1$ użytkowników jest obsługiwany tylko przez pojedyncze łącza, a ruch generowany przez jednego n -tego użytkownika jest obsługiwany przez $n-1$ szeregowo połączonych łączy, możliwe są m.in. następujące rozwiązania zadania rozdziału zasobów:

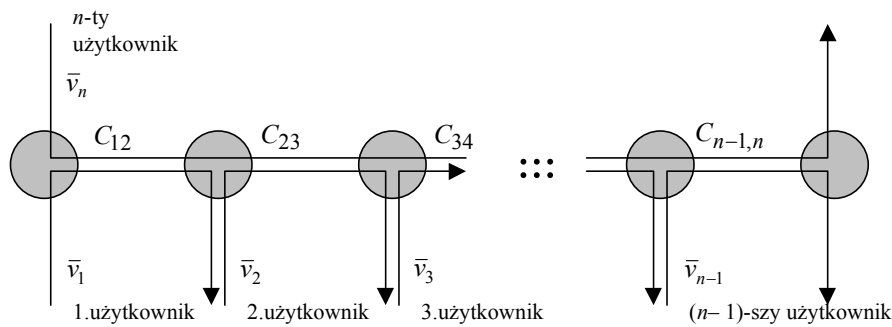
➤ **maksymalizujące przepustowość:** $n-1$ użytkownikom żądającym dostępu do pojedynczych łączy są przyznawane pojemności poszczególnych łączy, tzn. maksymalny przepływ jest równy odpowiednio: $\bar{v}_{1\max} = \bar{v}_{2\max} = \dots = \bar{v}_{(n-2)\max} = C$ oraz $\bar{v}_{(n-1)\max} = 2C$, n -temu użytkownikowi natomiast przyznawana jest pojemność zerowa, tzn. maksy-

malny przepływ jest zerowy: $\bar{v}_{n\max} = 0$; w tym przypadku maksymalna przepustowość systemu obsługi jest równa pojemności systemu obsługi:

$$\bar{v}_{\max} = \bar{v}_{1\max} + \bar{v}_{2\max} + \dots + \bar{v}_{(n-2)\max} + \bar{v}_{(n-1)\max} + \bar{v}_{n\max} = (n-2)C + 2C + 0 = nC,$$

➤ **sprawiedliwe:** $n - 2$ użytkownikom, żądającym dostępu do pojedynczych łączy, oraz n -temu użytkownikowi, żądającemu dostępu do wszystkich łączy, możliwe jest przyznanie jednakowych części pojemności poszczególnych łączy, tzn. maksymalny przepływ jest równy: $\bar{v}_{1\max} = \bar{v}_{2\max} = \dots = \bar{v}_{(n-2)\max} = \bar{v}_{n\max} = C/2$, a $n - 1$ użytkownikowi możliwe jest przyznanie pojemności umożliwiającej przepływ $\bar{v}_{(n-1)\max} = 3C/2$; w tym przypadku maksymalna przepustowość systemu obsługi jest równa pojemności systemu obsługi:

$$\bar{v}_{\max} = \bar{v}_{1\max} + \bar{v}_{2\max} + \dots + \bar{v}_{(n-2)\max} + \bar{v}_{(n-1)\max} + \bar{v}_{n\max} = \frac{(n-1)C}{2} + \frac{3C}{2} + \frac{C}{2} = \frac{(n+2)C}{2}.$$



$C_{12} = C_{21} = \dots = C_{n-2,n-1} = C$	maksymalna przepustowość:	$\bar{v}_1 = \bar{v}_2 = \dots = \bar{v}_{n-2} = C$
$C_{n-1,n} = 2C$		$\bar{v}_{n-1} = 2C$
		$\bar{v}_n = 0$
	sprawiedliwy rozdział:	$\bar{v}_1 = \bar{v}_2 = \dots = \bar{v}_{n-2} = \bar{v}_n = C/2$
		$\bar{v}_{n-1} = 3C/2$

Rysunek 5.7. Schemat systemu obsługi z rozdzielaniem zasobów maksymalizującym przepustowość i ze sprawiedliwym (różnym) rozdziałem zasobów

Z podanych przykładów wynika, że sprawiedliwy (w różnym stopniu) rozdział zasobów systemu obsługi odbywa się za cenę zmniejszenia efektywności wykorzystania zasobów systemu; w zamian jakość obsługi wszystkich lub wybranych użytkowników systemu obsługi jest mniejsza od żądanej.

Celem sterowania przepływem w sieci jest znajdowanie kompromisu pomiędzy sprawiedliwym rozdziałem zasobów i rozdziałem powodującym odrzucanie ruchu tych użytkowników, którzy mają największy udział w przeciążeniu sieci.

W praktyce stosuje się wiele różnych procedur sterowania przepływem, prowadzących do sprawiedliwego rozdziału zasobów pomiędzy przepływy współzawodniczące o te zasoby. Jednym z rozwiązań jest grupowanie użytkowników (lub przepływów) w klasy różniące się priorytetami. Sprawiedliwy rozdział zasobów może być ograniczony do przepływów należących do klas o najwyższych priorytetach. Jednocześnie istnieje wiele mechanizmów sterowania przepływem, których zasadniczą wadą jest to, że nie realizują zasady sprawiedliwego rozdziału zasobów. Jednym z przykładów takich procedur są mechanizmy progowe, które powodują odrzucanie jednostek danych w przypadku przepełnienia pamięci buforowych jednego z węzłów sieci, niezależnie od stanu zajętości pamięci buforowych innych węzłów sieci.

5.8. MAKSYMALNO-MINIMALNE STEROWANIE PRZEPŁYWEM

Głównym ograniczeniem stosowania metod sterowania przepływem z różnymi odmianami mechanizmu okna przesuwającego jest trudność implementacji idei sprawiedliwego rozdziału zasobów pomiędzy przepływy współdzielące zasoby sieci.

Realizacja dowolnego algorytmu sprawiedliwego rozdziału zasobów wymaga informacji o indywidualnym i całkowitym zapotrzebowaniu źródeł przepływów na zasoby. Pod tym względem sformułowania i algorytmu rozwiązywania zadań sprawiedliwego rozdziału zasobów są zbliżone do zadań wyznaczania tras w sieciach. Zadania te mogą być równoważne lub sprowadzalne do zadań optymalnego rozdziału zasobów.

Większość rozwiązań zadania sprawiedliwego rozdziału zasobów opiera się na koncepcji, zgodnie z którą efektywny i sprawiedliwy rozdział zasobów sieci jest możliwy przez ograniczanie szybkości przepływów przyjmowanych do obsługi w sieci. Realizacja tej koncepcji oznacza także, że procedury sterowania przepływem są uaktywniane i wyłączane w zależności od obciążenia sieci. Małe obciążenie sieci zwykle nie wymaga aktywności omawianych procedur. Zwiększenie obciążenia sieci, którego konsekwencją mogą być przeciążenia i pogorszenie jakości usług sieci, wymaga aktywacji procedur sterowania przepływami.

Idea sprawiedliwego rozdziału zasobów jest identyfikowana z maksymalno-minimalnym sterowaniem przepływem (ang. *max-min flow*), zgodnie z którym maksymalizacja wykorzystania zasobów systemu obsługi następuje po rozdziale zasobów użytkownikom dysponującym najmniejszymi ich ilościami. Tak sformułowane zadanie jest równoważne zadaniu maksymalizacji przydziału zasobów i -temu użytkownikowi po spełnieniu następującego ograniczenia: przyrost ilości zasobów dla i -tego użytkownika nie powoduje zmniejszenia ilości zasobów dla innego użytkownika, dysponującego zasobami w ilości równej lub mniejszej od ilości zasobów będących do dyspozycji i -tego użytkownika.

Zadanie sprawiedliwego rozdziału zasobów sieci jest formułowane jako zadanie wyznaczania stałych, pojedynczych tras w sieci. Modelem sieci jest tu graf skierowany

$G = (N, \Gamma)$, a S jest zbiorem wszystkich przepływów (sesji) w sieci. W modelu tym z każdym z przepływów s ($s \in S$) związana jest jedna stała trasa w sieci. W takim przypadku indeks przepływu jest jednocześnie indeksem trasy. Jeżeli z jednej trasy w sieci korzysta kilka przepływów, to kilka indeksów przepływów s odnosi się do tej samej trasy.

Jeżeli \bar{v}_s ($\bar{v}_s \geq 0, s \in S$) jest szybkością akceptowanego przez sieć s -tego przepływu, to sumaryczna szybkość przepływu f_{ij} w (i, j) -tym ($(i, j) \in \Gamma$) łączu o pojemności C_{ij} , spełniająca dla każdego $(i, j) \in \Gamma$ warunek $f_{ij} \leq C_{ij}$, jest określona wyrażeniem:

$$f_{ij} = \sum_{s \in S} \delta_s(i, j) \bar{v}_s,$$

$$\text{w którym } \delta_s(i, j) = \begin{cases} 1 & \text{gdy } (i, j) \in s \\ 0 & \text{gdy } (i, j) \notin s \end{cases}.$$

Rozwiązanie zadania polega na wyznaczeniu takiego wektora szybkości przepływów \bar{v} , którego elementami są wartości szybkości przepływów \bar{v}_s ($s \in S$), spełniające warunki sprawiedliwości przepływu w sensie maksymalno-minimalnego sterowania przepływem.

Wektor szybkości przepływów \bar{v} jest przepływem maksymalno-minimalnym, gdy:

➤ wektor \bar{v} jest wektorem dopuszczalnym; elementy \bar{v}_s ($s \in S$) wektora \bar{v} są przepływami dopuszczalnymi, tzn. $\bar{v}_s \geq 0$ dla każdego $s \in S$ i $f_{ij} \leq C_{ij}$ dla każdego łącza $(i, j) \in \Gamma$,

➤ szybkość dowolnego przepływu \bar{v}_s ($s \in S$) nie może być zwiększona inaczej, jak tylko przez zmniejszanie szybkości dowolnego innego przepływu $\bar{v}_{s'}$ ($s' \in S \setminus s$) takiego, że $\bar{v}_{s'} \leq \bar{v}_s$.

Innymi słowy, wektor szybkości przepływów \bar{v} jest przepływem maksymalno-minimalnym, gdy jest wektorem dopuszczalnym dla każdego innego wektora dopuszczalnego \tilde{v} oraz dla każdego s ($s \in S$); jeżeli $\bar{v}_s < \tilde{v}_s$ ($\bar{v}_s \in \bar{v}$, $\tilde{v}_s \in \tilde{v}$), to dla pewnego s' ($s' \in S \setminus s$) spełnione są warunki: $\bar{v}_s \geq \bar{v}_{s'}$ i $\bar{v}_{s'} > \tilde{v}_{s'}$.

Charakterystyczną i istotną właściwością wektora maksymalno-minimalnych, sprawiedliwych przepływów jest to, że dla każdego s -tego przepływu istnieje takie łącze (i, j) , należące do s -tej trasy, dla którego spełnione są warunki:

$$\text{➤ } f_{ij} = C_{ij},$$

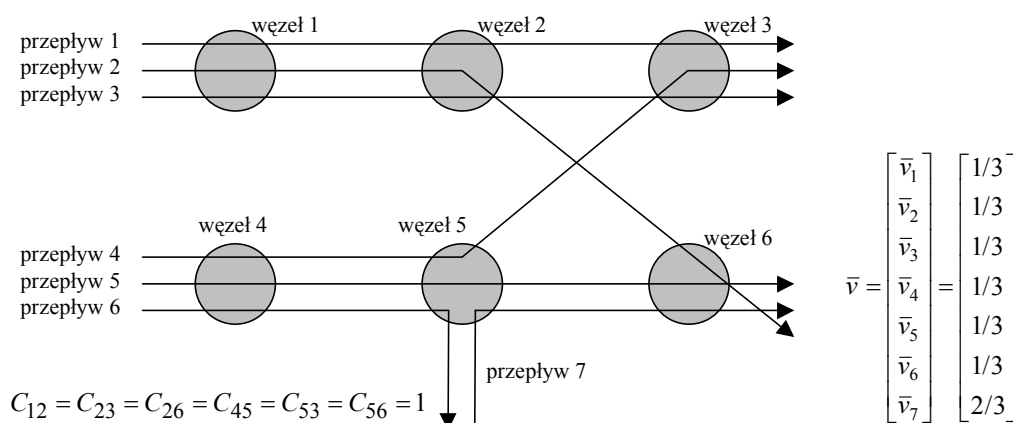
➤ szybkość przepływu \bar{v}_s jest przynajmniej tak duża, jak szybkość pewnego innego przepływu korzystającego z danego łącza.

Łącze (i, j) jest wąskim gardłem (ang. *bottleneck arc*) s -tej trasy.

Jeżeli nie istniałoby łącze spełniające powyższe warunki, to szybkość przepływu \bar{v}_s mogłaby być zwiększana bez konieczności zmniejszania szybkości innego przepływu $\bar{v}_{s'}$, takiego że $\bar{v}_{s'} \leq \bar{v}_s$; przeczyłoby więc założeniu o warunkach przepływu maksymalno-minimalnego.

Jednocześnie, jeżeli podane warunki są spełnione dla wszystkich przepływów pewnego wektora \bar{v} , to jest on wektorem maksymalno-minimalnych przepływów. Wynika to z tego, że jeżeli szybkość przepływu \bar{v}_s byłaby zwiększana, to szybkość co najmniej jednego innego przepływu obsługiwanego przez łącze (i, j) (wąskie gardło) musiałaby być zmniejszana.

Wektor maksymalno-minimalnych przepływów w przykładowej sieci przedstawiono na rysunku 5.8.



Rysunek 5.8. Maksymalno-minimalne przepływy w przykładowej sieci

Jeżeli pojemności wszystkich łączy w sieci są jednakowe i równe 1 ($C_{12} = C_{23} = C_{26} = C_{45} = C_{53} = C_{56} = 1$), to wąskimi gardłami dla przepływów (tras) 1, 2, 3, 4, 5, 6, 7 są odpowiednio łącza: (1, 2), (1, 2), (1, 2), (4, 5), (4, 5), (4, 5), (5, 6); dla każdego przepływu (trasy) istnieje łącze będące wąskim gardłem. Łącza (2, 3), (2, 6), (5, 3), (5, 6) nie są wąskimi gardłami dla żadnego z obsługiwanych przepływów, ponieważ pojemności tych łączy sieci nie są w pełni wykorzystane przez obsługiwane przepływy, tzn.: $f_{23} = f_{56} = 2/3$, $f_{26} = f_{53} = 1/3$. Łącze (5, 6) nie jest wąskim gardłem dla piątego przepływu, ponieważ jest ono współdzielone przez dwa przepływy, z tym że $\bar{v}_5 = 1/3 < \bar{v}_7 = 2/3$; łącze to jest wąskim gardłem dla siódmego przepływu.

Z właściwości wektora maksymalno-minimalnych przepływów bezpośrednio korzysta się w algorytmach wyznaczania takich przepływów. Algorytmy wyznaczania przepływów maksymalno-minimalnych opierają się na powszechnie stosowanej, w różnych zadaniach wyznaczania przepływów w sieciach, idei równomiernego zwiększania szybkości przepływów, aż do nasycenia wszystkich tras w sieci. Ogólna koncepcja algorytmu wyznaczania wartości wektora maksymalno-minimalnych przepływów polega na tym, że:

➤ stanem początkowym jest taki, w którym szybkości wszystkich przepływów są zerowe,

➤ w pierwszym kroku algorytmu szybkości przepływów na wszystkich trasach sieci są zwiększane równomiernie aż do chwili, gdy jedno lub więcej łączy zostanie nasyconych, tzn. kiedy sumaryczna szybkość przepływu w łączy (w łączach) jest równa pojemności łącza: $f_{ij} = C_{ij}$ dla $(i, j) \in \Gamma$,

➤ wszystkie przepływy, korzystające z nasyconego łącza, mają tę samą szybkość; łącze nasycone jest wąskim gardłem przepływów obsługiwanych przez to łącze,

➤ w kolejnym kroku algorytmu, szybkości wszystkich przepływów, które nie korzystają z wcześniej nasyconych łączy, są zwiększane równomiernie aż do chwili, gdy kolejne łącze (lub łącza) osiąga stan nasycenia,

➤ nasycenie nowego łącza mogą wywoływać zarówno przepływy, które powodowały nasycenie wcześniej nasyconych łączy, jak i przepływy, których zwiększone szybkości prowadzą do nasycenia nowego łącza; wszystkie przepływy należące do drugiej grupy mają taką samą szybkość,

➤ nowo nasycone łącze jest wąskim gardłem dla wszystkich tych przepływów, które nie były powodem nasycenia wcześniej nasyconych łączy,

➤ powtarzanie procedury równomiernego zwiększania szybkości przepływów nieobsługiwanych przez wcześniej nasycone łącza trwa dopóty, dopóki wszystkie przepływy nie są obsługiwane przez co najmniej jedno łącze nasycone.

Po oznaczeniu:

k – numer kroku algorytmu ($k = 1, 2, \dots$),

$A^{(k)}$ – zbiór wszystkich łączy w sieci nie będących w stanie nasycenia przed rozpoczęciem realizacji k -tego kroku algorytmu:

$A^{(k)} \subset \Gamma (\dots \subset A^{(k+1)} \subset A^{(k)} \subset A^{(k-1)} \subset \dots)$, gdzie Γ jest zbiorem wszystkich łączy sieci,

$B^{(k)}$ – zbiór wszystkich przepływów nieobsługiwanych przez żadne z nasyconych łączy przed rozpoczęciem realizacji k -tego kroku algorytmu:

$B^{(k)} \subset S (\dots \subset B^{(k+1)} \subset B^{(k)} \subset B^{(k-1)} \subset \dots)$, gdzie S jest zbiorem wszystkich przepływów,

$I_{ij}^{(k)}$ – liczba wszystkich przepływów przez łącze $(i, j) \in B^{(k)}$, tzn. liczba przepływów, których przyrosty szybkości będą korzystały z niewykorzystanej w poprzednich $(k - 1)$ krokach algorytmu, pojemności łącza (i, j) ,

$\Delta^{(k)}$ – przyrost szybkości przepływów należących do zbioru $B^{(k)}$ w czasie realizacji k -tego kroku algorytmu,

$f_{ij}^{(k)}$ – sumaryczna szybkość przepływu przez łącze $(i, j) \in \Gamma$ przed rozpoczęciem realizacji k -tego kroku algorytmu,

$\bar{v}_s^{(k)}$ – zaakceptowana do obsługi w sieci szybkość s -tego przepływu przed rozpoczęciem realizacji k -tego kroku algorytmu,

algorytm wyznaczenia wektora maksymalno-minimalnych przepływów jest następujący:

- warunki początkowe: $k = 1$,
 $f_{ij}^{(0)} = 0$ dla każdego $(i, j) \in \tilde{A}$,
 $\bar{v}_s^{(0)} = 0$ dla każdego $s \in S$,
 $\tilde{A}^{(1)} = \tilde{A}$, $\hat{A}^{(1)} = S$,
- krok 1: $l_{ij}^{(k)} := \sum_{s \in \hat{A}^{(k)}} \delta_s(i, j)$ dla każdego $(i, j) \in \tilde{A}$,
- krok 2: $\Delta^{(k)} := \min_{(i, j) \in \hat{A}^{(k)}} [C_{ij} - f_{ij}^{(k-1)}] / l_{ij}^{(k)}$,
- krok 3: $\bar{v}_s^{(k)} = \begin{cases} \bar{v}_s^{(k-1)} + \Delta^{(k)} & \text{gdy } s \in B^{(k)} \\ \bar{v}_s^{(k-1)} & \text{gdy } s \in S \setminus B^{(k)} \end{cases}$,
- krok 4: $f_{ij}^{(k)} = \sum_{s \in S} \delta_s(i, j) \bar{v}_s^{(k)}$ dla każdego $(i, j) \in \tilde{A}$,
- krok 5: $A^{(k+1)} := \{(i, j) \in \tilde{A} \mid C_{ij} - f_{ij}^{(k)} > 0\}$,
- krok 6: $\hat{A}^{(k+1)} := \{s \in S \mid \sum_{(i, j) \in A^{(k+1)}} \delta_s(i, j) = 0\}$,
- krok 7: jeżeli $\hat{A}^{(k+1)} \neq \emptyset$, to $k := k + 1$ i powrót do kroku 2,
jeżeli $\hat{A}^{(k+1)} = \emptyset$, to stop.

Realizacja k -tego kroku algorytmu polega na równomiernym zwiększaniu szybkości wszystkich przepływów należących do zbioru $B^{(k)}$ o wyznaczoną w tym kroku wartość przyrostu $\Delta^{(k)}$. Wynikiem zwiększenia szybkości jest nasycenie co najmniej jednego łącza w sieci oraz równa szybkość wszystkich przepływów należących do zbioru $B^{(k)}$. Szybkość wszystkich przepływów, należących do zbioru $B^{(k)}$ i obsługiwanych przez nasycane w tym kroku łącze (łącza), jest nie mniejsza od szybkości dowolnego innego przepływu obsługiwanego przez to łącze. Łącze nasycone w k -tym kroku jest dla przepływów, których szybkości są największe, wąskim gardłem.

Rozwiązaniem zadania wyznaczania maksymalno-minimalnych przepływów jest wektor $\bar{v} = \bar{v}^{(k)}$, gdzie k jest numerem ostatniej iteracji algorytmu. Ze względu na to, że w ograniczeniach zadania nie występują oferowane szybkości przepływów, rozwiązaniem zadania jest wektor maksymalno-minimalnych przepływów bez ograniczeń. Wartości otrzymanego wektora $\bar{v}^{(k)}$ mogą być interpretowane jako wartości maksymalne szybkości poszczególnych przepływów, tzn. dla każdego s -tego przepływu ($s \in S$) spełniony jest warunek: $\bar{v}_s \leq v_s$, gdzie v_s jest oferowaną szybkością przepływu.

Gdy w zbiorze przepływów S znajdują się przepływy, których oferowane szybkości są mniejsze od odpowiednich wartości wektora maksymalno-minimalnych przepływów bez ograniczeń, wówczas zbiór wszystkich przepływów S jest sumą dwóch rozłącznych podzbiorów: $S = S' \cup S''$:

➤ do S' należą wszystkie te przepływy, których zapotrzebowanie na zasoby sieci jest w pełni zaspokajane, tzn. dla których oferowane i akceptowane szybkości są równe: $v_s = \bar{v}_s$,

➤ do S'' należą wszystkie te przepływy, których zapotrzebowanie na zasoby sieci nie jest w pełni zaspokajane, tzn. dla których oferowane szybkości są większe od akceptowanych szybkości: $v_s > \bar{v}_s$.

Podane podstawowe sformułowanie zadania wyznaczania przepływów maksymalno-minimalnych może być modyfikowane w celu uzyskania różnych rozwiązań o określonych właściwościach.

Przykładami modyfikacji, mających praktyczne znaczenie, są m.in. następujące sformułowania:

1. Jeżeli wartość średniego opóźnienia sumarycznego przepływu o szybkości f_{ij} , wnoszona przez łącze o pojemności C_{ij} , jest określona wyrażeniem:

$$d(f_{ij}) = f_{ij}(C_{ij} - f_{ij}),$$

to wartość opóźnienia $d(f_{ij})$ zależy od różnicy $C_{ij} - f_{ij}$. Zastąpienie w zadaniu pojemności C_{ij} przez pojemność $C'_{ij} = \alpha_{ij}C_{ij}$, gdzie $0 < \alpha_{ij} < 1$ dla każdego $(i, j) \in \Gamma$, pozwala na wyznaczanie sumarycznych przepływów o szybkościach mniejszych od pojemności łącz transmisyjnych. Zmiana wartości współczynników α_{ij} powoduje zmianę wartości średniego opóźnienia.

2. Wprowadzenie do zadania pojemności:

$$C'_{ij} = C_{ij} - \Delta \quad (0 < \Delta < \min_{(i,j) \in \bar{A}} C_{ij}),$$

zamiast pojemności C_{ij} ($C'_{ij} < C_{ij}$), pozwala uzyskać rozwiązanie charakteryzujące się tym, że część pojemności każdego łącza nie jest wykorzystana i może być użyta do obsługi dodatkowych przepływów, np. przepływów o najwyższym priorytecie o sumarycznej szybkości mniejszej od Δ .

3. Jeżeli $g_s(\bar{v}_s)$ jest funkcją rosnącą, określającą priorytet s -tego przepływu o szybkości \bar{v}_s , to funkcja taka może być stosowana do przyznania różnych priorytetów różnym klasom przepływów oraz do uzależnienia priorytetów przepływów od ich szybkości. Zadanie wyznaczania przepływów, polegające na maksymalizacji szybkości s -tego przepływu zgodnie z zasadą, że dowolny wzrost szybkości s -tego przepływu \bar{v}_s powoduje zmniejszenie szybkości s' -tego przepływu $\bar{v}_{s'}$, gdy $g_{s'}(\bar{v}_{s'}) \leq g_s(\bar{v}_s)$, stwarza możliwość wyznaczania maksymalno-minimalnych przepływów z priorytetami.

Efektywne wyznaczanie przepływów maksymalno-minimalnych w sieci, podobnie jak efektywne wyznaczanie optymalnych tras w sieci, wymaga stosowania rozproszonych algorytmów, pozwalających na dynamiczne zmiany szybkości przepływów w następstwie zmiany liczby i szybkości przepływów współdzielących zasoby sieci.

5.9. IMPLEMENTACJA PROCEDUR STEROWANIA PRZEPIYWAMI

5.9.1. MECHANIZM OKNA

Wadą procedur sterowania przepływem, sterujących szybkością przepływu, jest wrażliwość na zmiany liczby obsługiwanych przepływów oraz zmiany ich szybkości (zmiany obciążenia sieci).

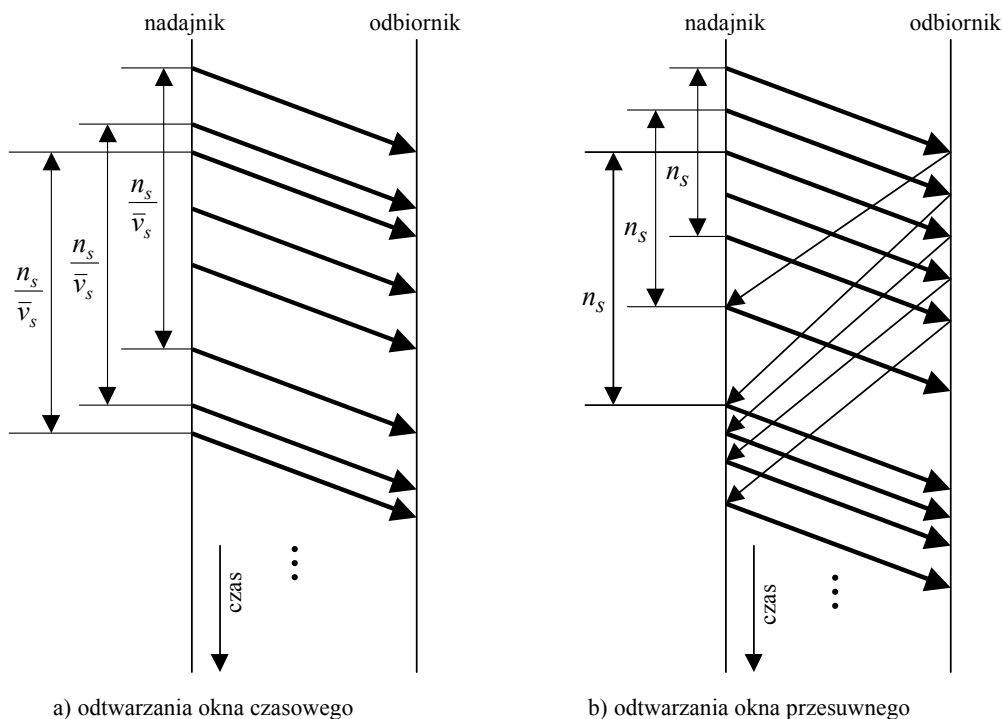
Bezpośrednia implementacja procedury, wymuszającej daną, akceptowaną przez sieć, szybkość przepływu \bar{v}_s , oznacza, że jedna jednostka danych (np. pakiet) jest wprowadzana do sieci raz na $1/\bar{v}_s$ jednostek czasu (np. sekundę). Taka realizacja sterowania przepływami jest równoważna schematowi zwielokrotniania z podziałem czasu (ang. *time division multiplexing*), którego podstawową wadą jest wnoszenie względnie dużych opóźnień w razie nieregularności przepływu powodowanej wybuchowością źródła przepływu.

W takich przypadkach korzystniejszym rozwiązaniem jest wprowadzanie do sieci n_s ($n_s > 1$) jednostek danych w kolejnych przedziałach czasu o długości n_s/\bar{v}_s . Rozwiązanie takie pozwala na wprowadzanie do sieci ograniczonej z góry liczby jednostek danych (co najwyżej n_s) nie w ustalonych szczelinach czasowych, ale w przedziałach czasu o ustalonej długości. Jednostki danych generowane przez wybuchowe źródło ruchu w czasie jego aktywności mogą być wprowadzane do sieci bez opóźnienia, tzn. bez dodatkowego oczekiwania na szczelinę czasową.

Przyznanie s -temu przepływowi n_s miejsc w oknie o ustalonej długości oraz wprowadzenie w źródle licznika niewykorzystanych miejsc count_s powoduje, że w zadanym przedziale czasu jednostki danych s -tego przepływu są wprowadzane do sieci dopóty, dopóki stan licznika jest różny od zera, tj. $\text{count}_s > 0$. Za każdym razem, gdy jednostka danych jest wprowadzana do sieci, stan licznika jest zmniejszany o jeden, a po upływie czasu równego n_s/\bar{v}_s od chwili zmniejszenia stanu licznika, stan licznika jest zwiększany o jeden. W omawianym schemacie sterowania przepływem ograniczenie szybkości wprowadzania jednostek danych s -tego przepływu do sieci jest determinowane długością przedziału czasu, po upływie którego następuje odtwarzanie rozmiaru okna po wcześniejszym jego zmniejszeniu.

Omawiany schemat sterowania przepływem z tzw. oknem czasowym (ang. *time window flow control*) o długości n_s/\bar{v}_s jest zbliżony do klasycznego sterowania z wykorzystaniem mechanizmu tzw. okna przesuwnego z oknem o długości n_s . Różnica polega na sposobie odtwarzania okna; w oknie czasowym odtwarzanie okna następuje po upływie n_s/\bar{v}_s jednostek czasu od chwili wprowadzenia jednostki danych do sieci (odtwarzanie czasowe), w oknie przesuwnym natomiast odtwarzanie następuje po otrzymaniu potwierdzenia wprowadzonej do sieci jednostki danych (odtwarzanie zdarzeniowe). Czasy odtwarzania okna czasowego po transmisji każdej jednostki danych

są jednakowe, podczas gdy czasy odtwarzania okna przesuwne po transmisji każdej jednostki danych są na ogół różne (rysunek 5.9).



Rysunek 5.9. Odtwarzanie okien czasowego i przesuwne

W oknie czasowym parametrami sterowania przepływem są rozmiar okna i czas odtwarzania jego rozmiaru. Wartości chwilowej i średniej szybkości przepływu nie zależą od obciążenia sieci. Parametrem sterowania przepływem korzystającego z mechanizmu okna przesuwne jest rozmiar okna. Czas odtwarzania rozmiaru okna, a tym samym wartości chwilowej i średniej szybkości przepływu, zależy od obciążenia i stanu sieci.

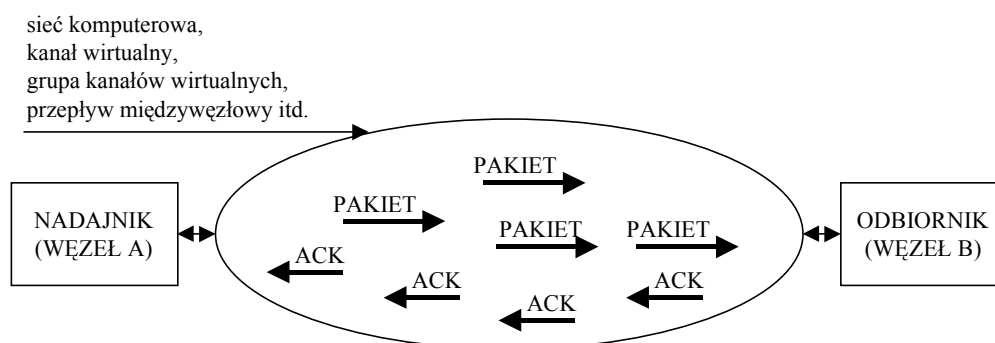
Z porównania mechanizmów okna czasowego i przesuwne wynika, że ten drugi jest bardziej wrażliwy na zmiany obciążenia i przeciążenia sieci. Wzrost obciążenia sieci zwiększa opóźnienia potwierzeń i, spowalniając odtwarzanie rozmiaru okna, spowalnia szybkość wprowadzania kolejnych, nowych jednostek danych do sieci. Mechanizm okna przesuwne jest efektywniejszy od mechanizmu okna czasowego w sieciach, w których zachodzą szybkie zmiany obciążenia sieci.

Realizacja zadania wymuszania przepływu o ustalonej wartości średniej szybkości jest możliwa także za pomocą mechanizmu okna przesuwne. Znana wartość szybkości s -tego przepływu \bar{v}_s pozwala na wyznaczenie rozmiaru okna dla połączenia

typu źródło–ujście przepływu. Z prawa Little’a wynika, że rozmiar okna dla s -tego przepływu w_s jest równy iloczynowi oczekiwanej szybkości przepływu \bar{v}_s i średniego czasu oczekiwania na potwierdzenia jednostek danych przepływu (ang. *round-trip delay*) d_s , tzn. $w_s = \bar{v}_s d_s$. Ograniczeniem tego sposobu wyznaczania rozmiaru okna przesuwanego, a tym samym efektywności sterowania przepływem, jest nieznaną – w chwili wyznaczania rozmiaru okna – wartość średniego czasu oczekiwania na potwierdzenia. Do obliczeń mogą być stosowane różne estymaty czasu oczekiwania, zależnie od charakteru zmienności obciążenia sieci, dostępnych danych itd.

5.9.2. WYZNACZANIE ROZMIARU OKNA

Mechanizm sterowania przepływem pomiędzy źródłem a ujściem jest zgodny z mechanizmem okna, jeżeli istnieje górne ograniczenie liczby jednostek danych, które zostały nadane przez źródło i których potwierdzenia, wygenerowane przez ujście, nie zostały odebrane przez źródło. Liczba jednostek danych wysłanych i niepotwierdzonych jest z góry ograniczona (rysunek 5.10).



Rysunek 5.10. Pakiety i potwierdzenia w sieci

Jednostkami danych w konkretnej implementacji mechanizmu okna mogą być np. wiadomości, pakiety, ramki, komórki, bajty. Jednostka danych jest determinowana procesem, w ramach którego odbywa się sterowanie przepływem. W każdym z przypadków proces, będący przedmiotem sterowania, może być jednoznacznie identyfikowany warstwą określonej architektury sieci. Podczas omawiania mechanizmu okna szczególną uwagę zwrócono na jednostki danych warstw sieciowych różnych architektur (pakietów), co nie ogranicza ogólności rozważań dotyczących mechanizmu okna.

Maksymalna liczba jednostek danych (liczba całkowita dodatnia), które zostały wysłane i których potwierdzenia nie zostały odebrane, jest rozmiarem okna. Górne ograniczenie liczby tych jednostek może dotyczyć różnych procesów komunikacyj-

nych, na różnych poziomach ich granulacji i w różnym zakresie. Wprowadzane jako rozmiar okna ograniczenie może obowiązywać dla pojedynczych, wybranych lub wszystkich procesów komunikacyjnych realizowanych w sieci, w grupie kanałów wirtualnych lub w jednym kanale wirtualnym, na trasie pomiędzy węzłami wejściowym i ujęciowym, na pojedynczym łączy danych w sieci, dla wyróżnionego strumienia danych itd. Wynika to z uniwersalności mechanizmu okna, decydującego o szerokim zakresie zastosowań w sieciach komunikacyjnych.

Ujęcie potwierdza źródłu odebranie jednostek danych, wysyłając specjalną wiadomość (potwierdzenie, pozwolenie, wiadomość alokacji itp.). Potwierdzenie może być przesyłane jako specjalny pakiet sterujący lub stanowić integralną część jednostek danych użytkowych w postaci tzw. niejawnego potwierdzenia (ang. *piggybacked*). Otrzymanie potwierdzenia przez źródło daje mu możliwość wysłania jednej, kolejnej jednostki danych do ujęcia, dlatego też przesłanie potwierdzenia do nadajnika i odebranie go jest tożsame z odebraniem przepustki (pozwolenia, paszportu), która może być przekazana kolejnej jednostce danych oczekującej na nadanie. Posiadanie takiego pozwolenia jest warunkiem koniecznym dostępu do zasobów komunikacyjnych sieci. Liczba przepustek, użytkowanych do komunikacji pomiędzy źródłem i ujęciem, nie może być większa od rozmiaru okna przyznanego tym dwóm komunikującym się węzłom.

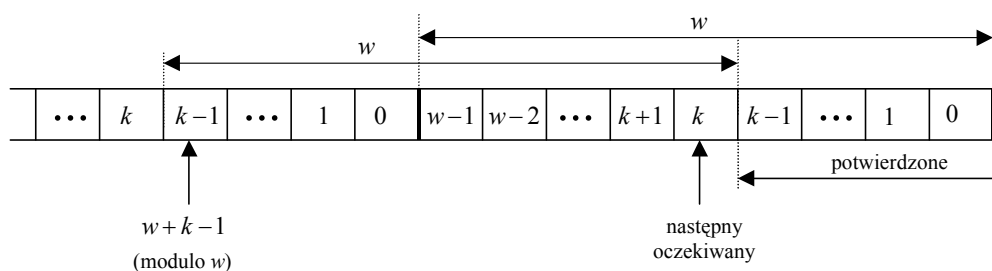
Ogólna zasada mechanizmu okna sprowadza się do tego, że szybkość wypływu jednostek danych ze źródła jest determinowana szybkością napływu do źródła potwierdzeń z ujęcia. Jeżeli w ujęciu lub w dowolnym elemencie trasy sieci pomiędzy źródłem i ujęciem wystąpi przeciążenie, to jego skutkiem jest zwolnienie przekazywania potwierdzeń i tym samym tłumienie szybkości nadawania przez źródło. Dodatkowo spowolnienie wysyłania potwierdzeń przez ujęcie, powodowane innymi niż przeciążenie przyczynami, również prowadzi do zmniejszenia szybkości nadawania jednostek danych przez źródło. Zasada sterowania szybkością generowania nowego ruchu przez nadajnik dzięki udzielaniu kredytu przez odbiornik, będąca podstawą mechanizmu okna, jest jedną ze skuteczniejszych i szeroko rozpowszechnionych zasad sterowania przepływem nie tylko w sieciach komputerowych z komutacją jednostek danych (wiadomości, pakietów, ramek itd.).

5.9.2.1. STEROWANIE PRZEPLÝWEM W PROCESIE KOMUNIKACYJNYM

W ogólnej, najczęściej stosowanej, wersji mechanizmu okna jego rozmiar (wielkość okna), oznaczony przez $w \times a$, gdzie w i a są dodatnimi liczbami całkowitymi, jest wyrażany liczbą w paczek jednostek danych o długości a . Za każdym razem, gdy paczka o długości a zostanie odebrana przez ujęcie, zwrotnie, tzn. do źródła, wysyłane jest potwierdzenie, będące jednocześnie pozwoleniem na wysłanie przez źródło kolejnej paczki jednostek danych. W skrajnych wariantach ogólnego mechanizmu okna możliwe są różne jego modyfikacje. Należy do nich schemat krokowy (schemat z rezerwacją), implementowany np. w architekturze SNA, zgodnie z którym pozwole-

nie na nadawanie kolejnej paczki jest wysyłane przed otrzymaniem jakiegokolwiek paczki – praktycznie jest to równoważne rozwiązaniu, w którym powiadomienie o możliwości nadawania przez źródło jest poprzedzone rezerwacją zasobów w ujściu. Dla ogólności rozważań nie jest istotny rozmiar paczki – można przyjąć, że długość paczki jest jednostkowa $a = 1$.

Na ogół do zarządzania pakietami i potwierdzeniami stosowany jest ten sam schemat numerowania. Numery potwierdzeń są zwykle numerami pakietów wcześniej transmitowanych. Jedną z wersji mechanizmu okna (implementowaną np. w protokołach łącza danych, takich jak HDLC i jego różne odmiany) jest procedura kroczącego okna (ang. *sliding window*), w której do koordynacji nadajnika i odbiornika oraz przekazywania potwierdzeń stosowany jest numer „następnego oczekiwanego” (ang. *next expected*) pakietu. Numer następnej oczekiwanej jednostki danych może być wykorzystywany w dwóch wersjach: do potwierdzania pojedynczych jednostek danych lub grup jednostek danych. W tym ostatnim przypadku przekazanie numeru następnej oczekiwanej k -tej jednostki danych jest tożsame z potwierdzeniem odebrania wszystkich pakietów o numerach mniejszych od k – źródło otrzymuje możliwość (pozwolenie) wysłania kolejnych jednostek danych o numerach do $w + k - 1$ (modulo w) włącznie (rysunek 5.11).



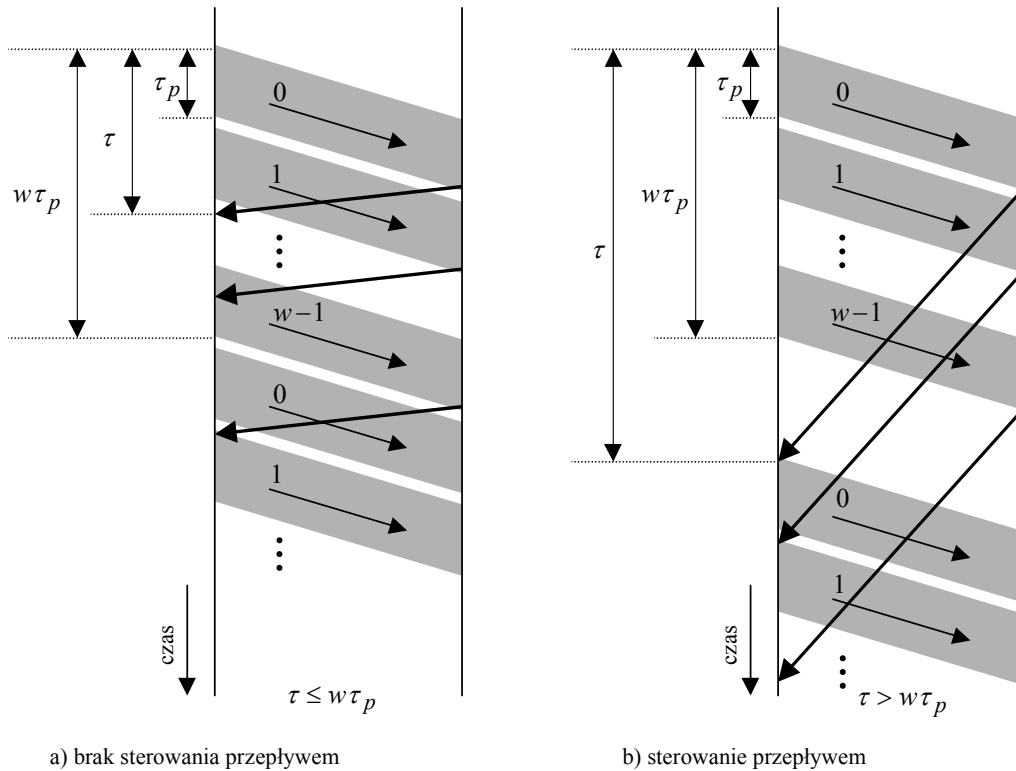
Rysunek 5.11. Schemat potwierdzania jednostek danych w mechanizmie okna

Dla uproszczenia rozważań, nie tracąc ich ogólności, można przyjąć, że sposób potwierdzania jednostek danych jest pomijalny; ważne jest to, że w danej chwili liczba wszystkich jednostek danych k , nadanych przez źródło i niepotwierdzonych przez ujście, nie może być większa od rozmiaru okna w , czyli $k \leq w$.

Przepływ jednostek danych pomiędzy źródłem i ujściem może być charakteryzowany m.in.:

- wartością średnią czasu opóźnienia potwierdzenia – czasu obsługi jednostki danych, tj. wartością czasu, po upływie którego źródło otrzymuje potwierdzenie nadanego pakietu z ujścia (ang. *round-trip delay*) τ ,

- wartością czasu nadawania w jednostek danych (liczby jednostek danych równej rozmiarowi okna) $w\tau_p$, gdzie τ_p jest czasem nadawania (transmisji) jednej jednostki danych przez źródło (czas nadawania jednostek danych jest stały dla wszystkich jednostek danych).



Rysunek 5.12. Warunki braku aktywności (a) i aktywności (b) procedury sterowania

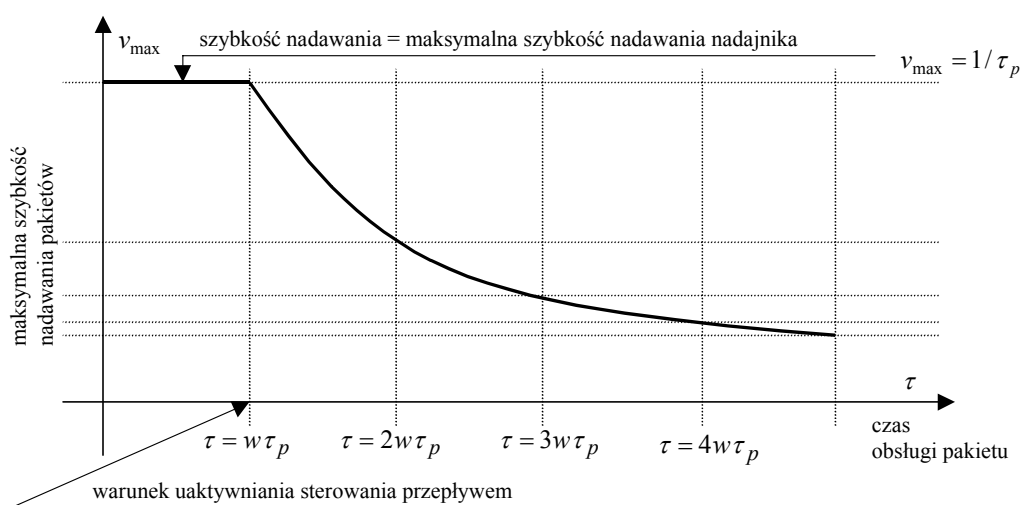
Po przyjęciu, że szybkości nadawania jednostek danych przez źródło oraz generowania potwierżeń jednostek danych przez ujście są stałe, możliwe są dwa przypadki:

1. Jeżeli średni czas potwierdzenia jednostki danych τ jest krótszy od czasu nadawania wszystkich jednostek danych w oknie $w\tau_p$ (czas nadawania całego okna z właściwą dla źródła szybkością nadawania) lub równy temu czasowi, tzn. gdy: $\tau \leq w\tau_p$, to szybkość nadawania jednostek danych v przez źródło nie jest spowalniana przez ujście. Sterowanie przepływem nie jest aktywne – jest zbędne (rysunek 5.12), wtedy szybkość nadawania jednostek danych przez źródło: $v = 1/\tau_p$.

2. W przeciwnym razie, tj. gdy średni czas potwierdzania jednostki danych d jest dłuższy od czasu nadawania całego okna $w\tau_p$: $\tau > w\tau_p$, wówczas szybkość nadawania jednostek danych v jest ograniczona przez szybkość generowania potwierżeń przez ujście. W tym przypadku sterowanie przepływem jest aktywne – jest konieczne, a efektywna szybkość nadawania jednostek danych przez źródło (pod warunkiem, że źródło ma zawsze jednostki danych oczekujące na nadawanie): $v = w/\tau$.

Połączenie omawianych przypadków oznacza, że maksymalna efektywna szybkość nadawania jednostek danych v_{\max} przez źródło, dla danej wartości czasu potwierdzenia jednostek danych d (rysunek 5.13):

$$v_{\max} = \begin{cases} \frac{1}{\tau_p} & \text{gdy } \tau \leq w\tau_p \\ \frac{w}{\tau} & \text{gdy } \tau > w\tau_p \end{cases}$$



Rysunek 5.13. Maksymalna szybkość nadawania jednostek danych w funkcji czasu obsługi jednostki danych

Zależność ta precyzuje podstawowe właściwości i zalety mechanizmu okna, jako sposobu sterowania przepływem:

- szybkość nadawania jednostek danych przez źródło jest stała i równa odwrotności czasu transmisji jednostki danych, jeżeli czas obsługi jednostki danych τ jest krótszy od czasu transmisji wszystkich jednostek danych w oknie $w\tau_p$, czyli $v_{\max} = 1/\tau_p$,
- szybkość nadawania jednostek danych przez źródło skraca się wraz z wydłużeniem czasu obsługi jednostki danych τ , powodowanego przeciążeniami i towarzyszącymi im opóźnieniami $v_{\max} = w/\tau$,
- czas reakcji metody okna na przeciążenie powstające w ujściu jest krótszy od czasu trwania transmisji co najwyżej w jednostek danych (liczby jednostek danych równej rozmiarowi okna) $w\tau_p$.

Liczba jednostek danych nadanych i niepotwierdzonych w sieci $l(\tau)$ w chwili nadawania jednej z jednostek danych okna, tzn. suma liczb jednostek danych w drodze

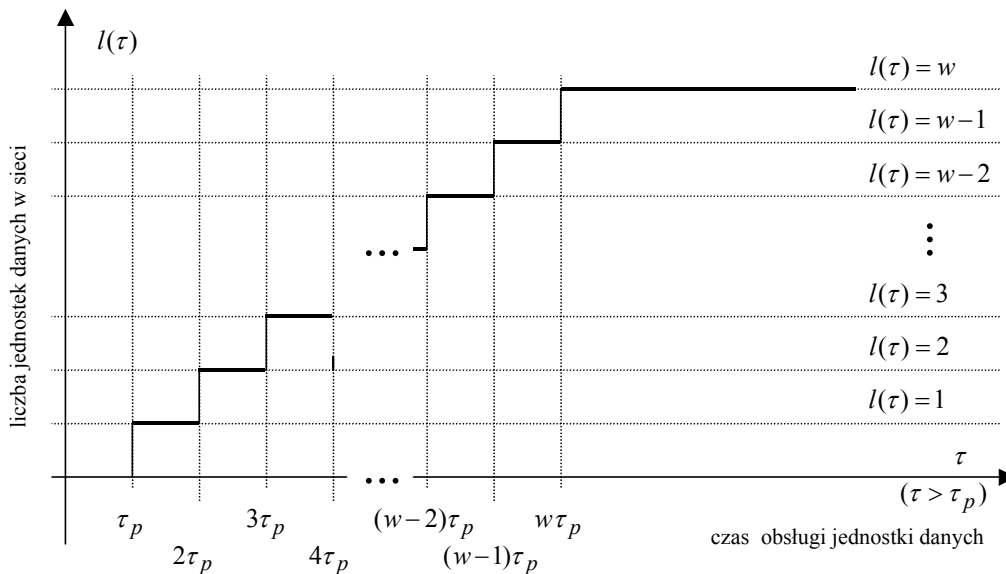
z nadajnika do odbiornika oraz potwierżeń w drodze z odbiornika do nadajnika, w funkcji czasu obsługi jednostki danych d , jest określona zależnością:

$$l(\tau) = \begin{cases} \left\lceil \frac{\tau - \tau_p}{\tau_p} \right\rceil & \text{gdy } \tau_p < \tau \leq w\tau_p, \\ w & \text{gdy } \tau > w\tau_p \end{cases},$$

w której $\lceil (\tau - \tau_p)/\tau_p \rceil$ jest liczbą całkowitą, nie mniejszą niż $(\tau - \tau_p)/\tau_p$ i spełniającą warunek:

$$1 \leq \left\lceil \frac{\tau - \tau_p}{\tau_p} \right\rceil \leq w.$$

Wartość różnicy $\tau - \tau_p$ jest czasem, który upływa od chwili zakończenia nadawania jednostki danych przez źródło do chwili nadejścia potwierzenia tej jednostki danych do źródła.



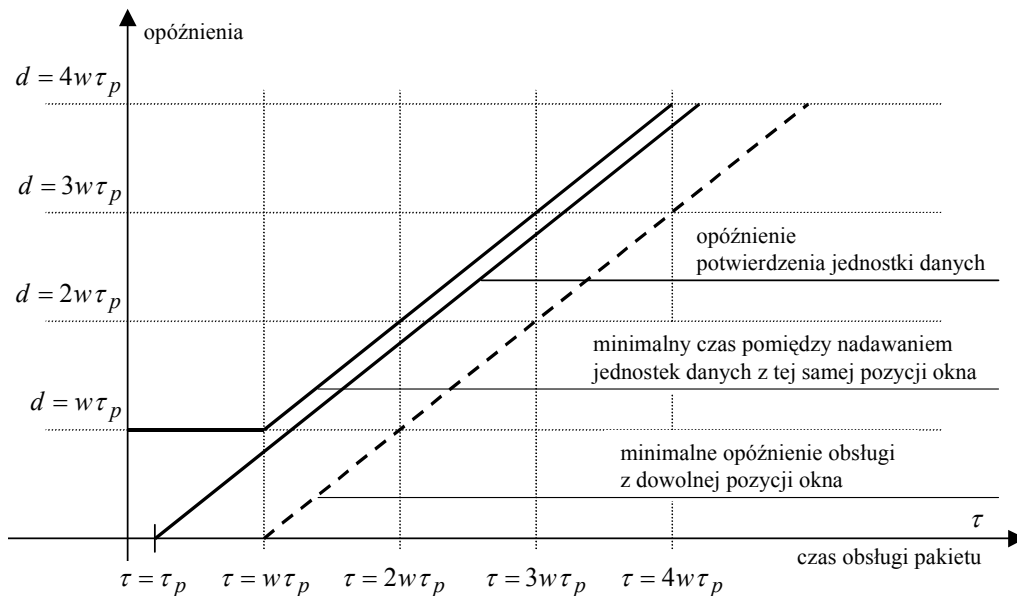
Rysunek 5.14. Maksymalna liczba jednostek danych niepotwierdzonych w sieci w funkcji czasu obsługi jednostki danych

Ponieważ zawsze spełniony jest warunek $\tau > \tau_p$ (czas obsługi jednostki danych jest dłuższy od czasu nadawania jednostki danych), w razie stosowania mechanizmu okna liczba jednostek danych nadanych i niepotwierdzonych w sieci jest więc równa jeden

tylko w przypadku (najkorzystniejszym), gdy potwierdzenie nadanej wcześniej jednostki danych nadejdzie w czasie nadawania następnej, kolejnej jednostki danych, tj. gdy $\tau_p < \tau \leq 2\tau_p$.

Jeżeli potwierdzenie jednej z wcześniej nadanych jednostek danych nadejdzie podczas nadawania ostatniej jednostki danych okna $((w-1)\tau_p < \tau \leq w\tau_p)$, to liczba jednostek danych nadanych i niepotwierdzonych w sieci jest nie większa niż $(w-1)$. Jednocześnie liczba pakietów nadanych i niepotwierdzonych w sieci $l(\tau)$ nie może być większa od rozmiaru okna w . Liczba ta jest równa rozmiarowi okna wtedy i tylko wtedy, gdy czas obsługi jednostki danych w sieci τ jest dłuższy od czasu nadawania w jednostek danych w nadajniku, tj. $\tau > w\tau_p$ (rysunek 5.14).

Z zależności pomiędzy czasem nadawania τ_p i czasem obsługi jednostek danych τ , dla danego rozmiaru okna w , wynikają wartości opóźnień wnoszonych przez mechanizm okna (rysunek 5.15).



Rysunek 5.15. Opóźnienia jednostek danych wnoszone przez mechanizm okna

Liczba jednostek danych w sieci (liczba jednostek danych nadanych i oczekujących na potwierdzenia) $l(\tau)$ w chwili nadawania jednej z jednostek danych okna, tzn. suma liczb jednostek danych w drodze z nadajnika do odbiornika oraz potwierdzeń w drodze z odbiornika do nadajnika, w funkcji czasu t , wpływającego od chwili rozpoczęcia obsługi jednostek danych okna, spełnia warunki:

$$0 \leq l(t) \leq w,$$

$$l(t) = \begin{cases} 0 & \text{gdy } t \leq \tau_p \\ \left\lceil \frac{\tau - \tau_p}{\tau_p} \right\rceil & \text{gdy } t = \tau \text{ oraz } \tau_p < \tau \leq w\tau_p \\ w & \text{gdy } t = \tau \text{ oraz } \tau > w\tau_p \end{cases}$$

Iloraz sumy liczby jednostek danych w sieci $l(t)$ i jednej jednostki danych nadawanej w chwili t oraz rozmiaru okna w w funkcji czasu dla danych: rozmiaru okna w , czasu nadawania jednostki danych τ_p i czasu obsługi jednostki danych τ , jest wartością współczynnika wykorzystania okna:

$$\alpha(t) = \begin{cases} \frac{l(t)+1}{w} & \text{gdy } l(t) < w \\ 1 & \text{gdy } l(t) = w \end{cases}$$

Podczas nadawania jednostek danych znajdujących się w oknie co najmniej jedno miejsce w oknie jest zaangażowane w zadanie sterowania przepływem. Wartość współczynnika wykorzystania okna spełnia warunek: $1/w \leq \alpha(t) \leq 1$.

Wartość współczynnika wykorzystania okna $\alpha(t)$ można interpretować jako wskaźnik chwilowego nadmiaru rozmiaru okna w w stosunku do potrzeb wynikających z zależności pomiędzy czasem trwania nadawania jednostki danych τ_p w nadajniku i czasem obsługi jednostki danych τ w sieci.

Względnie krótki czas reakcji mechanizmu okna na przeciążenia (mechanizm o cechach mechanizmu reaktywnego) wraz ze względnie niewielkim narzutem, wymaganym przez ten mechanizm (oszczędzający przepustowość), to podstawowe zalety mechanizmu okna w porównaniu z innymi mechanizmami sterowania przepływem.

5.9.2.2. OGRANICZENIA STEROWANIA PRZEPLYWEM Z WYKORZYSTANIEM MECHANIZMU OKNA

Celem różnych procedur sterowania przepływem implementowanych w sieciach jest, oprócz sprawiedliwego rozdziału zasobów sieci pomiędzy przepływy lub klasy przepływów, utrzymanie przyjętych wartości średniego opóźnienia jednostek danych i przepustowości w sieci lub w wybranych jej fragmentach. Właściwości mechanizmu okna przesuwne powodują, że tylko w określonych warunkach mechanizm ten jest skutecznym narzędziem realizacji celów sterowania przepływami.

Opóźnienia i przepustowość trasy

Efektywność mechanizmów sterowania przepływem korzystających z metody okna, mierzona jednocześnie średnim opóźnieniem jednostek danych i przepustowością sieci, silnie zależy od rozmiaru okna i obciążenia sieci.

Wartości średnich opóźnień jednostek danych i przepustowość sieci są wzajemnie zależne: małym wartościom średnich opóźnień odpowiada mała przepustowość sieci, a wraz ze wzrostem przepustowości zwiększają się wartości średnich opóźnień wnoszonych przez sieć.

Jeżeli przedmiotem sterowania przepływem w sieci (z wykorzystaniem mechanizmu okna przesuwne) jest wiele jednocześnie obsługiwanych przepływów, to średnia liczba jednostek danych w sieci zależy od: liczby wszystkich przepływów w sieci m , rozmiarów okien dla każdego z przepływów w_1, w_2, \dots, w_m oraz wartości średnich współczynników wykorzystania poszczególnych okien $\alpha_1, \alpha_2, \dots, \alpha_m$.

Wnoszone przez sieć średnie opóźnienie jednostki danych \bar{d} jest równe (twierdzenie Little'a) wartości ilorazu:

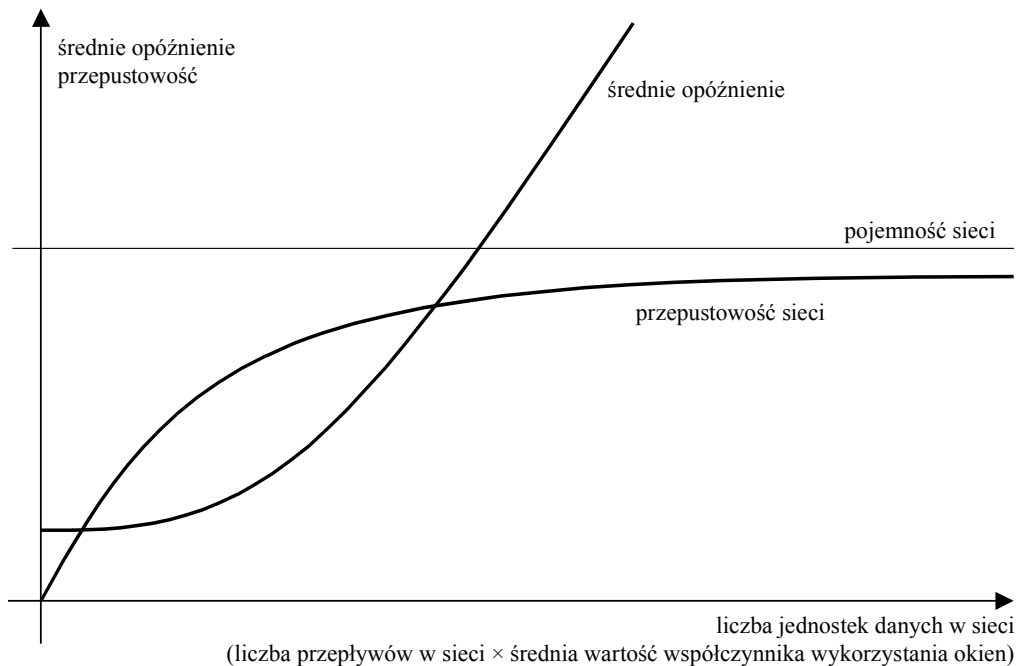
➤ średniej całkowitej liczby jednostek danych w sieci \bar{N} , tzn. sumy średnich liczb jednostek danych należących do poszczególnych przepływów $\alpha_i w_i$ ($i = 1, 2, \dots, m$) i obsługiwanych w sieci,

➤ przepustowości sieci równej średniej sumarycznej szybkości napływu jednostek danych do sieci \bar{v} , tzn. sumy akceptowanych przez sieć szybkości napływu jednostek danych w poszczególnych przepływach \bar{v}_i ($i = 1, 2, \dots, m$),

$$\bar{d} = \frac{\bar{N}}{\bar{v}} = \frac{\sum_{i=1}^m \alpha_i w_i}{\sum_{i=1}^m \bar{v}_i}.$$

Przepustowość sieci jest ograniczona z góry pojemnością sieci, której wartość jest stała dla danych: struktury topologicznej sieci, lokalizacji źródeł i ujść przepływów, metod wyznaczania tras itd. Jeżeli liczba obsługiwanych w danej sieci przepływów zwiększa się, to wzrost przepustowości w tej sieci jest ograniczony z góry stałą wartością pojemności danej sieci. Wzrost liczby przepływów, tożsamy ze zwiększeniem sumy rozmiarów okien dla wszystkich obsługiwanych przepływów, przy ograniczonej z góry przepustowości sieci i stałej wartości współczynników wykorzystania okien, oznacza, że zwiększa się wartość średniego opóźnienia jednostek danych w sieci. Średnie opóźnienie jednostek danych rośnie (w przybliżeniu) liniowo wraz ze zwiększeniem liczby jednostek danych w sieci, przepustowość sieci natomiast jest (w przybliżeniu) stała. Jeżeli liczba jednocześnie obsługiwanych przepływów jest bardzo duża, to utrzymywanie zadanych wartości średniego opóźnienia jednostek danych przez mechanizm okna jest praktycznie niemożliwe (rysunek 5.16).

W razie wzrostu średnich opóźnień jednostek danych, mechanizm okna powoduje dławienie akceptowanych przez sieć szybkości przepływów i zmniejszanie wartości współczynników wykorzystania okien. Naturalnym rozwiązaniem w takim przypadku jest zmniejszanie rozmiarów okien przypisanych poszczególnym przepływowi w sieci. Zmniejszanie rozmiaru okna pozwala na utrzymywanie zadanych wartości średniego opóźnienia jednostek danych wtedy, gdy rośnie obciążenie sieci.



Rysunek 5.16. Średnie opóźnienie jednostek danych i przepustowość sieci

Rozmiar okna w połączeniach źródło–ujście nie może być dowolnie mały. Zbyt mały może bowiem powodować przerwy w nadawaniu jednostek danych, a tym samym występowanie zbędnego dławienia oferowanych szybkości przepływów. Niepożądane przerwy w nadawaniu jednostek danych występują wtedy, gdy czas obsługi jednej jednostki danych jest dłuższy niż iloczyn rozmiaru okna i czasu nadawania jednej jednostki danych.

Jeżeli przyjmie się, że mechanizm okna jest implementowany do sterowania przepływem na trasie źródło–ujście, składającej się z n łączy, to czas transmisji jednostki danych na trasie jest równy sumie czasów transmisji w łączy. Gdy czas transmisji jednostki danych w każdym z łączy jest taki sam i równy τ_i , wtedy czas transmisji jednostki danych na trasie źródło–ujście jest nie krótszy niż czas $n\tau_i$. Jeżeli przyjmie się, dla uproszczenia rozważań, że czas transmisji jednostki danych w łączy τ_i jest równy czasowi nadawania jednostki danych τ_p , a czas transmisji potwierdzenia jest pomijalnie mały w porównaniu z czasem transmisji jednostki danych, to warunkiem ciągłości transmisji jednostek danych (transmisji bez przerw) jest rozmiar okna nie mniejszy niż liczba łączy na trasie od źródła do ujścia, tzn. $w > n$.

Jeżeli dodatkowo przyjmie się, że potwierdzenia odbioru jednostek danych są przekazywane jako części składowe nagłówek innych jednostek danych przesyłanych w odwrotnym kierunku (z ujścia do źródła) po tych samych trasach, to czas transmisji potwierdzenia jest porównywalny z czasem transmisji jednostki danych ze

źródła do ujścia. W takim przypadku warunkiem transmisji jednostek danych bez przerw, które mogą być powodowane oczekiwaniem jednostek danych w źródle na miejsce w oknie, jest rozmiar okna nie mniejszy od dwukrotnej długości trasy, wyrażonej liczbą łączy, tzn. $w > 2n$.

Minimalny rozmiar okna, gwarantujący transmisję jednostek danych bez przerw przy małym obciążeniu sieci, zależy od długości trasy (wyrażonej liczbą łączy na trasie) oraz względnych wartości czasów transmisji jednostek danych i potwierdzeń. Praktycznie stosowana zasada wyznaczania rozmiarów okna to przyjmowanie jego rozmiaru jako wielokrotności długości trasy wyrażonej w liczbie łączy $n < w \leq 3n$ wtedy, gdy czas (opóźnienie) transmisji jednostki danych jest znacznie dłuższy od czasu (opóźnienia) propagacji i czasu (opóźnienia) przetwarzania. Jeżeli warunki te nie są spełnione i np. czas propagacji jest porównywalny z czasem transmisji, to warunkiem zachowania ciągłości przepływu jest zwiększenie rozmiaru okna.

Z podanych ograniczeń mechanizmu okna wynika, że warunkiem sterowania przepływem, gwarantującego uzyskanie kompromisu pomiędzy średnim opóźnieniem jednostek danych i przepustowością sieci, jest możliwość zmiany rozmiarów okna w zależności od obciążenia sieci. W razie małego obciążenia sieci większe rozmiary okien są warunkiem ciągłości przepływów, natomiast przy dużym obciążeniu sieci mniejsze rozmiary okien ograniczają wartości średnich opóźnień jednostek danych.

Sprawiedliwy rozdział zasobów

Efektywność sterowania przepływem z wykorzystaniem mechanizmu okna zależy od rozmiaru okna i obciążenia sieci. Jeżeli:

- obciążenie sieci jest małe, to:
 - zbyt duży rozmiar okna gwarantuje ciągłość transmisji jednostek danych, ale może nie zapobiegać powstawaniu przeciążeń,
 - zbyt mały rozmiar okna może powodować przerwy w transmisji jednostek danych, tzn. zbędne dławienie oferowanej szybkości przepływu,
- obciążenie sieci jest duże, to:
 - zbyt duży rozmiar okna powoduje zwiększenie wartości średnich opóźnień jednostek danych i ogranicza zdolność do zapobiegania przeciążeniom,
 - zbyt mały rozmiar okna gwarantuje utrzymanie średniego opóźnienia jednostek danych kosztem ograniczania szybkości przepływu.

Rozmiary okien, przypisywanych poszczególnym przepływom na trasach źródło–ujście, można utożsamiać z ilością zasobów rozdzielanych pomiędzy przepływy współdzielące zasoby sieci. Przyjęcie zasady, zgodnie z którą rozmiar okna przypisywanego przepływowi jest proporcjonalny do liczby łączy składających się na trasę źródło–ujście, jest kompromisem pomiędzy wartością średniego opóźnienia wnoszonego przez mechanizm okna, przepustowością sieci i zdolnością do przeciwdziałania powstawaniu przeciążeń.

Rozmiary okien dla przepływów współdzielących zasoby sieci, wyznaczone zgodnie z podaną zasadą, nie gwarantują sprawiedliwego rozdziału zasobów sieci.

Omawiana zasada wyznaczania rozmiarów okien powoduje, że w silnie obciążonych łączach sieci przepływy korzystające z dłuższych tras mają więcej jednostek danych niż przepływy korzystające z krótszych tras. Więcej jednostek danych przepływu na dłuższej trasie w silnie obciążonym łączu oznacza większą przepustowość przepływu na dłuższej trasie. Silnie obciążone łącze można scharakteryzować w następujący sposób: na jego wejściu następuje akumulacja jednostek danych proporcjonalnie do rozmiarów okien. Jeżeli napływające jednostki danych poszczególnych przepływów są obsługiwane przez łącze w kolejności ich napływu, to względny udział każdego z przepływów w sumarycznym przepływie silnie obciążonego łącza jest proporcjonalny do rozmiarów okien tych przepływów. Stopień monopolizacji pojemności pamięci buforowej węzła, w przypadku przeciążenia łączy wychodzących z tego węzła, zależy od względnych rozmiarów okien wszystkich przepływów źródło–ujście, współdzielących pojemność buforową węzła. Jeżeli przyjmie się na przykład, że:

- ruch generowany przez n użytkowników jest obsługiwany przez zbiór szeregowo połączonych łączy o jednakowej pojemności $C_{ij} = C$ ($i, j = 1, 2, \dots, n, i \neq j$),
- ruch generowany przez $(n - 1)$ użytkowników jest obsługiwany tylko przez pojedyncze łącze,
- ruch generowany przez jednego, n -tego użytkownika jest obsługiwany przez $(n - 1)$ szeregowo połączonych łączy,

to rozmiary okien, proporcjonalne do długości tras, są odpowiednio równe: $w_1 = w_2 = \dots = w_{n-1} = 1$ i $w_n = n - 1$. Jeśli założy się, że wszystkie łącza są silnie obciążone, to udział akceptowanych szybkości poszczególnych przepływów \bar{v}_i ($i = 1, 2, \dots, n$) w poszczególnych łączach jest proporcjonalny do rozmiarów okien przepływów współdzielących te łącza: $\bar{v}_1 = \bar{v}_2 = \dots = \bar{v}_{n-1} = C/n$ oraz $\bar{v}_n = C(n-1)/n$ (rysunek 5.17).

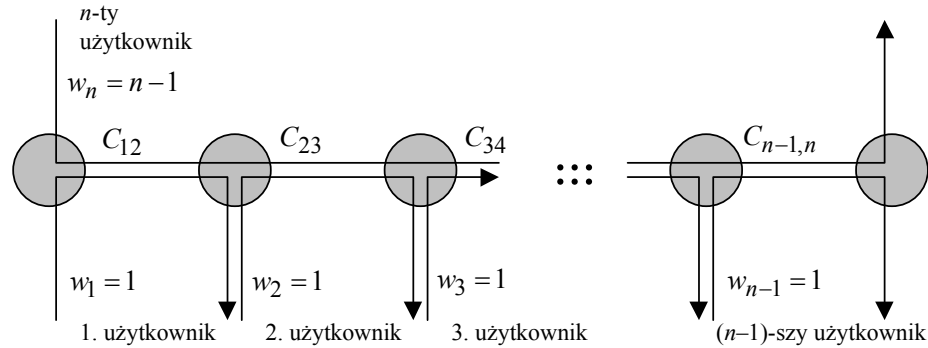
W omawianym przykładzie wraz ze wzrostem liczby n następuje majoryzowanie przepustowości systemu przez n -ty przepływ.

Maksymalna przepustowość omawianego systemu jest sumą akceptowanych szybkości przepływów:

$$\bar{v}_1 + \bar{v}_2 + \dots + \bar{v}_{n-1} + \bar{v}_n = \frac{(n-1)C}{n} + \frac{(n-1)C}{n} = \frac{2(n-1)C}{n},$$

która jest mniejsza od pojemności systemu równej $(n - 1)C$ dla $n > 2$.

Jednym ze sposobów podniesienia zdolności sprawiedliwego rozdziału zasobów przez mechanizm okna jest jego uzupełnienie o algorytm szeregowania jednostek danych, oczekujących w kolejce wejściowej na obsługę przez łącze trasy. Implementacja algorytmu szeregowania na wejściu każdego, silnie obciążonego, łącza trasy pozwala na zmianę względnego udziału jednostek danych poszczególnych przepływów w sumarycznym przepływie łącza.



$$C_{12} = C_{21} = \dots = C_{n-2,n-1} = C_{n-1,n} = C \quad \text{przepływy: } \bar{v}_1 = \bar{v}_2 = \dots = \bar{v}_{n-2} = \bar{v}_{n-1} = \frac{1}{n} C$$

$$\bar{v}_n = \frac{n-1}{n} C$$

Rysunek 5.17. Rozdział zasobów przez procedury sterowania przepływem z mechanizmem okna

Zadaniem algorytmu szeregowania jednostek danych jest wymuszenie kolejności obsługi jednostek danych oczekujących w kolejce wejściowej, różnej od kolejności ich napływu (rysunek 5.18). W przypadku dużego obciążenia łącza wyjściowego liczba jednostek danych poszczególnych przepływów oczekujących w pamięci buforowej węzła na obsługę $b_i(t)$ oraz $b_j(t)$ jest proporcjonalna do rozmiarów okien poszczególnych przepływów, tzn. w_i oraz w_j .

Jeżeli jednostki danych oczekujące w pamięci buforowej są obsługiwane w kolejności napływu, tzn. gdy w węzle brak jest procedury wymuszania przepływu, to całkowita pojemność łącza C jest współdzielona przez przepływy i -ty oraz j -ty, proporcjonalnie do zajętości pamięci buforowej węzła $b_i(t)$ oraz $b_j(t)$ i – tym samym – do rozmiarów okien w_i oraz w_j :

$$\frac{C_i}{C_j} \text{ i } C = C_i + C_j.$$

W przypadku ilustrowanym na rysunku 18a:

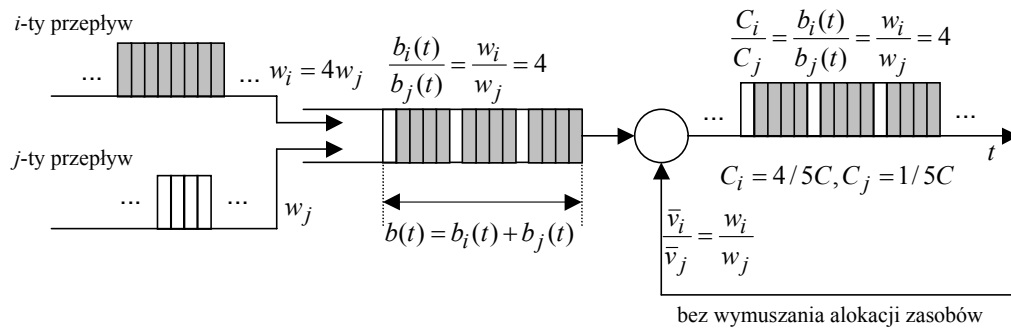
$$w_i = 4w_j \text{ oraz } \frac{\bar{v}_i}{\bar{v}_j} = \frac{w_i}{w_j} = 4,$$

z czego wynika, że:

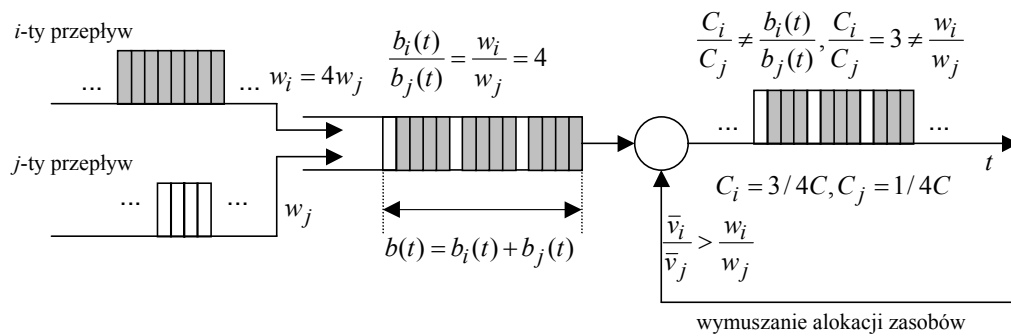
$$\frac{C_i}{C_j} = \frac{b_i(t)}{b_j(t)} = \frac{w_i}{w_j} = \frac{\bar{v}_i}{\bar{v}_j} = 4$$

oraz

$$C_i = \frac{\bar{v}_i C}{\bar{v}_i + \bar{v}_j} = \frac{4}{5} C \text{ i } C_j = \frac{\bar{v}_j C}{\bar{v}_i + \bar{v}_j} = \frac{1}{5} C.$$



a) kolejność obsługi zgodna z kolejnością napływu jednostek danych



b) kolejność obsługi inna od kolejności napływu jednostek danych – szeregowanie jednostek danych

Rysunek 5.18. Realizacja sprawiedliwego rozdziału zasobów przez implementację algorytmu szeregowania jednostek danych

Zmiana rozdziału zasobów, tzn. całkowitej pojemności łącza C , jest możliwa przez szeregowanie jednostek danych, tj. wymuszenie kolejności obsługi innej od kolejności napływu jednostek danych do pamięci buforowej węzła.

W przypadku ilustrowanym na rysunku 5.18b wprowadzenie w węzle algorytmu szeregowania, polegającego na naprzemiennym wymuszaniu obsługi jednostek danych i -tego oraz j -tego przepływów (ang. *round robin*) z szybkościami \bar{v}_i i \bar{v}_j takimi, że $\bar{v}_i/\bar{v}_j > w_i/w_j$, powoduje zmianę rozdziału pojemności łącza w porównaniu z obsługą bez wymuszania alokacji zasobów. Warunek $\bar{v}_i/\bar{v}_j > w_i/w_j$ oznacza, że względne zwiększenie szybkości obsługi j -tego przepływu odbywa się kosztem względnego zmniejszenia szybkości obsługi i -tego przepływu, a więc:

$$w_i = 4w_j \text{ oraz } \frac{\bar{v}_i}{\bar{v}_j} > \frac{w_i}{w_j},$$

z czego wynika, że:

$$\frac{b_i(t)}{b_j(t)} = \frac{w_i}{w_j}, \quad \frac{C_i}{C_j} = \frac{\bar{v}_i}{\bar{v}_j} = 3 < \frac{b_i(t)}{b_j(t)} = \frac{w_i}{w_j} = 4,$$

$$\text{oraz } C_i = \frac{3}{4C} \quad \text{i} \quad C_i = \frac{1}{4C}.$$

5.9.2.3. STEROWANIE PRZEPLYWEM Z WYKORZYSTANIEM HIERARCHII MECHANIZMÓW OKNA

Jednym ze sposobów poprawy efektywności sterowania przepływem, pomiędzy źródłem a ujściem przepływu, jest implementacja hierarchii mechanizmów okna.

Rozwiązanie dwupoziomowe polega na tym, że na poziomie nadrzędnym, tj. na poziomie łącza wirtualnego, przedmiotem sterowania jest przepływ pomiędzy źródłem i ujściem (ang. *virtual circuit flow control*), a na poziomie podrzędnym ten sam przepływ jest przedmiotem sterowania pomiędzy kolejnymi parami sąsiednich węzłów na trasie od źródła do ujścia (ang. *node-by-node flow control*).

Sterowanie przepływem pomiędzy parą sąsiednich węzłów trasy (nadajnikiem i odbiornikiem) z wykorzystaniem mechanizmu okna polega na tym, że zapobieganie przepełnieniom pamięci buforowej odbiornika, dedykowanej danemu przepływowi, odbywa się przez spowalnianie lub wstrzymywanie wysyłania potwierdzeń do nadajnika. Powoduje to spowalnianie nadawania lub wstrzymywanie nadawania jednostek danych przez nadajnik.

Efektywność hierarchicznego, dwupoziomowego sterowania przepływem zależy od długości trasy przepływu, wyrażonej liczbą węzłów na trasie n , rozmiarów okien dla przepływu w połączeniu wirtualnym w , rozmiarów okien dla przepływów pomiędzy kolejnymi parami węzłów w_{ij} ($i, j = 1, 2, \dots, n; i \neq j, j = i + 1$) oraz pojemności pamięci buforowej dla danego przepływu w kolejnych węzłach b_i ($i = 2, 3, \dots, n$) na trasie ze źródła do ujścia. Sumaryczna pojemność pamięci buforowych b we wszystkich węzłach odbiorczych trasy, rezerwowanych dla danego przepływu źródło–ujście, jest równa sumie pojemności rezerwowanych w poszczególnych węzłach odbiorczych i rozmiarowi okna połączenia wirtualnego (rysunek 5.19):

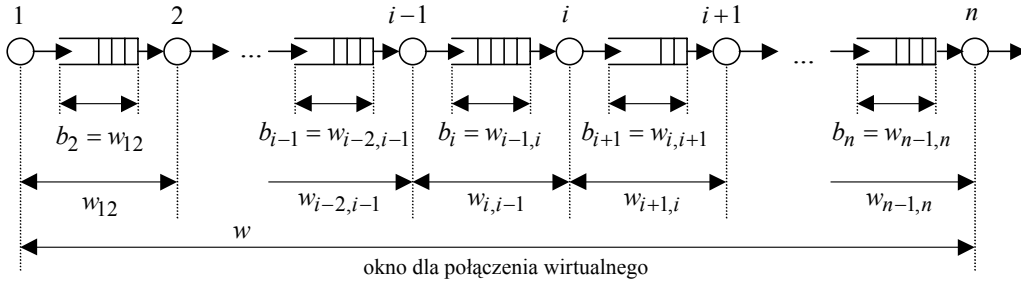
$$b = \sum_{i=2,3,\dots,n} b_i = w.$$

Warunkiem zachowania ciągłości przepływu w połączeniu wirtualnym jest suma rozmiarów okien pomiędzy sąsiednimi węzłami w_{ij} równa rozmiarowi okna dla połączenia wirtualnego w :

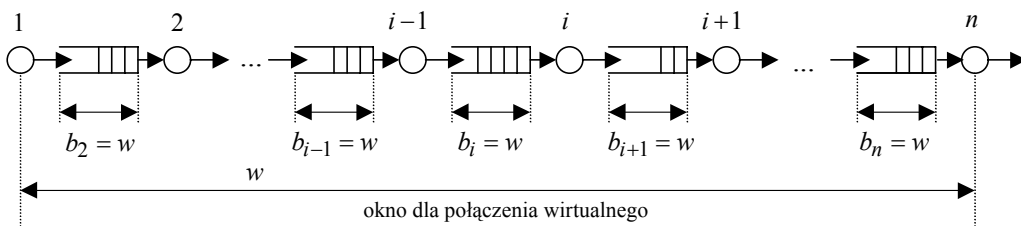
$$w = w_{12} + w_{23} + \dots + w_{i-1,i} + w_{i,i+1} + \dots + w_{n-1,n} = \sum_{i=1,2,\dots,n-1; j=i+1} w_{ij},$$

oraz pojemności pamięci buforowych b_i , zarezerwowanych dla danego przepływu w kolejnych węzłach odbiorczych na trasie, równe rozmiarom okien w łączach prowadzących do tych węzłów odbiorczych $w_{i-1,i}$:

$$b_i \leq w_{i-1,i} \text{ dla } i = 2, 3, \dots, n.$$



a) dwupoziomowe sterowanie przepływem z wykorzystaniem mechanizmu okna



b) jednopozomowe sterowanie przepływem z wykorzystaniem mechanizmu okna

Rysunek 5.19. Dwupoziomowe i jednopozomowe sterowanie przepływem z wykorzystaniem mechanizmu okna

Sterowanie przepływem w połączeniu wirtualnym jest ciągiem sterowań przepływem pomiędzy kolejnymi węzłami trasy. Jeżeli pamięć buforowa i -tego węzła jest zajęta (w całości lub powyżej ustalonego progu zajętości) przez jednostki danych nadesłane z $(i-1)$ -go węzła, to i -ty węzeł wstrzymuje wysłanie potwierdzenia do $(i-1)$ -go węzła. Warunkiem wznowienia przyjmowania kolejnych jednostek danych w i -tym węźle jest wysłanie potwierdzenia z i -tego węzła do $(i-1)$ -go węzła, którego warunkiem z kolei jest zwolnienie miejsca w pamięci buforowej i -tego węzła, tzn. wysłanie jednostki danych przez i -ty węzeł do $(i+1)$ -go węzła. Warunkiem wysłania jednostki danych przez i -ty węzeł jest otrzymanie przez ten węzeł potwierdzenia z $(i+1)$ -go węzła itd.

Wypełnienie pamięci buforowej i -tego węzła, powodując wstrzymanie przyjmowania nowych jednostek danych z $(i-1)$ -go węzła, może być przyczyną wypełnienia pamięci buforowej $(i-1)$ -go węzła i wstrzymania wysyłania potwierdzeń do węzła $(i-2)$ -go węzła, co z kolei może być przyczyną wypełnienia pamięci buforowej $(i-2)$ -go węzła itd.

Konsekwencją przeciążenia łącza pomiędzy węzłami i -tym i $(i + 1)$ -szym jest wsteczna propagacja (ang. *backpressure*) przeciążenia w kierunku do źródła przepływu, w wyniku której pamięci buforowe węzłów, znajdujących się pomiędzy przeciążonym łączem a źródłem przepływu ($i - 1, i - 2, \dots, 2, 1$), są kolejno wypełniane w oczekiwaniu na wznowienie obsługi przepływu przez przeciążone łącze. Stopniowe wypełnianie się pamięci buforowych węzłów jest równoważne wypełnianiu okien międzywęzłowych, i dalej – wypełnianiu okna połączenia wirtualnego. Wynikiem stopniowego wypełniania pamięci buforowych węzłów jest wstrzymanie wprowadzania nowych jednostek danych przez źródło przepływu. Wstrzymanie wprowadzania nowych jednostek danych przepływu w źródle oznacza, że liczba jednostek danych przepływu, oczekujących w pamięciach buforowych węzłów na wznowienie obsługi przepływu przez i -te, $(i + 1)$ -sze łącze, jest równa sumie pojemności buforowych węzłów:

$$N = b_i + b_{i-1} + \dots + b_2.$$

Cechą omawianej organizacji sterowania przepływem jest to, że całkowita liczba jednostek danych przepływu, oczekujących na wznowienie obsługi, jest równomiernie rozłożona pomiędzy wszystkie węzły znajdujące się pomiędzy przeciążonym łączem a źródłem przepływu.

W najgorszym razie, tzn. wtedy, gdy wstrzymywane jest wyprowadzanie jednostek danych w ujściu połączenia wirtualnego, możliwe jest wypełnienie pamięci buforowych wszystkich węzłów trasy. Jeżeli pojemności pamięci buforowych węzłów są jednakowe, tzn. jeżeli $w_{ij} = \bar{w}$ dla $i = 1, 2, \dots, n - 1; j = i + 1$, to maksymalna liczba jednostek danych oczekujących w węzłach trasy jest równa sumie rozmiarów okien międzywęzłowych i rozmiarowi okna połączenia wirtualnego:

$$N_{\max} = (n - 1)\bar{w} = w.$$

W przypadku implementacji jednopoziomowego sterowania przepływem pomiędzy źródłem i ujściem z wykorzystaniem mechanizmu okna o rozmiarze w , równoważnego omawianemu dwupoziomowemu sterowaniu przepływem, maksymalna liczba jednostek danych oczekujących na wznowienie obsługi $N_{\max} = w$. Zastosowanie jednopoziomowego sterowania przepływem oznacza, że jednostki danych oczekujące na wznowienie obsługi nie są równomiernie rozłożone pomiędzy węzły trasy, ale mogą być kumulowane w poszczególnych jej węzłach. Warunkiem efektywności jednopoziomowego sterowania przepływem jest, aby pojemności pamięci buforowych węzłów nie były mniejsze od rozmiaru okna w , tj. $b_i = w$ dla $i = 2, 3, \dots, n$. Suma pojemności pamięci buforowych węzłów trasy b , rezerwowanych dla danego przepływu, jest równa iloczynowi liczby węzłów odbiorczych na trasie i rozmiarowi okna dla przepływu źródło–ujście:

$$b = (n - 1)w.$$

Główną zaletą dwupoziomowej hierarchii mechanizmów okna w sterowaniu przepływem pomiędzy źródłem i ujściem, w porównaniu z mechanizmem okna implementowanym tylko na poziomie trasy źródło–ujście, jest:

➤ równomierne rozproszenie całkowitej liczby jednostek danych, oczekujących na wznowienie obsługi w przypadku przeciążenia dowolnego elementu trasy, pomiędzy wszystkie węzły znajdujące się pomiędzy przeciążonym elementem a źródłem przepływu,

➤ mniejsze zapotrzebowanie na pojemności pamięci buforowych w węzłach trasy pomiędzy źródłem a ujściem.

Zastosowanie dwupoziomowego sterowania przepływem dla każdego z połączeń wirtualnych, obsługiwanych w danym węźle sieci, oznacza także, że w razie przeciążenia łączy wychodzących z tego węzła stopień monopolizacji zajętości pamięci buforowej węzła przez jeden z przepływów zależy od względnych rozmiarów okien w przepływach pomiędzy węzłami trasy, tzn. jest ograniczony rozmiarami okien w przepływie od węzła do węzła, a nie rozmiarami okien przepływów źródło–ujście.

5.10. PODSUMOWANIE

Podstawowym celem implementacji procedur sterowania przepływem jest sprawiedliwy rozdział zasobów pomiędzy pojedyncze przepływy lub klasy przepływów oraz uzyskanie kompromisu pomiędzy średnim opóźnieniem jednostek danych a przepustowością sieci lub jej elementów.

W praktycznie stosowanych procedurach sterowania przepływem wykorzystywane są dwie grupy metod, oparte na różnych koncepcjach utrzymywania zapotrzebowania na zasoby sieci na ustalonym poziomie. Poziom zapotrzebowania na zasoby sieci jest determinowany wartościami parametrów wybranych wskaźników jakości działania sieci i usług sieci; zwłaszcza średnim opóźnieniem wnoszonym przez sieć i przepustowością sieci. Koncepcje te różnią się złożonością ich implementacji i możliwościami realizacji sprawiedliwego rozdziału zasobów pomiędzy przepływy współdzielące zasoby sieci.

Zadaniem pierwszej grupy metod jest utrzymywanie liczby jednostek danych (np. pakietów) w sieci lub jej wybranych elementach poniżej ustalonych wartości. Podstawowymi zaletami takich rozwiązań, na ogół korzystających z różnych odmian mechanizmu przesuwającego okna, jest względnie niewielki nadmiar organizacyjny i duża szybkość reagowania na przeciążenia sieci. Wadą tych metod jest mała skalowalność oraz trudność implementacji zasad sprawiedliwego rozdziału zasobów. Zwiększenie liczby przepływów poddanych sterowaniu prowadzi do dławienia źródeł przepływów, wzrostu wartości średnich opóźnień oraz pogorszenia przepustowości.

W drugiej grupie metod sterowanie przepływem polega na ograniczaniu szybkości wprowadzania jednostek danych do sieci lub jej elementów. Metody te charakteryzuje mała wrażliwość na zmiany liczby przepływów w sieci oraz łatwość implementacji algorytmów sprawiedliwego rozdziału zasobów.

Zadania sterowania przepływem są ściśle związane z zadaniami wyznaczania optymalnych tras w sieci. Oznacza to, że zadania wyznaczania optymalnych tras i sterowania przepływem mogą być formułowane rozdzielnie lub łącznie. Zadanie wyznaczania przepływów w sieci jest szczególnym przypadkiem zadania wyznaczania optymalnych tras.

LITERATURA

- [1] ATHANS M.A., FALB P.L., *Sterowanie optymalne*, WNT, Warszawa 1969.
- [2] ATKINS J.D., *Path control: A Transport Network of SNA*, IEEE Transactions on Communications, Vol. COM-28, Vol. 2, February 1980, s. 527–538.
- [3] BATYCKI T., GRZECH A., *A stochastic models of network's mode performance and transmission demand in computer communication network*, Proceedings of the 12th Hawaii International Conference on System Sciences, Vol. 2, Honolulu 1979, s. 280–288.
- [4] BATYCKI T., GRZECH A., KASPRZAK A., *Optimal routing in a hybrid packet and circuit computer communication network*, Systems Science, Vol. 8, No. 2/3, 1982, s. 137–147.
- [5] BATYCKI T., KASPRZAK A., *Wybrane algorytmy optymalizacji przepływów w sieciach teleinformatycznych*, Wydawnictwo Politechniki Wrocławskiej, Wrocław 1983.
- [6] BERSEKAS D., *Dynamic behaviour of shortest path routing algorithm for communication networks*, IEEE Transactions on Automatic Control, Vol. AC-27, Vol. 1, January 1982, s. 60–74.
- [7] BERTSEKAS D., GALLAGER R., *Data networks*, Prentice-Hall, Englewood Cliffs 1987.
- [8] BHARATH-KUMAR K., JAFFE J.M., *Routing to multiple destinations in computer networks*, IEEE Transactions on Communications, Vol. COM-31. No. 3, March 1983, s. 343–351.
- [9] BUBNICKI Z., *Identification of control plants*, Elsevier, Amsterdam 1980.
- [10] CALBOURN C.J., *The combinatorics of network reliability*, Oxford University Press, Oxford 1987.
- [11] CZACHÓRSKI T., *Modele kolejkowe w ocenie efektywności sieci i systemów komputerowych*, Pracownia Komputerowa Jacka Skalmierskiego, Gliwice 1999.
- [12] ELBAUM R., SIDI M., *Topological design of local-area networks using genetic algorithms*, IEEE Transactions on Networking, Vol. 4, No. 5, October 1996, s. 766–778.
- [13] FINDEISEN W., SZYMANOWSKI J., WIERZBICKI A., *Teoria i metody obliczeniowe optymalizacji*, PWN, Warszawa 1977.
- [14] FISZ M., *Rachunek prawdopodobieństwa i statystyka matematyczna*, PWN, Warszawa 1967.
- [15] FORD L.R., FULKERSON D.R., *Przepływy w sieciach*, PWN, Warszawa 1969.
- [16] FRATTA L., GERLA M., KLEINROCK L., *The Flow Deviation method: An approach to store-and-forward communication network design*, Networks, Vol. 3, 1973, s. 97–133.
- [17] GERLA M., KLEINROCK L., *On the topological design of distributed computer networks*, IEEE Transactions on Communications, Vol. COM-25, s. 48–60, 1977.

- [18] GRZECH A., *Metoda grupowego dostępu do kanału transmisyjnego*, Archiwum Automatyki i Telemekhaniki, t. 33, z. 1, 1988, s. 147–163.
- [19] GRZECH A., *Group access-control discipline for multiaccess channel*, International Seminar on Modelling and Traffic Control of Telecommunication Networks, Zeszyty Naukowe AGH nr 1180, Automatyka z. 44, Kraków 1989, s. 41–58.
- [20] GRZECH A., KASPRZAK A., CHOUIGUI N., *The new expressions for blocking probability for route in integrated services digital networks*, Proceedings of the 4th International Symposium on Systems Analysis and Simulation, Elsevier, Berlin 1992, s. 313–318.
- [21] GRZECH A., KASPRZAK A., *Optimal routing and frame distribution assignment problem in the integrated services digital network*, Applied Mathematics and Computer Science, Vol. 2, No. 1, 1992, s. 5–20.
- [22] GRZECH A., KASPRZAK A., CHOUIGUI N., *Blocking probability expressions and their applications to optimization in ISDN*, Systems Anal., Vol. 13, 1993, s. 175–188.
- [23] GRZECH A., KASPRZAK A., *Control of traffic in an integrated communication subsystem*, Proceedings of IEEE Singapore International Conference on Networks, Vol. 2, IEEE 1993, s. 761–765.
- [24] GRZECH A., *Integrated communication subsystems of local area networks*, Systems Science, Vol. 19, No. 2, 1993, s. 55–67.
- [25] GRZECH A., KASPRZAK A., *Blocking probability versus stream rate for route in ISDN*, Systems Science, Vol. 19, No. 4, 1993, s. 77–85.
- [26] GRZECH A., KOLLEK R., *Sub-optimal routing at computer communication networks with unreliable transmission channels*, Archiwum Informatyki Teoretycznej i Stosowanej, t. 13, z. 4, 2001, s. 375–387.
- [27] HUANG Y.M., GUAN S.U., *A refined cut-through buffer management scheme for layered protocol stacks*, IEEE Communications Magazine, Vol. 32, No. 3, March 1994, s. 82–86.
- [28] JAFFE J.M., *Flow control power is nondecentralizable*, IEEE Transactions on Communications, Vol. COM-29, No. 9, September 1981, s. 1301–1306.
- [29] KERMANI P., KLEINROCK L., *Dynamic flow control in store-and-forward computer networks*, IEEE Transactions on Communications, Vol. COM-28, No. 2, February 1980, s. 263–270.
- [30] KLEINROCK L., *Communication nets*, Volume I: *Stochastic message flow and delay*, Volume II: *Computer Applications*, Wiley, New York 1976.
- [31] KLEINROCK L., KERMANI P., *Static flow control in store-and-forward computer networks*, IEEE Transactions on Communications, Vol. COM-28, No. 2, February 1980, s. 271–278.
- [32] KULIKOWSKI J.L., *Zarys teorii grafów*, PWN, Warszawa 1986.
- [33] LAM Y.F., LI V.O.K., *An improved algorithm for performance analysis of networks with unreliable components*, IEEE Transactions on Communications, Vol. COM-23, 1975, s. 891–904.
- [34] MELSA J.L., SAGE A.P., *An introduction to probability and stochastic processes*, Prentice Hall, Englewood Cliffs 1973.
- [35] MISHRA S., WU L., *An evaluation of flow control in group communication*, IEEE/ACM Transactions on Networking, Vol. 6, No. 5, October 1998, s. 571–587.

-
- [36] ORDA A., *Routing with end-to-end QoS guarantees in broadband networks*, IEEE/ACM Transactions on Networking, Vol. 7, No. 3, June 1999, s. 365–374.
 - [37] PAPIR Z., *Ruch telekomunikacyjny i przeciążenia sieci pakietowych*, WKŁ, Warszawa 2001.
 - [38] SEIDLER J., *Analiza i synteza sieci łączności dla systemów teleinformatycznych*, PWN, Warszawa 1979.
 - [39] TAKAGI H., KLEINROCK L., *Throughput delay characteristics of some slotted-Aloha packet radio networks*, IEEE Transactions on Communications, Vol. COM-33, No. 6, June 1985, s. 1200–1207.
 - [40] TAKAGI H. (red.), *Stochastic analysis of computer and communication systems*, North-Holland, Amsterdam 1990.
 - [41] YAN J., *Adaptive configuration of elastic high-speed multiclass networks*, IEEE Communications Magazine, May 1998, s. 116–120.
 - [42] WOLFINGER B.E., *Characterization of mixed traffic load in service integrated networks*, Proceedings 6th Polish Teletraffic Symposium, Wrocław 1999, s. 11–28.

6. STEROWANIE PRZEPLYWEM

Wielopoziomowe, hierarchiczne architektury systemów teleinformatycznych wymuszają analogiczne struktury zarządzania przepływem w tych systemach, co oznacza m.in., że realizacja dowolnej procedury sterowania przepływem jest związana z wbudowaniem tych procedur do jednego lub wielu protokołów systemu.

Wielopoziomowa, hierarchiczna i zagnieżdżona struktura systemu sterowania umożliwia dekompozycję zadania sterowania przepływem na zadania lokalizowane w różnych warstwach architektury sieci i dotyczących zarówno przepływów na różnych poziomach granulacji przepływów, jak i różnych fragmentów procesów obsługi przepływów. Występowanie wielu poziomów sterowania przepływem w tym samym systemie teleinformatycznym, na których są stosowane procedury wzajemnie uzupełniające się i często redundantne, powoduje, że ważnym zadaniem projektowania i analizy są zagadnienia integracji i interakcji procedur sterowania. Szczególnie istotnymi zadaniami integracji i interakcji procedur sterowania przepływem są zadania formułowane m.in. w przypadkach łączenia i obsługi w systemie teleinformatycznym:

- różnych trybów komutacji, tzn. komutacji kanałów i różnych odmian komutacji jednostek danych (wiadomości, bloków danych, segmentów, pakietów, ramek, komórek),
- różnych trybów dystrybucji ruchu,
- różnych klas ruchu, wyróżnianych ze względu na wrażliwość na wartości wybranych miar jakości obsługi ruchu i (lub) aplikacje generujące ruch w sieci,
- wymagań z zakresu jakości usług sieci, dotyczących różnie formułowanych warunków gwarancji jakości, oraz efektywnego wykorzystania zasobów,
- różnych koncepcji sterowania przepływami, tzn. prewencyjnymi i reakcyjnymi, zależnymi od szybkości procesów realizowanych w systemach i bezwładności systemów,
- systemów o różnych architekturach, wymagających łączenia i konwersji różnych struktur i procedur sterowania przepływem.

6.1. ALOKACJA ZASOBÓW W SIECIACH TELEINFORMATYCZNYCH

Podsystem komunikacyjny każdej sieci komputerowej, a zwłaszcza sieci z komutacją jednostek danych (wiadomości, pakietów, ramek, komórek), można traktować

jako rozproszony zbiór zasobów współdzielonych pomiędzy współzawodniczące o te zasoby komunikujące się procesy (w węższym znaczeniu – użytkownicy). Cechami charakterystycznymi sieci z komutacją jednostek danych są:

- dopuszczalność sytuacji, kiedy chwilowe zapotrzebowanie na obsługę jest większe od pojemności sieci,

- dynamiczny przydział zasobów dla komunikujących się procesów.

Taki sposób przydziału zasobów zasadniczo różni sieci z komutacją jednostek danych od sieci z komutacją kanałów, charakteryzujących się tym, że decyzję o przyjęciu zadania do obsługi poprzedza przyznawanie zasobów dla komunikujących się procesów.

Podstawowe korzyści wynikające z dynamicznego przydzielania zasobów dla komunikujących się procesów to szybkość i elastyczność ustanawiania połączeń procesów poprzez podsystem komunikacyjny sieci oraz efektywniejsze – w porównaniu do sieci z komutacją łączy – wykorzystanie zasobów sieciowych. Osiągnięcie wymienionych korzyści odbywa się za cenę implementacji w sieci złożonych systemów sterowania. Nieuważne sterowanie ruchem wchodzącym do sieci oraz alokacją zasobów może spowodować powstawanie przeciążeń (ang. *congestion*) oraz zakleszczeń (ang. *deadlock*), niwelujących wcześniej wymienione korzyści stosowania techniki komutacji jednostek danych i prowadzących do degradacji jakości usług komunikacyjnych.

Zapobieganie przeciążeniom i likwidacja skutków przeciążeń jest podstawą klasyfikacji metod sterowania w sieciach na metody zapobiegania przeciążeniom (metody i sterowanie prewencyjne) oraz metody redukcji przeciążeń – zwłaszcza redukcji i eliminacji następstw przeciążeń (metody i sterowanie reaktywne).

Nie jest możliwe efektywne wykorzystanie zasobów sieci tylko za pomocą metod należących do jednej z wymienionych grup sterowania. Całkowita eliminacja przeciążeń w wyniku zapobiegania, tzn. na drodze limitowania ruchu wchodzącego do sieci byłaby możliwa tylko przez kierowanie zasobów do obsługi wchodzącego ruchu, co przeczyłoby idei sieci z komutacją jednostek danych. Z kolei sterowanie oparte tylko na metodach eliminacji skutków przeciążeń prowadziło do stanu permanentnego przeciążenia. Skuteczne sterowanie siecią i jakością usług sieciowych wymaga jednoczesnego stosowania algorytmów należących do obydwu wymienionych grup.

Podstawowym celem metod zapobiegania powstawaniu przeciążeń (ang. *congestion control*) i sterowania przepływem (ang. *flow control*) jest obsługa ruchu wprowadzonego do sieci przez zbiorowość jej użytkowników (ang. *offered load*), polegająca na:

- zapobieganiu wprowadzaniu do sieci ruchu nadmiarowego w porównaniu z możliwościami obsługi ruchu,

- efektywnej obsłudze przyjętego ruchu w sieci.

Są to zadania dynamicznego dostosowywania ilości ruchu teleinformatycznego, wprowadzanego do sieci przez komunikujące się procesy, do aktualnych możliwości obsługi ruchu przez sieć. Złożoność i wyrafinowanie metod i technik zapobiegania przeciążeniom i sterowania przepływem w sieciach wynika z tego, że elementy sieci, komunikując się między sobą, mogą koordynować swoje działania w ramach wspólnej rozproszonej strategii sterowania.

Do głównych zadań sterowania przepływem w sieciach z komutacją pakietów (w ogólnym przypadku w sieciach z komutacją jednostek danych) zalicza się:

- zapobieganie degradacji przepustowości (ang. *throughput degradation*) i utracie efektywności z powodu przepełnienia (ang. *overload*),
- sprawiedliwa alokacja (ang. *fair resource allocation*) współdzielonych zasobów pomiędzy procesy współzawodniczące o nie,
- unikanie zakleszczeń (ang. *deadlock avoidance*),
- dopasowanie szybkości (ang. *speed matching*) transferu pomiędzy siecią a procesami (użytkownikami) korzystającymi z usług sieci.

Zadania te są często sprzeczne. Na przykład skuteczne zapobieganie degradacji przepustowości i utracie efektywności może oznaczać konieczność odrzucenia ustalonych wcześniej klas wprowadzanego i obsługiwanego w sieci ruchu teleinformatycznego.

6.2. ZADANIA STEROWANIA PRZEPIŁYWEM

6.2.1. DEGRADACJA PRZEPUSTOWOŚCI

Główną przyczyną degradacji przepustowości jest marnotrawienie zasobów sieciowych, będące następstwem:

- konfliktu żądań dostępu do zasobów uniemożliwiającego skorzystanie z nich do chwili jego rozstrzygnięcia,
- nieprawidłowego przyznania zasobów w porównaniu z zapotrzebowaniem na nie lub możliwością ich wykorzystania.

Najczęściej trwonionymi zasobami sieciowymi są:

1. Przepustowość kanałów transmisyjnych – nieprawidłowy dobór zasobów lub zasad korzystania z nich, determinujących możliwość odbioru transmitowanych pakietów, może skutkować:

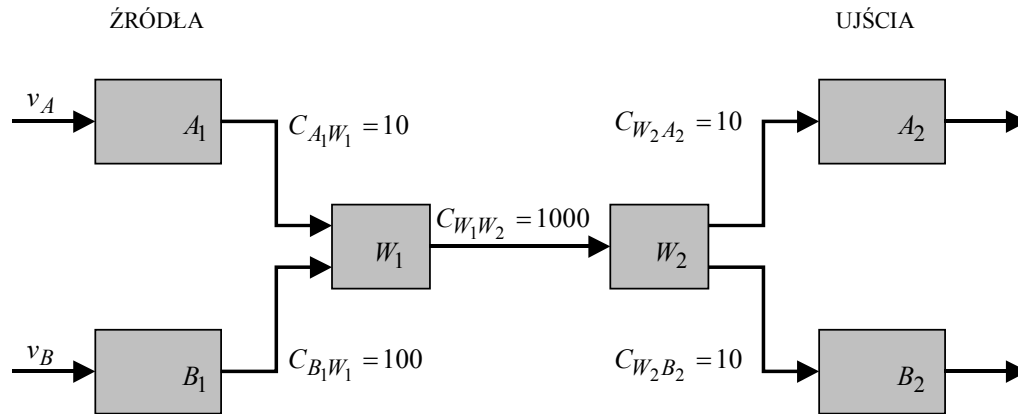
- niewykorzystaniem dostępnych przepustowości,
- wykorzystaniem dostępnych przepustowości na potrzeby retransmisji pakietów, powodowanych brakiem możliwości odbioru pakietów.

2. Pojemność pamięci buforowych – nieprawidłowy dobór zasobów lub reguł korzystania z zasobów, wyznaczających możliwość transmisji lub odbioru pakietów (lub wiadomości), może powodować:

- niewykorzystanie pojemności buforowych,
- zawłaszczenie pojemności pamięci buforowych w oczekiwaniu na zakończenie obsługi zadań transmisji pakietów (lub wiadomości).

Przykład 6.1: Degradacja przepustowości powodowana niedopasowaniem zasobów

Jednym z przykładów degradacji całkowitej przepustowości systemu jest degradacja powodowana niewłaściwym doбором zasobów transmisyjnych (decydującym o dostępie procesów do pamięci buforowej) i niewłaściwym zarządzaniem pojemnością współdzielonej pamięci buforowej (rysunek 6.1).



Rysunek 6.1. Transmisja pakietów pomiędzy dwoma parami źródeł i ujść

Transmisja pakietów pomiędzy dwiema parami źródeł i ujść, tj. A_1, A_2 i B_1, B_2 , odbywa się w systemie zawierającym dwa węzły: W_1 i W_2 (rysunek 6.1). Średnia szybkość napływu pakietów generowanych przez źródła A_1 i B_1 jest oznaczona – odpowiednio – przez v_A i v_B (pakietów na sekundę). Pojemność kanałów transmisyjnych wynosi: $C_{A_1W_1} = 10$, $C_{B_1W_1} = 100$, $C_{W_1W_2} = 1000$, $C_{W_2A_2} = 10$ oraz $C_{W_2B_2} = 10$ (pakietów na sekundę), pojemność pamięci buforowych węzłów jest natomiast skończona: $C_{W_1} = K_1$ i $C_{W_2} = K_2$ (wartości K_1 i K_2 są dobrane tak, że nie ograniczają obsługi generowanych strumieni pakietów).

Potencjalna przepustowość (maksymalna) omawianego systemu, oznaczana przez PS_p , jest równa sumie maksymalnych przepustowości tras (łączy) pomiędzy źródłami i ujściami (przepustowość łącza jest równa minimalnej pojemności kanałów transmisyjnych składających się na trasę): $PS_p = C_{W_2A_2} + C_{W_2B_2} = 20$.

Rzeczywista przepustowość omawianego systemu, oznaczana przez PS_r , zależy od zasad obsługi generowanych strumieni pakietów w węzle W_1 , pod warunkiem, że zasady obsługi strumieni pakietów w obydwu węzłach W_1 i W_2 są takie same (przy tym założeniu względnie duża wartość pojemności $C_{W_1W_2}$ powoduje, że stany węzła W_2 są powtórzeniami stanów węzła W_1).

Przyjęcie założenia, że pojemność pamięci buforowej węzłów jest nieskończona oraz że cała pamięć buforowa jest dostępna dla pakietów pochodzących z dowolnego źródła oznacza, że dla omawianego przykładowego systemu charakterystyczne są następujące przypadki:

Przypadek 1: $v_A < C_{A_1W_1} = 10$ oraz $v_B < C_{W_2B_2} = 10$

Szybkość generowania pakietów w źródle A_1 – mniejsza od pojemności łącza pomiędzy źródłem A_1 a ujściem A_2 , tj. $v_A < C_{A_1W_1} = C_{W_2A_2} \ll C_{W_1W_2}$, oraz szybkość gene-

rowania pakietów w źródle B_1 – mniejsza od pojemności łącza pomiędzy źródłem B_1 a ujściem B_2 , tj. $v_B < C_{W_2B_2} < C_{B_1W_1} \ll C_{W_1W_2}$ – powodują, że zasoby transmisyjne systemu, tzn. pojemność łączy transmisyjnych pomiędzy źródłami a ujściami, są większe niż zapotrzebowanie na zasoby. Szybkość napływu pakietów do systemu ze źródeł i szybkość ich wypływu z systemu do ujść nie jest ograniczana pojemnością kanałów transmisyjnych. Generowane strumienie pakietów nie powodują kolejek, w których pakiety oczekiwałyby na transmisję.

Rzeczywista przepustowość systemu jest równa sumie szybkości generowanych strumieni pakietów: $PS_r = v_A + v_B$. Równoważną miarą jakości systemu jest współczynnik jego wykorzystania: $a = (v_A + v_B) / (C_{W_2A_2} + C_{W_2B_2})$.

Przypadek 2: $v_A < C_{A_1W_1} = 10$ oraz $C_{W_2B_2} = 10 \leq v_B < C_{B_1W_1} = 100$

Szybkość generowania pakietów w źródle A_1 – mniejsza od pojemności łącza pomiędzy źródłem A_1 a ujściem A_2 , tj. $v_A < C_{A_1W_1} = C_{W_2A_2} \ll C_{W_1W_2}$, oraz szybkość generowania pakietów w źródle B_1 – nie mniejsza od pojemności łącza pomiędzy źródłem B_1 a ujściem B_2 i mniejsza od pojemności kanału transmisyjnego ze źródła B_1 do węzła W_1 , tj. $C_{W_2B_2} \leq v_B < C_{B_1W_1} \ll C_{W_1W_2}$ – powodują, że w stanie ustalonym szybkość napływu pakietów ze źródła B_1 do systemu jest większa od szybkości wypływu pakietów z systemu do ujścia B_2 . Kolejka pakietów oczekujących w pamięci buforowej węzła W_2 na transmisję w kanale transmisyjnym prowadzącym do ujścia B_2 rośnie. Zwiększenie liczby pakietów oczekujących na transmisję powoduje z czasem, że pamięć buforową węzłów zajmują pakiety pochodzące z omawianych dwóch źródeł proporcjonalnie do ilorazu szybkości ich napływu, tj. v_A/v_B . Wobec jednakowych pojemności kanałów wyjściowych z węzła W_2 do ujść A_2 i B_2 rzeczywistą przepustowość systemu determinuje szybkość opuszczania pamięci buforowej węzłów przez pakiety generowane w źródle B_1 i adresowane do ujścia B_2 oraz możliwość zajęcia zwalnianej w ten sposób pamięci buforowej przez pakiety generowane w źródle A_1 .

Nadmiar pakietów oczekujących na transmisję do ujścia B_2 , w porównaniu z pojemnością kanału transmisyjnego $C_{W_2B_2}$ używanego do tego celu, powoduje, że rzeczywista przepustowość systemu: $PS_r = C_{W_2B_2} + (v_A/v_B)C_{W_2A_2}$.

Przypadek 3: $v_A < C_{A_1W_1} = 10$ oraz $v_B \geq C_{B_1W_1} = 100$

Szybkość generowania pakietów w źródle A_1 – mniejsza od pojemności łącza pomiędzy źródłem A_1 a ujściem A_2 , tj. $v_A < C_{A_1W_1} = C_{W_2A_2} \ll C_{W_1W_2}$, oraz szybkość generowania pakietów w źródle B_1 – większa od pojemności łącza pomiędzy źródłem B_1 a ujściem B_2 i nie mniejsza niż pojemność kanału transmisyjnego ze źródła B_1 do węzła W_1 , tj. $C_{W_2B_2} < C_{B_1W_1} \leq v_B \ll C_{W_1W_2}$ – powodują, że (w stanie ustalonym) szybkość napływu pakietów do pamięci buforowej ze źródła B_1 jest ograniczona pojemnością

kanału transmisyjnego $C_{B_1W_1}$, a szybkość wypływu pakietów do ujścia B_2 – pojemnością $C_{W_2B_2}$. W tym przypadku – podobnym do poprzednio analizowanego – następuje zawłaszczenie pojemności pamięci buforowej węzłów przez strumień pakietów generowanych w źródle B_1 kosztem strumienia pakietów generowanego w źródle A_1 , rzeczywista przepustowość systemu: $PS_r = C_{W_2B_2} + (v_A / C_{B_1W_1}) C_{W_2A_2}$.

Przypadek 4: $v_A \geq C_{A_1W_1} = 10$ oraz $v_B < C_{B_1W_1} = 10$

Szybkość generowania pakietów w źródle A_1 – większa od pojemności łącza pomiędzy źródłem A_1 a ujściem A_2 , tj. $C_{A_1W_1} = C_{W_2A_2} < v_A \ll C_{W_1W_2}$, oraz szybkość generowania pakietów w źródle B_1 – mniejsza od pojemności łącza pomiędzy źródłem B_1 a ujściem B_2 , tj. $v_B < C_{W_2B_2} < C_{B_1W_1} \ll C_{W_1W_2}$ – powodują, że szybkość pakietów napływających ze źródła A_1 do systemu jest ograniczona pojemnością kanału transmisyjnego do węzła W_1 . Wypływ pakietów generowanych w źródle A_1 i zaakceptowanych przez system do ujścia A_2 nie jest ograniczany pojemnością kanału transmisyjnego z systemu do ujścia A_2 .

W omawianym przypadku przepustowość systemu jest równa sumie pojemności kanału transmisyjnego z A_1 do W_1 i szybkości strumienia pakietów generowanych w źródle B_1 : $PS_r = C_{A_1W_1} + v_B$.

Przypadek 5: $v_A \geq C_{A_1W_1} = 10$ oraz $C_{W_2B_2} = 10 \leq v_B < C_{B_1W_1} = 100$

Szybkość generowania pakietów w źródle A_1 – większa od pojemności łącza pomiędzy źródłem A_1 a ujściem A_2 , tj. $C_{A_1W_1} = C_{W_2A_2} < v_A \ll C_{W_1W_2}$, oraz szybkość generowania pakietów w źródle B_1 – nie mniejsza od pojemności łącza pomiędzy źródłem B_1 a ujściem B_2 i mniejsza od pojemności kanału transmisyjnego ze źródła B_1 do węzła W_1 , tj. $C_{W_2B_2} \leq v_B < C_{B_1W_1} \ll C_{W_1W_2}$ – powodują, że (w stanie ustalonym) napływ pakietów ze źródła A_1 jest ograniczony pojemnością kanału transmisyjnego z tego źródła do systemu. Z kolei wypływ pakietów zaakceptowanych przez system, a pochodzących ze źródła B_1 , jest ograniczony pojemnością kanału transmisyjnego z systemu do ujścia B_2 . Dysproporcja pomiędzy szybkością napływu do systemu i wypływu z systemu pakietów generowanych w źródle B_1 powoduje, że pamięć buforową zajmują pakiety generowane w źródłach A_1 i B_1 proporcjonalnie do ilorazu szybkości napływu: $C_{A_1W_1}/v_B$.

Zawłaszczenie pojemności pamięci buforowej przez pakiety pochodzące ze źródła B_1 powoduje, że rzeczywista przepustowość systemu: $PS_r = C_{W_2B_2} + (C_{A_1W_1}/v_B) C_{W_2A_2}$.

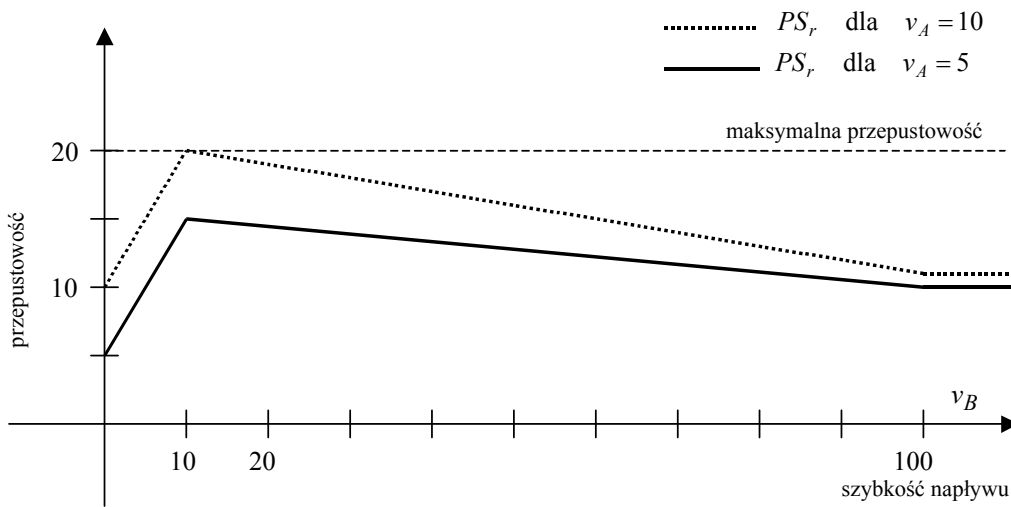
Przypadek 6: $v_A \geq C_{A_1W_1} = 10$ oraz $v_B \geq C_{B_1W_1} = 100$

Szybkość generowania pakietów w źródle A_1 – większa od pojemności łącza pomiędzy źródłem A_1 a ujściem A_2 , tj. $C_{A_1W_1} = C_{W_2A_2} \leq v_A \ll C_{W_1W_2}$, oraz szybkość gene-

rowania pakietów w źródle B_1 – większa od pojemności łącza pomiędzy źródłem B_1 a ujściem B_2 i nie mniejsza od pojemności kanału transmisyjnego ze źródła B_1 do węzła W_1 , tj. $C_{W_2B_2} < C_{B_1W_1} \leq v_B \ll C_{W_1W_2}$ – powodują, że szybkość pakietów napływających ze źródeł A_1 i B_1 do systemu jest ograniczona pojemnościami kanałów transmisyjnych prowadzących z tych źródeł do węzła W_1 .

Ponieważ możliwość wyprowadzania pakietów z systemu do ujść A_2 i B_2 jest determinowana pojemnościami kanałów transmisyjnych $C_{W_2A_2}$ i $C_{W_2B_2}$, większa szybkość napływu pakietów ze źródła B_1 powoduje więc zawłaszczenie przez to źródło pojemności pamięci buforowej, kosztem strumienia pakietów generowanego w źródle A_1 , proporcjonalnie do szybkości napływu pakietów – rzeczywista przepustowość systemu: $PS_r = C_{W_2B_2} + (C_{A_1W_1}/C_{B_1W_1})C_{W_2A_2}$ (dla danych z przykładu $PS_r = 11$).

Wyrażenia określające rzeczywistą przepustowość w omawianym systemie, dla ustalonych wartości strumienia v_A i zmiennych wartości strumienia v_B , są prawdziwe dla modelu obsługi ze stratami (ang. *loss model*), zgodnie z którym pakiety napływające do węzła są odrzucane, jeżeli nie ma możliwości ich obsługi, tzn. dalszej transmisji do węzła ujściowego (rysunek 6.2).



Rysunek 6.2. Zależność rzeczywistej przepustowości systemu dla ustalonych wartości strumienia v_A i zmiennych wartości strumienia v_B

Innym modelem obsługi jest model z retransmisjami (ang. *retransmission model*). Pakiety nieobsłużone w danym węźle są retransmitowane przez węzeł źródłowy. Przyjęcie modelu z retransmisjami oznacza, że szybkość napływu pakietów do węzła jest sumą szybkości napływu podstawowego strumienia pakietów oraz strumienia pakietów retransmitowanych. Otrzymane w taki sposób zwiększenie szybkości na-

plywu pakietów do węzła powoduje – w przypadkach, których to dotyczy – zwiększenie szybkości wypełniania pojemności pamięci buforowej w węzłach.

W omawianym przykładzie:

➤ przypadek 1. to przykład nadmiarowości zasobów (zasobów transmisyjnych) nie wpływających na wartość przyjętego kryterium jakości działania systemu,

➤ przypadki 2. i 3. to przykład nieprawidłowego dopasowania zasobów transmisyjnych obsługujących wejście i wyjście z systemu; dysproporcja pomiędzy pojemnością kanałów wejściowych i wyjściowych w połączeniu z brakiem podziału pamięci buforowej pomiędzy obsługiwane strumienie ruchu prowadzi do zawłaszczenia pamięci buforowej i degradacji przepustowości,

➤ przypadek 4. to przykład niedopasowania wielkości zasobów transmisyjnych do potrzeb obsługiwanych procesów,

➤ przypadki 5. i 6. to przykłady niedopasowania wielkości zasobów transmisyjnych do potrzeb dostępu oraz transferu w systemie, które – w połączeniu z brakiem strategii podziału pamięci buforowej – prowadzą do degradacji przepustowości oraz ograniczania dostępu do systemu.

Przykład 6.2: Degradacja przepustowości powodowana metodą dostępu

Przykładem systemu, w którym przyczyną degradacji całkowitej przepustowości systemu jest metoda dostępu do zasobów transmisyjnych jest system z losowym dostępem do współdzielonego kanału transmisyjnego.

Klasycznym przykładem losowego dostępu do współdzielonego kanału transmisyjnego są: prosty schemat ALOHA (ang. *pure ALOHA scheme*) i szczelinowy schemat ALOHA (ang. *slotted ALOHA scheme*). Z wymienionych schematów losowego dostępu wyprowadzono kilka powszechnie stosowanych w sieciach komputerowych metod dostępu, takich jak CSMA (ang. *Carrier Sense Multiple Access*), CSMA/CD (ang. *Carrier Sense Multiple Access with Collision Detection*), CSMA/CA (ang. *Carrier Sense Multiple Access with Collision Avoidance*).

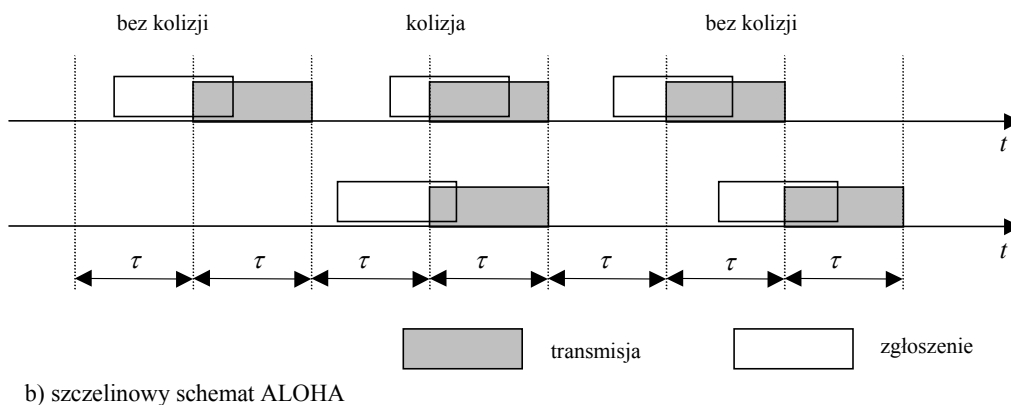
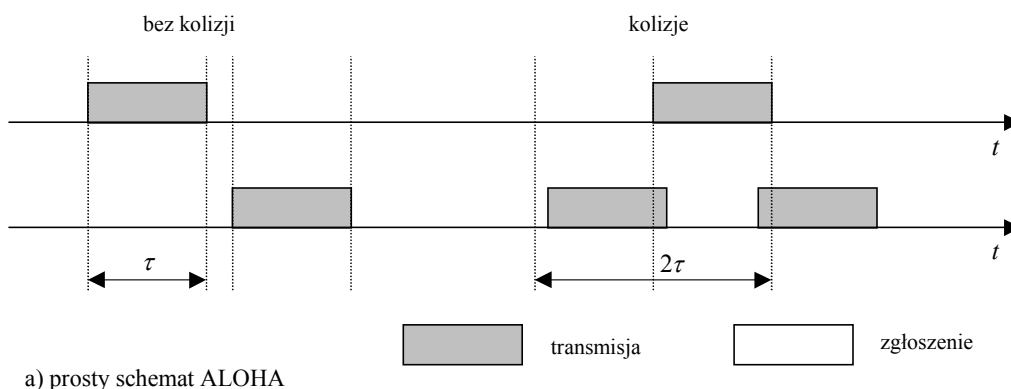
Prosty schemat ALOHA

W systemie z prostym schematem ALOHA pakiety są generowane i transmitowane przez źródła (terminale) losowo. Losowy charakter żądań dostępu do współdzielonego zasobu powoduje, że możliwe są kolizje pakietów w razie jednoczesnego wystąpienia więcej niż jednego żądania dostępu, tj. w przypadku jednoczesnego żądania dostępu przez więcej niż jedno źródło pakietów. Wystąpienie kolizji oznacza, że wszystkie pakiety w niej uczestniczące muszą być ponownie retransmitowane. W celu zmniejszenia prawdopodobieństwa ponownej kolizji żądań dostępu tych samych źródeł długości przedziałów czasu, po upływie których pakiety są retransmitowane, są one dobierane losowo przez każde ze źródeł.

Współdzielony kanał transmisyjny jest wykorzystywany przez nieskończoną liczbę źródeł generujących pakiety o stałej długości (stałym czasie trwania) τ zgodnie z rozkładem Poissona, ze średnią łączną intensywnością λ (pakietów na sekundę). Ze

względu na konieczność retransmisji wszystkich pakietów uczestniczących w kolizjach, całkowita średnia intensywność napływu pakietów, oznaczona przez $\bar{\lambda}$, jest większa od średniej intensywności napływu pakietów λ ($\bar{\lambda} > \lambda$).

Omawianą metodę dostępu charakteryzuje fakt, że nawet częściowe nałożenie się transmisji co najmniej dwóch pakietów, generowanych przez różne źródła, powoduje kolizję (rysunek 6.3a).



Rysunek 6.3. Warunki występowania kolizji w prostym i szczelinowym schemacie ALOHA

Prawdopodobieństwo braku kolizji pakietu w omawianym systemie, tzn. prawdopodobieństwo, że podczas transmisji pakietu nie dojdzie do kolizji, jest równe prawdopodobieństwu zdarzenia, że w czasie transmisji pakietu w przedziale o długości 2τ nie wystąpi żadne nowe żądanie dostępu $p_0 = \exp(-2\tau\bar{\lambda})$, wtedy prawdopodobieństwo retransmisji pakietu:

$$p_r = 1 - p_0 = 1 - \exp(-2\tau\bar{\lambda}).$$

Średnią liczbę retransmisji pojedynczego pakietu – oznaczoną przez \bar{n} – określa wyrażenie:

$$\bar{n} = 1 + p_r + p_r^2 + p_r^3 + \dots = \frac{1}{1 - p_r} = \exp(2\tau\bar{\lambda}).$$

Zależność pomiędzy średnią intensywnością napływu pakietów λ i całkowitą średnią intensywnością napływu pakietów $\bar{\lambda}$, określaną równaniami:

$$\bar{\lambda} = \bar{n}\lambda = \lambda \exp(2\tau\bar{\lambda}) \quad \text{lub} \quad \lambda = \bar{\lambda} \exp(-2\tau\bar{\lambda});$$

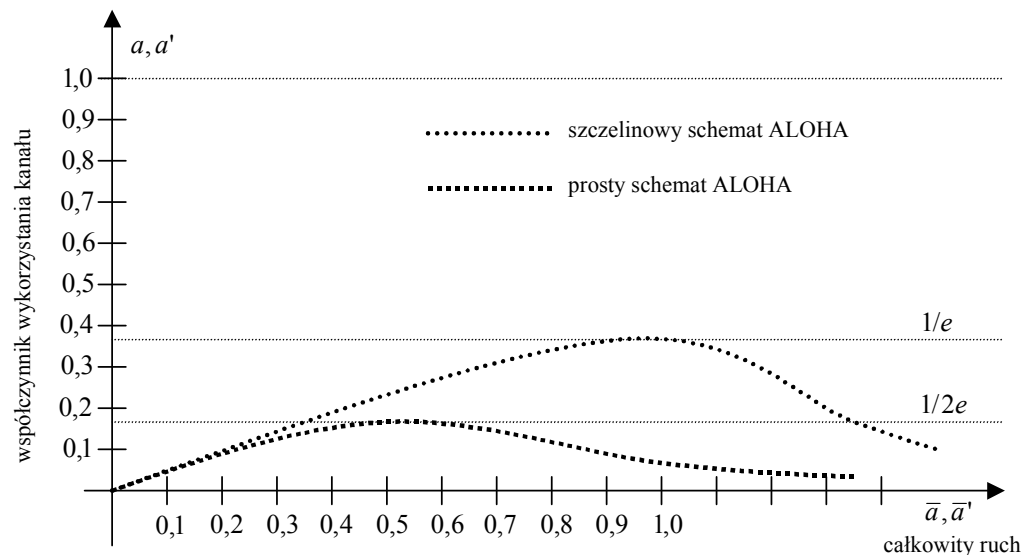
można interpretować następująco: efektywna obsługa generowanego strumienia pakietów o intensywności napływu λ wymaga obsługi całkowitego strumienia o intensywności napływu $\bar{\lambda}$.

Zależność pomiędzy obciążeniem współdzielonego kanału transmisyjnego generowanym przez wszystkie źródła (współczynnikiem wykorzystania kanału transmisyjnego), oznaczonym przez a ($a = \lambda \tau$), oraz całkowitym ruchem w kanale transmisyjnym (włącznie z retransmisjami), oznaczonym przez \bar{a} ($\bar{a} = \bar{\lambda} \tau$), określa wzór:

$$a = \bar{a} \exp(-2\bar{a}).$$

Losowa metoda dostępu powoduje, że zwiększenie nowo generowanego ruchu a jest źródłem wzrostu liczby kolizji prowadzącego do nasycenia kanału transmisyjnego.

Maksymalna wartość współczynnika wykorzystania kanału transmisyjnego dla prostego schematu ALOHA: $a_{\max} = 1/2e$ dla $\bar{a} = 1/2$ (rysunek 6.4).



Rysunek 6.4. Współczynnik wykorzystania kanału transmisyjnego dla prostego schematu ALOHA

Szczelinowy schemat ALOHA

Zwiększenie maksymalnej wartości współczynnika wykorzystania kanału transmisyjnego, uzyskiwanego w prostym schemacie ALOHA, umożliwia modyfikacja tej metody, polegająca na wprowadzeniu ograniczenia prawdopodobieństwa wystąpienia kolizji pakietów, m.in. przez koordynację chwil rozpoczynania transmisji pakietów. Jedną z takich modyfikacji jest szczelinowy schemat ALOHA. Transmisja lub retransmisja pakietów (wiadomości) o stałej długości może być rozpoczynana tylko z chwilą pojawiania się szczelin (długość szczeliny jest równa długości pakietu) (rysunek 6.3b). Wprowadzenie podziału czasu na szczeliny i wymuszanie transmisji tylko w chwilach rozpoczynania się szczelin eliminuje częściowe nakładanie się transmisji pakietów (częściowe kolizje). Takie wymuszenie chwili rozpoczynania transmisji pakietów powoduje, że zmniejsza się prawdopodobieństwo kolizji. Prawdopodobieństwo, że podczas transmisji pakietu nie będzie kolizji jest równe prawdopodobieństwu, że w czasie transmisji pakietu w przedziale o długości τ nie wystąpi żadne nowe żądanie dostępu:

$$p'_0 = \exp(-\bar{\lambda}'\tau);$$

w takim przypadku prawdopodobieństwo retransmisji pakietu:

$$p'_r = 1 - p'_0 = 1 - \exp(-\tau\bar{\lambda}').$$

Średnią liczbę retransmisji pojedynczego pakietu, oznaczoną przez \bar{n}' , określa wyrażenie:

$$\bar{n}' = 1 + p'_r + (p'_r)^2 + (p'_r)^3 + \dots = \frac{1}{1 - p'_r} = \exp(\tau\bar{\lambda}').$$

Zależność pomiędzy obciążeniem współdzielonego kanału transmisyjnego generowanego przez wszystkie źródła (współczynnikiem wykorzystania kanału transmisyjnego), oznaczonym przez a' ($a' = \lambda \tau$), oraz całkowitym ruchem w kanale transmisyjnym (włącznie z retransmisjami), oznaczonym przez \bar{a}' ($\bar{a}' = \bar{\lambda}'\tau$), określa wzór:

$$a' = \bar{a}' \exp(-\bar{a}'),$$

zgodnie z którym omawiana losowa metoda dostępu powoduje, że zwiększenie nowo generowanego ruchu a' jest źródłem wzrostu liczby kolizji prowadzącego do zwiększenia całkowitego ruchu w kanale transmisyjnym \bar{a}' , a tym samym do nasycenia kanału transmisyjnego.

Maksymalna wartość współczynnika wykorzystania kanału transmisyjnego dla szczelinowego schematu ALOHA: $a'_{\max} = 1/e$ dla $\bar{a}' = 1$ (rysunek 6.4).

Zwiększenie efektywności metod losowego dostępu wywodzących się z prostego i szczelinowego schematów ALOHA uzyskuje się po wprowadzaniu takich modyfikacji, których celem jest zmniejszanie prawdopodobieństwa wystąpienia kolizji (np.

wielodostępu z nasłuchiowaniem nośnej CSMA) lub jednocześnie prawdopodobieństwa wystąpienia kolizji i czasu trwania kolizji (np. wielodostępu z nasłuchiowaniem nośnej i detekcją kolizji CSMA/CD).

Wielodostęp z nasłuchiowaniem nośnej

Wielodostęp z nasłuchiowaniem nośnej CSMA jest metodą rywalizacyjnego (losowego) dostępu do współdzielonych zasobów, wywodzącą się z przedstawionych wcześniej schematów ALOHA.

Ograniczenie prawdopodobieństwa kolizji żądań dostępu w metodzie CSMA, w porównaniu z prawdopodobieństwem kolizji w schematach ALOHA, jest konsekwencją wyposażenia źródeł żądań dostępu w możliwość monitorowania współdzielonych zasobów transmisyjnych. Uzyskiwana w ten sposób możliwość wykrywania zajętości zasobów w chwili generowania żądania dostępu redukuje długość przedziału czasu, w którym generowane przez różne źródła żądania dostępu mogą kolidować. Zdolność do monitorowania zajętości zasobów transmisyjnych przez źródła generujące żądania dostępu powoduje, że nie do uniknięcia są kolizje tylko tych żądań dostępu, które w razie braku zajętości kanału transmisyjnego generowane są jednocześnie w odstępach czasu nie dłuższych niż czas propagacji.

Obsługa żądania dostępu generowanego w czasie, gdy monitorowany zasób znajduje się w stanie zajętości, jest odkładana (zgodnie z procedurą właściwą dla danej wersji metody) do czasu, gdy zasób jest wolny.

Stosowane są trzy odmiany metody CSMA, różniące się sposobem obsługi żądania dostępu po tym, jak obsługa tego żądania dostępu została odłożona z powodu zajętości zasobu:

- nietrwała metoda CSMA (ang. *nonpersistent CSMA*) – odłożona obsługa żądania dostępu (retransmisja pakietu) jest podejmowana natychmiast po stwierdzeniu zwolnienia zasobu,

- trwała metoda CSMA (ang. *persistent CSMA*) – odłożona obsługa żądania dostępu jest podejmowana po upływie przedziału czasu, którego długość jest realizacją zmiennej losowej o parametrach właściwych dla źródła żądania dostępu,

- *p*-trwała metoda CSMA (ang. *p-persistent CSMA*) – odłożona obsługa żądania dostępu jest, z prawdopodobieństwem p ($0 < p < 1$), podejmowana natychmiast po stwierdzeniu zwolnienia zasobu lub jest podejmowana, z prawdopodobieństwem $1 - p$, po upływie przedziału czasu, którego długość jest realizacją zmiennej losowej o parametrach właściwych dla źródła żądania dostępu.

Dla nietrwałej metody CSMA zależności pomiędzy obciążeniem współdzielonego kanału transmisyjnego generowanego przez wszystkie źródła (współczynnikiem wykorzystania kanału transmisyjnego) oraz całkowitym ruchem w kanale transmisyjnym (włącznie z retransmisjami) są określone wyrażeniami:

$$a_p = \frac{\bar{a}_p \exp(-\kappa \bar{a}_p)}{\bar{a}_p (1 + 2\kappa) + \exp(-\kappa \bar{a}_p)} \quad \text{– w wersji bez szczelin,}$$

$$a_s = \frac{\kappa \bar{a}_s \exp(-\kappa \bar{a}_s)}{(1 - \exp(-\kappa \bar{a}_s)) + \kappa} \text{ – w wersji szczelinowej,}$$

w których: κ – iloraz czasu propagacji i czasu transmisji pakietu w kanale transmisyjnym.

Maksymalna wartość rzeczywistej przepustowości kanału transmisyjnego, do którego dostęp odbywa się zgodnie z metodą CSMA, wynosi ponad 80% pojemności kanału transmisyjnego wtedy, gdy długość czasu propagacji w porównaniu z czasem transmisji pakietu jest pomijalnie mała ($\kappa \rightarrow 0$). Gdy czas propagacji jest pomijalnie mały, podane wyrażenia redukują się do postaci: $a_p = \bar{a}_p / (1 + \bar{a}_p)$ oraz $a_s = \bar{a}_s / (1 + \bar{a}_s)$.

Losowe metody dostępu do współdzielonych kanałów transmisyjnych (ogólnie zasobów komunikacyjnych) charakteryzują – z punktu widzenia wykorzystania pojemności tych zasobów – następujące właściwości:

- kanał transmisyjny może znajdować się w jednym z trzech stanów: transmisja pakietów bez kolizji, transmisja kolidujących pakietów i oczekiwanie na transmisję,

- w przypadku małych obciążeń kanał znajduje się w stanie transmisji pakietów bez kolizji lub w stanie oczekiwania na transmisję (dominujący stan) – przepustowość kanału transmisyjnego równa obciążeniu,

- w przypadku dużych obciążeń kanał znajduje się w stanie transmisji kolidujących pakietów (dominujący stan) i transmisji pakietów – wraz ze wzrostem obciążenia przepustowość maleje do zera,

- dla każdej metody losowego dostępu istnieje taka wartość obciążenia, dla której przepustowość osiąga wartość maksymalną – wartość ta wyznacza obszary stabilności i niestabilności metody dostępu,

- dla każdej wartości obciążenia istnieją dwie wartości całkowitego ruchu (można wykazać, że po zwiększeniu liczby źródeł i zachowaniu wartości obciążenia, całkowity ruch będzie się składał tylko z retransmisji); właściwość ta wskazuje na niestabilność losowych metod dostępu.

6.2.2. NIESPRAWIEDLIWY PODZIAŁ ZASOBÓW

Niesprawiedliwy (ang. *unfairness*) podział zasobów systemu komunikującym się procesom w systemie transferu danych jest konsekwencją braku sterowania współzawodnictwem źródeł strumieni ruchu (użytkowników) w ubieganiu się o ograniczone, współdzielone zasoby systemu. Niesprawiedliwy podział polega na przejęciu (przywłaszczeniu) przez niektóre (lub jedno) źródła znaczącej części lub wszystkich zasobów węzła (podsieci lub sieci) kosztem pozostałych źródeł. Skutkiem niesprawiedliwego podziału zasobów jest degradacja jakości obsługi strumieni generowanych przez dyskryminowane źródła oraz zmniejszenie efektywności wykorzystania zasobów systemu.

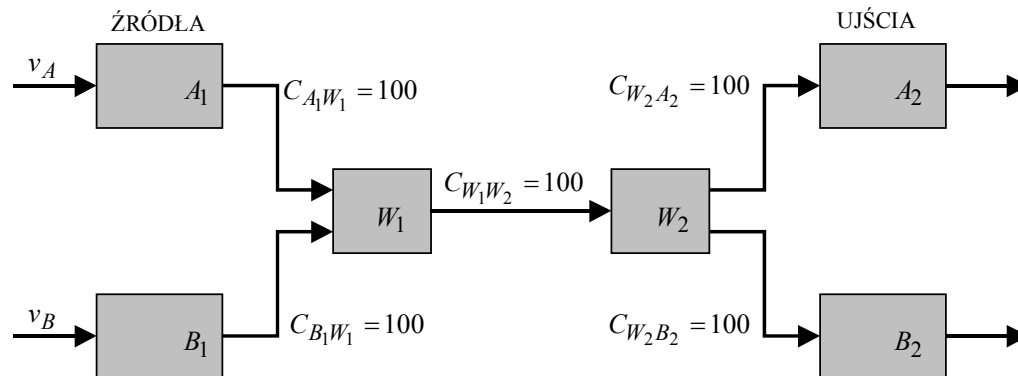
Niesprawiedliwy przydział zasobów jest konsekwencją:

- różnej, względnej pozycji, jaka charakteryzuje poszczególne, komunikujące się procesy, w zadaniach ubiegania się o współdzielone zasoby systemu,
- niewłaściwego doboru wartości parametrów systemu dla danych wartości parametrów obsługiwanych strumieni ruchu,
- niewłaściwego doboru wartości parametrów strumieni ruchu dla danych wartości parametrów systemu.

Przykład 6.3: Niesprawiedliwy podział zasobów

Transmisja pakietów pomiędzy dwiema parami źródeł i ujść, tj. (A_1, A_2) i (B_1, B_2) , odbywa się w systemie zawierającym dwa węzły: W_1 i W_2 (rysunek 6.5).

Strumienie pakietów generowane przez źródła A_1 i B_1 , odpowiednio ze średnią szybkością v_A i v_B (pakietów na sekundę) są obsługiwane przez kanały transmisyjne o pojemnościach: $C_{A_1W_1} = 100$, $C_{B_1W_1} = 100$, $C_{W_1W_2} = 100$, $C_{W_2A_2} = 100$ oraz $C_{W_2B_2} = 100$ (pakietów na sekundę), pojemność pamięci buforowych węzłów jest natomiast skończona: $C_{W_1} = K_1$ i $C_{W_2} = K_2$. Dla przyjętych założeń zasobem ograniczającym jakość obsługi strumieni pakietów jest pojemność kanału transmisyjnego pomiędzy węzłami W_1 i $W_2 - C_{W_1W_2}$.



Rysunek 6.5. Transmisja pakietów pomiędzy dwoma parami źródeł i ujść

Jeżeli pojemność pamięci buforowej węzła W_1 oraz pojemność kanału transmisyjnego $C_{W_1W_2}$ są w całości dostępne dla źródeł strumieni pakietów generowanych przez źródła A_1 i B_1 , to zajętość zasobów (pamięci buforowej i kanału transmisyjnego) przez dwa – wyróżnione w omawianym systemie – strumienie pakietów zależą od szybkości generowania pakietów w źródłach.

Przypadek 1: $v_A + v_B < C_{W_1W_2} = 100$

Suma szybkości napływu pakietów do pamięci buforowej jest mniejsza od szybkości wypływu pakietów z pamięci i zasoby systemu są dzielone pomiędzy współza-

wodniczące strumienie pakietów proporcjonalnie do szybkości generowania pakietów (podział zasobów jest podziałem sprawiedliwym). Rzeczywista przepustowość:

$$PS_r = PS_r(A_1, A_2) + PS_r(B_1, B_2) = v_A + v_B,$$

gdzie: $PS_r(A_1, A_2)$ i $PS_r(B_1, B_2)$ oznaczają – odpowiednio – rzeczywistą przepustowość tras (A_1, A_2) oraz (B_1, B_2) .

Przypadek 2: $v_A = v_B$ oraz $v_A + v_B \geq C_{W_1W_2} = 100$

Sumaryczna szybkość napływu pakietów do pamięci buforowej jest większa od szybkości wypływu pakietów z węzła W_1 , co powoduje, że pakiety są odrzucane lub retransmitowane.

Niezależnie od modelu obsługi strumieni pakietów (model ze stratami lub model z retransmisjami) pakiety generowane przez źródła mają jednakowe szanse na zaakceptowanie w pamięci buforowej i tym samym na transmisję w kanale transmisyjnym o pojemności $C_{W_1W_2}$.

Rzeczywista przepustowość omawianego systemu jest równa potencjalnej przepustowości systemu:

$$PS_r = PS_p = C_{W_1W_2} = 100,$$

natomiast rzeczywiste przepustowości tras (A_1, A_2) oraz (B_1, B_2) :

$$PS_r(A_1, A_2) = PS_r(B_1, B_2) = 0,5 C_{W_1W_2} = 50.$$

Przypadek 3: $v_A \neq v_B$ oraz $v_A + v_B \geq C_{W_1W_2} = 100$

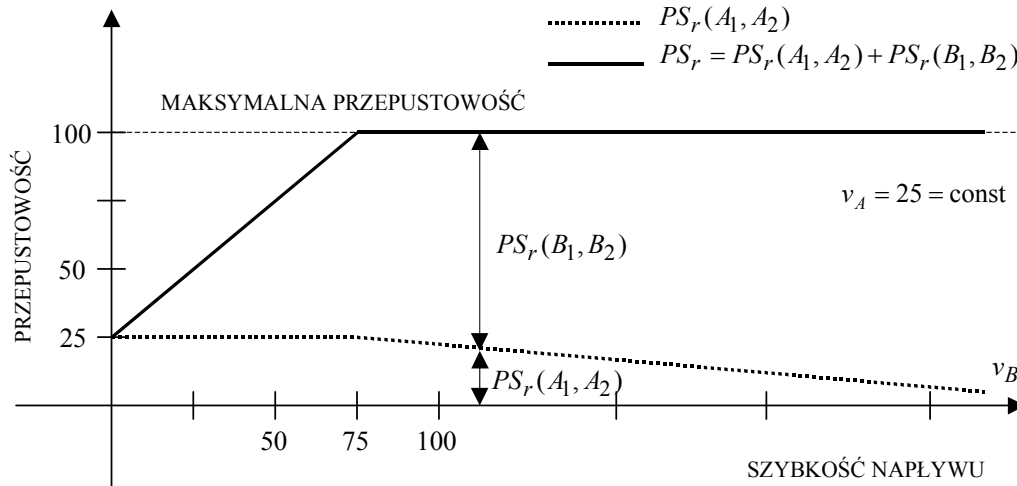
Sumaryczna szybkość napływu pakietów do pamięci buforowej jest większa od szybkości wypływu pakietów z węzła W_1 , co powoduje, że po wyczerpaniu pojemności pamięci buforowej $C_{W_1} = K_1$ pakiety są odrzucane lub retransmitowane.

Jeżeli $v_A > v_B$, to pakiety generowane przez źródło A_1 mają mniejszą szansę na zaakceptowanie w pamięci buforowej węzła niż pakiety generowane przez źródło B_1 , niezależnie od modelu obsługi strumieni pakietów mają różne szanse na zaakceptowanie w pamięci buforowej, i tym samym na transmisję w kanale transmisyjnym o pojemności $C_{W_1W_2}$. W stanie ustalonym zajętość pamięci buforowej przez pakiety generowane przez źródła A_1 i B_1 jest proporcjonalna do szybkości generowania pakietów, a rzeczywista przepustowość omawianego systemu (równa potencjalnej przepustowości systemu):

$$PS_r = PS_r(A_1, A_2) + PS_r(B_1, B_2) = 100,$$

gdzie: $PS_r(A_1, A_2) = v_A / C_{W_1W_2}$ i $PS_r(B_1, B_2) = v_B / C_{W_1W_2}$.

Dla ustalonej wartości v_A i rosnącej v_B wartość $PS_r(A_1, A_2)$ dąży do zera, a wartość $PS_r(B_1, B_2)$ dąży do $PS_r = 100$ (rysunek 6.6).



Rysunek 6.6. Rzeczywista przepustowość systemu

Brak w przypadku 3. podziału pamięci buforowej węzła W_1 na części dedykowane dla strumieni pakietów (A_1, A_2) oraz (B_1, B_2) prowadzi do zawłaszczania pojemności pamięci buforowej przez źródło o większej szybkości napływu pakietów i – w konsekwencji – do zawłaszczania pojemności kanału transmisyjnego. Powoduje to ograniczenie możliwości przesłania pakietów generowanych w źródle A_1 . Podział pamięci buforowej i pojemności kanału transmisyjnego nie jest sprawiedliwy.

Zadanie zapobiegania niesprawiedliwemu przydziałowi zasobów jest znacznie trudniejsze do rozwiązania niż zadanie przeciwdziałania degradacji całkowitej przepustowości systemu, m.in. dlatego, że jednoznaczne zdefiniowanie kryterium sprawiedliwego rozdziału zasobów w złożonym systemie komunikacji danych jest zagadnieniem bardzo trudnym, jeśli w ogóle możliwym.

6.2.3. ZAKLESZCZENIA

Zakleszczenia (zastoje) powstają w sieciach, w których generowany ruch (obciążenie sieci) jest większy od pojemności sieci. Najczęstszą przyczyną powstawania zakleszczeń (ang. *deadlock*) w sieciach jest taki przydział zasobów (pojemności pamięci buforowej) dla danych wartości parametrów ruchu i parametrów systemu transferu danych, który – w warunkach przeciążenia systemu – prowadzi do utworzenia się cyklicznych procesów oczekiwania na zwolnienie zasobów niezbędnych do zakończenia obsługi zadań transferu. Dla cykli takich charakterystyczne jest to, że będące w cyklu poszczególne, komunikujące się procesy są w posiadaniu części zasobów wymaganych do obsługi zadania komunikacji i oczekują na zwolnienie

(przez inne procesy) pozostałych, brakujących zasobów niezbędnych do zakończenia obsługi zadania.

Powstanie zakleszczenia skutkuje redukcją rzeczywistej przepustowości systemu do zera, powodując całkowity lub częściowy upadek (ang. *crash*) systemu.

Zakleszczenia są konsekwencją stosowania w sieciach pakietowych schematu obsługi polegającego na realizacji procedury, zgodnie z którą pakiet w węźle odbiorczym jest przyjmowany (zapamiętywany) i – po odpowiednim przetworzeniu – przekazywany dalej, do kolejnego węzła na trasie znanej a priori lub wybieranej dynamicznie.

Zakleszczenia, powodowane nieprawidłowym przydziałem pamięci buforowej, są możliwe tylko w sieciach, w których odrzucane pakiety są retransmitowane, tzn. w sieciach z retransmisjami (ang. *retransmission model*). W sieciach tych kopie pakietów transmitowanych z węzłów nadawczych do węzłów odbiorczych (kolejnych węzłów na trasach) są przechowywane w węźle nadawczym do chwili otrzymania potwierdzenia poprawnego odbioru pakietu. W razie odrzucenia pakietu z powodu braku możliwości jego odbioru (przepelnienie) lub przekłamania pakietu (zakłócenia) jest on retransmitowany. Nieodbieranie pakietów w węźle odbiorczym, spowodowane brakiem zasobów niezbędnych do ich odbioru, jest bezpośrednią przyczyną powstawania zakleszczeń.

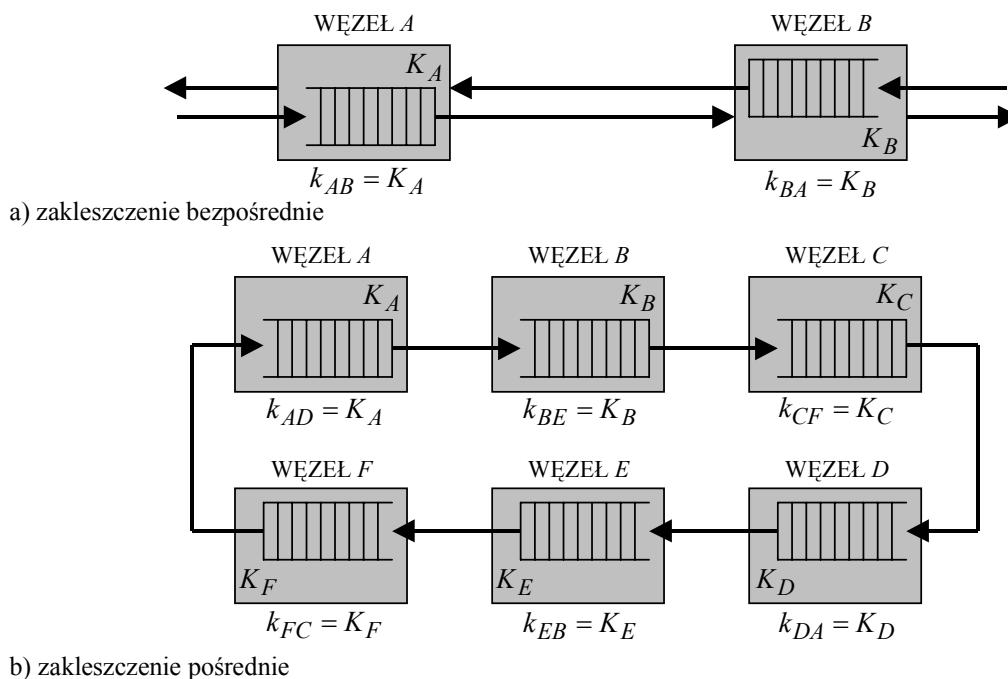
Bezpośrednią przyczyną powstawania zakleszczenia jest konieczność utrzymywania w węzłach nadawczych kopii wysyłanych pakietów do chwili uzyskania potwierdzenia ich odbioru. Ze względu na zakres (zasięg) oddziaływania, zakleszczenia dzieli się na tzw. zakleszczenia proste (bezpośrednie) (ang. *direct store-and-forward deadlock*) i tzw. zakleszczenia pośrednie (ang. *indirect store-and-forward deadlock*).

Zakleszczenia bezpośrednie to takie, których przyczyną jest stan zajętości zasobów w dwóch sąsiednich węzłach uniemożliwiający zakończenie rozpoczętej obsługi zadań (rysunek 6.7a). Zakleszczenia pośrednie to takie, których przyczyną jest sekwencja stanów zajętości zasobów w więcej niż dwóch węzłach sieci tworzących cykl uniemożliwiająca zakończenie rozpoczętej obsługi zadań (zakleszczenie bezpośrednie jest szczególnym przypadkiem zakleszczenia pośredniego) (rysunek 6.7b).

W przypadku zakleszczenia bezpośredniego pojemności pamięci buforowych węzłów A i B są w całości zawłaszczone – odpowiednio – przez pakiety transferowane z węzła A do węzła B ($k_{AB} = K_A$) oraz z węzła B do węzła A ($k_{BA} = K_B$), gdzie: k_{AB} i k_{BA} są długościami kolejek w węźle A – zawierających pakiety transferowane do węzła B oraz w węźle B – zawierających pakiety transferowane do węzła A , a K_A i K_B są pojemnościami pamięci buforowych węzłów A i B (rysunek 6.7a).

Przyczyną zakleszczenia pośredniego jest powstanie cyklu stanów nasycenia pamięci buforowych węzłów, spowodowanych zawłaszczeniem tych zasobów przez procesy, dla których węzły należące do cyklu są węzłami wejściowymi i wyjściowymi (rysunek 6.7b). W systemie tym, szczególny (symetryczny) przypadek zakleszczenia spowodowany jest zawłaszczeniem pojemności pamięci buforowych w węzłach należących do zbioru $I = \{A, B, C, D, E, F\}$ zgodnie z następującym wyrażeniem:

$$k_{AD} + k_{BE} + k_{CF} + k_{DA} + k_{EB} + k_{FC} = \sum_{i \in I} K_i.$$



Rysunek 6.7. Zakleszczenia bezpośrednie i pośrednie

Dla omawianego sześciowęzłowego podsystemu sieci zakleszczenie zachodzi po spełnieniu warunku:

$$\sum_{i,j \in \mathbb{E}, i \neq j} k_{ij} = \sum_{i \in \mathbb{E}} K_i,$$

w którym: k_{ij} – liczba pakietów (liczba zajętych buforów w pamięci buforowej) w kolejce wyjściowej i -tego węzła transferowanych do j -tego węzła.

Warto zwrócić uwagę, że powstawanie zakleszczenia w omawianym systemie jest możliwe nie tylko wtedy, gdy nastąpi zawłaszczenie całej pojemności buforowej – przypadek braku podziału pojemności pamięci buforowej. Powstanie zakleszczenia możliwe jest także wówczas, gdy nastąpi zawłaszczenie części pojemności pamięci buforowych, kierowanych do transferu pakietów w cyklu, w kierunku wyznaczonym przez sekwencję węzłów (A, B, C, D, E, F, A) – przypadek podziału pojemności pamięci buforowej. Warunkiem powstawania zakleszczenia jest jednoczesne spełnienie warunków:

$$k_{AD} = k_{A\max}, k_{BE} = k_{B\max}, k_{CF} = k_{C\max}, k_{DA} = k_{D\max}, k_{EB} = k_{E\max} \text{ oraz } k_{FC} = k_{F\max},$$

gdzie: $k_{i\max}$ – maksymalna pojemność części pamięci buforowej węzła i ($i \in I$) kierowanej do transferu pakietów na trasie wyznaczonej cyklem.

Zapobieganie powstawaniu zakleszczeń pośrednich w dowolnym systemie transferu danych wymaga takiego podziału zasobów i sterowania przepływem, który uniemożliwia powstanie dowolnego cyklu spełniającego podane warunki. Liczba wszystkich możliwych cykli o długości l ($l = 1, 2, \dots, N$) w danym systemie zależy od liczby wszystkich węzłów systemu N oraz struktury topologicznej połączeń pomiędzy węzłami opisywanej grafem. Gdy każdy węzeł systemu jest bezpośrednio połączony ze wszystkimi pozostałymi węzłami systemu (graf pełny), liczba wszystkich możliwych cykli N_c :

$$N_c = \sum_{l=2}^N N_l = \sum_{l=2}^N C_l^N,$$

gdzie: $C_l^N = N!/(N-l)!$.

Pośród wszystkich N_c cykli $N_2 = N(N-1)$ to cykle, w których możliwe jest powstanie zastoju bezpośrednich, w pozostałych natomiast – pośrednich.

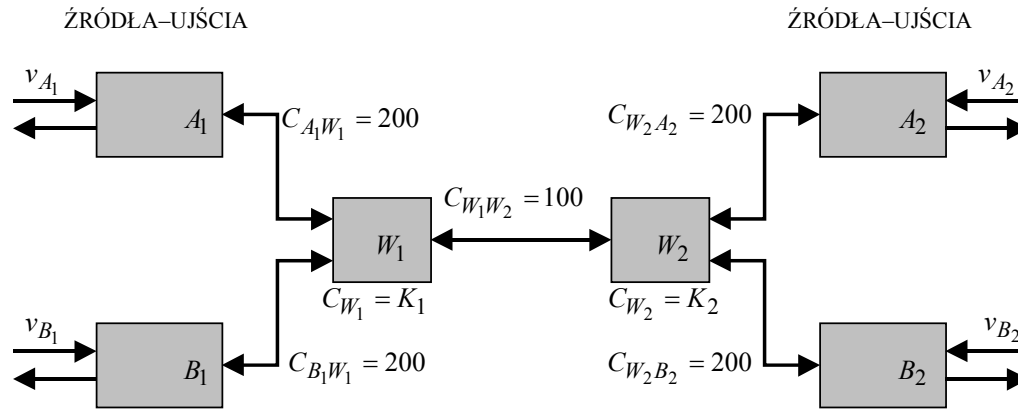
Liczba możliwych cykli w danym systemie transferu danych zwielokrotniona przez liczbę możliwych kombinacji zajętości zasobów w cyklu, powodujących powstanie zakleszczenia, pokazuje złożoność zadania zapobiegania zakleszczeniom bezpośrednim i pośrednim.

Zapobieganie zakleszczeniom bezpośrednim jest zadaniem względnie prostym, polega na odpowiednim podziale pojemności pamięci buforowych dwóch sąsiednich węzłów na części dedykowane dwóm kierunkom transferu pomiędzy tymi węzłami – zadanie podziału jest zadaniem lokalnym. Zapobieganie zakleszczeniom pośrednim jest, w porównaniu z zadaniami zapobiegania zakleszczeniom bezpośrednim, znacznie bardziej złożone, co wynika z tego, że – po pierwsze – nie każdy podział pojemności pamięci buforowej zapobiega powstawaniu zakleszczeniu pośredniego, i – po drugie – zapobiegający zakleszczeniom podział pojemności pamięci buforowych węzłów na części dedykowane poszczególnym kierunkom transferu musi być rozwiązaniem globalnym.

W sieciach z odrzuceniami (ang. *loss model*) – w przeciwieństwie do sieci z retransmisjami – w węzłach nadawczych, transmitujących pakiety, kopie wysyłanych pakietów nie są przechowywane. Sieci z odrzuceniami są wolne od zastoju bezpośrednich i pośrednich. Brak kopii pakietów w węzłach nadawczych oznacza, że w razie niemożności odbioru pakietu w węzle odbiorczym lub wystąpienia niekorygowalnych błędów podczas jego transmisji jest on stracony. Możliwość straty pakietu w sieci wymaga implementacji – w procesach końcowych, nadrzędnych nad procesami transferu pakietów w sieci – mechanizmów wykrywania strat pakietów, retransmisji straconych oraz – w przypadkach transmisji wiadomości – mechanizmów odtwarzania wiadomości w węzłach końcowych tras.

Przykład 6.4: Zakleszczenie bezpośrednie powodowane ograniczeniem pojemności kanału transmisyjnego

Przykładem ilustrującym warunki oraz konsekwencje powstania zakleszczenia w wyniku nieprawidłowego doboru zasobów komunikacyjnych jest system przedstawiony na rysunku 6.8.



Rysunek 6.8. Warunki i konsekwencje powstania zakleszczenia bezpośredniego

Transmisja pakietów pomiędzy czterema parami źródeł i ujść, tj. (A_1, A_2) , (A_2, A_1) , (B_1, B_2) i (B_2, B_1) , odbywa się w systemie zawierającym dwa węzły: W_1 i W_2 , które są połączone dwukierunkowym kanałem transmisyjnym o pojemności $C_{W_1W_2} = 100$ (rysunek 6.8). Strumienie pakietów generowane przez źródła A_1 , A_2 , B_1 i B_2 , oznaczone przez v_{A_1} , v_{A_2} , v_{B_1} oraz v_{B_2} (pakietów na sekundę), są obsługiwane przez dwukierunkowe kanały transmisyjne o pojemnościach: $C_{A_1W_1} = 100$, $C_{B_1W_1} = 100$, $C_{W_2A_2} = 100$ i $C_{W_2B_2} = 100$ (pakietów na sekundę). Pojemności pamięci buforowych węzłów są skończone: $C_{W_1} = K_1$ i $C_{W_2} = K_2$ i dobrane tak, że nie wnoszą ograniczeń dla przekazywanych strumieni pakietów.

Zasobem, który powoduje powstanie zakleszczenia w omawianym systemie, jest pojemność dwukierunkowego kanału transmisyjnego $C_{W_1W_2}$; potencjalna przepustowość systemu $PS = C_{W_1W_2} = 100$.

Przypadek 1: $v_{A_1} + v_{A_2} + v_{B_1} + v_{B_2} < C_{W_1W_2} = 100$

Jeżeli suma szybkości napływu pakietów do pamięci buforowych dwóch węzłów jest zawsze mniejsza od pojemności kanału transmisyjnego pomiędzy tymi węzłami, to oznacza, że suma ta jest zawsze mniejsza od sumy szybkości wypływu pakietów z tych węzłów. Rzeczywista przepustowość systemu:

$$PS_r = v_{A_1} + v_{A_2} + v_{B_1} + v_{B_2}.$$

Przypadek 2: $v_{A_1} + v_{B_1} + v_{A_2} + v_{B_2} > C_{W_1W_2} = 100$

Suma szybkości napływu pakietów do pamięci buforowych dwóch węzłów systemu jest większa od sumy szybkości wypływu pakietów z tych węzłów.

Po przyjęciu, że szybkości napływu pakietów do węzłów spełniają warunek $v_{A_1} < v_{B_1} < v_{A_2} < v_{B_2}$, suma szybkości napływu pakietów do węzła W_1 :

$$\min(v_{A_1}, C_{A_1W_1}) + \min(v_{B_1}, C_{B_1W_1})$$

oraz do węzła W_2 :

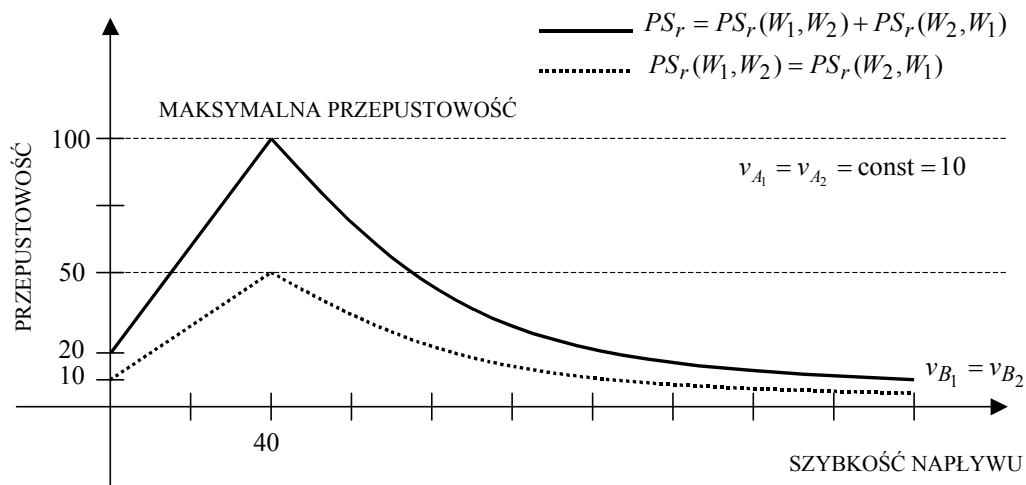
$$\min(v_{A_2}, C_{A_2W_2}) + \min(v_{B_2}, C_{B_2W_2})$$

jest większa od sumy szybkości wypływu pakietów z tych węzłów. Dodatnia różnica szybkości napływu i wypływu pakietów w pamięciach buforowych węzłów powoduje, że z upływem czasu pamięć buforowa węzła W_1 jest zawłaszczana przez źródło B_1 ($v_{B_1} > v_{A_1}$), a pamięć buforowa węzła W_2 jest z kolei zawłaszczana przez źródło B_2 ($v_{B_2} > v_{A_2}$).

Zawłaszczenie pamięci buforowej węzła W_1 przez źródło B_1 powoduje, że w węźle W_1 – z powodu braku wolnej pamięci buforowej – nie są akceptowane pakiety wysyłane z węzła W_2 . Zwolnienie pamięci buforowej w węźle W_1 jest możliwe tylko wtedy, gdy pakiety oczekujące w W_1 będą przyjmowane w węźle W_2 . Z kolei w węźle W_2 pakiety wysyłane z W_1 nie są akceptowane, bowiem pamięć buforowa węzła W_2 jest zawłaszczona przez źródło B_2 .

W omawianym przypadku rzeczywista przepustowość systemu dąży do zera.

Przykładową zależność rzeczywistej przepustowości systemu od szybkości napływu pakietów przedstawiono na rysunku 6.9, przyjmując symetrię szybkości generowanych strumieni pakietów: $v_{A_1} = v_{A_2} = \text{const} = 10$ oraz $v_{B_1} = v_{B_2}$.



Rysunek 6.9. Zależność rzeczywistej przepustowości systemu od szybkości napływu pakietów

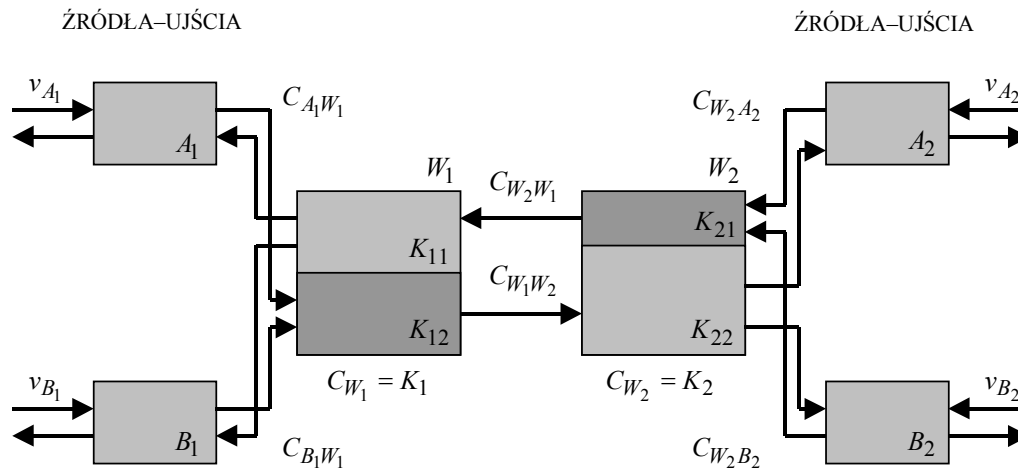
Zapobieganie powstawaniu zastoju w omawianym systemie polega na podziale pamięci buforowej w węzłach W_1 i W_2 , odpowiednio o pojemności K_1 i K_2 , na części dedykowane kierunkom transmisji:

➤ w pamięci buforowej węzła W_1 (węzeł wejściowy dla źródeł A_1 i B_1 i węzeł wyjściowy dla źródeł A_2 i B_2): K_{12} – dla pakietów przekazywanych ze źródeł A_1 i B_1 do węzła W_2 (do ujść A_2 i B_2) oraz K_{11} – dla pakietów transferowanych z węzła W_2 (ze źródeł A_2 i B_2) do ujść A_1 i B_1 ($K_{11} + K_{12} = K_1$),

➤ w pamięci buforowej węzła W_2 (węzeł wejściowy dla źródeł A_2 i B_2 i węzeł wyjściowy dla źródeł A_1 i B_1): K_{21} – dla pakietów transferowanych ze źródeł A_2 i B_2 do węzła W_1 (do ujść A_1 i B_1) oraz K_{22} – dla pakietów przekazywanych z węzła W_1 (ze źródeł A_1 i B_1) do ujść A_2 i B_2 ($K_{21} + K_{22} = K_2$),

oraz zastąpieniu dwukierunkowego kanału transmisyjnego pomiędzy węzłami W_1 i W_2 dwoma skierowanymi (jednokierunkowymi) kanałami transmisyjnymi:

- z węzła W_1 do węzła W_2 o pojemności $C_{W_1W_2}$,
- z węzła W_2 do węzła W_1 o pojemności $C_{W_2W_1}$ (rysunek 6.10).



Rysunek 6.10. Zapobieganie powstawaniu zakleszczenia poprzez podział pamięci buforowej oraz zwielokrotnienie kanału transmisyjnego

Dodatkowymi warunkami, których spełnienie daje gwarancję, że podzielone na części i dedykowane przeciwnym kierunkom transferu pakietów pojemności pamięci buforowych węzłów nie będą, w omawianym przypadku, źródłem zakleszczeń, są następujące:

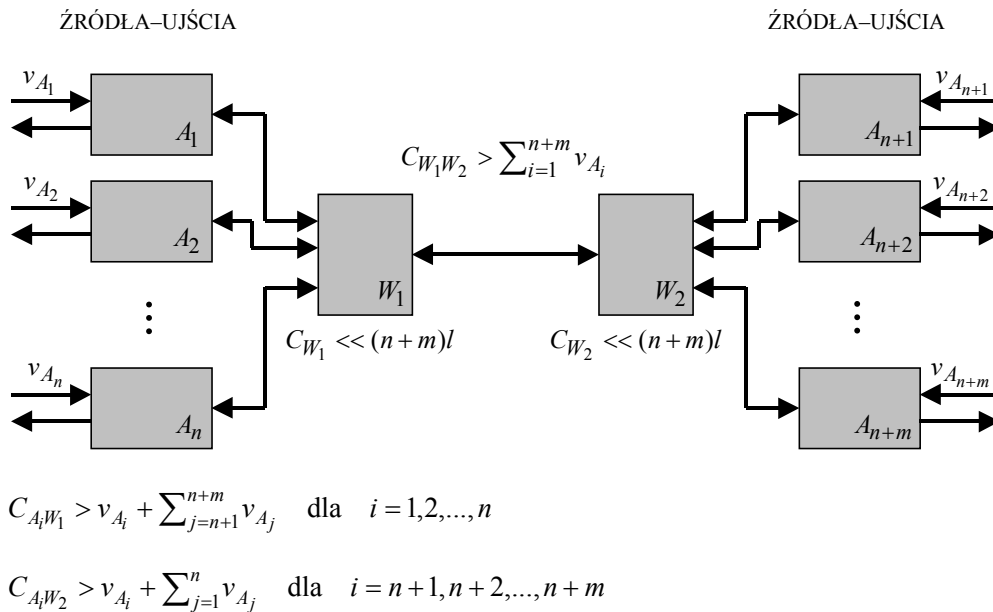
$$K_{12} \leq K_{22} \quad \text{oraz} \quad K_{21} \leq K_{11}.$$

Spełnienie tych warunków oznacza, że każdy pakiet wprowadzony do węzła źródłowego (węzeł W_1 jest węzłem wejściowym dla źródeł) będzie zawsze przyjęty przez węzeł wyjściowy.

Przykład 6.5: Zakleszczenie bezpośrednio powodowane ograniczoną pojemnością pamięci buforowej

Drugą, oprócz nieprawidłowego doboru lub przydziału zasobów transmisyjnych, przyczyną powstawania zakleszczeń w systemach transferu danych jest nieprawidłowy dobór lub przydział pojemności pamięci buforowych w węzłach systemu.

Przykładem ilustrującym warunki oraz konsekwencje powstania zakleszczenia w wyniku nieprawidłowego doboru całkowitej pojemności pamięci buforowej jest system przedstawiony na rysunku 6.11.



Rysunek 6.11. Warunki oraz konsekwencje powstania zakleszczenia w wyniku nieprawidłowego doboru całkowitej pojemności pamięci buforowej

Transmisja pakietów pomiędzy $n+n$ parami źródeł i ujęć odbywa się w systemie transferu danych zawierającym dwa węzły: W_1 i W_2 , które są połączone dwukierunkowym kanałem transmisyjnym o pojemności $C_{W_1W_2}$ (rysunek 6.11).

Strumienie pakietów generowane w źródłach A_i dla $i = 1, 2, \dots, n$ są przekazywane do węzła W_2 poprzez węzeł W_1 , natomiast strumienie generowane w źródłach A_i dla $i = n+1, n+2, \dots, n+m$ – do węzła W_1 za pośrednictwem węzła W_2 . Strumienie pa-

kietów generowane przez źródła A_i ($i = 1, 2, \dots, n-1, n, n+1, \dots, n+m$), odpowiednio ze średnią szybkością v_{A_i} (pakietów na sekundę), są obsługiwane przez dwukierunkowe kanały transmisyjne o pojemnościach $C_{A_i W_1}$, dla $i = 1, 2, \dots, n$, oraz $C_{A_i W_2}$, dla $i = n+1, n+2, \dots, n+m$.

Przyjęcie założeń, że pojemności dwukierunkowych kanałów transmisyjnych z dwóch grup wyróżnionych źródeł do dwóch węzłów spełniają warunki:

$$C_{A_i W_1} > v_{A_i} + \sum_{j=n+1}^{n+m} v_{A_j W_2} \quad \text{dla } i = 1, 2, \dots, n,$$

$$C_{A_i W_2} > v_{A_i} + \sum_{j=1}^n v_{A_j W_1} \quad \text{dla } i = n+1, n+2, \dots, n+m,$$

$$C_{W_1 W_2} > \sum_{i=1}^{n+m} v_{A_i},$$

jest tożsame z przyjęciem założenia, że zarówno pojemności kanałów transmisyjnych pomiędzy źródłami a węzłami, jak i pojemność dwukierunkowego kanału pomiędzy węzłami nie są zasobami krytycznymi systemu; nie powodują pogorszenia przepustowości systemu lub zakleszczeń.

Na potrzeby przykładu przyjęto kolejne założenie, że obsługa transferowanych pakietów odbywa się w postaci wiadomości o jednakowej długości l (pakietów). Obsługa transferu w postaci wiadomości oznacza, że pakiety wysyłane przez źródło są przekazywane do pamięci buforowych węzłów wyjściowych (węzeł W_2 dla źródeł A_i , gdzie $i = 1, 2, \dots, n$, oraz węzeł W_1 dla źródeł A_i , gdzie $i = n+1, n+2, \dots, n+m$), w których wiadomości są kompletowane, a następnie przekazywane do odpowiednich ujść.

Przyjęcie założenia, że pojemności pamięci buforowych węzłów, współdzielone pomiędzy wszystkie aktywne źródła strumieni pakietów, spełniają warunki: $C_{W_1} \ll (n+m)l$ oraz $C_{W_2} \ll (n+m)l$ oznacza niezerowe prawdopodobieństwo zdarzenia, że pojemności pamięci buforowych węzłów W_1 i W_2 są całkowicie zajęte przez pakiety należące do wiadomości, których skompletowanie nie jest możliwe ze względu na brak możliwości przyjęcia nowych pakietów. Spełnienie przez pojemności pamięci buforowych warunków: $C_{W_1} = (n+m)l$ oraz $C_{W_2} = (n+m)l$ oznacza, że w najmniej korzystnym przypadku, tzn. gdy wszystkie źródła jednocześnie generują pakiety, pamięci buforowe nie będą zasobem ograniczającym funkcjonalność omawianego systemu. Niemożność skompletowania (odtworzenia) wiadomości, spowodowanej brakiem wolnej pojemności pamięci buforowych, oznacza powstanie tzw. zakleszczenia odtwarzania (ang. *assembly deadlock*) i całkowitą degradację przepustowości, tzn. sytuację, kiedy rzeczywista przepustowość systemu jest równa zero.

Brak możliwości przyjęcia nowych pakietów (w węźle W_1 z węzła W_2 oraz w węźle W_2 z węzła W_1) może mieć, w omawianym przykładzie, dwie przyczyny:

➤ w razie zawłaszczenia pamięci buforowej węzła W_1 (lub W_2) przez strumienie pakietów wychodzących w danym węźle potrzebna do skompletowania co najmniej

jednej wiadomości pojemność pamięci buforowej jest zajęta przez pakiety będące składnikami innych wiadomości oczekujących w węźle W_1 (lub W_2) na skompletowanie,

➤ w przypadku niezawłaszczenia pamięci buforowej węzła W_1 (lub W_2) przez strumienie pakietów wychodzących w danym węźle potrzebna do skompletowania co najmniej jednej wiadomości pojemność pamięci buforowej węzła W_1 (lub W_2) jest zajęta przez pakiety oczekujące w węźle W_1 (lub W_2) na transfer do węzła W_2 (lub W_1).

Wystąpienie pierwszej z wymienionych przyczyn oznacza powstanie zakleszczenia jednokierunkowego, drugiej natomiast – powstanie zakleszczenia dwukierunkowego.

Rozwiązaniem zapobiegającym powstawaniu zakleszczeń jednokierunkowych (w omawianym przykładzie) jest zwiększenie pojemności pamięci buforowej do wartości gwarantującej brak wystąpienia konfliktów dostępu do tych zasobów (rozwiązanie nadmiarowe) i (lub) podział pojemności pamięci buforowych węzłów na części dedykowane poszczególnym kierunkom transferu pakietów.

Rozwiązaniem zapobiegającym powstawaniu zakleszczeń dwukierunkowych jest odpowiednie zwiększenie pojemności pamięci buforowych węzłów (rozwiązanie nadmiarowe) i (lub) podział pojemności pamięci buforowych węzłów na części kierowane do poszczególnych par źródło–ujście.

Szczególnymi przypadkami zakleszczeń bezpośrednich, powodowanych ograniczoną wielkością lub nieprawidłowym podziałem pojemności pamięci buforowej, są tzw. zakleszczenia kompletacji (ang. *reassembly deadlock*) i zakleszczenia sekwencji (ang. *sequence deadlock*).

Przykład 6.6: Zakleszczenie kompletacji

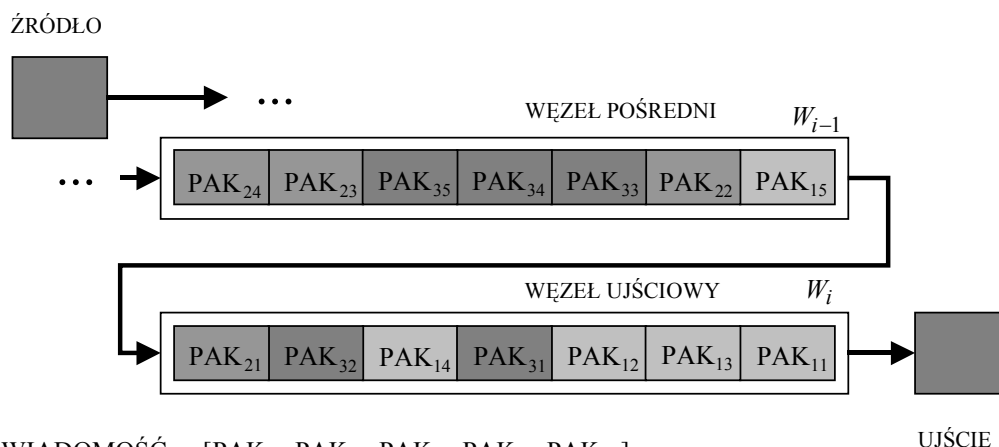
Zakleszczenie kompletacji powstaje w sieciach, w których pakiety przekazywane w sieci, pracującej w trybie datagramowym lub w trybie połączeń wirtualnych, mogą być wyprowadzane z sieci tylko i wyłącznie w postaci wiadomości, tj. uporządkowanych zbiorów pakietów. Rozpoczęcie wyprowadzania wiadomości jest możliwe tylko wtedy, gdy w pamięci buforowej węzła ujściowego znajdują się wszystkie pakiety wiadomości.

Powodem powstania zakleszczenia kompletacji jest taki stan zajętości pojemności pamięci buforowej, kiedy brak jest możliwości przyjęcia do pamięci buforowej węzła ujściowego pakietów niezbędnych do skompletowania co najmniej jednej wiadomości i rozpoczęcia jej wyprowadzania, a tym samym do rozpoczęcia zwalniania buforów pamięci buforowej.

Warunkiem koniecznym zapobiegania powstawaniu zakleszczenia kompletacji w sieci zawierającej n węzłów jest pojemność pamięci buforowych węzłów większa od maksymalnej długości wiadomości obsługiwanych w sieci l_{\max} , tzn. $C_{W_i} > l_{\max}$ dla $i = 1, 2, \dots, n$. Niespełnienie tego warunku oznacza, że w niektórych lub wszystkich węzłach sieci niemożliwe jest skompletowanie niektórych lub wszystkich wiadomości.

Spełnienie podanego warunku koniecznego nie zapobiega możliwości powstawania zakleszczeń kompletacji. W przypadku sieci z datagramowym trybem obsługi, ilustrowanym na rysunku 6.12, przyjęto – bez straty ogólności rozważań – że pomiędzy węzłem źródłem a węzłem ujściowym W_i o pojemności pamięci buforowej

$C_{W_i} = 7$ przekazywane są trzy różne wiadomości o różnej długości: $l_1 = 5$, $l_2 = 4$ i $l_3 = 5$ ($l_{\max} = 5$).



WIADOMOŚĆ₁ = [PAK₁₁, PAK₁₂, PAK₁₃, PAK₁₄, PAK₁₅]

WIADOMOŚĆ₂ = [PAK₂₁, PAK₂₂, PAK₂₃, PAK₂₄]

WIADOMOŚĆ₃ = [PAK₃₁, PAK₃₂, PAK₃₃, PAK₃₄, PAK₃₅]

Rysunek 6.12. Warunki powstania i konsekwencje zakleszczenia kompletacji

W sieci z datagramowym trybem obsługi możliwe jest, że – w wyniku współzawodnictwa o zasoby sieci na trasach pomiędzy węzłami źródłowymi a węzłem ujściowym – pakiety należące do tych samych wiadomości napływają do węzła ujściowego w kolejności innej od tej, w jakiej zostały wysłane z węzłów źródłowych (np. pakiety wiadomości pierwszej – PAK₁₂ i PAK₁₃). Kolejność napływu pakietów poszczególnych wiadomości nie ma wpływu na powstawanie zakleszczenia kompletacji.

Zarówno w sieci z datagramowym trybem obsługi, jak i w sieci z połączeniami wirtualnymi możliwe jest, że – ze względu na różne czasy napływu pakietów do węzła ujściowego – w czasie oddzielającym napływ kolejnych pakietów należących do tych samych wiadomości napływają pakiety należące do innych wiadomości (pakiety należące do tych samych wiadomości są „przedzielane” pakietami należącymi do innych wiadomości: np. pakiety PAK₁₂ i PAK₁₄ i pierwszej wiadomości są „przedzielone” pakietem PAK₃₁, należącym do trzeciej wiadomości). Dopuszczenie do możliwości napływu do pamięci buforowej pakietów należących do różnych wiadomości może doprowadzić do powstania sytuacji (jak na rysunku 6.12), że ograniczona pojemność pamięci buforowej węzła ujściowego (spełniająca warunek konieczny zapobiegania zakleszczeniu kompletacji, tj. $C_{W_i} = 7$, $C_{W_i} > l_{\max}$) jest zajęta w całości przez pakiety należące do trzech różnych wiadomości, z tym że są to tylko niektóre z pakietów tych wiadomości. W tej sytuacji żadna z trzech wiadomości nie może być skompletowana w pamięci buforowej węzła ujściowego, a tym samym nie może być rozpoczęte wy-

prowadzanie żadnej z tych wiadomości z tego węzła; powstaje zakleszczenie kompletacji.

Najprostszym sposobem przeciwdziałania powstawaniu zakleszczeń kompletacji jest:

- albo jednorazowe zarezerwowanie dla kolejno obsługiwanych wiadomości liczby buforów równej długościom tych wiadomości,
- albo sukcesywne zajmowanie zwalnianych buforów pamięci buforowej przez pakiety należące do wiadomości, której obsługa została rozpoczęta, i odrzucanie pakietów należących do innych wiadomości.

Rozpoczęcie obsługi jednej z wiadomości w chwili odebrania dowolnego jednego z pakietów tej wiadomości oznacza, że do zarezerwowanej części pojemności pamięci buforowej mogą być przyjmowane tylko pakiety należące do obsługiwanej wiadomości.

Po przyjęciu takiego sposobu obsługi odebranie pakietu PAK_{11} wiadomości pierwszej w omawianym przykładzie oznaczałoby:

- zarezerwowanie w pamięci buforowej $l_1 - 1$ buforów dla pozostałych pakietów pierwszej wiadomości, jeżeli pojemność dostępnej pamięci buforowej jest większa lub równa długości wiadomości pomniejszonej o odebrany pakiet, tj. $l_1 - 1$,
- zarezerwowanie w pamięci buforowej wszystkich wolnych buforów dla kolejnych pakietów pierwszej wiadomości, jeżeli pojemność dostępnej pamięci buforowej jest mniejsza od długości wiadomości pomniejszonej o odebrany pakiet, tj. $l_1 - 1$. Po wyczerpaniu zarezerwowanej puli buforów przyjmowanie kolejno nadchodzących pakietów wiadomości pierwszej następuje sukcesywnie wraz ze zwalnianiem buforów przez wcześniej obsługiwaną wiadomość.

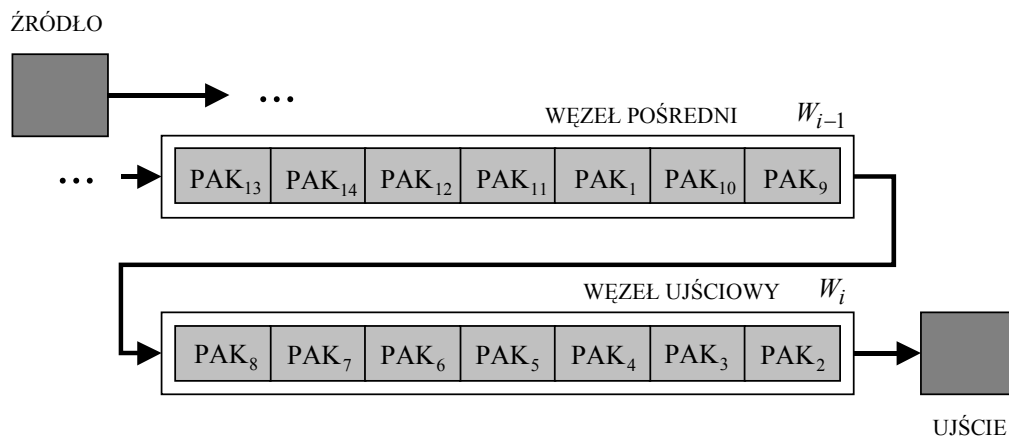
Przykład 6.7: Zakleszczenie sekwencji

Zakleszczenie sekwencji (kolejności) powstaje w sieciach, w których pakiety przekazywane w sieci, pracującej w trybie datagramowym lub w trybie połączeń wirtualnych, mogą być wyprowadzane z sieci tylko i wyłącznie w kolejności, w jakiej zostały do niej wprowadzone.

Możliwość zastoju sekwencji występuje w sieciach pakietowych z datagramowym trybem obsługi. Ze względu na to, że w tym trybie obsługi poszczególne pakiety, kolejno wprowadzane do sieci w węźle źródłowym, mogą być przekazywane do węzła ujściowego różnymi trasami, czasy przejścia poszczególnych pakietów mogą się znacznie różnić. Różnice opóźnień poszczególnych pakietów wnoszone przez sieć mogą powodować, że kolejność napływu pakietów do węzła ujściowego może się różnić od kolejności ich nadawania.

Brak sterowania przepływem, powodujący, że wszystkie bufory pamięci buforowej węzła ujściowego są zajęte przez pakiety, z których niemożliwe jest odtworzenie sekwencji, w jakiej zostały wprowadzone do sieci (rysunek 6.13), powoduje powstanie zastoju sekwencji. W omawianym przykładzie wszystkie bufory pamięci buforowej węzła ujściowego W_i są zajęte przez pakiety, wśród których nie ma pakietu PAK_1 .

Żaden pakiet znajdujący się w pamięci buforowej nie może być wyprowadzony z sieci, tzn. nie ma możliwości zwolnienia bufora niezbędnego do przyjęcia pakietu PAK_1 , którego wyprowadzenie umożliwi wyprowadzanie kolejnych pakietów wiadomości.



Rysunek 6.13. Warunki i konsekwencje powstania zakleszczenia sekwencji

W sieciach pakietowych z połączeniami wirtualnymi możliwość zakleszczenia sekwencji nie występuje wtedy, gdy procedura sterowania przepływem, implementowana w każdym łączu danych należącym do trasy danego połączenia wirtualnego, gwarantuje w węźle ujściowym łączu danych kolejność odbioru zgodną z kolejnością nadawania w węźle źródłowym łączu danych. W szczególności więc gwarantuje zachowanie kolejności pakietów w węźle ujściowym ostatniego łączu danych trasy połączenia wirtualnego, tj. w węźle ujściowym połączenia wirtualnego.

Zadanie zachowania kolejności pakietów jest istotne w rozwiązaniach sieci pakietowych z połączeniami wirtualnymi, w których sterowanie przepływem w poszczególnych łączach danych nie gwarantuje zachowania zgodności kolejności odbioru z kolejnością nadawania pakietów (zaburzenie kolejności pakietów może być np. skutkiem stosowania procedur selektywnej retransmisji pakietów uszkodzonych w czasie przekazywania w łączach danych sieci). W takich przypadkach zadanie zachowania kolejności pakietów jest zadaniem procedur sterowania przepływem implementowanych pomiędzy węzłami źródłowym i ujściowym połączenia wirtualnego.

Najprostszym sposobem zapobiegania zakleszczeniom sekwencji jest odrzucanie pakietów nadchodzących do węzła w kolejności innej od kolejności nadawania. W omawianym przypadku, ilustrowanym na rysunku 6.13, realizacja tego sposobu sterowania przepływem oznaczałaby, że wszystkie pakiety napływające do węzła ujściowego W_i i poprzedzające nadejście pakietu PAK_1 byłyby odrzucone. Taki sposób zapobiegania zakleszczeniom sekwencji wiąże się z dużym, możliwym marnotrawieniem zasobów sieci. Marnotrawienie zasobów zwiększa się wraz ze wzrostem długości sekwencji pakietów jednorazowo wprowadzanych do sieci.

6.3. FUNKCJE STEROWANIA PRZEPIYWEM

Procedury sterowania przepływem są definiowane jako protokoły (zbiory protokołów), których celem jest zabezpieczenie systemu przed konsekwencjami przeciążenia zasobów systemu lub braku dopasowania szybkości transferu (szybkości wprowadzania i szybkości wyprowadzania) w wybranych podsystemach systemu.

Poprawnie zaprojektowane i wprowadzone procedury sterowania przepływem mają na celu zachowanie efektywności systemu, sprawiedliwy przydział zasobów dla obsługiwanych procesów (przepływów) oraz zapobieganie zakleszczeniom (upadkom) systemu. Źle zaprojektowane i wprowadzone procedury sterowania przepływem mogą być dodatkowym źródłem degradacji jakości usług systemu. Wdrożenie procedur sterowania sprowadza się do zaprojektowania i wdrożenia zasad, zgodnie z którymi alokowane są pojemności pamięci buforowych oraz pojemności kanałów transmisyjnych. Alokacja pojemności kanałów transmisyjnych jest tożsama z wprowadzaniem ograniczeń (jeżeli są zasadne) szybkości transferu do systemu (w węzłach dostępowych systemu) oraz wewnątrz systemu (pomiędzy węzłami systemu). Wdrożenie procedur sterowania przepływem prowadzi do zmniejszenia efektywności systemu (w porównaniu z teoretycznie możliwą) i skutkuje zwiększeniem opóźnień oraz redukcją przepustowości. Przyrost opóźnienia (średniego lub maksymalnego) oraz stopień redukcji przepustowości są najczęściej stosowanymi miarami kosztu implementacji metod sterowania przepływem.

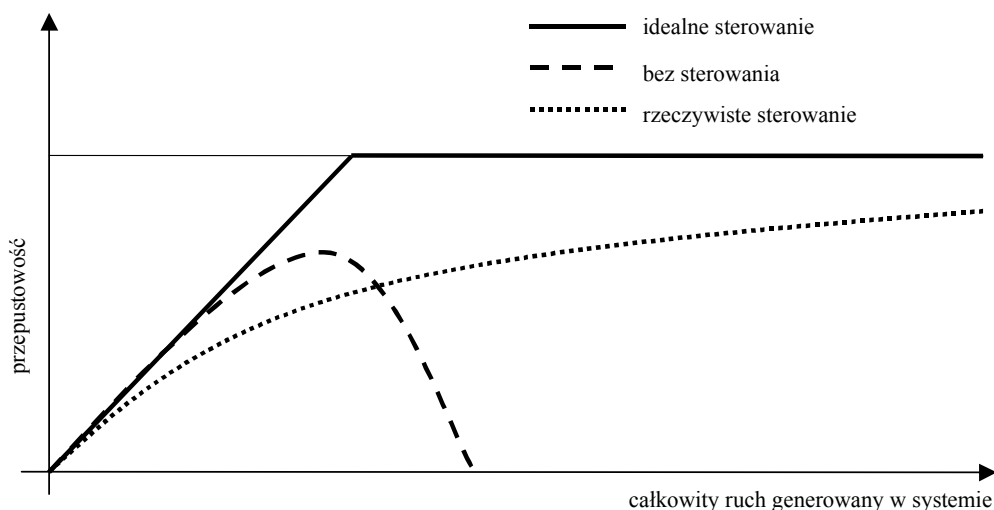
Osiągnięcie celów sterowania przepływem jest związane z koniecznością przeznaczenia części zasobów systemu na potrzeby obsługi procedur sterowania przepływem. Efektywność implementowanych procedur sterowania przepływem wymaga przede wszystkim:

- wymiany informacji o stanie systemu i jego elementów pomiędzy elementami systemu (najczęściej węzłami) niezbędnej do wyboru sterowania i (lub) parametrów sterowania,
- wymiany komend i odpowiedzi na komendy (zawierających wartości parametrów procedur sterowania i uzgodnień) niezbędnych do implementacji wcześniej ustalonego sterowania.

Implementacja procedur sterowania przepływem w systemach teleinformatycznych, opartych na technikach komutacji jednostek danych, w zależności od celów sterowania, wymaga kierowania określonych zasobów (pamięci buforowej i przepustowości) do poszczególnych obsługiwanych procesów i (lub) grup obsługiwanych procesów. Dedykowanie zasobów jest przyczyną częściowej redukcji potencjalnych korzyści, wynikających z możliwości dynamicznego przydziału zasobów systemu dla generowanych żądań dostępu do systemu, tzn. podstawowej cechy tych systemów.

Wybór procedur sterowania przepływem jest poprzedzany badaniami porównawczymi (ang. *tradeoff*) korzyści z wprowadzenia danej procedury sterowania oraz potencjalnych strat powodowanych brakiem określonej procedury sterowania. Istotę celu takich badań porównawczych ilustrują ogólne zależności pomiędzy wielkością całkowitego ruchu generowanego w systemie a przepustowością systemu: z idealnymi pro-

cedurami sterowania, bez procedur sterowania i z rzeczywistymi procedurami sterowania (rysunek 6.14).



Rysunek 6.14. Zależność przepustowości od sterowania przepływem

Systemem odniesienia dla każdego rzeczywistego systemu sterowania przepływem jest system z idealnymi procedurami sterowania przepływem (nieistniejącym w praktyce), tzn. system, w którym sterowanie przepływem jest perfekcyjne. Istnienie perfekcyjnego sterowania przepływem oznaczałoby, że kompletna wiedza o stanie całego systemu, pozyskiwana bez opóźnień i angażowania zasobów systemu, jest dostępna w każdej chwili i w każdym miejscu systemu. Przebieg zależności przepustowości w funkcji całkowitego ruchu generowanego w systemie z idealnym sterowaniem przepływem oznacza możliwość uzyskania i utrzymania maksymalnej i teoretycznie dostępnej przepustowości, niezależnie od wielkości generowanego ruchu.

W systemie bez żadnych procedur sterowania przepływem wszystkie ograniczone jego zasoby są udostępniane, na ustalonych dla niego zasadach, wszystkim współzawodniczącym o zasoby procesom (użytkownikom). Przebieg zależności przepustowości w funkcji całkowitego ruchu generowanego przez współzawodniczące procesy (w tym retransmisje) oznacza, że prawdopodobne jest powstanie przeciążeń i zakleszczeń prowadzących do całkowitej degradacji przepustowości i upadku systemu.

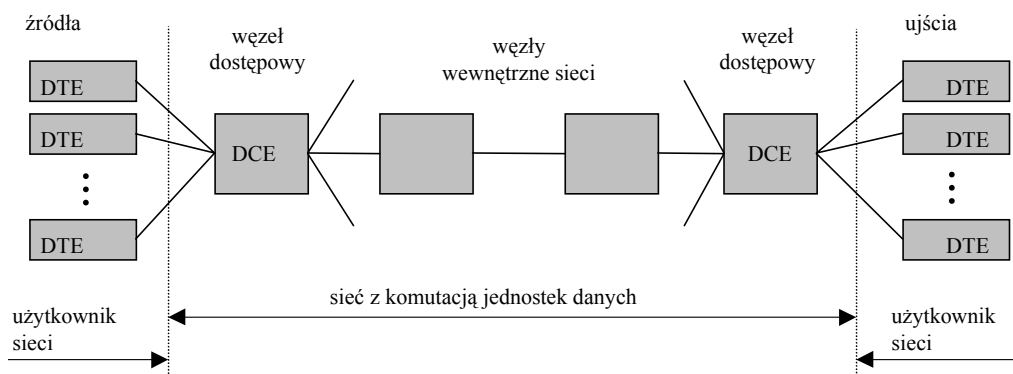
Obszar znajdujący się poniżej krzywej przedstawiającej przepustowość dla systemu z idealnym sterowaniem to taki, w którym mogą się znajdować krzywe odpowiadające różnym procedurom sterowania przepływem. Przebieg krzywej ilustrującej przepustowość w systemie z rzeczywistym sterowaniem oddaje ogólną zasadę sterowania przepływem. Za cenę korzystania z części zasobów na potrzeby sterowania (w tym przypadku pojemności) uzyskiwany jest stan systemu gwarantujący zachowanie określonej przepustowości systemu (mniejszej od teoretycznie możliwej), niez-

leżnie od wielkości generowanego ruchu. W niektórych przypadkach i dla określonych zakresów zmienności całkowitego ruchu rzeczywista przepustowość systemu bez sterowania może być większa od efektywności systemu ze sterowaniem przepływem. Należy jednak uwzględnić fakt, że procedury sterowania przepływem są projektowane i implementowane na potrzeby utrzymania efektywności systemu wówczas, gdy popyt na zasoby przewyższa ich podaż.

6.3.1. ARCHITEKTURA SYSTEMÓW STEROWANIA PRZEPŁYWEM

Procedury sterowania przepływem w systemie są projektowane i implementowane na zasadach determinowanych nadrzędną, wobec procedur sterowania przepływem, architekturą systemu transferu danych. Ze względu na to, że architektury współczesnych sieci teleinformatycznych są architekturami warstwowymi, zbiory procedur sterowania przepływem są projektowane i realizowane w postaci hierarchii komplementarnych procedur sterowania, tj. systemu sterowania przepływem.

Wyróżnienie zakresów funkcjonalnych procedur sterowania przepływem jest związane z ogólną architekturą systemów i sieci komputerowych, która definiuje standardowe interfejsy oraz zakresy funkcji dotyczących granulacji i zwielokrotniania generowanego oraz obsługiwanego ruchu teleinformatycznego. Podstawowym jest podział na tzw. domenę użytkownika (procesu) i domenę sieci, definiowany na styku urządzenia końcowego terminala DTE (ang. *Data Terminal Equipment*) i urządzenia końcowego sieci DCE (ang. *Data Circuit Equipment*) (rysunek 6.15).



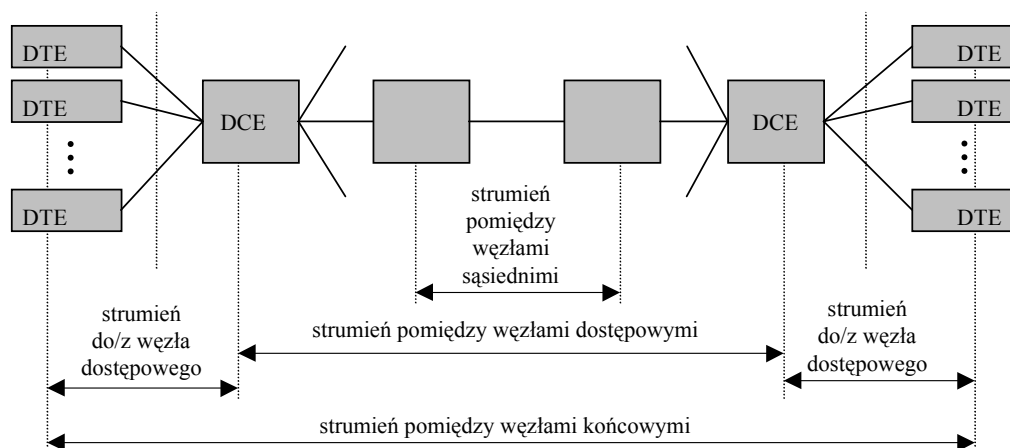
Rysunek 6.15. Ogólna struktura systemu transferu jednostek danych

Z uwagi na różną granulację ruchu teleinformatycznego w sieci, naturalny jest podział metod sterowania przepływem ze względu na wyróżnianie i możliwe do wyróżnienia w sieciach komputerowych strumienie ruchu identyfikowane przez procesy, w których są generowane, przetwarzane i odbierane. Podział taki jest ponadto zgodny

z powszechnie obowiązującą i stosowaną w sieciach architekturą warstwową. Konsekwencją architektury warstwowej sieci jest możliwość implementacji wzajemnie uzupełniających się (także nadmiarowych) procedur sterowania przepływem.

Klasyfikacji strumieni ruchu teleinformatycznego obsługiwanych w sieci z komutacją pakietów na (rysunek 6.16):

- strumień transferowane pomiędzy źródłem i ujściem informacji – elementarne strumienie ruchu identyfikowane procesami użytkowników,
- strumień obsługiwane w sieci – zbiór (suma) wszystkich elementarnych strumieni ruchu obsługiwanych w sieci,
- strumienie przekazywane pomiędzy węzłem wejściowym i węzłem wyjściowym (pomiędzy węzłami dostępowymi) sieci – strumienie identyfikowane tymi samymi miejscami wprowadzania ruchu do sieci i wyprowadzania ruchu z sieci (podzbiór zbioru wszystkich strumieni elementarnych),
- strumienie wymieniane pomiędzy sąsiednimi węzłami sieci – strumienie ruchu identyfikowane tymi samymi węzłami pośrednimi na trasie od węzła wejściowego do węzła wyjściowego sieci (złożenie strumieni elementarnych lub ich części),



Rysunek 6.16. Klasyfikacja strumieni ruchu w systemie transferu jednostek danych

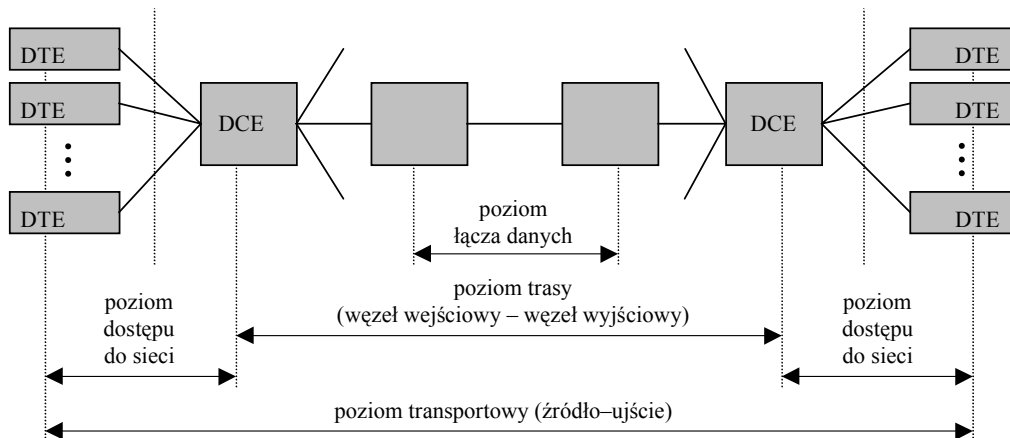
odpowiada wyróżnienie następujących poziomów metod sterowania przepływem (warstw systemu sterowania przepływem), odpowiadających różnym warstwom architektury sieci (rysunek 6.17):

1. Poziom źródło–ujście (poziom transportu) (ang. *transport level*). Celem sterowania jest niezawodne przekazanie jednostek danych (segmentów) w ramach połączenia wirtualnego i zapobieganie przeciążeniom zasobów będących do dyspozycji zdalnych, komunikujących się procesów; zwłaszcza dopasowanie szybkości nadawania do szybkości odbioru w procesie komunikacji źródło–ujście (przedmiotem dopasowania jest elementarny strumień ruchu).

2. Poziom dostępu do sieci (ang. *network access level*). Celem sterowania przepływem jest, w razie zaistnienia takiej potrzeby, hamowanie (dławienie) dopływu do sieci nowych pakietów (pakietów spoza sieci).

3. Poziom trasy (poziom węzeł wejściowy – węzeł wyjściowy) (ang. *entry-to-exit level*). Celem sterowania przepływem jest zapobieganie przeciążeniom zasobów zlokalizowanych w węzłach znajdujących się na trasie (zwłaszcza w węźle końcowym trasy) oraz dopasowanie szybkości nadawania w węźle wejściowym do szybkości odbioru w węźle wyjściowym (dopasowanie dotyczy sumy elementarnych strumieni ruchu).

4. Poziom łącza danych (ang. *hop level, node-to-node level*). Celem sterowania jest utrzymywanie równomiernego przepływu zwielokrotnianych strumieni ruchu zarówno do węzłów dostępowych (w łączach dostępowych do sieci), jak i pomiędzy sąsiednimi węzłami sieci (w łączach wewnętrznych sieci) z jednoczesnym zapobieganiem przeciążeniom zasobów zlokalizowanych w węzłach i zakleszczeniem.



Rysunek 6.17. Poziomy sterowania przepływem odpowiadające różnym warstwom architektury

Wyróżnienie wyżej wymienionych poziomów sterowania ma charakter ogólny. Każdy z tych poziomów sterowania może być przedmiotem dalszego podziału, o zasadności którego decyduje architektura konkretnego systemu lub sieci komputerowej, na potrzeby którego system sterowania przepływem jest projektowany i implementowany. W zależności od zastosowanej, w danym rozwiązaniu teleinformatycznym, hierarchii protokołów system sterowania przepływem może mieć różny zakres funkcjonalny. Hierarchia protokołów danego rozwiązania teleinformatycznego determinuje system sterowania przepływem w ten sposób, że każdy protokół zawiera w sobie właściwe dla niego mechanizmy sterowania przepływem. Istotne jest także to, że w wielu rzeczywistych protokołach współistnieją mechanizmy sterowania przepływem, które mogą być zaliczane do różnych poziomów sterowania przepływem.

Wyróżnienie takiej czy innej hierarchii poziomów sterowania przepływem nie oznacza konieczności stosowania mechanizmów, należących do każdego z tych

poziomów. Ze względu na komplementarność i redundancję funkcji mechanizmów sterowania przepływem na różnych poziomach, istnieje możliwość budowania wielu rozwiązań systemów sterowania przepływem o podobnych zakresach funkcjonalnych.

Projektowanie efektywnych strategii sterowania przepływem w sieciach pakietowych (ogólnie – w sieciach korzystających z technik komutacji jednostek danych) jest zadaniem złożonym. Podstawowym źródłem tej złożoności jest wielowarstwowość rozproszonego systemu sterowania przepływem, implementowanego najczęściej w postaci hierarchii protokołów i algorytmów sterowania dostępem do zasobów sieci.

Z warstwową architekturą systemu sterowania przepływem związane są naturalne dla takich systemów wymagania:

- procedury sterowania przepływem implementowane w danej warstwie i dotyczące strumieni ruchu będących przedmiotem oddziaływania w danej warstwie muszą spełniać warunki spójności, kompletności (procedura spełnia warunki automatu skończonego) i kompatybilności (z procedurami warstw, z których usług korzystają i na rzecz których świadczą usługi),

- procedury sterowania sąsiednich warstw wzajemnie uzupełniają się, a ich oddziaływania na strumieniu ruchu tworzące hierarchie muszą być skoordynowane; w uzasadnionych przypadkach możliwa jest redundancja funkcjonalna.

6.4. POZIOMY STEROWANIA PRZEPLYWEM

6.4.1. STEROWANIE PRZEPLYWEM NA POZIOMIE ŁĄCZA DANYCH

6.4.1.1. CEL STEROWANIA PRZEPLYWEM NA POZIOMIE ŁĄCZA DANYCH

Celem sterowania przepływem na poziomie łącza danych, tzn. w warstwie sterowania łączem danych (ang. *logical link control*) lub wyróżnionych podwarstwach tej warstwy, jest zapobieganie powstawaniu przeciążeń i zakleszczeń oraz przeciwdziałanie następstwom przeciążeń i zakleszczeń dla jakości usług.

Zadaniem sterowania przepływem na omawianym poziomie jest sterowanie napływem nowego ruchu do węzła odbiorczego w zależności od stanu lokalnych kolejek tworzonych w pamięciach buforowych węzłów odbiorczych. Napływ nowego ruchu do węzła odbiorczego jest spowalniany lub wstrzymywany wtedy, gdy przekraczane są wcześniej ustalone wartości progowe zajętości pamięci buforowej. Wartości progowe mogą być definiowane dla całości lub części pojemności pamięci buforowej. Spowalnianie (dławienie) lub wstrzymywanie napływu nowego ruchu oznacza istnienie mechanizmów odrzucania nadmiarowego ruchu w węzłach odbiorczych i powtarzania nadawania (retransmisji) wcześniej odrzuconego ruchu w węzłach nadawczych.

Tak sformułowane zadanie sterowania przepływem ma sens w systemach korzystających z komutacji pakietów (ogólnie jednostek danych) i pracujących w trybie zachowaj i prześlij dalej (ang. *store-and-forward*). Procedury sterowania przepływem na poziomie łącza danych są zwykle wbudowane w protokoły warstwy sterowania łączem danych.

Lokalność procedur sterowania na poziomie łącza danych, których przedmiotem jest ruch przekazywany pomiędzy sąsiednimi węzłami, nie wyklucza możliwości oddziaływania tych procedur na ruch w sieci lub jej wybranych fragmentach. Wynika to z tego, że konieczność zachowania ograniczeń, determinowanych wartościami progowymi zajętości pamięci buforowych, jest propagowana wstecznie (ang. *backpressure*) i wpływa na zmiany szybkości napływu ruchu do sieci oraz na szybkość generowania ruchu w poszczególnych źródłach.

6.4.1.2. KLASYFIKACJA SCHEMATÓW STEROWANIA PRZEPLYWEM NA POZIOMIE ŁĄCZA DANYCH

Zadanie sterowania przepływem na poziomie łącza danych polega na regulacji szybkości napływu jednostek danych z węzła nadawczego do pamięci buforowej węzła odbiorczego. Zasobem krytycznym dla jakości usług na tym poziomie, zależnej od procedur sterowania przepływem, jest pojemność pamięci buforowej węzła odbiorczego, a zwłaszcza sposoby przydziału i podziału tej pojemności determinowane przyjętym w systemie podziałem ruchu teleinformatycznego na klasy. Podział przenieszonego w systemie ruchu teleinformatycznego na klasy odzwierciedla, charakterystyczną dla danego systemu czy sieci komputerowej, filozofię sterowania przepływem.

Podział na klasy decyduje o granulacji ruchu będącego przedmiotem sterowania i – tym samym – o możliwych do sformułowania i osiągnięcia celach sterowania. Sterowanie przepływem na poziomie łącza danych można utożsamiać ze sterowaniem zajętością pamięci buforowej węzła odbiorczego.

Ruch teleinformatyczny (strumień jednostek danych napływające do danego węzła odbiorczego łącza) można dzielić, tzn. wyróżniać klasy ruchu, ze względu na wartości różnych parametrów charakterystycznych dla obsługiwanych strumieni jednostek danych (jednostkami danych w łączu danych są ramki, ale w uproszczeniu można je utożsamiać z pakietami).

Do podstawowych podziałów strumieni jednostek danych na klasy, najczęściej stosowanych w procedurach sterowania przepływem implementowanych w systemach teleinformatycznych, należą:

1. Podział ze względu na wyjściowe kanały transmisyjne (strumień wyjściowe) z danego węzła przeznaczone do transferu poszczególnych strumieni jednostek danych. Liczba klas ruchu w danym węźle odbiorczym zależy od struktury topologicznej sieci, jest stała i równa liczbie wyjściowych kanałów transmisyjnych z danego węzła (jest równa liczbie węzłów odbiorczych dla ruchu napływającego do danego węzła).

Klasy ruchu w danym węźle są identyfikowane adresami węzłów następnym na trasach wychodzących z danego węzła. Podział ten wiąże się z trasami przekazywania ruchu i zależy od reguł doboru tras (*routing*).

2. Podział ze względu na czas przebywania jednostek danych w sieci (mierzony liczbą węzłów, w których jednostka danych była obsługiwana przed nadejściem do danego węzła). Liczba klas ruchu (klas strumieni jednostek danych) zależy od struktury topologicznej sieci, jest stała i jest równa liczbie wszystkich długości tras w danej sieci (jest równa liczbie wszystkich węzłów pomniejszonej o jeden, co wynika z założenia braku pętli na trasach). Klasy ruchu w danym węźle identyfikowane są liczbą węzłów na trasach jednostek danych przebytych przed nadejściem do danego węzła. Podział ten wiąże się z ilością zasobów zaangażowanych w obsługę jednostek danych przed napływem do danego węzła.

3. Podział ze względu na połączenia wirtualne, do których należą napływające strumienie jednostek danych. Liczba klas ruchu w danym węźle jest zmienna, nie zależy od struktury topologicznej sieci i jest równa aktualnie utrzymywanym połączeniom wirtualnym. Klasy ruchu w danym węźle są identyfikowane adresami procesów końcowych połączenia wirtualnego.

Oprócz wymienionych podziałów ruchu na klasy możliwe są inne, wykorzystujące wartości innych parametrów charakterystycznych dla strumieni jednostek danych: adresy węzłów wejściowych strumieni pakietów, adresy węzłów wyjściowych strumieni jednostek danych, adresy węzłów wejściowych i wyjściowych strumieni jednostek danych, wartości wskaźników jakości itd.

Na jakość sterowania przepływem na poziomie łącza danych, niezależnie od przyjętego podziału ruchu na klasy, wpływ mają także sposoby przydziału i podziału całkowitej pojemności pamięci buforowej pomiędzy współzawodniczące o pamięć buforową poszczególne klasy ruchu.

Do najczęściej stosowanych sposobów przydziału pojemności pamięci buforowej należą:

- przydział statyczny – całkowita pojemność pamięci buforowej jest dzielona i przydzielana klasom ruchu niezależnie od aktualnych wielkości ruchu,
- przydział dynamiczny – całkowita pojemność pamięci buforowej jest przydzielana poszczególnym klasom ruchu w zależności od aktualnych wielkości ruchu.

Niezależnie od przyjętego sposobu przydziału pojemności pamięci buforowej stosowane są różne strategie podziału pamięci buforowej, a najczęściej:

- podział całkowity (bez podziału) – dla każdej klasy ruchu dostępna jest całkowita pojemność pamięci buforowej,
- podział równomierny – dla każdej klasy ruchu dostępna jest stała, jednakowa dla wszystkich klas, część pojemności pamięci buforowej niezależnie od wielkości ruchu danej klasy; suma części dostępnych dla poszczególnych klas jest nie większa od całkowitej pojemności pamięci buforowej,
- podział proporcjonalny – dla każdej klasy ruchu dostępna jest stała część pojemności pamięci buforowej, proporcjonalna do wielkości obsługiwanego ruchu danej

klasy; suma części dostępnych dla poszczególnych klas ruchu jest nie większa od całkowitej pojemności pamięci buforowej,

➤ podział proporcjonalny, nadmiarowy – dla każdej klasy ruchu dostępna jest stała część pojemności pamięci buforowej, proporcjonalna do wielkości obsługiwanego ruchu danej klasy; suma części dostępnych dla poszczególnych klas ruchu jest większa od całkowitej pojemności pamięci buforowej,

➤ podział mieszany – dla każdej klasy ruchu dostępna jest określona minimalna część pojemności pamięci buforowej (suma minimalnych części jest mniejsza od całkowitej pojemności); pozostała pojemność pamięci buforowej jest dostępna dla wszystkich klas ruchu.

6.4.1.3. STEROWANIE PRZEPLÝWEM DLA KLAS JEDNOSTEK DANYCH WYRÓŻNIANYCH ZE WZGLĘDU NA WYJŚCIOWE KANAŁY TRANSMISYJNE

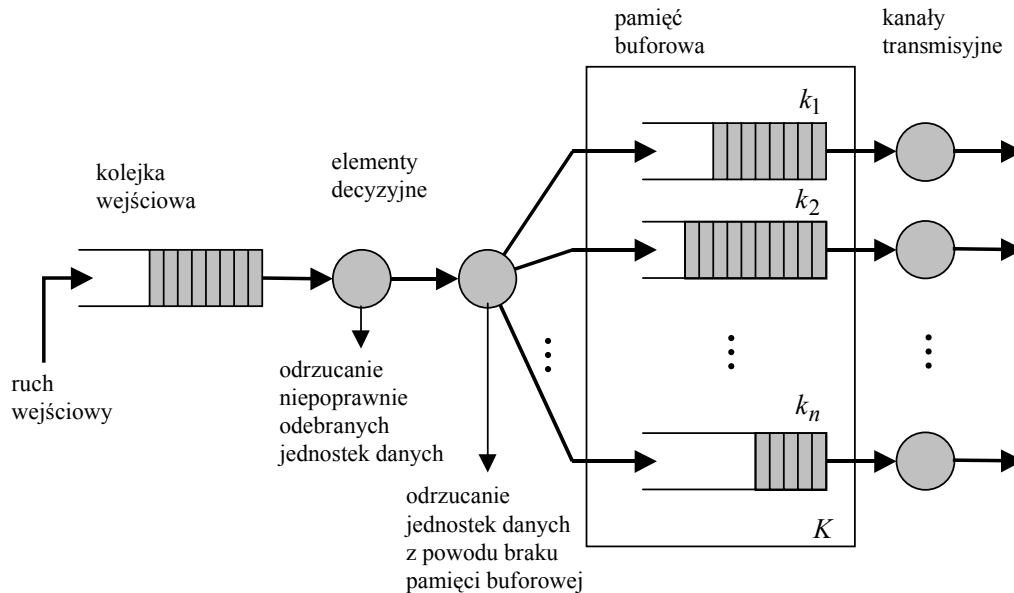
Przedmiotem sterowania przepływem są klasy ruchu teleinformatycznego, których liczba, w ogólnym przypadku, jest różna dla każdego węzła i równa liczbie wyjściowych kanałów transmisyjnych z danego węzła. Procedury sterowania przepływem, których przedmiotem są tak wyróżniane klasy ruchu, noszą nazwę procedur sterowania długością (ograniczeniem) kolejek wyjściowych (ang. *channel queue limit*).

Złożony strumień pakietów (pakiet jest utożsamiany z jednostką danych), na który składają się strumienie pakietów generowanych w węzłach źródłowych, wchodzi do kolejki wejściowej węzła. Obsługa pakietów w kolejce (zgodnie z przyjętą strategią obsługi) polega na podjęciu decyzji o jego przyjęciu lub odrzuceniu po sprawdzeniu poprawności odbioru i możliwości przyjęcia do pamięci buforowej. Jeżeli pakiet jest odebrany poprawnie, to zostaje w węźle i jest dalej przetwarzany; w przeciwnym razie jest odrzucany. Poprawnie odebrany pakiet jest przekazywany do kolejki wyjściowej, jeżeli w kolejce wyjściowej są wolne bufory; w przeciwnym razie jest odrzucany z powodu braku pamięci buforowej. Przekazanie poprawnie odebranego pakietu do kolejki wyjściowej w pamięci buforowej odbywa się zgodnie z zawartością tablicy tras routingu. W kolejce wyjściowej pakiet oczekuje na obsługę (transmisję) zgodnie z przyjętą strategią obsługi. Pakiet z kolejki jest usuwany natychmiast po jego transmisji (model z odrzuceniami) albo po otrzymaniu potwierdzenia jego poprawnego odbioru (model z retransmisjami) (rysunek 6.18).

Akceptacja lub odrzucenie pakietów w węźle wyjściowym zależy od zastosowanej strategii podziału pojemności pamięci buforowej i aktualnego stanu zajętości kolejek. W dalszej części przyjęto, bez ograniczania ogólności rozważań, że pojemność pamięci buforowej składa się z buforów, których pojemność odpowiada jednemu pakietowi.

W pamięci buforowej węzła odbiorczego o pojemności K buforów organizowanych jest n kolejek wyjściowych. Liczba kolejek jest równa liczbie kanałów wyjściowych z danego węzła. Pakiety napływające do danego węzła odbiorczego są akceptowane, jeżeli są odebrane poprawnie (bezbłędnie) oraz gdy spełniony jest warunek:

$k_i < K_i$, gdzie k_i oznacza liczbę zajętych buforów w kolejce wyjściowej pamięci buforowej o pojemności K_i ($i = 1, 2, \dots, n$); w innym przypadku pakiety są odrzucane.



Rysunek 6.18. Obsługa pakietów przez procedurę sterowania długością kolejek wyjściowych

Po przyjęciu statycznego przydziału pojemności pamięci buforowej możliwe są różne wersje sterowania przepływem, różniące się strategią podziału pojemności pamięci buforowej.

6.4.1.3.1. PODZIAŁ CAŁKOWITY

Strategia podziału całkowitego pojemności pamięci buforowej CS (ang. *Complete Sharing*) polega na tym, że maksymalne długości kolejek wyjściowych mogą być równe całkowitej pojemności pamięci buforowej.

Maksymalne długości wszystkich n kolejek wyjściowych, organizowanych w pamięci buforowej, są jednakowe i równe pojemności pamięci buforowej, z tym że suma zajętych buforów nie może być większa od całkowitej pojemności pamięci buforowej, tzn.:

$$0 \leq k_i \leq K_i, \quad 0 \leq K_i \leq K \quad \text{i} \quad \sum_{i=1}^n k_i \leq K \quad \text{dla} \quad i = 1, 2, \dots, n.$$

Udział długości poszczególnych kolejek wyjściowych w całkowitej pojemności pamięci buforowej węzła zależy od intensywności strumieni ruchu kierowanych do poszczególnych kanałów wyjściowych. Omawiana strategia podziału praktycznie nie ogranicza rywalizacji poszczególnych strumieni ruchu o dostęp do pojemności pamięci buforowej. Jedyną ingerencją strategii podziału to blokowanie dostępu w razie wyczerpania całkowitej pojemności pamięci buforowej.

Podział taki:

- nie zapobiega powstawaniu zakleszczeń bezpośrednich; możliwość zawłaszczenia całkowitej pojemności pamięci buforowej przez jedno ze źródeł ruchu może powodować zakleszczenia dla pozostałych klas ruchu,
- nie zapobiega pogorszeniu przepustowości powodowanemu jej marnotrawieniem wówczas, gdy następuje zawłaszczenie pojemności pamięci buforowej przez jedną lub kilka wybranych klas ruchu,
- nie zapobiega niesprawiedliwemu podziałowi zasobów; może powodować całkowitą degradację jakości obsługi wybranych klas ruchu oraz brak możliwości obsługi klas ruchu innych niż zawłaszczające pojemność pamięci buforowej.

6.4.1.3.2. PODZIAŁ RÓWNOMIERNY

Strategia podziału równomiernego pojemności pamięci buforowej CP (ang. *Complete Partitioning*) polega na tym, że maksymalna długość poszczególnych kolejek jest proporcjonalna do liczby organizowanych kolejek.

Dostępne dla poszczególnych klas ruchu maksymalne długości wszystkich n kolejek wyjściowych organizowanych w pamięci buforowej są jednakowe, a ich suma jest równa całkowitej pojemności pamięci buforowej. Spełniony jest warunek:

$$0 \leq k_i \leq K_i, \quad K_i = \frac{K}{n} \quad \text{dla } i = 1, 2, \dots, n.$$

Efektywność omawianego, konserwatywnego podziału pojemności pamięci buforowej zwiększa się wraz ze wzrostem równomierności udziału poszczególnych klas ruchu w napływającym do węzła strumieniu pakietów. Nierównomierność napływu pakietów należących do różnych klas zgłoszeń powoduje, że część pojemności pamięci buforowej może nie być wykorzystywana, podczas gdy pakiety należące do niektórych klas będą odrzucane po wyczerpaniu dostępnej pojemności kolejek wyjściowych.

Podział taki:

- zapobiega powstawaniu zakleszczeń bezpośrednich,
- nie zapobiega pogorszeniu przepustowości powodowanemu jej marnotrawieniem wówczas, gdy rezerwowane pojemności kolejek wyjściowych nie są wykorzystywane (nadmiar zasobów dla klas ruchu o względnie mniejszej intensywności napływu),
- nie zapobiega niesprawiedliwemu podziałowi zasobów; może powodować degradację jakości obsługi wybranych klas ruchu przez odrzucanie pakietów mimo niewykorzystanych zasobów (niedmiar zasobów dla klas ruchu o względnie większej intensywności napływu).

6.4.1.3.3. PODZIAŁ Z MAKSYMALNĄ DŁUGOŚCIĄ KOLEJKI

Strategia podziału pamięci buforowej z maksymalną długością kolejki SMQ (ang. *Sharing with Maximum Queues*) polega na nadmiarowym podziale pojemności pamięci buforowej, z tym że może ona być zmajoryzowana przez jedną z klas ruchu.

Dostępne dla wyróżnianych klas ruchu maksymalne długości wszystkich n kolejek wyjściowych, organizowanych w pamięci buforowej, są jednakowe, a ich suma jest większa od całkowitej pojemności pamięci buforowej:

$$0 \leq k_i \leq K_i, \quad K_i > \frac{K}{n} \quad \text{oraz} \quad \sum_{i=1}^n k_i \leq K \quad \text{dla} \quad i = 1, 2, \dots, n.$$

Efektywność takiego podziału pojemności pamięci buforowej zależy przede wszystkim od doboru wartości K_i . Podział ten jest reprezentatywny dla rozpowszechnionej w systemach i sieciach komputerowych zasady podziału zasobów. Nadmiarowość podziału zasobów jest z założenia kompensowana statystycznymi właściwościami napływających pakietów należących do różnych klas ruchu. Taki sposób podziału pamięci buforowej nie wyklucza możliwości jej zawłaszczenia przez kilka klas ruchu kosztem pozostałych.

Podział taki:

- nie zapobiega powstawaniu zakleszczeń bezpośrednich; możliwość zawłaszczenia pojemności pamięci buforowej przez wybrane klasy ruchu może powodować powstawanie zakleszczeń dla pozostałych klas ruchu,

- zapobiega pogorszeniu przepustowości powodowanemu jej marnotrawieniem; możliwość zawłaszczania pojemności pamięci buforowej powoduje, że pamięć buforowa nie jest wykorzystana mimo istnienia zapotrzebowania,

- nie zapobiega niesprawiedliwemu podziałowi zasobów; może powodować degradację jakości obsługi wybranych klas ruchu przez zawłaszczenie pojemności pamięci buforowej przez wybrane klasy ruchu.

6.4.1.3.4. PODZIAŁ Z MINIMALNĄ DŁUGOŚCIĄ KOLEJKI

Strategia podziału pamięci buforowej z minimalną długością kolejki SMA (ang. *Sharing with Minimum Allocation*) jest strategią podziału sprawiedliwego; każdej z klas ruchu gwarantowany jest dostęp do wydzielonej części pojemności pamięci buforowej, proporcjonalnej do liczby wyróżnionych klas ruchu.

Podział gwarantuje każdej z klas ruchu jednakową, minimalną długość kolejki wyjściowej. Suma minimalnych długości wszystkich kolejek wyjściowych jest mniejsza od całkowitej pojemności pamięci buforowej:

$$K_{\min} \leq \frac{K}{n} \quad \text{oraz} \quad \sum_{i=1}^n \max(0, k_i - K_{\min}) \leq K - nK_{\min} \quad \text{dla} \quad i = 1, 2, \dots, n.$$

Taki podział pojemności pamięci buforowej wyklucza możliwość zawłaszczenia całkowitej pojemności pamięci buforowej przez kilka klas ruchu kosztem pozostałych. Właściwy dobór wartości K_{\min} pozwala na zagwarantowanie określonej jakości obsługi dla klas ruchu o względnej, mniejszej intensywności napływu pakietów. Omawiany sposób podziału nie zapobiega zawłaszczeniu pojemności $K - nK_{\min}$, co może prowadzić do podziału obsługiwanych klas ruchu na dwie grupy – grupa klas pakietów obsługiwanych z wykorzystaniem pojemności K_{\min} oraz grupa klas pakie-

tów wykorzystujących, oprócz pojemności K_{\min} , także zawłaszczoną pojemność $K - nK_{\min}$.

Podział taki:

- zapobiega powstawaniu zakleszczeń bezpośrednich; każda klasa ruchu dysponuje co najmniej kolejką o długości K_{\min} ,
- ogranicza możliwość degradacji przepustowości systemu, powodowanej jej marnotrawieniem, do przypadków, kiedy rezerwowane minimalne pojemności kolejek wyjściowych nie są wykorzystywane; występuje nadmiar zasobów dla klas ruchu o mniejszym, w porównaniu z gwarantowanym minimalną długością kolejki wyjściowej K_{\min} , zapotrzebowaniu na pamięć buforową,
- zapobiega, z dokładnością wynikającą z wartości K_{\min} , niesprawiedliwemu podziałowi zasobów; każda klasa ruchu ma gwarantowaną minimalną długość kolejki wyjściowej.

6.4.1.3.5. PODZIAŁ Z MINIMALNĄ I MAKSYMALNĄ DŁUGOŚCIĄ KOLEJKI

Strategia podziału pamięci buforowej z minimalną i maksymalną długością kolejki SMAMQ (ang. *Sharing with Minimum Allocation and Maximum Queue*) polega na połączeniu sprawiedliwego i bezpiecznego podziału pojemności pamięci buforowej.

Podział gwarantuje każdej klasie ruchu minimalną (stałą dla wszystkich) długość kolejki wyjściowej, jednocześnie nie dopuszczając do zajęcia pojemności większej od określonej maksymalną długością kolejki:

$$\sum_{i=1}^n \max(0, k_i - K_{\min}) \leq K - nK_{\min}, \quad 0 \leq k_i \leq K_i \text{ dla } i = 1, 2, \dots, n.$$

Podział pojemności pamięci buforowej gwarantujący określoną minimalną długość kolejki wyjściowej i jednocześnie ograniczający jej maksymalną długość jest najbliższy podziałowi optymalnemu. Odpowiedni dobór wartości K_{\min} i K_i ($i = 1, 2, \dots, n$) (w ogólnym przypadku $K_{i\min}$ i K_i , tzn. minimalne długości kolejek wyjściowych dla różnych klas mogą być różne) pozwala na zapobieganie powstawaniu zakleszczeń oraz ograniczenie możliwości degradacji przepustowości dla każdej klasy obsługiwanego ruchu.

Podział taki:

- zapobiega powstawaniu zakleszczeń bezpośrednich; każda klasa ruchu dysponuje co najmniej kolejką o pojemności K_{\min} ,
- ogranicza możliwość pogorszenia przepustowości, powodowanego jej marnotrawieniem do przypadków, gdy rezerwowane minimalne pojemności kolejek wyjściowych nie są wykorzystywane (nadmiar zasobów dla klas ruchu o względnie mniejszej, w porównaniu z gwarantowaną pojemnością pamięci buforowej K_{\min} , intensywności napływu),
- gwarantuje, z dokładnością do wartości ilorazu K_i/K_{\min} , sprawiedliwy podział zasobów; każda klasa ruchu ma gwarantowaną minimalną pojemność kolejki wyjściowej, a jednocześnie nie może korzystać z pojemności większej niż zadana maksymalna długość kolejki K_i .

6.4.1.3.6. PODZIAŁ POJEMNOŚCI PAMIĘCI BUFOROWYCH W WĘZŁACH WYJŚCIOWYCH RUCHU

Zwiększenie elastyczności omawianych strategii podziału pamięci buforowej jest rezultatem różnicowania wartości parametrów poszczególnych kolejek organizowanych w pamięci buforowej i uzależniania ich od wartości i zakresu zmian intensywności napływu zgłoszeń należących do poszczególnych klas ruchu.

Rozwiązanie polegające na organizacji jednakowych kolejek wyjściowych jest stosowane w węzłach, z których pakiety są transmitowane wyłącznie do innych węzłów sieci. Gdy w danym węźle część wejściowego strumienia pakietów jest wyprowadzana poza sieć (do procesów, dla których dany węzeł jest węzłem wyjściowym) strategia podziału pojemności pamięci buforowej jest zwykle inna.

Po przyjęciu założenia, że:

- kolejki wyjściowe o indeksach $i = 1, 2, \dots, n - 1$ są kolejkami wyjściowymi kanałów transmisyjnych łączących dany węzeł z innymi węzłami sieci,
- kolejka wyjściowa o indeksie n jest kolejką, w której gromadzone są pakiety wprowadzane za zewnątrz sieci,

strategia podziału pojemności pamięci buforowej powinna uwzględniać wymagania związane z koniecznością kompletowania wiadomości z pakietów, dla których dany węzeł jest węzłem końcowym. W takich przypadkach pojemność pamięci buforowej jest dzielona na dwie części: K' – część pojemności pamięci buforowej dzielona pomiędzy $(n - 1)$ kanałów wyjściowych z danego węzła oraz K'' – część pojemności wykorzystywana do gromadzenia pakietów, z których są kompletowane wiadomości ($K = K' + K''$). Wartość pojemności K'' jest determinowana maksymalną długością wiadomości (w danym węźle lub w całej sieci) i zwykle jest kilkakrotnie większa od pojemności przydzielanych pozostałym kolejkom wyjściowym.

6.4.1.4. STEROWANIE PRZEPLYWEM DLA KLAS RUCHU JEDNOSTEK DANYCH WYRÓŻNIANYCH ZE WZGLĘDU NA CZAS PRZEBYWANIA W SIECI

Sterowanie przepływem dla klas pakietów wyróżnianych ze względu na czas przebywania w sieci polega na tym, że priorytet obsługi pakietu w węzłach sieci zwiększa się wraz z wydłużeniem czasu przebywania pakietu w sieci, tj. ze wzrostem numeru klasy, do której pakiet należy. Czas przebywania w sieci pakietu napływającego do danego węzła jest równy liczbie węzłów, w których dany pakiet był poprzednio obsługiwany, jednoznacznie określającej liczbę łączy międzywęzłowych wykorzystywanych do transferu pakietu, tj. liczbę wykonanych tzw. skoków pomiędzy węzłami (ang. *hops*).

Metody sterowania przepływem, polegające na statycznej lub dynamicznej alokacji pojemności pamięci buforowej w węzłach wyróżnianych w sieci – zgodnie z różnymi kryteriami – klas pakietów, są zaliczane do tzw. metod klas buforów (ang.

buffer class). Jedną z nich, stosującą do celów sterowania przepływem zagnieżdżony podział pojemności pamięci buforowych węzłów pomiędzy klasy pakietów o różnym czasie przebywania w sieci, jest metoda strukturalizacji pojemności pamięci buforowej SBP (ang. *Structured Buffer Pool*).

Wszystkie pakiety napływające do danego węzła są dzielone na klasy, a następnie obsługiwane w zależności od przynależności do określonej klasy. Zgodnie z zasadami strukturalizacji pojemności pamięci buforowej, liczba różnych klas pakietów w danej sieci jest stała; jest determinowana długością najdłuższej trasy (bez pętli) w danej sieci. Długość najdłuższej trasy w danej sieci zależy od struktury topologicznej sieci oraz przyjętych reguł wyznaczania tras (ang. *routing*) i jest mniejsza lub równa liczbie wszystkich węzłów sieci. Zmiana reguł wyznaczania tras w sieci może powodować zmianę długości najdłuższej trasy w sieci i tym samym maksymalnej liczby klas pakietów. Przy założeniu, że najdłuższa trasa w danej sieci zawiera m węzłów (węzeł początkowy trasy, $m - 2$ węzłów pośrednich i węzeł końcowy trasy) oraz $m - 1$ łączy międzywęzłowych, liczba wszystkich klas pakietów w sieci $L = m$ (gdzie $m \leq N$; N jest liczbą wszystkich węzłów sieci).

Zgodnie z podaną definicją czasu przebywania pakietu w sieci, pakiet wprowadzany z zewnątrz do węzła sieci (pierwszego węzła trasy) należy do klasy 0, ponieważ nie był wcześniej obsługiwany w żadnym węźle sieci. Pakiety napływające do danego węzła są zaliczane do najwyższej klasy, tj. do klasy L , gdy były wcześniej obsługiwane w L węzłach (wtedy dany węzeł jest węzłem ujściowym tych pakietów) albo gdy – niezależnie od czasu przebywania pakietów w sieci – dany węzeł jest węzłem ujściowym (końcowym) pakietów. Rozwiązanie takie jest podyktowane przyjęciem założenia, że wyprowadzanie pakietów z sieci odbywa się w postaci wiadomości i w węźle ujściowym potrzebna jest określona pojemność pamięci buforowej niezbędna do skompletowania (ang. *reassembly*) wiadomości. Zaliczanie pakietów wychodzących z sieci, niezależnie od czasu przebywania w niej, do najwyższej klasy jest wyrazem uprzywilejowania pakietów wychodzących z sieci; jest naturalne, że w razie konfliktu w dostępie do zasobów najwyższy priorytet obsługi w danym węźle mają pakiety wyprowadzane z sieci.

Zasady obsługi wyróżnianych klas pakietów odzwierciedla organizacja pamięci buforowej w węzłach sieci, zgodnie z którą każda klasa pakietów może korzystać ze ściśle zdefiniowanych zbiorów (części) buforów pamięci buforowej:

➤ pakiety należące do klasy zerowej Cl_0 ($i = 0$) mogą zajmować tylko bufory należące do zbioru K_0 ; liczba buforów K_0 w zbiorze K_0 (pojemność zbioru buforów K_0) jest dobrana tak, aby w zbiorze tym możliwe było odebranie i przechowanie najdłuższej – w danej sieci – wiadomości, tj. wiadomości o długości l_{\max} (długość wiadomości wyrażana w pakietach lub buforach): $K_0 > l_{\max}$,

➤ pakiety należące do i -tej klasy Cl_i ($i = 1, 2, \dots, l_{\max} - 1$) mogą zajmować wszystkie bufory dostępne dla pakietów $(i - 1)$ -szej klasy Cl_{i-1} oraz dodatkowo jeden bufor ($K_{i-1} \subset K_i$, $K_i = K_{i-1} + 1$); K_i jest maksymalną liczbą buforów dostępnych dla pakietów i -tej klasy,

wane pakiety należące do klasy Cl_0 ; w przeciwnym razie pakiety klasy Cl_0 mogą być przyjmowane,

➤ większa lub równa maksymalnej liczbie buforów dostępnych dla pakietów klasy Cl_0 pomniejszonej o liczbę buforów pozwalających na wprowadzenie z zewnątrz do sieci wiadomości o maksymalnej długości l_{\max} (pakietów), tj. $\sum_{i=1}^L k_i \geq K_0 - l_{\max}$, to do pamięci buforowej danego węzła nie mogą być przyjmowane pakiety należące do klasy Cl_0 wprowadzane do sieci w postaci wiadomości; w przeciwnym razie pakiety klasy Cl_0 , wprowadzane do sieci w postaci wiadomości, mogą być przyjmowane,

➤ jeżeli suma pakietów w pamięci buforowej węzła, należących do klas o numerach wyższych od i ($i = 1, 2, \dots, L - 1$), jest większa lub równa maksymalnej liczbie buforów dostępnych dla klasy Cl_i , tj. $\sum_{j>i}^L k_j \geq K_i$, to do pamięci buforowej węzła nie mogą być przyjmowane pakiety należące do klas o numerach równych i mniejszych od i ; pakiety należące do wymienionych klas są odrzucane.

Zastosowanie omawianej procedury sterowania powoduje, że w razie wystąpienia konfliktu żądań dostępu do pamięci buforowej w pierwszej kolejności są obsługiwane pakiety o dłuższym czasie przebywania w sieci, kosztem obsługi pakietów o krótszym czasie pobytu w sieci i (lub) kosztem chwilowego wstrzymania napływu nowych pakietów spoza systemu. Dłuższy czas przebywania pakietu w sieci to większa ilość zasobów zaangażowanych w jego dotychczasową obsługę, dlatego preferowanie przez procedurę sterowania przepływem pakietów o dłuższym czasie przebywania w sieci jest jednym ze sposobów ochrony wcześniejszych inwestycji w obsługę ruchu w sieci.

Omawiana procedura sterowania przepływem skutecznie zapobiega powstawaniu zakleszczeń bezpośrednich i pośrednich. Wzrost priorytetu obsługi pakietu w węźle wraz z rosnącym numerem klasy, do której pakiet należy, oraz brak możliwości zawłaszczenia całkowitej pojemności pamięci buforowej przez pakiety klasy innej niż najwyższa powoduje, że nie ma warunków powstawania cyklu charakterystycznego dla zakleszczeń bezpośredniego i pośredniego. W najmniej korzystnej sytuacji w systemie zawsze istnieje, być może ograniczona, możliwość przepływu pakietów do węzła końcowego. Dzieje się tak dlatego, że pojemność dostępna tylko dla pakietów z najwyższej klasy jest niedostępna dla pakietów klas niższych oraz że pojemność dostępna dla pakietów klasy wyższej jest zawsze większa (co najmniej o jeden bufor) niż pojemność dostępna dla pakietów klasy niższej. Ta właściwość struktury pamięci buforowej oraz rezerwacja jej pojemności, pozwalającej na skompletowanie wiadomości w celu wyprowadzenia pakietów z sieci, powoduje, że nie powstają warunki mogące doprowadzić do wstrzymania wyprowadzania pakietów poza sieć. Istnienie możliwości przepływu pakietów oznacza, że przepustowość systemu jest różna od zera; tym samym nie jest spełniony warunek konieczny zakleszczenia – całkowita degradacja przepustowości. Utrzymywanie ciągłej zdolności systemu do wyprowadzania pakietów powoduje z kolei, że zwalniane zasoby są udostępniane pakietom z klas niższych, które dzięki temu mogą przechodzić do klas wyższych. Wytworzony

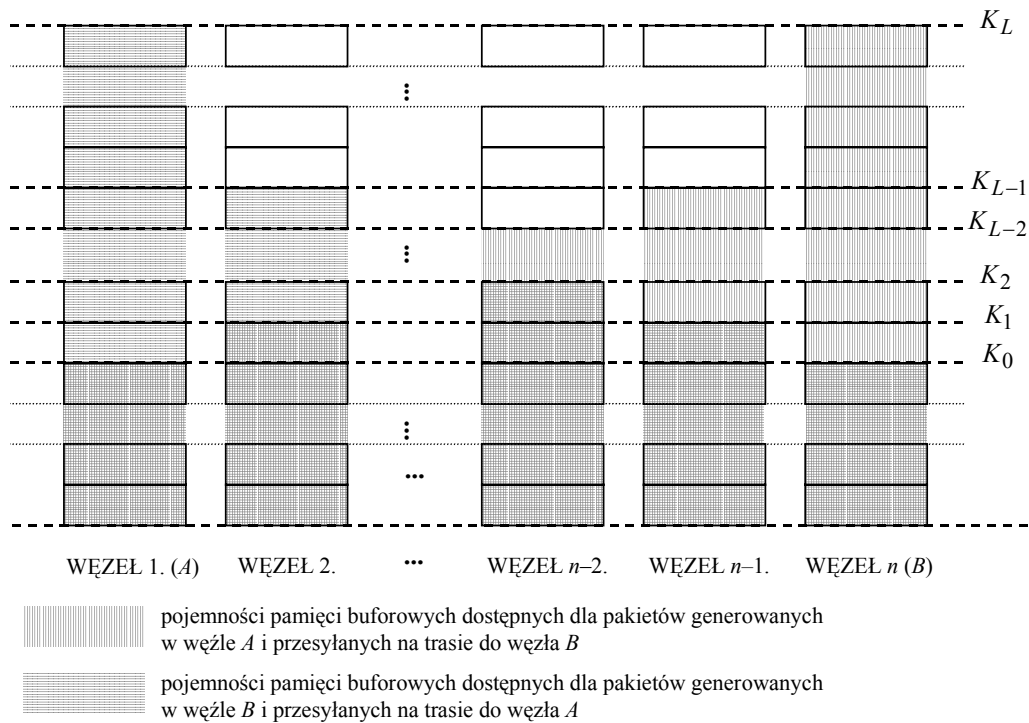
w ten sposób mechanizm wykorzystujący zasady propagacji wstecznej powoduje, że system z taką procedurą sterowania ma zdolność do samoregulacji.

Omawiana procedura sterowania przepływem zapobiega powstawaniu zakleszczeń, tzn. zapobiega całkowitej degradacji przepustowości systemu, ale nie zapobiega pogorszeniu przepustowości. Główną przyczyną możliwej degradacji przepustowości systemu jest to, że w warunkach dużego obciążenia systemu możliwe jest zawłaszczenie przez pakiety klasy Cl_0 maksymalnej liczby buforów dostępnych dla pakietów tej klasy, co powoduje, że na potrzeby obsługi pakietów klas wyższych pozostają co najwyżej buforów dostępne dla pakietów danej klasy i niedostępne dla pakietów klasy Cl_0 . Stopień degradacji przepustowości systemu zależy od ilościowych charakterystyk struktury podziału pojemności pamięci buforowej, przede wszystkim od bezwzględnego i względnego udziału liczby buforów dostępnych dla klasy Cl_0 (K_0) w całkowitej pojemności pamięci buforowej K , różnic liczby buforów dostępnych dla klas wyższych $K_{i+1} - K_i$ (gdzie $i = 1, 2, \dots, L - 2$) oraz liczby buforów dostępnych dla pakietów klasy Cl_L (K_L).

W przypadku omawianej procedury sterowania przepływem potencjalna przepustowość systemu (przyjmuje się dla uproszczenia, że rezerwowana dla poszczególnych klas pakietów pojemność pamięci buforowych we wszystkich węzłach jest taka sama) może być zmniejszona (zdegradowana) do chwilowej, rzeczywistej przepustowości, pozwalającej na transfer tylko jednego pakietu, wtedy, gdy dla przyjętego sposobu strukturalizacji pamięci buforowych, polegającego na zagnieżdżaniu pojemności dostępnej klasom niższym w pojemnościach dostępnych klasom wyższym, minimalna różnica pomiędzy pojemnościami pamięci buforowych dostępnych dla poszczególnych klas pakietów wynosi jeden bufor, tzn. jeżeli $\min(K_{i+1} - K_i) = 1$ dla $i = 1, 2, \dots, L - 2$. Różnica pomiędzy pojemnościami pamięci buforowych, rezerwowanych dla kolejnych coraz wyższych klas pakietów, powoduje, że w systemie istnieje co najmniej jedna trasa, pomiędzy jednym ze źródeł a jednym z ujść, o przepustowości równej minimalnej wartości różnic pojemności dostępnych dla poszczególnych klas ruchu. Istnienie takiej trasy oznacza, że rzeczywista minimalna przepustowość systemu jest różna od zerowej. Zwiększenie różnicy pomiędzy pojemnościami rezerwowanymi dla kolejnych coraz wyższych klas pakietów powoduje zwiększenie wartości minimalnej przepustowości systemu. I tak, jeżeli $\min(K_{i+1} - K_i) = 2$ dla $i = 1, 2, \dots, L - 2$, to minimalna przepustowość systemu zwiększa się dwukrotnie (rysunek 6.20).

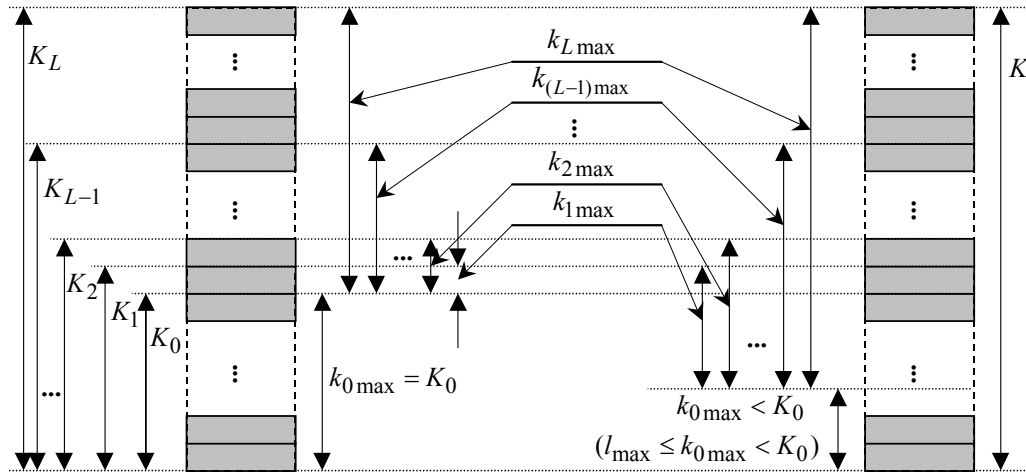
Celem różnych modyfikacji omawianej metody sterowania przepływem jest wprowadzanie zmian, których zadaniem jest ograniczenie stopnia możliwej degradacji przepustowości systemu. Zgodnie z wcześniejszymi uwagami główną przyczyną degradacji przepustowości jest możliwość zawłaszczenia – w pamięci buforowej o zagnieżdżonej strukturze pojemności dostępnych różnym klasom pakietów – części buforów przez pakiety napływające spoza systemu (pakiety klasy zerowej), powodująca znaczne ograniczenie liczby buforów dostępnych dla pakietów klas wyższych. Z tego powodu modyfikacje, prowadzące do poprawy jakości omawianej metody sterowania, polegają na jednoczesnym lub zamiennym:

- ograniczaniu napływu pakietów spoza systemu (pakietów klasy Cl_0) wraz ze wzrostem obciążenia systemu,
- rezygnowaniu z zagnieżdżonej struktury podziału pamięci buforowej i wprowadzeniu podziału polegającego na udostępnianiu poszczególnym klasom pakietów rozłącznych podzbiorów buforów pamięci buforowej węzłów.

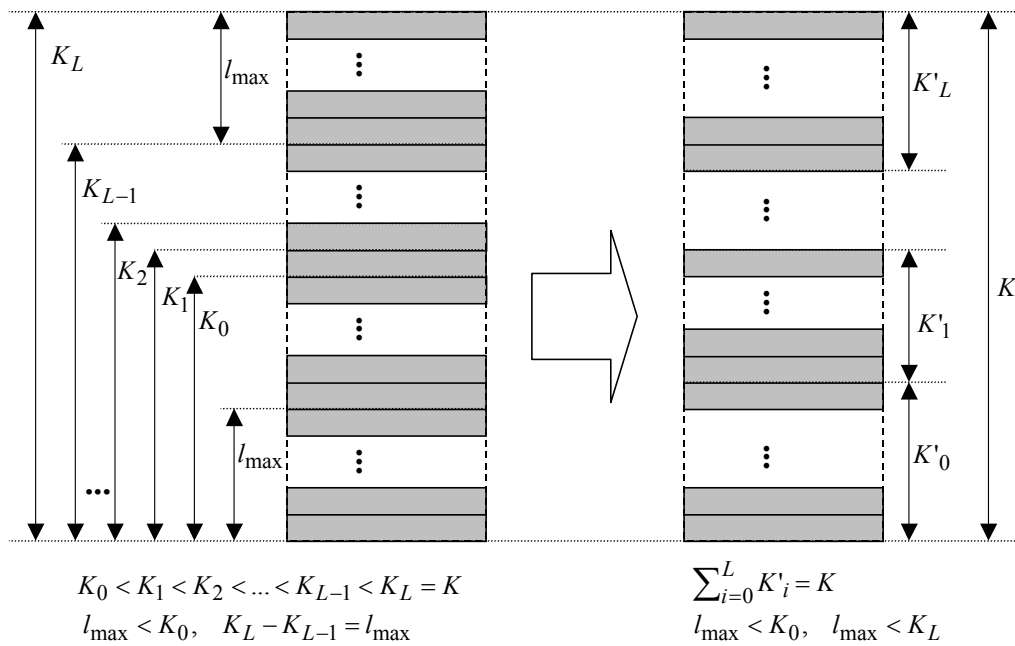


Rysunek 6.20. Zależność wartości minimalnej przepustowości systemu od różnic pojemności dostępnych dla poszczególnych klas ruchu

Celem ograniczania napływu pakietów spoza systemu, w przypadku zachowania zagnieżdżonej struktury zbiorów buforów dostępnych różnym klasom pakietów, jest zwiększenie szybkości obsługi pakietów już znajdujących się w systemie przez udostępnienie klasom tych pakietów większej liczby buforów. W razie względnie małego obciążenia systemu maksymalna liczba pakietów klasy Cl_0 w pamięci buforowej jest równa liczbie wszystkich buforów dostępnych dla klasy Cl_0 : $k_{0 \max} = K_0$. Ograniczenie napływu nowych pakietów spoza sieci odbywa się przez okresowe wprowadzanie ograniczenia maksymalnej liczby pakietów klasy Cl_0 w pamięci buforowej: $k_{0 \max} < K_0$ wtedy, gdy pakiety z zewnątrz są wprowadzane pojedynczo, oraz $l_{\max} \leq k_{0 \max} < K_0$, jeżeli pakiety są wprowadzane do systemu w postaci wiadomości o maksymalnej długości l_{\max} (rysunek 6.21). Wprowadzenie takich ograniczeń jest równoważne okresowemu zmniejszeniu wartości liczby buforów dostępnych dla klasy Cl_0 .



Rysunek 6.21. Wprowadzanie ograniczenia na maksymalną liczbę pakietów spoza sieci w zagnieżdżonej strukturze podziału pamięci buforowej



Rysunek 6.22. Zamiana zagnieżdżonej struktury podziału pamięci buforowej przez rozłączne podzbiory buforów dedykowane poszczególnym klasom pakietów

Rezygnacja z zagnieżdżonej struktury podziału pamięci buforowej (opartej na zależności: $l_{\max} \leq K_0 < K_1 < \dots < K_{L-1} < K_{L-1} + l_{\max} = K_L = K$) na rzecz jej podziału na

rozłączne podzbiory dedykowane poszczególnym klasom pakietów ($K_0 + K_1 + \dots + K_{L-1} + K_L = K$, $K_0 \geq l_{\max}$ i $K_L \geq l_{\max}$) oznacza wprowadzenie gwarancji określonej przepustowości systemu za cenę zmniejszenia elastyczności i efektywności wykorzystania pojemności pamięci buforowych węzłów (rysunek 6.22).

Ta sama pojemność pamięci buforowej K o dwóch różnych strukturach podziału, tzn. zagnieżdżonej i rozłącznej, stwarza różne możliwości obsługi ruchu teleinformatycznego o danej strukturze. Jeżeli ideę zagnieżdżonej struktury podziału pamięci buforowej można porównywać do idei komutacji pakietów, to wprowadzenie podziału polegającego na wyznaczaniu rozłącznych podzbiorów buforów dedykowanych klasom pakietów można porównać do rezerwacji przepustowości w systemach z komutacją łączy.

6.4.2. STEROWANIE PRZEPIŁYWEM W POŁĄCZENIACH WIRTUALNYCH

6.4.2.1. SIECI DATAGRAMOWE I SIECI POŁĄCZEŃ WIRTUALNYCH

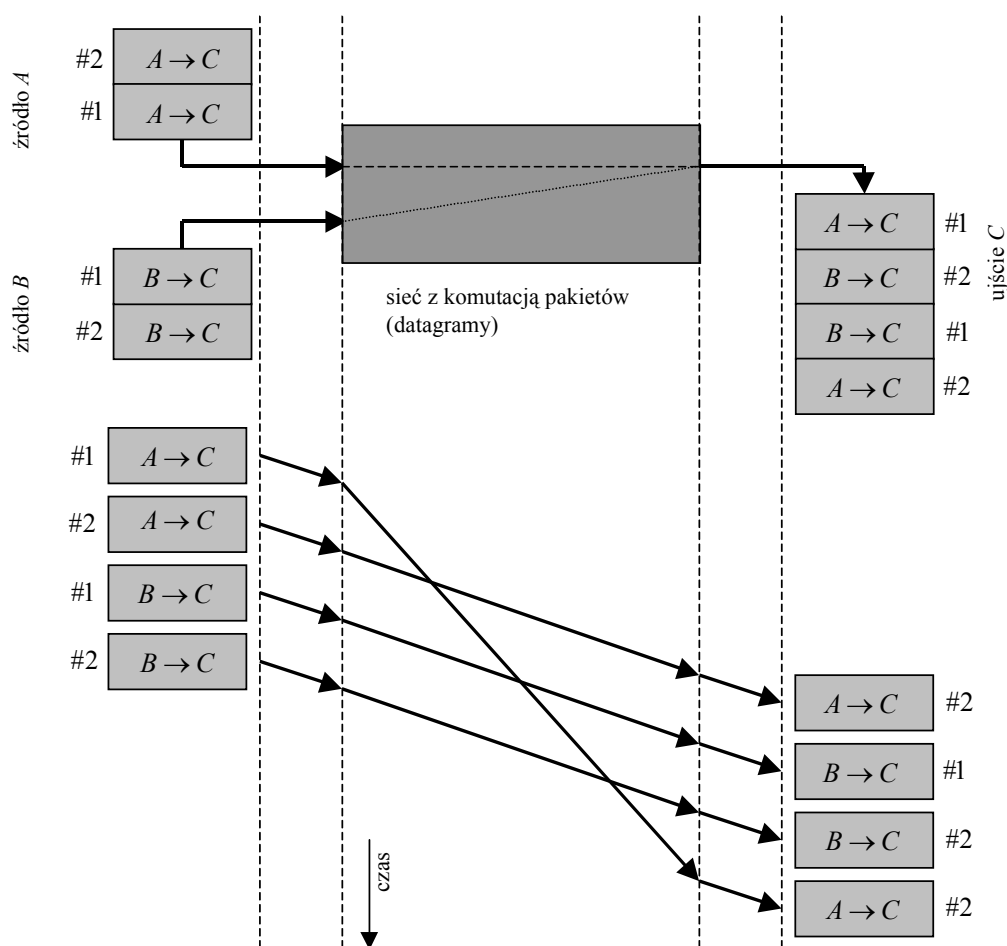
W sieciach komputerowych z komutacją pakietów możliwe jest wyróżnienie dwóch rozłącznych klas rozwiązań: sieci datagramowych (ang. *datagram networks*) i sieci połączeń wirtualnych (ang. *virtual circuit network*).

W pakietowych sieciach datagramowych każdy z pakietów, należący do dowolnej sesji, jest przekazywany w sieci niezależnie od innych pakietów tej samej sesji. Różne pakiety należące do tej samej sesji mogą być przekazywane różnymi trasami i w konsekwencji mogą być dostarczane do węzła końcowego (odbiorczego, docelowego, ujściowego) sieci w kolejności różnej od kolejności nadawania w węzle wejściowym sieci. Sieci datagramowe można traktować jako sposób przekazywania jednostek danych (części wiadomości), wyposażonych w informacje organizacyjne (adresy, pola sterowania, sumy kontrolne itp.), pozwalające na samodzielne i niezależne przemieszczanie się pakietów w sieci, oparty na prawie nieskrępowanej rywalizacji o dostęp do zasobów sieci (rysunek 6.23).

Usługi datagramowe należą do najbardziej kontrowersyjnych rozwiązań stosowanych w sieciach pakietowych. Są one najprostszymi i podstawowymi usługami możliwymi do udostępnienia w sieciach z komutacją pakietów. Implementacja usług datagramowych w sieciach dyktowana jest kilkoma praktycznymi względami, przede wszystkim niskim kosztem organizacji i działania sieci, wynikającym ze stosowania: nielicznych i prostych funkcji zarządzania, prostego interfejsu wymuszającego nieliczne ograniczenia w dostępie do sieci oraz prostych procedur dostępu skracających czas połączenia terminala z siecią itd.

Idea usługi połączenia wirtualnego, przeciwstawiana usłudze datagramowej, jest próbą łączenia korzyści wynikających z prostoty rozwiązań opartych na przekazywaniu danych w postaci niezależnych modułów (jednostek danych) o jednakowej długości oraz łatwości sterowania przepływem w rozwiązaniach wykorzystujących metody

komutacji kanałów. W sieciach połączeń wirtualnych przekazywanie pakietów należących do jednej sesji poprzedza zestawienie połączenia wirtualnego (łącza logicznego), stosowanego następnie do przekazywania pakietów. Połączenie wirtualne jest rodzajem tymczasowego połączenia pomiędzy węzłami źródłowym i ujściowym. Przekazywanie pakietów w ramach danego połączenia wirtualnego może się odbywać wieloma różnymi fizycznymi trasami. Zakończenie sesji powoduje zwolnienie (rozłączenie) połączenia wirtualnego. Ustanawianie (złożone z fazy I – żądania nawiązania połączenia i z fazy II – potwierdzenia nawiązania połączenia) oraz rozłączanie (złożone z fazy IV – żądania rozłączenia i z fazy V – potwierdzenia rozłączenia) połączeń wirtualnych wymaga wymiany przed i po transferze pakietów (transfer pakietów – faza III) nadmiarowych informacji organizacyjnych.



Rysunek 6.23. Przekazywanie jednostek danych w sieciach datagramowych

Połączenia wirtualne mogą być ustanawiane w dwóch formach:

➤ jednorazowych połączeń wirtualnych pomiędzy dwoma węzłami końcowymi, zlokalizowanymi na krańcach sieci, równoważnych implementacji jednokrokowego sterowania przepływem typu koniec–koniec (ang. *end-to-end control*, *source-sink control*), które charakteryzuje prostota i wysoki stopień wykorzystania zasobów,

➤ sekwencji połączeń wirtualnych pomiędzy kolejnymi węzłami pośrednimi na trasie od węzła źródłowego do węzła odbiorczego, równoważnych implementacji wielokrokowego sterowania przepływem typu „łącze po łączy” (ang. *link-by-link control*, *stepwise control*, *hop control*), dla którego charakterystyczna jest duża niezawodność transferu pakietów.

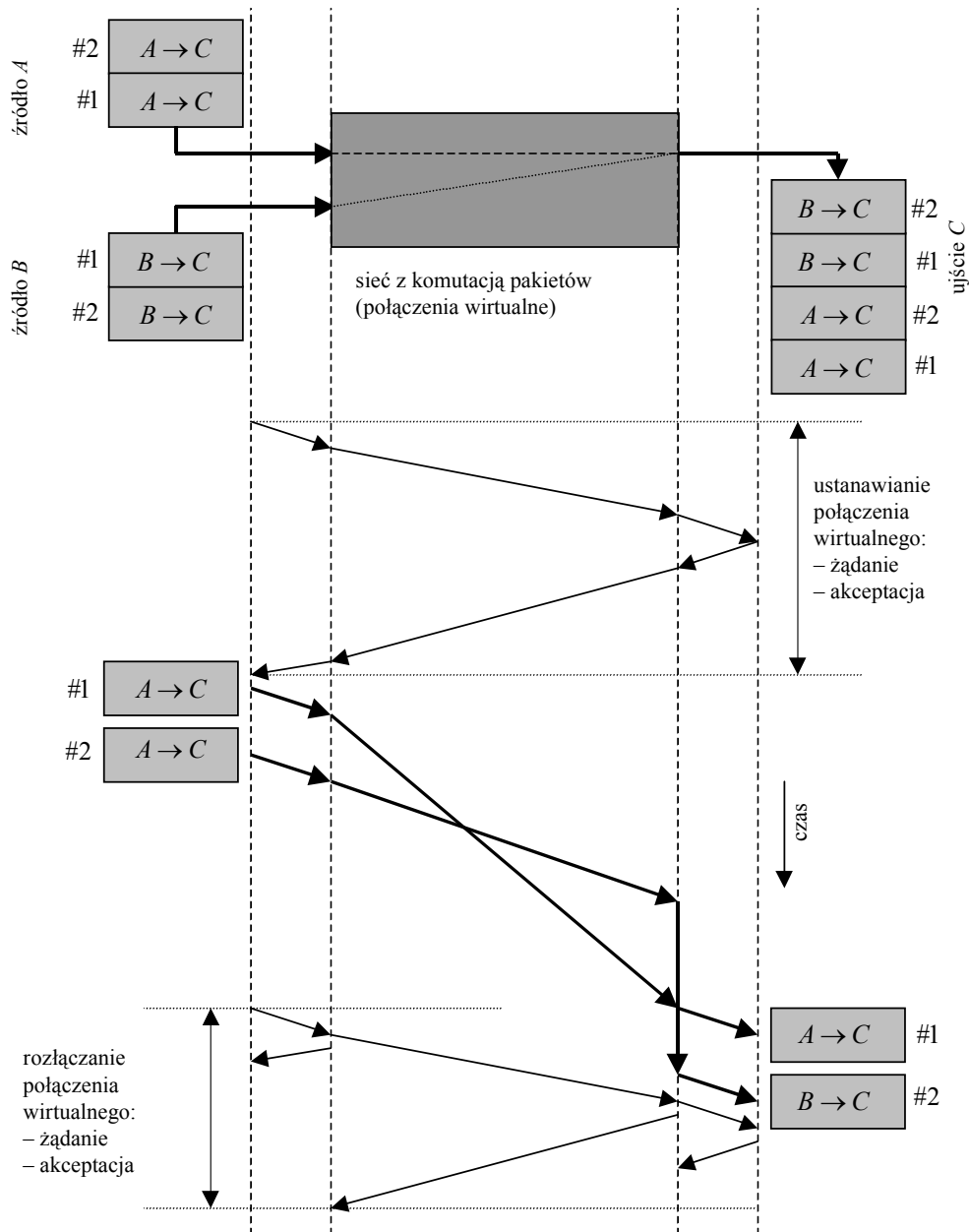
Istnienie stałych (ang. *permanent virtual circuit*) i tymczasowych (ang. *switched virtual circuit*) połączeń wirtualnych, dedykowanych poszczególnym sesjom, umożliwia przekazywanie i dostarczanie pakietów z zachowaniem kolejności nadawania. Zachowanie kolejności pakietów w węzłach pośrednich i węzle końcowym sieci znacznie redukuje ilości zasobów niezbędnych do przekazywania pakietów, pozwala zwłaszcza na redukcję zasobów i rezygnację z mechanizmów koniecznych do odtwarzania kolejności pakietów (rysunek 6.24).

Implementacja usług sieci połączeń wirtualnych jest uzasadniona m.in. – w porównaniu do usług datagramowych – większą niezawodnością usług dzięki istnieniu procedur zarządzania przeciążeniami, łatwiejszym sterowaniem przepływem i kolejnością przepływu wynikającą z istnienia łączy logicznych, wygodniejszym dla użytkownika końcowego sposobem dostarczania pakietów, wynikającym z zachowywania kolejności pakietów itd.

Podział sieci z komutacją pakietów na sieci datagramowe i sieci połączeń wirtualnych nie jest tożsamy z podziałem na sieci, w których przekazywanie pakietów odbywa się w trybie połączeniowym lub bezpołączeniowym. Realizacja przekazywania pakietów w sieci (datagramowej lub wirtualnej) w trybie połączeniowym (ang. *connection oriented*) oznacza, że istnieją w sieci mechanizmy sterowania przepływem gwarantujące niezawodne przekazanie pakietów z węzła nadawczego do węzła ujęciowego (na zasadach determinowanych klasą sieci). Przekaz pakietów w sieci pracującej w trybie bezpołączeniowym (ang. *connectionless*) oznacza brak gwarancji niezawodnego przekazania ich z węzła nadawczego do węzła odbiorczego. W takim przypadku pakiety stracone lub przekłamanie podczas przekazywania w sieci są odtwarzane przez procesy końcowe (poza siecią). Implementacja trybu bezpołączeniowego lub połączeniowego w sieci z komutacją pakietów decyduje o jakości obsługi strumieni pakietów generowanych w procesach końcowych.

Zakresy możliwego sterowania przepływem w sieciach datagramowych i w sieciach połączeń wirtualnych są różne. W sieciach połączeń wirtualnych istnieje możliwość monitorowania i sterowania przepływem w ramach poszczególnych połączeń wirtualnych. Połączenie wirtualne, i związany z nim przepływ, stwarza możliwość implementacji dodatkowego sterowania przepływem na – niedostępnym w sieciach datagramowych – poziomie granulacji strumieni pakietów. Sterowanie przepływem

w połączeniu wirtualnym jest sterowaniem uzupełniającym sterowania przepływem realizowane w węzłach sieci, które korzystają z różnych metod przydziału i podziału pojemności pamięci buforowej oraz pojemności kanałów transmisyjnych.



Rysunek 6.24. Przekazywanie jednostek danych w sieciach z połączeniami wirtualnymi

Dwa różne sposoby organizacji połączeń wirtualnych w sieciach pakietowych, tzn. organizacja jednorazowych połączeń wirtualnych typu koniec–koniec i organizacja sekwencji połączeń typu łącze-po-łączy na trasie od źródła do ujścia, decydują o dwóch rodzajach sterowania przepływem w połączeniach wirtualnych:

➤ krokowe (skokowe), którego zadaniem jest sterowanie przepływem w sekwencji części połączenia wirtualnego (wyznaczanych kolejnymi węzłami sieci na trasie) w celu zapobiegania przeciążeniom pamięci buforowych w węzłach,

➤ jednorazowe, którego zadaniem jest sterowanie przepływem w połączeniu wirtualnym, traktowanym jako całość, w celu takiego doboru szybkości transferu pakietów, aby jednocześnie maksymalizować rzeczywistą przepustowość połączenia wirtualnego i zapobiegać przeciążeniom pamięci buforowej węzła odbiorczego.

Dwojaka organizacja połączeń wirtualnych powoduje, że metody sterowania przepływem w połączeniach wirtualnych można zaliczać do grup metod sterowania przepływem w łączy danych lub sterowania przepływem na trasie. W szczególnych przypadkach sterowanie przepływem w połączeniu wirtualnym jest dwupoziomowe. Zakres sterowania na poziomie nadrzędnym dotyczy połączenia wirtualnego, zakres sterowania przepływem na poziomie podrzędnym obejmuje natomiast kolejne fragmenty połączenia wirtualnego.

Etapowość oddziaływania metod sterowania krokowego wynika z tego, że decyzje sterujące mają charakter decyzji lokalnych; są podejmowane na podstawie informacji o stanie pamięci buforowych węzłów pośrednich na trasach pomiędzy węzłami końcowymi połączenia wirtualnego i dotyczą intensywności transferu pakietów w łączy pomiędzy dwoma sąsiednimi węzłami. Lokalność oddziaływania metod sterowania krokowego na przepływ powoduje, że są to metody sterowania przepływem na poziomie łącza danych.

Metody jednorazowego sterowania przepływem mają cechy charakterystyczne sterowania dostępem do sieci – są implementowane w węzłach wejściowych sieci, a decyzje sterujące dotyczą przepływu traktowanego jako jednoetapowy transfer pakietów pomiędzy węzłami końcowymi połączenia wirtualnego.

6.4.2.2. KROKOWE STEROWANIE PRZEPLÝWEM W POŁĄCZENIU WIRTUALNYM

Podstawowa zasada działania metod krokowego sterowania przepływem w połączeniu wirtualnym polega na wyborze maksymalnej liczby pakietów M_{ij} , należących do danego i -tego połączenia wirtualnego, które w danej chwili mogą znajdować się w pamięci buforowej danego j -tego węzła pośredniego na trasie pomiędzy węzłami nadawczym i odbiorczym. Wartość M_{ij} ($i = 1, 2, \dots, I, j = 1, 2, \dots, N$, gdzie I jest liczbą wszystkich aktualnie utrzymywanych połączeń wirtualnych, a N jest liczbą wszystkich węzłów sieci) może być stała i dobierana w czasie ustanawiania połączenia wirtualnego w sieci lub zmienna, ustalana w zależności od aktualnego stanu zajętości pamięci buforowych węzłów i zmian zachodzących w obciążeniu sieci. Wprowadze-

nie ograniczenia liczby pakietów danego połączenia wirtualnego w danym węźle sieci oznacza, że działanie odpowiedniego protokołu sterowania polega na akceptacji lub odrzuceniu pakietów w kolejnych węzłach na trasie.

Przewaga metod krokowego sterowania przepływem w połączeniu wirtualnym nad wcześniej opisanymi metodami podziału pojemności pamięci buforowej (sterowanie długością kolejek wyjściowych i strukturalizacja pojemności pamięci buforowej) polega na większej efektywności i krótszym czasie wyjścia ze stanu przeciążenia.

Ze względu na to, że przedmiotem sterowania w węzłach są wyłącznie strumienie pakietów jednoznacznie identyfikowane numerem połączenia wirtualnego (tzn. także adresem źródłowym tego strumienia), możliwa jest szybka i efektywna:

- identyfikacja połączeń wirtualnych, których strumienie pakietów powodują lokalny stan przeciążenia,
- decyzja o spowolnieniu szybkości napływu strumieni pakietów zidentyfikowanych połączeń wirtualnych do obszaru przeciążenia,
- wsteczna propagacja decyzji sterujących o zmniejszeniu intensywności napływu strumieni pakietów z węzła źródłowego połączenia wirtualnego.

Względnie prosta, szybka i jednoznaczna identyfikacja połączeń wirtualnych, których przepływy przechodzą przez węzeł będący w stanie przeciążenia, umożliwiają szybką i skuteczną propagację decyzji sterującej o spowolnieniu wyłącznie do źródeł tych przepływów, tzn. bez zakłócania pracy pozostałych. Możliwość jednoznacznego wskazania źródeł przeciążenia powoduje, że zapobieganie rozprzestrzenianiu się stanu przeciążenia, a następnie jego likwidacja odbywa się bez blokowania ruchu generowanego w źródłach nieodpowiadających bezpośrednio za powstanie przeciążenia. Brak ingerencji w pracę źródeł innych niż odpowiedzialne za przeciążenie oznacza, że nie powstają warunki do zbędnej degradacji przepustowości i niesprawiedliwego podziału zasobów.

Podjęcie i realizacja decyzji o spowolnieniu (dławieniu) napływu pakietów do węzła będącego w stanie przeciążenia, w przypadku sterowania przepływem klas pakietów (pakiety o jednakowym czasie pobytu w sieci) lub długością kolejek wyjściowych (pakiety kierowane do tego samego łącza danych), jest zadaniem znacznie bardziej złożonym, ponieważ strumienie pakietów, będące przedmiotem sterowania, zawierają pakiety pochodzące z różnych źródeł. Podjęcie decyzji o spowolnieniu napływu pakietów wymaga w tym przypadku nie tylko identyfikacji źródeł strumieni pakietów powodujących przeciążenie, ale także oszacowania wpływu poszczególnych strumieni pakietów na powstanie stanu przeciążenia.

Złożoność tego zadania oraz konieczność jego rozwiązywania w czasie rzeczywistym powodują, że podejmowane decyzje nie są optymalne i obniżają efektywność wykorzystania pojemności pamięci buforowych. Brak jednoznacznej i szybkiej identyfikacji źródeł przeciążenia powoduje, że sposobem likwidacji stanu przeciążenia jest blokowanie ruchu dotyczące nie tylko źródeł odpowiedzialnych bezpośrednio za ten stan. Takie działanie powoduje pogorszenie rzeczywistej przepustowości systemu i zaburza sprawiedliwy podział zasobów pomiędzy obsługiwane procesy transferu pakietów.

W metodach sterowania przepływem w połączeniach wirtualnych, podobnie jak w omawianych przykładach sterowania długością kolejek wyjściowych czy sterowania klasami pakietów, możliwe są różne strategie podziału pojemności pamięci buforowych. Różnica polega na tym, że liczby kolejek wyjściowych i klas pakietów są stałe, podczas gdy liczba połączeń wirtualnych obsługiwanych w węzłach jest zmienna i trudno przewidywalna. W praktycznych realizacjach możliwe jest co najwyżej szacowanie chwilowej, maksymalnej liczby połączeń wirtualnych. Zakres zmienności liczby połączeń wirtualnych w danym węźle zależy od wielu czynników: od wielkości sieci (liczby węzłów i łączy międzywęzłowych), struktury topologicznej sieci, stosowanych w sieci reguł doboru tras, liczby stałych i tymczasowych połączeń wirtualnych, względnego udziału tymczasowych połączeń wirtualnych w liczbie wszystkich połączeń wirtualnych itd. Na wybór jednej z możliwych strategii podziału pojemności pamięci buforowej wpływ może mieć także czas (średni, maksymalny) utrzymywania połączeń wirtualnych, determinowany charakterystykami ruchu generowanego przez źródła oraz iloraz czasu wykorzystywania połączenia wirtualnego do sumy czasów ustanawiania i rozłączania połączeń wirtualnych w danej sieci.

Dwie skrajnie różne strategie alokacji pojemności pamięci buforowej węzłów dla połączeń wirtualnych to:

- kierowanie części pojemności pamięci buforowej do połączeń wirtualnych, obejmowanych w czasie ustanawiania (negocjowania wartości parametrów) połączenia wirtualnego,
- dokonywany na żądanie przydział części współdzielonej puli pojemności pamięci buforowej.

Strategia dedykowania buforów jest strategią prostą, której wadą jest niezapobieganie możliwości nieefektywnego wykorzystywania pojemności pamięci buforowej, a zaletą – gwarancja obsługi wolnej od zakleszczeń.

Utrzymywanie pamięci buforowych o pojemnościach obliczonych na obsługę maksymalnej, możliwej w sieci, liczby połączeń wirtualnych gwarantuje jakość obsługi połączeń za cenę niskiej efektywności wykorzystania pamięci. Ewentualne ograniczanie pojemności pamięci buforowych, obliczone na obsługę średniej liczby połączeń i dyktowane potrzebą efektywności jej wykorzystania, nie gwarantuje jakości obsługi połączeń w razie znaczącego zwiększenia ich liczby. Co więcej, ponieważ intensywność strumieni pakietów generowanych w ramach połączeń wirtualnych jest na ogół zmienna, różne zapotrzebowanie na pojemności pamięci buforowych udostępnianych na zasadach rezerwacji jest więc kolejną przyczyną niewielkiej efektywności zasobów.

Strategia kierowania buforów do połączeń wirtualnych eliminuje możliwość powstawania zakleszczeń. Dedykowanie określonej pojemności pamięci buforowej każdemu połączeniu wirtualnemu w każdym węźle sieci oznacza, że rzeczywista przepustowość sieci dla każdego połączenia wirtualnego jest różna od zerowej, nie ma więc warunków powstawania zakleszczeń bezpośrednich i pośrednich.

Mała efektywność wykorzystania zasobów w strategiach dedykowania buforów powoduje, że – mimo zapobiegania powstawaniu zakleszczeń – jest strategią rzadko

stosowaną w praktycznych rozwiązaniach sieciowych. Częściej stosowanym rozwiązaniem zadania przydziału pojemności pamięci buforowej dla połączeń wirtualnych jest przydział na żądanie. Strategie przydziału na żądanie dają gwarancję lepszej efektywności wykorzystania ograniczonej pojemności pamięci buforowej, nie gwarantując jednocześnie możliwości zakleszczeń. Implementacja strategii przydziału na żądanie wiąże się w ten sposób z koniecznością stosowania uzupełniających mechanizmów zapobiegających powstawaniu zakleszczeń.

6.4.2.3. PRZYKŁADOWE IMPLEMENTACJE METOD STEROWANIA PRZEPLYWEM W POŁĄCZENIACH WIRTUALNYCH

6.4.2.3.1. STEROWANIE PRZEPLYWEM W POŁĄCZENIACH WIRTUALNYCH SIECI TYMNET

Transferowane dane, należące do różnych połączeń wirtualnych, mogą być przekazywane w jednym pakiecie. Uzasadnieniem takiej organizacji przepływu jest możliwość redukcji ilości informacji organizacyjnych potrzebnych do organizacji przepływu i jego sterowania. Sieć TYMNET jest siecią zorientowaną znakowo w tym znaczeniu, że podstawową jednostką organizacyjną jest znak. Na potrzeby transferu między węzłami znaki (ang. *assembly*) są składane w pakiety w węzłach nadawczych; w węzłach odbiorczych następuje ich rozformowanie (ang. *disassembly*). Rozformowanie pakietów odbywa się także w węzłach pośrednich, w których organizacja pamięci buforowych pozwala przyjmować i przechowywać znaki.

Podczas ustanawiania nowego połączenia wirtualnego każdemu jest wyznaczana górna graniczna wartość przepustowości, która zależy od szybkości generowania ruchu w źródle obsługiwanych przez dane połączenie wirtualne. Wyznaczona przepustowość jest rezerwowana dla i -tego połączenia wirtualnego na całej trasie pomiędzy węzłami: nadawczym i odbiorczym. Przepustowością steruje się przez przypisanie każdemu połączeniu wirtualnemu maksymalnej liczby buforów (pojemność bufora jest równa jednemu znakowi) K_i ($i = 1, 2, \dots$), których dostępność w węźle następnym jest warunkiem wydania pozwolenia na transfer z węzła poprzedniego.

Okresowo każdy z węzłów wysyła do wszystkich węzłów sąsiednich informację o wszystkich obsługiwanych połączeniach wirtualnych. Informacja ta ma postać wektora propagacji wstecznej (ang. *backpressure vector*), w którym na pozycji odpowiadającej poszczególnym połączeniom wirtualnym b_i mogą być umieszczone wartości 1 lub 0. Jeżeli liczba buforów w węźle, aktualnie zajętych przez dane połączenie wirtualne, przekracza pojemność pamięci buforowej przydzielonej połączeniu wirtualnemu, to odpowiedni bit wektora przyjmuje wartość 0: $b_i = 0$, w innym przypadku wartość odpowiedniego bitu wynosi 1: $b_i = 1$.

Każde i -te połączenie wirtualne ma w węźle nadawczym swój licznik V_i , którego stan początkowy jest równy maksymalnej liczbie buforów K_i przypisanych danemu połączeniu wirtualnemu: $V_i = K_i$. Wysłanie każdego znaku oznacza zmniejszenie stanu

licznika o jeden. Jeżeli stan licznika osiągnie wartość zerową ($V_i = 0$), to transmisja znaków w danym połączeniu jest wstrzymywana. Po otrzymaniu wektora wstecznej propagacji z bitem $b_i = 1$ następuje zmiana stanu licznika na stan początkowy i transmisja znaków jest wznowiana. Bit $b_i = 1$ nie jest bitem aktywnym, jeżeli stan licznika jest różny od zera, tj. jeżeli $V_i \neq 0$.

Informacja o monitorowanym stanie zajętości pojemności pamięci buforowej zarezerwowanej w każdym węźle trasy dla danego połączenia wirtualnego jest przekazywana do źródła i może powodować wznowienie transmisji po wcześniejszym jej wstrzymaniu, spowodowanym wyczerpaniem limitu liczby znaków wprowadzanych jednorazowo do połączenia wirtualnego. Sterowanie przepływem w poszczególnych fragmentach połączenia wirtualnego jest osiągnięte przez wstrzymywanie wprowadzania znaków do połączenia wirtualnego.

Omawiana procedura sterowania przepływem jest hybrydowa, w tym znaczeniu, że oddziałuje zarówno na przepływ pomiędzy węzłami sąsiednimi, jak i na przepływ w całym połączeniu wirtualnym. Jest więc złożeniem sterowań przepływem na poziomie łącza danych i na poziomie trasy pomiędzy węzłami wejściowym i wyjściowym. Procedura ta jest przykładem rozwiązań często stosowanych w sieciach, w których jedna procedura sterowania, obejmując swym zakresem działania różne fragmenty przepływu, jest wykorzystywana do realizacji kilku celów sterowania.

6.4.2.3.2. STEROWANIE PRZEPŁYWEM W POŁĄCZENIACH WIRTUALNYCH SIECI TRANSPAC

Do sterowania przepływami i przeciążeniami w sieci połączeń wirtualnych TRANSPAC stosowany jest protokół X.25 (jako protokół międzywęzłowy) i charakterystyczna dla niego koncepcja klas przepustowości.

Każde żądanie utworzenia i -tego połączenia wirtualnego ($i = 1, 2, \dots$) zawiera informację o deklarowanej klasie przepustowości. Podczas ustanawiania połączenia wirtualnego informacja ta jest przenoszona do wszystkich węzłów na trasie. Deklaracja klasy przepustowości i -tego połączenia to deklaracja maksymalnej chwilowej szybkości transferu danych przez źródło generujące żądanie utworzenia danego połączenia wirtualnego: $v_{i \max}$.

W każdym j -tym węźle sieci ($j = 1, 2, \dots, N$) jest utrzymywana informacja o zagregowanej (sumarycznej) deklarowanej ($V_{j \max}$) i rzeczywistej (V_j) przepustowości wszystkich połączeń wirtualnych przechodzących przez ten węzeł:

$$V_{j \max} = \sum_{i \in \pi_j} v_{i \max} \quad \text{i} \quad V_j = \sum_{i \in \pi_j} v_i ,$$

gdzie:

π_j – zbiór wszystkich połączeń wirtualnych, których trasy zawierają j -ty węzeł,

v_i – rzeczywista przepustowość i -tego połączenia wirtualnego.

Oprócz sum deklarowanych i rzeczywistych przepustowości każdy z węzłów charakteryzuje średnia wartość współczynnika wykorzystania pojemności pamięci buforowej węzła u_j .

Konieczność obsługi sumy deklarowanych klas przepustowości w j -tym węźle ($V_{j \max}$) można interpretować jako wystąpienie najgorszego przypadku. Transfer we wszystkich połączeniach wirtualnych przechodzących przez j -ty węzeł odbywa się z maksymalną deklarowaną szybkością. Iloraz sum rzeczywistych i deklarowanych przepustowości określa chwilową wartość stopnia wykorzystania rezerwowanych (deklarowanych) klas przepustowości połączeń wirtualnych.

Ze względu na definicję deklarowanej przepustowości połączeń wirtualnych zwykle w węźle jest spełniona zależność: $V_j \ll V_{j \max}$. Oznacza to zasadność i możliwość przyjmowania do obsługi w sieci połączeń wirtualnych, których suma deklarowanych przepustowości przewyższa maksymalną przepustowość. Możliwość przyjęcia do jednoczesnej obsługi w sieci połączeń wirtualnych, których suma szybkości transferu jest większa od maksymalnej przepustowości, oznacza z kolei, że w sieci musi być implementowany mechanizm sterowania przepływem.

Mechanizm spowalniania lub wstrzymywania ruchu napływającego w połączeniach wirtualnych do poszczególnych węzłów jest następujący:

1. Alokacja napływających pakietów do buforów w węźle odbywa się na żądanie.
2. Pojemność pamięci buforowej j -tego węzła jest udostępniana połączeniom wirtualnym zgodnie ze strategią podziału całkowitego CS (ang. *Complete Sharing*), tzn. długości kolejek k_{ji} i maksymalne długości kolejek K_{ji} poszczególnych połączeń wirtualnych spełniają warunki:

$$0 \leq k_{ji} \leq K_{ji}, \quad 0 \leq K_{ji} \leq K_j \quad \text{ i } \quad \sum_{i=1}^{n_j} K_{ji} \leq K_j,$$

gdzie:

K_j – całkowita pojemność pamięci buforowej j -tego węzła,

n_j – liczba aktualnie obsługiwanych połączeń wirtualnych.

Suma zajętych buforów jest oznaczana przez k_j :

$$k_j = \sum_{i=1}^{n_j} k_{ji}.$$

3. Definiowane są trzy wartości progowe sumy zajętości pojemności pamięci buforowej: $K_j^{(1)}$, $K_j^{(2)}$ i $K_j^{(3)}$ spełniające warunek:

$$K_j^{(1)} < K_j^{(2)} < K_j^{(3)} < K.$$

4. Sterowanie przepływem realizowane w j -tym węźle, polegające na stopniowaniu restrykcji wobec nowo zgłaszanych oraz już istniejących i obsługiwanych połączeń wirtualnych, odbywa się według następującej procedury:

➤ jeżeli $k_j \leq K_j^{(1)}$, to nowe połączenia wirtualne są ustanawiane, a szybkość napływu pakietów w obsługiwanych połączeniach nie podlega żadnym ograniczeniom,

➤ jeżeli $K_j^{(1)} < k_j \leq K_j^{(2)}$, to nowe połączenia wirtualne, których żądania ustanowienia napływają do j -tego węzła, nie są akceptowane (brak możliwości ustanowienia nowego połączenia wirtualnego przechodzącego przez j -ty węzeł),

➤ jeżeli $K_j^{(2)} < k_j \leq K_j^{(3)}$, to nowe połączenia wirtualne nie są przyjmowane do obsługi w węźle, a napływ pakietów w dotychczas obsługiwanych połączeniach wirtualnych jest spowalniany (przez opóźnianie wysyłania potwierdzeń),

➤ jeżeli $K_j^{(3)} < k_j$, to wybrane połączenia wirtualne obsługiwane w j -tym węźle są rozłączane.

Wartości parametrów $K_j^{(1)}$, $K_j^{(2)}$ i $K_j^{(3)}$ są obliczane dynamicznie i zależą od wartości deklarowanych klas przepustowości oraz aktualnych wartości rzeczywistych przepustowości połączeń wirtualnych oraz aktualnego stanu zajętości pamięci buforowej węzła.

6.4.2.3.3. STEROWANIE PRZEPLYWEM W POŁĄCZENIACH WIRTUALNYCH SIECI GMDNET

Procedura sterowania przepływem w sieci połączeń wirtualnych GMDNET jest dwupoziomowa. Przedmiotem sterowania są przepływy w poszczególnych połączeniach wirtualnych (poziom podrzędny) oraz suma przepływów w poszczególnych węzłach sieci (poziom nadrzędny).

Do sterowania przepływem w węzłach sieci są stosowane procedury wykorzystujące metody strukturalizacji pamięci buforowej. W każdym węźle pakiety są dzielone na klasy ze względu na ich czas przebywania w sieci, a w przypadku powstania przeciążenia wyższy priorytet obsługi mają pakiety należące do klas wyższych.

Indywidualne sterowanie przepływem w każdym z połączeń wirtualnych (ang. *individual flow control*) jest hierarchiczne. Na poziomie wyższym przedmiotem sterowania jest przepływ w połączeniu wirtualnym, traktowany jako całość (poziom trasy), na poziomie niższym natomiast sterowanie dotyczy przepływu pomiędzy kolejnymi parami węzłów na trasie obsługującej dane połączenie wirtualne (poziom łącza danych).

Do sterowania na poziomach trasy i łącza danych stosuje się metodę okna (ang. *window method*). Rozmiar okna definiuje maksymalną liczbę pakietów, które mogą być wysłane przez węzeł nadawczy w oczekiwaniu na potwierdzenie ich odbioru przez węzeł odbiorczy, tzn. maksymalną liczbę pakietów wysłanych i niepotwierdzonych. Wznowienie nadawania jest możliwe po otrzymaniu potwierdzenia wcześniej wysłanych pakietów (jednego, kilku lub wszystkich), z tym że liczba pakietów, jakie mogą być wysłane po wznowieniu nadawania, zależy od numeru pakietu (pakietów) w oknie wskazanego w ostatnio otrzymanym potwierdzeniu.

Maksymalne rozmiary okna w (ang. *window size*) na dwóch wyróżnionych poziomach sterowania są na ogół różne: w_E – do sterowania przepływem na poziomie trasy (ang. *end-to-end control*) oraz w_L – do sterowania przepływem na poziomie łącza danych (ang. *hop level*). Sterowanie przepływem oznacza ustalenie wartości rozmiarów okien PUL (ang. *Packet Underway Limit*) dla przepływów w połączeniach wirtualnych PUL_i , gdzie $i = 1, 2, \dots$ jest indeksem połączenia wirtualnego oraz dla przepływów w poszczególnych łączach danych danego połączenia wirtualnego PUL_{ikl} , gdzie $(k, l) \in \pi_i$, $k \neq l$, a π_i jest zbiorem par numerów węzłów i -tej trasy.

Rozmiary okien PUL_i spełniające, dla każdego $i \in 1, 2, \dots$, warunek: $PUL_i \leq w_E$, mogą być:

➤ jednakowe dla wszystkich połączeń wirtualnych, tzn. $PUL_i = PUL_E$ dla każdego $i = 1, 2, \dots$,

➤ różne dla różnych połączeń wirtualnych, tzn. $PUL_i \neq PUL_j$ dla $i \neq j$ oraz $i, j = 1, 2, \dots$, dobierane dynamicznie w zależności od obciążenia sieci, obciążenia tras w sieci, obciążenia węzłów sieci, priorytetu obsługi itd.

Rozmiary okien PUL_{ikl} spełniające, dla każdego i -tego ($i = 1, 2, \dots$) połączenia wirtualnego oraz wszystkich łączy danych (k, l) należących do trasy π_i , warunki: $PUL_{ikl} \leq w_L$, mogą być:

➤ jednakowe we wszystkich łączach danych dla przepływów wszystkich połączeń wirtualnych, tzn. $\forall_{i=1,2,\dots} \forall_{(k,l) \in \pi_i} PUL_{ikl} = PUL_L$,

➤ jednakowe we wszystkich łączach danych dla przepływów i -tego ($i = 1, 2, \dots$) połączenia wirtualnego, tzn. $\forall_{i=1,2,\dots} PUL_{ikl} = PUL_{iL}$, gdzie $(k, l) \in \pi_i$ oraz $k \neq l$,

➤ różne we wszystkich łączach danych dla przepływów wszystkich połączeń wirtualnych.

Metoda okna szczególnie efektywnie zapobiega przeciążeniom na poziomie trasy wtedy, gdy szybkość wprowadzania pakietów do sieci przez źródło jest większa od szybkości wyprowadzania pakietów z sieci w ujściu. Dostosowanie szybkości wprowadzania pakietów do szybkości wyprowadzania pakietów tak, aby liczba pakietów niepotwierdzonych, będących w łączu danych połączenia wirtualnego lub połączeniu wirtualnym sieci, była mniejsza niż rozmiar okna, odbywa się przez wysyłanie z węzła ujściowego powiadomień (negatywnych lub pozytywnych), których odbiór w węzle źródłowym decyduje o przesuwaniu, wstrzymaniu przesuwania oraz o szybkości przesuwania okna w węzle nadawczym.

Indywidualne, hierarchiczne sterowanie przepływem w połączeniach wirtualnych i fragmentach tych połączeń metodą okna, bez nadrzędnego sterowania podziałem pojemności pamięci buforowej, nie zapobiega zakleszczeniom, degradacji przepustowości oraz niesprawiedliwemu podziałowi zasobów. Korzyści ze stosowania metody okna pojawiają się wtedy, gdy ten sposób sterowania przepływem wspiera nadrzędną procedurę przyjmowania i odrzucania pakietów, determinowaną ograniczonością zasobów pamięci buforowej.

Utrzymywanie liczby pakietów (wysłanych do sieci i dotychczas niepotwierdzonych przez węzły odbiorcze) poniżej ustalonej granicy (na poziomach: sieci, połączenia wirtualnego i łącza danych) powoduje, że w razie wystąpienia konieczności odrzucenia pakietów w czasie obsługi (w sieci, w połączeniu wirtualnym, w łączu danych) liczba odrzucanych pakietów jest ograniczona. Ograniczona z góry liczba ewentualnie odrzucanych pakietów oznacza, że ilość zasobów zaangażowanych w obsługę odrzucanych pakietów (ilość zasobów traconych) jest również ograniczona; może być kontrolowana przez dobór długości okien. Możliwość oddziaływania na ilość traconych zasobów (proporcjonalna do rozmiaru okna i zależna od długości tras w sieci) oznacza

także możliwość kontrolowania ilości traczonej przepustowości w sieci, tzn. stopnia degradacji przepustowości. Zadanie zachowania ustalonego stopnia degradacji przepustowości w danej sieci polega na wyznaczeniu rozmiarów okien.

Omawiana procedura dwupoziomowa sterowania przepływem pozwala na połączenie zalet metody strukturalizacji pamięci buforowej węzła oraz sterowania przepływem na trasie i w jej poszczególnych fragmentach. W wyniku połączenia otrzymuje się procedurę zapobiegającą powstawaniu zakleszczeń (właściwość metody strukturalizacji pamięci buforowej) oraz znaczącej degradacji przepustowości (właściwość indywidualnego sterowania przepływem w połączeniu wirtualnym). Należy zwrócić uwagę na to, że metody strukturalizacji pamięci buforowej zapobiegają całkowitej degradacji przepustowości. Maksymalny stopień możliwej degradacji przepustowości zależy od sposobu podziału pojemności pamięci pomiędzy różne klasy pakietów. W dwustopniowej procedurze indywidualne sterowanie przepływem w połączeniu wirtualnym (na poziomie trasy) zmniejsza maksymalny stopień degradacji przepustowości charakterystyczny dla metod strukturalizacji pamięci buforowej, które wykorzystują indywidualne sterowanie przepływem w połączeniu wirtualnym do wstecznej propagacji decyzji o spowolnieniu lub wstrzymaniu napływu do sieci nowych pakietów.

Procedura ta może gwarantować sprawiedliwy podział zasobów tylko w ograniczonym zakresie, w jakim możliwe jest skoordynowane wyznaczanie rozmiarów okien dla wszystkich połączeń wirtualnych obsługiwanych w danej chwili w sieci. Sprawiedliwy podział zasobów nie jest możliwy bez uzupełnienia omawianej metody o nadrzędny mechanizm kontroli liczby wszystkich pakietów wprowadzanych do sieci, tzn. mechanizm sterowania przepływem na poziomie sieci.

6.4.2.3.4. STEROWANIE PRZEPLYWEM W POŁĄCZENIACH WIRTUALNYCH SIECI CODEX

Dla każdego połączenia wirtualnego organizowanego w sieci jest wyznaczany rozmiar okna, mierzony w bajtach, proporcjonalny do liczby łączy danych składających się na połączenie wirtualne i nominalnej w nim szybkości transferu.

Przekazywanie potwierdzeń zwrotnych z węzła ujściowego do węzła źródłowego łączy się z przekazywaniem informacji wymaganych na potrzeby detekcji błędów, dlatego mechanizm potwierdzeń nie wnosi dodatkowego ruchu w połączeniach pomiędzy węzłami końcowymi.

Ramki DLC (ang. *Data Link Control*) są formowane w węzłach początkowych łączy danych z grup bajtów należących do różnych połączeń wirtualnych, których kolejki wejściowe w danym węźle nie są puste. Liczba bajtów danego połączenia wirtualnego w każdym węźle początkowym łączy danych połączenia wirtualnego jest z góry ograniczona. Istnienie maksymalnego rozmiaru grupy bajtów połączenia wirtualnego oznacza, że każdemu połączeniu wirtualnemu jest przydzielana ograniczona z góry pojemność pamięci buforowej. Obsługa połączeń wirtualnych w węzłach początkowych łączy, tzn. kompletowanie ramek DLC, odbywa się zgodnie z algorytmem cyklicznym, który gwarantuje sprawiedliwy rozdział zasobów komu-

nikacyjnych pomiędzy połączenia wirtualne, współzawodniczące o zasoby komunikacyjne w węzłach.

Sterowanie przepływem metodą okna jest uzupełniane dodatkowym mechanizmem, którego celem jest dławienie szybkości transferu w połączeniu wirtualnym w razie dużego obciążenia sieci. Dławienie szybkości transferu w danym połączeniu wirtualnym polega na wprowadzeniu górnego ograniczenia na liczbę bajtów połączenia wirtualnego w ramce DLC.

W razie konieczności spowolnienia szybkości transferu połączenia wirtualnego w łączy danych węzeł początkowy danego łączy danych generuje specjalną wiadomość, która jest wysyłana do węzła poprzedzającego na trasie połączenia wirtualnego. Odebranie takiej wiadomości powoduje ograniczenie liczby bajtów połączenia wirtualnego w ramce DLC. Krokowe powtarzanie tej procedury od węzła do węzła łączy danych, zlokalizowanych na trasie połączenia wirtualnego, zapobiega powstawaniu przeciążeń w wyniku dławienia szybkości generowania bajtów w węźle źródłowym połączenia wirtualnego. Wprowadzony sposób dławienia szybkości przepływu oraz względnie duże pojemności pamięci buforowych w węzłach redukują prawdopodobieństwo przepełnienia pojemności pamięci buforowych.

6.4.3. STEROWANIE PRZEPLYWEM NA POZIOMIE TRASY

6.4.3.1. ZAPOBIEGANIE PRZECIĄŻENIOM NA TRASIE

Implementacja mechanizmów zapobiegania przeciążeniom jest jednym z podstawowych zadań projektowania sieci komputerowych z komutacją wiadomości (w ogólnym przypadku – jednostek danych). Przeciążenie powstaje wtedy, gdy ilość ruchu wprowadzanego do sieci przekracza ilość, jaka w danej chwili może być w niej obsłużona. Źródłem różnic pomiędzy ilością wprowadzanego ruchu i możliwego do obsłużenia są albo przyrosty generowanego ruchu, albo pogorszenie przepustowości sieci, powodowane m.in. awariami jej elementów.

Proponowane są różne mechanizmy zapobiegania przeciążeniom sieci, które – z punktu widzenia przepływu w sieci – można zaliczyć do różnych klas. Jednym ze sposobów podziału mechanizmów zapobiegania przeciążeniom jest podział na:

- mechanizmy globalne, stosowane zwykle na poziomie dostępu do sieci (wprowadzane ograniczenia przepływów dotyczą wszystkich wiadomości w sieci, bez względu na ich cechy charakterystyczne, takie jak np. źródła i ujścia ruchu),
- mechanizmy pośrednie, stosowane zwykle na poziomie tras (wprowadzane, niezależnie od siebie, ograniczenia przepływów dotyczą grup wiadomości identyfikowanych źródłami i ujściami ruchu),
- mechanizmy lokalne, zwykle wspierające sterowanie przepływem na poziomie dostępu do sieci lub na poziomie tras (wprowadzane ograniczenia przepływów dotyczą wybranych fragmentów sieci, którymi są albo węzły – sterowanie pamięcią buforową, albo łączy – sterowanie przepustowością).

Sterowanie przepływem na trasie może być realizowane na dwa sposoby:

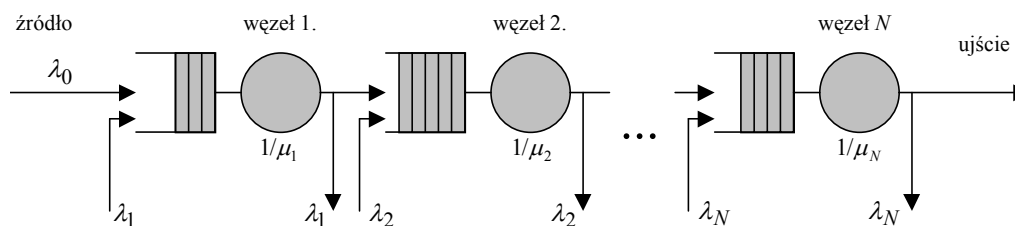
- sterowanie oddziałujące bezpośrednio na źródło ruchu wiadomości kierowanych na trasę do ujścia (przepływ na trasie jest traktowany jako całość),
- sekwencja sterowań oddziałujących lokalnie na poszczególne fragmenty trasy (przepływ na trasie jest traktowany jako sekwencja przepływów lokalnych).

6.4.3.2. MODELE KOLEJKOWE

Zapobiegania przeciążeniom może być formułowane jako zadanie zapobiegania przeciążeniom powodowanym przez źródła generujące ruch większy od pewnej nominalnej wartości. W sieci z komutacją wiadomości i ze stałą regułą doboru tras wiadomości przekazywane pomiędzy ustalonymi parami węzłów źródłowych i ujściowych są transferowane po ustalonych trasach, będących ciągami węzłów połączonych kanałami transmisyjnymi.

Tak definiowaną trasę można utożsamiać z połączeniem logicznym źródła i ujścia. Modelem połączenia logicznego jest ciąg pojedynczych systemów kolejkowych, w którym węzły są reprezentowane przez kolejki organizowane w pomieszczeniach buforowych, a kanały transmisyjne – przez stanowiska obsługi (serwery). Wiadomości generowane przez źródło połączenia logicznego, zwane wiadomościami wewnętrznymi trasy, są wprowadzane do pierwszego systemu kolejkowego (węzeł źródłowy trasy) ciągu pojedynczych systemów kolejkowych, gdzie oczekują na obsługę zgodnie z przyjętą strategią obsługi kolejki. Obsługa wiadomości polega na jej transferze do następnej kolejki na trasie. Procedura transferu jest kontynuowana aż do chwili, gdy dana wiadomość będzie obsłużona przez ostatnie na trasie stanowisko obsługi (węzeł ujściowy trasy) i wyprowadzona z trasy (z sieci) do ujścia.

W pojedynczych systemach kolejkowych obsługiwane są zarówno wiadomości wewnętrzne trasy, jak i pakiety zewnętrzne trasy. Źródłami pakietów zewnętrznych trasy są inne, różne od węzła źródłowego danej trasy, węzły sieci. Na ogół struktura napływu wiadomości zewnętrznych jest złożona i zależy od wielu czynników, w tym od przepływu na danej trasie i na pozostałych trasach sieci.



Rysunek 6.25. Model kolejkowy połączenia logicznego (trasy) bez procedury sterowania przepływem

W uproszczonych modelach kolejkowych trasy w sieci z komutacją wiadomości bez procedury sterowania przepływem zakłada się, że (rysunek 6.25):

➤ pojemność pamięci buforowych pojedynczych systemów kolejkowych na trasie jest nieskończona,

➤ procesy napływu pakietów zewnętrznych w każdym z węzłów trasy są wzajemnie niezależne (przyjęcie takiego założenia umożliwia odseparowanie zdarzeń zachodzących na danej trasie od zdarzeń zachodzących w tym samym czasie na pozostałych trasach w sieci),

➤ procesy napływu wiadomości wewnętrznych do pierwszego węzła trasy oraz wiadomości zewnętrznych do wszystkich węzłów trasy są niezależnymi procesami Poissona; średnia intensywność strumienia wiadomości wewnętrznych jest oznaczona przez λ_0 , a średnie intensywności strumieni wiadomości zewnętrznych przez λ_i , gdzie $i = 1, 2, \dots, N$, a N jest liczbą węzłów danej trasy,

➤ długości wszystkich wiadomości mają ten sam rozkład wykładniczy; przy założeniu, że wszystkie kanały transmisyjne mają stałą przepustowość, przyjęcie wykładniczego rozkładu długości wiadomości oznacza wykładniczy rozkład czasu obsługi wiadomości w każdym pojedynczym systemie kolejkowym trasy z wartością średnią $1/\mu_i$ dla $i = 1, 2, \dots, N$,

➤ czasy obsługi wiadomości na stanowiskach obsługi, w poszczególnych pojedynczych systemach kolejkowych, są wzajemnie niezależne.

Mechanizm powstawania przeciążenia w omawianym połączeniu jest następujący: Dopóki wartości średnich czasów międzyzgłoszeniowych wszystkich strumieni wiadomości są znacząco większe od wartości średnich czasów obsługi, dopóty długości kolejek w poszczególnych pojedynczych systemach kolejkowych są względnie małe, a opóźnienie wiadomości wnoszone przez system obsługi jest względnie małe, niezależnie od przyjętej strategii obsługi kolejki i bez konieczności implementowania procedury sterowania przepływem. Znaczący wzrost intensywności napływu wiadomości wewnętrznych powoduje zwiększenie średnich długości kolejek w poszczególnych pojedynczych systemach kolejkowych i tym samym wzrost wartości średnich czasu oczekiwania wiadomości (wewnętrznych i zewnętrznych) na obsługę w węzłach trasy.

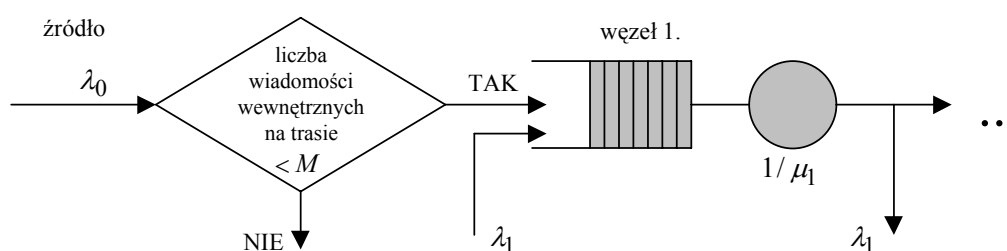
Jednym ze sposobów utrzymywania czasów oczekiwania zewnętrznych wiadomości na obsługę jest skrócenie średnich czasów obsługi wiadomości w węzłach trasy przez zwiększenie zasobów transmisyjnych trasy, tj. przez zwiększenie przepustowości kanałów transmisyjnych. Rozwiązanie takie jest uzasadnione wtedy, gdy przyrost średniej intensywności strumienia wiadomości wiąże się ze zmianą intensywności strumienia na stałe. Gdy przyrosty intensywności strumienia generowanych wiadomości są chwilowe, zwiększanie przepustowości kanałów transmisyjnych nie jest zasadne, wtedy celowe jest implementowanie procedur sterowania przepływem na trasie lub jej wyróżnionych fragmentach, polegających na chwilowym ograniczaniu liczby wiadomości wprowadzanych na trasę lub jej wybrane fragmenty. Ograniczenie liczby wprowadzanych wiadomości na trasę pozwala utrzymywać długości kolejek poniżej zadanych wartości i tym samym założonych wartości średnich czasów obsługi wiadomości.

Obsługa wiadomości zewnętrznych w węzłach sieci należących do danej trasy odbywa się kosztem jakości obsługi wiadomości wewnętrznych; powoduje zwłaszcza pogorszenie jakości obsługi wiadomości wewnętrznych, wyrażające się m.in. zwiększeniem wartości średniego czasu opóźnienia wnoszonego przez trasę.

Wiadomości wewnętrzne każdego połączenia logicznego są jednocześnie wiadomościami zewnętrznymi innych połączeń logicznych obsługiwanych w sieci. Wszystkie połączenia korzystają jednocześnie ze współdzielonych, ograniczonych zasobów sieci. Blokowanie wiadomości wewnętrznych danego połączenia jest ceną płaconą za zabezpieczenie ich jakości obsługi przed degradacją, powodowaną żądaniami obsługi wiadomości wewnętrznych innych połączeń.

Implementacja sterowania przepływem, polegająca na ograniczaniu liczby wiadomości wprowadzanych do obsługi na trasie, wiąże się z wprowadzeniem modelu obsługi ze stratami – niektóre wiadomości generowane przez źródło są odrzucane i nie są ponownie wprowadzane do obsługi. Sterowanie przepływem jest związane z redukcją liczby wiadomości wprowadzanych do obsługi. Miarą redukcji jest wartość prawdopodobieństwa, że wygenerowana wiadomość wewnętrzna będzie odrzucona (zablokowana) p_B .

Wprowadzenie procedur sterowania przepływem na poziomie trasy i na poziomie lokalnym modyfikuje działanie sieci kolejek (np. sieci kolejek przedstawionych na rysunku 6.25). Zgodnie z przyjętą procedurą sterowania przepływem na poziomie trasy wszystkie wiadomości wewnętrzne, generowane w źródle i napływające do węzła źródłowego, są akceptowane wtedy, gdy liczba wiadomości znajdujących się w trakcie obsługi w węzłach trasy jest mniejsza od M ; w przeciwnym razie nowo generowane przez źródło wiadomości wewnętrzne są odrzucane (rysunek 6.26). Decyzja o akceptacji lub odrzuceniu wiadomości wewnętrznej jest podejmowana w chwili jej napływu do węzła źródłowego tylko i wyłącznie na podstawie liczby wiadomości wewnętrznych znajdujących się w tym węźle.



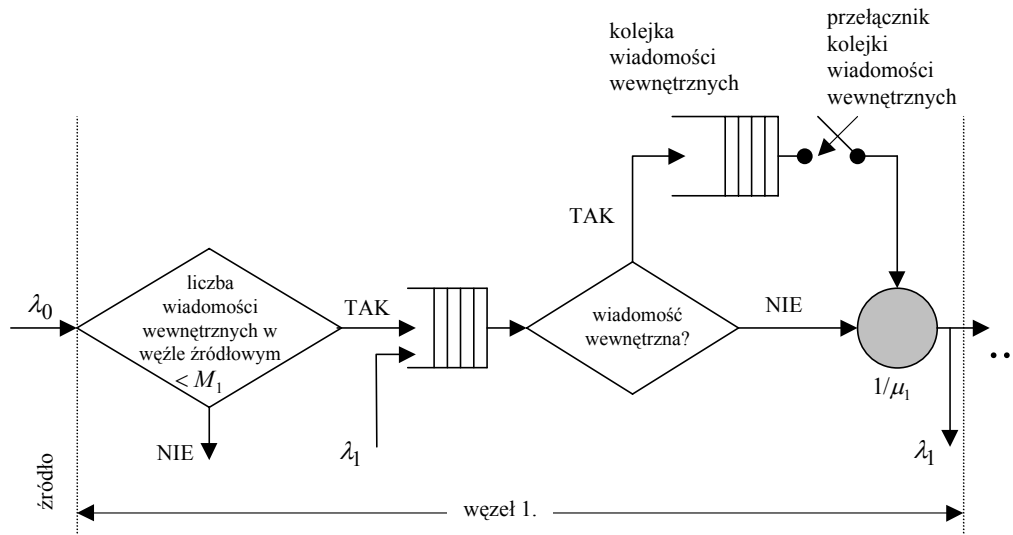
Rysunek 6.26. Procedura sterowania przepływem na poziomie trasy

Procedura lokalnego sterowania przepływem, implementowana w poszczególnych węzłach trasy, jest następująca:

Wiadomość wewnętrzna, znajdująca się na stanowisku obsługi w i -tym węźle ($i = 1, 2, \dots, N - 1$) jest blokowana wtedy, gdy liczba wiadomości wewnętrznych

w węźle $(i + 1)$ -szym jest równa M_{i+1} . Wiadomość jest blokowana dopóty, dopóki jedna z wiadomości wewnętrznych obsługiwanych w $(i + 1)$ -szym węźle nie zostanie obsłużona i przekazana do kolejnego $(i + 2)$ -go węzła. Podczas oczekiwania zablokowanej wiadomości wewnętrznej w i -tym węźle na zwolnienie miejsca w $(i + 1)$ -szym węźle dopuszczalna jest obsługa wiadomości zewnętrznych napływających do i -tego węzła i poprzedzanych przez zablokowaną wiadomość wewnętrzną. W takim przypadku zablokowanej wiadomości wewnętrznej nadawany jest priorytet pierwszeństwa obsługi (ang. *preemptive*) i wiadomość jest umieszczana w kolejce wiadomości wewnętrznych i -tego węzła.

Jeżeli w chwili napływu wiadomości wewnętrznej do węzła źródłowego (pierwszego węzła trasy) liczba wiadomości wewnętrznych, obsługiwanych i oczekujących na obsługę w tym węźle jest równa M_1 , to wiadomość jest odrzucana i tracona. W innym przypadku wiadomości wewnętrzne są kierowane do kolejki wiadomości wewnętrznych węzła pierwszego (rysunek 6.27).

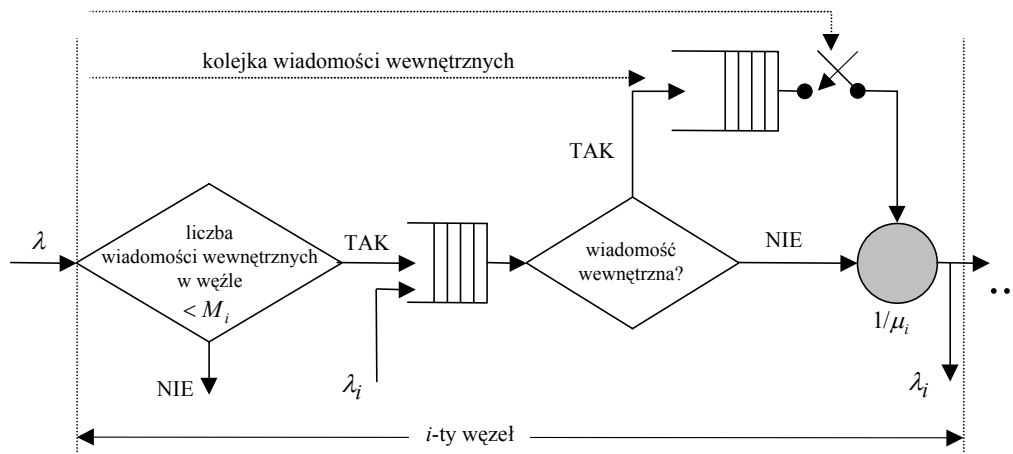


Rysunek 6.27. Model kolejkowy węzła źródłowego trasy z lokalną procedurą sterowania przepływem

Dopóki liczba wiadomości wewnętrznych znajdujących się w $(i + 1)$ -szym węźle jest mniejsza od M_{i+1} , dopóty jest zamknięty przełącznik kolejki wiadomości wewnętrznych w i -tym węźle i są one obsługiwane tak, jakby nie było procedury sterowania przepływem. Jeżeli liczba wiadomości wewnętrznych w $(i + 1)$ -szym węźle jest równa M_{i+1} , to przełącznik kolejki wiadomości wewnętrznych w i -tym węźle jest otwarty i nie są one obsługiwane. Po zniknięciu warunku blokady wiadomości wewnętrznych niezwłocznie rozpoczyna się obsługa wiadomości wewnętrznych znajdujących się w kolejce (rysunek 6.28).

przełącznik kolejki wiadomości wewnętrznych w i -tym węźle:

- otwarty, jeżeli liczba wiadomości wewnętrznych w węźle $(i + 1)$ -szym $= M_{i+1}$
- zamknięty, jeżeli liczba wiadomości wewnętrznych w węźle $(i + 1)$ -szym $< M_{i+1}$



Rysunek 6.28. Procedura obsługi wiadomości wewnętrznych i zewnętrznych w węzłach trasy

Tak sformułowany model kolejkowy nie pozwala na otrzymanie analitycznych wyrażeń, nawet dla przypadku braku wiadomości zewnętrznych w systemie obsługi, ze względu na to, że zawiera pojedyncze systemy kolejkowe z ograniczoną długością kolejek.

Parametrami dwóch implementowanych procedur sterowania są wartości liczby wszystkich wiadomości wewnętrznych we wszystkich węzłach trasy M i wartości liczb wiadomości wewnętrznych M_i ($i = 1, 2, \dots, N$).

Szacowanie i porównywanie różnych procedur sterowania wymaga wprowadzenia miary przeciążenia powodowanego wiadomościami generowanymi przez użytkowników węzłów trasy. Konsekwencją wzrostu liczby obsługiwanych wiadomości wewnętrznych jest wydłużenie czasu (opóźnienia) obsługi wiadomości zewnętrznych; dlatego jedną z możliwych, naturalnych miar przeciążenia jest względne wydłużenie czasu oczekiwania na obsługę wiadomości zewnętrznych, spowodowane zwiększeniem liczby obsługiwanych wiadomości wewnętrznych, odniesionej do liczby wszystkich wiadomości zewnętrznych – współczynnik obciążenia łącza (ang. *link-loading factor*).

Przy założeniu, że d_i jest wartością średnią czasu oczekiwania na obsługę wiadomości zewnętrznych w i -tym węźle (równą wartości średniej czasu oczekiwania wszystkich wiadomości na obsługę w i -tym węźle), a d_{mi} jest wartością średnią czasu oczekiwania wiadomości zewnętrznych na obsługę i -tym węźle, jaki byłby wtedy, gdyby wiadomości wewnętrznych nie było w i -tym węźle, współczynnik obciążenia trasy, złożonej z N węzłów – oznaczony przez α_N – jest określony wyrażeniem:

$$\alpha_N = \sum_{i=1}^N \frac{\lambda_i}{\sum_{j=1}^N \lambda_j} \frac{d_i - d_{mi}}{d_{mi}}$$

Wartość współczynnika obciążenia trasy można wyrazić jako zależną od wartości średnich liczb wiadomości zewnętrznych w węzłach trasy. Na podstawie zależności pomiędzy czasem oczekiwania w kolejce na obsługę i średnią długością kolejki (twierdzenie Little'a) średni czas oczekiwania d_i określa się wyrażeniem:

$$d_i = \frac{E(l_i)}{\lambda_i},$$

w którym $E(l_i)$ jest wartością średnią liczby wszystkich wiadomości w i -tym węźle l_i .

Po podstawieniu tej zależności do poprzedniego wzoru, wyrażenie określające α_N przyjmuje postać:

$$\alpha_N = \sum_{i=1}^N \frac{\lambda_i}{\sum_{j=1}^N \lambda_j} \frac{E(l_i) - E(l_{mi})}{E(l_{mi})},$$

gdzie:

l_i – liczba wiadomości zewnętrznych w i -tym węźle,

l_{mi} – liczba wiadomości zewnętrznych w i -tym węźle wtedy, gdy w węźle tym nie ma wiadomości wewnętrznych ($\lambda_0 = 0$).

Wartość oczekiwana liczby wiadomości zewnętrznych l_{mi} jest równa:

$$E(l_{mi}) = \frac{\lambda_i}{\mu_i - \lambda_i}.$$

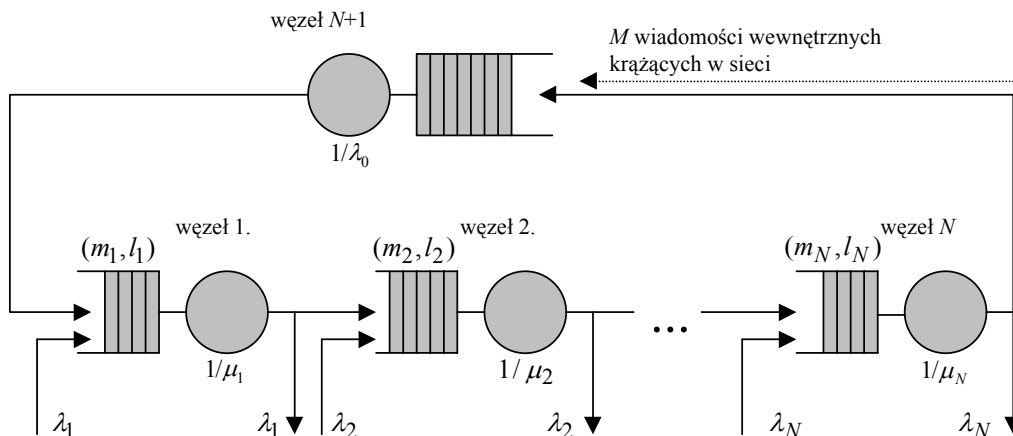
Zgodnie z tym równaniem przyrost liczby wiadomości wewnętrznych na trasie powoduje zwiększenie wartości średniej liczby wiadomości zewnętrznych w węzłach trasy $E(l_i)$. Podstawienie tego wyrażenia do zależności definiującej współczynnik obciążenia trasy L prowadzi do następującej postaci:

$$\alpha_N = \sum_{i=1}^N \frac{(\mu_i - \lambda_i) E(l_i) - \lambda_i}{\sum_{j=1}^N \lambda_j}.$$

Wartość współczynnika obciążenia trasy zależy tylko od wartości parametrów modelu kolejkowego i średniej liczby wiadomości zewnętrznych w każdym z węzłów trasy. Analiza działania procedur sterowania przepływem sprowadza się więc do otrzymania wartości wielkości modyfikowanych procedurami sterowania przepływem wiadomości – wartości prawdopodobieństwa blokowania napływu wiadomości wewnętrznych i średnich długości kolejek wiadomości zewnętrznych w każdym z węzłów trasy.

6.4.3.3. STEROWANIE PRZEPLÝWEM NA TRASIE

Sieć kolejek, będąca modelem sterowanego przepływu na trasie (pomiędzy węzłami źródłowym i ujściowym), przedstawiona na rysunku 6.26, jest równoważna zamkniętej sieci kolejek przedstawionej na rysunku 6.29.



Rysunek 6.29. Procedura sterowania przepływem na poziomie trasy

Ta postać modelu wielowęzłowego systemu obsługi wiadomości wewnętrznych i zewnętrznych pozwala na uzyskanie analitycznych postaci jego podstawowych charakterystyk.

Po oznaczeniu liczby wiadomości wewnętrznych i zewnętrznych w kolejce i -tego węzła przez – odpowiednio – m_i oraz l_i , stan sieci kolejek jest definiowany jako $2N$ -wymiarowy wektor (\bar{m}, \bar{l}) , gdzie $\bar{m} = \{m_1, m_2, \dots, m_N\}$ i $\bar{l} = \{l_1, l_2, \dots, l_N\}$. Równania równowagi dla stanu stacjonarnego sieci kolejek mają na ogół następującą postać układu równań algebraicznych:

$$\begin{aligned} & \left[\lambda_0 + \sum_{i=1}^N \lambda_i + \sum_{i=1}^N \mu_i \right] P(\bar{m}, \bar{l}) = \\ & = \lambda_0 P(m_1 - 1, m_2, m_3, \dots, m_N, \bar{l}) + \sum_{i=1}^N \lambda_i P(\bar{m}, l_1, l_2, \dots, l_i - 1, \dots, l_N) + \\ & + \sum_{i=1}^{N-1} \frac{m_i + 1}{m_i + 1 + l_i} \mu_i P(m_1, m_2, \dots, m_i + 1, m_{m_{i+1}}, \dots, m_N, \bar{l}) + \\ & + \frac{m_N + 1}{m_N + 1 + l_N} \mu_N P(m_1, m_2, \dots, m_N + 1, \bar{l}) + \sum_{i=1}^N \frac{l_i + 1}{m_i + l_i + 1} \mu_i P(\bar{m}, l_1, l_2, \dots, l_i + 1, \dots, l_N), \end{aligned}$$

gdzie $P(\bar{m}, \bar{l})$ jest prawdopodobieństwem przebywania sieci kolejek w stanie (\bar{m}, \bar{l}) .

Rozwiązanie układu równań algebraicznych ma postać:

$$P(\bar{m}, \bar{l}) = P(\bar{0}, \bar{0}) \prod_{i=1}^M \frac{(m_i + l_i)!}{m_i! l_i!} \left(\frac{\lambda_0}{\mu_i} \right)^{m_i} \left(\frac{\lambda_i}{\mu_i} \right)^{l_i},$$

gdzie $P(\bar{0}, \bar{0})$ jest współczynnikiem normalizującym, którego wartość jest wyznaczona z warunku, że suma prawdopodobieństw wszystkich możliwych stanów jest równa jedności.

$$\sum_{(\bar{m}, \bar{l}) \in \bar{\Omega}} P(\bar{m}, \bar{l}) = P(\bar{0}, \bar{0}) \sum_{(\bar{m}, \bar{l})} \prod_{i=1}^N \frac{(m_i + l_i)!}{m_i! l_i!} \left(\frac{\lambda_0}{\mu_i} \right)^{m_i} \left(\frac{\lambda_i}{\mu_i} \right)^{l_i},$$

$$\sum_{(\bar{m}, \bar{l}) \in \bar{\Omega}} P(\bar{m}, \bar{l}) = 1 = P(\bar{0}, \bar{0}) s(\bar{\lambda}, \bar{\mu}),$$

gdzie Ω jest zbiorem wszystkich stanów (\bar{m}, \bar{l}) .

Suma $s(\bar{\lambda}, \bar{\mu})$ może być przedstawiona, po kolejnych przekształceniach, w następującej postaci:

$$s(\bar{\lambda}, \bar{\mu}) = \frac{1}{P(\bar{0}, \bar{0})} = \left[\sum_{(\bar{m}, \bar{l}) \in \bar{\Omega}_{m \leq M}} \prod_{i=1}^N \left(\frac{\lambda_0}{\mu_i - \lambda_i} \right)^{m_i} \right] \left[\prod_{i=1}^N \left(1 - \frac{\lambda_i}{\mu_i} \right) \right]^{-1},$$

co pozwala na wyrażenie prawdopodobieństwa przebywania sieci kolejek w stanie (\bar{m}, \bar{l}) – $P(\bar{m}, \bar{l})$, w postaci równania:

$$P(\bar{m}, \bar{l}) = (Z_N^M)^{-1} \prod_{i=1}^N \frac{(m_i + l_i)!}{m_i! l_i!} \left(\frac{\lambda_0}{\mu_i} \right)^{m_i} \left(\frac{\lambda_i}{\mu_i} \right)^{l_i} \frac{\mu_i - \lambda_i}{\mu_i},$$

gdzie $\Omega_{m \leq M}$ ($\Omega_{m \leq M} \subset \Omega$) jest zbiorem tych wszystkich stanów spełniających warunek ograniczonej liczby wiadomości wewnętrznych w systemie obsługi: $m_1 + m_2 + \dots + m_N \leq M$, tzn. suma wszystkich wiadomości wewnętrznych w węzłach trasy jest nie większa od M , a Z_N^M jest określone wyrażeniem:

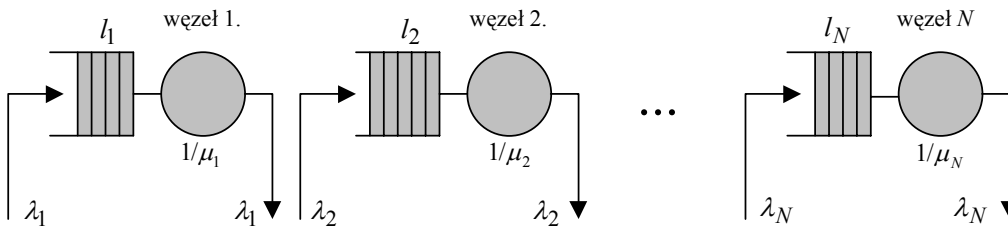
$$Z_N^M = \sum_{(\bar{m}, \bar{l}) \in \bar{\Omega}_{m \leq M}} \prod_{i=1}^N \left(\frac{\lambda_0}{\mu_i - \lambda_i} \right)^{m_i}.$$

Dla dwóch specjalnych, charakterystycznych przypadków obsługi wiadomości w węzłach trasy, tj. wtedy, gdy obsługiwane są tylko wiadomości wewnętrzne i wtedy, gdy są obsługiwane tylko wiadomości zewnętrzne, wzory określające prawdopodobieństwa stacjonarne stanów systemu obsługi mają uproszczoną postać.

Jeżeli w węzłach trasy są obsługiwane tylko zgłoszenia zewnętrzne ($\lambda_0 = 0$), to prawdopodobieństwo $P(\bar{0}, \bar{l})$ stanu $(\bar{0}, \bar{l})$:

$$P(\bar{0}, \bar{l}) = \prod_{i=1}^N \left(\frac{\lambda_i}{\mu_i} \right)^{l_i} \left(\frac{\mu_i - \lambda_i}{\mu_i} \right),$$

co oznacza, że wzór na prawdopodobieństwo stanu sieci kolejek zawierającej N pojedynczych systemów obsługi $P(\bar{m}, \bar{l})$, redukuje się do wzoru określającego prawdopodobieństwo stanu N niezależnych kolejek typu $M/M/1$ (rysunek 6.30).



Rysunek 6.30. Sieć kolejek – równoważny model systemu obsługi wiadomości zewnętrznych

Zgodnie z twierdzeniem Burke'a liczby wiadomości oczekujących w poszczególnych kolejkach oraz czasy obsługi wiadomości w poszczególnych kolejkach w takim systemie obsługi są wzajemnie niezależne i są niezależne od ciągu czasów zakończenia obsługi poprzednich wiadomości w N -tej kolejce systemu.

Jeżeli w węzłach trasy są obsługiwane tylko zgłoszenia wewnętrzne ($\lambda_i = 0$ dla $i = 1, 2, \dots, N$), to prawdopodobieństwo $P(\bar{m}, \bar{0})$ stanu $(\bar{m}, \bar{0})$ jest określone wyrażeniem:

$$P(\bar{m}, \bar{0}) = (Z_N^M)^{-1} \prod_{i=1}^N \left(\frac{\lambda_0}{\mu_i} \right)^{m_i},$$

co oznacza, że wzór określający prawdopodobieństwo stanu sieci kolejek zawierającej N pojedynczych systemów obsługi $P(\bar{m}, \bar{l})$ redukuje się do wyrażenia na prawdopodobieństwo stanu systemu cyklicznego N kolejek typu $M/M/1$.

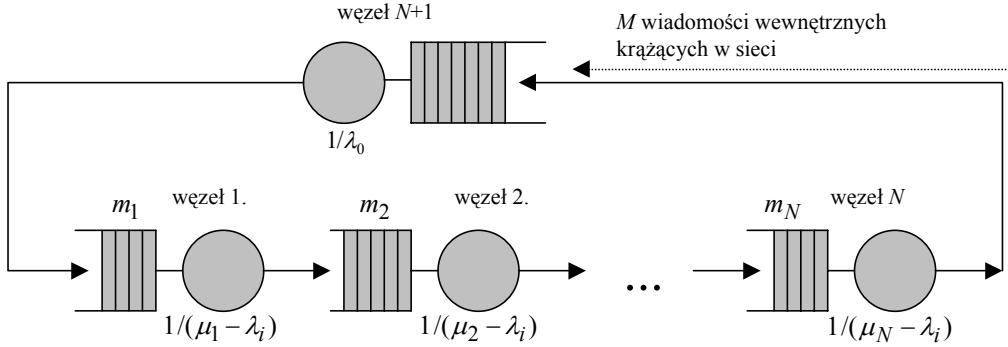
Rozkład prawdopodobieństwa stacjonarnego liczby wszystkich wiadomości wewnętrznych w N węzłach $P(\bar{m})$, otrzymany w wyniku sumowania prawdopodobieństw $P(\bar{m}, \bar{l})$ po wszystkich stanach (\bar{m}, \cdot) ($(\bar{m}, \cdot) \in \bar{U}$):

$$P(\bar{m}) = \sum_{\bar{l} \in \bar{U}} P(\bar{m}, \bar{l}) = (Z_N^M)^{-1} \prod_{i=1}^N \left(\frac{\lambda_0}{\mu_i - \lambda_i} \right)^{m_i},$$

jest rozkładem prawdopodobieństwa równoważnego systemu obsługi, w którym wartości średnie szybkości obsługi wiadomości μ_i zostały zastąpione wartościami $(\mu_i - \lambda_i)$ (rysunek 6.31).

Interpretacja tych zależności jest następująca: obsługa wiadomości zewnętrznych powoduje pomniejszenie wartości średnich szybkości obsługi wiadomości wewnętrznych o wartości średnie szybkości napływu wiadomości zewnętrznych. Obecność wiadomości zewnętrznych w systemie obsługi nie wpływa na wykładniczy rozkład

czasu obsługi wiadomości wewnętrznych; wpływa tylko na wartości średnie szybkości obsługi. Szybkość napływu wiadomości zewnętrznych jest szybkością obsługi tych wiadomości w stanie ustalonym.



Rysunek 6.31. Zamknięta sieć kolejek – równoważny model systemu obsługi wiadomości wewnętrznych

Z analitycznej postaci wyrażenia określającego $P(\bar{m}, \bar{l})$ wynika, że dla omawianego systemu obsługi i przyjętych założeń możliwe jest otrzymanie wielkości charakterystycznych dla sieci kolejek i procedury sterowania przepływem w postaci analitycznej. Wielkościami tymi są wartości średnie liczby wiadomości wewnętrznych i zewnętrznych w węzłach trasy oraz prawdopodobieństwo odrzucania napływających wiadomości wewnętrznych.

Średnia liczba wiadomości wewnętrznych m_i ($i = 1, 2, \dots, N$) w poszczególnych węzłach trasy:

$$E(m_i) = \sum_{(\bar{m}, \bar{l}) \in \bar{U}} m_i P(\bar{m}, \bar{l}) = (Z_N^M)^{-1} x_i \frac{1}{dx_i} Z_N^M,$$

gdzie $x_i = \lambda_0 / (\mu_i - \lambda_i)$ dla $i = 1, 2, \dots, N$.

Średnią liczbę wiadomości zewnętrznych l_i ($i = 1, 2, \dots, N$) w poszczególnych węzłach trasy określa wzór:

$$\begin{aligned} E(l_i) &= \sum_{(\bar{m}, \bar{l}) \in \bar{U}} l_i P(\bar{m}, \bar{l}) = \frac{\lambda_i}{\mu_i - \lambda_i} \left[1 + (Z_N^M)^{-1} x_i \frac{1}{dx_i} Z_N^M \right] = \\ &= \frac{\lambda_i}{\mu_i - \lambda_i} [1 + E(m_i)] = E(l_i | \lambda_0 = 0) [1 + E(m_i)], \end{aligned}$$

w którym: $E(l_i | \lambda_0 = 0)$ jest wartością średnią liczby wiadomości zewnętrznych w przypadku braku wiadomości wewnętrznych w systemie obsługi ($\lambda_0 = 0$). Konsekwencją obecności wiadomości wewnętrznych w systemie obsługi jest wzrost wartości średniej liczby wiadomości zewnętrznych ze współczynnikiem $(1 + E(m_i))$.

Prawdopodobieństwo blokady procesu przyjmowania nowych zgłoszeń wewnętrznych $P_B(M)$, tzn. prawdopodobieństwo zdarzenia, że jeżeli w systemie obsługi znajduje się dokładnie M wiadomości wewnętrznych, to nowe wiadomości wewnętrzne nie są przyjmowane w węzle źródłowym do obsługi na trasie:

$$\begin{aligned} P_B &= \sum_{(\bar{m}, \bar{l}) \in \dot{\Omega}_{m=M}} P(\bar{m}, \bar{l}) = \\ &= \sum_{(\bar{m}, \bar{l}) \in \dot{\Omega}_{m \leq M}} P(\bar{m}, \bar{l}) - \sum_{(\bar{m}, \bar{l}) \in \dot{\Omega}_{m \leq M-1}} P(\bar{m}, \bar{l}) = 1 - \sum_{(\bar{m}, \bar{l}) \in \dot{\Omega}_{m \leq M-1}} P(\bar{m}, \bar{l}) = \\ &= \left[\sum_{(\bar{m}, \bar{l}) \in \dot{\Omega}_{m \leq M-1}} \prod_{i=1}^N \left(\frac{\lambda_0}{\mu_i - \lambda_i} \right)^{m_i} \right] \left[\sum_{(\bar{m}, \bar{l}) \in \dot{\Omega}_{m \leq M}} \prod_{i=1}^N \left(\frac{\lambda_0}{\mu_i - \lambda_i} \right)^{m_i} \right]^{-1} = 1 - \frac{Z_N^{M-1}}{Z_N^M}, \end{aligned}$$

gdzie zbiory $\Omega_{m \leq M-1}$ ($\Omega_{m \leq M-1} \subset \Omega_{m \leq M}$, $\Omega_{m \leq M-1} \cup \Omega_{m=M} = \Omega_{m \leq M}$), $\Omega_{m \leq M}$ ($\Omega_{m \leq M} \subset \Omega$), $\Omega_{m=M}$ ($\Omega_{m=M} \subset \Omega_{m \leq M}$ oraz $\Omega_{m \leq M-1}$ są zbiorami wszystkich stanów (\bar{m}, \bar{l}) spełniających – odpowiednio – warunki $m_1 + m_2 + \dots + m_N = M$, $m_1 + m_2 + \dots + m_N \leq M - 1$ oraz $m_1 + m_2 + \dots + m_N \leq M$, a $x_i = \lambda_0 / (\mu_i - \lambda_i)$.

Analiza zależności wartości prawdopodobieństwa blokady $P_B(M)$ strumienia wiadomości wewnętrznych od wartości parametrów strumienia wiadomości wewnętrznych λ_0 i zewnętrznych λ_i ($i = 1, 2, \dots, N$) oraz wartości parametrów stanowisk obsługi $1/\mu_i$ ($i = 1, 2, \dots, N$) potwierdza intuicyjne wyniki. Dla ustalonych wartości intensywności strumienia wiadomości wewnętrznych (λ_0) i średniego czasu obsługi wiadomości w węzłach ($1/\mu_i$) zmniejszenie wartości $P_B(M)$ jest możliwe po zmniejszeniu intensywności napływu wiadomości zewnętrznych (λ_i). Stałą wartość $P_B(M)$, w przypadku wzrostu intensywności napływu wiadomości zewnętrznych, można utrzymać po zmniejszeniu intensywności napływu wiadomości wewnętrznych lub zmniejszeniu wartości średniego czasu obsługi wiadomości.

6.4.3.4. LOKALNE STEROWANIE PRZEPIŁYWEM

Ze względu na złożoność modelu kolejkowego lokalnych procedur sterowania przepływem, i powodowaną tym niemożność uzyskania analitycznych wyrażeń dla podstawowych charakterystyk systemu obsługi z lokalną procedurą sterowania przepływem, możliwa jest jedynie przybliżona, uproszczona analiza działania systemu obsługi.

W przybliżonej analizie wpływu lokalnych procedur sterowania przepływem na jakość obsługi wiadomości wykorzystywane są ogólne cechy systemu obsługi z procedurą sterowania na trasie i właściwości pojedynczych systemów kolejkowych $M/M/1$. Uproszczenia przyjmowane na potrzeby uproszczonej analizy są następujące:

1. Pominięcie wiadomości zewnętrznych w systemie obsługi pozwala na modelowanie systemu obsługi jako sieci pojedynczych systemów kolejkowych, w których

węzłach wartości średnich szybkości obsługi wiadomości wewnętrznych są pomniejszone o wartości średnie szybkości napływu wiadomości zewnętrznych. Przyjęcie tego uproszczenia pozwala na zastąpienie sieci kolejek, pokazanych na rysunkach 6.30 i 6.31, siecią kolejek o pojemności ograniczonej maksymalną dopuszczalną liczbą wiadomości wewnętrznych M_i ($i = 1, 2, \dots, N$) (parametry lokalnych procedur sterowania przepływem) w węzłach trasy tak, jak pokazano na rysunku 6.32.

2. Przyjęcie założeń o niezależności poszczególnych pojedynczych systemów kolejkowych (modelowanych jako $M/M/1$ o skończonej długości kolejki) oraz równości prawdopodobieństw blokady stanowiska obsługi w danym węźle i przepełnienia węzła następującego po danym węźle umożliwia oszacowanie wartości średnich długości kolejek i prawdopodobieństwa blokady systemu obsługi. Zgodnie z tym założeniem ostatni N -ty węzeł na trasie nie jest nigdy zablokowany. Prawdopodobieństwo, że N -ty węzeł jest przepełniony, jest równe prawdopodobieństwu blokady $(N-1)$ -szego węzła. Kontynuacja procedury szacowania prawdopodobieństwa i przepełnienia, polegająca na wstecznej propagacji przepełnienia węzłów aż do węzła źródłowego trasy, pozwala na oszacowanie charakterystyk całego systemu obsługi. Jedynym wymaganie techniki szacowania prawdopodobieństwa blokady jest znajomość warunków początkowych, tzn. znajomość przepustowości trasy λ , której wartość jest równa iloczynowi wartości średniej szybkości napływu wiadomości wewnętrznych λ_0 i wartości prawdopodobieństwa zdarzenia, że kolejka w węźle źródłowym trasy nie jest zajęta w całości p_1 , tzn. $\lambda = p_1 \lambda_0$.

3. Przyjęcie założenia, że średnia liczba wiadomości zewnętrznych w i -tym węźle ($i = 1, 2, \dots, N$) trasy jest określona wyrażeniem

$$E(l_i) = \frac{\lambda_i}{\mu_i - \lambda_i} [1 + E(m_i) p_{i+1}],$$

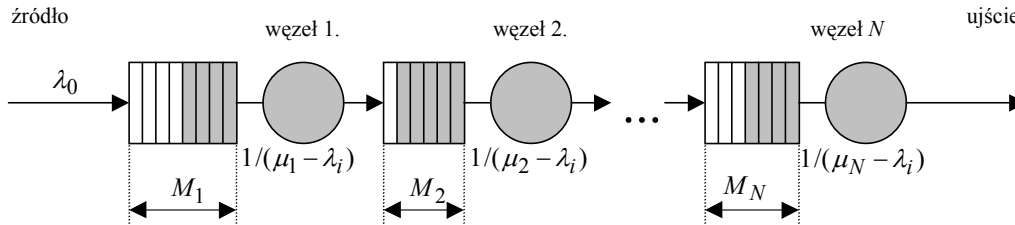
w którym p_{i+1} jest wartością prawdopodobieństwa zdarzenia, że kolejka w $(i+1)$ -szym węźle nie jest zajęta w całości ($p_{N+1} = 1$), powoduje, że:

➤ jeżeli kolejka w $(i+1)$ -szym węźle ($i = 1, 2, \dots, N-1$) trasy nie jest nigdy w całości zajęta ($p_{i+1} = 1$), to średnia liczba wiadomości zewnętrznych w i -tym węźle $E(l_i)$ jest określona wzorem:

$$E(l_i) = \frac{\lambda_i}{\mu_i - \lambda_i} [1 + E(m_i)],$$

➤ jeżeli kolejka w $(i+1)$ -szym węźle ($i = 1, 2, \dots, N-1$) trasy jest zawsze w całości zajęta ($p_{i+1} = 0$), to na stanowisku obsługi w i -tym węźle trasy są obsługiwane tylko wiadomości zewnętrzne – średnia liczba wiadomości zewnętrznych w i -tym węźle trasy $E(l_{mi})$ jest określona równaniem:

$$E(l_i) = E(l_{mi}) = \frac{\lambda_i}{\mu_i - \lambda_i}.$$



Rysunek 6.32. Sieć kolejek o pojemności ograniczonej maksymalną, dopuszczalną liczbą wiadomości wewnętrznych w węzłach trasy

Przyjmując założenie, że kolejki w węzłach trasy nie są nigdy w całości zajęte, tzn. $p_i = 1$ dla $i = 1, 2, \dots, N$, oszacowaniem wartości współczynnika obciążenia trasy α_N jest równanie:

$$\alpha_N = \sum_{i=1}^N \frac{\lambda_i}{\sum_{j=1}^N \lambda_j} \frac{\frac{\lambda_i}{\mu_i - \lambda_i} (1 + E(m_i) p_{i+1}) - \frac{\lambda_i}{\mu_i - \lambda_i}}{\frac{\lambda_i}{\mu_i - \lambda_i}} = \sum_{i=1}^N \frac{\lambda_i}{\sum_{j=1}^N \lambda_j} E(m_i) p_{i+1}.$$

Wartość współczynnika obciążenia trasy, jako miary przeciążenia trasy, zależy od:

- wartości średniej liczby wiadomości wewnętrznych w każdym węźle trasy (tym samym od liczby wszystkich wiadomości wewnętrznych na trasie),
- wartości współczynników wagowych równych udziałowi liczby wiadomości zewnętrznych, obsługiwanych w danym węźle, w całkowitej liczbie wiadomości zewnętrznych obsługiwanych we wszystkich węzłach trasy,
- wartości prawdopodobieństwa braku blokady wiadomości wewnętrznych w węzłach trasy.

6.4.3.5. CEL STEROWANIA PRZEPIŁYWEM NA TRASIE

Celem sterowania przepływem na poziomie trasy jest zapobieganie przeciążeniom i zakleszczeniom w pamięciach buforowych węzłów ujściowych tras, których przyczyną może być różnica pomiędzy szybkością wprowadzania ruchu do sieci w węźle źródłowym trasy i transferu poprzez sieć a szybkością wyprowadzania ruchu z sieci w węźle ujściowym trasy.

Różnica między szybkością wprowadzania ruchu do sieci i transferu ruchu w sieci a szybkością wyprowadzania z sieci ruchu obsługiwanego przez daną trasę może być spowodowana:

- różną szybkością transferu ruchu na trasie w sieci i transferu w połączeniach węzła ujściowego z terminalami,
- różną szybkością obsługi ruchu w węzłach źródłowym i pośrednich trasy oraz szybkością obsługi w węźle ujściowym trasy.

Przyczyną różnicy szybkości wprowadzania i wyprowadzania ruchu może być niedopasowanie przepustowości lokalnych połączeń, wykorzystywanych do wyprowadzania ruchu z węzła ujściowego do urządzeń końcowych, do przepustowości trasy w sieci. Istnienie tej różnicy oznacza możliwość powstania przeciążenia łączy lokalnych węzła ujściowego, którego następstwem może być częściowa degradacja przepustowości sieci. W takim przypadku przeciwdziałanie przeciążeniom łączy lokalnych węzła ujściowego trasy jest możliwe przez:

- zwiększenie przepustowości łączy lokalnych z węzła ujściowego, w celu dopasowania ich do przepustowości trasy w sieci,
- zmniejszenie szybkości wprowadzania pakietów w węzle źródłowym trasy w celu jej dopasowania do szybkości wyprowadzania pakietów w węzle ujściowym,
- skompensowanie różnicy szybkości napływu i wypływu pakietów w węzle ujściowym trasy odpowiednio dużą pojemnością pamięci buforowej węzła ujściowego.

Inną przyczyną różnej szybkości wprowadzania i wyprowadzania ruchu może być spowolnienie szybkości obsługi ruchu w węzle ujściowym trasy, w porównaniu z pozostałymi węzłami trasy. Spowolnienie takie jest powodowane koniecznością dodatkowego przetwarzania ruchu przed jego wyprowadzeniem z sieci, związanego m.in. z zadaniami: jednoczesnej akceptacji pakietów na poziomie łączy danych i trasy, odtwarzania wiadomości (ang. *reassemble*), odtwarzania kolejności pakietów (ang. *resequence*). Konieczność dodatkowego przetwarzania może powodować przeciążenie węzła i w konsekwencji częściową lub całkowitą degradację przepustowości (zakleszczenie) sieci. Brak skutecznego sterowania przepływem może doprowadzić do zakleszczeń kompletacji (ang. *reassemble deadlock*) i zakleszczeń sekwencji (ang. *resequence deadlock*).

Zapobieganie przeciążeniom węzła ujściowego trasy, powodowanych względnym spowolnieniem obsługi ruchu w węzle ujściowym, w porównaniu z innymi węzłami trasy, jest możliwe przez:

- skompensowanie dłuższego czasu obsługi ruchu w węzle ujściowym odpowiednio większą pojemnością pamięci buforowej węzła ujściowego, której wartość zależy od różnicy czasów przetwarzania w węzłach i czasu trwania sesji obsługiwanej przez daną trasę,
- zmniejszenie (dopasowanie) szybkości wprowadzania pakietów w węzle źródłowym trasy i szybkości transferu na trasie, kompensującego różnice czasu przetwarzania w węzle ujściowym i pozostałych węzłach trasy.

Jeżeli w danej sieci przepustowość połączeń lokalnych z węzła ujściowego trasy do urządzeń końcowych oraz pojemność jego pamięci buforowej są dane, jedynym sposobem przeciwdziałania przeciążeniom jest zmniejszenie szybkości wprowadzanego do sieci ruchu. Zadaniem procedur sterowania przepływem na poziomie trasy jest w tym przypadku regulacja szybkości wprowadzania pakietów do sieci w węzle źródłowym, w zależności od stanu węzła ujściowego, oraz odpowiedni podział pojemności pamięci buforowych węzłów.

Najczęściej spotykanymi sposobami regulacji szybkości wprowadzania pakietów do trasy sieci jest ograniczenie liczby pakietów mogących znajdować się na trasie bez potwierdzenia (metoda okna) lub wprowadzanie pakietów tylko wtedy, gdy na trasie są zarezerwowane zasoby niezbędne do obsługi wprowadzanego ruchu (metoda krokowa).

Dodatkową korzyścią wynikającą ze stosowania różnych metod sterowania przepływem na poziomie tras w sieci, zapobiegających bezpośrednio przeciążeniom tras i ich węzłów ujściowych, jest możliwość pośredniego, w przypadku ich skoordynowania, zapobiegania powstawaniu przeciążeń całej sieci, tj. przeciwdziałanie powstawaniu przeciążeń globalnych. Z kolei metody przeciwdziałania przeciążeniom globalnym mogą redukować możliwość powstawania przeciążeń na trasach.

Bezpośrednim skutkiem powstawania przeciążeń na trasach sieci i jej przeciążenia globalnego jest – odpowiednio – zwiększenie opóźnień pakietów i potwierdzeń tych pakietów na trasach sieci i w sieci. Dlatego podstawowym sposobem koordynowania metod zapobiegania przeciążeniom tras sieci i jej przeciążeniu globalnemu jest wykorzystanie wzrostu opóźnień na trasach sieci i w sieci do ograniczania liczby pakietów mogących znajdować się na poszczególnych trasach sieci bez potwierdzeń i ograniczania lub redystrybucji (pomiędzy różne trasy sieci) całkowitej liczby pakietów mogących znajdować się w sieci bez potwierdzeń.

Jeżeli maksymalne (limity) liczby pakietów mogących znajdować się bez potwierdzeń na (i, j) -tej trasie oraz w całej sieci, przed powstaniem i po powstaniu stanu przeciążenia, są oznaczone przez m_{ij} i m'_{ij} oraz m i m' ($i, j = 1, 2, \dots, N, i \neq j, N$ – liczba węzłów sieci) i spełniają warunki:

$$m = \sum_{i=1}^N \sum_{j=1, j \neq i}^N m_{ij} \quad \text{oraz} \quad m' = \sum_{i=1}^N \sum_{j=1, j \neq i}^N m'_{ij},$$

to koordynacja metod zapobiegania przeciążeniom tras sieci i przeciążeniu globalnemu polega na zmianie lub redystrybucji limitów pakietów dla poszczególnych tras i całej sieci. Obowiązują w tym przypadku następujące zasady:

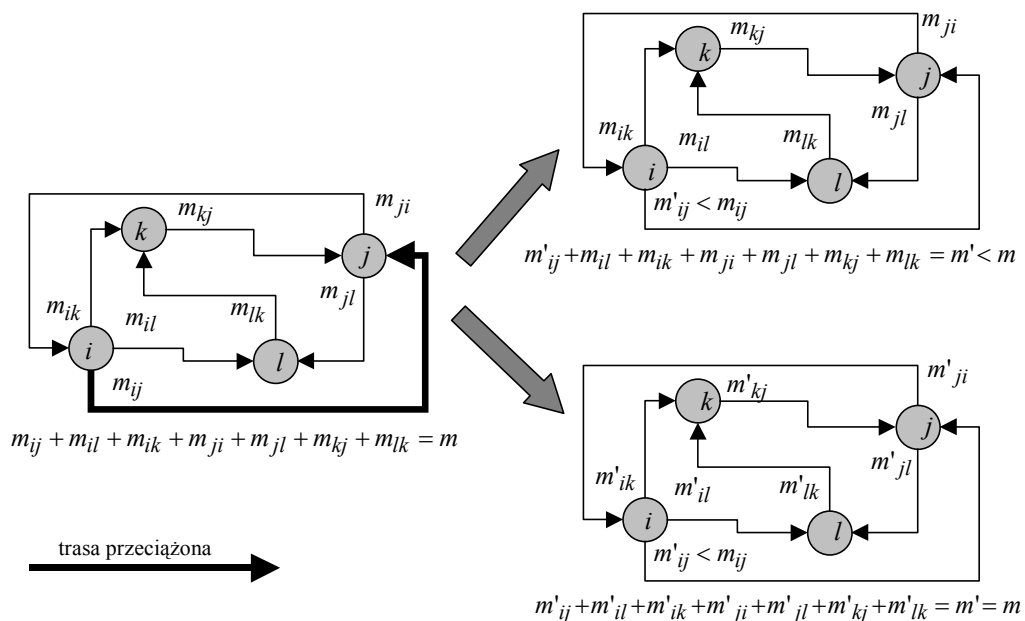
1. Stan przeciążenia (i, j) -tej trasy (przeciążenia lokalnego) może służyć do przeciwdziałania przeciążeniu globalnemu lub do redystrybucji zasobów sieci dla innych tras, w celu zwiększenia wykorzystania zasobów sieci. Wzrost opóźnień na (i, j) -tej trasie, powodujący zmniejszenie liczby pakietów wprowadzanych na daną trasę z m_{ij} na m'_{ij} ($m'_{ij} < m_{ij}$) i zmniejszenie opóźnienia na danej trasie, może prowadzić do (rysunek 6.33):

➤ zmniejszenia limitu liczby pakietów w sieci z m na m' ($m' < m, m' = m - (m_{ij} - m'_{ij})$), co zapobiega przeciążeniu globalnemu,

➤ zwiększenia limitu liczby pakietów na innych od (i, j) -tej trasach sieci ($m'_{kl} \geq m_{kl}$ dla wszystkich $kl \neq ij$, gdzie: $k, l \in 1, 2, \dots, N$) w ramach limitu liczby pakietów wprowadzanych do sieci ($m' = m$), a w konsekwencji do zwiększenia stopnia wykorzystania zasobów w sieci.

2. Stan przeciążenia globalnego wiąże się z koniecznością zmniejszenia limitu liczby pakietów mogących znajdować się bez potwierdzeń w całej sieci, tzn. ograniczenia liczby pakietów współzawodniczących o dostęp do zasobów wybranych lub

wszystkich tras sieci. Zwiększenie opóźnienia w sieci, powodujące zmniejszenie limitu liczby pakietów wprowadzanych na trasy w całej sieci z m na m' ($m' < m$), może być wykorzystane do zmniejszenia liczby pakietów wprowadzanych na wybrane trasy sieci ($m'_{kl} < m_{kl}$ dla $kl \in \Omega_L$ i $m'_{kl} = m_{kl}$ dla $kl \in \Omega_E$ oraz $\Omega_L \cap \Omega_E = \Omega$ i $\Omega_L \cup \tilde{U}_E = \emptyset$, gdzie Ω_L , Ω_E i Ω są, odpowiednio, zbiorami tras ze zmniejszonymi limitami liczby pakietów, zbiorami tras z niezmiennymi limitami liczby pakietów i zbiorem wszystkich tras sieci) lub wszystkie trasy sieci ($\tilde{U}_E = \emptyset$), prowadzącego do zapobiegania przeciążeniom lokalnym w wybranych fragmentach sieci.



Rysunek 6.33. Zmiany limitów pakietów w sieci dla trasy i sieci

6.4.3.6. PRZYKŁADOWE IMPLEMENTACJE METOD STEROWANIA PRZEPLYWEM NA POZIOMIE TRASY

Zasadnicze różnice pomiędzy różnymi procedurami sterowania przepływem na trasach są konsekwencją właściwości jednostek danych przekazywanych na trasach. Ze względu na różną długość obsługiwanych jednostek danych wyróżniane są dwie klasy mechanizmów sterowania przepływem:

- sterowanie przepływem pojedynczych pakietów (wiadomości jednopakietowych),
- sterowanie przepływem wielopakietowych wiadomości (o stałej lub zmiennej długości).

6.4.3.6.1. STEROWANIE PRZEPLYWEM NA POZIOMIE TRASY W SIECI ARPANET

Do sterowania przepływem w sieci ARPANET stosuje się mechanizm okna. Wszystkie wiadomości o różnej długości, adresowane z danego węzła źródłowego do danego węzła ujściowego, są przekazywane za pomocą tego samego połączenia logicznego (ang. *pipe*).

Przepływ w każdym połączeniu logicznym jest przedmiotem indywidualnego sterowania. Rozmiar okna obejmuje osiem wiadomości, z których każda zawiera od jednego do ośmiu pakietów. Transmitowana wiadomość ma numer wskazujący na jej pozycję w oknie.

Cele procedur sterowania przepływem na poziomie trasy w sieci ARPANET, charakterystyczne dla zadań zapobiegania przeciążeniom i zadań sterowania przepływem, są następujące:

- zapobieganie przeciążeniom trasy,
- detekcja zagubionych wiadomości,
- detekcja wiadomości napływających poza kolejnością,
- detekcja kopii napływających wiadomości,
- odtwarzanie kolejności wiadomości,
- odtwarzanie wiadomości wielopakietowych,
- zapobieganie zakleszczeniom sekwencji,
- zapobieganie zakleszczeniom kompletacji.

Cele te są osiąmane dzięki mechanizmom dwóch klasycznych metod sterowania przepływem w sieciach: metody okna oraz metody rezerwacji zasobów.

Metoda okna pozwala na wprowadzenie prostego, unikatowego i jednoznacznego sposobu numeracji wiadomości przekazywanych w danym połączeniu logicznym oraz prostej procedury sprawdzania poprawności procesu transferu wiadomości. Numerowanie wiadomości odbywa się w zakresie wyznaczonym rozmiarem okna.

Zastosowanie metody okna pozwala zatem na realizację następujących zadań sterowania przepływem:

- utrzymywanie liczby pakietów, niepotwierdzonych i znajdujących się na trasie pomiędzy węzłami źródłowym i ujściowym, zgodnie z limitem ustalonym dla danej trasy i równym rozmiarowi okna danej trasy (zwykle rozmiary okien wszystkich tras w sieci są równe),
- odtwarzanie kolejności odbioru pakietów zgodnej z kolejnością nadawania pakietów w węźle źródłowym,
- detekcja pakietów odbieranych z błędami w węźle ujściowym i wymuszanie ich retransmisji przez węzeł źródłowy,
- detekcja duplikatów pakietów w węźle odbiorczym i ich odrzucanie.

Sprawdzanie numerów wiadomości napływających do węzła ujściowego ma na celu:

- sprawdzanie zgodności numeru odbieranej wiadomości z zakresem numerów aktualnie obsługiwanych wiadomości; wiadomości o numerach spoza obowiązującego zakresu są odrzucane,

➤ rozpoznanie kolejności i detekcji zmian w kolejności napływu wiadomości (w porównaniu z kolejnością nadawania determinującego sposób numeracji wiadomości); wiadomości napływające poza kolejnością są odrzucane,

➤ wykrywanie wielokrotnych kopii tych samych wiadomości – wiadomości napływające, będące kopiami wiadomości wcześniej odebranych są odrzucane.

W omawianej procedurze sterowania przepływem zastosowano najprostszy sposób zapobiegania zakleszczeniom sekwencji. Poprawne wiadomości napływające poza kolejnością są odrzucane, ponieważ ich przechowywanie w oczekiwaniu na wiadomość niezbędną do odtworzenia kolejności mogłoby być przyczyną zakleszczenia bezpośredniego.

Poprawny odbiór każdej wiadomości powoduje wygenerowanie przez węzeł ujściowy potwierdzenia, które jest następnie przekazywane do węzła źródłowego. Potwierdzenie ma postać krótkiej wiadomości sterującej RFNM (ang. *Ready For Next Message*). Jednym z jej celów jest wskazanie numeru wiadomości, której poprawny odbiór jest potwierdzany. Odbiór każdej wiadomości RFNM w węzle źródłowym powoduje przesunięcie okna, a tym samym możliwość kontynuowania przez węzeł źródłowy wprowadzania do sieci kolejnych wiadomości.

Wiadomości RFNM są ponadto stosowane do sterowania obsługą błędów. Brak potwierdzenia odbioru wiadomości dla ustalonej wartości zwłoki czasowej (ang. *time-out*) powoduje wysłanie, przez węzeł źródłowy do węzła odbiorczego, wiadomości sterującej w celu zbadania przyczyn braku kontynuacji transferu. Wiadomości RFNM są także przeznaczone do przekazywania żądań retransmisji wiadomości w razie ich zagubienia lub przekłamania.

Mechanizmy właściwe dla metod rezerwacji zasobów są używane do odtwarzania wielopakietowych wiadomości w sposób zapobiegający powstawaniu zakleszczeń kompletacji. Istotą działania mechanizmu alokacji pamięci buforowej na potrzeby kompletacji wiadomości wielopakietowych RBA (ang. *Reassembly Buffer Allocation*) jest rozpoczęcie przekazywania wiadomości wielopakietowej pod warunkiem wcześniejszego zarezerwowania w pamięci buforowej węzła odbiorczego odpowiedniej liczby buforów. Rezerwacja odpowiedniej liczby buforów pozwala na przyjmowanie i przechowywanie pakietów do chwili skompletowania całej wiadomości, zapobiegając jednocześnie powstawaniu zakleszczeń kompletacji.

Rezerwacja liczby buforów, określonej długością wiadomości, odbywa się zgodnie ze schematem obowiązującym wiele protokołów implementowanych na różnych poziomach sterowania przepływem. Cechą tego schematu jest ściśle określony czas oczekiwania węzła źródłowego na odpowiedź z węzła ujściowego na żądanie alokacji oraz ograniczona i zdefiniowana liczba możliwych powtórzeń tego samego żądania alokacji.

Przekazywanie wiadomości wielopakietowej poprzedza wysłanie do węzła ujściowego trasy żądania alokacji buforów w postaci wiadomości REQALL (ang. *Request for Allocation*) i oczekiwanie na jej potwierdzenie, tj. na wiadomość ALL (ang. *Allocation*). W celu redukcji czasu oczekiwania na potwierdzenie alokacji bufo-

rów (tym samym w celu zwiększenia efektywnej przepustowości trasy) potwierdzenie alokacji buforów (lub powiadomienie o braku możliwości alokacji buforów) może być przekazywane w wiadomościach sterujących RFNM. Brak powiadomienia pozytywnego lub negatywnego żądania alokacji buforów w ściśle określonym przedziale czasu (ang. *time-out*) powoduje, że żądanie alokacji jest ponawiane przez węzeł źródłowy. Liczba ponawianych żądań alokacji pamięci buforowej dla jednej wiadomości jest ograniczona i ściśle określona. Gwarantuje to możliwość wykrycia niesprawności trasy i podjęcia (na poziomie sterowania przepływem nadrzędnym wobec poziomu trasy) odpowiednich działań przywracających funkcjonalność trasy.

Wiadomości jednopaketowe są transferowane bez wcześniejszego żądania alokacji buforów w węzle ujściowym. Jeżeli w chwili napływu wiadomości do węzła ujściowego jego pamięć buforowa jest w całości zajęta, to wiadomość jest odrzucana. Ponowne przekazanie wiadomości odbywa się albo automatycznie, albo po uprzednim uzyskaniu alokacji bufora w pamięci buforowej węzła ujściowego.

Wprowadzane modyfikacje mechanizmów sterowania przepływem na poziomie trasy w sieci ARPANET dotyczyły m.in. zmian wartości niektórych parametrów i niektórych procedur:

- zwiększenia rozmiaru okna (do 127) – w celu obsługi transferu wiadomości w łączach satelitarnych – co umożliwia efektywny transfer wiadomości w łączach satelitarnych, wnoszących względnie duże opóźnienie,

- zwiększenia liczby niezależnych, jednocześnie utrzymywanych połączeń logicznych pomiędzy tymi samymi dwoma węzłami (do 256) z niezależnymi oknami – rozwiązanie takie umożliwia elastyczność w obsłudze klas ruchu o różnych priorytetach i wymaganiach przepustowości,

- poprawy sprawiedliwości rozdziału zasobów (pamięci buforowej) w każdym węzle dla każdego z węzłów źródłowych,

- uproszczenia procedury rezerwacji zasobów dla wiadomości wielopakietowych; rezerwacja pamięci buforowej węzła jest dokonywana przez pierwszy pakiet wiadomości wielopakietowej (w razie niemożności rezerwacji zasobów pakiet jest odrzucony, a retransmisja odbywa się po upływie ustalonego przedziału czasu).

6.4.3.6.2. STEROWANIE PRZEPLYWEM NA POZIOMIE TRASY W SIECI GMDNET

Ze względu na to, że sieć GMDNET ma połączenia wirtualne, w której każdej sesji jest przyporządkowana trasa ustalana w czasie nawiązywania połączenia wirtualnego, przepływ w połączeniu wirtualnym jest tożsamy z przepływem na trasie. Sterowanie przepływem na trasie określa się mianem indywidualnego sterowania przepływem (ang. *individual flow control*), bowiem procedury sterowania są implementowane dla każdego połączenia wirtualnego z osobna.

Głównym celem sterowania przepływem pomiędzy węzłami źródłowym i ujściowym jest zapobieganie przepełnieniu pamięci buforowej węzła ujściowego w wyniku różnicy szybkości wprowadzania i wyprowadzania pakietów z węzła ujściowego. Zadaniem procedury sterowania przepływem jest (wtedy, gdy szybkość napływu jest

większa od szybkości wypływu pakietów) zmniejszenie szybkości wprowadzania pakietów przez węzeł źródłowy do połączenia wirtualnego.

Tak sformułowany cel realizuje się za pomocą mechanizmów metody okna ze zmienną jego długością. Zmiana długości okna dla danego połączenia wirtualnego oznacza zmianę liczby pakietów bez potwierdzeń mogących znajdować się w danym połączeniu wirtualnym.

Po oznaczeniu:

- z_i oraz u_i – odpowiednio – węzeł źródłowy i ujściowy i -tego połączenia wirtualnego ($i = 1, 2, \dots$),
- b_i – długość kolejki pakietów napływających w i -tym połączeniu wirtualnym i oczekujących na wyprowadzenie w węźle u_i ,
- h_i – długość połączenia wirtualnego wyrażonego liczbą łączy danych na trasie i -tego połączenia wirtualnego pomiędzy węzłem źródłowym z_i i węzłem ujściowym u_i ,
- PUL_i – rozmiar okna dla i -tego połączenia wirtualnego spełniający warunki: $PUL_{i \min} \leq PUL_i \leq PUL_{i \max}$ oraz $PUL_{i \max} \leq w_E$, gdzie: w_E jest maksymalnym rozmiarem okna dla połączeń wirtualnych w sieci,

ogólna zasada spowalniania szybkości wprowadzania pakietów do i -tego połączenia wirtualnego jest następująca:

➤ podstawowa (maksymalna) długość okna w_i (ustalona w wyniku badań symulacyjnych) zależy od długości połączenia wirtualnego h_i : $PUL_{i \max} = h_i + 1$,

➤ jeżeli w chwili napływu nowego pakietu do węzła u_i długość kolejki pakietów b_i jest równa PUL_i ($b_i = PUL_i$), to dany pakiet jest odrzucany i w węźle ujściowym jest generowane negatywne powiadomienie (NACK_{*i*}); w przeciwnym razie, tj. jeżeli $b_i < PUL_i$, napływający pakiet jest przyjmowany do kolejki i generowane jest potwierdzenie pozytywne (ACK_{*i*}),

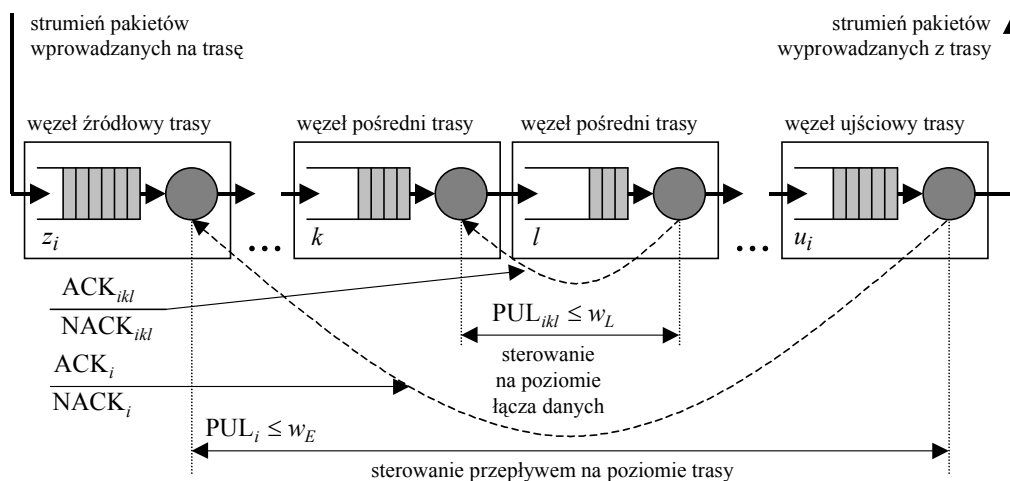
➤ odebranie negatywnego powiadomienia (NACK_{*i*}) w węźle z_i powoduje zmniejszenie o jeden rozmiaru okna: $PUL_i := PUL_i - 1$; każde negatywne powiadomienie oznacza zmniejszanie rozmiaru okna o jeden aż do osiągnięcia minimalnego rozmiaru okna $PUL_{i \min}$ ($PUL_{i \min} = 1$),

➤ odebranie pozytywnego potwierdzenia (ACK_{*i*}) w węźle z_i powoduje zwiększenie o jeden rozmiaru okna: $PUL_i := PUL_i + 1$; odebranie każdego pozytywnego potwierdzenia oznacza zwiększanie rozmiaru okna o jeden aż do osiągnięcia maksymalnego rozmiaru okna $PUL_{i \max}$.

Sterowanie przepływem na poziomie trasy, w omawianym przypadku, jest nadrzędne wobec niezależnego sterowania przepływem na poziomie łączy danych. Rozmiar okna na poziomie trasy zależy od zajętości pamięci buforowej węzła ujściowego, rozmiary okien na poziomie łączy danych zależą natomiast od stopnia przeciążenia węzłów pośrednich trasy (rysunek 6.34).

Nadrzędność sterowania przepływem na poziomie trasy wobec sterowania przepływem w łączach danych oznacza, że funkcje wymienionych poziomów sterowania

uzupełniają się nawzajem. Nie ma więc potrzeby realizacji (oprócz uzasadnionych przypadków wprowadzania redundancji) na poziomie trasy zadań związanych z kompletacją wiadomości, odtwarzaniem kolejności pakietów oraz wykrywania strat i kopii pakietów. Wymienione zadania są rozwiązywane na poziomie łącza danych. Ponieważ trasa jest ciągiem szeregowo połączonych łączy danych, właściwości przepływu w łączy danych są więc właściwościami przepływu na poziomie trasy.



Rysunek 6.34. Nadrzędne i podrzędne sterowanie przepływem na poziomie łącza danych

W procedurach sterowania przepływem na poziomie łącza danych, w pamięciach buforowych węzłów odbiorczych, znajdujących się na trasie połączenia wirtualnego, dla każdej wiadomości wielopakietowej rezerwowana jest liczba buforów równa maksymalnej długości obsługiwanych wiadomości. Oznacza to, że w procedurze sterowania przepływem na poziomie trasy nie ma potrzeby implementacji mechanizmów zapobiegających powstawaniu zakleszczeń kompletacji.

Tożsamość połączenia wirtualnego z ustaloną (dla połączenia wirtualnego) trasą przekazywania pakietów wraz z procedurami sterowania przepływem na poziomie łącza danych (gwarantujących zachowanie kolejności nadawania pakietów) oznacza, że na poziomie trasy nie ma potrzeby implementacji mechanizmu odtwarzania kolejności pakietów, tzn. nie ma potrzeby zapobiegania powstawaniu zakleszczeń sekwencji.

Sterowanie przepływem na poziomie łącza danych, zapobiegające utracie pakietów i powstawaniu kopii, oznacza, że strata pakietów w połączeniu wirtualnym tożsamym z trasą jest możliwa tylko wtedy, gdy wystąpi awaria co najmniej jednego węzła trasy. Ponieważ awaria węzła oznacza automatyczne odtworzenie połączenia wirtualnego (także trasy), to dopóki istnieje dane połączenie wirtualne, dopóty nie ma potrzeby stosowania mechanizmów wykrywania strat i kopii pakietów na poziomie trasy.

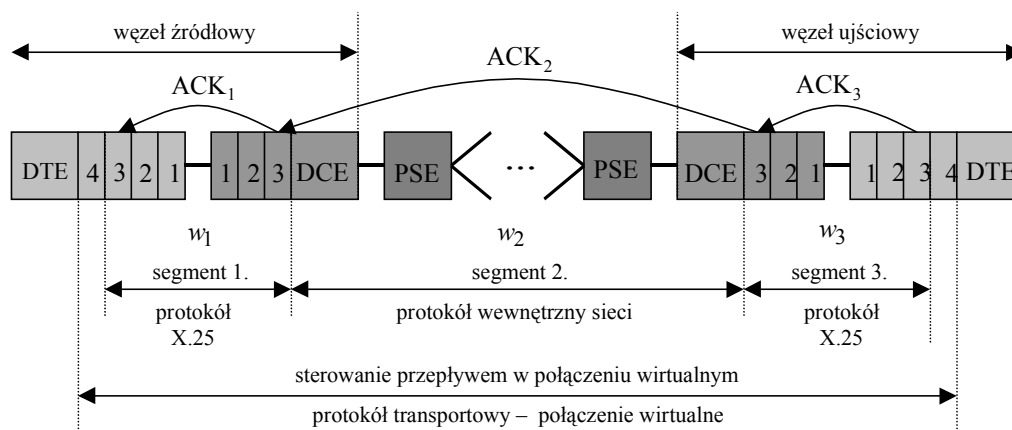
6.4.3.6.3. STEROWANIE PRZEPLYWEM NA POZIOMIE TRASY W SIECI DATAPAC

W omawianym przypadku sieci pakietowej usługi połączenia wirtualnego są dostępne w postaci protokołu warstwy transportowej, nadrzędnej w stosunku do dostępnych w warstwie trzeciej usług datagramowych. Sterowanie przepływem pomiędzy węzłem źródłowym i ujściowym dotyczy przepływu w połączeniu wirtualnym, które nie jest tożsame z jedną fizyczną trasą przekazywania pakietów. Dostarczanie usług połączenia wirtualnego z wykorzystaniem usług różnych fizycznych tras przekazywania datagramów znacznie komplikuje zadania odtwarzania pakietów i ich kolejności.

Połączenie wirtualne jest organizowane pomiędzy punktami końcowymi sesji poprzez kaskadowe złożenie trzech segmentów:

1. segment – pomiędzy źródłowym DTE i wejściowym DCE w węźle źródłowym,
2. segment – pomiędzy DCE węzła źródłowego i DCE węzła ujściowego,
3. segment – pomiędzy wyjściowym DCE i ujściowym DTE w węźle ujściowym.

W segmentach pierwszym i trzecim połączenia wirtualnego stosowany jest protokół warstwy trzeciej standardu X.25, w segmencie drugim natomiast na potrzeby obsługi wymiany pakietów implementowany jest protokół wewnętrzny sieci (rysunek 6.35).



DTE – Data Terminal Equipment
 DCE – Data Circuit Equipment
 PSE – Packet-Switching Exchange

Rysunek 6.35. Organizacja trzech segmentów połączenia wirtualnego

W każdym z omawianych trzech segmentów sterowanie przepływem odbywa się z wykorzystaniem metody okna. Sterowanie przepływem na poziomie trasy jest realizowane przez mechanizmy trzech połączonych kaskadowo i wzajemnie zsynchronizowanych metod okna. Synchronizację stosuje się do osiągnięcia podstawowego celu sterowania przepływem w połączeniu wirtualnym, jakim jest dopasowanie szybkości

wprowadzania pakietów w węzle źródłowym do szybkości ich wyprowadzania w węzle ujściowym połączenia.

Synchronizację okien uzyskuje się za pomocą prostego mechanizmu powstrzymywania generowania powiadomień pozytywnych. Wysyłanie ich w kierunku do węzła źródłowego przez węzły odbiorcze pierwszego lub drugiego segmentu jest wstrzymywane, jeżeli – odpowiednio – okno w drugim lub trzecim segmencie połączenia wirtualnego jest w pełni wykorzystane (pełne wykorzystanie okna oznacza, że chwilowo wstrzymana jest transmisja pakietów w kierunku do węzła ujściowego, ponieważ liczba pakietów wysłanych i dotychczas niepotwierdzonych jest równa rozmiarowi okna).

Przy założeniu, że rozmiary okien w segmentach wynoszą, odpowiednio, w_1 , w_2 oraz w_3 , przykładowe działanie mechanizmu spowalniania wprowadzania pakietów jest następujące:

➤ jeżeli okno segmentu drugiego o rozmiarze w_2 jest w pełni wykorzystane, tzn. jeżeli na obsługujących połączenie wirtualne różnych trasach od DCE w węzle źródłowym do DCE w węzle ujściowym znajduje się w_2 niepotwierdzonych pakietów, to nowy pakiet przekazywany w segmencie pierwszym (z DTE węzła źródłowego do DCE węzła źródłowego) zostanie przyjęty (pod warunkiem, że jest wolny bufor w pamięci buforowej), ale jego przyjęcie nie będzie potwierdzone,

➤ przyjęcie nowego pakietu w węzle odbiorczym segmentu pierwszego (DCE węzła źródłowego) jest potwierdzane przez wysłanie powiadomienia pozytywnego ACK_1 wtedy, gdy węzeł nadawczy segmentu drugiego (DCE węzła źródłowego) otrzyma powiadomienie pozytywne ACK_2 z węzła odbiorczego segmentu drugiego (DCE węzła ujściowego) otwierającego możliwość kontynuacji transferu pakietów w segmencie drugim.

W omawianym przypadku sterowanie przepływem na poziomie trasy jest realizowane na jednym poziomie przez wykorzystanie mechanizmów trzech połączonych kaskadowo metod okna, synchronizowanych mechanizmem powstrzymywania powiadomień. Jednopoziomowe sterowanie przepływem oznacza, że mechanizmy okna i powstrzymywania powiadomień są wykorzystywane jednocześnie do rozwiązywania zadań zapobiegania przeciążeniom węzła ujściowego, zachowania kolejności napływu pakietów, wykrywania i odzyskiwania straconych pakietów oraz wykrywania kopii pakietów.

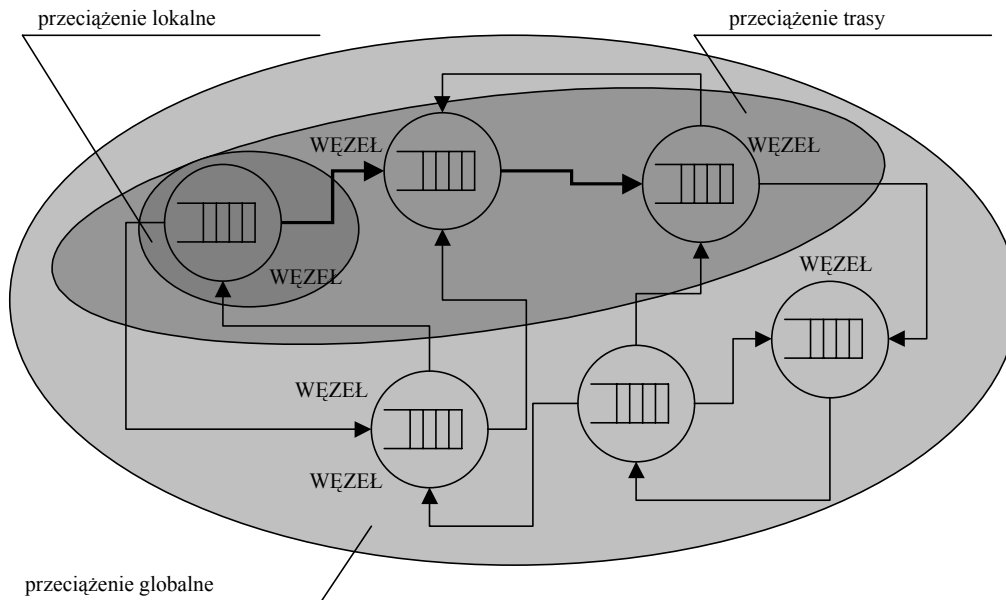
Zachowanie kolejności napływu pakietów uzyskuje się przez odrzucanie w węzłach odbiorczych każdego segmentu pakietów napływających poza kolejnością. Wymagana kolejność napływu jest wymuszana powiadomieniami.

Zadanie wykrywania i odzyskiwania traconych i odrzucanych pakietów polega na wykorzystaniu mechanizmów retransmisji i potwierdzania. Jeżeli ujściowe DCE (DCE węzła ujściowego) nie wysłało potwierdzenia poprawnie odebranego pakietu źródłowego DCE (DCE węzła źródłowego) z powodu przeciążenia węzła ujściowego, to węzeł źródłowy retransmituje pakiet. Retransmisja pakietu rozpoczyna się po upływie czasu równego wartości zwłoki czasowej od chwili zakończenia poprzedniej

transmisji danego pakietu (ang. *time-out*). Retransmisja pakietu odbywa się na podstawie założenia, że przyczyną braku potwierdzenia jest strata (zagubienie) pakietu podczas transmisji lub jego odrzucenie z powodu braku pamięci buforowej w węźle ujściowym.

Dodatkową przyczyną retransmisji pakietu może być niewłaściwy dobór wartości jednego z parametrów metody okna, jakim jest długość przedziału czasu, po upływie którego następuje retransmisja.

Jeżeli długość tego przedziału czasu t_{out} jest mniejsza od sumarycznego czasu obsługi pakietu T (suma składowych: opóźnienie propagacji pakietu – t_{PAKp} , czas transmisji pakietu – t_{PAKt} , czas przetwarzania pakietu – t_{PAKr} , opóźnienie propagacji potwierdzenia – t_{ACKp} , czas transmisji potwierdzenia – t_{ACKt} oraz czas przetwarzania pakietu – t_{ACKr} , tzn. $T = (t_{PAKp} + t_{PAKt} + t_{PAKr} + t_{ACKp} + t_{ACKt} + t_{ACKr}) > t_{out}$, to w węźle źródłowym generowana jest kopia pakietów (rysunek 6.36).



Rysunek 6.36. Hierarchia przeciążeń w sieci z komutacją pakietów

W omawianym sposobie sterowania przepływem prawdopodobieństwo generowania kopii pakietów jest zawsze większe od zera. Wartość tego prawdopodobieństwa zależy od wartości parametrów metody okna oraz parametrów sieci i może być minimalizowana.

Liczba retransmisji tego samego pakietu jest ograniczona. Liczba ta jest wartością jednego z parametrów (obok rozmiaru okna i wartości zwłoki czasowej) metody okna. Brak potwierdzenia, mimo wykonania ustalonej liczby retransmisji, powoduje rozłą-

czenie połączenia wirtualnego przez węzeł źródłowy. Rozłączenie połączenia wirtualnego z podanego powodu jest równoważne odwołaniu się omawianej procedury sterowania przepływem do nadrzędnego systemu sterowania.

6.4.3.6.4. STEROWANIE PRZEPLYWEM NA POZIOMIE TRASY W ARCHITEKTURZE SNA

Architektura SNA (ang. *Systems Network Architecture*) jest architekturą scentralizowanych i zdecentralizowanych systemów rozproszonego przetwarzania i komunikacji.

W architekturze SNA wyróżnia się cztery kategorie urządzeń:

- komputery główne (ang. *host computer*),
- sterowniki komunikacyjne (ang. *communication controller*),
- sterowniki skupień urządzeń końcowych (ang. *terminal cluster controller*),
- urządzenia końcowe (ang. *terminal device*).

Systemy zgodne z architekturą SNA mają dwupoziomą strukturę sieci: poziom niższy obejmuje urządzenia końcowe i sterowniki skupień (zbiorów) urządzeń końcowych, poziom wyższy natomiast – komputery główne i sterowniki komunikacyjne. Poszczególne elementy poziomu wyższego są, z punktu widzenia rozproszonego systemu komunikacji, węzłami sieci (ang. *subarea nodes*), do których dołącza się elementy poziomu niższego. Tak definiowane węzły można interpretować jako miejsca koncentracji ruchu generowanego i adresowanego do określonego podzbioru urządzeń końcowych sieci. W węzłach sieci realizuje się tzw. funkcje graniczne (ang. *boundary functions*), do których należą m.in. konwersja adresów globalnych i lokalnych, koncentracja ruchu wprowadzanego do węzłów sieci z poziomu niższego oraz rozdzielanie ruchu do elementów poziomu niższego.

Rozproszony system komunikacji pomiędzy węzłami sieci charakteryzują: komunikacja w trybie pakietowym z wykorzystaniem statycznych metod doboru trasy, sterowanie przepływem i globalny system adresowania.

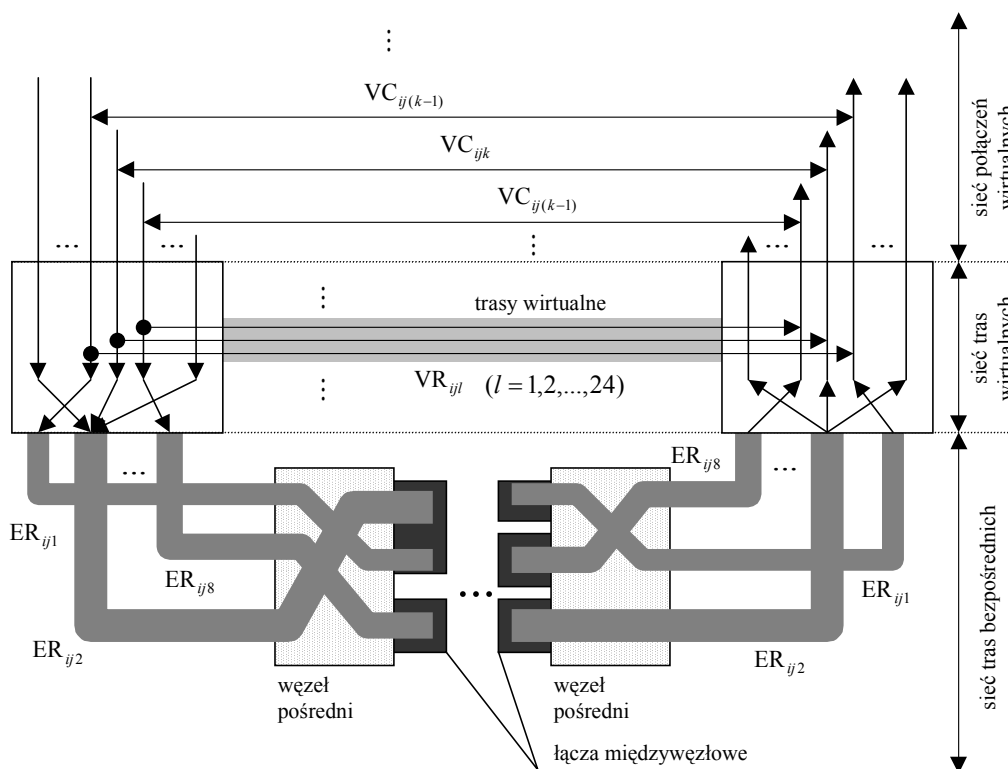
Sieci SNA są sieciami połączeń wirtualnych. Każdej sesji użytkownika sieci, tzn. każdemu połączeniu wirtualnemu VC (ang. *Virtual Connection*), odpowiadają wzajemnie odseparowane: trasa wirtualna VR (ang. *Virtual Route*) i trasa bezpośrednia ER (ang. *Explicit Route*). Trasy wirtualna i bezpośrednia są dobierane w czasie ustanawiania połączenia wirtualnego VC. Sterowanie przepływem dotyczy przepływu w trasach wirtualnych.

Dla każdej pary węzłów (na poziomie wyższym) źródłowego i ujściowego utrzymywana jest uporządkowana lista co najwyżej ośmiu różnych tras fizycznych określanych mianem tras bezpośrednich ER. Listy tras bezpośrednich, podobnie jak trasy bezpośrednie, są w sieci niezienne. Dowolna zmiana struktury sieci wymaga redefiniowania list i tras bezpośrednich. Każda trasa bezpośrednia, jednoznacznie identyfikowana numerem trasy, jest definiowana jako uporządkowany ciąg łączy sieci. Jeżeli podczas użytkowania danej trasy zostanie stwierdzona jej niesprawność, to zastępuje ją kolejna trasa z utrzymywanej listy tras bezpośrednich.

Trasy (ścieżki) wirtualne VR, definiowane dla każdej pary węzłów źródłowych i ujściowych, są nadrzędne w stosunku do tras bezpośrednich ER. Pomiędzy każdą

parą węzłów możliwe jest organizowanie trzech zbiorów połączeń wirtualnych różniących się poziomami priorytetu. W każdym ze zbiorów może być co najwyżej osiem połączeń wirtualnych. Pozwala to na organizowanie i jednoczesne utrzymywanie do 24 takich połączeń pomiędzy każdą parą węzłów. Lista połączeń wirtualnych, jednoznacznie identyfikowanych ich numerami, jest utrzymywana w każdym węźle sieci.

Ustanawianie nowego, żądanego w węźle źródłowym, połączenia wirtualnego polega na przeglądnięciu listy tras wirtualnych VR, utrzymywanych w danym węźle, i wyborze pierwszej dostępnej trasy wirtualnej o zadanym poziomie priorytetu. Możliwe jest zwielokrotnienie wielu połączeń wirtualnych VC w jednej trasie wirtualnej VR, podobnie jak możliwa jest obsługa wielu tras wirtualnych VR przez jedną trasę bezpośrednią ER, których z kolei wiele tras bezpośrednich ER może być zwielokrotnionych w jednym łączy sieci (rysunek 6.37).



Rysunek 6.37. Połączenia i trasy wirtualne

Wyróżnienie tras bezpośrednich i tras wirtualnych w sieci SNA powoduje odizolowanie (odizolowanie) trasy wirtualnej od fizycznej konfiguracji sieci. W konsekwencji takiej izolacji warstw do przekazywania pakietów w sieci jest wykorzystywany numer identyfikacyjny trasy bezpośredniej, numer identyfikacyjny trasy

wirtualnej natomiast jest wykorzystywany (sprawdzany) tylko w węźle ujściowym na potrzeby sterowania przepływem. Wymieniona cecha sieci SNA prowadzi do redukcji – w porównaniu do sieci z połączeniami wirtualnymi, w których pakiety identyfikowane są numerami połączeń wirtualnych – ilości pamięci buforowej i zasobów obliczeniowych niezbędnych do obsługi strumieni pakietów w węzłach pośrednich sieci. W przypadku identyfikacji pakietów numerami połączeń wirtualnych, komutacja pakietów w węzłach pośrednich sieci wymaga utrzymywania znaczących ilości informacji potrzebnych do odwzorowywania (mapowania) numerów wszystkich połączeń wirtualnych przechodzących przez dany węzeł na numery tras i łączy międzywęzłowych tras.

W sieci SNA przepływ jest sterowany w sieci tras wirtualnych (sieci wyższego poziomu) i dotyczy przepływu pomiędzy węzłami źródłowym i ujściowym dla każdej z tras wirtualnych z osobna. Do sterowania przepływem na trasie wirtualnej stosowana jest metoda krokowa (ang. *VR pacing*), łącząca cechy metod sterowania przepływem na poziomie trasy (ang. *end-to-end level*) oraz sterowania przepływem na poziomie łącza (ang. *hop level*).

Stosowana w sieci SNA metoda krokowa jest złożeniem mechanizmów metody okna i mechanizmów rezerwacji pojemności pamięci buforowej w węźle ujściowym:

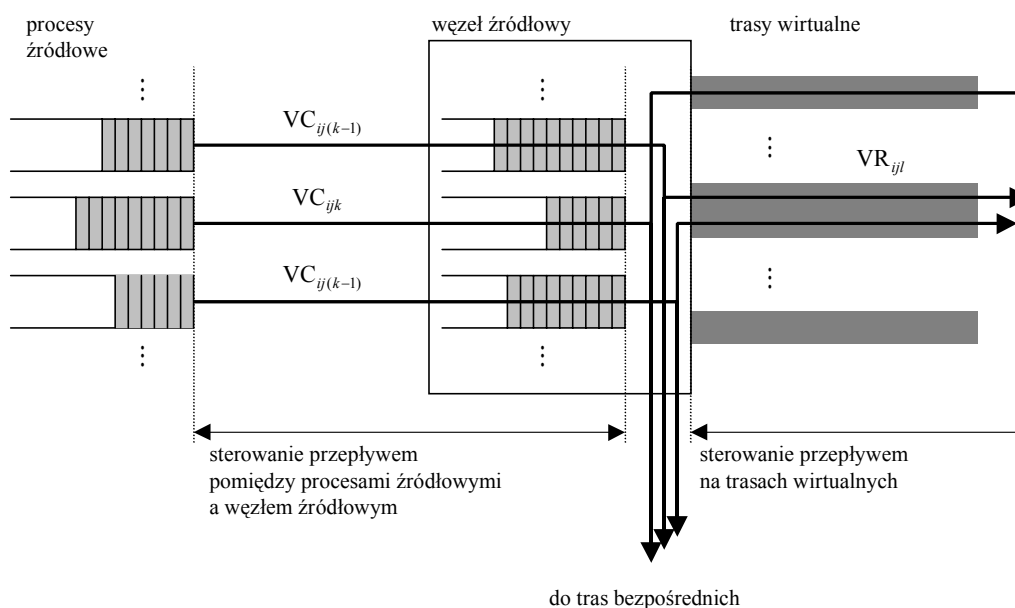
➤ liczba wszystkich pakietów wysłanych przez i -ty węzeł źródłowy i znajdujących się na trasie wirtualnej VR_{ijl} ($i, j = 1, 2, \dots, N, i \neq j, l = 1, 2, \dots, L_{ij}$, gdzie N jest liczbą wszystkich węzłów sieci, a L_{ij} jest liczbą wszystkich tras wirtualnych pomiędzy i -tym węzłem źródłowym oraz j -tym węzłem ujściowym) jest ograniczona z góry rozmiarem okna w_{ijl} ,

➤ rozpoczęcie nadawania kolejnej grupy w_{ijl} pakietów przez i -ty węzeł źródłowy jest możliwe po wcześniejszym wysłaniu żądania i uzyskaniu potwierdzenia rezerwacji w_{ijl} buforów w pamięci buforowej j -tego węzła ujściowego.

Dostępność żądanej liczby w_{ijl} buforów powoduje wygenerowanie przez węzeł ujściowy potwierdzenia żądania; w innym przypadku generacja potwierdzenia jest wstrzymywana do chwili zwolnienia żądanej liczby buforów.

Rozmiar okna w_{ijl} spełnia warunek: $h_{ER} \leq w_{ijl} \leq 3h_{ER}$, gdzie h_{ER} jest długością trasy bezpośredniej (liczba łączy międzywęzłowych składających się na trasę bezpośrednią), ponad którą utrzymywane jest połączenie wirtualne VR_{ijl} . Definiowany zakres zmienności rozmiaru okna pozwala m.in. na kompensowanie zmian długości tras pośrednich obsługujących daną trasę wirtualną oraz dynamiczne dostosowywanie rozmiaru okna do stanu sieci. W omawianej procedurze sterowania przepływem aktualny rozmiar okna w_{ijl} zależy zarówno od aktualnego obciążenia węzła ujściowego, jak i aktualnego obciążenia węzłów pośrednich.

Sterowanie przepływem w sieci SNA na poziomie tras wirtualnych VR jest uzupełniane sterowaniem przepływem na poziomie poszczególnych połączeń wirtualnych VC, utrzymywanych pomiędzy procesem źródłowym a węzłem źródłowym (rysunek 6.38).



Rysunek 6.38. Uzupełnianie sterowania przepływem na poziomie połączeń wirtualnych sterowaniem przepływem pomiędzy procesem źródłowym a węzłem źródłowym

Sterowanie przepływem na tym odcinku połączenia wirtualnego, podobnie jak sterowanie przepływem na poziomie trasy wirtualnej, odbywa się metodą krokową. Warunkiem rozpoczęcia nadawania pakietów przez proces źródłowy jest rezerwacja wcześniej określonej pojemności pamięci buforowej w węźle źródłowym. W tym przypadku sterowanie przepływem na poziomie połączenia wirtualnego VC jest sterowaniem na poziomie transportowym.

6.4.3.6.5. STEROWANIE PRZEPLYWEM NA POZIOMIE TRASY W SIECI X.25

Protokół X.25 jest protokołem wykorzystywanym w publicznych sieciach transmisji danych PDN (ang. *Public Data Network*). W sieciach PDN wyróżnia się dwie klasy rozwiązań: sieci z komutacją kanałów CSPDN (ang. *Circuit-Switched PDN*) i sieci z komutacją pakietów PSPDN (ang. *Packet-Switched PDN*).

Jedną z różnic pomiędzy CSPDN i PSPDN jest sposób kontroli błędów i sterowania przepływem. W sieciach CSPDN praktycznie nie ma żadnych procedur kontroli błędów i sterowania przepływem danych. Zadania te muszą być i są realizowane przez procesy będące użytkownikami sieci CSPDN. W odróżnieniu do sieci CSPDN, w sieciach PSPDN dla każdego łącza danych, pomiędzy każdą parą węzłów komutacji pakietów, implementowane są wyrafinowane procedury kontroli błędów i sterowania przepływem. Dzięki implementacji tych procedur jakość usług komunikacyjnych dostarczanych przez sieci PSPDN jest znacznie wyższa od jakości usług oferowanych przez sieci CSPDN.

W sieciach PSPDN wszystkie pakiety warstwy sieciowej, wprowadzane do sieci, są transferowane z DTE (ang. *Data Terminal Equipment*) do lokalnego węzła komutacyjnego (węzła źródłowego) pakietów PSE (ang. *Packet-Switching Exchange*) z wykorzystaniem usług dostarczanych przez warstwę łącza danych. W węźle PSE napływające pakiety są zapamiętywane (w trybie zapamiętaj i prześlij dalej (ang. *store-and-forward*)), a następnie sprawdzane są ich adresy docelowe (adresy sieciowe węzłów ujściowych).

W każdym węźle PSE implementowana jest tablica tras (ang. *routing directory*) specyfikująca trasy dla każdego z sieciowych adresów docelowych. Po odebraniu pakietu i ustaleniu jego trasy do węzła ujściowego, pakiety z węzła PSE są wysyłane z maksymalną dopuszczalną szybkością transferu. Podobnie dla każdego pakietu odbieranego (i przechowywanego) w pośrednich węzłach komutacyjnych PSE, znajdujących się na trasie od węzła źródłowego do węzła ujściowego, ustalana jest dalsza trasa, współdzielona z pakietami pochodzącymi z innych źródeł, po uwzględnieniu aktualnego obciążenia sieci. W ujściowym PSE, determinowanym przez sieciowy adres ujścia, pakiet jest odbierany i przekazywany do ujściowego DTE.

Jest możliwe, że pakiety generowane przez różne DTE napływają jednocześnie do węzła komutacyjnego PSE i są przekazywane tymi samymi trasami. Możliwość obsługi pakietów napływających z różnych DTE jednocześnie do tego samego PSE są determinowane możliwością przechowywania napływających pakietów (strategia przydziału pojemności pamięci buforowej) i możliwością wyprowadzania ich z PSE do innych PSE na trasach do ujściowych PSE (metody przydziału przepustowości tras). W celu ograniczenia czasu oczekiwania na obsługę wiadomości (zwłaszcza długich), wiadomości generowane w warstwie transportowej są, przed ich przekazaniem do warstwy sieciowej, dzielone na pakiety (pakietyzacja wiadomości). Odwrotny proces kompletacji wiadomości realizowany jest w procesie warstwy transportowej implementowanym w ujściowym DTE.

W sieciach X.25 do przekazywania pakietów warstwy sieciowej pomiędzy DTE i PSE, PSE i PSE oraz PSE i DTE wykorzystywane są usługi warstwy łącza danych dostarczane przez protokół warstwy łącza danych oparty na procedurze HDLC (ang. *High Data Link Control procedure*) (w standardzie X.25 procedura HDLC w wersji z trybem ABM (ang. *Asynchronous Balance Mode*) jest określana mianem LAPB (ang. *Link Access Procedure version B*). Ze względu na wysoką jakość i niezawodność usług (kontrola błędów) protokołów stosujących procedurę HDLC (lub jej odmiany), w warstwie sieciowej uwydatnione są funkcje związane ze sterowaniem przepływem.

W implementowanych w sieciach mechanizmach sterowania przepływem stosuje się metodę okna przesuwnego. Przepływ pakietów jest sterowany oddzielnie dla każdego połączenia logicznego i dla każdego kierunku połączenia. Przepływy pakietów dla każdego połączenia z DTE do PSE oraz z PSE do DTE są sterowane oddzielnie.

Implementacja mechanizmu okna oznacza, że wszystkie pakiety informacyjne (pakiety danych) mają numery sekwencyjne: nadawczy $P(S)$ (ang. *send sequence number*) i odbiorczy $P(R)$ (ang. *receive sequence number*). Numer sekwencyjny nadawczy $P(R)$, zawarty w każdym pakiecie, dotyczy przepływu pakietów w kierunku

przeciwnym. Alternatywnie, jeżeli brak jest pakietu oczekującego na transfer w kierunku przeciwnym, numer sekwencyjny $P(R)$ jest przekazywany za pomocą specjalnego pakietu zarządzającego gotowością do odbioru RR (ang. *Receive-Ready*).

Numer sekwencyjne $P(S)$ i $P(R)$, w przypadku transferu dwukierunkowego pakietów informacyjnych, są przekazywane w nagłówkach pakietów informacyjnych wymienianych pomiędzy węzłami końcowymi trasy (ang. *piggyback*). W razie braku transferu jednokierunkowego numery sekwencyjne $P(R)$ są przekazywane za pomocą pakietów zarządzających.

Pierwsze pakiety, dla każdego z kierunków transferu w połączeniu logicznym, otrzymują numery sekwencyjne równe 0. Każdy następny pakiet, transferowany w tym samym kierunku, zawiera numer sekwencyjny odbiorczy $P(S)$, którego wartość jest zwiększona o 1. Liczba pakietów tego samego kierunku połączenia, które mogą być wysłane przed uzyskaniem potwierdzenia, jest ograniczona do ustalonego rozmiaru okna w ($w = 7$ lub $w = 127$). Ze względu na implementowaną w protokole procedurę ciągłej retransmisji grupowej (ang. *continuous-go-back-w ARQ*) rozmiar okna w determinuje liczbę różnych numerów sekwencyjnych używanych do jednoznacznej identyfikacji transferowanych pakietów i detekcji duplikatów pakietów: $w + 1 = 8$ lub $w + 1 = 128$. Numeracja mod $(w + 1)$ umożliwia przydzielanie nadawanym pakietom $(w + 1)$ kolejnych numerów, tzn. 0, 1, ..., w . Numerowanie pakietów zgodnie z zasadą mod $(w + 1)$ powoduje, że po wykorzystaniu numerów z zakresu 0, ..., w numery pakietów zaczynają się cyklicznie powtarzać. Taki schemat numerowania pakietów zapobiega niejednoznaczności numeracji (ang. *sequence number starvation*) dla procedury ciągłej retransmisji grupowej ARQ.

Jeżeli węzeł nadrzędny zainicjuje transfer pakietów aż do wyczerpania okna, to transfer następnych pakietów jest wstrzymywany do chwili otrzymania albo kolejnego pakietu informacyjnego, albo pakietu zarządzającego RR, zawierających numer sekwencyjny $P(R)$. Pakiet RR wskazuje gotowość węzła podrzędnego do przyjęcia następnych pakietów, a zawarty w nim numer sekwencyjny odbiorczy $P(R)$ wskazuje numer pakietu oczekiwanego w węźle podrzędnym.

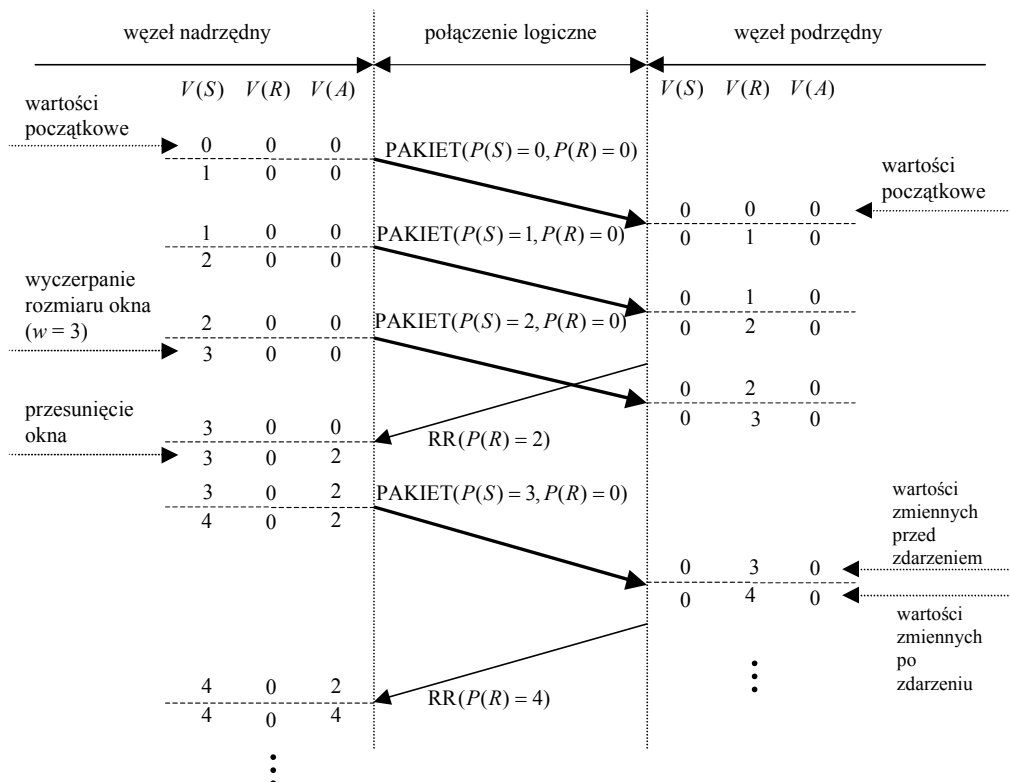
Implementacja mechanizmu okna wymaga utrzymywania przez DTE i PSE trzech zmiennych dla każdego aktywnego połączenia logicznego:

➤ $V(S)$ – zmiennej nadawania (ang. *send sequence variable*), jej wartość, równa wartości $P(S)$, będzie przypisana następnemu pakietowi informacyjnemu wysyłanemu w danym kanale logicznym,

➤ $V(R)$ – zmiennej odbioru (ang. *receive sequence variable*), jej wartość jest równa wartości $P(S)$ następnego w kolejności pakietu informacyjnego, którego odbiór jest oczekiwany w danym kanale logicznym,

➤ $V(A)$ – zmiennej potwierdzenia (ang. *acknowledgement variable*) stosowanej do wstrzymywania przepływu pakietów informacyjnych (aktualna wartość zmiennej $V(A)$ określa położenie strzałki (wskaźnika) końcowego okna; znajomość wartości zmiennej $V(A)$ pozwala wyznaczyć położenie strzałki (wskaźnika) prowadzącej okno, której wartość jest równa $V(A) + w$).

Sterowanie przepływem za pomocą mechanizmu okna implementuje się według następującego scenariusza (rysunek 6.39 – dla czytelności rysunku przyjęto istnienie jednego połączenia logicznego i jednokierunkowy przepływ pakietów informacyjnych):



Rysunek 6.39. Sterowanie przepływem za pomocą mechanizmu okna – transfer pakietów informacyjnych

- wszystkim wymienionym trzem zmiennym w chwili ustanawiania (lub odnawiania) połączenia logicznego przypisuje się wartość 0,
- każdemu pakietowi informacyjnemu przygotowanemu do transferu przypisuje się sekwencyjny $P(S)$, którego wartość jest równa aktualnej wartości zmiennej $V(S)$ zwiększanej następnie modulo- $(w + 1)$ (modulo-8 lub modulo-128),
- podobnie odbiór każdego pakietu informacyjnego lub pakietu zarządzającego RR oznacza otrzymanie odbiorczego numeru sekwencyjnego $P(R)$, stosowanego następnie do aktualizacji wartości zmiennej $V(A)$,
- węzeł nadawczy może kontynuować nadawanie pakietów albo do chwili wyczerpania okna (do chwili, gdy zwiększana wartość zmiennej $V(A)$ osiągnie wartość w , albo do chwili otrzymania pakietów informacyjnych lub pakietów zarządzających RR zawierających numery sekwencyjne $P(R)$ zwiększające aktualną wartość zmiennej potwierdzenia $V(A)$ (przesuwające wskaźnik końca okna)),

wszystkich lub w wybranych połączeniach logicznych. Do chwilowego wstrzymywania napływu pakietów w połączeniu logicznym, w którym do sterowania przepływem stosowany jest mechanizm okna, wykorzystywany jest pakiet zarządzający braku gotowości do odbioru – pakiet RNR (ang. *Receive-Not-Ready*). Przykładową sekwencję pakietów informacyjnych oraz pakietów zarządzających pokazano na rysunku 6.40.

Każdy pakiet RNR zawiera numer sekwencyjny $P(R)$ definiujący, po odebraniu w węźle nadrzędnym (nadajniku), nową wartość zmiennej potwierdzenia $V(A)$ dla danego połączenia. Po odbiorze pakietu RNR węzeł nadrzędny wstrzymuje nadawanie kolejnych pakietów, ale obsługa już wcześniej nadanych i będących w trakcie obsługi w chwili otrzymania pakietu RNR jest kontynuowana. Nadawanie kolejnych pakietów jest kontynuowane przez węzeł nadrzędny po otrzymaniu pakietu zarządzającego RR (wartości numerów sekwencyjnych odbiorczych w pakietach RR i poprzedzającym go pakiecie RNR mogą być różne – pakiet RNR mógł być wygenerowany przed zakończeniem transferu pakietów, będących w trakcie obsługi w chwili generowania tego pakietu RNR; jeżeli wartość $P(R)$ pakietu RR jest różna od wartości $P(R)$ pakietu RNR, to modyfikuje wartość zmiennej potwierdzenia $V(A)$ węzła nadrzędnego).

6.4.3.7. METODA OKNA W ZADANIACH STEROWANIA PRZEPLÝWEM NA POZIOMIE TRASY

Większość praktycznie implementowanych procedur sterowania przepływem na poziomie trasy (pomiędzy węzłami źródłowym i ujściowym) korzysta z różnych sposobów implementacji metody okna, których podstawowy podział wynika ze sposobu wyznaczania rozmiaru okna:

- grupa pierwsza implementacji metod okna to zastosowania, w których rozmiar okna jest stały, niezależny od aktualnego obciążenia sieci lub jej elementów (łącza, pamięci buforowej węzła, trasy, sieci),

- grupa druga implementacji metod okna to zastosowania, w których rozmiar okna jest dynamicznie zmieniany, w zależności od aktualnego stanu zajętości pamięci buforowej węzłów ujściowych i (lub) pośrednich sieci; na ogół zwiększenie obciążenia powoduje zmniejszenie rozmiaru okna.

Poprawę skuteczności działania metody okna w zadaniach sterowania przepływem na poziomie trasy uzyskuje się trzema sposobami:

- sterowaniem dwupoziomowym, polegającym na stosowaniu innych mechanizmów (np. rezerwacji) do modyfikacji wartości parametrów metody okna,

- sterowaniem sekwencyjnym, polegającym na uzupełnieniu metody okna o inne mechanizmy, których działanie polega na przeciwdziałaniu takim skutkom przeciążenia, którym nie zapobiega stosowanie metody okna,

- sterowaniem hierarchicznym, polegającym na budowaniu hierarchii procedur sterowania przepływami korzystających z metody okna. Przedmiotem sterowania są przepływy o różnym stopniu koncentracji ruchu (np. połączenia wirtualne i trasy wirtualne) i różnym zakresie oddziaływania (np. łącze i trasa).

Niezależnie od sposobu wykorzystywania metody okna w danej procedurze sterowania przepływem, skuteczność jej zależy od wartości dwóch krytycznych parametrów: rozmiaru okna (ang. *window size*) i zwłoki czasowej (czasu oczekiwania na retransmisję) (ang. *time-out*).

Opracowano wiele modeli analitycznych i symulacyjnych, których celem jest badanie wpływu wartości wymienionych parametrów metody okna na wartość miar (wskaźników) jakości przepływu w sieci. Najczęściej stosowanymi w tych badaniach miarami jakości przepływu są: przepustowość sieci lub jej elementów (łącza, trasy) oraz opóźnienie (średnie, maksymalne, fluktuacje) pakietów lub wielopakietowych wiadomości w sieci lub jej elementach (w łączu, na trasie).

6.4.4. STEROWANIE PRZEPLYWEM NA POZIOMIE DOSTĘPU DO SIECI

6.4.4.1. CEL STEROWANIA PRZEPLYWEM NA POZIOMIE DOSTĘPU DO SIECI

Celem sterowania przepływem na poziomie sieci jest regulacja intensywności wprowadzania nowych pakietów do sieci w zależności od jej obciążenia, w szczególności hamowanie (dławienie) intensywności napływu do sieci nowych pakietów spoza sieci wtedy, gdy sieć jest w stanie przeciążenia.

Decyzje podejmowane przez procedury sterowania przepływem na poziomie sieci, tzn. decyzje dotyczące intensywności wprowadzania nowych do niej pakietów, są podejmowane na podstawie rodzaju występujących w sieci przeciążeń oraz – związanych z nimi – wartości miar przeciążeń gromadzonych i dostępnych w węzłach źródłowych (węzłach dostępu) sieci. Przyjęta miara przeciążenia sieci jest jednocześnie miarą jakości działania procedury sterowania przepływem (bezpośrednio lub pośrednio).

Na potrzeby implementacji różnych procedur sterowania przepływem na poziomie sieci wyróżnia się następujące rodzaje przeciążeń sieci:

1. Przeciążenia lokalne (ang. *local congestion*). Miarą przeciążenia sieci jest stan zajętości zasobów o określonej lokalizacji w sieci i określonym przeznaczeniu (np. zajętość pamięci buforowej w węźle źródłowym sieci).

2. Przeciążenia tras (ang. *selective congestion*). Miarą przeciążenia sieci jest stan zajętości wybranych zasobów wykorzystywanych do obsługi ruchu na trasie lub trasach (np. zajętość pamięci buforowych w węzłach zlokalizowanych na trasie od węzła źródłowego do węzła ujściowego).

3. Przeciążenia globalne (ang. *global congestion*). Miarą przeciążenia sieci jest stan zajętości zasobów całej sieci o ściśle zdefiniowanym przeznaczeniu (np. zajętość pamięci buforowych we wszystkich węzłach sieci).

Informacja o warunkach powstawania przeciążeń, gromadzona w węzłach wejściowych sieci (w węzłach dostępowych do sieci), jest wykorzystywana do podejmowania decyzji dotyczących akceptacji, spowalniania lub wstrzymywania napływu do sieci strumieni pakietów zewnętrznych (pakietów wprowadzanych do sieci).

Procedury sterowania przepływem na poziomie dostępu do sieci charakteryzują i jednocześnie różnią od procedur sterowania na innych poziomach: zakres informacji o warunkach powstawania przeciążeń, miejsca gromadzenia oraz sposoby wykorzystywania tych informacji.

Dostęp procedur sterowania przepływem na poziomie sieci do informacji o stanie całej sieci powoduje, że cele sterowania na tym poziomie różnią się od celów sterowania na poziomach łącza danych i trasy. Różnica celów omawianych trzech poziomów sterowania przepływem wynika z zakresu oddziaływania sterowania determinowanego (ale i determinującego) możliwościami detekcji określonych rodzajów przeciążeń oraz ich skutków. I tak:

1. Zakres sterowania przepływem na poziomie łącza danych dotyczy przepływu pomiędzy dwoma sąsiednimi węzłami sieci, co pozwala na ograniczanie napływu pakietów do pamięci buforowej węzła odbiorczego łącza. Istnieje możliwość zapobiegania przeciążeniom lokalnym (ang. *local congestion*) i tym samym przeciwdziałania zakleszczeniom bezpośrednim (ang. *store-and-forward deadlock*).

2. Zakres sterowania przepływem na poziomie trasy obejmuje przepływ (o różnym stopniu koncentracji) pomiędzy dwoma wyróżnionymi węzłami końcowymi sieci, co pozwala na ograniczanie napływu pakietów do pamięci buforowej węzła ujściowego. Istnieje możliwość zapobiegania przeciążeniom lokalnym w węzłach ujściowych (na krańcach sieci) (ang. *destination congestion*) oraz przeciwdziałania powstawaniu zakleszczeń kompletacji w węzłach ujściowych (ang. *reassembly deadlock at the destination*).

3. Zakresem sterowania przepływem na poziomie dostępu do sieci, polegającym na ingerowaniu w ilość i szybkość wprowadzania nowego ruchu do sieci, objęte są wszystkie przepływy w sieci, co pozwala na koordynację ograniczania napływu pakietów do pamięci buforowych wszystkich węzłów sieci. Istnieje możliwość zapobiegania przeciążeniu sieci jako całości (ang. *overall internal congestion*).

Jakkolwiek mechanizmy procedur sterowania przepływem na poziomie łącza i na poziomie trasy wpływają pośrednio na szybkość wprowadzania nowych pakietów do sieci w węzłach źródłowych, są to jednak działania dotyczące przepływu w ściśle określonych fragmentach sieci. W przypadku procedur sterowania stosowanych na poziomie łącza są to np. mechanizmy oddziaływania zwrotnego (ang. *back-pressure mechanism*), natomiast w przypadku procedur sterowania implementowanych na poziomie trasy są to np. mechanizmy spowalniania (ang. *slowdown mechanism*). Lokalność i fragmentaryczność tych oddziaływań powoduje, że niezbędny jest wyższy poziom sterowania przepływem, pozwalający na oddziaływanie na przepływy w całej sieci. Warunki takie spełnia sterowanie przepływem na poziomie dostępu do sieci, którego zadaniem jest limitowanie całkowitej liczby pakietów w sieci.

Do najczęściej stosowanych metod sterowania przepływem na poziomie dostępu do sieci zalicza się:

➤ metodę izarytmiczną (przepustek) (ang. *isarithmic scheme*) – polegającą na ograniczeniu liczby pakietów jednocześnie przebywających w sieci przez stosowanie

ustalanej liczby przepustek (pozwoleń, zetonów) (ang. *permit*), będących w obiegu pomiędzy wszystkimi węzłami wejściowymi sieci, na dostęp pakietów do sieci,

➤ metodę bufora wejściowego (ograniczonej pojemności buforów wejściowych) IBLS (ang. *Input Buffer Limit Scheme*), polegającą na ograniczaniu pojemności pamięci buforowych dostępnych dla ruchu wejściowego w każdym z węzłów źródłowych sieci, podczas gdy ruch tranzytowy ma nieograniczony dostęp do pamięci buforowych,

➤ metodę z pakietem dławiącym (pakietów dławiących) CHPS (ang. *Choke Packet Scheme*) – selektywne zapobieganie powstawaniu przeciążeń, polegającą na generowaniu w obszarach sieci zagrożonych przeciążeniem specjalnych pakietów, powodujących dławienie intensywności napływu pakietów zewnętrznych do wskazanego obszaru sieci,

➤ metodę stopniowych ograniczeń GR (ang. *Gradual Restriction*), polegającą na stopniowaniu dławienia (przez wprowadzanie progów) intensywności napływu pakietów zewnętrznych do sieci w zależności od długości kolejek pakietów oczekujących na obsługę w węzłach sieci,

➤ metodę okna, polegającą na założeniu, że ograniczenie liczby pakietów w każdym z połączeń wirtualnych, utrzymywanych w sieci, i ograniczenie z góry liczby wszystkich połączeń wirtualnych w sieci, powoduje pośrednio zabezpieczenie całej sieci przed przeciążeniem.

6.4.4.2. METODA IZARYTMICZNA

Idea metody izarytmicznej (metody przepustek) wywodzi się z oczywistego spostrzeżenia, że podstawową przyczyną przeciążeń sieci, tj. ograniczonych zasobów sieci z komutacją pakietów, jest nadmiar (w porównaniu z zasobami) pakietów w sieci. Zapobieganie nadmiarowi pakietów w sieci jest warunkiem koniecznym, ale nie wystarczającym, przeciwdziałania przeciążeniom sieci.

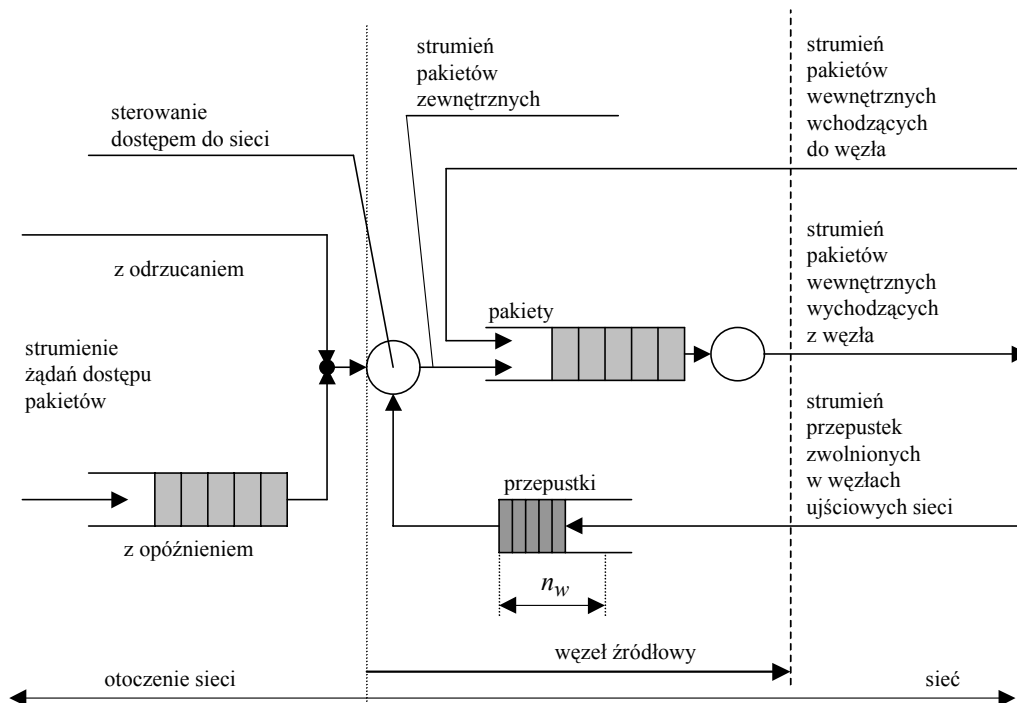
Jednym ze sposobów zapobiegania nadmiarowi pakietów jednocześnie przebywających w sieci – implementowanym w metodzie przepustek – jest wprowadzenie zasady, zgodnie z którą każdy pakiet wprowadzany do sieci musi mieć przepustkę (ang. *permit*), których liczba dla danej sieci jest ustalona i stała. Metoda izarytmiczna jest metodą globalnego sterowania przepływem na poziomie dostępu do sieci.

Celem stosowania metody izarytmicznej jest limitowanie liczby pakietów jednocześnie przebywających w sieci, tzn. takich, które zostały wprowadzone do sieci w węzłach źródłowych i których potwierdzenie odbioru w węzłach ujściowych nie zostało odebrane w węzłach źródłowych. Ograniczenie liczby pakietów zmniejsza możliwość powstania przeciążenia sieci, co jest równoznaczne ze zwiększeniem jakości obsługi pakietów wprowadzanych do sieci.

Zasady przydzielania i zwalniania przepustek, których całkowita liczba i rozmieszczenie w węzłach dostępowych do sieci decydują o jakości obsługi pakietów w sieci, w metodzie przepustek są na ogół następujące (rysunek 6.41):

➤ w chwili początkowej w sieci znajduje się określona liczba przepustek rozdzielonych pomiędzy wszystkie węzły dostępowe sieci,

- akceptacja żądania dostępu pakietu do sieci w węźle dostępowym wymaga posiadania przez pakiet przepustki na dostęp do sieci, która jest pozyskiwana z puli przepustek przechowywanych w danym węźle dostępowym,
- akceptacja żądania dostępu pakietu do sieci powoduje zmniejszenie o jeden liczby przepustek przechowywanych w węźle akceptującym żądanie dostępu,
- pakiet przyjęty do obsługi w sieci jest przekazywany od węzła źródłowego sieci (dostępowego) do węzła ujściowego sieci zgodnie z obowiązującymi w danej sieci procedurami sterowania przepływem na poziomie łącza i poziomie trasy,
- wyprowadzenie pakietu z sieci w węźle ujściowym sieci powoduje zwolnienie przepustki, która jest zwracana do puli przepustek przechowywanych w danym węźle ujściowym.



Rysunek 6.41. Zasady przydzielania i zwalniania przepustek w węzłach dostępowych do sieci

Do uzyskania efektywności metody przepustek konieczne jest, w większości przypadków, przesuwanie zwalnianych przepustek pomiędzy węzłami sieci, co ma na celu, wobec ograniczonej liczby przepustek w sieci, zapobieganie nadmiarom przepustek w jednych węzłach i brakom przepustek w innych. Brak takiego mechanizmu przesuwania przepustek może prowadzić do powstawania sytuacji, kiedy w jednych węzłach sieci znajdują się niewykorzystywane przepustki z powodu braku żądań dostępu, podczas gdy w innych węzłach żądania dostępu nowych pakietów są odrzucane z powodu braku wolnych przepustek.

W najprostszych rozwiązaniach przesuwanie przepustek jest wymuszane przez wprowadzenie ograniczenia liczby przepustek mogących przebywać w dowolnym węźle sieci. Jeżeli zwalniana przepustka nie może być przyjęta w danym węźle ujściowym (przepełnienie pamięci przepustek), to musi być odesłana do innego węzła sieci posiadającego wolne miejsce w puli przepustek zwolnionych. Do przekazywania zwolnionych przepustek pomiędzy węzłami wykorzystywane są zwykle pakiety danych i pakiety sterujące protokołu obsługującego wymianę pakietów danych w sieci; w tym przypadku przepustki są przekazywane „przy okazji” (ang. *piggyback*). Jeżeli w chwili zwolnienia przepustki i powstania konieczności jej przekazania brak jest pakietów danych zdolnych przenieść przepustkę, generowane są specjalne pakiety, których jedynym zadaniem jest przeniesienie przepustki.

Efektywność i skuteczność metody izarytmicznej w zadaniach zapobiegania przeciążeniom globalnym sieci wynikają wprost z analizy działania implementowanych w tej metodzie mechanizmów i są potwierdzane wynikami badań symulacyjnych.

Do najistotniejszych charakterystyk metody przepustek, ograniczających zakres jej stosowania, zalicza się:

1. Zależność jakości obsługi generowanych pakietów od rozkładu ruchu generowanego pomiędzy parami węzłów. Niezmienny w czasie i jednolity rozkład ruchu powoduje zwiększenie stopnia wykorzystania zasobów sieci, natomiast zmienność w czasie i niejednorodność może być źródłem pogorszenia przepustowości sieci i niesprawiedliwej alokacji zasobów.

2. Zależność efektywności wykorzystania zasobów sieci od przyjętego sposobu dystrybucji przepustek zwalnianych w węzłach ujściowych:

➤ równomierny (lub losowy) rozdział zwalnianych przepustek pomiędzy wszystkie węzły sieci może powodować konieczność oczekiwania na przepustki i tym samym ograniczać możliwości pełnego wykorzystania przepustowości sieci, zwłaszcza łączy i tras o dużej przepustowości – prowadzi to do degradacji przepustowości sieci,

➤ przekazywanie przez węzeł ujściowy zwalnianych przepustek do węzła źródłowego może prowadzić do zawłaszczania przepustek i tym samym do zawłaszczania zasobów sieci przez wybrane pary węzłów – powoduje to niesprawiedliwy podział zasobów sieci.

3. Konieczność implementacji systemu zarządzania przepustkami. Utrzymywanie stałej liczby przepustek w sieci wymaga stosowania systemu zapobiegania nieautoryzowanemu generowaniu przepustek i odtwarzania straconych przepustek.

Mimo wyżej wymienionych ograniczeń zastosowań metody przepustek, istnieją przypadki dużej efektywności jej zastosowań.

Przypadek pierwszy dotyczy sieci, w których nie są implementowane procedury sterowania przepływem na poziomie łącza (najniższy poziom sterowania przepływem w sieci), tzn. w tzw. sieciach słabo sterowanych (ang. *weakly controlled networks*). Zastosowanie metody przepustek eliminuje powstawanie przeciążeń i zakleszczeń powodowanych brakiem sterowania przepływem na poziomie łącza. Metoda przepustek jest tu ekwiwalentem metod sterowania przepływem na poziomie łącza.

Przypadek drugi dotyczy sieci, w których implementowane są procedury sterowania na poziomie łącza oraz proste procedury lokalnego sterowania dostępem, tzn. w tzw. sieciach silnie sterowanych (ang. *strongly controlled networks*). Zasadność stosowania metody przepustek, jako metody globalnego sterowania dostępem, zależy od skuteczności procedur lokalnego sterowania dostępem.

Procedury lokalnego sterowania dostępem skutecznie zapobiegają powstawaniu przeciążeń w węzłach sieci wtedy, gdy w sieci są stosowane stałe trasy przepływu pakietów. Niezmiennosc trasy pomiędzy węzłami w sieci powoduje, że ograniczenia ruchu wprowadzone na wejściu trasy mogą być i są przenoszone na wszystkie węzły trasy. W takim przypadku implementacja dodatkowego, globalnego sterowania dostępem, uzupełniającego lokalne sterowanie dostępem, prowadzi do redundancji sterowania przepływem na poziomie dostępu.

Zastosowanie metody przepustek jest uzasadnione wtedy, gdy w sieci (silnie sterowanej) są stosowane adaptacyjne reguły doboru tras. Lokalne procedury sterowania dostępem nie zapobiegają napływowi pakietów zewnętrznych powodujących przeciążenia pamięci buforowych w węzłach sieci (zwłaszcza w węzłach wewnętrznych sieci, tj. w węzłach sieci innych niż źródłowe i ujściowe). W takim przypadku metoda przepustek uzupełnia działanie procedur lokalnego sterowania dostępem.

Prostota metody przepustek, a tym samym łatwość jej implementacji w sieci, wynika z tego, że ustalanie wartości parametrów krytycznych metody sprowadza się do ustalenia maksymalnej całkowitej liczby przepustek w sieci n_p , maksymalnej liczby przepustek oczekujących w każdym z węzłów sieci n_w oraz schematu dystrybucji w sieci przepustek zwalnianych w węzłach ujściowych. Z badań symulacyjnych wynika, że maksymalną efektywność działania metody przepustek uzyskuje się dla następujących wartości parametrów krytycznych metody: $n_p = 3N$ oraz $n_w = 3$, gdzie N jest liczbą wszystkich węzłów sieci. Wzrost liczby przepustek w sieci może powodować powstawanie przeciążeń. Zwiększenie liczby przepustek oczekujących w węzłach może natomiast prowadzić do kumulacji przepustek w wybranych węzłach sieci i marnotrawienia przepustowości dostępnych z pozostałych jej węzłów, a w konsekwencji do niesprawiedliwej alokacji zasobów.

Pakiety zewnętrzne mogą być wprowadzone do sieci pod warunkiem, że uzyskają przepustkę. Intensywność napływu przepustek do węzła zależy od ich liczby w sieci, liczby węzłów oraz schematu dystrybucji przepustek zwalnianych w węzłach ujściowych sieci.

Intensywność napływu pakietów do sieci w węźle źródłowym jest równa:

➤ intensywności napływu przepustek; jeżeli intensywność żądań dostępu pakietów jest większa od intensywności napływu przepustek, to pakiety żądające dostępu i nieakceptowane z powodu braku przepustek są odrzucane (w modelu z odrzuceniami) lub oczekują w kolejce na przepustki (w modelu z opóźnieniami),

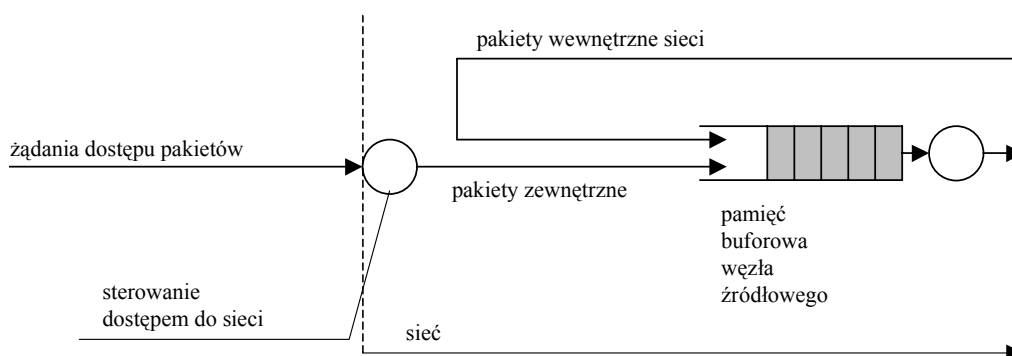
➤ intensywności żądań dostępu pakietów; jeżeli intensywność żądań dostępu pakietów jest mniejsza od intensywności napływu przepustek, to przepustki oczekują na żądania dostępu pakietów.

6.4.4.3. METODA BUFORA WEJŚCIOWEGO

Metoda bufora wejściowego (ograniczonej liczby buforów wejściowych), w przeciwieństwie do metody izarytmicznej, będącej metodą globalnego sterowania przepływem na poziomie dostępu do sieci, jest lokalną procedurą sterowania przepływem na poziomie dostępu do sieci. Stosowanie metody ograniczonej liczby buforów wejściowych, tzn. implementowanie we wszystkich węzłach źródłowych sieci działających jednocześnie lokalnych procedur sterowania, jako metody zapobiegania przeciążeniom globalnym w sieci, opiera się na założeniu (w wielu przypadkach prawdziwym), że:

- istnieje jednoznaczna zależność pomiędzy przeciążeniem węzłów źródłowych sieci i przeciążeniem wszystkich węzłów sieci, tj. przeciążeniem globalnym sieci,
- zapobieganie przeciążeniom węzłów źródłowych sieci (przeciążeniom na krańcach sieci) jest sposobem na zapobieganie przeciążeniu globalnemu sieci.

Sposób działania i nazwa omawianej metody sterowania przepływem na poziomie dostępu do sieci wywodzą się z ogólnej zasady, zgodnie z którą warunkiem przyjmowania nowych pakietów (pakietów zewnętrznych) do sieci jest dostępność buforów pamięci buforowej węzła źródłowego rezerwowanych w tym celu.



Rysunek 6.42. Zasada obsługi pakietów w metodzie bufora wejściowego

Z punktu widzenia omawianej metody sterowania dostępem do sieci wszystkie pakiety napływające do danego węzła źródłowego sieci należą do jednej z dwóch klas pakietów: pakietów wewnętrznych i pakietów zewnętrznych. Źródłem pakietów wewnętrznych w danym węźle są wszystkie pozostałe węzły sieci. Dla pakietów wewnętrznych dany węzeł jest węzłem ujściowym lub pośrednim na trasie do węzła ujściowego. Strumień pakietów zewnętrznych w danym węźle jest strumieniem zaakceptowanych żądań dostępu pakietów do sieci. Dla pakietów zewnętrznych dany węzeł jest węzłem źródłowym. Żądania dostępu pakietów do sieci są akceptowane lub odrzucone w zależności od stopnia zajętości pamięci buforowej węzła źródłowego. Jeżeli stopień zajętości pamięci buforowej węzła źródłowego jest wyższy od

założonego, to żądania dostępu pakietów są odrzucane; w innym przypadku – są akceptowane.

Omawiana zasada działania opiera się na intuicyjnie poprawnym oraz zweryfikowanym w badaniach analitycznych i symulacyjnych założeniu, że priorytet obsługi pakietów „starszych” (do obsługi których już wykorzystano zasoby sieci) powinien być wyższy od priorytetu obsługi pakietów „młodszych” (w których obsługę nie były dotychczas angażowane żadne zasoby sieci) (rysunek 6.42).

Znanych jest wiele różnych implementacji i propozycji metody ograniczonej liczby buforów wejściowych w zadaniach sterowania przepływem na poziomie dostępu do sieci.

6.4.4.4. IMPLEMENTACJA METODY BUFORA WEJŚCIOWEGO

Napływające do danego węzła źródłowego pakiety wewnętrzne należą do różnych klas Cl_i ($i = 1, 2, \dots, L$). Napływający pakiet wewnętrzny należy do i -tej klasy, jeżeli przed nadejściem do danego węzła był wcześniej obsługiwany w i węzłach, a L jest liczbą wszystkich klas równą długości najdłuższej trasy w danej sieci.

Pakiety zewnętrzne napływające do danego węzła źródłowego zalicza się do klasy zerowej Cl_0 ($i = 0$); mogą one zajmować tylko bufor należący do klasy zerowej buforów (bufory należące do zbioru K_0).

Zasady akceptacji żądań dostępu pakietów do sieci w węzle źródłowym są determinowane zasadami dostępu do pamięci buforowej pakietów należących do różnych klas. Sformułowano wiele różnych procedur akceptacji żądań dostępu pakietów wewnętrznych do pojemności pamięci buforowej, decydujących o właściwościach mechanizmów sterowania dostępem korzystającym z tych procedur. Zasady działania niektórych z procedur, tj. procedur GMDNET, Lama, Price'a oraz Kamouna, przedstawiono na rysunku 6.43.

Cechą charakterystyczną wszystkich wcześniej omawianych procedur sterowania dostępem do sieci realizowanych w węzłach źródłowych sieci jest faworyzowanie pakietów tranzytowych kosztem pakietów wejściowych. Wyższy priorytet obsługi dla pakietów tranzytowych jest uzasadniony tym, że z każdym pakietem tranzytowym związane są zasoby sieci angażowane w jego obsługę przed nadejściem do danego węzła. Faworyzowanie pakietów tranzytowych jest sposobem realizacji zasady ochrony wcześniej poczynionych inwestycji oraz źródłem wzrostu efektywności sieci (zwłaszcza zwiększenia przepustowości sieci), ale w niektórych przypadkach może prowadzić do niesprawiedliwego rozdziału zasobów pomiędzy strumienie pakietów współzawodniczące o zasoby sieci.

Procedura GMDNET

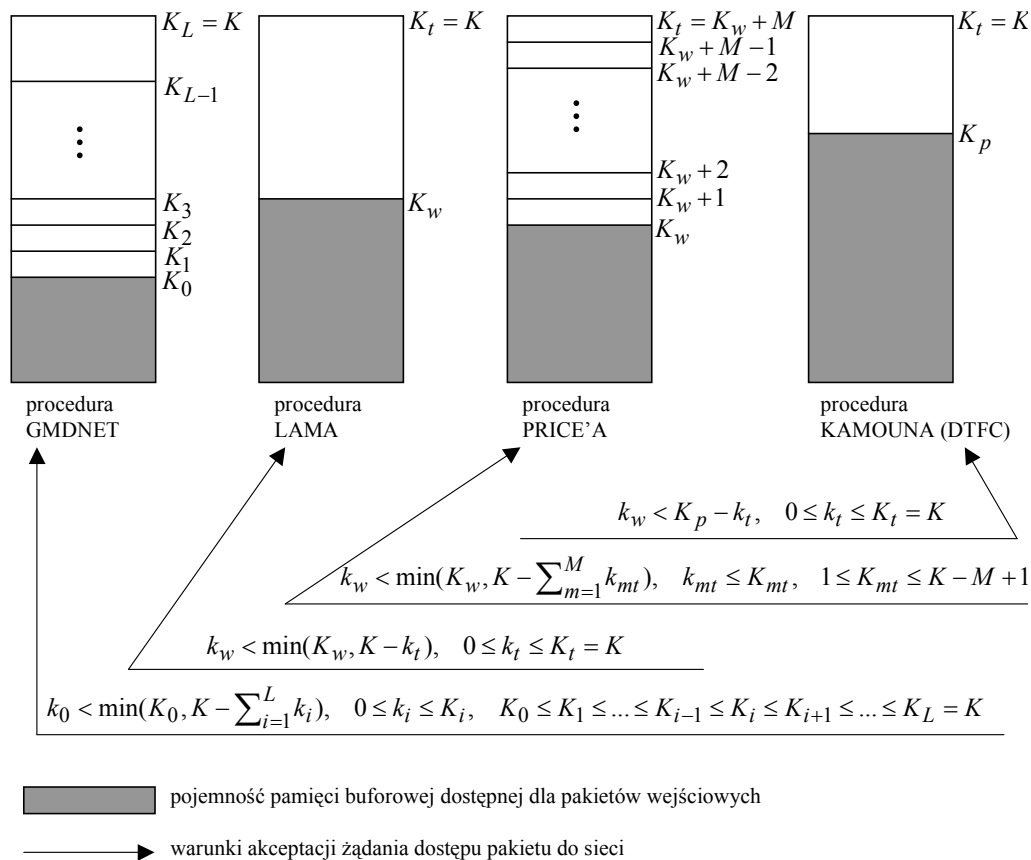
W procedurze GMDNET żądanie dostępu pakietu do sieci zostanie zaakceptowane wtedy, gdy:

➤ bufor należący do zbioru buforów K_0 nie są zajęte przez pakiety należące do klas wyższych (zbiór K_0 jest podzbiorem zbiorów K_i dla $i = 1, 2, \dots, L$),

➤ w zbiorze buforów K_0 jest co najmniej jeden wolny bufor, tzn. jeżeli spełniony jest warunek:

$$k_0 < \min(K_0, K - (k_1 + k_2 + \dots + k_L)),$$

gdzie k_i ($k_i \leq K_i$, $K_1 < K_2 < \dots < K_{i-1} < K_i < K_{i+1} < \dots < K_L = K$ dla $i = 1, 2, \dots, L$) jest liczbą pakietów i -tej klasy w pamięci buforowej węzła źródłowego; w innym przypadku żądanie dostępu pakietu do sieci zostanie odrzucone.



Rysunek 6.43. Zasady akceptacji żądań dostępu pakietów do sieci w węzle źródłowym

Bezwzględna liczba buforów K_0 należących do zbioru K_0 w pamięci buforowej węzła źródłowego, tzn. liczba buforów, które mogą być zajmowane przez pakiety klasy CL_0 (tj. pakiety zewnętrzne) oraz względny udział liczby buforów K_0 zbioru K_0 w liczbie wszystkich buforów pamięci buforowej węzła K , tzn. K_0/K są parametrami, których wartości mają decydujący wpływ na jakość działania sieci (np. przepustowość sieci) przy dużym jej obciążeniu.

Wyniki badań symulacyjnych dowodzą, że dla danej struktury topologicznej sieci i danych charakterystyk strumieni żądań dostępu do sieci istnieją optymalne wartości wymienionych parametrów maksymalizujące przepustowość sieci.

Procedura Lama

W procedurze Lama definiuje się dwie klasy pakietów w węzłach źródłowych sieci: klasę pakietów tranzytowych i klasę pakietów wejściowych. Odpowiednikami klasy pakietów tranzytowych w sieci GMDNET są pakiety wewnętrzne (pakiety wszystkich klas Cl_i ($i = 1, 2, \dots, L$)), a klasie pakietów wejściowych odpowiada klasa pakietów zewnętrznych (pakiety klasy Cl_0).

Na potrzeby sterowania dostępem przez ograniczenie liczby buforów wejściowych wyznaczana jest maksymalna liczba buforów pamięci buforowej K_w (które mogą być zajmowane przez pakiety wejściowe), spełniająca warunek:

$$K_w \leq K.$$

Pakiety tranzytowe mogą zajmować wszystkie bufony pamięci buforowej węzła, tj.

$$K_t = K,$$

gdzie:

K – całkowita liczba buforów w pamięci buforowej węzła,

K_t – maksymalna liczba buforów dostępnych dla pakietów tranzytowych.

Żądanie dostępu pakietu do sieci będzie zaakceptowane w węźle źródłowym sieci, gdy spełniony jest warunek:

$$k_w < \min(K_w, K - k_t),$$

gdzie: k_w i k_t są – odpowiednio – liczbami buforów pamięci buforowej zajętych przez pakiety wejściowe i tranzytowe: $k_w \leq K_w$ i $k_t \leq K_t = K$ (rysunek 6.43).

Ograniczenie liczby klas pakietów do dwóch powoduje ograniczenie złożoności reguły akceptacji żądań dostępu pakietów do sieci w węźle źródłowym, zgodnie z którą wszystkie bufony mogą być zajmowane przez pakiety tranzytowe, a liczba buforów zajętych przez pakiety wejściowe nie może być większa od limitu ustalonego dla pakietów wejściowych, tzn. spełnione są warunki:

$$k_t \leq K, k_w \leq K_w \text{ oraz } k_t + k_w \leq K,$$

gdzie: k_t i k_w są – odpowiednio – liczbami pakietów tranzytowych i pakietów wejściowych w pamięci buforowej (pomiędzy omawianymi metodami sterowania istnieje prosta zależność wyrażana następującymi równaniami: $k_w = k_0$, $K_w = K_0$, $k_t = k_1 + k_2 + \dots + k_L$ oraz $K_t = K_L = K$).

Uproszczenie reguły sterowania przepływem, wynikające z wprowadzenia dwóch klas pakietów oraz założeń upraszczających, dotyczących jednakowej wartości prawdopodobieństwa blokady we wszystkich węzłach sieci, umożliwia zastosowanie względnie prostego modelu sieci i uzyskanie analitycznego rozwiązania zadania optymalizacji przepustowości sieci jako funkcji ilorazu maksymalnej liczby buforów dedykowanych dla

pakietów wejściowych w pamięci buforowej i całkowitej pojemności pamięci buforowej, tj. K_w/K . Zgodnie z intuicją, potwierdzoną badaniami symulacyjnymi, optymalna wartość tego ilorazu pojemności pamięci buforowych jest równa (w przybliżeniu) wartości ilorazu przepustowości wejściowej i całkowitej węzła źródłowego. Przepustowość wejściowa i całkowita węzła są mierzone – odpowiednio – liczbą pakietów wejściowych i liczbą wszystkich pakietów obsługiwanych w węźle w jednostce czasu.

Wyniki badań omawianej metody sterowania dostępem wykazują nieczułość przepustowości węzła (a także przepustowości sieci) na względnie duże zmiany wartości ilorazu K_w/K względem optymalnej jego wartości, co oznacza, że metoda sterowania dostępem nie jest wrażliwa na zmiany charakterystyk strumieni pakietów tranzytowych i wejściowych oraz zmiany struktury topologicznej sieci.

Procedura Price'a

W procedurze sterowania przepływem na poziomie dostępu do sieci, zaproponowanej przez Price'a, podobnie jak w propozycji procedury Lama, wyróżnia się dwie klasy pakietów: wejściowych i tranzytowych, wprowadzane jest również ograniczenie liczby buforów pamięci buforowej, które mogą być zajmowane przez pakiety wejściowe.

Liczba buforów dostępnych w pamięci buforowej węzła źródłowego sieci dla pakietów wejściowych K_w spełnia następujący warunek:

$$K_w = K - M,$$

gdzie:

K – całkowita liczba buforów w pamięci buforowej węzła,

M – liczba kanałów wyjściowych (kolejek wyjściowych w pamięci buforowej) z danego węzła źródłowego.

Warunkiem akceptacji żądania dostępu do sieci w węźle źródłowym jest spełnienie następującej zależności:

$$k_w < \min(K_w, K - k_t) = \min(K_w, K - \sum_{m=1}^M k_{mt}),$$

w której:

k_w i k_t – odpowiednio – liczby pakietów wejściowych i tranzytowych w pamięci buforowej,

$k_t = k_{1t} + k_{2t} + \dots + k_{Mt}$ – liczba pakietów tranzytowych, równa sumie pakietów tranzytowych w kolejkach M kanałów wejściowych, spełniających warunki:

$$k_{mt} \leq K_{mt} = K - M + 1 \text{ dla } m = 1, 2, \dots, M,$$

K_{mt} – maksymalna liczba pakietów tranzytowych w m -tej kolejce wyjściowej węzła źródłowego (rysunek 6.43).

Procedura ta gwarantuje każdemu kierunkowi wyprowadzania pakietów tranzytowych z danego węzła jeden bufor, przez co zapobiega monopolizowaniu pamięci bu-

forowej węzła źródłowego przez pakiety wejściowe i sprawiedliwy podział pamięci buforowej (w ramach limitu równego M buforów) pomiędzy wszystkie – w danym węźle źródłowym – kanały wyjściowe. Gwarancja jednego bufora dla każdej kolejki wyjściowej węzła zapobiega powstawaniu zakleszczeń bezpośrednich.

Wyniki badań symulacyjnych dowodzą efektywności omawianej metody sterowania przepływem w sieciach jednopoziomowych.

Procedura Kamouna

Procedura Kamouna jest podobna do wcześniej omawianych propozycji Lama i Price'a. Istotna różnica między nimi polega na tym, że w tej procedurze pakiety wejściowe są przyjmowane wtedy, gdy całkowita liczba pakietów (całkowita liczba zajętych buforów) w pamięci buforowej jest mniejsza od ustalonej liczby (wartości progowej), tj. jeżeli $k_w + k_t < K_p \leq K$ (gdzie k_w i k_t są liczbami pakietów wejściowych i tranzytowych w pamięci buforowej). W procedurze Lama natomiast jednym z warunków akceptacji żądania dostępu pakietów do sieci jest nieprzekroczenie wartości maksymalnej liczby buforów dostępnych dla pakietów wejściowych, tj. $k_w \leq K_w \leq K$ (rysunek 6.43). Podobnie jak w procedurze Lama, pakiety tranzytowe mogą zajmować wszystkie bufory pamięci buforowej węzła źródłowego, tj. $k_t \leq K_t \leq K$.

Kolejną istotną różnicą pomiędzy metodą Kamouna i wcześniej omawianymi procedurami jest to, że pakiety tranzytowe, które napływają do węzła i nie mogą być przyjęte z powodu przepełnienia pamięci buforowej, są odrzucane i tracone (model ze stratami). Z przyjęcia takiego modelu obsługi pakietów tranzytowych wywodzi się nazwa metody, tzn. DTFC (ang. *Drop-and-Throttle Flow Control*).

Badania analityczne i symulacyjne metody DTFC wykazują, że dla każdej wartości obciążenia sieci istnieje optymalna wartość progowa K_p^* , maksymalizująca przepustowość sieci. Przyjęcie wartości progowej mniejszej od wartości optymalnej prowadzi do niewykorzystania (marnowania) zasobów sieci, przyjęcie natomiast wartości progowej wyższej od optymalnej powoduje przeciążenie sieci.

6.4.4.5. METODA Z PAKIETEM DŁAWIĄCYM

Idea sterowania przepływem, zastosowana w procedurze z pakietem dławiącym zaproponowanej dla sieci CYCLADES, polega na wykorzystaniu do powstrzymywania (dławienia) napływu pakietów do sieci informacji o przeciążeniach tras i poszczególnych łączy wchodzących w skład tras. Informacje o przeciążeniach są propagowane wraz z informacjami wykorzystywanymi do wyznaczania tras w sieci.

Podstawą procedury jest definicja przeciążenia trasy, zgodnie z którą trasa jest przeciążona, jeżeli co najmniej jedno łącze wchodzące w skład trasy jest przeciążone, a łącze jest definiowane jako przeciążone wtedy, gdy stopień jego wykorzystania (ang. *utilisation*) jest wyższy od ustalonej wartości progowej.

Informacja o przeciążeniu łącz jest propagowana w sieci w sposób pozwalający na utrzymywanie w każdym węźle sieci aktualnych informacji o odległościach (mierzonej liczbą łączy) i stanie przeciążenia najkrótszych tras prowadzących do każdego węzła ujściowego. Służy ona do sterowania przepływem pakietów dzielonych na wejściowe (ang. *input packets*) i tranzytowe (ang. *transit packets*) w następujący sposób:

1. Jeżeli do węzła napływa żądanie dostępu pakietu do sieci (pakiet wejściowy), kierowane do węzła ujściowego, do którego prowadzi trasa spełniająca warunki trasy przeciążonej, to nie jest ono akceptowane w węźle źródłowym.

2. Jeżeli napływający do węzła pakiet jest pakietem tranzytowym, kierowanym do węzła ujściowego, do którego prowadzi przeciążona trasa, to:

➤ pakiet jest wysyłany z danego węzła źródłowego do węzła ujściowego przeciążoną trasą,

➤ w danym węźle źródłowym generowany jest pakiet dławiący (ang. *choke packet*), którego adresatem jest węzeł źródłowy odebranego wcześniej pakietu tranzytowego. Celem generowanego pakietu jest przeniesienie informacji o przeciążeniu najkrótszej trasy prowadzącej do węzła ujściowego (wskazanego adresem ujścia pakietu tranzytowego powodującego generowanie pakietu dławiącego) i zablokowanie przeciążonej trasy.

Identyfikowane w danym węźle przeciążenie określonych tras powoduje wstrzymanie w węźle (źródłowym lub wewnętrznym) sieci akceptacji tylko tych żądań dostępu pakietów do sieci i wstrzymanie obsługi tylko tych pakietów tranzytowych, które są adresowane do węzłów ujściowych, do których prowadzą trasy zidentyfikowane jako przeciążone. Uznanie w danym węźle tylko niektórych tras za przeciążone nie powoduje wstrzymania akceptacji żądań dostępu pakietów do sieci i obsługi pakietów tranzytowych przesyłanych z danego węzła trasami innymi od tras zidentyfikowanych jako przeciążone. Oznacza to, że metoda pakietów dławiących działa selektywnie w każdym węźle sieci. Selektywność działania metody polega na tym, że w tym samym węźle sieci niektóre z żądań dostępu są akceptowane, a inne odrzucane oraz – podobnie – niektóre ze źródeł pakietów tranzytowych są blokowane, inne natomiast nie są blokowane.

Trasa prowadząca do określonego węzła ujściowego jest odblokowywana stopniowo. Sygnałem do odblokowywania trasy w węźle źródłowym jest brak napływu pakietów sterujących w określonym przedziale czasu.

Działanie metody z pakietem dławiącym opiera się na założeniu, że istnieje jednoznaczna zależność pomiędzy wartością przepływu na trasie i szybkością akceptacji żądań dostępu pakietów w węzłach źródłowych sieci. Taka jednoznaczna zależność zachodzi tylko w sieciach, w których są stosowane sztywne reguły doboru i wyboru tras. W sieciach pakietowych z adaptacyjnymi regułami doboru tras nie ma jednoznaczności odwzorowania szybkości akceptacji żądań dostępu na przepływ na trasach sieci.

Podobnie jak w omawianych wcześniej metodach sterowania dostępem do sieci, opartych na ograniczaniu pojemności pamięci buforowej dla pakietów zewnętrznych,

w metodzie z pakietem dławiącym pakiety tranzytowe są faworyzowane kosztem pakietów wejściowych.

6.4.4.6. CAŁOŚCIOWE I SELEKTYWNE STEROWANIE PRZEPLYWEM

Podstawowa różnica pomiędzy metodą bufora wejściowego a metodą z pakietem dławiącym polega na stosowaniu różnych miar przeciążenia do podejmowania decyzji o akceptacji lub odrzuceniu żądań dostępu pakietów do sieci. Przyjęcie różnych miar przeciążenia elementów sieci i wnioskowanie na tej podstawie o jej przeciążeniu powoduje, że różne są zakresy podejmowanych decyzji sterujących dotyczących różnie definiowanych przepływów.

W metodach bufora wejściowego miarą przeciążenia jest stopień zajętości pamięci buforowej węzła źródłowego. Stopień zajętości pamięci buforowych węzłów źródłowych sieci (niektórych węzłów sieci) służy do aproksymacji stopnia przeciążenia sieci (wszystkich węzłów sieci). Wzrost stopnia zajętości pamięci buforowych węzłów źródłowych jest utożsamiany ze zwiększeniem zajętości wszystkich węzłów sieci (wszystkich jej zasobów).

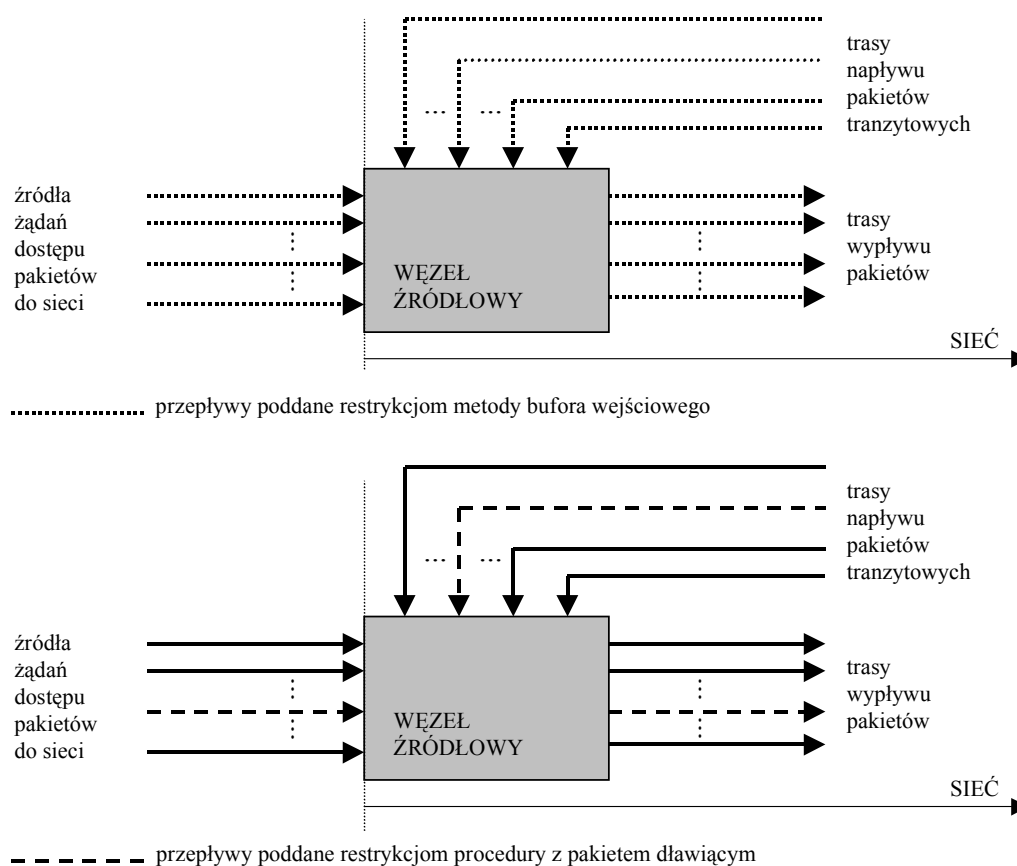
Zasadnicza koncepcja realizowana w omawianych metodach sterowania przepływem polega na przyjęciu założenia, że skuteczne zapobieganie przeciążeniom węzłów źródłowych jest tożsame ze skutecznym zapobieganiem przeciążeniom sieci. Ponieważ zajętość pamięci buforowej danego węzła jest skutkiem napływu pakietów ze wszystkich węzłów źródłowych (względem danego węzła), przedmiotem sterowania jest więc całkowity przepływ przez dany węzeł, a zapobieganie przeciążeniu polega na wprowadzaniu restrykcji dotyczących (w różnym stopniu) całkowitego przepływu przez dany węzeł, tj. wszystkich strumieni pakietów napływających i wypływających do i z danego węzła. W konsekwencji restrykcje dotyczą żądań dostępu pakietów do sieci ze wszystkich źródeł względem danego węzła oraz przepływów na wszystkich trasach wchodzących i wychodzących do i z danego węzła źródłowego (rysunek 6.44).

W metodzie z pakietem dławiącym miarą przeciążenia jest stopień wykorzystania trasy, definiowany jako równy stopniowi wykorzystania najbardziej obciążonego elementu trasy (łącza). Ponieważ stopień wykorzystania danej trasy jest skutkiem napływu pakietów kierowanych do obsługi przez daną trasę, przedmiotem sterowania jest więc przepływ na danej trasie, a zapobieganie przeciążeniu trasy polega na wprowadzaniu restrykcji dotyczących (w różnym stopniu) tylko tych strumieni pakietów, które kierowane są na daną trasę (rysunek 6.44).

Sterowanie przepływem w węźle, polegające na wprowadzaniu restrykcji dla wszystkich napływających do tego węzła strumieni pakietów, jest określane mianem całościowego (ang. *complex*). Przykładem metody całościowego sterowania przepływem jest metoda bufora wejściowego.

Sterowanie przepływem pakietów realizowanym w węźle i polegające na wprowadzaniu restrykcji tylko w odniesieniu do niektórych napływających do tego węzła

strumieni pakietów jest określane mianem sterowania wybiórczego (selektywnego) (ang. *selective*). Przykładem metody selektywnego sterowania przepływem jest metoda z pakietem dławiącym.



Rysunek 6.44. Różne zakresy restrykcji wprowadzane przez metody bufora wejściowego i metody z pakietem dławiącym

Możliwość stosowania całościowego lub selektywnego sterowania przepływem wynika z definicji przeciążenia sieci i jej elementów.

6.4.5. STEROWANIE PRZEPLYWEM NA POZIOMIE TRANSPORTOWYM

6.4.5.1. CEL STEROWANIA PRZEPLYWEM NA POZIOMIE TRANSPORTOWYM

Sterowanie przepływem na poziomie transportowym polega – podobnie jak w przypadku sterowania przepływem na poziomach łącza, trasy i dostępu do sieci – na

implementacji określonych mechanizmów w określonym protokole, w tym przypadku w protokole transportowym, którego zadaniem jest zarządzanie transferem danych oraz informacji sterujących pomiędzy komunikującymi się poprzez sieć pojedynczymi procesami użytkowników.

Podstawowe zadania protokołu transportowego są następujące:

1. Zapewnienie efektywnego i niezawodnego transferu wiadomości w ramach każdej sesji użytkowników sieci. Realizacja tak formułowanego celu sterowania dotyczy m.in. zadań:

- pakietyzacji wiadomości (ang. *packetization*),
- odtwarzania kolejności pakietów (ang. *resequencing*),
- odtwarzania utraconych pakietów (ang. *recovery from loss*),
- eliminacji duplikatów (ang. *elimination of duplicates*),
- odtwarzania wiadomości (ang. *reassembly*).

2. Zapewnienie efektywnego wykorzystania współdzielonych zasobów sieciowych. Realizacja tak formułowanego zadania wiąże się ze zwielokrotnianiem jednocześnie obsługiwanych sesji użytkowników w kanałach transmisyjnych i utrzymywaniem systemu priorytetów odzwierciedlającego hierarchię ważności różnych sesji użytkowników.

Z efektywnym i niezawodnym transferem segmentu, tzn. jednostki danych protokołu transportowego, w postaci sekwencji pakietów łączy się przede wszystkim konieczność zapewnienia odpowiedniej pojemności pamięci buforowej w węźle ujściowym (w DTE węzła ujściowego), niezbędnej do przyjęcia wszystkich pakietów segmentu i jego odtworzenia.

Jedną z podstawowych funkcji każdego protokołu transportowego jest sterowanie przepływem na poziomie transportowym (ang. *transport level flow control*), którego celem jest zapobieganie przeciążeniom (ang. *congestion*) i przepełnieniom (ang. *overflow*) pamięci buforowej węzła ujściowego przy jednoczesnym zagwarantowaniu dostępu do pamięci buforowej w ilościach niezbędnych do odtwarzania transferowanych segmentów.

Do osiągnięcia celu sterowania przepływem na poziomie transportowym najczęściej stosuje się metodę krokową (ang. *pacing method*), zwaną także metodą kroczącego okna (ang. *pacing window*), tzn. mechanizm łączący w sobie cechy metody kredytowania i metody okna.

Ogólna zasada metody krokowej polega na tym, że odbiorca, przekazując nadawcy uprawnienia do transmisji (kredyt), informuje o możliwości przyjęcia określonej liczby wiadomości do pamięci buforowej. Warunkiem przekazania kredytu przez odbiorcę jest dostępność pamięci buforowej odtwarzania wiadomości (ang. *reassembly buffers*), tzn. dostępność buforów przeznaczonych do przyjmowania pakietów, z których w węźle ujściowym są odtwarzane wiadomości wyprowadzane z sieci. Otrzymanie kredytu przez nadawcę jest równoznaczne z uzyskaniem autoryzacji (pozwolenia) na transmitowanie wiadomości o ustalonej długości, mierzonej liczbą pakietów. Wysłanie kredytu jest wstrzymywane, gdy pamięć buforowa odtwarzania wiadomości jest zajęta. Wysłanie kredytu jest wznowiane po zwolnieniu pamięci buforowej odtwarzania wiadomości, tj. po wyprowadzeniu wiadomości z sieci.

Skuteczność i efektywność sterowania przepływem metodą krokową zależy od niezawodności używanego w protokole transportowym sposobu przekazywania kredytu, tj. od zabezpieczeń przed stratą kredytu. Strata kredytu może spowodować zawieszenie połączenia (sesji), polegające na tym, że nadawca, nie mając kredytu, nie może transmitować wiadomości do oczekującego na wiadomość odbiorcy. Jednym ze sposobów zabezpieczenia przed utratą kredytu jest numerowanie, polegające na tym, że kolejno wysyłane przez odbiorcę kredyty zawierają kolejne numery wiadomości N oraz rozmiar okna w . Otrzymanie takiego kredytu przez nadawcę jest potwierdzeniem odbioru wiadomości o numerach mniejszych od N i upoważnieniem nadawcy do wysłania wszystkich wiadomości o numerach wskazanych w kredycie, tj. aż do $N + w$ wiadomości włącznie. W razie utraty jednego z wysyłanych przez odbiorcę wiadomości kredytu nie następuje zawieszenie połączenia; kolejny wysłany kredyt pozwala bowiem nadawcy wiadomości odtworzyć informację o stanie sesji i numerach potwierdzonych i oczekiwanych przez odbiorcę wiadomości.

Skutki działania procedur sterowania przepływem na poziomie transportowym z wykorzystaniem mechanizmu kredytowania są dwojakie: bezpośrednie i pośrednie. Bezpośrednio zapobiegają przeciążeniom pamięci buforowej węzłów ujściowych, tj. przeciążeniom lokalnym, pośrednio natomiast zapobiegają przeciążeniom w węzłach leżących na trasach z węzłów źródłowych do węzłów ujściowych, tj. przeciążeniom globalnym. Działanie pośrednie polega na tym, że czas przekazania kredytu od odbiorcy do nadawcy zależy od opóźnienia wnoszonego przez sieć. Po zwiększeniu obciążenia sieci rośnie wartość średniego opóźnienia wnoszonego przez sieć, powodując wzrost średniego czasu przekazywania kredytu i – w konsekwencji – zmniejszenie szybkości wprowadzania nowych wiadomości do sieci.

6.4.5.2. PRZYKŁADOWE IMPLEMENTACJE STEROWANIA PRZEPLYWEM NA POZIOMIE TRANSPORTOWYM

W różnych architekturach i rozwiązaniach sieciowych stosuje się różne odmiany sterowania przepływem na poziomie transportowym.

O wyborze różnych procedur i (lub) różnych wartości parametrów tych samych procedur sterowania na poziomie transportowym decyduje wiele różnych czynników; do najważniejszych można zaliczyć: ilość i jakość procedur sterowania przepływem na poziomach niższych (łącza, trasy i dostępu do sieci) wspierających nadrzędne wobec nich procedury sterowania na poziomie transportowym oraz zakładaną w danej sieci redundancję implementowanej hierarchii procedur sterowania przepływem.

6.4.5.2.1. PROCEDURA STEROWANIA PRZEPLYWEM NA POZIOMIE TRANSPORTOWYM SIECI ARPANET

W implementowanym w pierwszych wersjach sieci ARPANET protokole transportowym NCP (ang. *Network Control Program*) stosowany jest bardzo prosty schemat nienumerowanego kredytowania, którego zadaniem jest zapobieganie powstawa-

niu zakleszczeń odtwarzania wiadomości. Schemat kredytowania jest wbudowany w mechanizm rezerwacji (alokacji) liczby buforów w pamięci buforowej węzła ujściowego, niezbędnych do skompletowania i odtworzenia pojedynczych wielopakietowych wiadomości. Nadawanie wielopakietowej wiadomości można rozpocząć po uzyskaniu z węzła ujściowego potwierdzenia (ALL) wysłanego wcześniej żądania alokacji buforów (REQALL). Warunkiem rozpoczęcia nadawania wiadomości jest potwierdzona rezerwacja odpowiedniej liczby buforów w węzle ujściowym.

Charakterystyczne dla omawianej implementacji jest to, że procedura żądania alokacji i potwierdzania przyznania zasobów w odpowiedzi na żądanie jest uruchamiana dla każdej nadawanej wiadomości indywidualnie. Taka procedura rezerwacji buforów powoduje, że otrzymane potwierdzenie (kredyt) dotyczy tylko jednej wiadomości ($w = 1$), co eliminuje celowość numerowania nadawanych wiadomości.

Korzystanie z nienumerowanego schematu kredytowania wywołuje potrzebę implementacji mechanizmu zapobiegającego konsekwencjom utraty w sieci potwierdzenia rezerwacji buforów.

6.4.5.2.2. PROCEDURA STEROWANIA PRZEPLYWEM NA POZIOMIE SIECI CYCLADES

W stosowanym w sieci CYCLADES protokole transportowym TS (ang. *Transport Station*) korzysta się z numerowanego schematu pojedynczego kredytowania wiadomości. Każdy otrzymany kredyt upoważnia nadawcę do transferu pojedynczej wiadomości (wiadomości mogą mieć różne długości), zwanej listem (ang. *letter*). W omawianym protokole funkcje sterowania przepływem są łączone z funkcjami detekcji i korekcji błędów (ang. *error control*). Kredyty są przekazywane w potwierdzeniach poprawnego odbioru wiadomości.

6.4.5.2.3. PROCEDURA STEROWANIA PRZEPLYWEM NA POZIOMIE TRANSPORTOWYM W ARCHITEKTURZE ISO/OSI

W architekturze ISO/OSI warstwa transportowa pełni kluczową rolę w organizacji współpracy różnych protokołów warstw wyższych i niższych. W standardach protokołów transportowych zdefiniowane są zarówno połączeniowe, jak i bezpołączeniowe usługi transportowe. Implementacja usług transportowych w wersji połączeniowej oznacza możliwość zastosowania zarówno połączeniowych, jak i bezpołączeniowych wersji usług warstw niższych.

Ze względu na to, że różne typy usług wymagają zróżnicowanego poziomu niezawodności dostarczanego przez rozmaite rozwiązania transportowe, zdefiniowanych jest pięć klas protokołów warstwy transportowej.

Jednym z wymagań stawianych połączeniowym protokołom transportowym jest sterowanie przepływem pomiędzy procesami końcowymi (ang. *end-to-end flow control*).

W połączeniowych protokołach transportowych architektury OSI implementowany jest schemat numerowanego kredytowania, który wykorzystuje do przekazywania kredytów numerowanie sekwencyjne wiadomości, używane jednocześnie do korekcji błędów – przekazywania potwierdzeń poprawnego odbioru wiadomości.

Początkowo, w czasie ustanawiania połączenia obejmującego m.in. negocjacje wartości parametrów połączenia, nadawca otrzymuje kredyt o wartości numeru sekwencyjnego równej rozmiarowi okna (parametr protokołu). Rozmiar okna determinuje liczbę dopuszczalnych transmisji. Żadna wiadomość o numerze sekwencyjnym wyższym od numeru wskazywanego przez strzałkę prowadzącą (ang. *leading edge*) okna nie może być wysłana przez nadawcę. Strzałka końcowa (ang. *trailing edge*) okna po wysłaniu kolejnej wiadomości jest każdorazowo przesuwana, co powoduje zwięzanie okna. Kontynuacja transmisji wymaga przekazania nowego kredytu lub pozwolenia na przesunięcie strzałki prowadzącej okna przez odbiorcę. Wielkość (rozmiar) otrzymanego kredytu decyduje o tym, jak daleko do przodu nadawca może przesunąć strzałkę prowadzącą okna. Z kolei odbiorca może wstrzymać wysyłanie kolejnych kredytów wtedy, gdy stan zajętości pamięci buforowej jest wyższy od przyjętej wartości progowej. Jednoczesne nałożenie się strzałki końcowej na strzałkę prowadzącą oraz brak nowego kredytu powoduje, że nadawca, nie mogąc przesunąć strzałki prowadzącej, jest zmuszony do wstrzymania procesu wysyłania kolejnych wiadomości.

Przekazywanie potwierdzeń odbioru wiadomości i przekazywanie kredytów przez odbiorcę jest niezależne. Przekazanie kredytu nie oznacza przekazania nowego potwierdzenia odbioru, podobnie jak przekazanie potwierdzenia odbioru nie oznacza przekazania nowego kredytu.

6.4.5.2.4. PROCEDURA STEROWANIA PRZEPLYWEM NA POZIOMIE TRANSPORTOWYM W ARCHITEKTURZE TCP/IP

Głównym zadaniem protokołu transportowego TCP (ang. *Transmission Control Protocol*), odpowiadającego protokołowi klasy 4. architektury ISO/OSI, jest niezawodne przekazywanie strumienia danych pomiędzy procesami końcowymi. Jakikolwiek przepływ danych poprzedza ustanowienie połączenia transportowego, polegającego na trzyetapowym (ang. *three-way handshake*) potwierdzeniu gotowości stron do współpracy i uzyskaniu zgody na początkowe wartości pary numerów sekwencyjnych, stosowanych następnie do potwierdzeń odbioru i sterowania przepływem.

TCP otrzymuje strumień bajtów, z których składane są pakiety zawierające dane i nagłówki protokołu. Każdy pakiet jest przekazywany w sieci w trybie datagramowym. Za kolejność i niezawodność przekazywanych w sieci datagramów odpowiada protokół TCP.

Informacje umieszczone w zawartych w nagłówku segmentu TCP polach: numeru sekwencyjnego (ang. *sequence number*), numeru potwierdzenia (ang. *acknowledge number*) i okna (ang. *window*) są stosowane do – opartego na metodzie przesuwanego okna (ang. *sliding window*) – sterowania potwierdzeniami i przepływem.

Numery sekwencyjne, przypisywane bajtom, są przeznaczone do nadzorowania kolejności napływu danych i zagwarantowania potwierdzenia każdego elementu strumienia danych. Numer sekwencyjny w nagłówku danego segmentu TCP wskazuje pozycję w strumieniu bajtów danych w pakiecie, numer potwierdzenia wskazuje na-

tomiast najstarszy bajt, który został odebrany przez źródło tego pakietu. Numer sekwencyjny odnosi się więc do strumienia danych płynących w tym samym kierunku, w którym przekazywany jest pakiet niosący ten numer. Z kolei numer potwierdzenia odnosi się do strumienia danych płynących w kierunku przeciwnym do kierunku, w którym przekazywany jest pakiet niosący ten numer potwierdzenia.

W protokole TCP na potrzeby efektywnego sterowania przepływem dopuszczalne są różne (zmiennie) rozmiary okna. O rozmiarze okna decyduje odbiorca. W każdym potwierdzeniu, określającym numery odebranych bajtów, przekazywany jest wskaźnik rozmiaru okna (ang. *window-size-indicator*). Wartość wskaźnika określa na przyjęcie ilu dodatkowych (względem potwierdzanego) bajtów przygotowany jest odbiorca. Wzrost wartości wskaźników przeciążenia, tzn. stopnia zajętości pamięci buforowej w węzle ujściowym i średniego czasu obsługi bajtów, mierzonego opóźnieniem potwierdzenia bajtu (ang. *round-trip delay*), powoduje zmniejszanie przez odbiorcę wartości tego wskaźnika (rozmiaru okna). Rozmiar okna może także zmniejszyć nadawca.

6.4.5.2.5. PROCEDURA STEROWANIA PRZEPLYWEM NA POZIOMIE TRANSPORTOWYM W ARCHITEKTURZE SNA

W klasycznej architekturze SNA, dotyczącej rozwiązań o strukturze scentralizowanej, sterowanie przepływem pomiędzy procesami końcowymi na poziomie transportowym jest realizowane w warstwie sterowania transmisją (ang. *transmission control*), położoną pomiędzy warstwami sterowania przepływem danych (ang. *data flow control*) i sterowania przepływem na trasie (ang. *path flow control*). Sterowanie przepływem na tym poziomie dotyczy każdej sesji z osobna.

Celem sterowania przepływem w warstwie sterowania transmisją w architekturze SNA, realizowanym metodą krokową na poziomie sesji (ang. *session level pacing*), jest dostosowanie tempa nadawania do tempa odbioru danych w ramach poszczególnych sesji, tzn. zapobieganie wysyłania danych przez proces źródłowy sesji szybciej niż mogą one być przetwarzane przez proces ujściowy sesji.

Do sterowania przepływem w warstwie sterowania transmisją stosowana jest metoda krokowa, używana także do sterowania przepływem w warstwach niższych.

W metodzie krokowej SNA stosuje się ogólne koncepcje zasady kredytowania i metody okna. Proces ujściowy sesji potwierdza kredyty dla procesu źródłowego sesji w zależności od dostępności buforów i możliwości przetworzenia odbieranych danych. W omawianej metodzie otrzymanie kredytu przez nadawcę jest równoznaczne z otrzymaniem informacji o rozmiarze okna, wcześniej wynegocjowanego przez nadawcę i odbiorcę. Rozmiar okna określa liczbę wiadomości (lub maksymalną liczbę wiadomości), które mogą być wysłane przez nadawcę przed uzgodnieniem i uzyskaniem kolejnego okna (kredytu). Rozmiary okien mogą być różne; ich długość zależy od aktualnego obciążenia sieci. Przyznanie okna przez odbiorcę jest przekazywane w postaci potwierdzenia wysyłanego w odpowiedzi na żądanie kredytu przez nadawcę, tj. żądanie przyznania następnego okna.

Ogólna zasada żądania i przyznawania kredytów (okien) jest następująca:

- nadawca wysyła pierwszą wiadomość okna pierwszego i żąda przyznania okna drugiego,
- odbiorca potwierdza przyznanie okna drugiego,
- nadawca wysyła pozostałe wiadomości okna pierwszego,
- nadawca wysyła pierwszą wiadomość okna drugiego i żąda przyznania okna trzeciego,
- odbiorca potwierdza przyznanie okna trzeciego,
- nadawca wysyła pozostałe wiadomości okna drugiego itd.

Żądanie następnego okna jest generowane w chwili rozpoczęcia wysyłania wiadomości aktualnie obsługiwanego okna. Celem takiego rozwiązania jest eliminacja przerw w nadawaniu wiadomości, jakie występowałyby, gdyby żądania były wysyłane w chwili zakończenia obsługi wszystkich wiadomości w oknie. Nadawca może mieć co najwyżej dwa okna.

W architekturze SNA proces przekazywania kredytów (ang. *pacing response*) jest realizowany na dwóch poziomach: sesji oraz dostępu do sieci. W węzłach sieci SNA (odgrywających rolę węzłów koncentracji ruchu generowanego na poziomie niższym sieci i węzłów dostępowych z poziomu niższego do poziomu wyższego sieci) istnieje możliwość sterowania przepływem z poziomu niższego sieci przez wstrzymywanie kredytów dla danej sesji, jeżeli pamięć buforowa węzła jest przeciążona lub jeżeli kolejka trasy wirtualnej dla danej sesji jest przeciążona. Możliwość przechwytywania kredytów w węzłach sieci SNA (w węzłach dostępowych do poziomu wyższego sieci) pozwala na segmentację sesji oraz jest sposobem sterowania dostępem do poziomu wyższego sieci ruchu generowanego na poziomie niższym sieci (ruchu generowanego przez terminale). Tak realizowana metoda krokowa na poziomie sesji może być interpretowana jako sterowanie przepływem w dwóch połączonych kaskadowo segmentach: segmencie dostępu do sieci (z poziomu niższego do poziomu wyższego) oraz segmencie odpowiadającym trasie wirtualnej pomiędzy węzłem źródłowym i węzłem ujściowym (na poziomie wyższym sieci). Sterowanie przepływem w drugim z wymienionych segmentów dotyczy przepływu na trasie wirtualnej (sterowanie przepływem na poziomie trasy) i jest realizowane z wykorzystaniem metody krokowej.

6.4.5.2.6. PROCEDURA STEROWANIA PRZEPLYWEM NA POZIOMIE TRANSPORTOWYM

W ARCHITEKTURZE APPN

W architekturze APPN (ang. *Advanced Peer-to-Peer Networking*), będącej rozwinięciem SNA i pozwalającej na dynamiczne definiowanie zasobów, uaktualnianie struktury topologicznej, rekonfiguracje w razie uszkodzeń oraz dynamiczne reguły doboru tras do sterowania przepływem w warstwie sterowania transmisją stosowana jest adaptacyjna metoda krokowa (ang. *adaptive pacing*).

Ogólna zasada działania adaptacyjnej metody krokowej jest taka sama jak klasycznej metody krokowej, z tą jednak różnicą, że w metodzie adaptacyjnej dopusz-

czalne są różne rozmiary okien oraz możliwa jest zmiana już przyznaných rozmiarów okien.

Adaptacyjna metoda krokowa jest stosowana indywidualnie do każdej sesji z osobna. Indywidualne sterowanie przepływem dla każdej z sesji uważane jest za korzystniejsze od rozwiązań, w których przepływy wielu jednocześnie obsługiwanych sesji mogą wzajemnie na siebie wpływać.

Adaptacyjna metoda krokowa zapewnia węzłowi (obsługującemu wiele sesji jednocześnie) możliwość dynamicznej alokacji zasobów do poszczególnych sesji o różnej intensywności generowania ruchu oraz odzyskiwanie zasobów od nieaktywnych w danej chwili sesji. Celem takiego sterowania przepływem jest:

- zapobieganie przeciążeniom pamięci buforowych węzłów pośrednich (wewnętrznych) i ujściowych,
- eliminacja niepotrzebnego odrzucania wiadomości,
- eliminacja możliwości powstawania zakleszczeń.

Adaptacyjna metoda krokowa stosowana jest niezależnie dla każdego kierunku przepływu i każdego połączenia międzywęzłowego na trasie sesji. Obsługa żądania wydania pozwolenia na transmisję polega na zarezerwowaniu buforów i wysłaniu potwierdzenia, tzn. informacji o rozmiarze przyznanego okna. W ten sposób każdy węzeł odbiorczy na trasie sesji informuje każdy węzeł nadawczy o możliwości wysłania określonej liczby wiadomości. W celu zagwarantowania możliwości odbioru wiadomości, w węzłach odbiorczych rezerwowana jest pamięć buforowa w ilości pozwalającej na odebranie wiadomości o maksymalnych, dopuszczalnych w sieci rozmiarach.

Rozpoczęcie sterowania przepływem, zgodnie z omawianą metodą, wymaga uzgodnienia trybu pracy, tj. ustalenia, czy używane będą zmienne czy też stałe rozmiary okna, oraz wyznaczenia jego początkowego rozmiaru. Wersja ze stałym rozmiarem okna używana jest wtedy, gdy obydwa procesy końcowe sesji żądają tego trybu pracy; w pozostałych przypadkach stosowana jest wersja ze zmiennym rozmiarem okna. Uzgodnienia, o których mowa, odbywają się przez wymianę komend zawierających informacje o żądanym trybie pracy oraz rozmiarach dwóch pierwszych okien.

Rozmiar okna wyznaczany przez odbiorcę zależy od stopnia wykorzystania wcześniej przyznaných okien, liczby dostępnych buforów oraz wartości przyjętych wskaźników przeciążenia węzła ujściowego (jednym ze wskaźników przeciążenia, przykładowym, jest liczba odebranych i nieprzetworzonych wiadomości). Na rozmiar okna wpływ ma także nadawca – gdy długość kolejki wejściowej przekroczy ustaloną wartość progową, wówczas nadawca może żądać przyznania większego okna. Możliwość dynamicznego dostosowywania rozmiarów okien do aktualnego stanu sieci i intensywności generowanego ruchu, tzn. możliwość dynamicznej dystrybucji pamięci buforowej pomiędzy sesje, pozwala na wzrost stopnia wykorzystania pamięci buforowej.

W omawianej metodzie możliwe są arbitralne, dowolne zmiany (zmniejszanie) rozmiarów okien przyznaných nadawcy przez odbiorcę. W celu przyśpieszenia reak-

cji, w razie powstania przeciążenia, możliwe jest zarówno żądanie (przez odbiorcę) rezygnacji nadawcy z niewykorzystanej części aktualnie obsługiwanego okna, jak i zmiana rozmiaru wcześniej przyznanego, następnego okna.

Wynikiem sterowania przepływem w ramach poszczególnych sesji oraz poszczególnych połączeń międzywęzłowych trasy sesji adaptacyjną metodą krokową jest globalne sterowanie przepływem. Redukcja rozmiarów okien w węźle ujściowym powoduje, na skutek występowania efektu propagacji wstecznej (ang. *backpressure effect*), wzrost stopnia zajętości pamięci buforowych w węzłach pośrednich. Wzrost ten, będący sygnałem niebezpieczeństwa powstania przeciążenia węzła pośredniego, skutkuje ograniczeniem rozmiaru okien. Kontynuacja tego procesu prowadzi w konsekwencji do ograniczania napływu nowych wiadomości do sieci.

W razie przekroczenia wartości progowych stopnia zajętości pamięci buforowych stosowana jest procedura dławienia strumienia napływających wiadomości. W procedurze tej, zamiast penalizacji nowych sesji, wprowadzane są redukcje rozmiarów najdłuższych dotychczas przyznanых okien. Redukcja ta polega na tym, że po przekroczeniu omawianej wartości progowej zajętości pamięci, wyszukiwana jest sesja, której okno jest największe. Rozmiar takiego okna jest zmniejszany o jeden. Proces ten jest kontynuowany aż do chwili, gdy zajętość pamięci buforowej będzie mniejsza od ustalonej wartości progowej.

6.4.6. PODSUMOWANIE

6.4.6.1. HIERARCHICZNA STRUKTURA PROCEDUR STEROWANIA PRZEPLYWEM W SIECIACH

Wielopoziomowa hierarchiczna struktura zagnieżdżanych procedur sterowania przepływem odpowiada ogólnej zasadzie budowy warstwowych architektur systemów i sieci komputerowych, zwłaszcza podsystemów komunikacyjnych systemów i sieci komputerowych. Odpowiedniość taka jest jednym z warunków implementacji procedur sterowania, ze względu na to, że realizacja każdej procedury sterowania przepływem wymaga jej wbudowania w jeden lub kilka protokołów systemu lub sieci teleinformatycznej. Taka struktura procedur sterowania umożliwia dekompozycję sterowania przepływem na niezależne, wspierające się moduły oraz niezależne definiowanie, analizę i porównywanie różnych rozwiązań na tych samych poziomach sterowania przepływem.

Modularność struktury procedur umożliwia elastyczność, otwartość i rozwijalność struktur sterowania, przede wszystkim wymiennność modułów na poszczególnych poziomach sterowania przepływem. Globalna jakość hierarchicznej struktury procedur sterowania jest wypadkową jakości poszczególnych procedur sterowania i zasad współpracy (interakcji) procedur na różnych poziomach.

Dekompozycja struktur procedur sterowania przepływem na moduły umożliwia modelowanie i analizę różnych – stosowanych i proponowanych – schematów dla różnych poziomów sterowania przepływem. W wielu przypadkach uzyskanie efek-

tywnych modeli analitycznych i symulacyjnych procedur sterowania jest możliwe tylko wtedy, gdy dotyczą one poszczególnych rozwiązań.

6.4.6.2. INTEGRACJA I OPTIMALIZACJA PROCEDUR STEROWANIA PRZEPLYWEM

W rzeczywistych strukturach procedur sterowania dostępem często występuje łączenie funkcji różnych poziomów sterowania przepływem w jednej procedurze. Rozwiązania takie spotyka się najczęściej w systemach i sieciach komputerowych o architekturach homogenicznych (np. architektura SNA). Możliwa jest wtedy zwykle większa dowolność doboru procedur sterowania i ich zakresu funkcjonalnego. W architekturach heterogenicznych (ang. *heterogeneous network*) otwartych systemów i sieci komputerowych dowolność doboru rozwiązań jest ograniczona koniecznością stosowania standardów i spełniania wymagań wynikających z zasad łączenia systemów otwartych.

Występowanie wielu poziomów sterowania przepływem, na których są stosowane procedury wzajemnie się uzupełniające i często redundantne, powoduje, że ważnym zadaniem projektowania i analizy są integracja i interakcja procedur na różnych poziomach. Zadanie modelowania i analizy interakcji mechanizmów sterowania przepływem na różnych poziomach jest znacznie trudniejsze od analogicznych zadań dla poszczególnych procedur sterowania. Ocena łącznego efektu jednoczesnego działania uzupełniających się procedur na różnych poziomach sterowania nie jest zadaniem trywialnym.

Ze względu na to, że implementacja dowolnej struktury procedur sterowania oznacza konieczność wyłączenia części zasobów sieci z obsługi ruchu generowanego przez użytkowników, zadaniem o szczególnym znaczeniu jest optymalizacja struktury procedur sterowania. Jest oczywiste, że względnie bardziej restrykcyjne procedury sterowania na poziomach niższych wymagają mniej restrykcyjnych procedur na poziomach wyższych i odwrotnie, mniejsze możliwości eliminowania przyczyn ewentualnych przeciążeń i zakleszczeń przez mechanizmy sterowania przepływem na poziomach niższych muszą być kompensowane przez bardziej restrykcyjne mechanizmy na poziomach wyższych. Decyzja o wyborze koncepcji sterowania przepływem nie jest zagadnieniem prostym. Wymaga przede wszystkim rozstrzygnięcia, na ile korzyści wynikające z rezygnacji ze złożonych procedur sterowania przepływem na poziomach niższych kompensują ewentualny spadek efektywności wykorzystania zasobów w przypadku konieczności odrzucenia obsłużonego ruchu przez procedury sterowania na poziomach wyższych.

Oczywistość wspomnianych zależności pomiędzy procedurami sterowania przepływem przy realizacji różnych koncepcji sterowania przepływem dla jakości działania sieci nie może zastąpić jakościowych i ilościowych badań tych zależności.

Nie bez znaczenia dla efektywności struktury procedur sterowania przepływem są decyzje dotyczące: koncepcji sterowania przepływem, zakresu funkcjonalnego poszczególnych poziomów sterowania przepływem oraz doboru procedur sterowania dla poszczególnych poziomów sterowania przepływem.

6.4.6.3. STEROWANIE PRZEPLYWEM W SIECIACH Z INTEGRACJĄ TRYBÓW KOMUTACJI

Złożoność zadań sterowania przepływem zwiększa się w przypadku jednoczesnego stosowania w systemie lub sieci dwóch różnych trybów komutacji: komutacji kanałów i komutacji wiadomości (pakietów). Sterowanie przepływem w sieciach hybrydowych wymaga współistnienia dwóch różnych struktur procedur sterowania przepływem, których złożoność zależy od sposobu separacji ruchu obsługiwanego różnymi trybami komutacji. Zwielokrotnianie struktur sterowania przepływem występuje także w tych rozwiązaniach, w których do zapewnienia jakości usług stosuje się różne modele usług współdzielące zasoby sieci.

Podczas wydzielenia w systemie lub sieci zasobów kierowanych do obsługi ruchu w różnych trybach komutacji struktury sterowania przepływem są różne i niezależne. W przypadku dynamicznego współdzielenia zasobów przez różne tryby komutacji, struktura sterowania przepływem musi zawierać rozwiązania zapewniające sprawiedliwy podział zasobów.

6.4.6.4. STEROWANIE PRZEPLYWEM W SIECIACH Z INTEGRACJĄ RUCHU

Zdecydowanie inne wymagania pod adresem struktur i procedur sterowania przepływem są stawiane w systemach transmisji danych i w systemach transmisji głosu (w ogólnym przypadku do obsługi ruchu w czasie rzeczywistym). Systemy i sieci z integracją różnych klas ruchu wymagają współistnienia różnych struktur i procedur sterowania przepływem.

Zasadnicze różnice pomiędzy strukturami i procedurami sterowania przepływem dla ruchu niewrażliwego na opóźnienia (systemy z opóźnieniami – np. dane) oraz dla ruchu w czasie rzeczywistym (systemy z odrzuceniami – np. głos) wynikają z tego, że w systemach z odrzuceniami nie jest możliwe buforowanie i opóźnianie ruchu jako sposób zapobiegania przeciążeniom.

W przypadku procedur sterowania przepływem ruchu w systemach z opóźnieniem zapobieganie przeciążeniom jest osiągnięte dzięki jednoczesnemu stosowaniu mechanizmów prewencyjnych i reakcyjnych. Dopuszczalność buforowania i opóźnień powoduje, że działanie mechanizmów prewencyjnych może być i jest uzupełniane działaniem mechanizmów reakcyjnych. Działanie tych ostatnich polega na tym, że w razie wystąpienia stanu zagrażającego przeciążeniem, mimo działania mechanizmów prewencyjnych, możliwa jest detekcja tego stanu i podjęcie działań, których celem jest odtworzenie stanu niezagrażającego przeciążeniem.

Brak możliwości buforowania i opóźnień podczas obsługi ruchu w czasie rzeczywistym oznacza, że podstawowym celem procedur sterowania przepływem – a w konsekwencji struktur procedur sterowania przepływem – jest prewencja, której celem jest blokowanie połączeń mogących skutkować przeciążeniem.

6.4.6.5. STEROWANIE PRZEPLYWEM I ZARZĄDZANIE RETRANSMISJAMI

Podstawowymi zasobami, jakimi się dysponuje w każdym systemie komunikacyjnym, są: moc sygnału, czas i przepustowość. W każdym środowisku systemu komunikacyjnego istnieje możliwość zastępowania i kompensowania jednych zasobów innymi w celu uzyskania możliwie dużego transferu danych przy możliwie małym zaangażowaniu przepustowości (cel procedur sterowania przepływem) z zachowaniem akceptowalnego przez użytkowników poziomu jakości komunikacji (cel procedur kontroli błędów).

Procedury sterowania przepływem (ang. *flow control*) i procedury kontroli błędów (ang. *error control*) są głównymi komponentami każdego protokołu, rozumianego jako zbiór ustaleń i reguł, których przestrzeganie przez komunikujące się strony jest warunkiem poprawnego odbioru i interpretacji przekazywanych danych.

Różne procedury kontroli błędów stosowane w protokołach komunikacyjnych do detekcji i korekcji błędów zalicza się do dwóch grup metod: automatycznego żądania retransmisji ARQ (ang. *Automatic Retransmission (repeat) request*) oraz kodowania korekcyjnego FECC (ang. *Forward Error Correcting Coding*).

Dla pierwszej z wymienionych grup metod charakterystyczna jest korekcja błędów przez retransmisję jednostek danych, w których wykryty został błąd (za pomocą kodu detekcyjnego). W metodach należących do drugiej grupy przesyłane jednostki danych są kodowane za pomocą kodów korekcyjnych, pozwalających w razie detekcji błędów na korekcję błędnie odebranych danych. Obydwie grupy procedur kontroli błędów różnią strategię zarządzania retransmisjami (ang. *retransmission control*).

Z punktu widzenia zadań sterowania przepływem nie jest istotne, do której z wyżej wymienionych grup należy implementowana w danym protokole procedura kontroli błędów. Korzystanie z takiej czy innej procedury kontroli błędów skutkuje przyrostem liczby obsługiwanych jednostek danych (podczas retransmisji) lub przyrostem długości i liczby jednostek danych (w przypadku kodowania korekcyjnego), co zwykle zwiększa wymiar ilościowy zadania sterowania przepływem.

Realizowane w protokołach komunikacyjnych systemów i sieci komputerowych zadania sterowania przepływem i zarządzania retransmisjami są bardzo ściśle ze sobą powiązane, wzajemnie się determinują i są komplementarne.

Zadaniami komplementarnych wielopoziomowych struktur sterowania przepływem i struktur zarządzania retransmisjami są, odpowiednio, zapobieganie przeciążeniom, przez nakładanie ograniczeń na wielkość ruchu akceptowanego do obsługi w wyróżnionych fragmentach systemu lub sieci komputerowej, oraz efektywne retransmitowanie jednostek danych skierowanych do obsługi w wyróżnionych fragmentach systemu lub sieci komputerowej w razie ich odrzucenia.

Warunkiem skuteczności procedur sterowania przepływem (na każdym poziomie sterowania) jest istnienie adekwatnej do potrzeb procedury strategii zarządzania retransmisjami (i związanych z nimi zasadami przekazywania potwierdzeń). Implementacja dowolnej procedury sterowania przepływem wymusza wprowadzenie odpo-

wiedniej strategii zarządzania retransmisjami. Zwykle też strategia retransmisji spełnia, oprócz zadania retransmisji, zadania dodatkowe, związane m.in. ze sterowaniem przepływem. Strategia retransmisji gwarantuje np. zachowanie kolejności napływu jednostek danych. Zachowanie kolejności napływu może być zadaniem procedury sterowania przepływem lub strategii retransmisji.

Jedną z miar jakości usług sieci teleinformatycznej i skuteczności procedur sterowania przepływem jest udział retransmisji obsługiwanych jednostek danych w całkowitym ruchu obsługiwanym przez system lub sieć komputerową. Przyczyną retransmisji jednostek danych w sieci komputerowej jest odrzucanie jednostek danych powodowane zarówno jakością systemu transmisji danych (dane są retransmitowane z powodu błędów transmisji), jak i jakością procedur sterowania przepływem (dane są retransmitowane z powodu ich odrzucania w stanach przeciążenia). Średnia liczba jednostek danych retransmitowanych w odpowiedzi na pojawienie się odrzucenia jednostki danych zależy od stosowanej strategii retransmisji (np. retransmitowane są tylko odrzucone jednostki danych, retransmitowane są wszystkie jednostki danych następujące po odrzuconej jednostce danych).

Ograniczenie liczby retransmitowanych jednostek danych jest możliwe dzięki poprawie jakości systemu transmisji danych, skuteczności procedur sterowania przepływem i przez dobór strategii retransmisji odrzuconych jednostek danych, z tym że wybór strategii retransmisji nie jest niezależny od procedury sterowania przepływem.

Z punktu widzenia zarządzania retransmisjami nie ma znaczenia, czy jednostka danych została odrzucona z powodu wystąpienia błędu transmisji czy też z powodu braku zasobów. Z punktu widzenia procedury sterowania przepływem jest istotne, czy jednostka danych została odrzucona z powodu wystąpienia błędu transmisji czy też z powodu braku efektywności procedury sterowania przepływem.

Przy założeniu ustalonej i niezmiennej jakości systemu transmisji danych możliwa jest redukcja liczby retransmitowanych jednostek danych dzięki poprawie efektywności procedur sterowania przepływem lub strategii retransmisji wspierającej procedurę sterowania przepływem. W krańcowym przypadku liczba retransmisji jednostek danych, powodowanych procedurami sterowania przepływem, może być zredukowana do zera, jeżeli w procedurach sterowania przepływem stosowane są tylko mechanizmy prewencyjne i brak jest mechanizmów reakcyjnych.

Ścisły związek pomiędzy procedurami sterowania przepływem i strategiami retransmisji objawia się m.in. jednoczesną implementacją w protokołach architektury warstwowych i wykorzystywaniem do realizacji swych zadań tych samych zbiorów informacji (numerów sekwencyjnych, numerów potwierżeń, rozmiarów okien itd.). Wyrazem ich ścisłego powiązania jest także to, że decyzja o realizacji lub braku procedur sterowania przepływem na określonym poziomie sterowania przepływem jest decyzją o stosowaniu lub nie określonej strategii zarządzania retransmisjami.

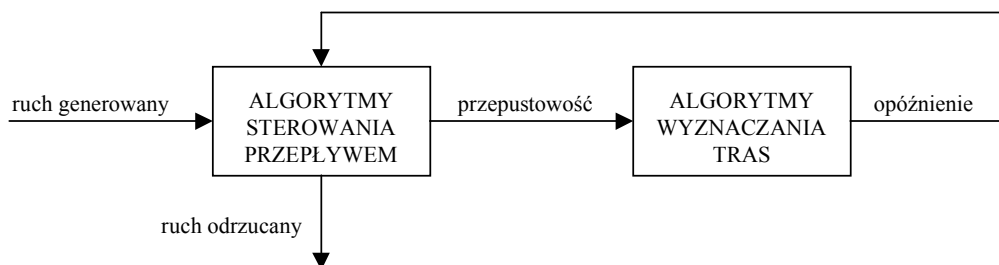
Realizacja procedur sterowania przepływem, kontroli błędów i zarządzania retransmisjami zawsze się wiąże z koniecznością przeznaczenia części zasobów na cele organizacyjne. Jednym ze sposobów minimalizacji ilości tak wykorzystywanych za-

sobów są efektywne strategie zarządzania retransmisjami wspierającymi jednocześnie sterowanie przepływem i kontrolę błędów.

6.4.6.6. STEROWANIE PRZEPLYWEM I ZADANIA WYZNACZANIA TRAS

Sterowanie przepływem oraz wyznaczanie tras, po których realizowany jest przepływ, są ściśle ze sobą powiązane. Stopień powiązania procedur realizujących wymienione zadania ma znaczący wpływ na efektywność wykorzystania zasobów, podobnie jak kolejność stosowania tych procedur.

Dwie podstawowe miary jakości usług sieci, na wartości których wpływają algorytmy wyznaczania tras, to przepustowość i średnie opóźnienie pakietów. Algorytmy wyznaczania tras oraz procedury sterowania przepływem determinują wartości wymienionych miar jakości zgodnie z mechanizmem sprzężenia zwrotnego (rysunek 6.45).

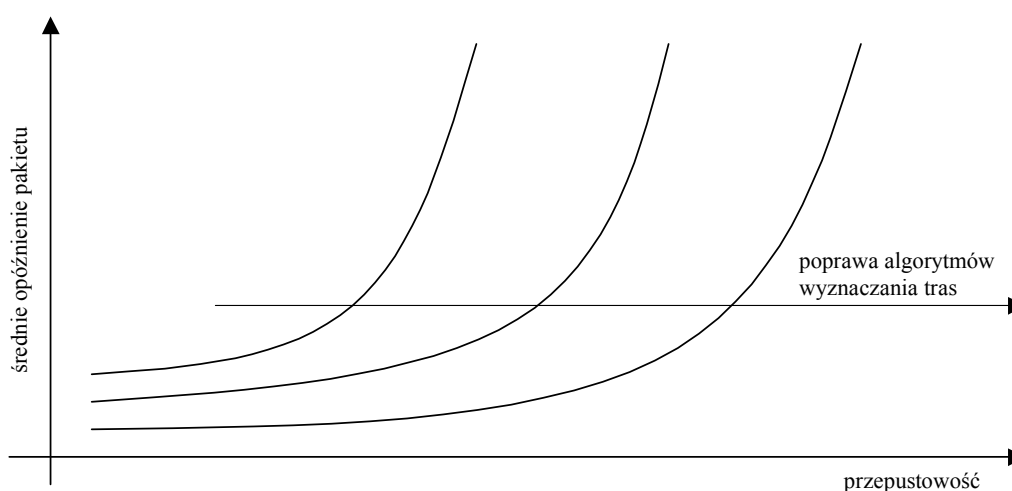


Rysunek 6.45. Zasada współdziałania procedur sterowania przepływem i algorytmów wyznaczania tras

Jeżeli ilość ruchu generowanego przez źródło jest względnie mała, to jest on w całości akceptowany do obsługi w sieci: generowany ruch jest równy przepustowości sieci. Jeżeli ilość ruchu generowanego rośnie, to jego część może być odrzucana przez procedury sterowania przepływem – w takim przypadku przepustowość jest równa różnicy między ruchem generowanym i odrzucanym. Średnie opóźnienie, wnoszone przez sieć dla ruchu przyjętego do obsługi w sieci, zależy od tras wybieranych przez algorytmy wyznaczania tras. Od nich zależy także (pośrednio) przepustowość sieci, ze względu na to, że typowe działanie procedur sterowania przepływem polega na równoważeniu przepustowości i opóźnienia; ilość odrzucanego ruchu zwiększa się, jeżeli rośnie wartość opóźnienia. Jeśli więc algorytmy wyznaczania tras prowadzą do niższych wartości opóźnień, to procedury sterowania przepływem przyjmują do obsługi w sieci większą ilość generowanego ruchu. Efektem działania lepszego algorytmu wyznaczania tras, w warunkach dużego obciążenia sieci, jest korzystniejszy przebieg krzywej przepustowość–opóźnienie, wyznaczającej warunki obsługi ruchu w sieci (rysunek 6.46).

Od dobrego algorytmu wyznaczania tras oczekuje się zwiększenia przepustowości dla tych samych wartości średniego opóźnienia w warunkach dużego obciążenia sieci, oraz zmniejszenia średniego opóźnienia przy małym obciążeniu sieci.

W tradycyjnych rozwiązaniach podsystemów komunikacyjnych sieci z komutacją pakietów (wiadomości) zadania sterowania przepływem i doboru tras rozwiązywane są niezależnie. Konsekwencją założenia o rozłączności wspomnianych zadań jest niezależność procedur sterowania przepływem i zadań doboru tras. W takim przypadku zadaniem sterowania przepływem jest zapobieganie nadmiarowości ruchu obsługiwanego w systemie lub sieci, podczas gdy zadaniem procedur doboru tras jest efektywny transfer ruchu zaakceptowanego do obsługi. Rozłączność procedur sterowania przepływem i doboru tras jest rozwiązaniem efektywnym w przypadku trybu datagramowego komutacji pakietów i statycznych reguł doboru tras.



Rysunek 6.46. Krzywe przepustowość–średnie opóźnienie dla różnych algorytmów wyznaczania tras

Złożoność zadań dynamicznego sterowania przepływem i algorytmów adaptacyjnego doboru tras w systemach i sieciach z komutacją pakietów w trybie połączeń wirtualnych powoduje potrzebę integracji procedur.

W rozwiązaniach z połączeniami wirtualnymi zmienia się, w porównaniu z rozwiązaniami datagramowymi, rola procedur sterowania przepływem i doboru tras, zwiększa się znaczenie algorytmów dynamicznego doboru tras i prewencyjnych metod sterowania przepływem. Zwykle nadrzędną rolę pełnią reguły doboru tras. Realizacja procedury wyznaczania trasy polega na ocenie możliwości zestawienia w podsystemie komunikacyjnym trasy z niewykorzystywanych aktualnie (resztkowych) pojemności kanałów transmisyjnych. Jeżeli zestawienie trasy o wymaganych charakterystykach nie jest możliwe, żądanie obsługi połączenia wirtualnego jest odrzucane w węźle źródłowym przez procedurę sterowania przepływem na poziomie dostępu do sieci. Oznacza to, że sterowanie przepływem ma charakter sterowania prewencyjnego.

Mniej efektywne jest rozwiązanie wykorzystujące reakcyjne sterowanie przepływem, tzn. takie, gdy żądania zestawienia połączenia wirtualnego są – mimo braku gwa-

rancji możliwości jego obsłużenia – akceptowane. W takim przypadku, jeżeli w trakcie obsługi jednostek danych transferowanych w zestawionym połączeniu wirtualnym podsystem komunikacyjny przechodzi w stan przeciążenia, następuje odrzucanie transferowanych jednostek. Odrzucanie jednostek danych trwa aż do chwili odtworzenia stanu normalnego (stanu bez przeciążenia) podsystemu komunikacyjnego.

6.4.6.7. STEROWANIE PRZEPIYWEM W SYSTEMACH I SIECIACH ROZGŁOSZENIOWYCH

Zadanie sterowania przepływem, którego celem jest zapobieganie przeciążeniom, w wielodostępnych podsystemach komunikacyjnych wykorzystujących łącza transmisyjne typu punkt–punkt (ang. *single hop*) jest znacznie prostsze niż w wielodostępnych podsystemach komunikacyjnych korzystających z wielodostępnych łączy transmisyjnych (ang. *multihop*) współdzielonych pomiędzy wiele źródeł generujących ruch niezależnie od siebie.

W podsystemach komunikacyjnych opartych na transmisji punkt–punkt zadanie zapobiegania przeciążeniom rozwiązywane jest efektywnie przez procedury sterowania przepływem na poziomie łącza i trasy, tzn. procedury implementowane w protokołach dostępu do kanału transmisyjnego lub trasy (ciągu szeregowo połączonych kanałów transmisyjnych). W omawianych podsystemach skuteczne zapobieganie przeciążeniom pamięci buforowych jest równoznaczne ze skutecznym zapobieganiem przeciążeniom łączy transmisyjnych.

W podsystemach komunikacyjnych opartych na niezależności źródeł ruchu współdzielących kanały transmisyjne zadanie zapobiegania przeciążeniom jest znacznie bardziej złożone. Przyczyną tego jest, inny niż w przypadku łączy typu punkt–punkt, charakter zależności pomiędzy przeciążeniem pamięci buforowych i przeciążeniem kanałów transmisyjnych. Brak przeciążenia pamięci buforowych węzłów nie eliminuje możliwości wystąpienia przeciążeń kanałów transmisyjnych, a przeciążenie kanałów transmisyjnych może być przyczyną powstania przeciążenia pamięci buforowych. Skuteczne sterowanie przepływem w omawianym przypadku wymaga koordynacji procedur alokacji pamięci buforowych i procedur retransmisji.

6.4.6.8. STEROWANIE PRZEPIYWEM W SYSTEMACH I SIECIACH HETEROGENICZNYCH

Łączenie sieci teleinformatycznych jest na ogół związane m.in. z koniecznością łączenia i konwersji różnych struktur i procedur sterowania przepływem.

W przypadku łączenia systemów i sieci, zbudowanych według tej samej architektury, struktury i procedury sterowania przepływem są niezmiennie. Wynikiem łączenia jest przyrost wartości średniego opóźnienia pakietów (wiadomości, bajtów) i potwierdzeń wnoszonych niewpływający na efektywność prewencyjnych, znacznie natomiast zmniejszający efektywność reakcyjnych metod sterowania przepływem.

W przypadku łączenia systemów i sieci zbudowanych na podstawie różnych architektur konieczna jest konwersja procedur sterowania przepływem na różnych poziomach. W ogólnym przypadku łączenie realizujących konwersję protokołów odbywa się za pomocą bramy (ang. *gateway*). Konwersja protokołów oznacza także konwersję implementowanych w nich różnych mechanizmów, w tym mechanizmów doboru tras i sterowania przepływem.

Bramy, implementowane w sieciach złożonych na stykach pomiędzy różnymi podsieciami, można traktować jako dodatkowe węzły, w których następuje zamiana hierarchii protokołów, wpływająca na zamianę procedur sterowania przepływem zależnie od zasięgu ich działania na poszczególnych poziomach sterowania. Wpływ ten można scharakteryzować następująco:

1. Ze względu na to, że zasięg działania procedur sterowania przepływem na poziomie łącza jest lokalny i ograniczony do pojedynczego łącza, w bramie – traktowanej jako kolejny węzeł na trasie – następuje zamiana jednej procedury sterowania przepływem na poziomie łącza na inną. W strukturze procedur sterowania lokalne zadania zapobiegania przeciążeniom węzłów są realizowane przez ciąg różnych, niezależnych procedur sterowania.

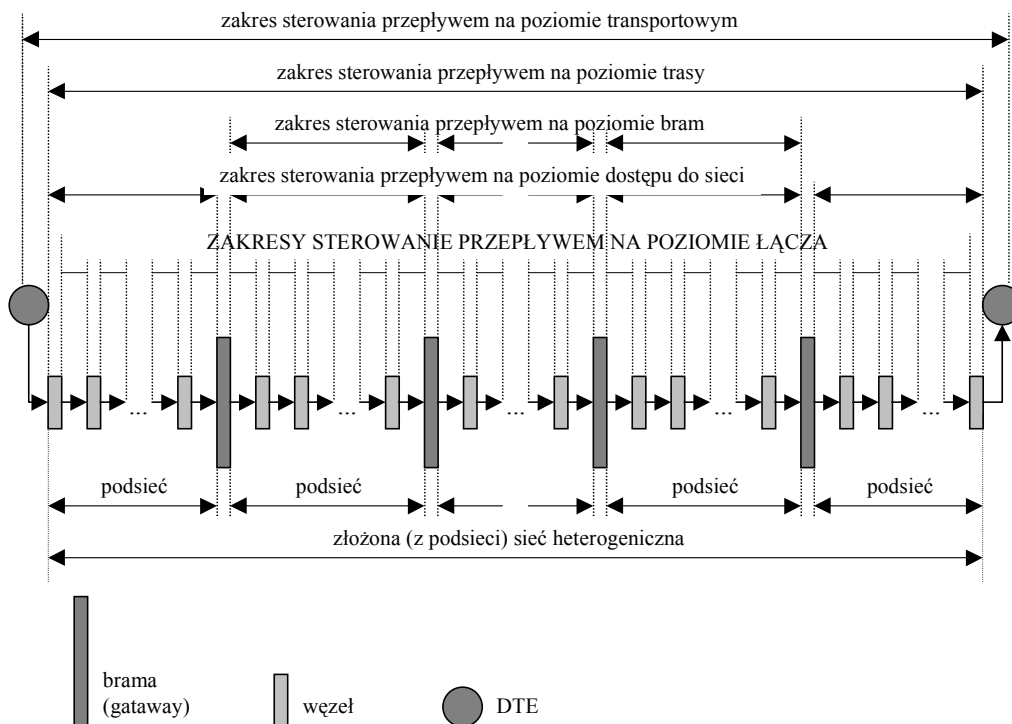
2. Wprowadzenie bram pomiędzy poszczególnymi podsieciami, podobnie jak w przypadku procedur sterowania przepływem na poziomie łącza, nie powoduje potrzeby zmian procedur sterowania przepływem na poziomie dostępu do poszczególnych podsieci składowych domen. Z punktu widzenia sterowania przepływem na poziomie dostępu brama jest węzłem ujściowym podsieci. Z punktu widzenia struktury sterowania przepływem lokalne zadania zapobiegania przeciążeniom globalnym podsieci są realizowane przez ciąg różnych, niezależnych procedur sterowania dostępem. Intensywność wprowadzania nowych jednostek danych do sieci jest determinowana możliwością obsługi przez podsieć o najmniejszej przepustowości.

3. Ponieważ zasięg sterowania przepływem na poziomie trasy wykracza poza podsieci, a jego efektywność zależy od stanu wszystkich węzłów na trasie, zmiana reguł doboru tras realizowana w bramach może więc w istotny sposób wpływać na sposoby zapobiegania przeciążeniom trasy. Oznacza to, że w procedurze sterowania przepływem na poziomie trasy muszą być uwzględniane obciążenia pamięci buforowych bram oraz sąsiednich podsieci. Konieczność przekazywania tych informacji do węzłów źródłowych uzasadnia dodatkowy poziom sterowania przepływem – na poziomie bramy. Celem tego poziomu sterowania przepływem jest zapobieganie przeciążeniom bram, z tym że implementowane tu procedury sterowania muszą być zgodne (zgodność wertykalna) ze wszystkimi procedurami stosowanymi w podsieci (podsieciach) pomiędzy bramami.

4. Wprowadzanie bram między podsieciami nie wymaga ingerencji w procedury sterowania przepływem na poziomie transportowym dlatego, że dotyczy ono przepływów pomiędzy procesami końcowymi; z tego punktu widzenia sterowanie przepływem pomiędzy bramami jest sterowaniem wspierającym.

Łączenie różnych struktur sterowania przepływem różnych podsieci w nadrzędną strukturę sterowania przepływem sieci heterogenicznej wymaga wprowadzenia dodat-

kowego poziomu sterowania przepływem, zlokalizowanym pomiędzy sterowaniem przepływem na poziomie dostępu do sieci a sterowaniem przepływem na poziomie transportowym. Od procedur sterowania przepływem na poziomie bram wymagana jest, oprócz ich zgodności wertykalnej z procedurami sterowania na pozostałych poziomach sterowania w poszczególnych podsieciach sieci, ich zgodność horyzontalna na trasach poprzez różne podsieci – zgodność łańcucha procedur (rysunek 6.47).



Rysunek 6.47. Zgodność wertykalna i horyzontalna procedur sterowania przepływem w sieciach heterogenicznych

Konsekwencją zgodności wertykalnej jest m.in. to, że efektywność sterowania przepływem na poziomie bram jest determinowana efektywnością hierarchii procedur sterowania przepływem podsieci łączących bramy. Skutkiem zgodności horyzontalnej jest to, że efektywność procedur sterowania przepływem na poziomie trasy i poziomie transportowym jest determinowana przez procedurę sterowania przepływem na poziomie bram o najmniejszej efektywności w łańcuchu. Znaczące różnice w efektywności procedur sterowania na poziomie bram mogą być kompensowane tylko i wyłącznie efektywnością procedur sterowania przepływem na poziomie transportowym, jako poziomie nadrzędnym wobec poziomu bram.

LITERATURA

- [1] ADAS A., *Traffic models in broadband telecommunications networks*, IEEE Communications Magazine, July 1997.
- [2] ATKINS J., NORRIS M., *Total Area Networking*, Wiley, Chichester 1995.
- [3] BERTSEKAS D., GALLAGER R., *Data networks*, Prentice-Hall International, Englewood Cliffs 1987.
- [4] BHARATH-KUMAR K., JAFFE J.M., *A new approach to performance oriented flow control*, IEEE Transactions on Communications, Vol. COM-29, No. 4, April 1981.
- [5] BHARATH-KUMAR K., JAFFE J.M., *Routing to multiple destinations in computer networks*, IEEE Transactions on Communications, Vol. COM-31. V 3, March 1983, s. 343–351.
- [6] CHENG R.G., CHANG C.J., *Design of a fuzzy traffic controller for ATM networks*, IEEE/ACM Transactions on Networking, Vol. 4, No. 3, June 1996, s. 160–168.
- [7] CHEUNG S.Y., PENCEA S., *Pipelined sections: a new buffer management discipline for scalable QoS provision*, Emory University (www.itpapers.com).
- [8] COURTOIS P.J., *The M/G/1 finite capacity queue with delays*, IEEE Transactions on Communications, Vol. COM-28, February 1982, s. 165–172.
- [9] CZACHÓRSKI T., *Modele kolejkowe w ocenie efektywności sieci i systemów komputerowych*, Pracownia Komputerowa Jacka Skalmierskiego, Gliwice 1999.
- [10] DOVROLIS C., RAMANATHAN P., *A case for Relative Differentiated Services and the Proportional Differentiation Model*, IEEE Network, September/October 1999, s. 26–34.
- [11] DEO N., *Teoria grafów i jej zastosowania w technice i informatyce*, PWN, Warszawa 1980.
- [12] DOLAN A., ALDOUS J., *Networks and algorithms*, Wiley, Chichester 1993.
- [13] FINDEISEN W., SZYMANOWSKI J., WIERZBICKI A., *Teoria i metody obliczeniowe optymalizacji*, PWN, Warszawa 1977.
- [14] FORD L.R., FULKERSON D.R., *Przepływy w sieciach*, PWN, Warszawa 1969.
- [15] GERLA M., KLEINROCK L., *On the topological design of distributed computer networks*, IEEE Transactions on Communications, Vol. COM-25, 1977, s. 48–60.
- [16] GERLA M., KLEINROCK L., *Flow control: A comparative survey*, IEEE Transactions on Communications, Vol. COM-28, No. 4, 1980, s. 553–574.
- [17] GRZECH A., *Channel queue limit flow control method in packet-switched computer communication network*, Proceedings of Fourth International Conference on Systems Engineering, Coventry 1985, s. 359–367.
- [18] GRZECH A., *Hop level flow control by dynamical restrictions in the node of computer communications network*, Systems Science, Vol. 11, No. 3/4, 1982, s. 95–111.
- [19] GRZECH A., *Decentralized control of perturbed flows in computer communication network*, Proceedings of the 18th Annual Hawaii International Conference on System Sciences, Vol. 1, Honolulu 1985, s. 383–390.
- [20] GRZECH A., *Flow control in virtual circuit-oriented packet-switched computer communication network*, System analysis and simulation 1985. Proceedings of the (2nd) International Symposium, Vol. 1 – Theory and foundations, Verlag, Berlin 1985, s. 437–440.
- [21] GRZECH A., *Sterowanie przepływem w węźle sieci komputerowej z wykorzystaniem funkcji ograniczającej*, II krajowa konferencja naukowo-techniczna „Komputerowe sterowanie transmisją informacji”, t. 1, Bydgoszcz 1986, s. 132–139.

- [22] GRZECH A., KASPRZAK A., *An exact algorithm for topology, capacity and multi-commodity maximal flow assignment problem in a computer network*, Systems Science, Vol. 20, No. 2, s. 83–94, 1994.
- [23] GRZECH A., KASPRZAK A., *Static and quasi-static routing in integrated services digital networks*, Applied Mathematics and Computer Science, Vol. 4, No. 1, 1994, s. 107–124.
- [24] GRZECH A., *Control and traffic organisation in segmented local area networks*, Proceedings of IEEE Singapore International Conference on Networks, IEEE, 1995, s. 131–135.
- [25] KERMANI P., KLEINROCK L., *Dynamic flow control in store-and-forward computer networks*, IEEE Transactions on Communications, Vol. COM-28, No. 2, February 1980, s. 263–270.
- [26] KLEINROCK L., *Communication nets*, Volume I: *Stochastic message flow and delay*, Volume II: *Computer Applications*, Wiley, New York 1976.
- [27] KULKARNI A., LI A.Q., *Performance analysis of a rate-based feedback control scheme*, IEEE/ACM Transactions on Networking, Vol. 6, No. 6, December 1998, s. 797–810.
- [28] LAW A.M., McCOMAS G., *Simulation software for communications networks: The state of the art*, IEEE Communications Magazine, March 1994, s. 44–50.
- [29] MISHRA S., WU L., *An evaluation of flow control in group communication*, IEEE/ACM Transactions on Networking, Vol. 6, No. 5, October 1998, s. 571–587.
- [30] ONVURAL R.O., *Asynchronous Transfer Mode; Performance issues*, Artech House, Boston 1995.
- [31] PAPIR Z. (red.), *Sieci z komutacją pakietów – od X.25 do Frame Relay i ATM*, Wydawnictwo Fundacji Postępu Telekomunikacji, Kraków 1997.
- [32] PAPIR Z., *Ruch telekomunikacyjny i przeciążenia sieci pakietowych*, WKŁ, Warszawa 2001.
- [33] PAXSON V., *End-to-end Internet packet dynamics*, IEEE/ACM Transactions on Networking, Vol. 7, No. 3, June 1999, s. 277–292.
- [34] PERROS H.G., ELSAYED K.M., *Call Admission Control schemes: a review*, North Carolina State University (<ftp://eceyv.ncsu.edu>).
- [35] SAITO H., *Call Admission Control in an ATM network using upper bound of Cell Loss Probability*, IEEE Transactions on Communications, Vol. 40, No. 9, September 1992, s. 1512–1521.
- [36] SEIDLER J., *Analiza i synteza sieci łączności dla systemów teleinformatycznych*, PWN, Warszawa 1979.
- [37] STALLINGS W., *High-speed networks: TCP-IP and ATM design principle*, Prentice-Hall, New York 1998.
- [38] TAKAGI H. (red.), *Stochastic analysis of computer and communication systems*, North-Holland, Amsterdam 1990.
- [39] TURNER J.S., *Managing bandwidth in ATM networks with bursty traffic*, IEEE Network, 1992, s. 50–58.
- [40] WOLFINGER B.E., *Characterization of mixed traffic load in service integrated networks*, Proceedings 6th Polish Teletraffic Symposium, Wrocław 1999, s. 11–28.

7. PRZECIWDZIAŁANIE PRZECIĄŻENIOM

Architektury sieci teleinformatycznych korzystają z różnych zbiorów i hierarchii metod przeciwdziałania przeciążeniom i sterowania przepływem. Uniwersalność metod przeciwdziałania przeciążeniom wynika stąd, że do rozwiązywania różnych zadań związanych z powstawaniem i rozładowywaniem przeciążeń są stosowane podobne sposoby detekcji przeciążeń i algorytmy rozładowywania przeciążeń. W większości przypadków nie ma jednoznacznego podziału metod przeciwdziałania przeciążeniom na prewencyjne i reakcyjne, w zależności bowiem od miejsca w sieci oraz sposobu implementacji danej metody przeciwdziałania, w różnym stopniu wykorzystywane są jej możliwości do działań prewencyjnych i reakcyjnych.

W pewnym uproszczeniu można przyjąć, że podstawowym kryterium podziału metod przeciwdziałania przeciążeniom na prewencyjne i reakcyjne jest rodzaj danych, na podstawie których są podejmowane decyzje sterujące. W metodach prewencyjnych decyzję podejmuje się w zasadzie na podstawie danych *a priori*, dotyczących obsługiwanego strumienia jednostek danych i dostępnych zasobów. W metodach reakcyjnych zazwyczaj są uwzględniane dane *a posteriori*, opisujące konsekwencje wcześniej podjętych decyzji o przyjęciu wywołań i przydziale zasobów.

7.1. STEROWANIE PREWENCYJNE I REAKCYJNE

Mechanizmy przeciwdziałania przeciążeniom, proponowane m.in. w architekturach sieci ATM (ang. *Asynchronous Transfer Mode*), sieci FR (ang. *Frame Relay*) i sieci TCP/IP (ang. *Transmission Control Protocol/Internet Protocol*) dzieli się na ogół na dwie grupy: mechanizmy prewencyjne (ang. *preventive*) i mechanizmy reakcyjne (ang. *reactive*).

Zakresy zastosowań i efektywność implementowanych mechanizmów przeciwdziałania przeciążeniom ściśle zależą od architektury sieci, hierarchii i struktury dostarczanych usług, poziomu granulacji przepływów oraz czasu trwania sterowanych

procesów i zdarzeń. W zasadzie każda architektura sieci, charakteryzowana m.in. rozmiarem jednostek danych, zakresem usług, liczbą parametrów jakości usług, determinuje zbiór i hierarchię metod przeciwdziałania przeciążeniom.

Zadaniem technik sterowania prewencyjnego jest zapobieganie przeciążeniom w wyniku podejmowania działań uprzedzających – tzn. zmierzających do zapobiegania zdarzeniom, których następstwem są przeciążenia. Sterowanie prewencyjne nie eliminuje przeciążeń, nie jest wystarczające do zapobiegania przeciążeniom i jeżeli przeciążenie powstanie, niezbędne są działania zmierzające do jego usunięcia. Dzieje się tak dlatego, że efektywność metod prewencyjnych zwiększa się wtedy, gdy zasoby sieci są rozdzielane deterministycznie, a to z kolei powoduje niewielką efektywność wykorzystania zasobów.

Zadaniem metod reakcyjnych jest odtworzenie stanu sieci poprzedzającego powstanie stanu przeciążenia (rozładowanie przeciążenia) i odbywa się zwykle zgodnie z pewnymi ogólnymi zasadami. Detekcja stanu przeciążenia oznacza uruchomienie wcześniej zdefiniowanych procedur, których celem jest na ogół spowolnienie lub wstrzymanie napływu nowych żądań obsługi aż do chwili wyjścia sieci ze stanu przeciążenia.

Głównym ograniczeniem skuteczności mechanizmów reakcyjnych jest wartość ilorazu czasu propagacji do czasu transmisji jednostki danych. Wzrost wartości tego ilorazu oznacza zwiększenie prawdopodobieństwa, że podjęta akcja może być spóźniona. Zjawisko to jest określane mianem bezwładności sieci, a jego miarą jest wartość iloczynu czasu propagacji jednostki danych i obciążenia sieci.

Dodatkowym kryterium wyróżniania kategorii mechanizmów sterowania przepływem i przeciwdziałania przeciążeniom są różne długości przedziału czasu, wpływającego od chwili podjęcia decyzji do chwili pojawienia się skutków podjętej decyzji. Mechanizmy prewencyjne praktycznie nie mają takich ograniczeń; są zwykle stosowane w punktach dostępu do usług sieci. Skuteczność mechanizmów reakcyjnych zależy od szybkości zmian procesów, których dotyczą. Mechanizmy te działają (są skuteczne) w czasie nie krótszym od czasu propagacji.

Innym kryterium porównywania skuteczności mechanizmów prewencyjnych i reakcyjnych jest jakość dostarczanych usług oraz stopień wykorzystania zasobów. Mechanizmy prewencyjne, z natury bardziej zachowawcze, umożliwiają uzyskanie wyższej jakości usług za cenę obniżenia efektywności wykorzystania zasobów.

W rzeczywistych rozwiązaniach sieciowych metody prewencyjne i reakcyjne współistnieją i uzupełniają się. Zwykle większa skuteczność stosowanych metod prewencyjnych powoduje mniejsze zapotrzebowanie na metody reakcyjne. Skuteczność metod prewencyjnych i reakcyjnych zależy od charakteru zmienności ruchu generowanego przez źródła. Mniejszy zakres zmienności zwiększa skuteczność działania metod prewencyjnych.

Ze względu na to, że skuteczność metod reakcyjnych, determinowana bezwładnością sieci, maleje wraz ze wzrostem szybkości obsługi jednostek danych, w sieciach szybkich i szerokopasmowych główne znaczenie mają metody prewencyjne.

7.1.1. PRZECIWDZIAŁANIE PRZECIĄŻENIOM W SIECI ATM

7.1.1.1. ZADANIA PRZECIWDZIAŁANIA PRZECIĄŻENIOM W SIECI ATM

Charakterystyczny dla techniki ATM asynchroniczny, skalowalny oraz zorientowany połączeniowo przekaz jednostek danych odbywa się z wykorzystaniem ograniczonego zbioru mechanizmów przeciwdziałania przeciążeniom, rozładowywania przeciążeń, sterowania przepływem oraz korekcji błędów, co powoduje m.in. możliwość znacznego ograniczenia czasu przetwarzania komórek w węzłach sieci i obsługi ruchu z dużymi szybkościami.

Trudność konstruowania i implementacji efektywnych procedur sterowania przepływem jednostek danych w sieci ATM wynika zarówno z ograniczeń architektury sieci ATM, jak i z przeznaczenia techniki ATM do jednoczesnej obsługi klas ruchu, charakteryzowanych różną liczbą parametrów o dużych zakresach wartości.

Istotne, z punktu widzenia przeciwdziałania przeciążeniom, ograniczenia architektury ATM to niewielka nadmiarowość nagłówka komórki, redukująca możliwość implementowania różnych procedur detekcji i przeciwdziałania przeciążeniom. Konieczność jednoczesnej obsługi klas ruchu o różnych charakterystykach, zwłaszcza klas ruchu o różnej wrażliwości na opóźnienia i straty powoduje, że w sieci ATM maleje znaczenie technik reakcyjnych, tzn. technik sterowania przepływem polegających na dławieniu szybkości przepływu jako sposobów przeciwdziałania i rozładowywania przeciążeń. Innym powodem małej skuteczności reakcyjnych technik przeciwdziałania przeciążeniom, opartych na propagacji informacji o przeciążeniu, jest szybkość transferu w sieci ATM. Porównywalny czas propagacji i czas transmisji komórki ATM wymusza stosowanie technik prewencyjnego sterowania przepływem.

Ze względu na ograniczone możliwości sterowania przepływem w sieciach ATM za pomocą mechanizmu sprzężenia zwrotnego, podstawowym mechanizmem przeciwdziałania przeciążeniom jest sterowanie dostępem do sieci, tzn. przeciwdziałanie powstawaniu przeciążeń odbywa się przez reglamentowanie ilości ruchu wprowadzanego do sieci. Reglamentacja ruchu polega na sprawdzaniu możliwości zestawienia nowych połączeń, spełniających wymagania dotyczące jakości usług. Ponieważ prewencyjne sterowanie przepływem nie jest warunkiem wystarczającym zapobiegania przeciążeniom, sterowanie prewencyjne jest więc uzupełniane, w ograniczonym zakresie, procedurami reakcyjnymi. Warunkiem przyjęcia nowego wywołania do obsługi w sieci ATM jest możliwość zestawienia żadanego połączenia o ustalonych charakterystykach ilościowych i jakościowych, precyzowanych w kontrakcie ruchowym (ang. *traffic contract*). Gwarancją skuteczności prewencyjnych procedur przeciwdziałania przeciążeniom (ang. *congestion avoidance*), polegających na reglamentacji przyjmowanych do obsługi wywołań, jest efektywność procedur:

- sprawdzenia możliwości przyjęcia nowego wywołania,
- wymuszania przestrzegania kontraktu ruchowego przez źródło strumieni komórek oraz przez sieć.

7.1.1.2. USŁUGI, PARAMETRY RUCHOWE ŹRÓDEŁ I PARAMETRY JAKOŚCI USŁUG W SIECI ATM

Skuteczna implementacja procedur prewencyjnego przeciwdziałania przeciążeniom, korzystających z mechanizmów sprawdzania możliwości przyjmowania wywołań i wymuszania przestrzegania kontraktów ruchowych, wymaga kategoryzacji: usług sieci (warstwy) ATM, źródeł strumieni komórek ATM oraz parametrów jakości usług. Kategoryzacja, polegająca na ograniczeniu liczby różnych usług sieci, liczby parametrów ruchowych opisujących źródła ruchu oraz liczby parametrów opisujących jakość usług sieci, jest warunkiem ograniczenia liczby różnych algorytmów przeciwdziałania przeciążeniom i rozładowywania przeciążeń.

W przypadku sieci ATM ograniczone zbiory:

1. Usług sieci:

- o stałej szybkości bitowej CBR (ang. *Constant Bit Rate*),
- o zmiennej szybkości bitowej realizowanej w czasie rzeczywistym rt-VBR (ang. *real time – Variable Bit Rate*),
- o zmiennej szybkości bitowej nierealizowanej w czasie rzeczywistym nrt-VBR (ang. *non real time – Variable Bit Rate*),
- o nieokreślonej szybkości bitowej UBR (ang. *Unspecified Bit Rate*),
- o dostępnej szybkości bitowej ABR (ang. *Available Bit Rate*);

2. Parametrów ruchowych źródeł:

- maksymalna szybkość generowania komórek PCR (ang. *Peak Cell Rate*),
- średnia szybkość generowania komórek SCR (ang. *Sustainable Cell Rate*),
- minimalna szybkość generowania komórek MCR (ang. *Minimum Cell Rate*),
- maksymalny rozmiar paczki komórek MBS (ang. *Maximum Burst Size*),
- zakres zmienności (tolerancja) opóźnienia komórek CVDT (ang. *Cell Variation Delay Tolerance*);

3. Parametrów jakości usług:

- maksymalne opóźnienie komórki CTD_{max} (ang. *maximum Cell Transfer Delay*),
- średnie opóźnienie komórki CTD_{mesn} (ang. *mean Cell Transfer Delay*),
- maksymalny zakres zmienności opóźnienia komórek CDV_{max} (ang. *peak-to-peak Cell Delay Variation*),
- prawdopodobieństwo utraty komórki CLR (ang. *Cell Loss Rate*),
- prawdopodobieństwo przekłamania komórki CER (ang. *Cell Error Rate*),
- prawdopodobieństwo przekłamania paczki komórek SECBR (ang. *Severely Errored Cell Block Ratio*),
- intensywność błędnie komutowanych komórek CMR (ang. *Cell Misinsertion Rate*);

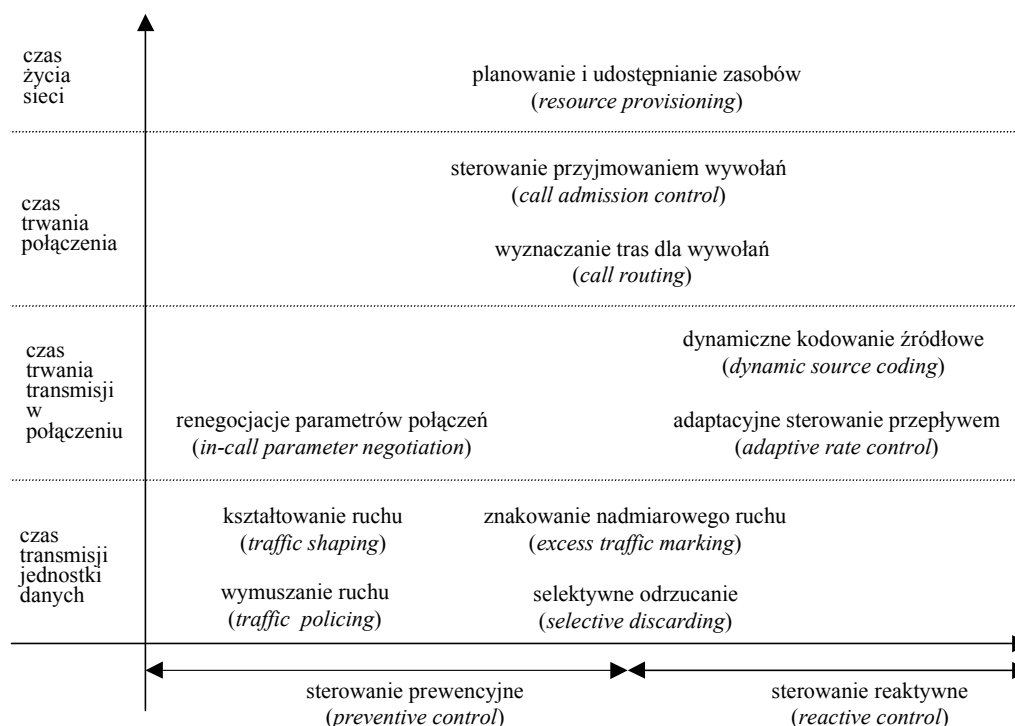
są stosowane do formułowania różnych procedur prewencyjnego i reakcyjnego przeciwdziałania przeciążeniom, które dotyczą:

- różnych usług o różnej jakości, dostarczanych w sieci dla różnych klas ruchu definiowanych wartościami parametrów ruchowych źródeł,

- różnych procesów realizowanych na potrzeby transferu komórek w sieci,
- alokacji zasobów w sieci i jej wybranych elementach itd.

7.1.1.3. METODY PREWENCYJNEGO I REAKCYJNEGO PRZECIWDZIAŁANIA PRZECIĄŻENIOM W SIECI ATM

Zbiór metod prewencyjnych i reakcyjnych przeciwdziałania przeciążeniom w sieci ATM przedstawiono na rysunku 7.1.



Rysunek 7.1. Klasyfikacja prewencyjnych i reakcyjnych metod przeciwdziałania przeciążeniom w sieciach ATM

Ograniczenia architektury sieci ATM oraz szeroki zakres zmienności wartości parametrów opisujących klasy ruchu obsługiwane w tych sieciach decydują o podstawowym, dla tych sieci, znaczeniu prewencyjnych metod przeciwdziałania przeciążeniom i marginalnym znaczeniu metod reakcyjnych. Prewencyjny charakter metod przeciwdziałania przeciążeniom w sieciach ATM determinuje zbiór procedur (rysunek 7.1) dotyczących:

- zarządzania zasobami w sieci (ang. *network resource management*),
- alokacji zasobów sieci (ang. *network resource allocation*),
- sterowania przyjmowaniem wywołań (ang. *connection admission control*),
- nadzorowania kontraktów ruchowych (ang. *usage parameter control*),

➤ wymuszania przestrzegania zawartych kontraktów ruchowych przez źródła ruchu (ang. *traffic shaping procedure*) itd.

Wiele z mechanizmów i procedur przeciwdziałania przeciążeniom stosowanych w technice ATM ma charakter uniwersalny w tym znaczeniu, że albo korzystają z rozwiązań stosowanych w innych architekturach sieci, albo są wykorzystywane w innych koncepcjach i architekturach sieci zorientowanych na dostarczanie usług o zdefiniowanej jakości.

7.1.2. PRZECIWDZIAŁANIE PRZECIĄŻENIOM W SIECI FR

Rozwiązania z zakresu sterowania przepływem i przeciwdziałania przeciążeniom w sieciach FR (ang. *Frame Relay*), ze względu na rodowód i cechy techniki FR, są porównywane zarówno z rozwiązaniami stosowanymi w sieciach X.25, jak i sieciach ATM. Uproszczona budowa ramki protokołu sieci FR, w porównaniu z budową pakietu protokołu X.25, uniemożliwia stosowanie m.in. metody okna, jako skutecznej metody przeciwdziałania przeciążeniom.

Uproszczenie budowy ramki sieci FR stosowanej w przezroczystym kanale transmisyjnym organizowanym ponad łączem transmisyjnym o dużej niezawodności, w porównaniu z nadmiarowością protokołu X.25 uzasadnioną zawodnością łączy analogowych, pozwala na zmniejszenie opóźnień tranzytowych i wzrost przepustowości.

Degradacja jakości usług sieci FR, tzn. zwiększenie opóźnienia oraz zmniejszenie przepustowości, jest powodowana nadmiernym wzrostem zajętości pamięci buforowych w węzłach sieci. Dlatego celem metod przeciwdziałania przeciążeniom w sieciach FR jest ograniczanie zajętości pamięci buforowych.

Długość kolejek ramek w węzłach sieci FR ograniczają mechanizmy: usuwania ramek oraz dwukierunkowa (wstecz i wprzód) sygnalizacja przeciążeń.

Mechanizm usuwania ramek w sieci FR jest metodą zapobiegania przeciążeniom, powodowanym nadmierną, w porównaniu z uzgodnioną w kontrakcie, szybkością wprowadzania ramek do sieci. Utrzymanie jakości usług sieci FR, polegającej na gwarantowaniu szybkości transmisji CIR (ang. *Committed Information Rate*), wymusza konieczność znakowania ramek generowanych z naruszeniem warunków kontraktu ruchowego. Jeżeli całkowite obciążenie sieci nie grozi powstaniem przeciążenia, nadmiarowy – w porównaniu z warunkami kontraktu – ruch jest obsługiwany na tych samych zasadach, na jakich obsługiwany jest ruch generowany zgodnie z warunkami kontraktu. Zagrożenie przeciążeniem powoduje, że odpowiednio oznakowane ramki nadmiarowe są odrzucane w węzłach sieci.

Mechanizm usuwania ramek, który można interpretować jako przestrzeganie warunków kontraktu i wygładzania charakterystyk ruchu w sieci FR, nie jest wystarczający do zapobiegania przeciążeniom; jest konieczny. Mechanizm usuwania ramek nie zapobiega przeciążeniom dlatego, że zwielokrotnianie ruchu w sieci FR jest nadmiarowym zwielokrotnianiem statystycznym. Jednoczesna realizacja ruchu o gwarantowanej szybkości

CIR oznacza generowanie ruchu sumarycznego przekraczającego pojemność sieci. Taka organizacja jakości usług powoduje konieczność stosowania mechanizmów powiadamiania (wstecz i w przód) o przeciążeniach. Powiadamianie o przeciążeniach odbywa się przez ustawianie odpowiednich bitów sygnalizacji przeciążenia w ramach FR, tzn. bitów BECN (ang. *Backward Explicit Congestion Notification*) i FECN (ang. *Forward Explicit Congestion Notification*). Do dystrybucji informacji o przeciążeniach stosowane są ramki informacyjne (sygnalizacja włożona) lub niezależne ramki sterowania.

Do ustawiania bitów sygnalizacji przeciążenia BECN i FECN w ramach połączeń wirtualnych stosuje się mechanizm progowy. Warunkiem ustawienia bitów sygnalizacji jest przeciążenie kanału. Kanał jest przeciążony, jeżeli przekroczony został zadany poziom zajętości kolejek ramek dla połączeń wirtualnych oczekujących na obsługę.

Bity sygnalizacji przeciążenia wstecz (ang. *backward*) są generowane w celu powiadomienia źródeł strumieni ramek w połączeniach wirtualnych, obsługiwanych przez przeciążony kanał, o konieczności zmniejszenia szybkości generowania ramek. Zniesienie ograniczenia szybkości generowania ramek polega na wstrzymaniu generowania bitów sygnalizacji.

Bity sygnalizacji przeciążenia w przód (ang. *forward*) są generowane w celu powiadomienia ujść strumieni ramek w połączeniach wirtualnych, obsługiwanych przez przeciążony kanał, o konieczności przeciwdziałania następstwom powstałym przeciążeń. Jednym ze sposobów jest np. zwiększenie czasu oczekiwania (zwłoki czasowej) na ramki w połączeniach wirtualnych. Bit sygnalizacji w przód jest ustawiany i dystrybuowany wtedy, gdy w ujściu istnieje możliwość zapobiegania skutkom przeciążeń w sieci.

7.1.3. PRZECIWDZIAŁANIE PRZECIĄŻENIOM W SIECI TCP/IP

Metody przeciwdziałania przeciążeniom oraz sterowania przepływem, implementowane w sieciach TCP/IP, są ściśle związane z przeznaczeniem i właściwościami tych sieci. Architektura TCP/IP zaprojektowano do obsługi ruchu elastycznego, który nie jest wrażliwy na fluktuacje opóźnień, ale jest wrażliwy na straty. Co więcej, cechą architektury TCP/IP jest funkcjonalna nadmiarowość uzasadniona przeznaczeniem do korzystania z zawodnych i względnie wolnych mediów transmisyjnych.

Ze względu na funkcjonalność uzupełniających się mechanizmów trybu połączeniowego w warstwie transportowej i bezpołączeniowego w warstwie sieciowej zadania dwukierunkowej i niezawodnej transmisji są realizowane przez protokół TCP, wspierany przez protokół IP. Z takiego podziału funkcji wynika, że przeciwdziałanie przeciążeniom i sterowanie przepływem jest zadaniem protokołu TCP.

Do sterowania przepływem segmentów w połączeniu transportowym, zgodnie z protokołem TCP, stosowany jest mechanizm kredytowego sterowania przepływem. Mechanizm ten można traktować jako odmianę mechanizmu okna, w której udzielanie kredytu na kolejne transfery jednostek danych nie zależy od otrzymania potwierdzenia poprzednio nadawanych jednostek danych.

Jakość sterowania przepływem segmentów TCP, z punktu widzenia aplikacji, jest wynikiem złożenia jakości kredytowego mechanizmu okna oraz warunków, w jakich się z tego mechanizmu korzysta. Jakość mechanizmu okna zależy od przyjętego zarządzania okna, tzn. parametryzacji mechanizmu okna. Warunki implementacji mechanizmu okna są determinowane procedurami wymiany danych pomiędzy warstwą transportową i warstwami wyższymi (aplikacji) w źródłach i ujściach przepływu TCP oraz wymiany segmentów pomiędzy źródłem i ujściem.

Kredytowy mechanizm okna

Kredytowy mechanizm okna, podobnie jak inne bardziej konserwatywne odmiany mechanizmu okna, spełnia dwie funkcje: zapobiega powstawaniu przeciążeń oraz steruje przepływem w połączeniu organizowanym pomiędzy dwoma systemami końcowymi.

Zarządzanie mechanizmem kredytowym sterowania przepływem dla przepływów TCP, podobnie jak w innych implementacjach tego mechanizmu, polega na odpowiednim doborze wartości parametrów:

- okna nadawczego,
- zwłoki czasowej retransmisji.

Parametryzacja okna nadawczego dotyczy ustalania wartości tych algorytmów i procedur, stosowanych w mechanizmie okna, które zapobiegają ewentualnym przeciążeniom. Są to:

1. Procedury spowolnionego startu SSP (ang. *Slow Start Procedure*). Po nawiązaniu połączenia, transmisja segmentów rozpoczyna się z wykorzystaniem zredukowanego rozmiaru okna, który jest zwiększany o jeden aż do chwili wejścia w stan ustalony.

2. Procedury dynamicznego wymiarowania okna DWSP (ang. *Dynamic Window Sizing Procedure*). Zwiększanie rozmiaru okna, rozpoczynane od zredukowanego jego rozmiaru, może być spowolnione przez wprowadzenie wartości progowej, po przekroczeniu której szybkość wzrostu rozmiaru okna zależy od wartości parametrów połączenia.

3. Procedury szybkiej retransmisji FRP (ang. *Fast Retransmission Procedure*). Żądanie retransmisji segmentu poza kolejnością ma na celu przyśpieszenie komplekcji ciągu segmentów wyprowadzanego w węzle ujściowym do aplikacji.

4. Procedury szybkiego odtwarzania przepustowości FRP (ang. *Fast Recovery Procedure*). Realizacja żądania szybkiej retransmisji oznacza redukcję wcześniejszej przepustowości połączenia, którego odtworzenie nie wymaga rygorów obowiązujących dla spowolnionego startu.

Długość zwłoki czasowej retransmisji (ang. *retransmission time-out*) segmentu jest długością przedziału czasu, po upływie którego następuje retransmisja niepotwierzonego segmentu. Zbyt mała wartość zwłoki czasowej retransmisji może powodować nieuzasadnione zwiększenie liczby retransmitowanych segmentów i przeciążenia połączenia i sieci. Z kolei duża wartość zwłoki czasowej powoduje spowolnienie efektywnej szybkości transferu segmentów pomiędzy systemami końcowymi. Celem różnych propozycji sza-

cowania długości przedziału zwłoki czasowej retransmisji jest uzyskanie gwarancji płynności przepływu segmentów, będącej kompromisem pomiędzy żadaną efektywnością wykorzystania zasobów i oczekiwanym prawdopodobieństwem powstania przeciążenia.

Jakość sterowania przepływem w połączeniu protokołu TCP, z punktu widzenia sieci, jest jednoznacznie określona procedurami i wartościami parametrów procedur stosowanych w mechanizmie okna. Do oceny jakości sterowania przepływem w połączeniu TCP z punktu widzenia aplikacji konieczne jest uwzględnienie wpływu ustalonych procedur wymiany segmentów pomiędzy aplikacją a siecią. Procedury te mogą wносить dodatkowe opóźnienia, kompensowane odpowiednim doбором parametrów okna.

Przyczyną dodatkowego obciążenia sieci są m.in. procedury:

1. Łączenia bloków danych w segmenty protokołu TCP w węzłach źródłowych:

➤ jeden segment odpowiada jednemu blokowi danych; szybkość generowania segmentów jest równa szybkości napływu bloków danych,

➤ jeden segment składa się z kilku bloków danych; łączenie bloków jest źródłem opóźnienia, którego wartość zależy od miejsca zajmowanego w segmencie.

2. Przyjmowania segmentów w węzłach ujściowych:

➤ segmenty są przyjmowane w kolejności nadawania; usuwanie segmentów odbieranych jako poprawne poza kolejnością nadawania powoduje zwiększenie obciążenia sieci oraz uproszczenie procedur obsługi pamięci buforowej węzła i obsługi procesu wyprowadzania segmentów do aplikacji,

➤ segmenty są przyjmowane niezależnie od kolejności nadawania; zwiększa się efektywność wykorzystania zasobów sieci za cenę wzrostu złożoności procedur obsługi pamięci buforowej węzła ujściowego.

3. Wyprowadzania bloków danych z sieci do aplikacji, którego zasady są wymuszone łączeniem lub nie bloków danych w segmenty w węzle źródłowym.

4. Retransmisji bloków danych.

7.1.3.1. PRZECIWDZIAŁANIE PRZECIĄŻENIOM W SIECIACH TCP Z USŁUGAMI BEZ GWARANCJI JAKOŚCI

W tradycyjnych sieciach pakietowych TCP/IP wszystkie pakiety traktowane są tak samo, tzn. bez jakiegokolwiek dyskryminacji lub gwarancji dostarczenia. Taki sposób obsługi pakietów jest obsługą bez gwarancji jakości (ang. *best effort*). Przyrzeczenie obsługi polega na zapewnieniu wszelkich starań w celu dostarczenia wprowadzonych do sieci pakietów bez jakichkolwiek zobowiązań dotyczących ilościowych miar jakości. Model obsługi bez gwarancji jakości nie ma formalnej specyfikacji – jest specyfikowany operacyjnie, tzn. dostarczenie pakietów może być oczekiwane, a nie wykluczone.

Problem przeciążenia

Przeciążenie (ang. *congestion*) jest stanem długotrwałego przeładowania (ang. *sustained overload*) sieci, kiedy zapotrzebowanie na ograniczone jej zasoby jest zbliżone

do pojemności sieci lub ją przekracza. W większości przypadków problem przeciążenia sieci nie może być rozwiązany przez przyjęcie założenia o nieskończoności zasobów sieci ze względu na to, że wartości miar jakości, takich jak opóźnienie pakietów czy zwłoka czasowa retransmisji, nie mogą rosnąć w nieskończoność.

Konsekwencje przeciążenia

Przeciążenie w sieci pakietowej powoduje straty pakietów, zwiększenie opóźnienia i może być przyczyną upadku sieci (ang. *congestion collapse*). Upadek sieci definiuje się jako stan, w którym wzrost obciążenia sieci prowadzi do zmniejszenia przydatności pracy wykonywanej przez sieć. Rezultatem przeciążenia jest stabilny stan sieci, w którym rzeczywista przepustowość sieci jest ułamkiem normalnej. Inne przypadki upadku sieci z powodu przeciążeń są powodowane:

1. Niedostarczaniem pakietów (ang. *undelivered packets congestion collapse*). Pojemność łącza jest marnowana na przenoszenie pakietów, które są odrzucane przed ich dostarczeniem do węzła ujściowego.

2. Transmitowaniem przez sieć fragmentów pakietów (ang. *fragmentation-based congestion collapse*), które są odrzucane przed wykorzystaniem ich do odtworzenia pakietów.

3. Transmitowaniem pakietów nieprzydatnych dla użytkownika (ang. *congestion collapse from stale packets*) z powodu nadmiernego opóźnienia.

Zasięg i lokalizacja mechanizmów przeciwdziałania przeciążeniom

Naturalne rozproszenie zarządzania zasobami w sieci uzasadnia podział mechanizmów przeciwdziałania przeciążeniom ze względu na zasięg oddziaływania decyzji mającej na celu przeciwdziałanie przeciążeniu, tzn. na dotyczące połączeń (ang. *end-to-end*) i dotyczące poszczególnych łączy (ang. *local*). Podział taki jest tożsamy z podziałem mechanizmów przeciwdziałania przeciążeniom ze względu na miejsce ich implementowania: w węźle źródłowym połączenia (ang. *host-based*) i w routerach na trasie (*router-based*).

Mechanizmy przeciwdziałania przeciążeniom połączeń są oparte na względnie prostej zasadzie: detekcja przeciążenia w źródle, możliwa na podstawie informacji zwrotnej (ang. *feedback*) z przeciążonego zasobu sieci, powoduje odpowiedź źródła w postaci spowolnienia szybkości transmisji pakietów. Różne rozwiązania zadań przeciwdziałania przeciążeniom połączeń uzyskuje się w wyniku stosowania różnych kombinacji sposobów pozyskiwania informacji zwrotnej o przeciążeniu (lub symptomach przeciążenia) i sposobów generowania odpowiedzi na pozyskaną z sieci informację zwrotną.

Mechanizmy przeciwdziałania przeciążeniom w routerach dzieli się na dwie grupy ze względu na sposób zarządzania zasobami: szeregowania (ang. *scheduling*) oraz zarządzania pamięcią buforową i kolejkami (ang. *queue/buffer management*).

Fazy przeciwdziałania przeciążeniom

Najprostszym sposobem przeciwdziałania przeciążeniom jest jego unikanie przez utrzymywanie niskiego poziomu obciążenia sieci, co powoduje spadek efektywności

wykorzystania zasobów sieci. Celem każdego mechanizmu przeciwdziałania przeciążeniom jest więc – ze względu na stopień wykorzystania zasobów – operowanie możliwie blisko całkowitej pojemności sieci.

Wyróżnia się dwie podstawowe fazy przeciwdziałania przeciążeniom:

- unikanie przeciążeń (ang. *congestion avoidance*),
- regeneracja przeciążeń (ang. *congestion recovery*); zadanie to jest określane także mianem sterowania przeciążeniem (ang. *congestion control*).

Sposoby przeciwdziałania przeciążeniom połączeń

Przeciwdziałania przeciążeniom połączeń są klasycznymi zadaniami deterministycznego lub dynamicznego sterowania ze sprzężeniem zwrotnym, w których zmiana wejścia układu następuje na podstawie obserwacji wyjścia układu sterowania. Alokacja szybkości transmisji pakietów w źródle ma na celu dopasowanie szybkości transmisji pakietów do szybkości, odpowiadającej sprawiedliwemu współdzieleniu przeciążonego łącza. O wyborze sposobu i jakości sterowania decydują m.in. następujące czynniki:

- postać funkcji decyzyjnej (ang. *decision function*), określającej sposoby interpretacji informacji zwrotnej o przeciążeniu,
- częstotliwość sterowania (ang. *frequency of control*), determinująca częstotliwość podejmowania decyzji o dopasowaniu szybkości transmisji oraz rozmiaru okna,
- postać funkcji sterowania (ang. *control function*), decydującej o sposobie zmiany szybkości transmisji i rozmiaru okna.

Sposoby pozyskiwania informacji zwrotnej o przeciążeniu

Powiadamanie źródła przez routery trasy lub węzły końcowe o przeciążeniu lub symptomach przeciążenia opiera się na mechanizmie sprzężenia zwrotnego. Źródło może otrzymywać informację zwrotną albo bezpośrednio z sieci (ang. *explicit feedback*), albo pośrednio z sieci za pośrednictwem węzła ujściowego (ang. *implicit feedback via receiver*).

Najczęściej stosowanym sposobem bezpośredniego powiadamiania o przeciążeniu jest odrzucanie pakietów (ang. *packet drop*). Informacja zwrotna pośrednia może mieć postać zgłoszenia przeciążenia (ang. *congestion notification*) lub wskazania szybkości transmisji (ang. *rate indication*). Informacja zwrotna pośrednia może być binarna (ang. *binary explicit feedback*) lub wielowartościowa (ang. *multivalued explicit feedback*).

Szeregowanie pakietów

Zadaniem szeregowania w lokalnych mechanizmach przeciwdziałania przeciążeniom jest ustalanie kolejności obsługi (transmisji) pakietów. Dyscyplina szeregowania steruje alokacją pojemności łącza przez wybieranie do obsługi określonej liczby pakietów każdego połączenia w ustalonym przedziale czasu. Szeregowanie jest jedynym efektywnym sposobem zarządzania zasobami transmisyjnymi i najczęściej implementuje się wraz z różnymi strategiami zarządzania pamięcią buforową lub zarządzania kolejkami.

Wybór dyscypliny szeregowania pakietów, implementowanej w mechanizmie przeciwdziałania przeciążeniom, zależy od kilku czynników, w tym m.in. od:

- liczby poziomów priorytetów (ang. *priority levels*), którym odpowiada liczba oddzielnych kolejek pakietów w routerze,
- kolejności obsługi pakietów o tym samym poziomie priorytetu, zwłaszcza wtedy, gdy dana wartość priorytetu odpowiada przepływowi zagregowanemu,
- stopnia agregacji przepływu (ang. *degree of flow aggregation*) na jednym poziomie priorytetu, decydującego o tym, czy dyscyplina szeregowania rozróżnia pakiety przepływów składowych (ang. *per-flow scheduling*), czy nie rozróżnia (ang. *per-class scheduling*),
- rodzaju dyscypliny szeregowania z punktu widzenia zależności pomiędzy zajętością kolejki i zajętością serwera pakietów w kolejce, tzn. od tego, czy serwer kolejki zachowuje pracę (ang. *work-conserving*) czy nie (ang. *non-work-conserving*),
- cech dyscypliny szeregowania, w szczególności sprawiedliwości (ang. *fairness*) szeregowania.

Zarządzanie pamięcią buforową i kolejkami

Celem metod zarządzania pojemnością pamięci buforowej jest podział dostępnej puli buforów pomiędzy przepływy współzawodniczące o ten zasób sieci. Opracowano i wdrożono wiele różnych strategii podziału pamięci buforowych charakteryzujących się różną dynamiką alokacji, różnymi kryteriami efektywności alokacji i zajętości pamięci, granulacją przepływów itd. Podstawowym kryterium podziału metod alokacji pojemności pamięci buforowej jest sposób udostępniania pamięci poszczególnym przepływowi: współdzielenie całkowitej pojemności przez wszystkie przepływy (ang. *shared buffer pool*) lub dedykowanie pojemności dla przepływów (ang. *per-flow allocation*).

Celem metod zarządzania kolejkami jest sterowanie długością kolejek i przepływami, które kolejkę zajmują. Sterowanie to polega na wyborze pakietów do odrzucenia i czasu ich odrzucania. Metody zarządzania kolejkami są komplementarne dla dyscyplin szeregowania (decydujących o kolejności obsługi pakietów oczekujących w kolejkach) i metod zarządzania pamięcią buforową (decydujących o liczbie organizowanych kolejek i agregacji przepływów w kolejkach). Kryterium ogólnego podziału metod zarządzania jest zdolność do zapobiegania przeciążeniom lub regeneracji przeciążenia; z tego względu wyróżnia się dwie obszerne grupy metod zarządzania kolejkami:

- reaktywne, tzn. regenerujące przeciążenia QMCR (ang. *Queue Management for Congestion Recovery*),
- prewencyjne, tzn. zapobiegające powstawaniu przeciążeń AQMCA (ang. *Active Queue Management for Congestion Avoidance*).

Najczęściej stosowane rozwiązania z grupy metod reaktywnych QMCR polegają na odrzucaniu pakietów w razie przepełnienia kolejki. Różne przykładowe odmiany metod odrzucania pakietów to:

- odrzucanie „ogonów” (ang. *tail drop*) – przepełnienie kolejki powoduje odrzucanie pakietów w kolejności ich napływania do routera,
- odrzucanie z czoła kolejki (ang. *drop-from-front*) – przepełnienie kolejki powoduje odrzucanie pakietów znajdujących się na czele kolejki,
- odrzucanie losowe (ang. *random drop*) – nadejście nowego pakietu i przepełnienie kolejki uruchamia procedurę losowania i odrzucania wylosowanego pakietu.

Metody prewencyjne z grupy AQMCA to tzw. aktywne metody zarządzania kolejkami AQM (ang. *Active Queue Management*). Cechą charakterystyczną tej grupy metod jest to, że – w przeciwieństwie do metod grupy QMCR – odrzucanie pakietów nie jest jedynym sposobem powiadamiania źródła o przeciążeniu lub wystąpieniu symptomów przeciążenia; funkcje takie może pełnić znakowanie pakietów.

Do zadań metod AQMCA, oprócz zapobiegania przeciążeniom, zalicza się także: eliminację skutków „wybuchowości” źródeł, gwarancję zachowania górnych ograniczeń długości kolejek, nawet w przypadku obecności tzw. niewspółpracujących przepływów (ang. *noncooperative flows*), karanie (ang. *penalize*) tzw. zachłanych przepływów (ang. *aggressive flows*), redukcję liczby odrzucanych pakietów, gwarancję ograniczonych z góry opóźnień pakietów.

W metodach AQM powszechnie stosowanym mechanizmem detekcji przeciążenia lub jego symptomów jest znakowanie pakietów. Właściwości poszczególnych metod zależą od stosowanej miary przeciążenia i funkcji znakowania pakietów, których parametrami są wartości miar przeciążenia.

Metody AQMCM są podstawowymi rozwiązaniami zarządzania pamięcią w sieciach z różnicowaniem usług.

Obsługa bez gwarancji jakości i różnicowanie jakości usług w połączeniach TCP

Mechanizmy przeciwdziałania przeciążeniom, stosowane w sieciach bez gwarancji jakości usług, mogą służyć do realizacji usług bez gwarancji jakości o względnie zróżnicowanej jakości RDS (ang. *Relative Differentiated Services*). Oznacza to możliwość jednoczesnego powoływania i utrzymywania w sieci bez gwarancji jakości takich klas ruchu, których charakterystyki jakości usług są jakościowo różne. Istnienie różnych jakościowo klas ruchu w sieci bez gwarancji jakości nie oznacza istnienia jakichkolwiek gwarancji ilościowych jakości usług.

7.1.3.2. KONCEPCJE PRZECIWDZIAŁANIA PRZECIĄŻENIOM W SIECIACH TCP/IP

W sieciach z komutacją pakietów różne zasoby sieci są współdzielone przez wielu użytkowników. Wspólnymi zasobami są m.in. pojemności łączy, pojemności pamięci buforowych w węzłach ujściowych, pamięci buforowe węzłów pośrednich (tranzytowych). Stopień wykorzystania zasobów jest determinowany przede wszystkim szybkością, z jaką węzeł źródłowy transmituje pakiety do sieci. Z tego powodu o współdzieleniu zasobów sieci, tzn. o ilościach zasobów wykorzystywanych przez

poszczególne źródła decyduje implementowany w sieci algorytm wyznaczania (alokacji) szybkości transmisji pakietów. Algorytm alokacji szybkości transmisji pakietów jest jednocześnie algorytmem przeciwdziałania przeciążeniom. Odpowiedzią na wykrycie stanu przeciążenia jest zmiana (zmniejszenie) szybkości transmisji pakietów w połączeniach korzystających z przeciążonych zasobów sieci.

Z analizy działania protokołu TCP wynika, że nie jest on wyposażony w efektywne mechanizmy zabezpieczające przed przeciążeniem sieci. Informacja o ewentualnych przeciążeniach zasobów wykorzystywanych przez połączenia TCP jest dostępna pośrednio przez obserwację czasu oczekiwania na potwierdzenia odbioru segmentu, upływu zwłoki czasowej retransmisji czy wzrostu liczby duplikatów potwierdzeń.

W sieciach TCP do wyznaczania szybkości transmisji pakietów stosuje się rozproszone algorytmy, których zadaniem jest wyznaczanie szybkości transmisji pakietów na podstawie znajomości informacji zwrotnej w postaci powiadomień przekazywanych przez ujścia do źródeł. Informacja zwrotna jest dostępna w postaci powiadomień pozytywnych (transmisja zakończona powodzeniem) i powiadomień negatywnych (transmisja zakończona stratą pakietu). Ta szczególna forma pośredniej informacji zwrotnej o ewentualnym przeciążeniu nie wymaga żadnego wsparcia ze strony sieci, oprócz przekazywania pakietów ze źródła do ujścia i powiadomień z ujścia do źródła. W przypadku korzystania z alternatywnych form informacji zwrotnej o przeciążeniach, tzn. z bezpośredniej sygnalizacji wystąpienia przeciążenia ECN (ang. *Explicit Congestion Notification*), niezbędne jest dodatkowe wsparcie ze strony sieci za cenę dodatkowych korzyści.

Dopuszczenie w sieci TCP możliwości różnych reakcji źródeł ruchu na informacje zwrotne o stanie zasobów, generowane przez użytkownika końcowego w postaci powiadomień (pośrednia informacja zwrotna) lub przez sieć w postaci znakowania pakietów przez przeciążone zasoby (bezpośrednia informacja zwrotna), pozwala na różnicowanie jakości usług. Z tego punktu widzenia różnicowanie jakości usług w wyniku różnych reakcji źródeł ruchu na otrzymaną informację zwrotną może być interpretowane jako wynik działania algorytmu ważonej sprawiedliwej alokacji zasobów.

Wynikiem działania algorytmów przeciwdziałania przeciążeniom w połączeniach TCP jest alokacja szybkości transmisji: szybkość transmisji zwiększa się liniowo aż do chwili wykrycia straty pakietu w połączeniu TCP, kiedy to szybkość transmisji zmniejsza się dwukrotnie. Zadaniem algorytmu przeciwdziałania przeciążeniom jest osiągnięcie stanu równowagi pomiędzy stopniem wykorzystania zasobów sieci i poziomem strat pakietów.

Istnieje wiele różnych algorytmów przeciwdziałania przeciążeniom, których wynikiem działania jest alokacja szybkości transmisji pakietów w połączeniach TCP.

Jednym z prostszych modeli, charakteryzujących stan ustalony przepustowości połączenia TCP dla ustalonej wartości dopuszczalnego prawdopodobieństwa strat pakietów p , jest zależność określająca szybkość transmisji pakietów v w pojedynczym połączeniu:

$$v = \frac{1}{rtt} \sqrt{\frac{2}{p}},$$

gdzie:

rtt – wartość opóźnienia potwierdzenia pakietu (ang. *round-trip time*),

p – prawdopodobieństwo straty pakietu w połączeniu na trasie od źródła do ujścia.

Analiza tego wyrażenia wykazuje silną zależność szybkości transmisji pakietów v od opóźnienia potwierdzenia pakietu rtt . Dla ustalonej wartości prawdopodobieństwa strat pakietów p zwiększenie wartości rtt powoduje zmniejszenie szybkości transmisji v . Można wykazać, że wzór określający szybkość transmisji pakietów maksymalizuje wartość wskaźnika jakości (funkcji celu) w postaci:

$$Q(v) = g(v) - pv,$$

gdzie:

$$g(v) = \kappa - 2/(rtt^2v),$$

κ – dowolna stała.

Algorytm przeciwdziałania przeciążeniom alokujący szybkość transmisji w połączeniu TCP zgodnie z podanym wyrażeniem można interpretować jako wyznaczający szybkość transmisji optymalizującą współczynnik wykorzystania sieci netto, którego wartość jest różnicą wartości funkcji $g(v)$ i kosztu pv , odnoszącego się do szybkości strat transmitowanych pakietów. Wartość funkcji $g(v)$ jest kosztem uzyskania większej pojemności, która jest następnie pomniejszana o koszt proporcjonalny do szybkości straty pakietów.

W różnych wariantach algorytmów przeciwdziałania przeciążeniom mogą być stosowane alternatywne postacie funkcji $g(v)$, z tym że ogólna ich interpretacja jest zawsze podobna do podanej wcześniej. Wynika to z tego, że przeciwdziałanie przeciążeniom zasobów polega na redukcji szybkości transmisji pakietów, która jest proporcjonalna do początkowej szybkości transmisji. Koszt większej początkowej szybkości transmisji jest większy, ale także koszt straty większej liczby pakietów jest większy.

Wynikiem działania omawianego sposobu alokacji szybkości transmisji pakietów w połączeniach TCP, będącego jednocześnie algorytmem przeciwdziałania przeciążeniom zasobów wykorzystywanych przez te połączenia, jest to, że podobne szybkości transmisji pakietów są przydzielane połączeniom korzystającym z zasobów o zbliżonym poziomie przeciążenia oraz takich samych opóźnieniach potwierdzenia pakietu i takich samych dopuszczalnych prawdopodobieństwach strat. Różnicowanie jakości usług przez alokację różnych szybkości transmisji jest możliwe wtedy, gdy dopuszczalne prawdopodobieństwo strat pakietów jest różne dla połączeń o takich samych wartościach opóźnienia potwierdzenia pakietu.

Współdzielenie zasobów przez wiele połączeń (przepływów) oznacza, że alokacja większej szybkości transmisji pakietów dla jednego z połączeń nie tylko zwiększa negatywne skutki wystąpienia określonego poziomu przeciążenia dla tego połączenia,

ale także wpływa na poziom przeciążenia zasobów postrzegany przez pozostałe połączenia współdzielące zasoby. Każde z połączeń współdzielących zasoby ma zarówno swój wkład w powstawanie przeciążenia, jak i ponosi konsekwencje powstania przeciążenia. Optymalizacja podanej funkcji kosztu $Q(v)$ pozwala więc na równoważenie korzyści połączenia wynikających z wyboru większej szybkości transmisji pakietów oraz wszystkich konsekwencji takiego wyboru dla wszystkich współużytkowników zasobów sieci. Wartość pv można interpretować jako cenę przeciążenia (ang. *congestion price*) – im wyższy jest poziom przeciążenia zasobów, tym większa jest cena przeciążenia pv i silniejszy bodziec do obniżania szybkości transmisji pakietów.

Przedstawiona analiza kosztów alokacji szybkości transmisji pakietów oraz kosztów strat pakietów może być stosowana w algorytmie przeciwdziałania przeciążeniom przez alokację szybkości transmisji z wykorzystaniem bezpośredniej informacji zwrotnej o przeciążeniu, tj. szybkości znakowania pakietów, która jest miarą przeciążenia zasobów.

Załóżmy, że zadaniem i -tego użytkownika (i -te połączenie) jest ustalenie takiej szybkości transmisji pakietów v_i , dla której dopuszczalna szybkość ich znakowania wynosi z_i (znakowanych pakietów na sekundę). Wartość z_i można interpretować jako zdolność i -tego użytkownika do ponoszenia kosztów proporcjonalnych do szybkości znakowania pakietów, wyznaczającą np. dopuszczalny poziom ceny przeciążenia. Można dalej założyć, że w środowisku rozproszonym każdy i -ty użytkownik ustala swoją szybkość transmisji pakietów v_i tak, aby dopasować swoją zdolność do ponoszenia kosztów za szybkość znakowania pakietów z_i zgodnie np. z następującym algorytmem adaptacji szybkości transmisji:

$$\frac{dv_i(t)}{dt} = \zeta_i (z_i - v_i(t)\eta_i(t)),$$

$$\eta_i(t) = \sum_{k \in R_i} \vartheta_k(u_k),$$

gdzie:

- u_k – sumaryczne obciążenie k -tego zasobu wykorzystywanego przez i -tego użytkownika (np. długość kolejki pakietów oczekujących na obsługę),
- R_i – zbiór indeksów zasobów wykorzystywanych przez i -tego użytkownika,
- $\vartheta_k(u_k)$ – wartość funkcji znakowania pakietów przez k -ty zasób, która zależy od sumarycznego przeciążenia k -tego zasobu ($0 \leq \vartheta_k(u_k) \leq 1$); wartość funkcji znakowania określa udział liczby znakowanych pakietów do wszystkich transmitowanych przez k -ty zasób,
- $\eta_i(t)$ – znormalizowana suma wartości funkcji znakowania pakietów przez zasoby wykorzystywane przez i -te połączenie ($0 \leq \eta_i(t) \leq 1$),
- $v_i(t)\eta_i(t)$ – rzeczywista szybkość znakowania,
- ζ_i – stała liczba dodatnia – współczynnik wzmocnienia, od którego wartości zależy szybkość adaptacji i stabilność procesu adaptacji.

Informacja zwrotna o przeciążeniu jest dostępna dla i -tego użytkownika w postaci szybkości znakowania pakietów $v_i(t)\eta_i(t)$. Wartość różnicy dopuszczalnej i rzeczywistej szybkości znakowania pakietów pozwala na stosowanie algorytmu wyznaczenia szybkości transmisji pakietów v_i . Działanie takiego algorytmu zależy od postaci funkcji znakowania $\mathfrak{G}_k(u_k)$, stosowanej przez poszczególne zasoby do detekcji poziomu przeciążenia zasobów.

W przypadku zastosowania omawianego algorytmu alokacji szybkości transmisji pakietów przez wszystkich użytkowników, proces alokacji jest zbieżny z punktem równowagi, w którym wartości szybkości transmisji pakietów określa zależność:

$$v_i(t) = \frac{z_i}{\eta_i(t)}.$$

Cechą charakterystyczną tej zależności jest to, że szybkość transmisji pakietów nie zależy od opóźnienia ich retransmisji. Zależność ta jest prawdziwa dla stałej dopuszczalnej wartości szybkości znakowania pakietów. Zmiana tej wartości przez użytkownika daje możliwość dopasowania szybkości transmisji pakietów do jego preferencji.

7.1.3.3. AKTYWNE ZARZĄDZANIE KOLEJKAMI

Jednym z rozwiązań zwiększających możliwość detekcji oraz przeciwdziałania przeciążeniom w sieciach TCP – w porównaniu z metodami przeciwdziałania przeciążeniom opartych na informacji o przeciążeniach dostępnej pośrednio – jest stosowanie mechanizmów aktywnego zarządzania kolejkami AQM (ang. *Active Queue Management*), takich jak np.: losowego odrzucania RD (ang. *Random Drop*), odrzucania ogona DT (ang. *Drop Tail*), wczesnego losowego odrzucania ERD (ang. *Early Random Drop*), wczesnej losowej detekcji RED (ang. *Random Early Detection*) oraz losowego wykładniczego znakowania REM (ang. *Random Exponential Marking*).

Podstawowym celem aktywnego zarządzania kolejkami AQM jest dostarczenie do źródeł przepływów informacji o przeciążeniu zasobów, która jest następnie stosowana do wyznaczenia aktualnych szybkości transmisji pakietów.

Cechami wspólnymi mechanizmów aktywnego zarządzania kolejkami są:

- stosowanie bezpośredniej sygnalizacji wystąpienia przeciążenia ECN do celów wyprzedzającego informowania o narastającym przeciążeniu,

- różne postaci funkcji progowej (ang. *threshold function*) w algorytmach znakowania (ang. *marking algorithm*) pakietów.

Projektowanie mechanizmów (algorytmów) aktywnego zarządzania kolejkami, w których znakowanie pakietów jest losowe (probabilistyczne), wymaga ustalenia odpowiedzi na następujące pytania:

- jaka jest miara przeciążenia? – z definicji miary przeciążenia wynikają sposoby jej aktualizacji,

- jak miara przeciążenia jest implementowana w funkcji losowej generującej prawdopodobieństwa, tzn. w funkcji znakowania pakietów?

➤ jak informacja o znakowaniu pakietów jest dostarczana do źródeł przepływów?

Omawiane dalej mechanizmy detekcji przeciążenia RED i REM są charakterystyczne dla wielu różnych mechanizmów przeciwdziałania przeciążeniom, których zadaniem jest wczesna detekcja przeciążenia i zapobieganie przeciążeniom przez odpowiednią alokację szybkości transmisji pakietów w połączeniach.

7.1.3.3.1. MECHANIZM WCZESNEJ LOSOWEJ DETEKCJI PRZECIĄŻENIA RED

Działanie mechanizmu RED opiera się na następujących założeniach:

➤ miarą przeciążenia jest średnia długość kolejki pakietów,
➤ funkcja losowa znakowania pakietów jest przedziałami liniową i niemalejącą funkcją miary przeciążenia,

➤ informacja o przeciążeniu jest dostarczana do źródła przez losowe odrzucanie pakietów lub ustawianie bitów ECN w nagłówkach pakietów.

Mechanizm RED wywodzi się z mechanizmu ERD, którego ogólna idea polega na tym, że na potrzeby detekcji wczesnego etapu powstawania przeciążenia monitorowana jest średnia długość kolejki. W razie wykrycia przeciążenia, tzn. stwierdzenia przekroczenia ustalonej wartości progowej odrzucania (ang. *drop level*), rozpoczyna się losowe odrzucanie pakietów zgodnie z ustaloną wartością prawdopodobieństwa odrzucania (ang. *drop probability*). Celem zmian oryginalnego mechanizmu ERD była eliminacja ograniczeń mechanizmu powodowana stałą wartością progową odrzucania i stałą wartością prawdopodobieństwa odrzucania.

Zasady działania mechanizmu RED są bardzo ogólne i dlatego mechanizm ten może być stosowany do sterowania średnią długością kolejek w sieciach, w których protokół transportowy ma zdolność reagowania na sygnalizację przeciążenia pochodzącą od sieci. Działanie mechanizmu RED można interpretować jako działanie filtra dolnoprzepustowego, którego zadaniem jest przenoszenie średnich długości kolejki i dławienie „wybuchów” długości kolejki. Wejściem filtra jest chwilowa długość kolejki, a jego wyjściem – estymata średniej długości kolejki. Cechą mechanizmu RED jest jego zorientowanie na sieci z takimi protokołami, w których oznakowanie lub odrzucenie jednego z pakietów jest wystarczające do sygnalizacji przeciążenia.

Zadaniem mechanizmu sterowania przeciążeniami RED jest monitorowanie średniej długości każdej wyjściowej kolejki w węźle oraz, w razie wystąpienia symptomów przeciążenia w węźle, losowe wybieranie połączeń powiadamianych o przeciążeniu. Przejściowe przeciążenia (ang. *transient congestion*) są rozwiązywane przez chwilowe wydłużanie długości kolejek; wystąpienie trwałych przeciążeń (ang. *longer-lived congestion*), powodujące wzrost wyliczanych średnich długości kolejek, skutkuje natomiast losowym powiadamianiem połączeń o konieczności zmniejszenia rozmiaru okien. Prawdopodobieństwo, że dane połączenie jest powiadamiane o przeciążeniu jest proporcjonalne do udziału danego połączenia w całkowitej przepustowości węzła.

Detekcja powstawania przeciążenia przed przepełnieniem kolejki nie ogranicza się do odrzucania pakietu jako metody powiadamiania połączenia o przeciążeniu. Możli-

we jest, w zależności od protokołu transportowego, albo znakowanie (ang. *mark*) pakietu przez jego odrzucenie (ang. *drop*), albo znakowanie przez ustawienie (ang. *setting*) odpowiedniego bitu w nagłówku pakietu. Jeżeli średnia długość kolejki przekroczy ustaloną, maksymalną wartość progową (ang. *threshold*), to znakowane są wszystkie nadchodzące pakiety. Jeżeli znakowanie pakietów odbywa się przez ich odrzucanie, a nie przez ustawianie bitu nagłówka, to możliwe jest sterowanie średnią długością kolejki nawet wtedy, gdy protokół transportowy nie jest wyposażony w mechanizm reagowania na znakowanie pakietów.

W mechanizmie RED, w odróżnieniu od mechanizmu ERD, średnie długości kolejek są mierzone, działanie mechanizmu nie jest ograniczone wyłącznie do odrzucania pakietów oraz prawdopodobieństwo odrzucania jest funkcją średniej długości kolejki.

Zasady działania mechanizmu RED są następujące:

Funkcja znakowania pakietów

Gdy w systemie transmisyjnym znajduje się jeden przeciążony węzeł (router) o wyjściowej pojemności C i całkowitej pojemności pamięci buforowej B , wówczas implementacja mechanizmu RED jest charakteryzowana funkcją znakowania (odrzućcia) pakietów (ang. *packet marking/discard function*) $\mathfrak{G}(l_{\text{avg}})$, której argumentem jest estymata średniej długości kolejki l_{avg} :

$$\mathfrak{G}(l_{\text{avg}}) = \begin{cases} 0 & \text{gdy } 0 \leq l_{\text{avg}} \leq l_{\text{min}} \\ \frac{l_{\text{avg}} - l_{\text{min}}}{l_{\text{max}} - l_{\text{min}}} p_{\text{max}} & \text{gdy } l_{\text{min}} \leq l_{\text{avg}} \leq l_{\text{max}} \\ 1 & \text{gdy } l_{\text{max}} \leq l_{\text{avg}} \leq B \end{cases}$$

gdzie:

l_{min} – długość kolejki, której osiągnięcie uaktywnia mechanizm detekcji przeciążenia (bezpośredniego powiadamiania o możliwości powstania przeciążenia),

l_{max} – długość kolejki, której osiągnięcie jest tożsame z powstaniem przeciążenia,

p_{max} – wartość prawdopodobieństwa,

są parametrami mechanizmu RED implementowanego w węźle (rysunek 7.2a). Wartość $\mathfrak{G}(l_{\text{avg}})$ spełnia warunek: $0 \leq \mathfrak{G}(l_{\text{avg}}) \leq 1$.

Dla każdego pakietu nadchodzącego do węzła wyznacza się estymatę średniej długości kolejki l_{avg} , dla której jest wyznaczana wartość funkcji znakowania pakietów $\mathfrak{G}(l_{\text{avg}})$ ($\mathfrak{G}(l_{\text{avg}})$ jest prawdopodobieństwem). Jeżeli:

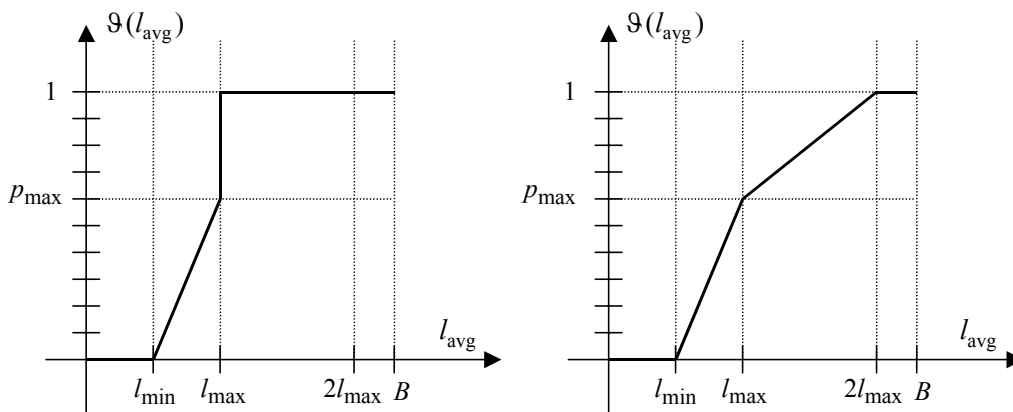
➤ spełniony jest warunek: $l_{\text{min}} \leq l_{\text{avg}} \leq l_{\text{max}}$, to nadchodzące pakiety są znakowane z prawdopodobieństwem $\mathfrak{G}(l_{\text{avg}})$,

➤ w przeciwnym razie, tzn. gdy spełniony jest warunek: $l_{\text{avg}} \geq l_{\text{max}}$, wszystkie nadchodzące pakiety są znakowane.

Jedną z możliwych wersji mechanizmu RED jest mechanizm G-RED (ang. *gentle-RED*), dla którego funkcja znakowania (odrzućcia) pakietów ma następującą postać:

$$\mathfrak{P}(l_{\text{avg}}) = \begin{cases} 0 & \text{gdy } 0 \leq l_{\text{avg}} \leq l_{\text{min}} \\ \frac{l_{\text{avg}} - l_{\text{min}}}{l_{\text{max}} - l_{\text{min}}} p_{\text{max}} & \text{gdy } l_{\text{min}} \leq l_{\text{avg}} \leq l_{\text{max}} \\ p_{\text{max}} + \frac{l_{\text{avg}} - l_{\text{max}}}{l_{\text{max}}} (1 - p_{\text{max}}) & \text{gdy } l_{\text{max}} \leq l_{\text{avg}} \leq 2l_{\text{max}} \\ 1 & \text{gdy } l_{\text{max}} \leq l_{\text{avg}} \leq B \end{cases},$$

gdzie: l_{min} , l_{max} , ($l_{\text{max}} < 1/2B$) i p_{max} są parametrami mechanizmu G-RED (rysunek 7.2b).



a) funkcja znakowania mechanizmu RED

b) funkcja znakowania mechanizmu G-RED

Rysunek 7.2. Wykresy funkcji znakowania (odrzućcia) pakietów w mechanizmach RED i G-RED

Podane postacie funkcji znakowania pakietów są prawdziwe przy założeniu, że zarządzanie ruchem w węźle spełnia następujące warunki:

- prawdopodobieństwo nadejścia pakietu wtedy, gdy kolejka jest całkowicie zajęta, jest równe zero,
- prawdopodobieństwo, że węzeł nie transmituje pakietów podczas przeciążenia jest równe zero.

Model mechanizmu RED

Do opisu działania mechanizmu RED dla jednego przeciążonego routera można zastosować m.in. model przepływowy, zgodnie z którym dynamika zmian rozmiaru

okien poszczególnych przepływów TCP zależy od zmian stanu kolejki w węźle oraz prawdopodobieństwa znakowania (odrzućcia) pakietów.

Model przepływowy węzła sieci z implementacją mechanizmu RED, stosowany do opisu dynamiki zmian w węźle obsługującym przepływy TCP, jest następujący:

➤ każdy i -ty ($i = 1, 2, \dots, N$) przepływ TCP, obsługiwany przez router sieci, jest charakteryzowany w chwili t ($t \geq 0$) wartościami: rozmiaru okna $w_i(t)$ oraz opóźnieniem potwierdzenia pakietu (segmentu) $rtt_i(t)$,

➤ opóźnienie potwierdzenia pakietu jest określone wzorem:

$$rtt_i(t) = rtt_i(q(t)) = \alpha_i + \frac{q(t)}{C},$$

w którym:

α_i – stałe opóźnienie propagacji,

$q(t)/C$ – czas oczekiwania pakietu w kolejce,

➤ $B_i(t) = w_i(t)/rtt_i(t)$ jest chwilową przepustowością i -tego ($i = 1, 2, \dots, N$) przepływu TCP,

➤ proces strat pakietów i -tego ($i = 1, 2, \dots, N$) przepływu TCP jest procesem Poissona $\{n_i(t)\}$ o zmiennej w czasie intensywności napływu $\lambda_i(t)$, gdzie $n_i(t)$ jest liczbą pakietów odrzućonych lub oznakowanych w przedziale czasu $\{0, t\}$.

Zachodzące w omawianym systemie obsługi przepływów zmiany są charakteryzowane zmianami: rozmiarów okien N przepływów, długości kolejki i prawdopodobieństwa odrzućcia (znakowania) pakietów.

Rozmiary okien

Zmiany rozmiaru okna $w_i(t)$ i -tego ($i = 1, 2, \dots, N$) przepływu TCP wyraża równanie:

$$dw_i(t) = \frac{dt}{rtt_i(q(t))} - \frac{w_i(t)}{2} dn_i(t) \text{ dla } i = 1, 2, \dots, N.$$

określające tzw. addytywno rosnące i multiplikatywno malejące zmiany rozmiaru okna. Rozmiar okna zwiększa się z szybkością odwrotnie proporcjonalną do zwłoki czasowej retransmisji i zmniejsza się dwukrotnie w przypadku otrzymania informacji o odrzućciu pakietu ($dn_i(t) = 1$).

Wartości oczekiwane obydwu stron powyższego równania prowadzą do wzoru:

$$E[dw_i(t)] = dE[w_i(t)] = E\left[\frac{dt}{rtt_i(q(t))}\right] - \frac{E[w_i(t)dn_i(t)]}{2} \approx E\left[\frac{dt}{rtt_i(t)}\right] - \frac{E[w_i(t)]}{2} \lambda_i(t) dt,$$

w którym $\lambda_i(t)$ jest intensywnością powiadomienia o odrzućciach źródła i -tego ($i = 1, 2, \dots, N$) przepływu.

Po przyjęciu, że:

➤ powiadomienie o oznakowaniu (odrzućciu) pakietu jest odbierane w źródle przepływu po upływie czasu τ_i od chwili odrzućcia (oznaczenia) pakietu (τ_i jest rozwiązaniem układu dwóch równań: $t = rrt_i(q(t')) + t'$ oraz $t' = t - \tau_i$),

➤ jeżeli przepustowość dla i -tego przepływu w chwili $t - \tau_i$ jest równa $B_i(t - \tau_i)$, to wartość intensywności powiadamiania o odrzucaniu $\lambda_i(t)$ w chwili t jest określona równaniem:

$$\lambda_i(t) = \mathfrak{G}(\bar{l}_{\text{avg}}(t - \tau_i)) B_i(t - \tau_i) = \mathfrak{G}(\bar{l}_{\text{avg}}(t - \tau_i)) \frac{\bar{w}_i(t - \tau_i)}{rrt_i(\bar{q}(t - \tau_i))},$$

w którym:

$$\bar{l}_{\text{avg}}(t - \tau_i) = E[l_{\text{avg}}(t - \tau_i)],$$

$$\bar{w}_i(t - \tau_i) = E[w_i(t - \tau_i)],$$

$$\bar{q}(t - \tau_i) = E[q(t - \tau_i)],$$

to wartość oczekiwana rozmiaru okna jest określona następującym wzorem:

$$\begin{aligned} \frac{d\bar{w}_i(t)}{dt} &\approx \frac{dt}{rrt_i(\bar{q}(t - \tau_i))} - \frac{\bar{w}_i(t)}{2} \mathfrak{G}(\bar{l}_{\text{avg}}(t - \tau_i)) \frac{\bar{w}_i(t - \tau_i)}{rrt_i(\bar{q}(t - \tau_i))} dt = \\ &= \frac{dt}{rrt_i(\bar{q}(t - \tau_i))} - \frac{\bar{w}_i(t)\bar{w}_i(t - \tau_i)}{2rrt_i(\bar{q}(t - \tau_i))} \mathfrak{G}(\bar{l}_{\text{avg}}(t - \tau_i)). \end{aligned}$$

Długość kolejki

Zmiany wartości oczekiwanej długości kolejki $\bar{q}(t)$ w routerze określa się według wzoru:

$$\frac{d\bar{q}(t)}{dt} = -C + \sum_{i=1}^N E\left[\frac{w_i(t)}{rrt_i(t)}\right] \approx -C + \sum_{i=1}^N \frac{\bar{w}_i(t)}{rrt_i(t)},$$

w którym pierwszy składnik prawej strony równania jest szybkością zmniejszania się kolejki powodowanej obsługą kolejki przez łącze o pojemności C , a drugi składnik jest wartością oczekiwaną szybkości wzrostu długości kolejki powodowanej napływem kolejnych pakietów.

Zmiany wartości estymaty średniej długości kolejki są opisywane przy założeniu, że wartość estymaty jest średnią ważoną wartości próbek pobieranych co δ jednostek czasu, daną wyrażeniem:

$$l_{\text{avg}}((k+1)\delta) = (1 - \beta) l_{\text{avg}}(k\delta) + \beta q(k\delta),$$

gdzie β ($0 < \beta < 1$) jest współczynnikiem wagowym,

lub równoważnym układem równań:

$$\frac{dl_{\text{avg}}}{dt} = Al_{\text{avg}}(t) + Bq(t),$$

$$l_{\text{avg}}(t_{k+1}) = e^{A(t_{k+1}-t_k)} l_{\text{avg}}(t_k) + q(t_k) B \int_{t_k}^{t_{k+1}} e^{A(t_{k+1}-\tau)} d\tau,$$

w których: $A = -B = \ln(1 - \beta)/\delta$.

Średnia długość kolejki

W konsekwencji podanych wyrażen i zależności, zmiany wartości średniej długości kolejki l_{avg} , tj. parametru funkcji znakowania (odrzućania) pakietów, są okrećone równaniem:

$$\frac{dl_{\text{avg}}(t)}{dt} = \frac{\ln(1-\beta)}{\delta} (l_{\text{avg}}(t) - q(t))$$

lub wyrażeniem z wykorzystaniem wartości oczekiwanych:

$$\frac{d\bar{l}_{\text{avg}}(t)}{dt} = \frac{\ln(1-\beta)}{\delta} (\bar{l}_{\text{avg}}(t) - \bar{q}(t)).$$

Układ $N + 1$ równań, tzn.: N równań $d\bar{w}_i(t)/dt$ dla $i = 1, 2, \dots, N$, równań $d\bar{q}(t)/dt$ oraz $d\bar{l}_{\text{avg}}(t)/dt$ pozwala na wyznaczenie wartości $N + 2$ zmiennych $(\bar{w}_1, \bar{w}_2, \dots, \bar{w}_N, \bar{q}, \bar{l}_{\text{avg}})$, których wartości z kolei mogą być wykorzystane do wyznaczenia wartości zwłoki czasowej retransmisji, średniej szybkości odrzućania (znakowania) oraz innych parametrów systemu.

Omawiany model przeciążenia pojedynczego węzła sieci może być rozszerzony i zastosowany do modelowania i analizy systemów obsługi złożonych z wielu węzłów (routerów) o różnej strukturze topologicznej.

Ze szczegółowych analiz i badań symulacyjnych różnych wariantów mechanizmu RED wynika, że mechanizm ten charakteryzują dwa podstawowe ograniczenia:

1. Stabilność systemu. Zwiększenie szybkości reakcji mechanizmu na przeciążenia związane jest z ograniczeniem obszaru stabilności.

2. Zależność stanu ustalonego kolejki od obciążenia systemu. Konsekwencją przepełnienia systemu jest podwójne „karanie” przepływów, zarówno zwiększeniem opóźnienia, jak i zwiększeniem liczby odrzućanych pakietów.

Wynikiem analizy właściwości i ograniczeń różnych wersji mechanizmu RED są inne propozycje sterowników realizujących funkcje aktywnego zarządzania kolejkami AQM, m.in. dla przepływów TCP, w tym m.in. mechanizm REM (ang. *Random Exponential Marking*).

7.1.3.3.2. MECHANIZM LOSOWEGO WYKŁADNICZEGO ZNAKOWANIA REM

Mechanizm REM różni się od mechanizmu RED definicją miary przeciążenia i losową funkcją znakowania pakietów. Różnice te decydują o dwóch zasadniczych cechach mechanizmu REM i jego przewadze nad mechanizmem RED:

1. Stabilności. Szybkości transmisji pakietów są stabilne dla dowolnych długości kolejek, niezależnie od liczby użytkowników (połączeń) współdzielących zasoby.

2. Obserwowane przez źródło prawdopodobieństwo znakowania (odrzućania) pakietów zależy w prosty i dokładny sposób od sumy miar przeciążenia w każdym z routerów na trasie połączenia.

Miara przeciążenia

Pierwszą istotną ideą realizowaną w algorytmie REM jest stabilizacja zarówno szybkości transmisji pakietów wokół wartości pojemności łącza, jak i długości kolejki wokół dowolnie małej docelowej długości kolejki, niezależnie od liczby użytkowników (połączeń) współdzielących łącze. Sposobem na realizację tej idei jest wybór odpowiedniej miary przeciążenia zasobów.

Dla każdej kolejki, w której implementowany jest mechanizm REM, utrzymywana jest zmienna, zwana ceną (ang. *price*), będąca miarą przeciążenia. Wartość tej zmiennej jest stosowana do wyznaczania prawdopodobieństwa znakowania pakietów. Cena jest aktualizowana okresowo lub asynchronicznie na podstawie niedopasowania szybkości (ang. *rate mismatch*), tzn. różnicy pomiędzy wejściową szybkością transmisji a pojemnością łącza, oraz niedopasowania długości kolejki (ang. *queue mismatch*), tzn. różnicy pomiędzy długością kolejki a docelową (oczekiwaną) długością kolejki.

Cena wzrasta, jeżeli suma ważona dwóch wymienionych różnic jest dodatnia, w innym przypadku maleje. Suma ważona jest dodatnia wtedy, gdy zarówno wejściowa szybkość transmisji jest większa od pojemności łącza, jak i aktualna długość kolejki jest większa od założonej docelowej (kiedy konieczne jest rozładowanie kolejki). W innych przypadkach suma ważona jest ujemna. Jeżeli liczba użytkowników (połączeń) się zwiększa, wzrasta zarówno niedopasowanie szybkości, jak i długości kolejek, powodując wzrost ceny i tym samym prawdopodobieństwa znakowania pakietów. Zwiększenie prawdopodobieństwa znakowania oznacza wzrost sygnału przeciążenia (ang. *congestion signal*) przekazywanego do źródeł, które po jego odebraniu redukują szybkość transmisji pakietów.

Jeżeli szybkości transmisji pakietów są zbyt małe, to niedopasowanie szybkości ma wartość ujemną, co powoduje obniżenie ceny i prawdopodobieństwa znakowania pakietów oraz zwiększenie szybkości transmisji pakietów aż do chwili, gdy niedopasowanie uzyskuje wartość zerową, co oznacza osiągnięcie stanu równowagi, charakteryzującego się dużym stopniem wykorzystania zasobów oraz pomijalnymi stratami i opóźnieniem pakietów.

Dla danej j -tej kolejki cena $\xi_j(k\delta)$ jest uaktualniana zgodnie z następującym wyrażeniem:

$$\xi_j((k+1)\delta) = \left[\xi_j(k\delta) + \gamma \left(\alpha_j (q_j(k\delta) - q_j^*) + (v_j(k\delta) - C_j(k\delta)) \right) \right]^+,$$

w którym:

- $q_j(k\delta)$ – długość j -tej kolejki w przedziale czasu $k\delta$,
- q_j^* – ustalona docelowa długość j -tej kolejki,
- $v_j(k\delta)$ – szybkość transmisji pakietów do j -tej kolejki w przedziale czasu $k\delta$,
- $C_j(k\delta)$ – pojemność dostępna dla j -tej kolejki w przedziale czasu $k\delta$,
- γ ($\gamma > 0$) i α_j ($\alpha_j > 0$) – stałe,
- $[z]^+ = \max\{z, 0\}$.

Wartości różnic $(q_j(k\delta) - q_j^*)$ i $(v_j(k\delta) - C_j(k\delta))$ są – odpowiednio – miarami niedopasowania długości kolejek i szybkości transmisji pakietów. Stała α_j może być ustalana dla każdego z połączeń osobno i pełni rolę współczynnika wagowego dla wartości niedopasowania długości kolejek, stała γ steruje natomiast reakcją algorytmu na zmiany warunków w węzle sieci.

Cena rośnie, gdy wartość sumy ważonej $\alpha_j(q_j(k\delta) - q_j^*) + (v_j(k\delta) - C_j(k\delta))$ jest dodatnia, i maleje, gdy wartość sumy ważonej jest ujemna. Cena nie ulega zmianie w punkcie równowagi, którego warunkiem istnienia jest zerowa wartość sumy ważonej, tzn. $\alpha_j(q_j - q_j^*) + (v_j - C_j) = 0$. Punkтови równowagi odpowiada stan, kiedy szybkość transmisji jest równa pojemności ($v_j = C_j$) oraz długość kolejki pakietów (zaległość w obsłudze pakietów) jest równa docelowej długości kolejki ($q_j = q_j^*$).

Wyrażenie, według którego uaktualnia się cenę, można zmodyfikować w następujący sposób. Wartość różnicy $v_j(k\delta) - C_j(k\delta)$ jest szybkością wydłużania kolejki wtedy, gdy nie jest ona pusta i może być aproksymowana zmianą jej długości, tzn. wartością różnicy $q_j((k+1)\delta) - q_j(k\delta)$. W takim przypadku wyrażenie na aktualizację ceny przyjmuje postać:

$$\begin{aligned} \xi_j((k+1)\delta) &= \left\{ \xi_j(k\delta) + \gamma \left[\alpha_j(q_j(k\delta) - q_j^*) + [q_j((k+1)\delta) - q_j(k\delta)] \right] \right\}^+ = \\ &= \left\{ \xi_j(k\delta) + \gamma \left[q_j((k+1)\delta) - (1 - \alpha_j) q_j(k\delta) - \alpha_j q_j^* \right] \right\}^+, \end{aligned}$$

wynika z tego, że do aktualizacji ceny wystarcza znajomość aktualnej i poprzedniej długości j -tej kolejki. Jest to o tyle wygodne, że znacznie łatwiej jest próbować długość kolejki niż szybkość transmisji pakietów.

Zasada aktualizacji ceny, będącej parametrem funkcji znakowania pakietów w algorytmie REM, decyduje o jednej z podstawowych różnic pomiędzy algorytmami REM i RED. W algorytmie RED parametrem funkcji znakowania pakietów jest średnia długość kolejki. Jeżeli zwiększa się liczba użytkowników, to musi także wzrastać średnia długość kolejki po to, aby rosła wartość prawdopodobieństwa znakowania pakietów w celu wzmacniania sygnalizacji przeciążenia. W przeciwieństwie do algorytmu RED, w algorytmie REM długość kolejki jest używana do aktualizacji ceny, która z kolei jest parametrem funkcji znakowania służącej do wyznaczania prawdopodobieństwa znakowania pakietów. Oznacza to, że w algorytmie REM zwiększenie liczby użytkowników powoduje wzrost ceny, długość kolejki natomiast stabilizuje się w pobliżu docelowej długości kolejki q_j^* .

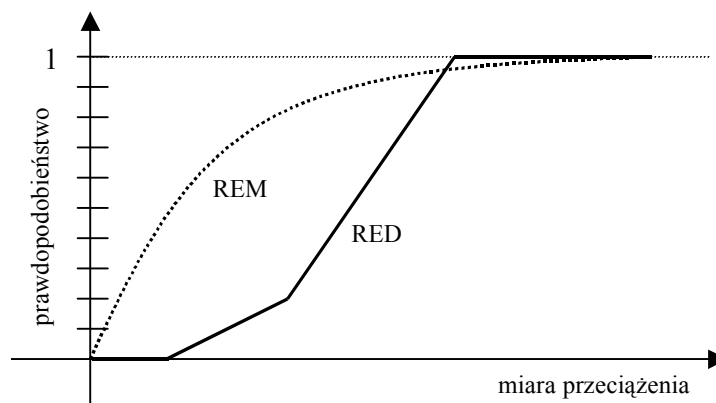
Suma miar przeciążenia

Drugą istotną ideą implementowaną w algorytmie REM jest zastosowanie sumy miar przeciążenia zasobów na trasie jako miary przeciążenia trasy, która następnie jest stosowana do wyznaczania, obserwowanego w źródłach, prawdopodobieństwa zna-

kowania pakietów na trasie. Realizacja omawianej idei ma szczególne znaczenie w sieciach, w których prawdopodobieństwo znakowania pakietu na trasie od źródła do ujścia (ang. *end-to-end marking probability*) zależy od znakowania w każdym węzle trasy (ang. *local marking probability*).

Sposobem realizacji tej idei jest odpowiedni wybór funkcji znakowania pakietów, tj. sposobu wyznaczania prawdopodobieństwa znakowania pakietów.

Dla algorytmu REM charakterystyczne jest to, że w kolejnych kolejkach na trasie połączenia znakowane są tylko te pakiety, które nie były wcześniej oznakowane w kolejkach położonych na trasie i poprzedzających daną kolejkę (ang. *upstream queues*). Pakiety są znakowane z prawdopodobieństwem rosnącym wykładniczo w funkcji ceny (rysunek 7.3).



Rysunek 7.3. Wykresy funkcji znakowania (odrzućcia) pakietów w mechanizmach RED i REM

Wykładnicza postać funkcji znakowania pakietów ma oczekiwane właściwości: jeżeli prawdopodobieństwa znakowania w poszczególnych kolejkach są wykładniczymi funkcjami cen, to prawdopodobieństwo znakowania pakietów na trasie jest wykładniczą funkcją sumy cen w poszczególnych przeciążonych kolejkach na tej trasie. Suma cen w przeciążonych kolejkach na trasie jest miarą przeciążenia trasy. Suma cen, stosowana w funkcji znakowania pakietów na trasie, może być łatwo estymowana w źródłach przez zliczanie udziału oznakowanych pakietów w liczbie wszystkich transmitowanych pakietów. Przy założeniu, że:

- pakiet jest transmitowany trasą składającą się z J łączy, tzn. trasą, na której znajduje się J kolejek charakteryzowanych cenami $\xi_j(k\delta)$ dla $j = 1, 2, \dots, J$,
- funkcja znakowania (prawdopodobieństwo znakowania) pakietów w j -tej kolejce ma postać:

$$p_j(k\delta) = 1 - a^{-\xi_j(k\delta)},$$

gdzie a jest stałą większą od jednościci ($a > 1$),

funkcja znakowania pakietów na trasie ma następującą postać:

$$p(k\delta) = 1 - \prod_{j=1}^J (1 - p_j(k\delta)) = 1 - a^{-\sum_{j=1}^J \xi_j(k\delta)}.$$

Interpretacja składników tego wyrażenia jest następująca:

➤ jeżeli miara przeciążeń na trasie $\xi(k\delta)$, tj. suma cen w poszczególnych kolejkach trasy $\sum_{j=1}^J \xi_j(k\delta)$ wzrasta, to zwiększa się prawdopodobieństwo znakowania pakietów na trasie $p(k\delta)$,

➤ jeżeli prawdopodobieństwo znakowania pakietów $p_j(k\delta)$ w poszczególnych kolejkach jest niewielkie, co oznacza, że ceny $\xi_j(k\delta)$ są małe, to prawdopodobieństwo znakowania pakietów na trasie jest w przybliżeniu proporcjonalne do sumy cen w poszczególnych kolejkach trasy, tzn.:

$$p(k\delta) \approx (\ln a) \sum_{j=1}^J \xi_j(k\delta).$$

Wymienione wcześniej cechy mechanizmu REM, tzn.:

➤ zastosowanie ceny jako miary przeciążenia pozwala na optymalizację stopnia wykorzystania pojemności łączy oraz na stabilizację długości kolejki na dowolnie ustalonym poziomie,

➤ zastosowanie wykładniczej postaci funkcji znakowania pakietów powoduje, że parametrem funkcji znakowania pakietów dla całej trasy jest suma parametrów funkcji znakowania pakietów we wszystkich routerach na trasie,

mogą być implementowane niezależnie od siebie.

Rozwiązania stosowane w mechanizmie REM powodują, że miarą przeciążenia jest cena, która nie zależy od liczby obsługiwanych połączeń, tzn. miara przeciążenia nie jest sprzężona (ang. *decoupled*) z miarami jakości obsługi pakietów, takimi jak: prawdopodobieństwo strat, opóźnienie, długość kolejki. W takim przypadku zwiększenie wartości miary przeciążenia oznacza wzrost kosztów łączy transmisyjnego. W rozwiązaniach takich jak np. RED miara przeciążenia jest sprzężona (ang. *coupled*) z miarami jakości obsługi pakietów, tzn. zwiększenie liczby połączeń oznacza zwiększenie średniej długości kolejki, np. wzrost opóźnienia pakietów. W tym przypadku wzrost wartości miary przeciążenia oznacza pogorszenie jakości obsługi.

7.2. METODY PRZECIWDZIAŁANIA PRZECIĄŻENIOM

Metody przeciwdziałania powstawaniu przeciążeń w sieciach obejmują bardzo szerokie spektrum procedur i mechanizmów, których zadaniem jest regulacja proporcji pomiędzy aktualnym obciążeniem sieci a zasobami udostępnianymi do obsługi ruchu w sieci.

Zwiększanie tej proporcji na ogół oznacza wzrost wartości współczynnika wykorzystania zasobów, ale jest związane ze wzrostem udziału reakcyjnych metod sterowania, których warunkiem skuteczności jest czas na podjęcie decyzji i dostępność informacji o stanie sieci. Przydatność metod reakcyjnych maleje wraz ze wzrostem szybkości procesów realizowanych w sieci. Z kolei ograniczenia czasu podejmowania decyzji w przypadkach reagowania na symptomy przeciążenia wymuszają poszukiwanie i stosowanie metod przybliżonych, których złożoność obliczeniowa i pamięciowa jest mniejsza od złożoności metod dokładnych.

Metody prewencyjne sterowania dotyczą wolnozmiennych procesów w sieci, zwykle zachodzących na jej brzegu, co daje możliwość wyznaczania dokładnych rozwiązań. Metody prewencyjne charakteryzują się względnie małymi, w porównaniu z metodami reakcyjnymi, wartościami współczynników wykorzystania zasobów.

Skuteczność zapobiegania przeciążeniom w danej sieci jest wynikiem jednoczesnego działania zbioru procedur prewencyjnych i reakcyjnych. Udział metod należących do wymienionych grup zależy od wielu czynników, w tym od przyjętych wskaźników jakości działania sieci, struktury ruchu obsługiwanego przez sieć itd.

7.2.1. PLANOWANIE I UDOSTĘPNIANIE ZASOBÓW

Planowanie i udostępnianie zasobów (ang. *resource provisioning*) można zaliczyć zarówno do prewencyjnych, jak i do reakcyjnych metod zarządzania zasobami i ich udostępniania. W zasadzie są metodami planowania zasobów.

Zadania zarządzania zasobami mają charakter prewencyjny wtedy, gdy sieć i jej zasoby są planowane. Na podstawie rozpoznanych i zidentyfikowanych wymagań aplikacji, liczby użytkowników i charakterystyk przewidywanego ruchu definiowana jest struktura topologiczna sieci, liczba kanałów transmisyjnych i ich pojemności, liczba węzłów komutacyjnych i wartości ich parametrów, liczba węzłów dostępowych, nadmiarowość zasobów transmisyjnych i komutacyjnych itd. Zmiana aplikacji, liczby użytkowników, wymagań ilościowych, jakościowych itp. powoduje, że niezbędne są zmiany w zasobach sieci. Zmiany te polegają na zmianach struktury topologicznej, liczby kanałów transmisyjnych i ich przepustowości, liczby i rodzaju węzłów komutacyjnych, liczby i parametrów węzłów dostępowych itd.

Zarządzanie zasobami i udostępnianie zasobów można zaliczyć do dobrze zdefiniowanych zadań projektowania klasy TCFA (ang. *Topology, Capacity and Flow Assignment*), omawianych często w literaturze, dotyczącej różnych sformułowań tych zadań oraz metod ich dokładnego i przybliżonego rozwiązywania. Zadania projektowania najczęściej są formułowane jako zadania optymalizacji, jedno- lub wielokryterialnej, dla różnych zbiorów ograniczeń. Do najczęściej stosowanych kryteriów optymalizacji można zaliczyć prawdopodobieństwo odrzucenia zgłoszeń, średnie opóźnienie wnoszone przez sieć itd., najczęściej formułowanymi ograniczeniami są koszty: budowy sieci, jej utrzymania oraz niezawodność itd. Do rozwiązywania tak formułowanych zadań optymalizacji, charakteryzujących się dużą złożonością obli-

zioniową, stosuje się metody analityczne, symulacyjne i – coraz częściej – metody sztucznej inteligencji.

7.2.2. STEROWANIE PRZYJMOWANIEM WYWOŁAŃ

7.2.2.1. ZADANIA STEROWANIA PRZYJMOWANIEM WYWOŁAŃ

Sterowanie przyjmowaniem wywołań CAC (ang. *Call Admission Control*) obejmuje działania uruchamiane w chwili nadejścia nowego wywołania (zgłoszenia), których efektem jest decyzja o przyjęciu lub odrzuceniu wywołania. Wywołanie jest przyjmowane wtedy, gdy istnieją wolne zasoby pozwalające obsłużyć wywołanie, gwarantując jednocześnie żądany poziom jakości usług i nie obniżając poziomu jakości usług dla wywołań już obsługiwanych w sieci.

O jakości mechanizmów przyjmowania wywołań (sterowania dostępem) decyduje zdolność do oszacowania, na podstawie wartości parametrów opisujących aktywność źródła wywołania, wielkości żądanej i przydzielonej pojemności (alokacja pojemności) oraz zdolność predykcji wpływu udostępnianego połączenia na jakość usług dla już obsługiwanych w sieci wywołań, z którymi nowe wywołanie jest multipleksowane (superpozycja strumieni zgłoszeń).

Od każdego z mechanizmów przyjmowania wywołań wymagane jest działanie w czasie rzeczywistym i zorientowanie na maksymalizację stopnia wykorzystania zasobów. Proces przyjmowania wywołań jest dwuetapowy i zwykle składa się z:

- analizy możliwości przyjęcia danego wywołania w wyniku szacowania skutków przyjęcia wywołania; wymaga to stosowania modeli sieci opartych m.in. na metodzie superpozycji strumieni jednostek danych (ang. *superposition of arrival streams*) przyjętych i przyjmowanych wywołań,

- przydzielenia, po zaakceptowaniu wywołania, pojemności (ang. *bandwidth allocation*) w ilościach wyznaczonych zgodnie z obowiązującym algorytmem przyjmowania wywołań (ang. *call admission algorithm*).

7.2.2.2. SUPERPOZYCJA STRUMIENI ZGŁOSZEŃ

Większość sieci teleinformatycznych (np. sieci ATM, FR, TCP/IP), niezależnie od implementowanej techniki komutacji i poziomu granulacji przepływów, można traktować jako środowisko, w którym jednocześnie obsługuje się wiele różnych wywołań. Wywołania te różnią: źródła i ujścia, wielkość jednostek danych, czas trwania połączenia, szybkość napływu zgłoszeń, zakres zmienności szybkości napływu zgłoszeń, żądany poziom obsługi itd.

Dla sieci jednocześnie obsługujących wiele wywołań charakterystyczne jest to, że w każdym z jej węzłów (zakładając, że każdy z węzłów może jednocześnie pełnić rolę węzła źródłowego, ujściowego lub pośredniego) zachodzą procesy: akceptacji wywołań i agregacji strumieni ruchu (funkcja węzła źródłowego), wyprowadzania i roz-

dzielania zagregowanych strumieni jednostek danych (funkcja węzła ujściowego) oraz rozdzielania i superpozycji strumieni jednostek danych (funkcja węzła pośredniego).

Nadejście nowego wywołania wymaga jego akceptacji lub odrzucenia. Akceptacja oznacza, że pozytywnie została oceniona możliwość zagwarantowania wymaganej jakości obsługi wywołania bez naruszania jakości obsługi wcześniej przyjętych i obsługiwanych wywołań. Gwarancja jakości obsługi nowo przyjmowanych i obsługiwanych wywołań oznacza z kolei, że oszacowane zostały zmiany wartości parametrów opisujących działanie sieci spowodowane zmianami w zasobach, jakie powoduje akceptacja wywołania. Oszacowanie tych wartości wymaga stosowania odpowiednich modeli sieci, których rozwiązania są podstawą podejmowania decyzji o akceptacji lub odrzuceniu wywołań.

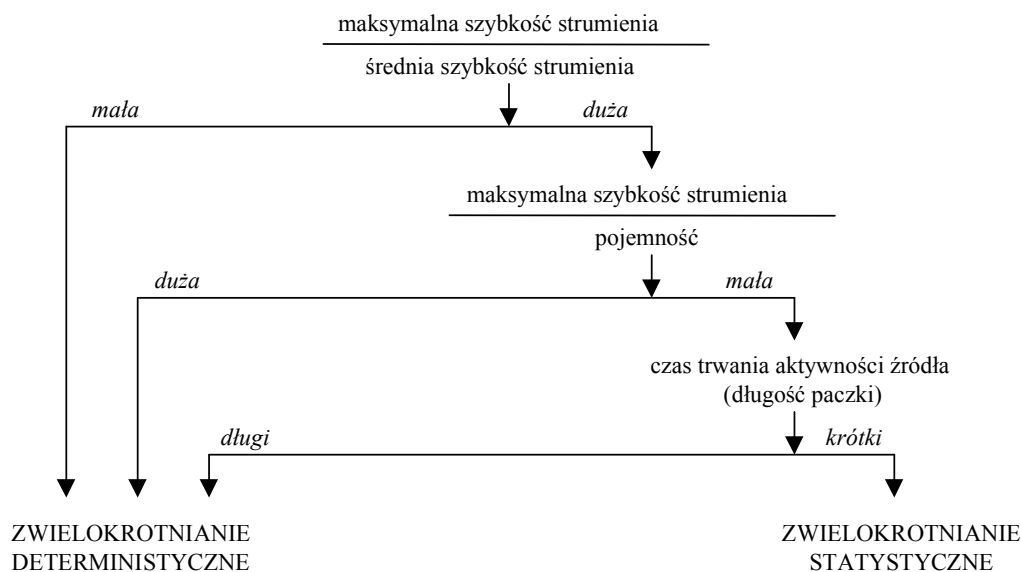
Sieć teleinformatyczna może być rozpatrywana jako sieć kolejek, której struktura topologiczna jest tożsama ze strukturą topologiczną modelowanej sieci. W modelu takim ograniczona pojemność kolejki odpowiada pamięci buforowej, a stanowisko obsługi kolejki – łączu transmisyjnemu. Model taki można zbudować dla każdej sieci, ale praktyczne możliwości uzyskania dokładnych rozwiązań numerycznych w czasie rzeczywistym, nawet dla niewielkich sieci, są praktycznie żadne; stąd wynika niewielka przydatność takich modeli.

Jednym z często stosowanych rozwiązań jest dekompozycja dużych sieci kolejkowych, a następnie analiza wyizolowanych, pojedynczych kolejek; prowadzi to do znacznej redukcji złożoności obliczeniowej. Kolejne stosowane uproszczenia polegają na minimalizacji liczby strumieni wejściowych w takich prostych systemach kolejkowych. W najczęściej stosowanym podejściu do każdej kolejki dochodzą dwa strumienie zgłoszeń: jeden – odpowiadający nowemu wywołaniu i drugi – będący superpozycją strumieni zgłoszeń odpowiadających wszystkim już obsługiwanym wywołaniom. Dalsze uproszczenie to redukcja wymiaru zadania przez złożenie wszystkich strumieni zgłoszeń w jeden. Podejście takie, jakkolwiek prowadzi do wyników przybliżonych, jest z powodzeniem stosowane do szacowania wartości parametrów jakości działania sieci i jej komponentów. O jakości uzyskiwanego przybliżenia decyduje w dużej mierze sposób modelowania superpozycji strumieni zgłoszeń.

7.2.2.3. ALOKACJA POJEMNOŚCI

Zadanie alokacji pojemności (szerokości pasma) (ang. *bandwidth allocation*) polega na przydzieleniu napływającemu wywołaniu (żądaniu dostępu) pojemności gwarantującej jakość usług żadaną dla danego wywołania.

Metody alokacji pojemności dzieli się na dwie grupy: zwielokrotniania deterministycznego (ang. *deterministic multiplexing*) i zwielokrotniania statystycznego (ang. *statistical multiplexing*). Wybór jednego ze sposobów zwielokrotniania zależy od wielu czynników, z tym że podstawowymi kryteriami wyboru są: zakres zmienności szybkości strumienia ruchu, stopień zajętości pojemności oraz czas trwania aktywności źródła strumienia ruchu (rysunek 7.3).



Rysunek 7.3. Ogólne kryteria wyboru pomiędzy zwielokrotnianiem deterministycznym i statystycznym

Zwielokrotnianie deterministyczne (deterministyczna alokacja pojemności) jest konserwatywnym, restrykcyjnym, mało elastycznym i niezgodnym z ideą komutacji wiadomości (tym bardziej, im wiadomości są krótsze) sposobem przydzielania zasobów. Polega na przydzieleniu każdemu nowemu połączeniu (wywołaniu) zasobów w ilości równej wartości maksymalnej (wartości szczytowej) zapotrzebowania na zasoby (pojemność). Taki sposób alokacji zasobów, właściwy dla strumieni o stałej (lub zbliżonej do stałej) szybkości napływu jednostek danych, charakteryzują:

- prostota implementacji,
- całkowita eliminacja przeciążeń,
- zależność stopnia wykorzystania przydzielonych zasobów od wartości ilorazu wartości maksymalnej szybkości napływu jednostek danych do wartości średniej szybkości; stopień wykorzystania zasobów maleje wraz ze wzrostem wartości tego ilorazu,
- brak zapobiegania przepelnieniom buforów, a tym samym stratom jednostek danych w węzłach pośrednich na trasach w sieci.

Zwielokrotnianie statystyczne jest sposobem przydzielania zasobów zgodnym z ideą komutacji, polegającą na równoważeniu w czasie zmiennego zapotrzebowania na zasoby i stałych zasobów sieci, tj. równoważeniu przesuniętych w czasie: chwilowego nadmiaru zapotrzebowania (w porównaniu z dostępnymi zasobami) oraz chwilowego nadmiaru zasobów sieci (w porównaniu z zapotrzebowaniem). Przyjmuje się, że złożenie wielu strumieni o różnej i zmiennej szybkości napływu jednostek danych, generowanych przez wiele różnych źródeł wywołań, prowadzi do „wygładzania” ruchu (ang. *traffic smoothing*) charakterystyk ruchu zagregowanego, przekazywanego

między węzłami sieci i uzasadnia stosowanie statystycznych metod zwielokrotniania. Skala efektu „wygładzania” ruchu zależy od charakterystyk agregowanych strumieni składowych. Efekt „wygładzania” strumienia zagregowanego zależy od względnej szybkości składowych strumieni jednostek danych.

Do pomiaru i porównywania algorytmów zwielokrotniania statystycznego stosuje się m.in. wartość poziomu zwielokrotniania statystycznego (ang. *level of statistical multiplexing*), odpowiadającego zdolności „wypełnienia” przez przyjmowane wywołania łącza o zadanej pojemności.

Ogólna zasada zwielokrotniania statystycznego polega na przydziale nowemu wywołaniu, o zmiennej szybkości napływu jednostek danych, pojemności statystycznej (ang. *statistical bandwidth*) mniejszej od wartości szczytowej szybkości napływu i większej od wartości średniej szybkości napływu tak, aby suma losowego zapotrzebowania na pojemność w czasie trwania połączenia była mniejsza od pojemności przydzielonej na czas trwania połączenia. Omawiany sposób alokacji charakteryzują:

- niedokładne metody szacowania zapotrzebowania na pojemność strumieni o zmiennej szybkości napływu jednostek danych oraz mało znana natura procesów statystycznej agregacji strumieni jednostek danych w węzłach sieci,

- złożoność algorytmów i sposobów ich implementacji,

- zależność stopnia wykorzystania pojemności (zasobów) od przydzielonej statystycznej pojemności: zwiększa się – gdy przydzielona pojemność statystyczna zbliża się do wartości odpowiadającej średniej szybkości napływu i maleje – gdy zbliża się do wartości odpowiadającej szczytowej szybkości napływu,

- zależność stopnia wykorzystania pojemności od wartości ilorazu wartości szczytowej szybkości napływu do wartości pojemności łącza; jeżeli wartość ilorazu maleje, to zwiększa się efektywność zarządzania pamięcią buforową w węzłach oraz efektywność zwielokrotniania,

- możliwość zrealizowania (zwielokrotniania) większej liczby połączeń o zmiennej szybkości napływu jednostek danych (w porównaniu z alokacją deterministyczną) i tym samym zwiększenia stopnia wykorzystania pojemności (zasobów),

- zależność wartości pojemności statystycznej przydzielonej dla połączenia nie tylko od statystycznych charakterystyk tego połączenia (charakterystyk strumienia jednostek danych), ale także od charakterystyk połączeń (charakterystyk strumieni jednostek danych) już obsługiwanych w sieci,

- zależność efektywności statystycznego zwielokrotniania od współczynnika aktywności (stopnia wykorzystania) źródła (ang. *source utilization*) oraz średniego czasu trwania połączenia, w którym jednostki danych są generowane z szybkością szczytową (ang. *burst length of connection*). W przypadku dwóch strumieni – o tych samych wartościach szczytowej i średniej szybkości napływu jednostek danych oraz tych samych wymaganiach co do jakości obsługi – strumień, dla którego czas generowania jednostek danych z wartością szczytową jest dłuższy, wymaga większych zasobów sieci.

7.2.2.4. ALGORYTMY PRZYJMOWANIA WYWOŁAŃ

Liczba algorytmów przyjmowania wywołań jest duża i wciąż rośnie. Do porównywania istniejących i nowo opracowywanych algorytmów można stosować wiele różnych wskaźników.

Najczęściej używanymi wskaźnikami jakości algorytmów przyjmowania wywołań są następujące:

1. Przeszacowanie, tzn. przydzielanie akceptowanemu wywołaniu pojemności większej od potrzebnej. Przeszacowanie gwarantuje żądaną jakość obsługi wywołania, ale obniża stopień wykorzystania zasobów sieci.

2. Niedooszacowanie, tzn. przydzielanie akceptowanemu wywołaniu przepustowości mniejszej od potrzebnej. Niedooszacowanie wymaga do gwarantowania żądanej jakości obsługi wywołania wprowadzenia dodatkowych zabezpieczeń powodujących obniżenie stopnia wykorzystania zasobów.

3. Profilowanie ruchu, tzn. wprowadzanie ograniczeń charakterystyk źródeł i generowanego przez te źródła ruchu, których spełnienie jest warunkiem przydziału przepustowości gwarantujących żądaną jakość usług.

4. Rozmiar sieci, tzn. wprowadzanie ograniczeń wielkości sieci (liczby węzłów i łączy transmisyjnych) oraz złożoności jej struktury topologicznej, których spełnienie jest warunkiem uzyskania rozwiązania.

5. Złożoność obliczeniowa i pamięciowa.

Algorytmy sterowania przyjmowaniem wywołań w różnym stopniu korzystają z różnych parametrów, których wartości jednoznacznie określają charakterystyki ruchu generowanego przez źródło. Do podstawowych parametrów charakteryzujących źródła strumieni jednostek danych zalicza się:

- szczytową (maksymalną) szybkość generowania zgłoszeń (ang. *peak rate*),
- średnią (graniczną) szybkość generowania zgłoszeń (ang. *sustainable rate*),
- maksymalny rozmiar paczki zgłoszeń (ang. *maximum burst size*) definiowany dla źródła generującego zgłoszenia ze średnią szybkością,
- rozkłady opisujące zmienny charakter aktywności źródła.

7.2.2.4.1. ALGORYTM GA

Działanie algorytmu GA (ang. *Gaussian Approximation*) polega na akceptacji lub odrzuceniu nowych wywołań w zależności od dostępności pojemności, z tym że nowe wywołanie jest przyjmowane do obsługi wtedy, gdy nie powoduje to zwiększenia chwilowej szybkości zagregowanego strumienia jednostek danych o wcześniej zadaną wartość.

Na potrzeby algorytmu GA przyjmowane są następujące założenia:

- zagregowany strumień jednostek danych (np. komórek) jest sumą n niezależnych strumieni jednostek danych,
- szybkość transmisji w każdym połączeniu (szybkość strumienia jednostek danych) i ($i = 1, 2, \dots, n$) jest niezależną zmienną losową X_i o rozkładzie normalnym

(Gausa) $N(m_i, \sigma_i)$ (gdzie m_i – wartość średnia, a σ_i – wartość odchylenia standardowego),

➤ szybkość transmisji zagregowanego strumienia, powstającego w wyniku zwielokrotnienia (sumowania) szybkości transmisji strumieni składowych strumieni jednostek danych, jest zmienną losową X o rozkładzie normalnym $N(m, \sigma)$ (gdzie m ($m = m_1 + m_2 + \dots + m_n$) – wartość średnia, a σ ($\sigma^2 = \sigma_1^2 + \sigma_2^2 + \dots + \sigma_n^2$) – odchylenie standardowe zagregowanego strumienia zgłoszeń).

Celem algorytmu GA jest wyznaczenie takiej żądanej wartości pojemności C_0 , by prawdopodobieństwo tego, że chwilowa szybkość transmisji strumienia zagregowanego X jest większa od wartości żądanej pojemności C_0 była mniejsza od dowolnej, zadanej wartości ε , tzn. żądanej pojemności spełniającej warunek: $p(X > C_0) < \varepsilon$. Dla sumy n niezależnych zmiennych losowych o rozkładach normalnych $N(m_i, \sigma_i)$:

$$p(X > C_0) = p\left(\frac{X - m}{\sigma} > \frac{C_0 - m}{\sigma}\right) \approx p\left(X_{01} > \frac{C_0 - m}{\sigma}\right) = p(X_{m\sigma} > C_0),$$

gdzie:

X_{01} i $X_{m\sigma}$ – zmienne losowe o rozkładach normalnych $N(0,1)$ i $N(m,\sigma)$,

m i σ – wartość średnia i odchylenie standardowe zagregowanego strumienia jednostek danych,

skąd wynika, że:

$$p(X > C_0) \approx p(X \geq m + \alpha\sigma) \approx \varepsilon,$$

co pozwala na obliczenie, dla zadanej wartości prawdopodobieństwa ε , wartości współczynnika α , określonego zależnością:

$$p(|X - m| > \alpha\sigma) = \varepsilon = \frac{2}{2\pi} \int_{\alpha}^{\infty} e^{-x^2/2} dx.$$

Jednym ze stosowanych przybliżonych sposobów wyznaczania wartości współczynnika α jest formuła:

$$\alpha = \sqrt{2 \ln \frac{1}{\varepsilon} - \ln 2\pi}.$$

Wyznaczona w ten sposób wartość żądanej pojemności C_0 jest górnym ograniczeniem pojemności wymaganej do obsługi zagregowanego strumienia jednostek danych złożonego z n niezależnych strumieni, gwarantującej spełnienie zadanego warunku, tj. ograniczenia prawdopodobieństwa, że szybkość sumarycznego strumienia jednostek danych jest większa od pojemności. Warunkiem obsługi sumarycznego strumienia o żądanej pojemności C_0 jest całkowita pojemność kanału większa od żądanej pojemności, tj. $C_0 \leq C$, gdzie C jest całkowitą pojemnością kanału transmisyjnego.

Jeżeli nadchodzące jednostki danych nie są buforowane, prawdopodobieństwo przekroczenia przez szybkość strumienia zagregowanego pojemności jest równe

prawdopodobieństwu odrzucania jednostek danych. Jeżeli sumowanie składowych strumienia sumarycznego odbywa się w buforze, który (w przypadku posiadania odpowiedniej pojemności) może spełniać rolę kompensatora chwilowych przekroczeń żądanej wartości C_n przez wartość strumienia zagregowanego X , to wartość prawdopodobieństwa strat jednostek danych może być zmniejszona. Procedura podejmowania decyzji o przyjęciu lub odrzuceniu nowego strumienia jednostek danych, będącego zmienną losową X_{n+1} o rozkładzie normalnym $N(m_{n+1}, \sigma_{n+1})$ wtedy, gdy:

➤ w łączu o całkowitej pojemności C jest obsługiwany sumaryczny strumień złożony z n niezależnych strumieni jednostek danych,

➤ sumaryczny strumień jest zmienną losową X o rozkładzie normalnym $N(m, \sigma)$, gdzie: $(m = m_1 + m_2 + \dots + m_n)$ oraz $\sigma^2 = \sigma_1^2 + \sigma_2^2 + \dots + \sigma_n^2$,

jest następująca:

1. oblicza się nowe wartości średniej i odchylenia standardowego:

$$m' = m + m_{n+1} = (m_1 + m_2 + \dots + m_n) + m_{n+1},$$

$$\sigma' = \sqrt{\sigma^2 + \sigma_{n+1}^2} = \sqrt{(\sigma_1^2 + \sigma_2^2 + \dots + \sigma_n^2) + \sigma_{n+1}^2};$$

2. oblicza się nową wartość żądanej pojemności łącza:

$$C_0 = m' + \alpha\sigma' = \sum_{i=1}^n m_i + m_{n+1} + \alpha \sqrt{\sum_{i=1}^n \sigma_i^2 + \sigma_{n+1}^2};$$

3. jeżeli spełniony jest warunek:

$$C_0 \leq C,$$

to nowe $(n + 1)$ -sze wywołanie jest przyjmowane do obsługi; w innym przypadku wywołanie jest odrzucane.

W celu wyeliminowania niedokładności obliczeń współczynnika α wtedy, gdy nie jest spełnione założenie o rozkładzie normalnym zmiennej losowej opisującej strumień zagregowany, stosuje się rozwiązanie, w którym do obliczeń wykorzystywane są odpowiednio zmodyfikowane (powiększone) wartości parametrów rozkładów normalnych, tzn. zamiast rozkładów $N(m_i, \sigma_i)$ stosowane są rozkłady $N(m_i + a\sigma_i, b\sigma_i)$.

W takim przypadku wartość żądanej pojemności dla strumienia złożonego z n niezależnych strumieni jednostek danych oblicza się za pomocą następującego wzoru:

$$C_0 = \sum_{i=1}^n (m_i + a\sigma_i) + \alpha b \sqrt{\sum_{i=1}^n \sigma_i^2},$$

w którym a i b są wartościami stałymi, do wyznaczenia których niezbędne jest monitorowanie rzeczywistych rozkładów strumieni jednostek danych.

Algorytm GA jest bardzo prosty i oparty na uproszczeniach znacznie ograniczających zakres jego stosowalności oraz jakość otrzymywanych rozwiązań, ponieważ:

1. Założenie o rozkładzie gaussowskim strumienia zagregowanego jest realistyczne, gdy wartości parametrów strumieni składowych (wartość średnia i odchylenie

standardowe) są porównywalne. Strumienie składowe mogą się znacznie różnić, zwłaszcza wartości ich odchyłeń standardowych.

2. Wszystkie połączenia są traktowane tak, jakby żądana dla nich jakość obsługi (wyrażana prawdopodobieństwem utraty zgłoszeń) była taka sama. W rzeczywistości różne połączenia mogą się znacznie różnić wymaganiami w tym zakresie.

3. Nie ma odwołania do cech statystycznego zwielokrotniania, wymagającego uwzględnienia kompensującej roli pojemności pamięci buforowej. W rzeczywistości cechy zagregowanych strumieni zgłoszeń są pochodną typu stosowanej w sieci metody zwielokrotniania (multipleksacji) statystycznego.

7.2.2.4.2. ALGORYTM FBR

W algorytmie FBR (ang. *Fast Buffer Reservation*) zastosowano mechanizm rezerwacji części pojemności pamięci buforowej. Podstawą podejmowania decyzji o przyjęciu lub odrzuceniu nowego wywołania jest stan zajętości pamięci buforowej.

Zagregowany strumień jednostek danych jest sumą niezależnych strumieni jednostek danych generowanych przez źródła, które mogą się znajdować w jednym z dwóch stanów: aktywności lub ciszy. Na czas trwania stanu aktywności i -tego źródła, w pamięci buforowej łączy wyjściowego, charakteryzowanej całkowitą liczbą B szczelin pamięci buforowej (ang. *buffer slot*) oraz liczbą dostępnych (wolnych) szczelin pamięci buforowej B_a , rezerwowana jest wcześniej ustalona liczba szczelin B_i ($B_i < B$), która jest dostępna dla zgłoszeń generowanych przez i -te źródło. Liczba szczelin zajętych w danej chwili jest oznaczona przez b_i ($b_i \leq B$). Po ustaniu aktywności źródła wszystkie rezerwowane szczeliny pamięci buforowej B_i są zwalniane i zwracane do puli o pojemności B .

Początek aktywności i -tego źródła wiąże się z wysłaniem specjalnego zgłoszenia startu okresu aktywności (ang. *start-of-the burst*), w którym przenoszona jest informacja o liczbie rezerwowanych szczelin wyjściowej pamięci buforowej łączy. Jeżeli liczba żądanych szczelin pamięci buforowej nie jest dostępna, to wszystkie jednostki danych tego połączenia są odrzucane aż do chwili nadejścia nowego zgłoszenia startu aktywności. W algorytmie zastosowano mechanizm pozwalający na wymuszenie powrotu do stanu ciszy. Jeżeli zostanie przekroczony ustalony limit czasu liczonego od chwili napływu ostatniego zgłoszenia do pamięci buforowej, to zarezerwowane wcześniej szczeliny pamięci buforowej są zwalniane.

Decyzja o przyjęciu do obsługi lub odrzuceniu wywołania (strumienia jednostek danych) jest podejmowana na podstawie obliczanej wartości prawdopodobieństwa nadmiaru żądań (popytu) p_e (ang. *excess demand probability*), tj. prawdopodobieństwa, że liczba szczelin pamięci buforowej, żądanych przez wszystkie źródła (dotychczas obsługiwane i aktualnie żądające dostępu), jest większa od liczby wszystkich szczelin (pojemności) pamięci buforowej. Jeżeli obliczona wartość prawdopodobieństwa p_e jest większa od ustalonej wartości \bar{p}_e , to żądanie dostępu jest odrzucane, w innym przypadku wywołanie (żądanie dostępu) jest przyjmowane do obsługi.

Przy założeniu, że:

- $v_{i \max}$ – wartość szczytowa szybkości generowania zgłoszeń przez i -te źródło,
- \bar{v}_i – wartość średnia szybkości generowania zgłoszeń i -tego źródła,
- x_i – całkowitoliczbowa zmienna losowa oznaczająca liczbę szczelin pamięci buforowej żądanych przez strumień jednostek danych generowany przez i -te źródło,
- v – szybkość obsługi zgłoszeń połączenia wyjściowego,

zmienna losowa x_i przyjmuje wartości $x_i = 0$ (dla $s_i = 0$ – stan ciszy) oraz $x_i = B_i$ (dla $s_i = 1$ – stan aktywności) z prawdopodobieństwem:

$$p_i(x_i) = \begin{cases} \frac{\bar{v}_i}{v_{i \max}} & \text{dla } x_i = B_i \\ 1 - \frac{\bar{v}_i}{v_{i \max}} & \text{dla } x_i = 0 \end{cases},$$

gdzie liczbę szczelin pamięci buforowej żądanych przez aktywne źródło jednostek danych B_i określa wyrażenie:

$$B_i = \left\lceil \frac{Bv_{i \max}}{v} \right\rceil,$$

($\lceil z \rceil$ oznacza najmniejszą liczbę całkowitą większą od z).

Całkowita liczba szczelin pamięci buforowej żądanych przez n źródeł, tzn. zmienna losowa X , jest sumą n całkowitoliczbowych zmiennych losowych:

$$X = x_1 + x_2 + \dots + x_n.$$

Jeżeli zmienne losowe x_i ($i = 1, 2, \dots, n$) są wzajemnie niezależne, funkcję tworzącą prawdopodobieństwa całkowitoliczbowej zmiennej losowej X określa wyrażenie:

$$\varphi_X(s) = \prod_{i=1}^n \left((1 - p_i) + p_i s^{B_i} \right) = \sum_{l=0}^L c_l s^l = c_0 + c_1 s + c_2 s^2 + \dots + c_L s^L,$$

gdzie:

$$s \in [-1, 1],$$

L – suma żądanych liczb szczelin pamięci buforowej przez wszystkie źródła strumieni zgłoszeń: $L = B_1 + B_2 + \dots + B_n$,

c_l – prawdopodobieństwo ($c_l = P[X = l]$), że całkowita liczba żądanych szczelin pamięci buforowej jest równa l ($l = 0, 1, 2, \dots, L$).

Prawdopodobieństwo nadmiaru żądań p_e jest dane zależnością:

$$p_e = P[X \geq C] = 1 - \sum_{j=1}^B c_j = \sum_{j=B+1}^L c_j.$$

Jeżeli w chwili aktywności n źródeł generowane jest żądanie przyznania określonej liczby szczelin pamięci buforowej przez $(n + 1)$ źródło zgłoszeń, to funkcja tworząca prawdopodobieństwo jest dana wyrażeniem:

$$\begin{aligned}\bar{\varphi}_X(s) &= \prod_{i=1}^{n+1} \left((1-p_i) + p_i s^{B_i} \right) = \left((1-p_{n+1}) + p_{n+1} s^{B_{n+1}} \right) \prod_{i=1}^n \left((1-p_i) + p_i s^{B_i} \right) = \\ &= \bar{c}_0 + \bar{c}_1 s + \bar{c}_2 s^2 + \dots + \bar{c}_L s^L = \left((1-p_{n+1}) + p_{n+1} s^{B_{n+1}} \right) (c_0 + c_1 s + c_2 s^2 + \dots + c_L s^L),\end{aligned}$$

gdzie $L = B_1 + B_2 + \dots + B_n + B_{n+1}$.

Zależności pomiędzy wartościami współczynników (prawdopodobieństw) \bar{c}_l oraz c_l , dla każdego l ($l = 0, 1, 2, \dots, L$), określa wyrażenie:

$$\bar{c}_l = (1-p_{n+1})c_l + p_{n+1}c_{l-B_{n+1}},$$

gdzie $c_k = 0$ dla $k < 0$.

W przypadku zwolnienia B_j szczelin pamięci buforowej, wynikającego z zakończenia obsługi j -tego źródła strumienia (jednego spośród n strumieni), funkcję tworzącą prawdopodobieństwo określa równanie:

$$\begin{aligned}\tilde{\varphi}_X(s) &= \frac{\prod_{i=1}^n \left((1-p_i) + p_i s^{B_i} \right)}{(1-p_j) + p_j s^{B_j}} = \tilde{c}_0 + \tilde{c}_1 s + \tilde{c}_2 s^2 + \dots + \tilde{c}_L s^L = \\ &= \frac{\sum_{l=1}^L c_l s^l}{(1-p_j) + p_j s^{B_j}} = \frac{c_0 + c_1 s + c_2 s^2 + \dots + c_L s^L}{(1-p_j) + p_j s^{B_j}},\end{aligned}$$

gdzie: $L = B_1 + B_2 + \dots + B_{j-1} + \dots + B_{j+1} + B_n$,

z którego wynika, że zależność pomiędzy wartościami prawdopodobieństw \tilde{c}_l oraz c_l jest określona następującym wzorem:

$$\tilde{c}_l = \frac{c_l}{1-p_j} + \frac{p_j c_{l-B_j}}{(1-p_j)^2},$$

w którym $c_{l-B_j} = 0$ dla $l - B_j < 0$.

Podane wyrażenia rekurencyjne obliczania wartości współczynników funkcji tworzących prawdopodobieństwa pozwalają w łatwy sposób obliczyć wartości prawdopodobieństwa nadmiaru żądań p_e , na podstawie którego podejmowane są decyzje o akceptacji lub odrzuceniu strumieni zgłoszeń.

W niektórych wybranych przypadkach algorytm ten może generować błędne decyzje. Jeżeli przyjąć dla przykładu, że dwa żądania dostępu ($i = 1, 2$), wymagające w okresie aktywności źródła strumieni zgłoszeń wszystkich szczelin pamięci buforowej ($B_1 = B_2 = B = 10$) z prawdopodobieństwem, odpowiednio $p_1 = 0,9$ i $p_2 = 0,001$, to funkcja tworząca $\varphi_X(s)$ i prawdopodobieństwo nadmiaru żądań p_e są dane wyrażeniami:

$$\begin{aligned}\varphi_X(s) &= \left((1-p_1) + p_1 s^{B_1} \right) \left((1-p_2) + p_2 s^{B_2} \right) = c_0 + c_{10} s^{10} + c_{20} s^{20} = \\ &= (0,1 + 0,9 s^{10}) (0,999 + 0,001 s^{10}) = 0,0999 + 0,8992 s^{10} + 0,0009 s^{20},\end{aligned}$$

$$p_e = 1 - (c_0 + c_{10} + c_{20}) = 0,0009.$$

Jeżeli natomiast decyzja o przyjęciu do obsługi pierwszego żądania dostępu będzie pozytywna, gdy $\bar{p}_e > p_e$ (\bar{p}_e jest zadaną wartością prawdopodobieństwa nadmiaru żądań); w omawianym przypadku – dla $\bar{p}_e = 0,0008$ – żądanie zostanie przyjęte, mimo że nie będzie mogło być obsłużone z braku wolnych szczelin pamięci buforowej.

Przytoczony przykład pokazuje, że algorytm FBR prowadzi do błędnych decyzji w sytuacjach, gdy sumy żądanych liczb szczelin pamięci buforowej znacznie przekraczają liczbę wszystkich szczelin pamięci buforowej.

Działanie algorytmu można poprawić po zastąpieniu w procedurze podejmowania decyzji o przyjęciu lub odrzuceniu żądania dostępu prawdopodobieństwa nadmiaru żądań prawdopodobieństwem, że żądana liczba szczelin jest niedostępna w czasie trwania okresu aktywności i -tego źródła (żądania dostępu, połączenia). Wartość tego prawdopodobieństwa jest ograniczona z góry przez wartość ($p_{ir} = P(X - x_i > B - B_i)$), będącego prawdopodobieństwem rywalizacji (ang. *concatenation probability*) i -tego źródła.

Charakter zależności pomiędzy wartościami prawdopodobieństw rywalizacji p_{ir} i nadmiaru żądań p_e wynika z następujących wyrażen:

$$\begin{aligned} P[X = B] &= (1 - p_i) P[X - x_i = B] + p_i P[X - x_i = B - B_i] \Rightarrow \\ &\Rightarrow P[X > B] = (1 - p_i) P[X - x_i > B] + p_i P[X - x_i > B - B_i] \Rightarrow \\ &\Rightarrow P[X - x_i > B - B_i] = \frac{1}{p_i} P[X > B] - \frac{1 - p_i}{p_i} P[X - x_i > B] \Rightarrow \\ &\Rightarrow p_{ir} = P[X - x_i > B - B_i] \leq \frac{1}{p_i} P[X > B] = \frac{1}{p_i} p_e \Rightarrow p_{ir} \leq \frac{p_e}{p_i}, \end{aligned}$$

gdzie p_i jest prawdopodobieństwem żądania przez i -te źródło dostępu do wszystkich szczelin zarezerwowanej pamięci buforowej B_i .

Z ostatniego wyrażenia wynika, że różnica pomiędzy prawdopodobieństwami p_{ir} i p_e rośnie wraz ze zmniejszaniem się wartości prawdopodobieństwa p_i (dla danych z powyżej przytoczonego przykładu: wartości $p_1 = 0,9$ odpowiada wartość $p_{1r} = 0,001$, $p_2 = 0,001$ odpowiada wartość $p_{2r} = 0,9$, natomiast $p_e = 0,0009$).

Wartość różnicy pomiędzy wartościami p_{ir} i p_e wynika z następujących zależności rekurencyjnych:

$$\begin{aligned} P[X - x_i > B - B_i] &= \frac{1}{p_i} P[X > B] - \frac{1 - p_i}{p_i} P[X - x_i > B] \Rightarrow \\ &\Rightarrow P[X - x_i > B] = \frac{1}{1 - p_i} P[X > B] - \frac{p_i}{1 - p_i} P[X - x_i > B - B_i], \\ P[X - x_i > B - B_i] &= \frac{1}{1 - p_i} P[X > B - B_i] - \frac{p_i}{1 - p_i} P[X - x_i > B - 2B_i], \\ P[X - x_i > B - 2B_i] &= \frac{1}{1 - p_i} P[X > B - 2B_i] - \frac{p_i}{1 - p_i} P[X - x_i > B - 3B_i], \dots \end{aligned}$$

Po wykonaniu odpowiednich podstawień:

$$\begin{aligned}
 P[X - x_i > B] &= \frac{1}{1-p_i} P[X > B] - \frac{p_i}{1-p_i} P[X - x_i > B - B_i] = \\
 &= \frac{1}{1-p_i} P[X > B] - \frac{1}{1-p_i} \frac{p_i}{1-p_i} P[X > B - B_i] + \left(\frac{p_i}{1-p_i}\right)^2 P[X - x_i > B - 2B_i] = \\
 &= \frac{1}{1-p_i} P[X > B] - \frac{1}{1-p_i} \frac{p_i}{1-p_i} P[X > B - B_i] - \\
 &\quad - \frac{1}{1-p_i} \left(\frac{p_i}{1-p_i}\right)^2 P[X > B - 2B_i] - \left(\frac{p_i}{1-p_i}\right)^3 P[X - x_i > B - 3B_i] = \dots
 \end{aligned}$$

$$P[X - x_i > B] = \frac{1}{1-p_i} \sum_{l=1}^{k-1} \left(\frac{-p_i}{1-p_i}\right)^l P[X > C - lC_i] + \left(\frac{-p_i}{1-p_i}\right)^k P[X > C - kC_i],$$

gdzie: $k = \lfloor B/B_i \rfloor$, a $\lfloor a \rfloor$ oznacza największą liczbę całkowitą mniejszą od liczby a .

Ogólna zasada działania algorytmu FBR jest następująca:

1. Każda transmisja jednostki danych (np. komórki) z pamięci buforowej oznacza pomniejszenie liczby zajętych szczelin b_i o 1.
2. Jeżeli i -te źródło jest w stanie ciszy, to:
 - 2.1. jeżeli napływająca jednostka danych jest zgłoszeniem startu strumienia jednostek danych i -tego źródła (rozpoczynającym okres aktywności i -tego źródła), to:
 - 2.1.1. jeżeli wartość prawdopodobieństwa nadmiaru jednostek danych p_e (lub wartość prawdopodobieństwa rywalizacji p_{ir}) jest większa od założonych wartości \bar{p}_e (lub \bar{p}_{ir}), to strumień jednostek danych jest odrzucany,
 - 2.1.2. jeżeli wartość prawdopodobieństwa nadmiaru jednostek danych p_e (lub wartość prawdopodobieństwa rywalizacji p_{ir}) jest mniejsza od założonych wartości \bar{p}_e (lub \bar{p}_{ir}), to strumień jednostek danych jest akceptowany,
 - 2.1.2.1. następuje ustawienia zegara,
 - 2.1.2.2. zmieniana jest wartość wskaźnika stanu i -tego źródła $s_i = 1$ (stan aktywności),
 - 2.1.2.3. aktualizowana jest liczba dostępnych szczelin pamięci buforowej $B_a = B_a - B_i$, oraz
 - 2.1.2.4. jeżeli $b_i < B_i$, to:
 - 2.1.2.4.1. następuje zmiana wartości liczby zajętych szczelin $b_i = b_i + 1$,
 - 2.1.2.4.2. dana jednostka danych jest umieszczana w szczelinie pamięci buforowej jako jednostka danych nieoznaczona (ang. *unmarked*),
 - 2.1.2.5. jeżeli $b_i = B_i$, to dana jednostka danych jest umieszczana w szczelinie pamięci buforowej jako jednostka danych oznaczona (ang. *marked*),

- 2.2. jeżeli napływająca jednostka danych nie jest jednostką danych specyfikującą nowy strumień jednostek danych i -tego źródła, to jednostka danych jest odrzucana.
3. Jeżeli i -te źródło jest w stanie aktywności, to:
 - 3.1. jeżeli napływająca jednostka danych nie jest końcową jednostką danych strumienia, to:
 - 3.1.1. następuje ponowne ustawienie (ang. *reset*) zegara, oraz:
 - 3.1.2. jeżeli $b_i < B_i$, to:
 - 3.1.2.1. następuje zmiana wartości liczby zajętych szczelin $b_i = b_i + 1$,
 - 3.1.2.2. dana jednostka danych jest umieszczana w szczelinie pamięci buforowej jako nieoznaczona,
 - 3.1.3. jeżeli $b_i = B_i$, to dana jednostka danych jest umieszczana w szczelinie pamięci buforowej jako nieoznaczona,
 - 3.2. lub jeżeli napływająca jednostka danych jest końcową jednostką danych w danym strumieniu jednostek danych (oznaczającą koniec okresu aktywności i -tego źródła) lub stan zegara jest zerowy, to:
 - 3.2.1. zmieniana jest wartość wskaźnika stanu źródła $s_i = 0$ (stanu ciszy),
 - 3.2.2. aktualizowana jest liczba dostępnych szczelin pamięci buforowej $B_a = B_a + B_i$.

Decyzja o przyjęciu lub odrzuceniu nowego wywołania jest podejmowana na podstawie obliczanej wartości prawdopodobieństwa przekroczenia zasobów (ang. *excess demand probability*); prawdopodobieństwa, że żądana liczba szczelin pamięci buforowej jest większa od dostępnej. Jeżeli obliczona dla nowego wywołania wartość tego prawdopodobieństwa jest większa od wcześniej założonej (ustalanej) wartości, to wywołanie jest odrzucane, w innym przypadku jest akceptowane.

Różne odmiany tego algorytmu, stosowanego np. do sterowania ruchem w sieci ATM, uzyskuje się przez wprowadzenie czterech (wykorzystanie dwóch bitów w nagłówku komórki ATM) typów zgłoszeń oznaczonych (ang. *marked*). Oprócz zgłoszeń startu są to: zgłoszenia pośrednie (ang. *middle*), końcowe (ang. *end-of-the burst*) oraz pojedyncze (ang. *loner*). Te ostatnie są wykorzystywane w przypadku transmisji strumieni jednogłoszeniowych lub do celów specyfikacji priorytetów zgłoszeń.

Wprowadzenie różnych typów zgłoszeń pozwala na modyfikację algorytmu umożliwiającego np. efektywną obsługę źródeł strumieni jednostek danych wrażliwych na straty lub tolerujących straty.

W przypadku strumieni jednostek danych wrażliwych na straty jednostek danych (ang. *loss-sensitive*), tzn. strumieni, dla których strata jednostek danych wiąże się z koniecznością retransmisji części lub wszystkich jednostek danych generowanych w okresie aktywności źródła, wszystkie transmitowane jednostki danych są oznaczone. Zgłoszeniu startowemu towarzyszą zgłoszenia pośrednie, a pojedynczy ciąg zgłoszeń, odpowiadający przedziałowi aktywności źródła, jest zakończony zgłoszeniem końcowym. Jeżeli żądana przez zgłoszenie startowe pamięć buforowa jest niedostępna, to wszystkie jednostki danych należące do pojedynczego ciągu są tracone.

W przypadku strumieni tolerujących straty (ang. *loss-tolerant*) wszystkie transmitowane jednostki danych, oprócz zgłoszenia końcowego, są zgłoszeniami startowymi.

Jeżeli nawet jedna z jednostek danych będzie stracona z powodu braku szczeliny pamięci buforowej, to pozostałe nie tracą możliwości wejścia do pamięci buforowej.

O jakości algorytmu FBR decyduje możliwość obliczania wartości prawdopodobieństwa przekroczenia zasobów, dokładność obliczeń oraz założona wartość prawdopodobieństwa (parametr algorytmu), z którą porównuje się obliczaną w czasie rzeczywistym wartość prawdopodobieństwa przekroczenia zasobów.

7.2.2.4.3. ALGORYTM EC

W algorytmie równoważnej pojemności EC (ang. *Equivalent Capacity*) zastosowano dwustanowy model przepływu ze skończoną pojemnością pamięci buforowej, w którym źródła strumieni jednostek danych są obsługiwane przez łącze o stałej szybkości obsługi jednostek danych. Połączenie charakteryzowane jest przez dwustanowy model, w którym strumień jednostek danych jest generowany z maksymalną, szczytową szybkością w okresie aktywności źródła, w okresie ciszy natomiast nie są generowane żadne jednostki danych.

Dynamikę kolejki w tym modelu przepływu, dla danych szybkości napływu i obsługi jednostek danych, definiuje się następująco:

1. Jeżeli szybkość napływu jednostek danych jest mniejsza od szybkości obsługi jednostek danych, to:

- gdy pamięć buforowa jest pusta, pozostaje pusta,
- gdy pamięć buforowa nie jest pusta, jej zawartość maleje ze stałą szybkością proporcjonalną do różnicy szybkości obsługi i szybkości napływu.

2. Jeżeli szybkości napływu i obsługi jednostek danych są sobie równe, to zajętość (stan) pamięci buforowej nie zmienia się.

3. Jeżeli szybkość napływu jednostek danych jest większa od szybkości obsługi zgłoszeń, to:

- gdy pamięć buforowa nie jest wypełniona, tzn. całkowicie zajęta, wówczas jej zajętość zwiększa się ze stałą szybkością, proporcjonalną do różnicy szybkości obsługi i szybkości napływu,
- gdy pamięć buforowa jest wypełniona, napływające jednostki danych są odrzucane w ilości proporcjonalnej do różnicy szybkości obsługi i szybkości napływu.

Przy założeniu, że:

- $v_{i \max}$ – wartość szczytowa szybkości generowania jednostek danych przez i -te źródło,
- \bar{v}_i – wartość średnia szybkości generowania zgłoszeń i -tego źródła,
- a_i – średni czas trwania okresu aktywności i -tego źródła,
- u_i – współczynnik aktywności i -tego źródła (prawdopodobieństwo, że i -te źródło jest w stanie aktywności),
- η_i – szybkość przełączania ze stanu aktywności i -tego źródła ($\zeta_i = 1/a_i$),
- ζ_i – szybkość przełączania ze stanu ciszy i -tego źródła ($\eta_i = u_i/(a_i(1 - u_i))$),
- B_i – pojemność pamięci buforowej przydzielonej do obsługi strumienia zgłoszeń generowanego przez i -te źródło,

C_i – pojemność łącza wyjściowego do obsługi strumienia zgłoszeń generowanego przez i -te źródło,

oraz że:

➤ każde i -te połączenie jest opisywane dwustanowym modelem: w stanie aktywności zgłoszenia generowane są z szybkością szczytową $v_{i \max}$, w stanie ciszy natomiast szybkość generowania jest zerowa,

➤ czasy trwania okresów aktywności i ciszy mają rozkład wykładniczy i są niezależne,

każde połączenie, charakteryzowane przez trzy wielkości – szybkość szczytową i szybkość średnią napływu jednostek danych oraz średni czas przebywania źródła w stanie aktywności – jest obsługiwane przez łącze wyjściowe, którego zdolność do obsługi (potencjał transmisyjny) charakteryzuje pojemność pamięci buforowej i stała przepustowość łącza wyjściowego.

Wartość żądanej pojemności łącza wyjściowego C_i , tzw. równoważnej pojemności połączenia (ang. *equivalent capacity*), obliczanej niezależnie dla każdego nowego wywołania, jest rozwiązaniem zadania polegającego na wyznaczeniu takiej szybkości obsługi jednostek danych, dla której wartość prawdopodobieństwa przepełnienia pamięci buforowej nie jest większa od wartości zadanej.

Jeżeli $p_{ij}(t, x_i)$ jest prawdopodobieństwem zdarzenia polegającego na tym, że i -te dwustanowe źródło w chwili t jest w stanie j ($j = 1$ – stan aktywności, $j = 0$ – stan ciszy), a zajętość pamięci buforowej łącza wyjściowego jest równa b_i , to omawiany system kolejkowy określają następujące równania:

$$p_{i0}(t + dt, b_i) = (1 - \eta_i dt) p_{i0}(t, b_i + C_i dt) + \zeta_i dt p_{i1}(t, b_i + (C_i - v_{i \max}) dt),$$

$$p_{i1}(t + dt, b_i) = \eta_i dt p_{i0}(t, b_i + C_i dt) + (1 - \zeta_i dt) p_{i1}(t, b_i + (C_i - v_{i \max}) dt),$$

których przekształcenie prowadzi do następujących równań różniczkowych:

$$\frac{\partial p_{i0}(t, b_i)}{\partial t} - C_i \frac{\partial p_{i0}(t, b_i)}{\partial b_i} = -\eta_i p_{i0}(t, b_i) + \zeta_i p_{i1}(t, b_i),$$

$$\frac{\partial p_{i1}(t, b_i)}{\partial t} - (C_i - v_{i \max}) \frac{\partial p_{i1}(t, b_i)}{\partial b_i} = \eta_i p_{i0}(t, b_i) - \zeta_i p_{i1}(t, b_i).$$

W stanie ustalonym, w którym prawdopodobieństwa stanów kolejki są niezależne od czasu, powyższe równania przyjmują postać układu równań:

$$-C_i \frac{\partial p_{i0}(b_i)}{\partial b_i} = -\eta_i p_{i0}(b_i) + \zeta_i p_{i1}(b_i),$$

$$-(C_i - v_{i \max}) \frac{\partial p_{i1}(b_i)}{\partial b_i} = \eta_i p_{i0}(b_i) - \zeta_i p_{i1}(b_i),$$

gdzie: $p_{ij}(b_i) = \lim_{t \rightarrow \infty} (p_{ij}(t, b_i))$ dla $j \in \{0, 1\}$.

Po wprowadzeniu oznaczeń:

$$\mathbf{p}_i(b_i) = [p_{i0}(b_i), p_{i1}(b_i)]^T,$$

$$\mathbf{p}'_i(b_i) = \left[\frac{\partial p_{i0}(b_i)}{\partial b_i}, \frac{\partial p_{i1}(b_i)}{\partial b_i} \right],$$

układ równań można zapisać jako:

$$\begin{bmatrix} -C_i & 0 \\ 0 & -(C_i - v_{i\max}) \end{bmatrix} \mathbf{p}'_i(b_i) = \begin{bmatrix} -\eta_i & \zeta_i \\ \eta_i & -\zeta_i \end{bmatrix} \mathbf{p}_i(b_i).$$

Rozwiązanie układu równań ma postać:

$$\begin{aligned} \mathbf{p}_i(b_i) &= k_0 \begin{bmatrix} \zeta_i \\ \eta_i \end{bmatrix} + k_1 \begin{bmatrix} (v_{i\max} - C_i)/C_i \\ 1 \end{bmatrix} \exp\left(b_i \frac{\zeta_i(C_i - v_{i\max}) + \eta_i C_i}{(C_i - v_{i\max})C_i} \right) = \\ &= k_0 \begin{bmatrix} 1/a_i \\ u_i/a_i(1-u_i) \end{bmatrix} + k_1 \begin{bmatrix} (v_{i\max} - C_i)/C_i \\ 1 \end{bmatrix} \exp\left(-b_i \frac{C_i - (1-u_i)v_{i\max}}{a_i(1-u_i)(v_{i\max} - C_i)C_i} \right), \end{aligned}$$

gdzie wartości współczynników k_0 i k_1 są obliczane dla warunków brzegowych.

Jeżeli szybkość napływu jednostek danych jest większa od szybkości obsługi jednostek danych, tzn. jeżeli spełniona jest nierówność $v_{i\max} > C_i$, to warunki brzegowe definiuje się następująco:

➤ prawdopodobieństwo zdarzenia, że zajętość pamięci buforowej b_i jest mniejsza od pojemności pamięci buforowej B_i jest równe prawdopodobieństwu, że i -te źródło znajduje się w okresie ciszy, tzn.:

$$p_{i0}(b_i < B_i) = k_0 \frac{1}{a_i} + k_1 \frac{v_{i\max} - C_i}{C_i} \exp\left(-B_i \frac{C_i - (1-u_i)v_{i\max}}{a_i(1-u_i)(v_{i\max} - C_i)C_i} \right) = 1 - u_i,$$

➤ prawdopodobieństwo zdarzenia, że zajętość pamięci buforowej b_i jest równa zero wtedy, gdy i -te źródło znajduje się w okresie aktywności, jest równe zero, tzn.:

$$p_{i1}(b_i = 0) = k_0 \frac{u_i}{a_i(1-u_i)} + k_1 = 0.$$

Wartości współczynników k_0 i k_1 , będących rozwiązaniem równań wyprowadzonych dla warunków brzegowych, dane są wyrażeniami:

$$k_0 = \frac{a_i(1-u_i)^2 C_i}{\Delta},$$

$$k_1 = \frac{-u_i(1-u_i)C_i}{\Delta},$$

w których:

$$\Delta = (1-u_i)C_i - u_i(v_{i\max} - C_i) \exp\left(-B_i \frac{C_i - (1-u_i)v_{i\max}}{a_i(1-u_i)(v_{i\max} - C_i)C_i}\right).$$

Rozkład prawdopodobieństwa długości kolejki jednostek danych, oczekujących na obsługę:

$$p_i(q \leq b_i) = p_{i0}(b_i) + p_{i1}(b_i),$$

jest określony wyrażeniem:

$$p_i(q \leq b_i) = \begin{cases} \frac{(1-u_i)C_i}{\Delta} - \frac{u_i(1-u_i)v_{i\max}}{\Delta} \exp\left(-B_i \frac{C_i - (1-u_i)v_{i\max}}{a_i(1-u_i)(v_{i\max} - C_i)C_i}\right) & \text{dla } b_i < B_i \\ 1 & \text{w in. przyp.} \end{cases}.$$

Prawdopodobieństwo przepełnienia $p_{i0}(B_i)$ pamięci buforowej o pojemności B_i jest równe prawdopodobieństwu zdarzenia, że i -te źródło jest w okresie aktywności i pamięć buforowa jest całkowicie zajęta, które jest uzupełnieniem zdarzenia, że i -te źródło jest w okresie aktywności (prawdopodobieństwo u_i) i zajętość pamięci buforowej jest mniejsza od B_i (prawdopodobieństwo $p_{i1}(b_i < B_i)$), tzn.:

$$u_i = p_{i0}(B_i) + p_{i1}(b_i < B_i),$$

gdzie:

$$p_{i1}(b_i < B_i) = \frac{u_i(1-u_i)C_i}{\Delta} - \frac{u_i(1-u_i)C_i}{\Delta} \exp\left(-B_i \frac{C_i - (1-u_i)v_{i\max}}{a_i(1-u_i)(v_{i\max} - C_i)C_i}\right),$$

z czego wynika, że prawdopodobieństwo przepełnienia pamięci buforowej $p_{i0}(B_i)$ jest dane wyrażeniem:

$$p_{i0}(B_i) = u_i - p_{i1}(b_i < B_i) = \frac{u_i(C_i - u_i v_{i\max}) \exp\left(-B_i \frac{C_i - (1-u_i)v_{i\max}}{a_i(1-u_i)(v_{i\max} - C_i)C_i}\right)}{(1-u_i)C_i - u_i(v_{i\max} - C_i) \exp\left(-X_i \frac{C_i - (1-u_i)v_{i\max}}{a_i(1-u_i)(v_{i\max} - C_i)C_i}\right)}.$$

Wartość tzw. równoważnej pojemności i -tego połączenia C_i z pamięcią buforową o pojemności B_i jest taką pojemnością łącza, która dla wejściowego strumienia jednostek danych, charakteryzowanego parametrami $v_{i\max}$, \bar{v}_i oraz a_i , gwarantuje żadaną wartość prawdopodobieństwa przepełnienia p_{ie} pamięci buforowej B_i . Po przyjęciu, że:

$$p_{ie} u_i = p_{i0}(B_i),$$

$$\delta_i = \frac{C_i - (1-u_i)v_{i\max}}{a_i(1-u_i)(v_{i\max} - C_i)C_i},$$

wyrażenie na żadaną wartość prawdopodobieństwa przepełnienia pojemności buforowej B_i przyjmuje postać:

$$p_{ie} = \frac{(C_i - u_i v_{i\max}) \exp(-\delta_i B_i)}{(1 - u_i) C_i - u_i (v_{i\max} - C_i) \exp(-\delta_i B_i)}$$

Warunek, którego spełnienie powoduje, że prawdopodobieństwo przepełnienia pamięci buforowej jest mniejsze od danej wartości p_{ie} , jest następujący:

$$\exp(-\delta_i B_i) \leq \frac{p_{ie} (1 - u_i) C_i}{(C_i - u_i v_{i\max}) + p_{ie} u_i (v_{i\max} - C_i)},$$

gdzie C_i jest poszukiwaną wartością równoważnej pojemności i -tego połączenia.

Analityczne i dokładne wyznaczenie wartości C_i nie jest możliwe ze względu na złożoną postać nierówności. Z kolei numeryczne wyznaczenie wartości C_i jest czasochłonne; jest nie do przyjęcia w zadaniu sterowania przyjmowaniem wywołań ze względu na konieczność wyznaczania wartości C_i w czasie rzeczywistym. Z podanych powodów, zamiast wyznaczania dokładnej wartości C_i , wyznaczane jest górne ograniczenie wartości C_i . Warunkiem możliwości wyznaczenia górnego ograniczenia wartości C_i jest przyjęcie założenia, że:

$$\frac{(1 - u_i) C_i}{(C_i - u_i v_{i\max}) + p_{ie} u_i (v_{i\max} - C_i)} = 1,$$

które prowadzi do następującego uproszczenia:

$$\exp(-\delta_i B_i) \leq p_{ie} \frac{(1 - u_i) C_i}{(C_i - u_i v_{i\max}) + p_{ie} u_i (v_{i\max} - C_i)} = p_{ie}.$$

Z tego uproszczenia wynika, że:

$$\exp(-\delta_i B_i) \leq p_{ie} \Rightarrow \delta_i B_i \leq \ln \frac{1}{p_{ie}} \Rightarrow \frac{C_i - (1 - u_i) v_{i\max}}{a_i (1 - u_i) (v_{i\max} - C_i) C_i} B_i \leq \ln \frac{1}{p_{ie}} = \beta.$$

Rozwiązanie równania kwadratowego:

$$(C_i - (1 - u_i) v_{i\max}) B_i = \beta a_i (1 - u_i) (v_{i\max} - C_i) C_i \Rightarrow$$

$$\Rightarrow \beta a_i (1 - u_i) C_i^2 + [B_i - \beta a_i (1 - u_i) v_{i\max}] C_i - (1 - u_i) v_{i\max} B_i = 0,$$

ma następującą postać:

$$C_i = \frac{\beta a_i (1 - u_i) v_{i\max} - B_i + \sqrt{(\beta a_i (1 - u_i) v_{i\max} - B_i)^2 + 4 \beta a_i (1 - u_i)^2 v_{i\max} B_i}}{2 \beta a_i (1 - u_i) v_{i\max}}.$$

Wartość górnego ograniczenia równoważna pojemności dla i -tego połączenia (i -tego źródła) C_i zależy przede wszystkim od wartości pojemności pamięci buforowej B_i przydzielonej na potrzeby obsługi strumienia zgłoszeń $(v_{i\max}, \bar{v}_i, a_i)$ generowanego przez i -te źródło. Wartości C_i zależą od strategii podziału pamięci buforowej i dla różnych strategii podziału mogą się znacznie różnić.

Równoważna pojemność połączenia zagregowanego C_0 jest sumą równoważnych pojemności połączeń składowych połączenia zagregowanego:

$$C_0 = C_1 + C_2 + \dots + C_n,$$

gdzie n jest liczbą wszystkich połączeń składowych połączenia zagregowanego.

Tak obliczona wartość równoważnej pojemności połączenia zagregowanego C_0 może znacznie przekraczać rzeczywiste zapotrzebowanie na przepustowość, dlatego że równoważne pojemności połączeń składowych (C_1, C_2, \dots, C_n) są wyznaczone z pominięciem wzajemnych zależności pomiędzy poszczególnymi połączeniami składowymi.

Obliczanie równoważnej pojemności połączenia C_i dla danego i -tego wywołania może być interpretowane jako niezależny (od innych obsługiwanych połączeń) przydział maksymalnej części pojemności łącza, dla której prawdopodobieństwo przekroczenia pojemności pamięci buforowej B_i jest poniżej zadanej wartości p_{ie} . Odpowiada to dążeniu, przez przydział pojemności łącza, do minimalizacji czasu obsługi danego połączenia przy zadanych ograniczeniach pojemności pamięci buforowej.

Algorytm EC nie wykorzystuje ani nie wspiera statystycznych właściwości połączenia zagregowanego, nie uwzględnia rozłożenia w czasie okresów aktywności i ciszy źródeł generujących ruch w poszczególnych połączeniach składowych. Równoważna pojemność połączenia zagregowanego jest skumulowaną w czasie sumą wartości równoważnych pojemności połączeń składowych.

W praktyce algorytm EC jest stosowany w połączeniu z algorytmem GA, który niweluje nadmiarowość algorytmu EC w następujący sposób:

Dla każdego nowego ($n + 1$)-go połączenia (wywołania) składowego:

➤ zgodnie z algorytmem EC wyznaczana jest suma równoważnych pojemności: nowego połączenia składowego C_{n+1} oraz istniejącego n -elementowego połączenia zagregowanego C_0 , tzn.:

$$C_{EC} = C_0 + C_{n+1} = (C_1 + C_2 + \dots + C_n) + C_{n+1},$$

➤ zgodnie z algorytmem GA wyznaczana jest suma zapotrzebowania na pojemność nowego połączenia składowego i istniejącego n -elementowego połączenia zagregowanego, tzn.:

$$C_{GA} = m + m_{n+1} + \alpha \sqrt{\sigma^2 + \sigma_{n+1}^2},$$

gdzie:

$$m = m_1 + m_2 + \dots + m_n,$$

$$\sigma^2 = \sigma_1^2 + \sigma_2^2 + \dots + \sigma_n^2,$$

$$\alpha = \sqrt{2 \ln \frac{1}{p_e} - \ln 2\pi},$$

$$m_i = v_{i \max} a_i \text{ i } \sigma_i^2 = m_i (v_{i \max} - m_i) \text{ dla } i = 1, 2, \dots, n.$$

Jako pojemność C'_0 , tzn. równoważna pojemność połączenia zagregowanego, złożonego z $(n + 1)$ połączeń, przyjmowana jest mniejsza z dwóch obliczonych wartości pojemności:

$$C'_0 = \min(C_{EC}, C_{GA}).$$

Jeżeli tak obliczona i wybrana wartość C'_0 równoważnej pojemności jest mniejsza od pojemności łącza, to nowe $(n + 1)$ wywołanie jest przyjmowane do obsługi; w innym przypadku nowe $(n + 1)$ wywołanie jest odrzucane.

Algorytm EC można traktować jako modyfikację algorytmu GA i odwrotnie. Działanie algorytmu GA, niwelujące nadmiarowość przydziału zasobów przez algorytm EC, polega na wykorzystaniu statystycznych właściwości połączenia zagregowanego, z których wynika, że prawdopodobieństwo jednoczesnego żądania maksymalnej pojemności przez wszystkie połączenia składowe jest bardzo małe, tzn. żądania pojemności przez połączenia składowe są rozproszone w czasie. Rozproszenie w czasie żądań dostępu połączeń składowych połączenia zagregowanego do maksymalnych pojemności umożliwia taki przydział pojemności połączeniom składowym, że suma przydzielonych pojemności jest większa od całkowitej pojemności łącza wyjściowego.

Działanie algorytmu EC charakteryzuje względnie wysoki stopień statystycznego zwielokrotniania przy jednoczesnej gwarancji zachowania żądanych wartości prawdopodobieństwa odrzucania zgłoszeń.

7.2.2.4.4. ALGORYTM FA

Algorytm aproksymacji przepływu FA (ang. *Flow Approximation to loss rate*) jest zorientowany na obsługę źródeł strumieni komórek ATM o zmiennej szybkości, tj. VBR (ang. *Variable Bit Rate*).

Sterowanie dostępem zgodnie z algorytmem aproksymacji przepływu dla potrzeb szacowania szybkości utraty komórek ATM polega na tym, że decyzje o akceptacji lub odrzucaniu żądań dostępu wywołań podejmowane są na podstawie szacowanych wartości szybkości utraty komórek.

Stosowany w algorytmie aproksymacji przepływu FA model łącza obsługującego zagregowane połączenie, złożone z połączeń o znanych wartościach szczytowych i średnich szybkości napływu komórek, jest następujący: sumaryczny strumień komórek napływa do kolejki, z której zgłoszenia wypływają z szybkością równą pojemności łącza. Jeżeli sumaryczna szybkość napływu komórek do kolejki jest większa od szybkości wypływu, to kolejka wypełnia się i napływające zgłoszenia są odrzucane. Szybkość odrzucania komórek CLR (ang. *Cell Loss Rate*) jest proporcjonalna do różnicy szybkości napływu i wypływu komórek strumienia sumarycznego oraz od stanu początkowego

kolejki, tj. ilości niezajętej pamięci buforowej. Na podstawie szybkości odrzucania komórek szacowana jest wartość prawdopodobieństwa odrzucenia (utruty) komórek.

Podczas realizacji algorytmu aproksymacji przepływu FA korzysta się z minimalnej ilości informacji, w porównaniu z ilością informacji zwykle stosowaną do opisu źródeł o zmiennej szybkości generowania ruchu. Konsekwencją ograniczenia ilości informacji jest względnie uproszczony model źródeł strumienia ruchu.

Do opisu strumienia ruchu generowanego przez i -te ($i = 1, 2, \dots, n$) źródło ruchu VBR (ang. *Variable Bit Rate*) przyjmuje się:

$v_{i \max}$ – wartość szczytowa szybkości generowania komórek,

\bar{v}_i – wartość średnia szybkości generowania komórek,

p_{ai} – wartość prawdopodobieństwa aktywności źródła generowanego ruchu:

$$p_{ai} = \bar{v}_i / v_{i \max},$$

p_{ci} – wartość prawdopodobieństwa aktywności ciszy źródła generowanego ruchu:

$$p_{ci} = 1 - (\bar{v}_i / v_{i \max}).$$

Sumaryczny strumień jednostek danych, generowanych przez n niezależnych źródeł ruchu, przyjmowanych do pamięci buforowej o całkowitej pojemności B i obsługiwanych przez łącze o pojemności C , charakteryzowany jest wartościami zmiennych losowych:

v_i ($i = 1, 2, \dots, n$) – oznaczających szybkość strumienia jednostek danych, generowanego przez i -te źródło,

v – oznaczającej szybkość zagregowanego strumienia jednostek danych, złożonego ze strumieni jednostek danych, generowanych przez n źródeł, tzn. $v = v_1 + v_2 + \dots + v_n$.

W omawianym algorytmie obsługi strumieni ruchu ciągły strumień jednostek danych o sumarycznej szybkości v napływa do kolejki organizowanej w pamięci buforowej i wypływa z szybkością C . Przy założeniu, że B_a jest liczbą dostępnych buforów pamięci buforowej o pojemności B ($B_a \leq B$), szybkość z jaką są tracone (odrzucone) napływające jednostki danych v_L określa wyrażenie:

$$v_L = \sum_{z > C} (z - C) P(v = z) [1 - P(B_a > 0 | v = z)],$$

tzn. że jeżeli $v = z > C$ i $B_a = 0$, to szybkość z jaką są tracone jednostki danych, bez względu na to, z którego źródła ruchu pochodzą, jest równa $v_L = z - C = v - C$.

Podstawowa idea algorytmu aproksymacji przepływu FA wynika z następującego spostrzeżenia: jeżeli długości okresów aktywności źródeł rosną, podczas gdy maksymalne i średnie szybkości napływu jednostek danych pozostają stałe, to prawdopodobieństwo, że liczba dostępnych buforów B_a jest większa od zera, tj. $P(B_a > 0 | v = z)$ maleje do zera, jeżeli szybkość napływu zgłoszeń zagregowanego strumienia zgłoszeń jest większa od pojemności kanału transmisyjnego, tzn. dla $z > C$. Na podstawie tego spostrzeżenia w miejsce wyrażenia na szybkość tracenia jednostek danych v_L można wprowadzić górne ograniczenie szybkości $\sup v_L$, z jaką jednostki danych należące do

zagregowanego strumienia są tracone ze względu na długość czasu trwania aktywności źródeł. Górne ograniczenie szybkości traceniam jednostek danych $\sup v_L$ jest dane zależnością:

$$\sup v_L = \sum_{z>C} (z - C) P(v = z) = \max_{p_{ai} \rightarrow 1} \sum_{z>C} (z - C) P(v = z) [1 - P(P_a > 0 | v = z)].$$

Wartość górnego ograniczenia szybkości traceniam jednostek danych $\sup v_L$ pozwala na zdefiniowanie górnego ograniczenia wartości prawdopodobieństwa odrzucania zgłoszeń, określonego wyrażeniem:

$$p_L = \frac{\sup v_L}{\bar{v}_1 + \bar{v}_2 + \dots + \bar{v}_n},$$

którego wartość jest wykorzystywana do podejmowania decyzji o akceptacji lub odrzuceniu nowego wywołania. Możliwość korzystania z wartości prawdopodobieństwa p_L do podejmowania decyzji o akceptacji lub odrzuceniu nowego wywołania wymaga wyrażenia wartości p_L w zależności od, przyjętych jako znane, wartości parametrów opisujących źródła zgłoszeń, tj. wartości szczytowych $v_{i \max}$ i średnich \bar{v}_i szybkości generowania zgłoszeń oraz wartości prawdopodobieństwa aktywności p_{ai} i ciszy p_{ci} źródeł ($i = 1, 2, \dots, n$).

Po oznaczeniu przez Ω ($\Omega = \{\Omega_k\}$) zbioru wszystkich zbiorów indeksów źródeł (połączeń) Ω_k ($\Omega_k = \{i_1, i_2, \dots, i_k\}$), które spełniają warunek:

$$v_{i_1 \max} + v_{i_2 \max} + \dots + v_{i_k \max} > C,$$

dla zmiennych losowych v_i ($i = 1, 2, \dots, n$) przyjmujących wartości ze zbiorów dwuelementowych, tj. $v_i \in \{0, v_{i \max}\}$, wyrażenie na górne ograniczenie szybkości traceniam jednostek danych $\sup p_L$ przyjmuje postać:

$$\sup v_L = \sum_{\bar{U}_k \in \bar{U}} \left[\left[\sum_{j \in \bar{U}_k} v_{j \max} \right] \prod_{j \in \bar{U}_k} \frac{\bar{v}_j}{v_{j \max}} \prod_{l \in \bar{U}_k} \left(1 - \frac{\bar{v}_l}{v_{l \max}} \right) \right],$$

gdzie \bar{U}_k są zbiorami indeksów spełniających warunki: $\bar{U}_k \cup \hat{U}_k = \{1, 2, \dots, n\}$ dla każdego k . Gdy $\bar{U}_k = \emptyset$, wartość produktu prawdopodobieństwa przebywania źródeł w stanie ciszy równa jest 1.

Wartość wyrażenia na $\sup v_L$ zależy tylko od wartości szczytowych i średnich szybkości generowania zgłoszeń, tzn. $v_{i \max}$ i \bar{v}_i , ze względu na to, że wartości prawdopodobieństw p_{ai} i p_{ci} są definiowane z wykorzystaniem wartości $v_{i \max}$ i \bar{v}_i .

Wartość $\sup v_L$ może być wykorzystana do podejmowania decyzji o akceptacji lub odrzuceniu nowego wywołania. Przydatność tego wyrażenia do podejmowania decyzji jest ograniczona ze względu na jego złożoność obliczeniową, tj. ze względu na czas potrzebny do obliczenia dokładnej wartości $\sup v_L$. Przydatność omawianego wyrażenia maleje wraz ze wzrostem liczby wszystkich jednocześnie obsługiwanych n stru-

mieni składowych strumienia sumarycznego. Na potrzeby podejmowania decyzji w czasie rzeczywistym wymagany jest taki algorytm szacowania wartości p_L , który umożliwi uzyskanie oszacowania w czasie determinowanym koniecznością podejmowania decyzji w czasie rzeczywistym.

Jednym z możliwych rozwiązań, spełniających zarówno wymagania dotyczące dokładności, jak i czasu obliczania przybliżeń, jest obliczanie przybliżeń wartości $\sup v_L$, a tym samym przybliżeń wartości prawdopodobieństwa p_L , według następującej procedury:

Wartość górnego ograniczenia $\sup v_L$ jest określona wyrażeniem:

$$\sup v_L = -\int_C^{v_{\max}} (z - C) dP(v > z) = \int_C^{v_{\max}} P(v > z) dz,$$

w którym v_{\max} jest sumą wartości szczytowych szybkości generowania jednostek danych przez wszystkie źródła strumieni składowych strumienia sumarycznego, tj.

$$v_{\max} = v_{1\max} + v_{2\max} + \dots + v_{n\max}.$$

Przy założeniu, że wartość prawdopodobieństwa $P(v > z)$ jest ograniczona z góry – zgodnie z ograniczeniem Chernoffa – wartością wyrażenia $E(e^{sv})/e^{sz}$, gdzie s jest pewną liczbą dodatnią ($s > 0$), wartość wyrażenia $\sup p_L$ dla $v_{\max} \rightarrow \infty$ spełnia warunek:

$$\sup v_L = \int_C^{v_{\max}} P(v > z) dz < E(e^{sv}) \int_C^{v_{\max}} e^{-sz} dz < \frac{E(e^{sv})}{se^{sC}} = f(s).$$

Funkcja $f(s)$ jest funkcją wypukłą i ma minimum $s = s^*$, będące pierwiastkiem następującego równania:

$$\frac{\partial \ln E(e^{sv})}{\partial s} - \frac{1}{s} - C = 0.$$

Ponieważ:

- wartość zmiennej losowej v jest sumą wartości zmiennych losowych v_i ($i = 1, 2, \dots, n$),
- wartość zmiennej losowej v_i jest równa:

$v_i = v_{i\max}$ z prawdopodobieństwem $p_{ia} = \bar{v}_i/v_{i\max}$ oraz

$v_i = 0$ z prawdopodobieństwem $p_{ic} = 1 - (\bar{v}_i/v_{i\max})$,

wartość wyrażenia $E(e^{sv})$ jest więc następująca:

$$\begin{aligned} E(e^{sv}) &= \prod_{i=1}^n E(e^{sv_i}) = \prod_{i=1}^n \left(e^{sv_{i\max}} \frac{\bar{v}_i}{v_{i\max}} + 1 - \frac{\bar{v}_i}{v_{i\max}} \right) = \\ &= \prod_{i=1}^n \left(\frac{\bar{v}_i}{v_{i\max}} (e^{sv_{i\max}} - 1) + 1 \right). \end{aligned}$$

Wartość $s = s^*$, minimalizująca wartość $E(e^{sv})$, jest pierwiastkiem wyrażenia:

$$\frac{\partial \ln E(e^{sv})}{\partial s} - s^{-1} - C = \sum_{i=1}^n \left(\frac{\bar{v}_i e^{s v_{i \max}}}{\bar{v}_i / v_{i \max}} (e^{s v_{i \max}} - 1) + 1 \right) - s^{-1} - C = 0.$$

Wartość górnego ograniczenia prawdopodobieństwa odrzucenia zgłoszeń p_L , oznaczonego przez \bar{p}_L , dla znanej – wyznaczonej wcześniej – wartości pierwiastka s^* jest dana wyrażeniem:

$$p_L < \bar{p}_L = \frac{\sup v_L}{\sum_{i=1}^n \bar{v}_i} = \frac{E(e^{s^* v})}{s^* e^{s^* C}} = \frac{E(e^{s^* v})}{s^* e^{s^* C} \sum_{i=1}^n \bar{v}_i} = \prod_{i=1}^n \frac{\frac{\bar{v}_i}{v_{i \max}} (e^{s^* v_{i \max}} - 1) + 1}{s^* e^{s^* C} \sum_{i=1}^n \bar{v}_i}.$$

Procedura podejmowania decyzji o akceptacji lub odrzuceniu nowego wywołania odbywa się na zasadach stosowanych w innych procedurach sterowania dostępem wywołań. Decyzja o przyjęciu lub odrzuceniu nowego wywołania jest podejmowana na podstawie szacowanej wartości górnego ograniczenia prawdopodobieństwa utraty jednostek danych dla nowego strumienia sumarycznego, powstałego w wyniku złożenia już obsługiwanego sumarycznego strumienia jednostek danych oraz strumienia związanego z nowym wywołaniem. Żądanie obsługi nowego wywołania (połączenia) oznacza obliczenie nowej wartości górnego ograniczenia prawdopodobieństwa utraty zgłoszeń. Jeżeli obliczona wartość \bar{p}_L jest mniejsza od wartości dopuszczalnej p_{dop} , która jest ustalona na podstawie żądanej jakości obsługi strumienia zagregowanego, to wywołanie jest przyjmowane do obsługi; w innym przypadku wywołanie jest odrzucane.

Wartość górnego ograniczenia prawdopodobieństwa \bar{p}_L jest wartością górnego ograniczenia prawdopodobieństwa odrzucania jednostek danych w strumieniu zagregowanym. Jednym ze sposobów szacowania wartości górnego ograniczenia prawdopodobieństwa odrzucania jednostek danych, należących do jednego ze strumieni składowych strumienia zagregowanego, jest przyjęcie założenia o proporcjonalności liczby odrzucanych i obsługiwanych jednostek danych. Zgodnie z tym założeniem iloraz wartości górnego ograniczenia szybkości odrzucania jednostek danych generowanych przez i -te ($i = 1, 2, \dots, n$) źródło, tj. $\sup v_{iL}$ oraz wartości górnego ograniczenia szybkości odrzucania w strumieniu zagregowanym, tj. $\sup v_L$, jest równy ilorazowi wartości szczytowych szybkości generowania przez i -te źródło oraz wartości sumy szczytowych wartości szybkości generowania w strumieniu zagregowanym. Rezultatem podstawienia zależności pomiędzy górnymi ograniczeniami szybkości odrzucania:

$$\sup v_{iL} = \frac{v_{i \max}}{\sum_{j=1}^n v_{j \max}} \sup v_L$$

do wzoru określającego górne ograniczenie prawdopodobieństwa utraty jest następujące wyrażenie:

$$\bar{p}_{iL} = \frac{E(e^{s^*v})}{s^* e^{s^*C}} \frac{v_{i\max}}{\sum_{j=1}^n v_{j\max}} \frac{1}{\bar{v}_i} = \bar{p}_L \frac{v_{i\max}}{\sum_{j=1}^n v_{j\max}} \left(\frac{\bar{v}_i}{\sum_{j=1}^n \bar{v}_j} \right)^{-1}.$$

Obliczanie wartości \bar{p}_{iL} ($i = 1, 2, \dots, n$) umożliwia podejmowanie decyzji przyjmowania lub odrzucania wywołań pochodzących od poszczególnych źródeł. W przypadku nowego żądania obsługi obliczana jest nowa wartość górnego ograniczenia prawdopodobieństwa utraty jednostek danych \bar{p}_{iL} dla i -tego połączenia. Jeżeli wartość \bar{p}_{iL} jest mniejsza od zadanej wartości $p_{i\text{dop}}$, to wywołanie jest przyjmowane do obsługi, w innym przypadku wywołanie jest odrzucane. Wartości $p_{i\text{dop}}$ są ustalane na podstawie żądanej jakości obsługi dla poszczególnych źródeł. Na ogół wartości $p_{i\text{dop}}$ są różne dla różnych wywołań, tj. dla różnych źródeł i ($i = 1, 2, \dots, n$).

Jakość obsługi jednostek danych, zgodnie z procedurami sterowania dostępem, wykorzystującymi albo wartości p_{dop} , albo zbiór wartości $\{p_{i\text{dop}}\}$, jest różna. Druga z wymienionych procedur gwarantuje sprawiedliwy podział pojemności i zapobiega zawłaszczeniu pojemności kanału transmisyjnego przez strumienie jednostek danych, które są generowane w jednym lub kilku źródłach.

Algorytm FA ma dwa istotne ograniczenia:

1. Do obliczania wartości górnego ograniczenia prawdopodobieństwa odrzucania jednostek danych nie jest wykorzystywana informacja o całkowitej pojemności pamięci buforowej (kolejki). Ogranicza to możliwości efektywnego wykorzystania pojemności łącza, wynikających ze statystycznych właściwości agregowanych strumieni jednostek danych.

2. Wartość górnego ograniczenia prawdopodobieństwa odrzucania jednostek danych jest obliczana dla strumienia zagregowanego. Nie ma możliwości rozróżnienia indywidualnych wymagań poszczególnych połączeń, dotyczących utraty jednostek danych (możliwość taka jest dostępna w niektórych modyfikacjach omawianego algorytmu).

7.2.2.4.5. ALGORYTM NA

Sterowanie przyjmowaniem wywołań, zgodnie z algorytmem tzw. nieparametrycznego podejścia NA (ang. *Nonparametric Approach*), polega na tym, że decyzje o akceptacji lub odrzuceniu żądań dostępu są podejmowane na podstawie szacowanych górnych ograniczeń wartości prawdopodobieństwa zablokowania systemu obsługi, tj. na podstawie wartości oszacowania górnego ograniczenia wartości prawdopodobieństwa odrzucania jednostek danych. Nazwa algorytmu wywodzi się stąd, że do szacowania wartości prawdopodobieństwa odrzucania jednostek danych wykorzystywane są tylko wartości podstawowych parametrów ruchu; nie jest wymagana znajomość modelu procesu napływu jednostek danych. W omawianym algorytmie przyjmowane jest także założenie, że wartość pojemności pamięci buforowej jest tak dobrana, że jedynym powodem odrzucania jednostek danych jest ograniczona pojemność kanału transmisyjnego.

Zadaniem procedury przyjmowania wywołań NA jest rozstrzygnięcie o przyjmowaniu do obsługi lub odrzuceniu nowych żądań dostępu (żądań obsługi nowych połączeń) na podstawie antycypowanych charakterystyk ruchu tych wywołań oraz stanu obciążenia systemu obsługi tak, aby zachowany był żądany poziom jakości obsługi strumieni jednostek danych. O jakości algorytmu przyjmowania nowych wywołań decyduje efektywna przepustowość systemu obsługi oraz jakość i ilość danych, na podstawie których podejmowane są decyzje o akceptacji lub odrzuceniu. Jakość i ilość danych decyduje także o możliwościach przewidywania i szacowania jakości działania systemu obsługi po przyjęciu nowego wywołania.

W algorytmie NA stosuje się uproszczony opis źródeł strumieni jednostek danych. Decyzja o przyjęciu lub odrzuceniu kolejno generowanego i -tego wywołania zależy od wartości dwóch parametrów charakteryzujących wywołanie:

- szczytowej (ang. *peak bit rate*) szybkości strumienia jednostek danych $v_{i \max}$,
- średniej (ang. *average bit rate*) szybkości strumienia jednostek danych \bar{v}_i .

Kanał transmisyjny o pojemności C jest stanowiskiem obsługi strumieni jednostek danych o stałej długości. Takie zdefiniowanie pojemności kanału transmisyjnego o wartości C jest równoznaczne z definicją szczeliny czasowej w kanale transmisyjnym o długości $1/C$, tj. przedziału czasu o długości niezbędnej do obsługi jednej jednostki danych. Definiowanie szczelin czasowych jest równoważne możliwości definiowania w kanale transmisyjnym łącz danych o stałych pojemnościach (ang. *constant link rate*), będących wielokrotnościami szczelin czasowych. Dla tak zdefiniowanego kanału transmisyjnego, obsługującego n wywołań, możliwe jest zdefiniowanie łącznego prawdopodobieństwa utraty jednostek danych (ang. *customer loss probability*) $p_{\text{CLP}}(n)$.

Na potrzeby szacowania górnego ograniczenia prawdopodobieństwa odrzucenia jednostek danych $p_{\text{CLP}}(n)$ mogą być wykorzystywane wartości maksymalne liczby jednostek danych, wartości średnie liczby jednostek danych i wariancje liczby jednostek danych w poszczególnych, obsługiwanych połączeniach.

Jedno z możliwych oszacowań górnego ograniczenia prawdopodobieństwa odrzucenia jednostek danych $p_{\text{CLP}}(n)$ to oszacowanie z wykorzystaniem wartości maksymalnych i średnich liczb jednostek danych napływających w poszczególnych n połączeniach, oznaczane dalej jako $\bar{p}_{\text{CLP}}(n, s)$.

Dla każdego obsługiwanego i -tego ($i = 1, 2, \dots, n$) połączenia (wywołania) monitorowane są liczby jednostek danych napływających ze szczytową $v_{i \max}$ i średnią \bar{v}_i szybkością w przedziałach czasu o długości s szczelin czasowych (przedziałach obserwacji). Wynikiem monitorowania są maksymalne i średnie liczby jednostek danych napływających w i -tym połączeniu, oznaczone odpowiednio jako $\bar{l}_{i \max}(s) = s v_{i \max} / C$ oraz $\bar{l}_i(s) = s \bar{v}_i / C$.

Prawdopodobieństwo odrzucenia jednostek danych $p_{\text{CLP}}(n)$ jest ograniczone z góry przez wartość wyrażenia $B(p_1, p_2, \dots, p_n | s)$ definiowanego jako:

$$p_{\text{CLP}}(n) \leq B(p_1, p_2, \dots, p_n | s) = \frac{\sum_{k=0}^{\infty} \max(0, k-s) p_1 * p_2 * \dots * p_n(k)}{\sum_{k=0}^{\infty} k p_1 * p_2 * \dots * p_n(k)},$$

gdzie:

- $p_i(j)$ – prawdopodobieństwo, że w obsługiwanym i -tym ($i = 1, 2, \dots, n$) wywołaniu, w czasie trwania s szczelin czasowych, napływie j ($j = 1, 2, \dots, n$) jednostek danych,
- $*$ – operacja splotu,
- $p_1 * p_2 * \dots * p_n(k)$ – rozkład prawdopodobieństwa sumy n niezależnych zmiennych losowych (liczby jednostek danych napływających w n połączeniach).

Jeżeli $\theta_i = \{\theta_i(j)\}$ jest zbiorem następujących rozkładów j jednostek danych w i -tym połączeniu:

$$\theta_i(j) = \begin{cases} \bar{l}_i(s)/l_{i\max}(s) & \text{dla } j = \bar{l}_i(s) \\ 1 - \bar{l}_i(s)/l_{i\max}(s) & \text{dla } j = 0 \\ 0 & \text{w innym przypadku} \end{cases},$$

charakteryzowanym wartościami $l_{i\max}(s)$ i $\bar{l}_i(s)$, tzn. liczbami jednostek danych, napływających w przedziałach o długości s szczelin czasowych, to θ_i jest rozkładem napływu maksymalnej liczby jednostek danych lub braku ich napływu w połączeniu o średniej liczbie napływających jednostek danych $\bar{l}_i(s)$. Wartość wyrażenia $B(p_1, p_2, \dots, p_n | s)$ jest ograniczona z góry zgodnie z następującą zależnością:

$$\begin{aligned} B(p_1, p_2, \dots, p_n | s) &\leq B(p_1, p_2, \dots, p_{n-1}, \theta_n | s) \leq B(p_1, p_2, \dots, \theta_{n-1}, \theta_n | s) \leq \dots \\ &\leq B(\theta_1, \theta_2, \dots, \theta_{n-1}, \theta_n | s) = \\ &= \frac{\sum_{k=0}^{\infty} \max(0, k-s) \theta_1 * \theta_2 * \dots * \theta_n(k)}{\sum_{k=0}^{\infty} k \theta_1 * \theta_2 * \dots * \theta_n(k)} = \\ &= \frac{\sum_{k=s+1}^{\infty} (k-s) \theta_1 * \theta_2 * \dots * \theta_n(k)}{\sum_{k=0}^{\infty} k \theta_1 * \theta_2 * \dots * \theta_n(k)} = \bar{p}_{\text{CLP}}(n, s). \end{aligned}$$

W procedurze akceptacji i odrzucania wywołań wykorzystywane jest górne oszacowanie $\bar{p}_{\text{CLP}}(n, s)$ prawdopodobieństwa blokady kanału transmisyjnego $p_{\text{CLP}}(n)$ spełniające warunek:

$$p_{\text{CLP}}(n) \leq \bar{p}_{\text{CLP}}(n, s).$$

Obsługa nowo generowanego $(n + 1)$ -go wywołania wymaga wyznaczenia wartości oszacowania prawdopodobieństwa blokady kanału transmisyjnego $\bar{p}_{\text{CLP}}(n + 1, s)$.

Jeżeli wartość oszacowania $\bar{p}_{\text{CLP}}(n + 1, s)$ nie jest większa od zadanej wartości prawdopodobieństwa utraty jednostek danych (parametru jakości obsługi strumieni jednostek danych) p^*_{CLP} :

$$\bar{p}_{\text{CLP}}(n + 1, s) \leq p^*_{\text{CLP}},$$

to $(n + 1)$ wywołanie jest przyjmowane do obsługi; w innym przypadku, tzn. gdy:

$$\bar{p}_{\text{CLP}}(n + 1, s) > p^*_{\text{CLP}},$$

żądanie ustanowienia (obsługi) $(n + 1)$ -go połączenia jest odrzucone.

Implementacja algorytmu NA oznacza, że decyzja o akceptacji lub odrzuceniu $(n + 1)$ -go wywołania musi być przygotowana w czasie rzeczywistym. Z tego punktu widzenia istotne znaczenie ma czas obliczania wartości górnego ograniczenia prawdopodobieństwa $\bar{p}_{\text{CLP}}(n + 1, s)$. Wartość oszacowania prawdopodobieństwa $\bar{p}_{\text{CLP}}(n + 1, s)$ może być obliczana:

➤ bezpośrednio, tzn. do wyrażenia $\bar{p}_{\text{CLP}}(n + 1, s) = B(\theta_1, \theta_2, \dots, \theta_{n-1}, \theta_n, \theta_{n+1} | s)$ podstawiane są wartości parametrów ruchu już obsługiwanych n wywołań oraz żądanego dostępu $(n + 1)$ -go wywołania,

➤ pośrednio (rekurencyjnie), tzn. do obliczania wartości prawdopodobieństwa $\bar{p}_{\text{CLP}}(n + 1, s)$ w systemie obsługującym $(n + 1)$ połączeń wykorzystywane są informacje o stanie systemu obsługującego n połączeń, tj. o stanie systemu obsługi w chwili poprzedzającej nadejście $(n + 1)$ -go wywołania.

Bezpośrednie obliczanie oszacowania wartości prawdopodobieństwa $\bar{p}_{\text{CLP}}(n + 1, s)$ na ogół jest zadaniem złożonym obliczeniowo. Wartość wyrażenia $B(\theta_1, \theta_2, \dots, \theta_{n-1}, \theta_n, \theta_{n+1} | s)$ zależy tylko od wartości rozkładu prawdopodobieństwa splotu $\theta_1 * \theta_2 * \dots * \theta_n * \theta_{n+1}$. Jeżeli strumienie jednostek danych wszystkich obsługiwanych połączeń mają te same wartości parametrów ruchu, tzn. $l_{i\text{max}}(s) = l_{\text{max}}(s)$ i $\bar{l}_i(s) = \bar{l}(s)$ dla $i = 1, 2, \dots, n + 1$, to rozkład prawdopodobieństwa splotu $\theta_1 * \theta_2 * \dots * \theta_n * \theta_{n+1}$ redukuje się do rozkładu dwumianowego z parametrami $n + 1$ i $\bar{l}(s)/l_{\text{max}}(s)$. W przypadku zwiększenia liczby obsługiwanych połączeń, tzn. wzrostu liczby obsługiwanych źródeł strumieni zgłoszeń, wartość splotu $\theta_1 * \theta_2 * \dots * \theta_n * \theta_{n+1}$ może być aproksymowana rozkładem normalnym. Aproksymacja taka znacznie redukuje czas obliczania oszacowania wartości prawdopodobieństwa $\bar{p}_{\text{CLP}}(n + 1, s)$.

W procedurze rekurencyjnej szacowania wartości prawdopodobieństwa odrzucania zgłoszeń $\bar{p}_{\text{CLP}}(n + 1, s)$ zgodnie z algorytmem NA wykorzystywane są właściwości wektora stanu obciążenia systemu obsługi. Właściwości te decydują o tym, że obliczenie wartości wyrazów wektora stanu obciążenia $\mathbf{w}_{n+1} = \{w_{n+1}(m)\}$ systemu

obsługującego $(n + 1)$ wywołań jest możliwe na podstawie znajomości wartości wyrazów wektora stanu obciążenia $\mathbf{w}_n = \{w_n(m)\}$ systemu obsługującego n wywołań oraz wartości parametrów ruchu w $(n + 1)$ -szym wywołaniu. Podobnie możliwe jest obliczenie wyrazów wektora $\mathbf{w}_{n-1} = \{w_{n-1}(m)\}$ na podstawie znajomości wyrazów wektora $\mathbf{w}_n = \{w_n(m)\}$ oraz parametrów ruchu w n -tym wywołaniu wtedy, gdy zakończona zostanie obsługa n -tego wywołania.

W przypadku stosowania algorytmu NA w wersji rekurencyjnej wartość górnego oszacowania prawdopodobieństwa odrzucania zgłoszeń $\bar{p}_{\text{CLP}}(n, s)$ określa wyrażenie:

$$\bar{p}_{\text{CLP}}(n, s) = \frac{w_n(s+1)}{\bar{l}(n, s)},$$

w którym:

$w_n(s+1)$ – wartość $(s+1)$ -go wyrazu aktualnego wektora stanu obciążenia systemu obsługi \mathbf{w}_n ,

$\bar{l}(n, s)$ – suma średnich liczb jednostek danych napływających we wszystkich obsługiwanych n wywołaniach, tj. $\bar{l}(n, s) = \bar{l}_1(s) + \bar{l}_2(s) + \dots + \bar{l}_n(s)$.

Wyrazy wektora aktualnego obciążenia systemu obsługi:

$$\mathbf{w}_n = \{w_n(m); m = 0, 1, 2, \dots\},$$

dla jednocześnie obsługiwanych n wywołań, definiowane są w następujący sposób:

$$w_n(m) = \sum_{k=m}^{\infty} Q(k) = \sum_{k=m}^{\infty} \sum_{j=k}^{\infty} \theta_1 * \theta_2 * \dots * \theta_n(j),$$

gdzie $Q(k)$ jest uzupełniającym rozkładem prawdopodobieństwa łącznego liczby jednostek danych napływających do systemu obsługi w przedziale o długości s szczylin czasowych.

Wartość m -tego wyrazu wektora obciążenia systemu obsługi \mathbf{w}_n jest wartością sumy wszystkich łącznych rozkładów prawdopodobieństwa zdarzeń, że w przedziale o długości s szczylin czasowych w systemie obsługi, obsługującym n połączeń, mogą być obsługiwane wszystkie kombinacje liczby zgłoszeń, których suma jest nie mniejsza niż m .

Iloraz wartości $(s+1)$ -go wyrazu wektora obciążenia systemu obsługi \mathbf{w}_n ($w_n(s+1)$) oraz aktualnej sumy średnich liczb zgłoszeń $\bar{l}(n, s)$ jest wartością górnego oszacowania prawdopodobieństwa odrzucania zgłoszeń $\bar{p}_{\text{CLP}}(n, s)$:

$$\bar{p}_{\text{CLP}}(n, s) = \frac{w_n(s+1)}{\bar{l}(n, s)} = \frac{\sum_{k=s+1}^{\infty} \sum_{j=k}^{\infty} \theta_1 * \theta_2 * \dots * \theta_n(j)}{\bar{l}(n, s)}.$$

Wartość tego ilorazu można interpretować jako wartość łącznego prawdopodobieństwa zajętości wszystkich szczylin przedziału czasu o długości s szczylin, przypadającego na pojedynczą jednostkę danych w łącznym strumieniu jednostek danych, napływających ze średnią szybkością równą $\bar{v}(n)$ ($\bar{v}(n) = \bar{v}_1 + \bar{v}_2 + \dots + \bar{v}_n$).

W przypadku wygenerowania żądania kolejnego $(n + 1)$ -go wywołania, charakteryzowanego wartościami parametrów $l_{(n+1)\max}(s)$ i $\bar{l}_{n+1}(s)$, oblicza się:

- wartość $(s + 1)$ -go wyrazu nowego wektora obciążenia systemu obsługi w_{n+1} ,
- nową wartość górnego oszacowania prawdopodobieństwa odrzucenia zgłoszeń $\bar{p}_{CLP}(n + 1, s)$ według zależności:

$$\bar{p}_{CLP}(n + 1, s) = \frac{w_{n+1}(s + 1)}{\bar{l}(n + 1, s)},$$

w której $w_{n+1}(s + 1)$ i $\bar{l}(n + 1, s)$ dane są wyrażeniami:

$$\begin{aligned} w_{n+1}(s + 1) &= F_{s+1} \left[\{w_n(m)\}, l_{(n+1)\max}(s), \frac{\bar{l}_{n+1}(s)}{l_{(n+1)\max}(s)} \right] = \\ &= (1 - \alpha_{n+1})w_n(s + 1) + \begin{cases} \alpha_{n+1}w_n(s + 1 - l_{(n+1)\max}(s)) & \text{dla } s + 1 \geq l_{(n+1)\max}(s) \\ \alpha_{n+1}[(l_{(n+1)\max}(s) - (s + 1)) + w_n(0)] & \text{dla } s + 1 < l_{(n+1)\max}(s) \end{cases}, \end{aligned}$$

oraz

$$\bar{l}(n + 1, s) = \bar{l}(n, s) + \bar{l}_{n+1}(s).$$

Obliczenie, zgodnie z podanymi wyrażeniami, wartości $\bar{p}_{CLP}(n + 1, s)$ pozwala na podjęcie decyzji o akceptacji lub odrzuceniu żądania nowego $(n + 1)$ -go wywołania. Żądanie wywołania jest akceptowane po spełnieniu warunku:

$$\bar{p}_{CLP}(n + 1, s) \leq p_{CLP}^*.$$

Jeżeli $(n + 1)$ -sze wywołanie zostanie zaakceptowane, to wartości pozostałych wyrazów $(m = 0, 1, 2, \dots, s, s + 2, \dots)$ wektora stanu obciążenia w_{n+1} systemu obsługującego $(n + 1)$ wywołań, są obliczane zgodnie z następującym wyrażeniem:

$$w_{n+1}(m) = F_m(\{w_n(m)\}, j, \alpha_{n+1}) = (1 - \alpha_{n+1})w_n(m) + \begin{cases} \alpha_{n+1}w_n(m - j) & \text{dla } m \geq j \\ \alpha_{n+1}[(j - m) + w_n(0)] & \text{dla } m < j \end{cases}$$

gdzie:

$$\alpha_{n+1} = \bar{l}_{n+1}(s)/l_{(n+1)\max}(s), \quad j = l_{(n+1)\max}(s) \quad \text{oraz } m = 0, 1, 2, \dots$$

Obliczone w ten sposób wartości wyrazów wektora stanu obciążenia systemu w_{n+1} będą wykorzystane:

- albo do obliczenia wartości $w_{n+2}(s + 1)$, niezbędnej do podjęcia decyzji o akceptacji lub odrzuceniu kolejnego $(n + 2)$ -go żądania połączenia (wywołania) wygenerowanego w czasie obsługi $(n + 1)$ połączeń,
- albo do obliczenia wartości wyrazów wektora w_n , jeżeli przed wygenerowaniem kolejnego żądania połączenia (wywołania) zostanie zakończona obsługa jednego z $(n + 1)$ obsługiwanych wywołań.

Jeżeli obsługa jednego z $(n + 1)$ wcześniej zaakceptowanych wywołań zostanie zakończona przed nadejściem kolejnego żądania połączenia, to wartości wyrazów nowego wektora stanu \tilde{w}_n są obliczane zgodnie z następującymi wyrażeniami:

$$\tilde{w}_n(m) = F_m^{-1} \left[\{w_{n+1}(m)\}, l_{j \max}(s), \frac{\bar{l}_j(s)}{l_{j \max}(s)} \right] \text{ oraz } \bar{l}(n, s) = \bar{l}(n + 1, s) - \bar{l}_j(s),$$

gdzie j jest indeksem wywołania, którego obsługa została zakończona ($j \in \{1, 2, \dots, n, n + 1\}$).

Rekurencyjna procedura obliczania wartości wyrazów wektora stanu obciążenia systemu i ich przechowywanie jest wystarczającym warunkiem możliwości szacowania wartości prawdopodobieństwa odrzucania jednostek danych na podstawie wartości parametrów połączenia, dla którego żądanie połączenia jest generowane. Procedura ta jest prosta w implementacji i znacznie redukuje czas potrzebny do wyznaczenia decyzji, co wynika m.in. z ograniczonej liczby parametrów źródeł strumieni jednostek danych wykorzystywanych do podejmowania decyzji o akceptacji lub odrzuceniu wywołań.

7.2.2.4.6. ALGORYTM HTA

W algorytmie sterowania przyjmowaniem wywołań HTA (ang. *Heavy Traffic Approximation*) wykorzystywane są asymptotyczne właściwości ogona rozkładu długości kolejki o nieograniczonej pojemności. Jednostki danych, znajdujące się w pamięci buforowej, są obsługiwane (wyprowadzane) przez stanowisko obsługi ze stałą szybkością, a napływ jednostek danych do kolejki jest procesem Markowa, definiowanym macierzą prawdopodobieństw $P(z)$.

Dla odpowiednio dużych wartości k wartość prawdopodobieństwa, że liczba jednostek danych w systemie obsługi, tj. suma jednostek danych oczekujących na obsługę i obsługiwanej N jest większa od k , jest dana wyrażeniem $p(N > k) = \alpha(1/z^*)^k$, gdzie α jest nieznaną stałą, której obliczenie wartości jest na ogół zadaniem trudnym. Właściwości ogona rozkładu długości kolejki w stanie ustalonym, w omawianym systemie obsługi jednostek danych, są charakteryzowane najmniejszą wartością pierwiastka z^* wyznacznika macierzy prawdopodobieństw.

Znajomość wartości prawdopodobieństwa $p(N > k)$ jest równoważna znajomości wartości:

- prawdopodobieństwa odrzucania jednostek danych powodowanego przekroczeniem ustalonej wartości czasu oczekiwania na obsługę, tj. odrzucania z powodu nadmiernego opóźnienia wnoszonego przez system obsługi,
- prawdopodobieństwa odrzucania jednostek danych powodowanego przekroczeniem pojemności pamięci buforowej, tj. odrzucania z powodu nadmiernego napływu jednostek danych do pamięci buforowej o skończonej pojemności.

Wartość prawdopodobieństwa $p = p(N > k)$ spełnia warunek, który może być stosowany do podejmowania decyzji w procedurze sterowania przyjmowaniem wywołań do systemu obsługi. Proces napływu jednostek danych do systemu obsługi, charakte-

ryzowany macierzą prawdopodobieństw, jest na tyle ogólny, że może być wykorzystany do modelowania superpozycji strumieni zgłoszeń generowanych przez n źródeł. W algorytmie HTA przyjęto, że na potrzeby modelowania superpozycji strumieni jednostek danych, pochodzących z różnych źródeł strumieni jednostek danych, wystarczająca jest znajomość wartości szczytowych i średnich szybkości napływu jednostek danych oraz czasu trwania przedziału aktywności źródeł. Jeżeli wartość prawdopodobieństwa p_{n+1} , obliczona dla systemu obsługującego n wywołań i żądania dostępu dla $(n + 1)$ -go wywołania, jest mniejsza od zadanej wartości prawdopodobieństwa p , to żądanie dostępu jest akceptowane; w innym przypadku żądanie jest odrzucane.

Zadanie wyznaczenia wartości p_{n+1} sprowadza się do wyznaczenia wartości stałej α_{n+1} oraz wartości pierwiastka z_{n+1}^* .

Po przyjęciu, że dwustanowe źródło (ang. *on/off source*) strumienia jednostek danych, generowanych w i -tym połączeniu ($i = 1, 2, \dots, n$) charakteryzują:

- wartość szczytowa szybkości generowania zgłoszeń przez i -te źródło: $v_{i \max}$,
- wartość średnia szybkości generowania zgłoszeń przez i -te źródło: \bar{v}_i ,
- wartość średniego czasu trwania przedziału aktywności i -tego źródła: a_i ,
- wartości współczynnika wykorzystania i -tego źródła: ρ_i ($\rho_i = \bar{v}_i/v_{i \max}$),

wartość stałej α_n jest równa wartości współczynnika wykorzystania n źródeł strumieni zgłoszeń:

$$\alpha_n = \frac{\bar{v}_1}{v_{1 \max}} + \frac{\bar{v}_2}{v_{2 \max}} + \dots + \frac{\bar{v}_n}{v_{n \max}}.$$

Jeżeli długości przedziałów aktywności i ciszy źródeł mają rozkład wykładniczy, to wartość z_{Mn}^* (M oznacza rozkład wykładniczy) jest aproksymowana wartością wyrażenia:

$$z_{Mn}^* = 1 + \frac{1 - \alpha_n}{\sum_{i=1}^n \bar{v}_i (1 - \rho_i)^2 a_i}.$$

Gdy długości przedziałów aktywności i ciszy mają rozkład dowolny, wartość z_{Gn}^* (G oznacza rozkład dowolny) jest przybliżana wartością wyrażenia:

$$z_{Gn}^* = 1 + \frac{2(1 + \alpha_n)}{\sum_{i=1}^n \bar{v}_i (1 - \rho_i)^2 a_i [\sigma_{ai}^2 + \sigma_{ci}^2]},$$

w którym: σ_{ai}^2 oraz σ_{ci}^2 oznaczają wartości wariancji długości przedziałów aktywności i ciszy dla i -tego wywołania.

Znajomość wartości stałej α_n i oszacowania wartości pierwiastka z_{Mn}^* (lub z_{Gn}^*) jest równoznaczna ze znajomością wartości prawdopodobieństwa p_{Mn} (lub p_{Gn}):

$$p_{Mn} = \left(\sum_{i=1}^n \frac{\bar{v}_i}{v_{i \max}} \right) \left[1 + \frac{1 + \alpha_n}{\sum_{i=1}^n \bar{v}_i (1 - \rho_i)^2 a_i} \right]^{-k}.$$

W przypadku generacji nowego żądania dostępu dla $(n + 1)$ -go wywołania obliczana jest wartość prawdopodobieństwa $p_{M(n+1)}$ (lub $p_{G(n+1)}$); dla $p_{M(n+1)}$ zgodnie z wyrażeniem:

$$p_{M(n+1)} = \left(\sum_{i=1}^{n+1} \frac{\bar{v}_i}{v_{i \max}} \right) \left[1 + \frac{1 + \alpha_{n+1}}{\sum_{i=1}^{n+1} \bar{v}_i (1 - \rho_i)^2 a_i} \right]^{-k} = \left[\alpha_n + \frac{\bar{v}_{n+1}}{v_{(n+1) \max}} \right] \times$$

$$\times \left[1 + \frac{1 + \alpha_n + \frac{\bar{v}_{n+1}}{v_{(n+1) \max}}}{\sum_{i=1}^n \bar{v}_i (1 - \rho_i)^2 a_i + \bar{v}_{n+1} (1 - \rho_{n+1})^2 a_{n+1}} \right]^{-k}.$$

Jeżeli wartość obliczonego prawdopodobieństwa $p_{M(n+1)}$ jest mniejsza lub równa zadanej wartości prawdopodobieństwa p zdarzenia, że długość kolejki jest równa zadanej wartości k :

$$p_{M(n+1)} \leq p,$$

to $(n + 1)$ -sze wywołanie jest przyjmowane do obsługi; w innym przypadku:

$$p_{M(n+1)} > p,$$

$(n + 1)$ -sze wywołanie jest odrzucane.

Efektywność algorytmu HTA w rzeczywistych systemach obsługi ruchu, tzn. w systemach obsługi ze skończoną pojemnością pamięci buforowej, zwiększa się wraz ze wzrostem wartości ilorazu pojemności pamięci buforowej i długości przedziałów aktywności źródeł. Niewielkie rozmiary pamięci buforowej powodują, że działanie omawianego algorytmu jest zbliżone do działania algorytmu alokacji deterministycznej, tzn. przydzielania zasobów (pojemności) w ilościach odpowiadających wartości szczytowej generowanego ruchu.

7.2.3. KSZTAŁTOWANIE RUCHU

Zadania zwielokrotniania statystycznego są charakteryzowane ogólnymi właściwościami:

➤ źródeł strumieni jednostek danych, tzn. zmienną szybkością napływu zgłoszeń modelowanych i opisywanych najczęściej przez szczytowe i średnie wartości szybkości napływu zgłoszeń oraz rozkład czasu trwania przedziałów aktywności i ciszy,

➤ stosowanych algorytmów zwielokrotniania statystycznego, tj. sposobów wykorzystywania przedziałów ciszy (braku aktywności) źródła do obsługi jednostek danych, generowanych w poprzedzających je przedziałach aktywności,

➤ otrzymywanych rozwiązań, tzn. ilością zasobów (zwłaszcza pojemności łącza) przydzielonych nowemu wywołaniu (połączeniu), których górne i dolne ograniczenia zależą – odpowiednio – od wartości szczytowej i średniej szybkości strumieni jednostek danych.

Zależność rozwiązań alokacji zasobów od maksymalnych i średnich szybkości strumieni jednostek danych jest cechą charakterystyczną rozwiązań zadań komunikacyjnych, w których korzysta się z różnych odmian techniki komutacji jednostek danych, tj. komutacji pakietów, komutacji ramek, komutacji komórek itd.

Wszystkie efektywne algorytmy rozwiązywania zadań statystycznego zwielokrotniania strumieni ruchu prowadzą do kształtowania (wygładzania) ruchu (ang. *traffic shaping*) polegającego na tym, że:

➤ gdy szybkość napływu jednostek danych jest większa od szybkości obsługi, nadmiar napływających jednostek danych jest przechowywany w kolejkach pamięci buforowych – długości kolejek wejściowych rosną,

➤ gdy szybkość napływu jednostek danych jest mniejsza od szybkości obsługi, dodatkowo obsługiwane są wcześniej zapamiętane i przechowywane zgłoszenia – długości kolejek wejściowych maleją i tym samym uwalniane są wcześniej zajęte zasoby pamięci buforowej.

Stosowanie algorytmu kształtowania ruchu, realizowanego w źródle lub węźle dostępowym do sieci, przy ograniczonych pojemnościach łącza i pamięci buforowej, mniejszych od wartości szczytowych składowych obsługiwanego połączenia zagregowanego, jest powodem utraty i opóźnienia zgłoszeń.

Jakość algorytmów zwielokrotniania statystycznego jest mierzona stopniem wykorzystania pojemności łącza oraz pojemności pamięci buforowej przy spełnieniu ograniczeń, jakimi najczęściej są zachowanie żądanej (zadanej) wartości prawdopodobieństwa utraty (odrzućcia) zgłoszeń i (lub) wartość średniego opóźnienia zgłoszeń, wnoszonego przez system obsługi.

7.2.4. UTRZYMYWANIE RUCHU

7.2.4.1. REGLAMENTACJA ZASOBÓW I ZAPOTRZEBOWANIA NA ZASOBY

Procedury sterowania przyjmowaniem wywołań (sterowania dostępem) do systemu obsługi można utożsamiać z przyjęciem do obsługi wywołań na podstawie wcześniej wynegocjowanych warunków obsługi strumieni jednostek danych. W procedurach sterowania dostępem implementowane są algorytmy podejmowania decyzji, wykorzystujące znane *a priori* wartości parametrów strumieni jednostek danych. Różnice pomiędzy wartościami parametrów stosowanych w algorytmie akceptacji wywołań oraz generowanych rzeczywistych strumieni jednostek danych mogą powodować

zarówno przeciążenia systemu obsługi, jak i pogorszenie jakości usług dostarczanych wywołaniom zaakceptowanym do obsługi.

Do podstawowych przyczyn, z powodu których procedury sterowania przyjmowaniem wywołań (sterowania dostępem) nie gwarantują skutecznego zapobiegania powstawania przeciążeń, zalicza się:

1. Nieznajomość lub niedokładne oszacowanie lub zmienność charakterystyk generowanego ruchu i wymagań z nim związanych. Wartości parametrów wywołań, na podstawie których wywołanie zostało przyjęte do obsługi mogą być różne od wartości parametrów generowanego rzeczywistego strumienia zgłoszeń.

2. Zawodność sprzętu i oprogramowania. Wynegocjowane wartości parametrów jakości obsługi strumieni zgłoszeń w poszczególnych wywołaniach mogą być różne od żądanych i (lub) oczekiwanych.

3. Zaniżanie wartości zapotrzebowania na zasoby komunikacyjne. Stosowane taryfy mogą uzasadniać różne od pożądanych, przez operatora systemu obsługi, świadome lub przypadkowe działania użytkowników, których celem jest optymalizacja kosztów obsługi.

4. Działania zagrażające bezpieczeństwu sieci. Brak polityki bezpieczeństwa systemu obsługi i wspierających politykę bezpieczeństwa mechanizmów może być przyczyną świadomych lub przypadkowych działań, których skutkiem jest przeciążenie lub upadek systemu obsługi i tym samym częściowa lub całkowita degradacja jakości usług systemu.

Z podanych powyżej powodów wynika, że procedury sterowania dostępem, zaliczane do grupy metod prewencyjnych (zapobiegawczych), są jedynie warunkiem koniecznym, ale nie wystarczającym, zapobiegania powstawaniu przeciążeń. Wywiązanie się z wynegocjowanych warunków jakości obsługi oraz podniesienie stopnia zabezpieczenia systemu obsługi przed powstawaniem przeciążeń wymaga uzupełnienia mechanizmów sterowania dostępem o mechanizmy nadzorowania zgodności wartości parametrów generowanego ruchu z wartościami parametrów wykorzystywanych na etapie przyjmowania wywołań do obsługi.

Zbiór mechanizmów kontrolowania i nadzoru zgodności wartości parametrów planowanego (akceptowanego) i rzeczywistego ruchu jest zbiorem procedur utrzymywania ruchu (ang. *traffic policing*) lub sterowania parametrami użytkownika (ang. *usage parameter control*).

Zadaniem mechanizmów utrzymywania ruchu jest:

➤ detekcja połączeń naruszających wynegocjowane warunki obsługi (ang. *non-conforming connections*), dla których wartości parametrów wynegocjowanych i rzeczywistych nie są zgodne,

➤ w przypadku detekcji naruszenia warunków obsługi podejmowanie działań, których celem jest minimalizacja negatywnych konsekwencji wykrytych różnic dla jakości usług w połączeniach nienaruszających wynegocjowanych warunków obsługi (ang. *conforming connections*).

Jakość mechanizmów reaktywnych utrzymywania ruchu, których działania nie powinny mieć charakteru działań represyjnych, determinują:

➤ czas reakcji na wykryte naruszenia warunków obsługi (wykrytą rozbieżność wartości parametrów),

➤ stopień przezroczystości mechanizmu dla połączeń nienaruszających warunków obsługi, mierzony wzrostem opóźnienia obsługi jednostek lub zwiększeniem liczby traconych jednostek danych.

Do działań podejmowanych w ramach omawianej funkcji monitorowania w razie wykrycia naruszenia kontraktu zalicza się:

➤ zaniechanie obsługi jednostek danych naruszających kontrakt,

➤ opóźnienie obsługi jednostek danych naruszających kontrakt w taki sposób, że jednostki danych danego połączenia są obsługiwane zgodnie z kontraktem,

➤ oznaczenie jednostek danych naruszających kontrakt tak, aby w przypadku zagrożenia powstaniem przeciążenia mogły być traktowane inaczej niż jednostki danych generowane z zachowaniem warunków wcześniej wynegocjowanego kontraktu,

➤ adaptacyjne sterowanie ruchem przez informowanie źródła o naruszeniu kontraktu.

W mechanizmach reakcyjnych przedmiotem nadzoru i kontroli są wartości zbioru parametrów przyjętych jako charakterystyczne dla źródeł obsługiwanego ruchu jednostek danych i połączeń ustanawianych dla tych źródeł, tzw. zbiór deskryptorów ruchu. Zbiór deskryptorów ruchu jest determinowany przez:

➤ implementowany algorytm sterowania przyjmowaniem wywołań (sterowania dostępem),

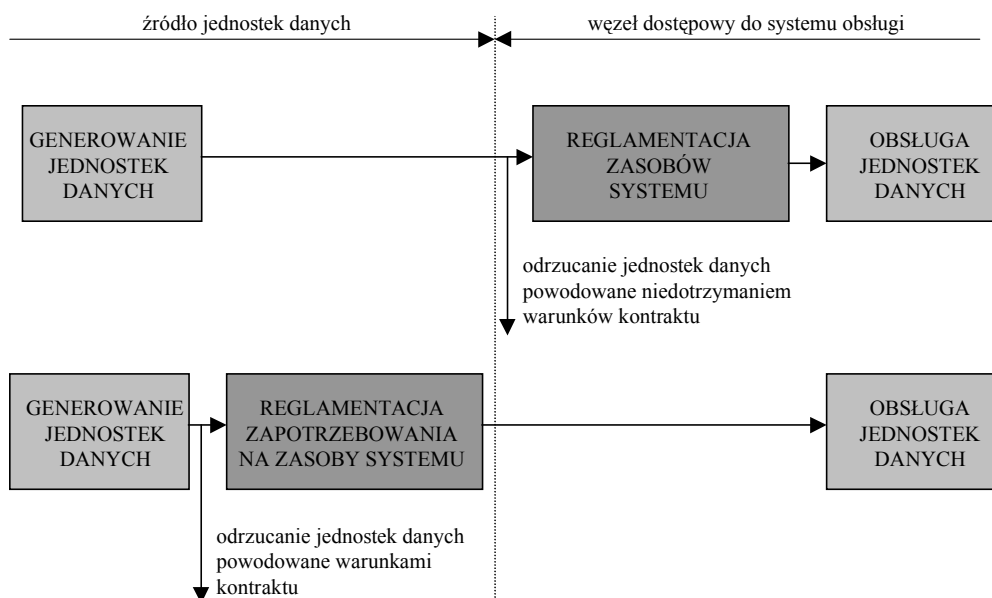
➤ algorytm badania zgodności wartości parametrów rzeczywistych strumieni napływających jednostek danych z wartościami parametrów uzgodnionych w kontrakcie.

Mimo iż opracowano wiele różnych metod badania zgodności wartości rzeczywistych parametrów źródeł strumieni jednostek danych oraz związanych z nimi połączeń z wartościami parametrów uzgodnionych w kontrakcie, to żadna z nich nie jest w pełni zgodna z oczekiwaniami. Przyczyną rozbieżności oczekiwań i realizacji jest złożoność zadań detekcji i reakcji oraz czas rozwiązania zadania. Jakość mechanizmu utrzymywania ruchu, a tym samym jakość rozwiązania omawianych zadań detekcji i przeciwdziałania w razie naruszenia warunków kontraktu, zależy od szybkości podejmowania decyzji o różnicach wartości parametrów mechanizmu sterowania dostępem. Jakość podejmowania decyzji o wartościach deskryptorów ruchu, generowanego przez źródła, zależy od jakości predykcji wartości parametrów opisujących zachowanie źródła. Jakość predykcji rośnie wraz z wydłużeniem czasu obserwacji. Sprzeczność polega na tym, że im czas podjęcia decyzji o wartościach parametrów rzeczywistego ruchu zgłoszeń jest krótszy, tym gorsza jest jakość predykcji zachowań źródeł strumienia jednostek danych.

W różnych metodach utrzymywania ruchu, tzn. do kontroli (badania) zgodności rzeczywistych i uzgodnionych (wynegocjowanych) deskryptorów ruchu w systemie obsługi, stosuje się zasady (rysunek 7.3):

1. Reglamentacji zasobów systemu (np. metoda ciekącego wiadra). Zgodność wartości parametrów i reakcja na wykrycie niezgodności są sprawdzane przez mechanizmy implementowane w systemie obsługi strumieni jednostek danych.

2. Reglamentacji zapotrzebowania na zasoby systemu (np. metoda okna). Zgodność i reakcja na wykrycie niezgodności wartości parametrów jest sprawdzana przez mechanizmy implementowane w źródłach strumieni jednostek danych.



Rysunek 7.3. Zasady reglamentacji zasobów systemu i zapotrzebowania na zasoby

7.2.4.2. REGLAMENTACJA ZASOBÓW SYSTEMU

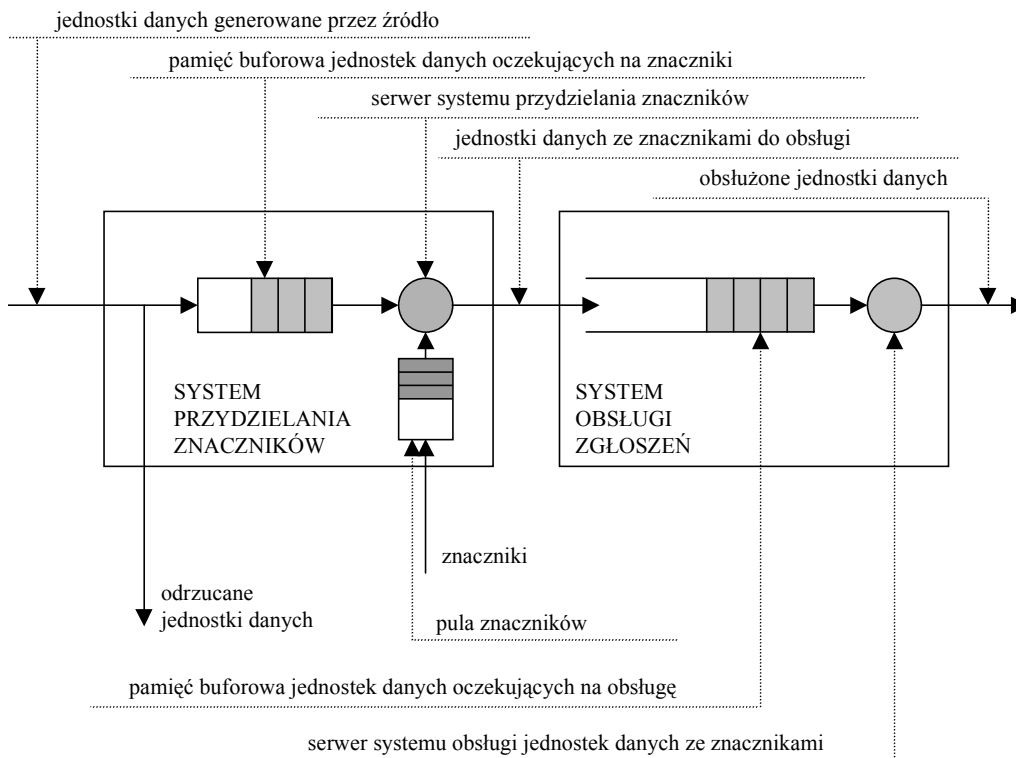
Przykładem rozwiązań, wykorzystujących w zadaniach utrzymania ruchu zasady reglamentowania zasobów, są różne odmiany algorytmu ciekącego wiadra.

7.2.4.2.1. ALGORYTM CIEKĄCEGO WIADRA

Podstawową algorytmu ciekącego wiadra LB (ang. *Leaky Bucket*) jest rozpowszechniona w algorytmach przeciwdziałania przeciążeniom oraz sterowania przepływem dostępu idea, zgodnie z którą warunkiem uzyskania dostępu nowego wywołania do sieci (systemu obsługi) jest posiadanie znacznika (przepustki, żetonu) (ang. *token*), pobieranego z odpowiedniego zbioru (puli) znaczników (ang. *token pool*). Jednostka danych (zgłoszenie), napływająca do systemu obsługi, pobiera znacznik w serwerze (stanowisku obsługi) algorytmu LB i natychmiast opuszcza stanowisko przydzielania znaczników, pod warunkiem, że pula znaczników nie jest pusta. Stała szybkość generowania znaczników powoduje, że algorytm LB może być stosowany

do sterowania ruchem jednostek danych napływających ze szczytową lub ze średnią szybkością.

Implementacja algorytmu LB, podobnie jak implementacja każdego innego algorytmu utrzymywania ruchu, wiąże się z wprowadzeniem do węzła źródłowego (dostępowego) systemu obsługi dodatkowego modułu, pośredniczącego w transferze jednostek danych pomiędzy źródłem jednostek danych i stanowiskiem obsługi jednostek danych (rysunek 7.4).



Rysunek 7.4. Struktura funkcjonalna węzła źródłowego (dostępowego) systemu obsługi

Strumień jednostek danych, generowany przez źródło w ramach wywołania wcześniej zaakceptowanego do obsługi przez sieć, jest przekazywany do systemu przydzielania znaczników, którego zadaniem jest, w zależności od implementowanej wersji mechanizmu LB, sprawdzanie zgodności wartości parametrów generowanego strumienia jednostek danych z wartościami parametrów ustalonych w kontrakcie ruchowym, tzn. wartości parametrów wykorzystanych do podjęcia decyzji o przyjęciu żądania dostępu wywołania do obsługi. Wszystkie jednostki danych, które nie zostały odrzucone przez system przydzielania znaczników, są przekazywane do systemu obsługi jednostek danych. Zadanie systemu przyznawania znaczników można interpretować jako zadanie filtrowania strumienia jednostek danych generowanych przez źró-

dło. Wartości parametrów filtra to wartości parametrów kontraktu zawartego na etapie przyjmowania żądania obsługi wywołania.

Algorytm LB charakteryzuje się tym, że znaczniki, napływające do puli znaczników, są generowane ze stałą szybkością. Pojemność puli znaczników, w której znaczniki mogą oczekiwać na jednostki danych, jest ograniczona. Gdy pojemność puli znaczników jest całkowicie zajęta, napływające znaczniki są tracone. W przypadku dwustanowego źródła jednostek danych (ang. *on/off source*) pojemność puli znaczników determinuje maksymalną długość przedziału aktywności źródła i liczbę jednostek danych przekazywanych w czasie trwania przedziału aktywności źródła. Szybkość napływu jednostek danych do systemu obsługi może być większa od szybkości generowania znaczników. Maksymalna liczba jednostek danych przekazywanych w czasie trwania przedziału aktywności źródła jest większa od pojemności puli znaczników. Znaczniki pobierane z puli znaczników przez nadchodzące jednostki danych są zastępowane nowymi znacznikami, napływającymi ze stałą szybkością. Maksymalna liczba jednostek danych zależy od wartości: szczytowej szybkości generowania jednostek danych, pojemności puli znaczników i szybkości generowania znaczników.

O jakości działania algorytmu LB decydują:

- całkowita, maksymalna liczba znaczników dostępnych w systemie obsługi dla napływających jednostek danych,

- sposób dystrybucji całkowitej liczby dostępnych znaczników pomiędzy poszczególne węzły dostępowe systemu obsługi (w przypadku wielowęzłowego systemu obsługi),

- szybkość generowania znaczników,

- wartość ilorazu maksymalnej liczby jednostek danych przyjmowanych do obsługi w systemie obsługi lub w danym węźle dostępu oraz maksymalnej liczby znaczników dostępnych w systemie lub danym węźle dostępu,

- pojemności pamięci buforowych dostępnych dla jednostek danych w systemie obsługi oraz w poszczególnych węzłach systemu obsługi.

Stosowanie algorytmu LB, podobnie jak implementacja dowolnego mechanizmu utrzymywania ruchu, zawsze wiąże się ze wzrostem średnich wartości opóźnienia obsługiwanych jednostek danych. Zwiększenie liczby znaczników dostępnych jednostkom danych, dla danej szybkości napływu jednostek danych, powoduje zwiększenie:

- wartości opóźnienia jednostek danych wnoszonego przez system obsługi,

- wartości prawdopodobieństwa obsługi dłuższych paczek jednostek danych, tj. wzrost jakości obsługi jednostek danych w poszczególnych połączeniach składowych połączeń zagregowanych, kosztem degradacji średniej jakości obsługi w połączeniach zagregowanych.

Podstawą wyróżnienia różnych wersji algorytmu LB są przyjęte w systemie obsługi:

- zasady dostępu jednostek danych do znaczników, tzn. istnienie lub brak możliwości oczekiwania jednostki danych na znacznik, gdy w chwili napływu jednostki danych w węźle dostępu brak jest znaczników; istnienie lub brak możliwości kolejowania jednostek danych w węźle dostępowym do systemu obsługi,

➤ restrykcyjność systemu obsługi jednostek danych, tzn. istnienie lub brak możliwości obsługi w systemie obsługi jednostki danych nie posiadającej znacznika.

Przyjęcie jednej z czterech możliwych kombinacji zasad dostępu do systemu obsługi i restrykcyjności systemu obsługi jest podstawą wyróżnienia czterech podstawowych wersji algorytmu LB (rysunek 7.5):

1. Wersja LB(a) – jeżeli w chwili napływu jednostki danych do systemu przydzielania znaczników w puli znaczników brak jest znacznika wolnego, to z powodu braku możliwości oczekiwania jednostki danych na znacznik napływająca jednostka danych jest odrzucana (tracona); w innym przypadku jednostka danych jest obsługiwana.

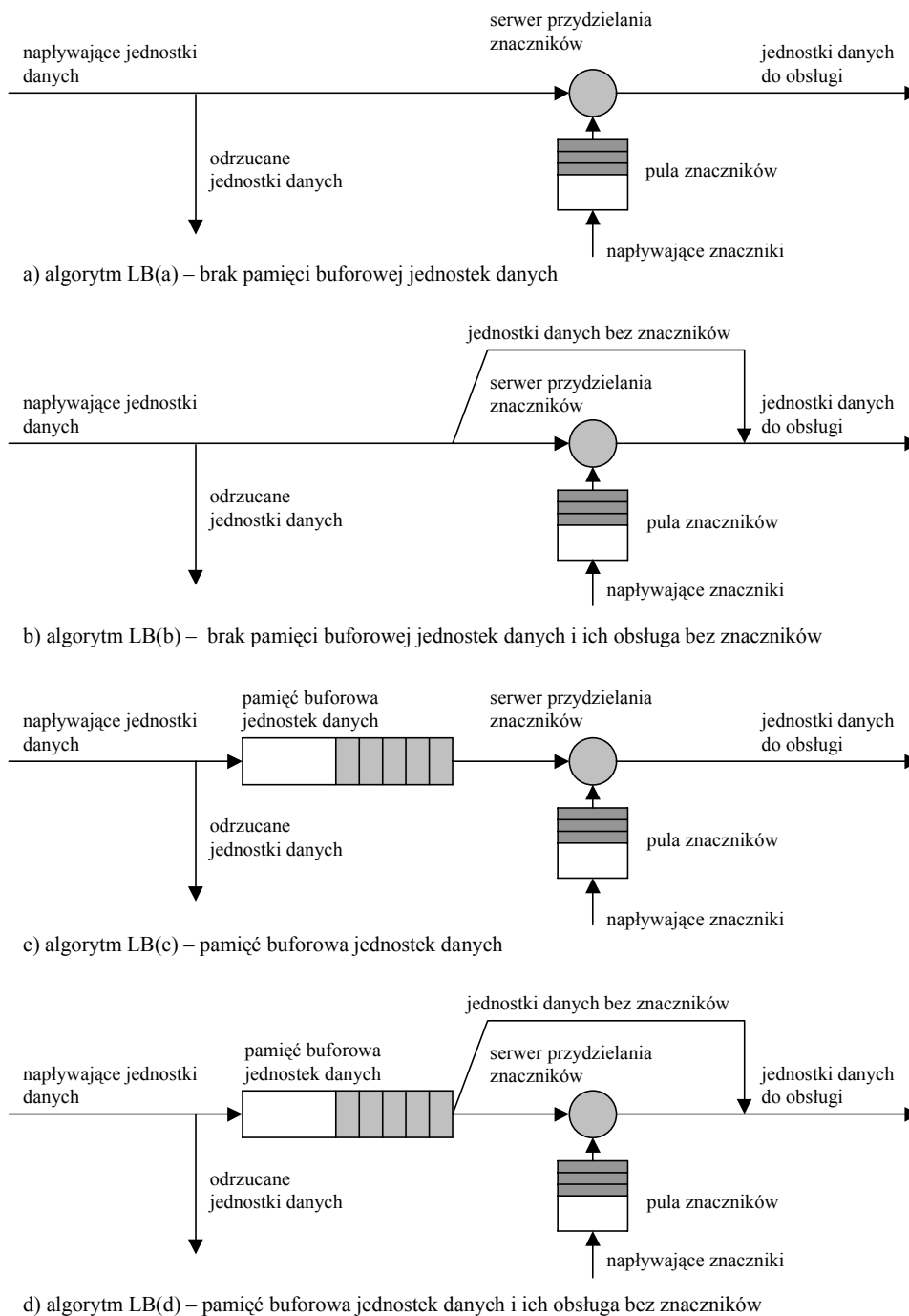
2. Wersja LB(b) – jeżeli w chwili napływu jednostki danych pula znaczników jest pusta, a liczba jednostek danych w systemie obsługi jednostek danych (kryterium globalne) lub węzle dostępu do systemu obsługi jednostek danych (kryterium lokalne) jest mniejsza od ustalonej maksymalnej wartości progowej liczby jednostek danych (globalnej lub lokalnej), to jednostka danych jest odpowiednio znakowana i przyjmowana do obsługi; w innym przypadku jednostka danych jest odrzucana.

3. Wersja LB(c) – jeżeli w chwili napływu jednostki danych pula znaczników jest pusta, to jednostka danych jest przyjmowana do systemu przydzielania znaczników wtedy i tylko wtedy, gdy jest wolne miejsce (bufor) w pamięci buforowej o skończonej pojemności i oczekuje w kolejce na napływ znacznika do puli znaczników; w innym przypadku jednostka danych jest odrzucana.

4. Wersja LB(d) – jeżeli w chwili napływu jednostki danych w puli znaczników brak jest znacznika oraz miejsca w pamięci buforowej o skończonej pojemności, a liczba jednostek danych (w systemie obsługi zgłoszeń lub węzle dostępu systemu obsługi zgłoszeń) jest mniejsza od ustalonej maksymalnej wartości progowej liczby jednostki danych, to jednostka danych jest odpowiednio znakowana i przyjmowana do obsługi; w innym przypadku jednostka danych jest odrzucana.

W wersjach LB(a) i LB(b) algorytmu jednostki danych napływające do systemu przydzielania znaczników są albo przyjmowane do systemu obsługi, albo są odrzucone. Wersje LB(a) i LB(b) różni to, że w przypadku drugiej akceptowane jednostki danych należą do dwóch klas jednostek danych. Odpowiednio znakowane jednostki danych bez znaczników są tracone w pierwszej kolejności wtedy, gdy występuje przeciążenie systemu obsługi. W wersjach LB(a) i LB(b) system przydzielania znaczników jest przezroczysty dla strumieni jednostek danych. W przypadku przydzielenia znacznika nie wnosi opóźnienia jednostki danych.

W wersjach LB(c) i LB(d) algorytmu przyjęcie jednostki danych do systemu przydzielania znaczników nie jest tożsame z przyjęciem jednostki danych do obsługi. Ewentualność konieczności oczekiwania na znacznik powoduje możliwość powstania opóźnienia. System przydzielania znaczników nie jest przezroczysty dla obsługiwanych strumieni jednostek danych. Wartość opóźnienia wnoszonego przez system przydzielania znaczników zależy od pojemności pamięci buforowej determinującej maksymalny czas oczekiwania jednostki danych na znacznik.



Rysunek 7.5. Podstawowe wersje algorytmu LB, wyróżnione z punktu widzenia pojemności pamięci buforowej jednostek danych i obsługi jednostek danych bez znaczników

W wersjach LB(b) i LB(d) algorytmu możliwe jest przyjęcie jednostek danych, których zarówno obsłużenie, jak i jakość obsługi nie są gwarantowane. Jednostki danych przyjęte bez znacznika są odpowiednio znakowane i obsługiwane warunkowo. Obsługa warunkowa oznacza, że jednostki danych bez znaczników są obsługiwane tylko wtedy, gdy ich obsługa nie obniża jakości obsługi jednostek danych ze znacznikami. Oznakowane wcześniej jednostki danych są odrzucone w pierwszej kolejności, jeżeli w czasie ich obsługi wystąpi stan przeciążenia systemu obsługi. W razie wystąpienia stanu przeciążenia odrzucanie jednostek danych bez znaczników jest rodzajem kary za nieznaną wartość parametrów charakteryzujących strumień jednostek danych i (lub) za niedotrzymanie warunków kontraktu.

Wersje LB(a) i LB(c) algorytmu są wersjami konserwatywnymi, zorientowanymi na dysponowanie zasobami systemu w sposób gwarantujący przyjmowanym do obsługi jednostkom danych wynegocjowanej w kontrakcie ruchowym jakości obsługi.

Wersje LB(b) i LB(d) algorytmu są wersjami, w których korzysta się ze statystycznych właściwości ruchu generowanego przez źródła jednostek danych. Podejmowane jest ryzyko zwiększenia stopnia wykorzystania zasobów systemu obsługi za cenę niewykonania obsługi jednostek danych przyjętych do obsługi lub obniżenia jakości obsługi jednostki danych, które są przyjmowane do obsługi jako nadmiarowe.

Dodatkowe, możliwe wersje algorytmu LB są generowane wtedy, gdy pojemność puli znaczników jest zerowa, tzn. jednostki danych są wprowadzane do systemu obsługi z szybkością nie większą od szybkości napływu znaczników. W takim przypadku, będącym szczególnym przypadkiem wersji LB(c) algorytmu, jednostka danych może oczekiwać na napływ znacznika, ale nie ma możliwości oczekiwania znaczników na jednostki danych. Jeżeli w chwili napływu znacznika brak jest jednostki danych oczekującej na znacznik, to dany znacznik jest tracony.

W niektórych odmianach algorytmu LB(c) i LB(d) do kontroli zgodności napływu jednostki danych do systemu obsługi z warunkami kontraktu ruchowego korzysta się z dwóch rodzajów znaczników, tzn. znaczników o dwóch kolorach (ang. *green and red tokens*), z tym że każda jednostka danych może uzyskać tylko jeden znacznik. Znaczniki o różnych kolorach są generowane według różnych schematów, w szczególności z różną szybkością. Znaczniki jednego koloru (zwykle zielone) są wykorzystywane (globalnie lub lokalnie) przez jednostki danych wtedy, gdy liczba jednostek danych, obsługiwanych i oczekujących na obsługę w systemie przydzielania znaczników, jest mniejsza od ustalonej wartości progowej. Znaczniki drugiego koloru (zwykle czerwone) są wykorzystywane przez napływające jednostki danych wtedy, gdy liczba jednostek danych, oczekujących w kolejce pamięci buforowej systemu przydzielania znaczników, jest większa od ustalonego, założonego limitu jednostek danych; zwłaszcza wtedy, gdy przekroczone są ustalone wartości progowe zajętości pamięci buforowej. Jednostki danych ze znacznikami czerwonymi mają zwykle niższy priorytet obsługi. W razie zagrożenia systemu obsługi powstaniem stanu przeciążenia, te jednostki danych są usuwane z systemu obsługi w pierwszej kolejności, tzn. przed jednostkami danych ze znacznikami zielonymi.

Różne wersje algorytmu LB mogą być implementowane w różnych systemach obsługi jednostek danych w różnym zakresie. Przedmiotem sterowania przepływem, tożsamego z kontrolą zgodności wartości parametrów strumienia jednostek danych z wartościami parametrów kontraktu ruchowego, mogą być deskrytory ruchu dotyczące:

1. Systemu jako całości, gdzie zapotrzebowanie na znaczniki i pojemność pamięci buforowej jest kompensowane w ramach globalnych pul znaczników i pamięci buforowych systemu przydzielania znaczników (kontrola jednopoziomowa).

2. Wybranych części systemu (zwłaszcza węzłów dostępowych), gdzie zapotrzebowanie na znaczniki i pojemność pamięci buforowej jest kompensowane w ramach limitów lokalnych, których suma nie przekracza limitów globalnych (kontrola dwupoziomowa).

3. Indywidualnych połączeń (np. połączeń wirtualnych), gdzie zapotrzebowanie na znaczniki i pojemność pamięci buforowej jest kompensowane w ramach limitów przydzielanych połączeniom indywidualnym, których suma:

- nie przekracza limitów globalnych (kontrola dwupoziomowa),
- nie przekracza limitów lokalnych, których suma, z kolei, nie przekracza limitów globalnych (kontrola trypoziomowa).

Zróznicowane sposoby obsługi jednostek danych przez różne wersje algorytmu LB charakteryzuje rozmaita złożoność implementacji i różna efektywność. Zwiększenie liczby poziomów realizacji funkcji utrzymywania ruchu poprawia na ogół efektywność oraz wzrost złożoności implementacji i realizacji mechanizmów utrzymywania ruchu.

Różny zakres i złożoność implementacji algorytmu LB ilustruje zastosowanie tego algorytmu np. w sieciach ATM i TCP/IP. Algorytm LB może być implementowany zarówno na poziomie ścieżki wirtualnej VP (*Virtual Path*), jak i na poziomie kanału wirtualnego VC (*Virtual Channel*) (w warstwowej, hierarchicznej architekturze ATM poziom kanału wirtualnego jest nadrzędny wobec poziomu ścieżki wirtualnej).

Administrowanie ścieżką wirtualną na ogół jest związane z monitorowaniem wartości szczytowych szybkości transferu komórek. Na potrzeby utrzymywania ruchu na ścieżce wirtualnej, tzn. do nadzorowania zgodności wartości szczytowej szybkości transferu komórek z warunkami kontraktu ruchowego, warunkiem wystarczającym jest zastosowanie algorytmu LB, którego szybkość generowania znaczników jest równa pojemności (maksymalnej przepustowości) ścieżki wirtualnej.

Administrowanie kanałem wirtualnym wymaga zwykle monitorowania co najmniej wartości szczytowych szybkości transferu ścieżek wirtualnych i średniej szybkości transferu zagregowanego ruchu (ruchu transferowanego w ścieżkach wirtualnych kanału wirtualnego). Jednoczesna realizacja tych zadań wymaga stosowania hierarchii mechanizmów LB:

- warunkiem koniecznym efektywności kontroli warunków kontraktu w kanale wirtualnym jest zastosowanie algorytmów LB oddzielnie dla każdej ze ścieżek wirtualnych, wchodzących w skład kanału wirtualnego, o szybkościach generowania znaczników równych pojemności ścieżek wirtualnych,

➤ warunkiem wystarczającym efektywności kontroli warunków kontraktu w kanale wirtualnym jest kontrola zgodności średniej szybkości transferu komórek z warunkami kontraktu, tzn. warunkiem jest dodatkowa implementacja algorytmu LB o szybkości generowania znaczników równej pojemności kanału wirtualnego.

7.2.4.2.2. ANALIZA DZIAŁANIA ALGORYTMU LB

W literaturze przedmiotu istnieje wiele różnych modeli algorytmu LB, których liczba wynika z:

1. Istnienia wielu różnych wersji algorytmu LB.
2. Wykorzystywania różnych kryteriów jakości utrzymywania ruchu, zależnych od kryteriów przyjmowania wywołań w procedurach sterowania dostępem.

3. Przyjmowania do analizy działania algorytmu LB różnego rodzaju założeń upraszczających, dotyczących m.in.:

- charakterystyk zagregowanego strumienia jednostek danych i jego strumieni składowych,
- strategii obsługi kolejek jednostek danych oczekujących na znaczniki,
- strategii generowania znaczników oraz strategii obsługi kolejek znaczników.

W przykładowym zadaniu analizy działania systemu przydzielania znaczników, zgodnie z algorytmem LB(c), tzn. w wersji z możliwością oczekiwania jednostek danych na znaczniki i bez możliwości wprowadzania do obsługi jednostki danych bez znaczników, przyjmuje się następujące założenia:

- jednostki danych obsługiwane w systemie obsługi mają jednakową, stałą długość,
- wartości pojemności pamięci buforowych jednostki danych B_{JD} i znaczników B_{ZN} są skończone,

➤ znaczniki generowane są ze stałą szybkością v_{ZN} . $1/v_{ZN}$ jest stałą liczbą szczelin pomiędzy chwilami generacji kolejnych znaczników, tj. długość szczeliny jest równa długości czasu transferu jednej jednostki danych, tzn. ilorazowi długości jednostki danych i szczytowej szybkości napływu jednostek danych.

Jakość algorytmu LB(c) mierzy się wartością następujących wskaźników:

1. Prawdopodobieństwem utraty jednostki danych w pamięci buforowej jednostki danych.

2. Opóźnieniem jednostki danych wnoszonym przez pamięć buforową jednostek danych.

3. Dokładnością LB(c) w zadaniu utrzymywania ruchu, wyrażoną:

➤ maksymalną długością paczki jednostki danych obsługiwanej w systemie przydzielania znaczników bez ingerencji algorytmu LB(c),

➤ charakterystykami długości przedziałów czasu pomiędzy kolejnymi wyprowadzeniami jednostki danych z systemu przydzielania znaczników.

Wartości prawdopodobieństwa:

- utraty jednostki danych,
- średniej liczby szczelin pomiędzy kolejnymi wyprowadzeniami jednostek danych z systemu przydzielania znaczników,

zależą tylko od sumy pojemności pamięci buforowych jednostek danych i znaczników, tzn. pojemności całkowitej pamięci buforowej B ($B = B_{JD} + B_{ZN}$), a nie zależą od pojemności poszczególnych pamięci buforowych. Udział i wartości pojemności poszczególnych pamięci buforowych w danej, całkowitej pojemności pamięci buforowej, ma wpływ na wartości innych wskaźników jakości: dla danej wartości pojemności całkowitej B czas oczekiwania jednostek danych w kolejce pamięci buforowej wydłuża się wraz ze wzrostem B_{JD} i odwrotnie.

Jednym z możliwych sposobów charakteryzowania zagregowanego strumienia zgłoszeń, napływającego do systemu przydzielania znaczników (do algorytmu LB(c)), jest przedstawienie tego strumienia jako dwustanowego procesu Bernoulliego zmodulowanego procesem Markowa – MMBP (ang. *Markov Modulated Bernoulli Process*).

7.2.4.2.3. PROCES BERNOULLIEGO ZMODULOWANY PROCESEM MARKOWA

W procesie MMBP czas jest dzielony na szczeliny o jednakowej długości. W ogólnym przypadku procesu MMBP wartość prawdopodobieństwa zdarzenia, że w danej szczelinie znajduje się jednostka danych (zgłoszenie) jest realizacją zmiennej losowej z rozkładem dwumianowym (proces Bernoulliego). Zmienne wartości parametru rozkładu dwumianowego są realizacjami k -stanowego procesu Markowa, niezależnego od procesu napływu jednostek danych.

W końcu każdej szczeliny czasowej następuje zmiana stanu procesu Markowa zgodnie z następującą regułą: zmiana stanu procesu, z i -tego stanu do j -tego stanu, odbywa się z prawdopodobieństwem p_{ij} lub proces pozostaje w i -tym stanie z prawdopodobieństwem p_{ii} , gdzie $p_{i1} + p_{i2} + \dots + p_{ii} + \dots + p_{ik} = 1$ dla $i = 1, 2, \dots, k$. Wartości prawdopodobieństwa zdarzenia, że w i -tym stanie szczelina zawiera lub nie zawiera jednostki danych wynoszą odpowiednio: α_i i $(1 - \alpha_i)$.

Wartości prawdopodobieństwa przejść i prawdopodobieństwa napływu jednostek danych w szczelinach są wykorzystywane do znalezienia funkcji generacyjnej zmiennej losowej, oznaczanej przez $T(z)$, której realizacjami są długości przedziałów czasu pomiędzy kolejnymi chwilami napływu jednostek danych, tzn. długości czasu międzyzgłoszeniowego.

Proces MMBP jest charakteryzowany:

➤ kwadratową macierzą wartości prawdopodobieństw przejść $\mathbf{P} = [p_{ij}]_{k \times k}$,

➤ diagonalną macierzą wartości prawdopodobieństw napływu jednostek danych $\mathbf{A} = \text{diag} [\alpha_1, \alpha_2, \dots, \alpha_k]$.

Wartości prawdopodobieństwa zdarzenia, że w stanie ustalonym proces MMBP jest w i -tym stanie ($i = 1, 2, \dots, k$) – π_i , są rozwiązaniem systemu równań $\boldsymbol{\delta} \mathbf{P} = \boldsymbol{\delta}$, gdzie $\boldsymbol{\delta} = [\pi_1, \pi_2, \dots, \pi_k]$. Wartości prawdopodobieństw warunkowych b_i ($i = 1, 2, \dots, k$) zdarzeń polegających na tym, że proces MMBP znajduje się w i -tym stanie i napływa jednostka danych, są rozwiązaniami równania $\mathbf{b} = \boldsymbol{\delta} \mathbf{A} / \boldsymbol{\delta} \mathbf{a}$, gdzie $\mathbf{a} = [\alpha_1, \alpha_2, \dots, \alpha_k]^T$ i $\mathbf{b} = [b_1, b_2, \dots, b_k]$.

Funkcję generującą momenty $T(z)$ określa wyrażenie:

$$T(z) = \mathbf{b}z\{\mathbf{I} - z\mathbf{P}(\mathbf{I} - \tilde{\mathbf{A}})\}^{-1} \mathbf{P}\tilde{\mathbf{a}}.$$

Kolejne n -te pochodne $T^{(n)}(z)$ funkcji generującej momenty $T(z)$ są dane wyrażeniami:

$$\begin{aligned} T^{(n)}(z) &= \\ &= n! \mathbf{b} \left[\{\mathbf{I} - z\mathbf{P}(\mathbf{I} - \tilde{\mathbf{A}})\}^{-1} \mathbf{P}(\mathbf{I} - \tilde{\mathbf{A}}) \right]^{n-1} \{\mathbf{I} - z\mathbf{P}(\mathbf{I} - \tilde{\mathbf{A}})\}^{-1} [\mathbf{I} + z\mathbf{P}(\mathbf{I} - \tilde{\mathbf{A}})\{\mathbf{I} - z\mathbf{P}(\mathbf{I} - \tilde{\mathbf{A}})\}^{-1}] \mathbf{P}\tilde{\mathbf{a}}; \end{aligned}$$

n -ty moment zmiennej losowej T , oznaczany przez $T[n]$, jest równy n -tej pochodnej funkcji generującej momenty $T^{(n)}(z)$ dla $z = 1$, tzn.:

$$E\{T[n]\} = T^{(n)}(1) = n! \mathbf{b} [\mathbf{I} - \mathbf{P}(\mathbf{I} - \tilde{\mathbf{A}})]^{-(n+1)} [\mathbf{P}(\mathbf{I} - \tilde{\mathbf{A}})]^{n-1} \mathbf{P}\tilde{\mathbf{a}},$$

a wartość prawdopodobieństwa, że czas międzyzgłoszeniowy jest równy m szczełom:

$$\mathbf{P}\{T = m\} = \mathbf{b} [\mathbf{P}(\mathbf{I} - \tilde{\mathbf{A}})]^{m-1} \mathbf{P}\tilde{\mathbf{a}}.$$

Jeżeli τ_l jest długością przedziału czasu pomiędzy $(l - 1)$ i l -tą jednostkami danych, to wartość momentu pierwszego rzędu dwuwymiarowej zmiennej losowej (τ_l, τ_{l+k}) , oznaczona przez $E\{\tau_l, \tau_{l+k}\}$, jest wartością wyrażenia:

$$E\{\tau_l, \tau_{l+k}\} = \mathbf{b} [\mathbf{I} - \mathbf{P}(\mathbf{I} - \tilde{\mathbf{A}})]^{-2} \mathbf{P}\tilde{\mathbf{A}} T^{(k-1)}(z=1) [\mathbf{I} - \mathbf{P}(\mathbf{I} - \tilde{\mathbf{A}})]^{-2} \mathbf{P}\tilde{\mathbf{a}},$$

a wartość współczynnika korelacji długości przedziału czasu międzyzgłoszeniowego dla danego procesu MMBP i dla opóźnienia równego k szczełom, jest wartością wyrażenia:

$$\Psi_k = \frac{E\{\tau_l, \tau_{l+k}\} - E\{\tau_l\}^2}{\text{var}\{\tau_l\}}.$$

W przypadku dwustanowego procesu MMBP macierze prawdopodobieństwa przejść \mathbf{P} i \mathbf{A} są następujące:

$$\mathbf{P} = \begin{bmatrix} p_{11} & 1 - p_{11} \\ 1 - p_{22} & p_{22} \end{bmatrix} \quad \tilde{\mathbf{A}} = \begin{bmatrix} \alpha_1 & 0 \\ 0 & \alpha_2 \end{bmatrix}.$$

Zgodnie z wartościami macierzy \mathbf{P} , jeżeli proces Markowa jest w stanie 1. (2.), to wartość prawdopodobieństwa pozostania w tym stanie jest równa p_{11} (p_{22}), a wartość prawdopodobieństwa przejścia do stanu 2. (1.) wynosi $1 - p_{11}$ ($1 - p_{22}$).

Wartości prawdopodobieństw w stanie ustalonym przebywania dwustanowego procesu MMBP w stanie i ($i \in \{1, 2\}$), równe odpowiednio π_1 i π_2 , są rozwiązaniami układu równań:

$$\begin{cases} [\pi_1, \pi_2] \mathbf{P} = [\pi_1, \pi_2] \\ \pi_1 + \pi_2 = 1 \end{cases},$$

$$\text{tzn. } \pi_1 = \frac{1-p_{22}}{2-p_{11}+p_{22}} \text{ i } \pi_2 = \frac{1-p_{11}}{2-p_{11}-p_{22}}.$$

Wartość prawdopodobieństwa zdarzenia, że w danej szczelinie znajduje się jednostka danych, tzn. wartość współczynnika wykorzystania źródła, jest dana wyrażeniem:

$$\rho = \pi_1 \alpha_1 + \pi_2 \alpha_2 = \frac{(1-p_{22})\alpha_1 + (1-p_{11})\alpha_2}{2-p_{11}-p_{22}}.$$

Przy założeniu, że jeżeli w danej szczelinie proces MMBP jest w i -tym stanie ($i \in \{1, 2\}$), to w następnej szczelinie:

Dla $i = 1$:

➤ proces MMBP może pozostać w stanie 1. i może napłynąć jednostka danych – prawdopodobieństwo takiego zdarzenia równe jest $p_{11}\alpha_1$,

➤ proces MMBP może przejść ze stanu 1. do stanu 2. i może napłynąć jednostka danych – prawdopodobieństwo takiego zdarzenia wynosi $(1-p_{11})\alpha_2$,

➤ proces MMBP może pozostać w stanie 1. i nie napłynie jednostka danych – prawdopodobieństwo takiego zdarzenia równe jest $p_{11}(1-\alpha_1)$,

➤ proces MMBP może przejść ze stanu 1. do 2. stanu i nie napłynie jednostka danych – prawdopodobieństwo takiego zdarzenia jest równe $(1-p_{11})(1-\alpha_2)$.

Dla $i = 2$:

➤ proces MMBP może pozostać w stanie 2. i może napłynąć jednostka danych – prawdopodobieństwo takiego zdarzenia równe jest $p_{22}\alpha_2$,

➤ proces MMBP może przejść z 2. do 1. stanu i może napłynąć jednostka danych – prawdopodobieństwo takiego zdarzenia wynosi $(1-p_{22})\alpha_1$,

➤ proces MMBP może pozostać w stanie 2. i nie napłynie jednostka danych – prawdopodobieństwo takiego zdarzenia równe jest $p_{22}(1-\alpha_2)$,

➤ proces MMBP może przejść z 2. do 1. stanu i nie napłynie jednostka danych – prawdopodobieństwo takiego zdarzenia jest równe $(1-p_{22})(1-\alpha_1)$.

Jeżeli proces MMBP, w danej szczelinie, jest w i -tym stanie a T_i ($i \in \{1, 2\}$) oznacza warunkową długość czasu oczekiwania (wyrażoną w szczelinach) na napływ następnej jednostki danych, to prawdziwe są następujące zależności:

dla T_1 :

$$\begin{cases} p(T_1 = 1) = p_{11}\alpha_1 + (1-p_{11})\alpha_2 \\ p(T_1 = 1 + T_1) = p_{11}(1-\alpha_1) \\ p(T_1 = 1 + T_2) = (1-p_{11})(1-\alpha_2) \end{cases}$$

oraz **dla T_2 :**

$$\begin{cases} p(T_2 = 1) = p_{22}\alpha_2 + (1-p_{22})\alpha_1 \\ p(T_2 = 1 + T_2) = p_{22}(1-\alpha_2) \\ p(T_2 = 1 + T_1) = (1-p_{22})(1-\alpha_1) \end{cases}.$$

Transformaty Z długości czasów oczekiwania na napływ zgłoszenia $T_1(z)$ i $T_2(z)$ dane są układem równań:

$$\begin{aligned} T_1(z) &= zp(T_1=1) + zT_1(z)p(T_1=1+T_1) + zT_2(z)p(T_1=1+T_2) = \\ &= z[p_{11}\alpha_1 + (1-p_{11})\alpha_2] + zT_1(z)[p_{11}(1-\alpha_1)] + zT_2(z)[(1-p_{11})(1-\alpha_2)], \\ T_2(z) &= zp(T_2=1) + zT_2(z)p(T_2=1+T_2) + zT_1(z)p(T_2=1+T_1) = \\ &= z[p_{22}\alpha_2 + (1-p_{22})\alpha_1] + zT_1(z)[(1-p_{22})(1-\alpha_1)] + zT_2(z)[p_{22}(1-\alpha_2)], \end{aligned}$$

których rozwiązanie jest następujące:

$$\begin{aligned} T_1(z) &= \frac{[p_{11}\alpha_1 + (1-p_{11})\alpha_2]z + \alpha_1(1-\alpha_2)(1-p_{11}-p_{22})z^2}{1 - [p_{11}(1-\alpha_1) + p_{22}(1-\alpha_2)]z + (1-\alpha_1)(1-\alpha_2)(p_{11} + p_{22} - 1)z^2}, \\ T_2(z) &= \frac{[p_{22}\alpha_2 + (1-p_{22})\alpha_1]z + \alpha_2(1-\alpha_1)(1-p_{11}-p_{22})z^2}{1 - [p_{22}(1-\alpha_2) + p_{11}(1-\alpha_1)]z + (1-\alpha_1)(1-\alpha_2)(p_{11} + p_{22} - 1)z^2}. \end{aligned}$$

Z założonej dla procesu MMBP niezależności procesu napływu jednostek danych i procesu Markowa wynika, że wartości prawdopodobieństw zdarzeń, że proces MMBP jest w i -tym ($i \in \{1, 2\}$) stanie i napływa jednostka danych, oznaczone przez a_i , są następujące:

$$a_1 = \pi_1\alpha_1 \quad \text{i} \quad a_2 = \pi_2\alpha_2.$$

Z równości tych wynika, że wartości prawdopodobieństw b_i , iż proces MMBP jest w i -tym ($i \in \{1, 2\}$) stanie, gdy napływa jednostka danych, są określone zależnościami:

$$\begin{aligned} b_1 &= \frac{a_1}{a_1 + a_2} = \frac{\pi_1\alpha_1}{\pi_1\alpha_1 + \pi_2\alpha_2} = \frac{(1-p_{22})\alpha_1}{\alpha_1(1-p_{22}) + \alpha_2(1-p_{11})}, \\ b_2 &= \frac{a_2}{a_1 + a_2} = \frac{\pi_2\alpha_2}{\pi_1\alpha_1 + \pi_2\alpha_2} = \frac{(1-p_{11})\alpha_2}{\alpha_1(1-p_{22}) + \alpha_2(1-p_{11})}. \end{aligned}$$

Transformata Z długości czasu międzypogłoszeniowego oznaczonego przez T , tzn. $T(z)$, jest sumą ważoną transformat Z wartości warunkowych (sumowanie wartości warunkowych długości czasu oczekiwania na napływ jednostki danych T_i ($i \in \{1, 2\}$) w poszczególnych stanach procesu Markowa (w wyniku sumowania wartości warunkowych T_i otrzymywana jest wartość bezwarunkowa $T(z)$):

$$T(z) = b_1T_1(z) + b_2T_2(z) = \frac{\beta_1z + \beta_2z^2}{\eta_0 + \eta_1z + \eta_2z^2},$$

gdzie:

$$\beta_1 = (1 - p_{22})\alpha_1[p_{11}\alpha_1 + (1 - p_{11})\alpha_2] + (1 - p_{11})\alpha_2[p_{22}\alpha_2 + (1 - p_{22})\alpha_1],$$

$$\beta_2 = (1 - p_{11} - p_{22})[(1 - p_{22})(1 - \alpha_2)\alpha_1^2 + (1 - p_{11})(1 - \alpha_1)\alpha_2^2],$$

$$\eta_0 = (1 - p_{22})\alpha_1 + (1 - p_{11})\alpha_2,$$

$$\eta_1 = -[(1 - p_{22})\alpha_1 + (1 - p_{11})\alpha_2][p_{22}(1 - \alpha_2) + p_{11}(1 - \alpha_1)],$$

$$\eta_2 = (1 - \alpha_1)(1 - \alpha_2)(p_{11} + p_{22} - 1)[(1 - p_{22})\alpha_1 + (1 - p_{11})\alpha_2].$$

W ogólnym przypadku n -ty moment zmiennej losowej długości czasu międzyzgłoszeniowego T jest wyznaczany (z wykorzystaniem funkcji generującej momenty $T(z)$ dla $z = 1$) według zależności:

$$E\{T^n\} = n! \mathbf{b}(\mathbf{I} - \mathbf{F})^{-n} \mathbf{F}^{n-1} \mathbf{e},$$

w której: $\mathbf{b} = [b_1, b_2]$ jest wektorem wartości prawdopodobieństw zdarzeń, polegających na tym, że proces MMBP znajduje się i -tym stanie ($i \in \{1, 2\}$), $\mathbf{e} = [1, 1]^T$, a $\mathbf{F} = \mathbf{P}(\mathbf{I} - \mathbf{A}) = [f_{ij}]_{2 \times 2}$ jest macierzą prawdopodobieństw, daną wyrażeniem:

$$\mathbf{F} = \begin{bmatrix} p_{11} & 1 - p_{11} \\ 1 - p_{22} & p_{22} \end{bmatrix} \times \begin{bmatrix} 1 - \alpha_1 & 0 \\ 0 & 1 - \alpha_2 \end{bmatrix} = \begin{bmatrix} p_{11}(1 - \alpha_1) & (1 - p_{11})(1 - \alpha_{22}) \\ (1 - p_{22})(1 - \alpha_1) & p_{22}(1 - \alpha_2) \end{bmatrix}.$$

Wartość średnia $E\{T\}$ i wariancja $c^2\{T\}$ długości czasu międzyzgłoszeniowego dane są przez odpowiednie pochodne funkcji generującej momenty $T(z)$:

$$E\{T\} = \frac{2 - p_{11} - p_{22}}{(1 - p_{22})\alpha_1 + (1 - p_{11})\alpha_2},$$

$$c^2\{T\} = \frac{2[(1 - p_{22})\alpha_1 + (1 - p_{11})\alpha_2]}{(1 - p_{22})\alpha_1 + (1 - p_{11})\alpha_2 + (1 - p_{11})(1 - p_{22})\alpha_1\alpha_2} - \frac{(1 - p_{22})\alpha_1 + (1 - p_{11})\alpha_2}{2 - p_{11} - p_{22}} + \frac{2(p_{11} + p_{22} - 1)[(1 - p_{11})\alpha_1 + (1 - p_{22})\alpha_2][(1 - p_{22})\alpha_1 + (1 - p_{11})\alpha_2]}{(2 - p_{11} - p_{22})^2 [(1 - p_{22})\alpha_1 + (1 - p_{11})\alpha_2 + (p_{11} + p_{22} - 1)\alpha_1\alpha_2]} - 1.$$

Procedura wyznaczania korelacji długości czasu między zgłoszeniowego jest podobna do procedury wyznaczania wartości momentów długości czasu międzyzgłoszeniowego.

Przyjmując, że t_{ij} oznacza długość przedziału czasu pomiędzy daną szczeliną, w której proces MMBP jest w i -tym stanie, a szczeliną, w której napływa następna jednostka danych, i w której proces MMBP jest w j -tym stanie, oznacza, że w przypadku omawianego dwustanowego procesu ($i, j \in \{1, 2\}$) zachodzi:

dla t_{11} :

$$\begin{cases} p(t_{11} = 1) = p_{11}\alpha_1 \\ p(t_{11} = 1 + t_{11}) = p_{11}(1 - \alpha_1) \\ p(t_{11} = 1 + t_{21}) = (1 - p_{11})(1 - \alpha_2) \end{cases},$$

dla t_{12} :

$$\begin{cases} p(t_{12} = 1) = (1 - p_{11})\alpha_2 \\ p(t_{12} = 1 + t_{12}) = p_{11}(1 - \alpha_1) \\ p(t_{12} = 1 + t_{22}) = (1 - p_{11})(1 - \alpha_2) \end{cases},$$

dla t_{21} :

$$\begin{cases} p(t_{21} = 1) = (1 - p_{22})\alpha_1 \\ p(t_{21} = 1 + t_{21}) = p_{22}(1 - \alpha_2) \\ p(t_{21} = 1 + t_{11}) = (1 - p_{22})(1 - \alpha_1) \end{cases},$$

dla t_{22} :

$$\begin{cases} p(t_{22} = 1) = p_{22}\alpha_2 \\ p(t_{22} = 1 + t_{22}) = p_{22}(1 - \alpha_2) \\ p(t_{22} = 1 + t_{12}) = (1 - p_{22})(1 - \alpha_1) \end{cases}.$$

Po oznaczeniu przez:

s_l – stan procesu MMBP w chwili napływu l -tej jednostki danych,

τ_l – długość przedziału czasu pomiędzy napływem dwóch kolejnych jednostek danych: $(l - 1)$ -szą oraz l -tą,

τ_{ij} – długość przedziału czasu pomiędzy napływem dwóch kolejnych jednostek danych: $(l - 1)$ -szą oraz l -tą wtedy, gdy w chwili napływu l -tej jednostki danych proces MMBP jest w j -tym stanie,

spełniony jest warunek:

$$g_{ij}(z) = E\{\tau_{ij}(z) | s_{l-1} = i\} = E\{t_{ij}(z)\},$$

gdzie transformaty Z wyrażeń g_{ij} , dla $i, j \in \{1, 2\}$, określa układ równań:

$$g_{11}(z) = p_{11}\alpha_1 z + p_{11}(1-\alpha_1)zg_{11}(z) + (1-p_{11})(1-\alpha_2)zg_{21}(z),$$

$$g_{12}(z) = (1-p_{11})\alpha_2 z + p_{11}(1-\alpha_1)zg_{12}(z) + (1-p_{11})(1-\alpha_2)zg_{22}(z),$$

$$g_{21}(z) = (1-p_{22})\alpha_1 z + p_{22}(1-\alpha_2)zg_{21}(z) + (1-p_{22})(1-\alpha_1)zg_{11}(z),$$

$$g_{22}(z) = p_{22}\alpha_2 z + p_{22}(1-\alpha_2)zg_{22}(z) + (1-p_{22})(1-\alpha_1)zg_{12}(z).$$

Po wprowadzeniu dodatkowych oznaczeń:

$$h_i(z) = E\{\tau_l(z) | s_{l-1} = i\} \text{ i } c_i(z_1, z_2) = E\{\tau_{l-1}(z_1), \tau_l(z_2) | s_{l-2} = i\} \text{ dla } i \in \{1, 2\},$$

$$\mathbf{G}(z) = \begin{bmatrix} g_{11}(z) & g_{12}(z) \\ g_{21}(z) & g_{22}(z) \end{bmatrix}, \mathbf{H}(z) = \begin{bmatrix} h_1(z) \\ h_2(z) \end{bmatrix} \text{ i } \mathbf{C}(z_1, z_2) = \begin{bmatrix} c_1(z_1, z_2) \\ c_2(z_1, z_2) \end{bmatrix},$$

dla $i, j \in \{1, 2\}$ zachodzą zależności:

$$h_1(z) = g_{11}(z) + g_{12}(z) \text{ i } h_2(z) = g_{21}(z) + g_{22}(z),$$

$$c_1(z_1, z_2) = g_{11}(z_1)h_1(z_2) + g_{12}(z_1)h_2(z_2) \text{ i } c_2(z_1, z_2) = g_{21}(z_1)h_1(z_2) + g_{22}(z_1)h_2(z_2),$$

tzn. $\mathbf{C}(z_1, z_2) = \mathbf{G}(z_1)\mathbf{H}(z_2)$.

Zapis układu równań $g_{ij}(z)$ ($i, j \in \{1, 2\}$) w postaci macierzowej jest następujący:

$$\begin{bmatrix} 1-p_{11}(1-\alpha_1)z & -(1-p_{11})(1-\alpha_2)z \\ -(1-p_{22})(1-\alpha_1)z & 1-p_{22}(1-\alpha_2)z \end{bmatrix} \times \begin{bmatrix} g_{11}(z) & g_{12}(z) \\ g_{21}(z) & g_{22}(z) \end{bmatrix} = \\ = \begin{bmatrix} p_{11}\alpha_1 z & (1-p_{11})\alpha_2 z \\ (1-p_{22})\alpha_1 z & p_{22}\alpha_2 z \end{bmatrix},$$

który jest równoważny zapisowi:

$$\left(\left(\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} - z \begin{bmatrix} p_{11} & 1-p_{11} \\ 1-p_{22} & p_{22} \end{bmatrix} \right) \times \left(\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} - \begin{bmatrix} \alpha_1 & 0 \\ 0 & \alpha_2 \end{bmatrix} \right) \right) \times \begin{bmatrix} g_{11}(z) & g_{12}(z) \\ g_{21}(z) & g_{22}(z) \end{bmatrix} = \\ = \begin{bmatrix} p_{11} & 1-p_{11} \\ 1-p_{22} & p_{22} \end{bmatrix} \times \begin{bmatrix} \alpha_1 & 0 \\ 0 & \alpha_2 \end{bmatrix} z,$$

tzn. zapisowi $[\mathbf{I} - z\mathbf{P}(\mathbf{I} - \mathbf{A})]\mathbf{G}(z) = \mathbf{P}\mathbf{A}z$.

Wartości macierzy $\mathbf{P}(\mathbf{I} - \mathbf{A})$ reprezentują prawdopodobieństwa przejścia pomiędzy stanami procesu MMBP bez napływu jednostki danych, a wartości macierzy $\mathbf{P}\mathbf{A}$ re-

prezentują przejścia pomiędzy stanami procesu MMBP w przypadku napływu jednostki danych.

Przekształcenie wyrażenia:

$$[\mathbf{I} - z\mathbf{P}(\mathbf{I} - \mathbf{A})]\mathbf{G}(z) = \mathbf{P}\mathbf{A}z$$

ze względu na $\mathbf{G}(z)$ prowadzi do równania:

$$\mathbf{G}(z) = [\mathbf{I} - z\mathbf{P}(\mathbf{I} - \mathbf{A})]^{-1} \mathbf{P}\mathbf{A}z.$$

Ze względu na to, że $h_1(z) = g_{11}(z) + g_{12}(z)$ oraz $h_2(z) = g_{21}(z) + g_{22}(z)$, wektor $\mathbf{H}(z)$ można przedstawić w postaci:

$$\mathbf{H}(z) = [\mathbf{I} - z\mathbf{P}(\mathbf{I} - \mathbf{A})]^{-1} \mathbf{P}\mathbf{A}\mathbf{e}z,$$

gdzie: $\mathbf{e} = [1, 1]^T$.

Z definicji $c_i(z_1, z_2)$ oraz równości: $p(s_{l-2} = 1) = b_1$ i $p(s_{l-2} = 2) = b_2$ wynika, że:

$$\begin{aligned} E\{\tau_{l-1}(z_1), \tau_l(z_2)\} &= p(s_{l-2} = 1)c_1(z_1, z_2) + p(s_{l-2} = 2)c_2(z_1, z_2) = \\ &= [p(s_{l-2} = 1), p(s_{l-2} = 2)] [c_1(z_1, z_2), c_2(z_1, z_2)]^T = [b_1, b_2] \times \mathbf{G}(z_1) \times \mathbf{H}(z_2) = \\ &= \left[\frac{(1-p_{22})\alpha_1}{(1-p_{22})\alpha_1 + (1-p_{11})\alpha_2}, \frac{(1-p_{11})\alpha_2}{(1-p_{22})\alpha_1 + (1-p_{11})\alpha_2} \right] \times \mathbf{G}(z_1) \times \mathbf{H}(z_2). \end{aligned}$$

Wartość oczekiwana $E\{\tau_{l-1}, \tau_l\}$ zmiennej losowej (τ_{l-1}, τ_l) jest wyznaczana jako wartość pierwszej pochodnej funkcji generującej $E\{\tau_{l-1}(z_1), \tau_l(z_2)\}$ dla $z_1 = 1$ i $z_2 = 1$ według następującej zależności:

$$\begin{aligned} E\{\tau_{l-1}, \tau_l\} &= [b_1, b_2] \times \frac{\partial \mathbf{G}(z_1)}{\partial z_1} \times \frac{\partial \mathbf{H}(z_2)}{\partial z_2} \Big|_{z_1=1, z_2=1} = \\ &= [b_1, b_2] \times [\mathbf{I} - \mathbf{P}(\mathbf{I} - \mathbf{A})]^{-2} \mathbf{P}\mathbf{A}[\mathbf{I} - \mathbf{P}(\mathbf{I} - \mathbf{A})]^{-2} \mathbf{P}\mathbf{A}\mathbf{e}. \end{aligned}$$

Wartość funkcji korelacji długości czasu międzyzgłoszeniowego, dla opóźnienia równego jeden, jest określona wyrażeniem:

$$\begin{aligned} \varphi_1 &= \frac{\text{cov}\{\tau_{l-1}, \tau_l\}}{\text{var}\{\tau_l\}} = \frac{1}{c^2\{\tau_l\}} \left[E\{\tau_{l-1}, \tau_l\} - E\{\tau_{l-1}\}E\{\tau_l\} \right] = \frac{1}{c^2\{\tau_l\}} \left[E\{\tau_{l-1}, \tau_l\} - E\{\tau_l\}^2 \right] = \\ &= \frac{1}{c^2(\tau)} \frac{\alpha_1\alpha_2(\alpha_1 - \alpha_2)^2(1-p_{11})(1-p_{22})(p_{11} + p_{22} - 1)^2}{(2-p_{11}-p_{22})^2 [\alpha_1(1-p_{22}) + \alpha_2(1-p_{11}) + \alpha_1\alpha_2(p_{11} + p_{22} - 1)]^2}. \end{aligned}$$

7.2.4.2.4. PRZYDZIELANIE ZNACZNIKÓW ZGODNIE Z ALGORYTMEM LB(c)

Jeżeli zagregowany strumień jednostek danych jest realizacją dwustanowego procesu MMBP, to stan systemu przydzielania znaczników zgodnie z algorytmem LB(c) jest definiowany za pomocą trzech zmiennych (b, z, i), gdzie:

b – liczba jednostek danych w pamięci buforowej jednostek danych ($0 \leq b \leq B_{JD}$),

z – liczba znaczników w pamięci buforowej znaczników ($0 \leq z \leq B_{ZN}$),

i – liczba stanów procesu MMBP ($i \in \{1, 2\}$).

Ze względu na przyjęte założenie, że suma pojemności B pamięci buforowych jednostek danych i znaczników ($B = B_{JD} + B_{ZN}$) jest stała, co oznacza, że:

➤ jeżeli kolejka znaczników jest pusta ($z = 0$), to długość kolejki jednostek danych jest większa od zera ($b > 0$),

➤ jeżeli kolejki jednostek danych jest pusta ($b = 0$), to długość kolejki znaczników jest większa od zera ($z > 0$),

stan systemu LB(c), oznaczony przez (b, z), może być przedstawiony jako zmienna x ; $x = B_{ZN} - z + b$ ($0 \leq x \leq B_{JD} + B_{ZN}$).

Przyjęcie takiej definicji zmiennej, chwilowej wartości stanu systemu oznacza, że można wyróżnić następujące, charakterystyczne przypadki dla różnych wartości zmiennej x :

➤ jeżeli $x = 0$, to $z = B_{ZN}$ i $b = 0$; w pamięci buforowej jednostek danych nie ma jednostek danych, a pamięć buforowa znaczników jest całkowicie zajęta przez znaczniki oczekujące na jednostki danych,

➤ jeżeli $x = B_{JD} + B_{ZN}$, to $z = 0$ i $b = B_{JD}$; pamięć buforowa jednostek danych jest całkowicie zajęta przez jednostki danych oczekujące na znaczniki, a w pamięci buforowej znaczników nie ma znaczników,

➤ jeżeli $x < B_{ZN}$, to $B_{ZN} - x$ znaczników w pamięci buforowej znaczników oczekuje na nadejście jednostek danych,

➤ jeżeli $x > B_{ZN}$, to $x - B_{ZN}$ jednostek danych w pamięci buforowej jednostek danych oczekuje na nadejście znacznika.

Wprowadzenie zmiennej x pozwala definiować stan systemu LB(c) za pomocą dwóch zmiennych x oraz i , gdzie x jest chwilową wartością stanu systemu, a i jest jednym z dwóch stanów procesu MMBP. Dla tak opisanego systemu LB(c) możliwe jest otrzymanie rozwiązania w postaci analitycznej jako rozwiązania zbioru równań liniowych:

$$\tilde{\alpha}Q = \tilde{\alpha} \quad \text{i} \quad \tilde{\alpha}e = 1,$$

gdzie:

$\tilde{\alpha}$ ($\tilde{\alpha} = [v_0^1, v_0^2, v_1^1, v_1^2, \dots, v_{B_{JD}+B_{ZN}}^1, v_{B_{JD}+B_{ZN}}^2]$) jest $2(B_{JD} + B_{ZN} + 1)$ -elementowym wektorem wartości prawdopodobieństw przebywania (w stanie ustalonym) systemu LB(c) w stanie x ($0 \leq x \leq B_{JD} + B_{ZN}$) dla dwóch różnych stanów i ($i \in \{1, 2\}$) procesu MMBP,

\mathbf{e} ($\mathbf{e} = [1, 1, \dots, 1]^T$) jest $2(B_{JD} + B_{ZN} + 1)$ -elementowym wektorem jednostkowym, \mathbf{Q} jest macierzą szybkości napływu jednostek danych.

Wartości wyrazów macierzy \mathbf{Q} zależą od liczby jednostek danych napływających w czasie trwania N szczelin czasowych i wartości prawdopodobieństwa przejścia procesu MMBP ze stanu i -tego do j -tego (w przypadku dwustanowego procesu MMBP $i, j \in \{1, 2\}$). Liczba N szczelin zależy od stałej szybkości napływu znaczników; znaczniki napływają z szybkością $1/N$, a długość jednej szczeliny jest czasem transferu jednej jednostki danych z szybkością równą wartości szczytowej.

W celu zdefiniowania wyrazów macierzy \mathbf{Q} definiowane są następujące prawdopodobieństwa:

- $a_{ij}(l, N)$ – prawdopodobieństwo zdarzenia polegającego na tym, że w przedziale o długości N szczelin napływie l jednostek danych i proces MMBP przejdzie ze stanu i -tego do stanu j -tego,
- ϕ_{ij} – prawdopodobieństwo zdarzenia polegającego na tym, że proces MMBP przejdzie ze stanu i -tego do stanu j -tego podczas trwania jednej szczeliny i że w tym czasie nie nadejdzie jednostka danych,
- ϕ_i – prawdopodobieństwo zdarzenia, że w czasie trwania jednej szczeliny nadejdzie jednostka danych pod warunkiem, że na początku szczeliny proces MMBP jest w i -tym stanie.

Jeżeli przyjmie się dalej, że proces MMBP jest procesem dwustanowym ($i, j \in \{1, 2\}$), zdefiniowanym przez:

► kwadratową macierz wartości prawdopodobieństw przejść pomiędzy stanami, tzn. $\mathbf{P} = [p_{ij}]_{2 \times 2}$, gdzie: $p_{11} = p_{11}$, $p_{12} = 1 - p_{11}$, $p_{21} = 1 - p_{22}$ oraz $p_{22} = p_{22}$,

► diagonalną macierz wartości prawdopodobieństw napływu jednostek danych w szczelinie w i -tym stanie, tzn. $\mathbf{A} = [\alpha_{ij}]_{2 \times 2}$, gdzie: $\alpha_{11} = \alpha_1$, $\alpha_{12} = 0$, $\alpha_{21} = 0$ oraz $\alpha_{22} = \alpha_2$,

macierz $\mathbf{\check{O}}$ ($\mathbf{\check{O}} = [\phi_{ij}]_{2 \times 2}$) oraz wektor $\mathbf{\check{A}}$ ($\mathbf{\check{A}} = [\phi_{ij}]_{1 \times 2}$) są następujące:

$$\begin{aligned} \mathbf{\check{O}} &= \mathbf{P}(\mathbf{I} - \mathbf{\check{A}}) = \begin{bmatrix} p_{11} & 1 - p_{11} \\ 1 - p_{22} & p_{22} \end{bmatrix} \times \begin{bmatrix} 1 - \alpha_1 & 0 \\ 0 & 1 - \alpha_2 \end{bmatrix} = \\ &= \begin{bmatrix} p_{11}(1 - \alpha_1) & (1 - p_{11})(1 - \alpha_2) \\ (1 - p_{22})(1 - \alpha_1) & p_{22}(1 - \alpha_2) \end{bmatrix}, \\ \mathbf{\check{A}} &= \mathbf{P}\mathbf{\check{a}} = \begin{bmatrix} p_{11} & 1 - p_{11} \\ 1 - p_{22} & p_{22} \end{bmatrix} \times \begin{bmatrix} \alpha_1 \\ \alpha_2 \end{bmatrix} = \begin{bmatrix} p_{11}\alpha_1 + (1 - p_{11})\alpha_2 \\ p_{22}\alpha_2 + (1 - p_{22})\alpha_1 \end{bmatrix}. \end{aligned}$$

Dla omawianego, dwustanowego procesu MMBP, wartości prawdopodobieństw $a_{ij}(0, 1)$ są wyrazami macierzy \mathbf{A}_0 , a ich wartości są równe wartościom wyrazów macierzy $\mathbf{\check{O}}$, tj.

$$A_0 = \begin{bmatrix} a_{11}(0,1) & a_{12}(0,1) \\ a_{21}(0,1) & a_{22}(0,1) \end{bmatrix} = \begin{bmatrix} \phi_{11} & \phi_{12} \\ \phi_{21} & \phi_{22} \end{bmatrix} = \mathbf{\phi},$$

wartości prawdopodobieństw $a_{ij}(1, 1)$, będące wyrazami macierzy A_1 , są natomiast równe wartościom wyrazów iloczynu macierzy P wartości prawdopodobieństw przejść między stanami procesu MMBP oraz macierzy A wartości prawdopodobieństw napływu jednostek danych w szczelinie wtedy, gdy proces MMBP znajduje się w danym, i -tym stanie, tzn. $A_1 = PA$:

$$A_1 = \begin{bmatrix} a_{11}(1,1) & a_{12}(1,1) \\ a_{21}(1,1) & a_{22}(1,1) \end{bmatrix} = \begin{bmatrix} p_{11} & 1-p_{11} \\ 1-p_{22} & p_{22} \end{bmatrix} \times \begin{bmatrix} \alpha_1 & 0 \\ 0 & \alpha_2 \end{bmatrix} = \begin{bmatrix} p_{11}\alpha_1 & (1-p_{11})\alpha_2 \\ (1-p_{22})\alpha_1 & p_{22}\alpha_2 \end{bmatrix}.$$

Po wprowadzeniu oznaczenia, zgodnie z którym wartości prawdopodobieństw $a_{ij}(l, N)$ są wyrazami kwadratowych macierzy A_l , tzn. $A_l = [a_{ij}(l, N)]$, macierze A_l są l -tymi elementami pierwszego wiersza macierzy R^N , gdzie macierz $R^{(N+1) \times (N+1)}$ ma następującą postać:

$$R = \begin{bmatrix} A_0 & A_1 & 0 & 0 & \dots & 0 & 0 \\ 0 & A_0 & A_1 & 0 & \dots & 0 & 0 \\ \cdot & \cdot & \cdot & \cdot & \dots & \cdot & \cdot \\ 0 & 0 & 0 & 0 & \dots & A_1 & A_0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}.$$

Dla tak zdefiniowanych macierzy – A_0, A_1, A_l oraz R – macierz szybkości napływu jednostek danych Q , której wyrazy są liczbą jednostek danych napływających w czasie trwania N szczelin czasowych, ma postać:

$$Q = \begin{bmatrix} A_0 & A_1 & A_2 & \dots & \dots & A_{B_{JD}+B_{ZN}}^* \\ A_0 & A_1 & A_2 & \dots & \dots & A_{B_{JD}+B_{ZN}}^* \\ 0 & A_0 & A_1 & \dots & \dots & A_{B_{JD}+B_{ZN}-1}^* \\ \cdot & \cdot & \cdot & \dots & \dots & \dots \\ 0 & 0 & 0 & \dots & A_0 & A_1^* \end{bmatrix}.$$

Wyrazami macierzy $A_l^* = [a_{ij}^*(l, N)]$ dla $l \in \{1, 2, \dots, B_{JD} + B_{ZN}\}$, tzn. $a_{ij}^*(l, N)$ są wartości prawdopodobieństw zdarzeń polegających na tym, że w przedziale o długości N szczelin napłynie co najmniej l jednostek danych, a proces MMBP przejdzie ze stanu i -tego do stanu j -tego.

Wartości prawdopodobieństw $a_{ij}^*(l, N)$ dla $i, j \in \{1, 2\}$ określa wyrażenie:

$$a_{ij}^*(l, N) = \begin{cases} 1 - \sum_{k=0}^{l-1} a_{ij}(k, N) - \sum_{k=0, j' \neq j}^{B_{JD}+B_{ZN}} a_{ij'}(k, N), & l \in \{0, 1, \dots, N\} \\ 0, & l > N \end{cases}.$$

Wektor wartości prawdopodobieństw systemu LB(c), dla dwustanowego procesu MMBP znajdującego się w stanie ustalonym:

$$\vec{\delta} = [\nu_0^1, \nu_0^2, \nu_1^1, \nu_1^2, \dots, \nu_{B_{JD}+B_{ZN}}^1, \nu_{B_{JD}+B_{ZN}}^2]$$

jest rozwiązaniem układu równań $\nu Q = \vec{\delta}$. Znajomość wartości wektora $\vec{\delta}$ jest równoważna znajomości wektora wartości prawdopodobieństw:

$$\vec{\delta} = [\bar{\nu}_0, \bar{\nu}_1, \bar{\nu}_2, \dots, \bar{\nu}_{B_{JD}+B_{ZN}}],$$

gdzie: $\bar{\nu}_k = \nu_k^1 + \nu_k^2$ dla $k \in \{0, 1, 2, \dots, Z_{B_{JD}+B_{ZN}}\}$.

Znajomość wartości wektora $\vec{\delta}$ umożliwia obliczenie wartości wskaźników jakości systemu przydzielania znaczników w systemie LB(c):

- prawdopodobieństwa utraty znaczników (ang. *token loss probability*) – p_{TLP} ,
- prawdopodobieństwa utraty jednostek danych (ang. *customer loss probability*) – p_{CLP} ,
- średniej szybkości wypływu jednostek danych (ang. *customer depart rate*) – $\bar{\nu}_{CDR}$,
- średniej liczby szczelin pomiędzy dwoma kolejnymi, wychodzącymi jednostkami danych (ang. *mean interdeparture time*) \bar{l}_{MIT} ,
- średniej liczby jednostek danych oczekujących w pamięci buforowej jednostek danych na znaczniki (ang. *mean number of waiting customers*) \bar{l}_W .

Wartość prawdopodobieństwa utraty znaczników p_{TLP} jest równa wartości prawdopodobieństwa zdarzenia polegającego na tym, że w chwili napływu znacznika pamięć buforowa znaczników jest całkowicie zajęta przez znaczniki oczekujące na jednostki danych, to znaczy wartości prawdopodobieństwa przebywania systemu w stanie $x = 0$, tzn.:

$$p_{TLP} = \bar{\nu}_0.$$

Każdy znacznik przyjęty do pamięci buforowej znaczników (z prawdopodobieństwem $1 - p_{TLP}$), podobnie jak każda jednostka danych przyjęta do pamięci buforowej jednostek danych (z prawdopodobieństwem $1 - p_{CLP}$) są wyprowadzane z systemu LB(c) z taką samą średnią szybkością. Wartość oczekiwana liczby jednostek danych \bar{l}_{IN} , napływających w czasie trwania N szczelin (gdzie N szczelin jest długością czasu międzyzgłoszeniowego dla znaczników) jest równa iloczynowi liczby szczelin N i wartości oczekiwanej liczby jednostek danych w jednej szczelinie ρ , tzn. $\bar{l}_{IN} = \rho N$. Wartość oczekiwanej liczby jednostek danych ρ , dla dwustanowego procesu MMBP, zdefiniowanego wartościami macierzy \mathbf{P} i \mathbf{A} , jest wartością wyrażenia:

$$\rho = \pi_1 \alpha_1 + \pi_2 \alpha_2 = \frac{1 - p_{22}}{2 - p_{11} - p_{22}} \alpha_1 + \frac{1 - p_{11}}{2 - p_{11} - p_{22}} \alpha_2,$$

gdzie wektor $\delta = [\pi_1, \pi_2]$ jest wektorem wartości prawdopodobieństw przebywania procesu MMBP w jednym z dwóch stanów w stanie ustalonym; jest rozwiązaniem równania $\pi P = \delta$. Wartości prawdopodobieństw π_1 oraz π_2 są sumami odpowiednich wartości prawdopodobieństw wektora δ , tzn.:

$$\pi_1 = v_0^1 + v_1^1 + v_2^1 + \dots + v_{B_{JD} + B_{ZN}}^1 \quad \text{oraz} \quad \pi_2 = v_0^2 + v_1^2 + v_2^2 + \dots + v_{B_{JD} + B_{ZN}}^2.$$

Z warunków zachowania przepływu wynika, że wartości średnich liczb napływających jednostek danych i napływających znaczników w czasie trwania N szczełin są równe, tzn.

$$(1 - p_{CLP})\rho N = 1 - p_{TLP},$$

co oznacza, że wartość prawdopodobieństwa odrzucania jednostek danych p_{CLP} w systemie LB(c) jest wartością wyrażenia:

$$p_{CLP} = 1 - \frac{1 - p_{TLP}}{\rho N}.$$

Wartość średniej szybkości wyprowadzania jednostek danych z systemu LB(c) jest równa wartości średniej liczby jednostek danych przyjmowanych do systemu, tzn.:

$$\bar{v}_{CDR} = \rho(1 - p_{CLR}).$$

Wartość średniej liczby szczełin \bar{l}_{MIT} pomiędzy dwiema kolejno wyprowadzany mi jednostkami danych jest równa odwrotności średniej szybkości wyprowadzania zgłoszeń:

$$\bar{l}_{MIT} = [\rho(1 - p_{CLR})]^{-1}.$$

Wartość średniej liczby jednostek danych oczekujących w pamięci buforowej jednostek danych na znaczniki jest sumą iloczynów wartości liczb jednostek danych oczekujących na znaczniki $x - B_{ZN}$ oraz prawdopodobieństwa przebywania systemu LB(c) w stanie x ($B_T < x \leq B_{JD} + B_{ZN}$) i określa ją równanie:

$$\bar{l}_W = \sum_{x=B_{ZN}+1}^{B_{JD}+B_{ZN}} (x - B_{ZN})\bar{v}_x = \sum_{x=B_{ZN}+1}^{B_{JD}+B_{ZN}} (x - B_{ZN})(v_x^1 + v_x^2).$$

7.2.4.3. REGLAMENTACJA ZAPOTRZEBOWANIA NA ZASOBY SYSTEMU

Przykładem rozwiązań, wykorzystujących w zadaniach utrzymania ruchu zasady reglamentowania zapotrzebowania na zasoby, są różne rozwiązania z zastosowaniem mechanizmu okna (ang. *window-based technique*).

W ogólnym przypadku cechą technik korzystających z mechanizmu okna jest to, że dla każdego źródła strumienia jednostek danych, objętego tym mechanizmem sterowania, istnieje limit liczby jednostek danych, które w danej chwili mogą znajdować

się w systemie obsługi. Przekroczenie limitu powoduje wstrzymanie generowania przez źródło (przyjmowania przez system obsługi) nowych jednostek danych.

Miarą skuteczności kontroli zgodności wartości parametrów rzeczywistego ruchu z kontraktem dla technik wykorzystujących mechanizm okna jest prawdopodobieństwo naruszenia kontraktu (ang. *violation probability*), tzn. prawdopodobieństwo zdarzenia, że jednostka danych generowana z zachowaniem warunków kontraktu zostanie odrzucona w wyniku działania mechanizmu kontroli. Przyjmuje się, że mechanizm sterowania nie narusza warunków kontraktu, gdy wartość omawianego prawdopodobieństwa jest mniejsza od 10^{-9} .

Różne techniki sterowania przepływem, należące do omawianej grupy rozwiązań, różnią definicje okna, sposoby generowania okna oraz sposoby wypełniania okna.

7.2.4.3.1. TECHNIKA JW

Istotą techniki JW (ang. *Jumping Window*) jest istnienie górnego ograniczenia dużej liczby (limitu) jednostek danych, jakie mogą być wygenerowane przez źródło (przyjęte ze źródła do systemu obsługi) w powtarzających się przedziałach czasu o ustalonej długości (definiowanych jako rozmiar okna). Oznacza to, że po wyczerpaniu limitu źródło nie generuje nowych jednostek danych (albo każda nowa jednostka danych jest odrzucona). W przypadku granicznym, tzn. gdy limit wynosi jeden, a długość przedziału czasu jest równa czasowi generacji znacznika, technika JW jest równoważna algorytmowi LB.

Jakość omawianej techniki kontroli zależy od dopasowania długości przedziału czasu (rozmiaru okna) dla danego źródła. Zwiększanie rozmiaru okna zmniejsza możliwości detekcji niezgodności wartości parametrów rzeczywistego ruchu z wartościami wynegocjowanymi. Z kolei zmniejszanie rozmiaru okna może prowadzić do ingerowania w proces generowania jednostek danych przez źródła utrzymujące wartości deskryptorów ruchu w granicach zawartego kontraktu.

Technika JW nie może być stosowana jednocześnie do kontroli napływu z maksymalną (szczytową) i średnią szybkością. W niektórych implementacjach techniki JW dopuszczalne jest generowanie (przyjmowanie) jednostek danych powyżej ustalonego limitu, które mają niższy priorytet obsługi w systemie.

7.2.4.3.2. TECHNIKA TJW

W przeciwieństwie do techniki JW, gdzie początek okna jest niezależny od źródła (nie jest zsynchronizowany z aktywnością źródła), w technice TJW (ang. *Triggered Jumping Window*) początek aktywności źródła pełni rolę mechanizmu spustowego (ang. *triggered*), wyzwalającego proces generacji okna. Oznacza to, że w technice TJW kolejne okna nie muszą następować bezpośrednio po sobie. Okno TJW pojawia się tylko wtedy, gdy generowana jest pierwsza jednostka danych w kolejnym przedziale aktywności źródła.

Przyjęcie mechanizmu wyzwalania okna powoduje, że mniejsze, w porównaniu z techniką JW, jest prawdopodobieństwo zdarzenia polegającego na ingerowaniu w proces generowania jednostek danych przez źródło zachowujące ograniczenia kontraktu.

7.2.4.3.3. TECHNIKA MW

W technice tej, podobnie jak w technice JW, maksymalna liczba jednostek danych dopuszczalnych w ustalonym przedziale czasu jest stała. Podstawowa różnica pomiędzy technikami JW i MW (ang. *Moving Window*) polega na tym, że w technice MW każda jednostka danych jest lokowana w tym samym jednym oknie. Dla techniki MW charakterystyczne jest ciągle, zgodne z upływającym czasem, przesuwanie okna o stałej długości.

Do analizy techniki MW najczęściej stosowany jest model systemu wieloserwerowego (liczba serwerów jest równa długości okna mierzonego liczbą jednostek danych) bez pamięci buforowej (bez możliwości oczekiwania na obsługę).

Wartość prawdopodobieństwa naruszenia kontraktu jest – w tym przypadku – wartością prawdopodobieństwa zdarzenia, że jednostka danych generowana w ramach kontraktu zostanie odrzucona, ponieważ w oknie brak jest wolnego miejsca.

7.2.4.3.4. TECHNIKA EWMA

Idea techniki EWMA (ang. *Exponentialy Weighted Moving Average*) jest zbliżona do idei techniki JW, z tą różnicą, że w technice MW dopuszczalna jest zmienna długość (rozmiar) okna.

Cechą charakterystyczną omawianej techniki jest adaptacyjna procedura wyznaczania rozmiaru okna. Rozmiar kolejnego okna wyznacza się na podstawie znajomości poprzednio wyznaczonych rozmiarów okien oraz stopnia ich dopasowania do charakteru generowanego ruchu (miarą dopasowania jest wartość prawdopodobieństwa naruszenia kontraktu). Początkowa wartość rozmiaru okna jest parametrem procedury.

Celem adaptacyjnej procedury jest uzyskanie elastyczności (mierzonej wartością tak zwanego współczynnika elastyczności (ang. *flexibility factor*)) w dopasowywaniu rozmiaru okna do charakterystyk źródła. Zerowa wartość współczynnika elastyczności oznacza stały rozmiar okna i wtedy technika MW jest tożsama z techniką JW.

Technika EWMA jest najbardziej efektywną techniką kontroli spośród wszystkich wykorzystujących mechanizm okna. Jej efektywność jest porównywalna z efektywnością algorytmu LB.

Trzy spośród omawianych technik kontroli wykorzystujących mechanizm okna, tzn. JW, TJW oraz MW, są przykładem realizacji przewencyjnego sterowania przepływem. Stały rozmiar okna powoduje, że brak jest możliwości eliminowania chwilowych wzrostów liczby odrzucanych jednostek danych, powodowanych chwilowym wzrostem liczby źródeł jednocześnie generujących jednostki danych. Technika EMWA ma cechy zarówno sterowania przewencyjnego, jak i reaktywnego. Mechanizm adaptacji rozmiaru okna do zmieniających się wartości deskryptorów ruchu pozwala na uwzględnianie na bieżąco statystycznych właściwości generowanych strumieni jednostek danych.

7.2.5. ODRZUCANIE SELEKTYWNE

W niektórych implementowanych w sieciach schematach kontroli zgodności napływu jednostek danych generowanych przez źródła z wartościami parametrów wyneogojowanych w kontrakcie ruchowym istnieje możliwość przyjmowania do sieci (jako systemu obsługi) jednostek danych niespełniających warunków kontraktu. Zadaniem mechanizmów kontroli zgodności jest zabezpieczenie sieci przed powstawaniem przeciążeń oraz ruchu generowanego przez użytkowników dotrzymujących warunków kontraktu przed degradacją jakości obsługi. Dopuszczenie jednostek danych generowanych poza warunkami kontraktu ma na celu z jednej strony złagodzenie restrykcyjnego charakteru kontraktu opartego na niepełnej znajomości i przewidywalności generowanych strumieni jednostek danych, a z drugiej – podwyższenie stopnia wykorzystania zasobów sieci.

Współistnienie w sieci jednostek danych w ramach i spoza kontraktów w naturalny sposób wiąże się z ich znakowaniem (ang. *marking*) oraz kategoryzacją. W sytuacjach zagrażających sieci oraz jakości usług dla ruchu generowanego w ramach kontraktu jednostek danych spoza kontraktów mogą być i będą odrzucane (ang. *selective discarding*).

Zasada priorytetowania, ze względu na jej uniwersalność i elastyczność, może być przeniesiona (rozszerzona) na etap kontraktowania obsługi strumieni jednostek danych. Ograniczoność zasobów sieci może uzasadniać potrzebę kategoryzacji strumieni jednostek danych przez użytkowników sieci. Zasadne jest nadawanie wyższego priorytetu strumieniom jednostek danych związanych z aplikacjami wrażliwymi na ich utratę (np. głos). Zakres hierarchizacji w sieci zależy od możliwości odnotowania różnych poziomów priorytetu w nagłówkach obsługiwanych jednostek danych (np. w komórkach ATM konstrukcja nagłówka komórki pozwala na wprowadzenie i obsługę dwóch poziomów priorytetu).

Obsługa priorytetów oznacza m.in. istnienie w węzłach (dostępowych i pośrednich) sieci mechanizmów rozpoznawania priorytetu obsługiwanych jednostek danych, a w razie zaistnienia takiej potrzeby – odrzucania jednostek danych o niższych priorytetach.

7.2.5.1. MECHANIZM WYPYCHANIA

Działanie mechanizmu wypychania (ang. *push-out*) charakteryzują następujące działania:

1. Jednostki danych o różnych priorytetach są przyjmowane do sieci dopóty, dopóki w pamięci buforowej węzła istnieje wolne miejsce.

2. Jeżeli w chwili napływu jednostki danych brak jest wolnej pamięci buforowej, jednostka danych jest:

- odrzucana, gdy w pamięci buforowej znajdują się tylko jednostki danych o tym samym lub wyższym priorytecie,

➤ przyjmowana, gdy w pamięci buforowej znajdują się – nie będące w trakcie obsługi – jednostki danych o niższym priorytecie; przyjęcie tej jednostki danych odbywa się kosztem usuwanej jednostki danych o najniższym (spośród znajdujących się w pamięci buforowej) priorytecie.

Podstawowym ograniczeniem tego mechanizmu jest złożoność jego implementacji. Zasada wypychania jednostek danych o niższych priorytetach przez jednostki danych o wyższych priorytetach może – w ogólnym przypadku – zaburzać kolejność jednostek danych (kolejność w pamięci buforowej może być inna od kolejności generowania jednostek danych). Oznacza to dalej, że strategia obsługi pamięci buforowej nie może być prosto zarządzalną strategią typu FCFS (ang. *First Come First Served*). Złożoność niezbędnego systemu zarządzania pamięcią buforową, którego zadaniem jest nadzorowanie i odtwarzanie na wyjściu z pamięci buforowej kolejności jednostek danych zgodnej z kolejnością generowania, wprowadza duży narzut organizacyjny i skutkuje dodatkowym opóźnieniem.

7.2.5.2. MECHANIZM PROGOWY

W mechanizmie progowym (ang. *threshold*) do sterowania liczbą jednostek danych, o różnych priorytetach, w pamięci buforowej węzła stosuje się wartości progowe.

W przypadku systemu obsługującego jednostki danych o dwóch priorytetach działanie mechanizmu jest następujące:

1. Wszystkie jednostki danych są przyjmowane dopóty, dopóki liczba przyjętych jednostek danych nie przekroczy wartości progowej niższej od całkowitej pojemności pamięci buforowej.

2. Jeżeli liczba przyjętych jednostek danych przekroczy wartość progową, to:

➤ nowo nadchodzące jednostki danych o niższym priorytecie są odrzucane dopóty, dopóki liczba jednostek danych w pamięci buforowej nie będzie niższa od wartości progowej,

➤ nowo nadchodzące jednostki danych o wyższym priorytecie są przyjmowane dopóty, dopóki nie zostanie wyczerpana pojemność pamięci buforowej.

Efektywność omawianego mechanizmu jest determinowana zarówno sposobem wyznaczania wartości progowej, jak i wartością progową, w szczególności ilorazem wartości progowej do wartości całkowitej pojemności pamięci buforowej.

Wartość progowa może być wyznaczana na podstawie wartości średnich charakterystyk obsługiwanych źródeł jednostek danych i udziału w generowanych strumieniach jednostek danych o różnych priorytetach lub zmieniać się w zależności od zmiany charakterystyk obsługiwanego ruchu. Wybór pomiędzy różnymi sposobami wyznaczania wartości progowej jest wyborem pomiędzy złożonością procedur wyznaczania i efektywnością mechanizmu.

Przyjęcie małej (względem wartości pojemności całkowitej) wartości progowej może powodować nadmierne ograniczenie napływu jednostek danych o niższym prio-

rytecie i prowadzić do niewykorzystania zasobów sieci. Z kolei przyjęcie dużej wartości progowej może być przyczyną odrzucania jednostek danych o wyższym priorytecie i degradacji jakości obsługi.

Mechanizm progowy może być zaliczany zarówno do prewencyjnych, jak i reaktywnych metod sterowania przepływem. Kwalifikacja mechanizmu do jednej z wymienionych klas metod sterowania zależy od tego, czy i jak zmiany deskryptorów rzeczywistego ruchu wpływają na częstotliwość zmian i wartość progową.

7.3. PODSUMOWANIE

Jakość usług sieci teleinformatycznych ściśle zależy od efektywności struktur i mechanizmów zarządzania przeciążeniami i sterowania przepływem. O złożoności omawianych zagadnień decydują m.in.:

- duża rozpiętość wartości parametrów opisujących wymagania użytkowników co do jakości usług,
- słaba znajomość charakterystyk strumieni jednostek danych niektórych typów usług,
- znaczące różnice w szybkości nadawania różnych źródeł strumieni jednostek danych,
- zmienność w czasie charakterystyk strumieni jednostek danych dla tych samych aplikacji,
- możliwość generowania różnych klas strumieni jednostek danych przez jedno źródło,
- potrzeba mierzenia nie tylko prawdopodobieństwa odrzucania wywołań (ang. *call blocking*) i straty jednostek danych, ale także wariacji opóźnienia (ang. *delay variation*), maksymalnego opóźnienia (ang. *maximum delay*) i asymetrii (ang. *skewness*),
- niewielka przewidywalność wartości parametrów opisujących jakość działania sieci dla statystycznie zwielokrotnianych (multipleksowanych) strumieni o stałej i zmiennej szybkości nadawania,
- wrażliwość jakości usług czasu rzeczywistego na degradację jakości działania sieci,
- duża bezwładność procedur zarządzania i sterowania (rosnąca wartość ilorazu czasu trwania zgłoszenia i czasu transmisji jednostki danych powoduje, że w dowolnej chwili w sieci znajduje się względnie dużo jednostek danych, co prowadzi do „odłożenia w czasie” efektów działania zastosowanego mechanizmu),
- ograniczona możliwość stosowania przetwarzania w locie (ang. *on-the-fly processing*) w węzłach pośrednich sieci ze względu na duże szybkości transmisji.

Wymienione wyżej charakterystyczne cechy sieci powodują, że implementowane w tych sieciach mechanizmy sterowania przepływem i przeciwdziałania przeciążeniom powinny spełniać następujące wymagania:

- ze względu na szybkości transmisji algorytmy sterowania powinny być możliwie jak najprostsze, tak aby mogły być realizowane układowo,
- wartości parametrów jakości usług w fazie negocjowania połączenia powinny być wprowadzane konsekwentnie dla każdego z połączeń, niezależnie od charakterystyk ruchu obsługiwanych przez inne połączenia w sieci oraz zachowań źródeł strumieni jednostek danych,
- ze względu na częsty brak możliwości określenia dokładnych charakterystyk źródeł strumieni jednostek danych, wprowadzane techniki sterowania z jednej strony powinny być oparte na uproszczonych (intuicyjne) charakterystykach źródeł, a z drugiej – powinny być odporne na konsekwencje korzystania z niedokładnych specyfikacji źródeł,
- ze względu na możliwość pominięcia istotnych cech obsługiwanego strumienia jednostek danych nie jest dopuszczalne stosowanie mechanizmów sterowania wykorzystujących sztucznie wprowadzane ograniczenia na charakterystyki źródeł (np. redukcja szczytowych wartości intensywności strumienia przez spowalnianie napływu w czasie obsługi aplikacji wrażliwych na opóźnienia),
- mechanizmy sterowania (zwłaszcza prewencyjnego) nie powinny prowadzić do sytuacji, w których zasoby sieci nie są w pełni wykorzystane (ang. *underutilization*),
- ze względu na nieznaną przyszłych zastosowań implementowane mechanizmy powinny być otwarte i skalowalne na potrzeby organizacji nowych usług i zapewniania żądanej ich jakości.

LITERATURA

- [1] ATHURALIYA S., LOW S.H., LI V.H., YIN Q., *REM: Active Queue Management*, IEEE Network, May/June 2001, s. 48–53.
- [2] BARAKAT Ch., *TCP/IP modelling and validation*, IEEE Network, May/June 2001, s. 38–47.
- [3] BĄK A., *Reakcyjne metody sterowania ruchem w sieci ATM*, Warszawa 1997 (praca doktorska).
- [4] BENSOU B., TSANG D.H.K., CHAN K.T., *Credit-based Fair Queueing (CBFQ): A simple service-scheduling algorithm for packet-switched networks*, IEEE/ACM Transactions on Networking, Vol. 9, No. 5, October 2001, s. 591–604.
- [5] BERTSEKAS D., GALLAGER R., *Data networks*, Prentice-Hall, Englewood Cliffs 1987.
- [6] BHARATH-KUMAR K., JAFFE J.M., *Routing to multiple destinations in computer networks*, IEEE Transactions on Communications, Vol. COM-31, No. 3, March 1983, s. 343–351.
- [7] BU T., TOWSLEY D., *Fixed Point approximation for TCP behaviour in an AQM network* (www.itpapers.com).
- [8] CHAPMAN A., KUNG H.T., *Enhancing transport networks with Internet protocols*, IEEE Communications Magazine, May 1998, s. 100–104.

- [9] CHENG R.G., CHANG C.J., *Design of a fuzzy traffic controller for ATM networks*, IEEE/ACM Transactions on Networking, Vol. 4, No. 3, June 1996, s. 160–168.
- [10] CHEUNG S.Y., PENCEA S., *Pipelined sections: a new buffer management discipline for scalable QoS provision*, Emory University (www.itpapers.com).
- [11] CZACHÓRSKI T., *Modele kolejkowe w ocenie efektywności sieci i systemów komputerowych*, Pracownia Komputerowa Jacka Skalmierskiego, Gliwice 1999.
- [12] DOVROLIS C., RAMANATHAN P., *A case for Relative Differentiated Services and the Proportional Differentiation Model*, IEEE Network, September/October 1999, s. 26–34.
- [13] DEO N., *Teoria grafów i jej zastosowania w technice i informatyce*, PWN, Warszawa 1980.
- [14] DOLAN A., ALDOUS J., *Networks and algorithms*, Wiley, Chichester 1993.
- [15] EL-SAGHIR Z., GRZECH A., *A high performance ATM switch design and analysis*, Proceedings IEEE International Conference on Telecommunications, Vol. 1, Bucharest 2001, s. 513–518.
- [16] FILIPIAK J., *Structured systems analysis methodology for design of an ATM network architecture*, IEEE Transactions on Selected Areas of Communications, Vol. SAC-7, No. 8, October 1989, s. 1263–1273.
- [17] FINDEISEN W., SZYMANOWSKI J., WIERZBICKI A., *Teoria i metody obliczeniowe optymalizacji*, PWN, Warszawa 1977.
- [18] FISZ M., *Rachunek prawdopodobieństwa i statystyka matematyczna*, PWN, Warszawa 1967.
- [19] FLOYD S., JACOBSON V., *Random Early Detection gateways for congestion avoidance*, IEEE/ACM Transaction on Networking, Vol. 1, No. 8, August 1993.
- [20] FLOYD S., FALL K., *Promoting the use of end-to-end congestion control in the Internet*, IEEE/ACM Transactions on Networking, Vol. 7, No. 4, s. 459–472, August 1999.
- [21] FORD L.R., FULKERSON D.R., *Przepływy w sieciach*, PWN, Warszawa 1969.
- [22] GERLA M., KLEINROCK L., *On the topological design of distributed computer networks*, IEEE Transactions on Communications, Vol. COM-25, 1977, s. 48–60.
- [23] GEVROS P., CROWCROFT J., KIRSTEIN P., BHATTI S., *Congestion control mechanisms and the best effort service model*, IEEE Network, May/June 2001, s. 16–26.
- [24] GIACOMMAZZI P., MUSUMECI L., *Transport of IP controlled-load service over ATM network*, IEEE Network, January/February 1999, s. 36–47.
- [25] GIBBENS R., KEY P., *Distributed control and resource marking using best-effort routers*, IEEE Network, May/June 2001, s. 54–59.
- [26] GIBBENS R., KELLY F.P., *Resource pricing and the evolution of congestion control*, Automatica, No. 35, 1999 (www.statslab.com.ac.uk), s. 1969–1985.
- [27] GOVINDAN R., ALAETTINOGLU C., EDDY G., KESSENS D., KUMAR S., LEE W.S., *An architecture for stable, analyzable Internet routing*, IEEE Network, January/February 2001, s. 29–35.
- [28] GUERIN R., AHMADI H., NAGHSHINEH M., *Equivalent Capacity and its application to bandwidth allocation in High Speed Networks*, IEEE Journal on Selected Areas in Communications, Vol. 9, No. 7, September 1991, s. 968–981.
- [29] HABIB I. (red.), *Bandwidth allocation in ATM networks*, IEEE Communications Magazine, Vol. 35, No. 5, May 1997, s. 120–162.

- [30] HOLLOT C.V., MISRA V., TOWSLEY D., GONG W.B., *On designing improved controllers for AQM routers supporting TCP flows*, IEEE INFOCOM'01 (www.itpapers.com).
- [31] HOLLOT C.V., MISRA V., TOWSLEY D., GONG W.B., *A control theoretic analysis of RED*, IEEE INFOCOM '01 (www.itpapers.com).
- [32] IEEE Journal on Communications – special issue on congestion control in computer networks, Vol. COM-29, No. 4, April 1981.
- [33] JACOBSON V., *Congestion analysis and control*, Proceedings of SIGCOMM'88, August 1988, s. 314–329.
- [34] KARANDIKAR S., KALYANARAMAN S., BAGAL P., POCKER B., *TCP rate control* (www.itpapers.com).
- [35] KELLY F.P., KEY P.B., ZACHARY S., *Distributed admission control*, IEEE Journal of Selected Areas in Communications, Vol. 35, No. 12, December 2000, s. 2617–2628.
- [36] KERMANI P., KLEINROCK L., *Dynamic flow control in store-and-forward computer networks*, IEEE Transactions on Communications, Vol. COM-28, No. 2, February 1980, s. 263–270.
- [37] KLEINROCK L., *Communication nets*, Volume I: *Stochastic message flow and delay*, Volume II: *Computer Applications*, Wiley, New York 1976.
- [38] LAW A.M., McCOMAS G., *Simulation software for communications networks: The state of the art*, IEEE Communications Magazine, March 1994, s. 44–50.
- [39] LAW S.H., PETERSON L., WANG L., *Optimization flow control, I: Basic algorithm and convergence*, IEEE/ACM Transactions on Networking, Vol. 7, No. 6, December 1999, s. 861–874.
- [40] MELSA L.J., SAGE A.P., *An introduction to probability and stochastic processes*, Prentice-Hall, Englewood Cliffs 1973.
- [41] MISHRA P.P., SARAN H., *Capacity management and routing policies for Voice over IP traffic*, IEEE Network, March/April 2001, s. 20–27.
- [42] MISHRA S., WU L., *An evaluation of flow control in group communication*, IEEE/ACM Transactions on Networking, Vol. 6, No. 5, October 1998, s. 571–587.
- [43] MISHRA V., GONG W.B., TOWSLEY D., *Fluid-based analysis of a network of AQM routers supporting TCP flows with an application to RED*, SIGCOMM'00 (www.itpapers.com).
- [44] ONVURAL R.O., *Asynchronous Transfer Mode; Performance issues*, Artech House, Boston 1995.
- [45] PAPIR Z. (red.), *Sieci z komutacją pakietów – od X.25 do Frame Relay i ATM*, Wydawnictwo Fundacji Postępu Telekomunikacji, Kraków 1997.
- [46] PAPIR Z., *Ruch telekomunikacyjny i przeciążenia sieci pakietowych*, WKŁ, Warszawa 2001.
- [47] PAXSON V., *End-to-end Internet packet dynamics*, IEEE/ACM Transactions on Networking, Vol. 7, No. 3, June 1999, s. 277–292.
- [48] PENNOTTI M.C., SCHWARTZ M., *Congestion control in store-and-forward tandem links*, IEEE Transactions on Communications, Vol. COM-23, No. 12, s. 1434–1443, December 1975.
- [49] PERROS H.G., ELSAYED K.M., *Call Admission Control schemes: a review*, North Carolina State University (ftp://eceyv.ncsu.edu).

-
- [50] SAITO H., *Call Admission Control in an ATM network using upper bound of Cell Loss Probability*, IEEE Transactions on Communications, Vol. 40, No. 9, September 1992, s. 1512–1521.
 - [51] STALLINGS W., *High-speed networks: TCP-IP and ATM design principle*, Prentice-Hall, New York 1998.
 - [52] TAKAGI H. (red.), *Stochastic analysis of computer and communication systems*, North-Holland, Amsterdam 1990.
 - [53] WIDMER J., DENDA R., MAUVE M., *A survey of TCO-friendly congestion control*, IEEE Network, May/June 2001, s. 28–37.
 - [54] WOLFINGER B.E., *Characterization of mixed traffic load in service integrated networks*, Proceedings 6th Polish Teletraffic Symposium, Wrocław 1999, s. 11–28.

8. KONCEPCJE RÓŻNICOWANIA JAKOŚCI USŁUG SIECI TELEINFORMATYCZNYCH

U powszechnienie sieci teleinformatycznych i różnicowanie wymagań aplikacji, obsługiwanych przez te sieci, wymusza potrzebę dostarczania rozmaitych usług sieciowych o różnej, zadanej jakości. Zdolność sieci do różnicowania jakości usług sieciowych i gwarantowania jakości różnych dostarczanych usług sieciowych przez dostarczanie w tym celu niezbędnych zasobów wyznacza zakres problematyki jakości usług QoS (ang. *Quality of Service*) w sieciach.

Realizacja sieci spełniającej warunki sieci wielousługowej (ang. *multiservice network*) oznacza implementację mechanizmów wspierających zarówno różnicowanie usług, jak i możliwość kierowania określonych zasobów sieciowych do poszczególnych klas użytkowników, użytkowników lub strumieni zgłoszeń (jednostek danych).

Aplikacje wspierane przez sieci teleinformatyczne charakteryzują różne wymagania i różną wrażliwość aplikacji na wartości wybranych wskaźników jakości usług dostarczanych przez sieci. Przekroczenie określonych wartości wskaźników jakości usług dla niektórych aplikacji oznacza praktycznie bezużyteczność usług sieciowych. Różnicowanie jakości usług sieciowych jest zagadnieniem istotnym zarówno dla odbiorców, jak i dostawców tych usług dla wybranych grup użytkowników.

Wymagania aplikacji i użytkowników dotyczące jakości usług sieci teleinformatycznych są na tyle różne, że opracowanie i wdrożenie jednego modelu świadczenia usług sieciowych jest niemożliwe i zbędne. Jednocześnie konieczne są takie zmiany architektur sieci pakietowych, które pozwalają na implementację w sieciach mechanizmów różnicowania jakości usług w sposób umożliwiający różnym grupom aplikacji i użytkowników sieci dostęp do usług o wyższej jakości za większą cenę.

Konieczność i możliwość dostarczania usług o różnej jakości w tej samej sieci teleinformatycznej wymaga odejścia od koncepcji sterowania rozdziałem zasobów sieciowych na krańcach sieci. Koncepcje takie, realizowane w sieciach pakietowych (ang. *packet-switched*), zapewniają obsługę użytkowników sieci i generowanego przez nich ruchu bez gwarancji jakości (ang. *best-effort*). Oznacza to praktycznie, że wszyscy użytkownicy sieci otrzymują podobną jakość usług sieciowych. Wynika to z jed-

nakowego traktowania w sieci wszystkich jednostek danych (pakietów), niezależnie od ich przynależności do różnych przepływów i klas ruchu. Rozwiązanie takie jest nie do przyjęcia w rozwiązaniach, w których sieć jest wykorzystywana do jednoczesnej obsługi ruchu elastycznego i strumieniowego, generowanych przez aplikacje o skrajnie różnych wymaganiach jakościowych i ilościowych. Różnicowanie jakości usług w sieciach teleinformatycznych dotyczy dwóch grup zagadnień:

- architektury alokacji zasobów sieci i modeli usług,
- optymalizacji wykorzystania udostępnianych zasobów sieci.

8.1. KONCEPCJE ARCHITEKTUR JAKOŚCI USŁUG W SYSTEMACH TELEINFORMATYCZNYCH

Sieć teleinformatyczna, najprościej rzecz ujmując, jest zbiorem współdzielonych zasobów przeznaczonych do jednoczesnej obsługi ruchu elastycznego i strumieniowego generowanego przez aplikacje (użytkowników) współzawodniczące o zasoby sieci.

Realizacja zadania polegającego na różnicowaniu usług sieci teleinformatycznej polega na implementacji w sieci mechanizmów zdolnych różnicować:

- zasady dostępu do zasobów (ang. *admission control*),
- zasady przyznawania (alokacji) zasobów (ang. *resource allocation*),
- ilości przyznawanych zasobów poszczególnym użytkownikom.

Przykładami różnych koncepcji (architektur) alokacji zasobów są:

1. Koncepcja usług dostarczanych bez gwarancji jakości (ang. *best effort*), zgodnie z którą alokacja zasobów jest wynikiem współzawodnictwa przepływów.

2. Koncepcja usług zintegrowanych intServ (ang. *Integrated Services*), zgodnie z którą alokacja zasobów odbywa się przez ich rezerwację dla poszczególnych przepływów generowanych przez aplikacje.

3. Koncepcja usług zróżnicowanych diffServ (ang. *Differentiated Services*), zgodnie z którą alokacja zasobów odbywa się przez ich udostępnianie zróżnicowanym klasom ruchu. Efektywność alokacji zasobów uzyskuje się przez zarządzanie ilością ruchu wprowadzanego do sieci, albo ruchu jako całości, albo w obrębie wyróżnionych klas ruchu.

Wyróżnione trzy koncepcje dostarczania jakości usług nie są jednoznaczne i wykluczające się.

Koncepcja obsługi bez gwarancji jakości wynika wprost z filozofii sieci z komutacją wiadomości i opiera się na założeniu, że jednakową (w sensie średnim) jakość usług można uzyskać przez reglamentację ilości ruchu (w zasadzie elastycznego) wprowadzanego do sieci. Z tego punktu widzenia koncepcja ta ma wiele wspólnego z usługą sterowanego obciążenia w koncepcji usług zintegrowanych i koncepcją usług zróżnicowanych.

Koncepcje usług zintegrowanych i zróżnicowanych różni od koncepcji obsługi ruchu teleinformatycznego bez gwarancji jakości założenie o konieczności i możliwości różnicowania jakości usług sieci. Koncepcje usług zintegrowanych i zróżnicowanych różni z kolei filozofia dostarczania jakości usług. Koncepcja usług zintegrowanych

opiera się na założeniu, że efektywnym sposobem dostarczania jakości usług jest sterowanie poszczególnymi przepływami. Konceptcja usług zróżnicowanych zakłada natomiast, że skutecznym sposobem dostarczania jakości usług jest różnicowanie ich jakości dla wyróżnianych klas ruchu, tzn. dla przepływów zagregowanych.

Różne odmiany konceptcji i architektur jakości usług dostarczanych w sieciach teleinformatycznych, podobnie jak wiele innych konceptcji implementowanych w sieciach, są zorientowane na różne architektury sieci, różniące się przydatnością do obsługi różnych klas ruchu i efektywnością wykorzystania zasobów.

8.1.1. KONCEPCJA USŁUG BEZ GWARANCJI JAKOŚCI

Podstawą konceptcji usług sieciowych dostarczanych bez gwarancji jakości jest założenie, że reglamentacja ilości ruchu wprowadzanego do sieci o ograniczonych zasobach jest skutecznym sposobem zapewniania zakładanej średniej jakości usług. Konceptcja ta opiera się na filozofii systemów obsługi z opóźnieniem lub ze stratami, zgodnie z którymi chwilowe nadmiary zapotrzebowania na zasoby, tzn. ruchu wprowadzanego do sieci, w porównaniu z dostępnymi zasobami systemu, mogą być:

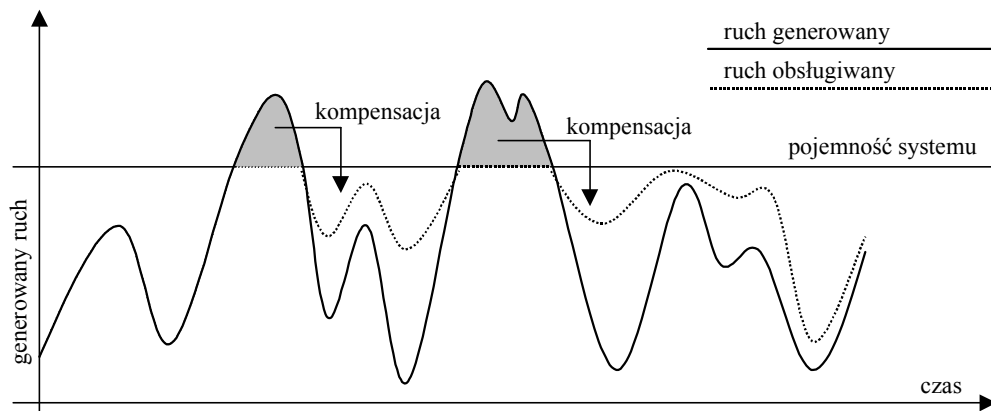
- kompensowane możliwością przechowywania chwilowych nadmiarów ruchu teleinformatycznego (systemy z opóźnieniami),
- odrzucane (systemy ze stratami).

W systemach z opóźnieniami nadmiarowe zapotrzebowanie na zasoby sieci, przekraczające maksymalną pojemność sieci, jest buforowane i wprowadzane do obsługi z opóźnieniem (rysunek 8.1a). Taki sposób obsługi ruchu nadmiarowego powoduje zmianę rzeczywistych charakterystyk ruchu w sieci, w porównaniu z charakterystykami ruchu wprowadzanego do sieci, oraz wpływa na zmiany wartości wskaźników jakości usług sieci. Kompensacja chwilowych różnic zapotrzebowania na zasoby i dostępnych zasobów, tzn. przesunięcie w czasie obsługi nadmiarowego ruchu, powoduje zmiany przepustowości oraz wzrost opóźnienia i fluktuacji opóźnień (rysunek 8.1c).

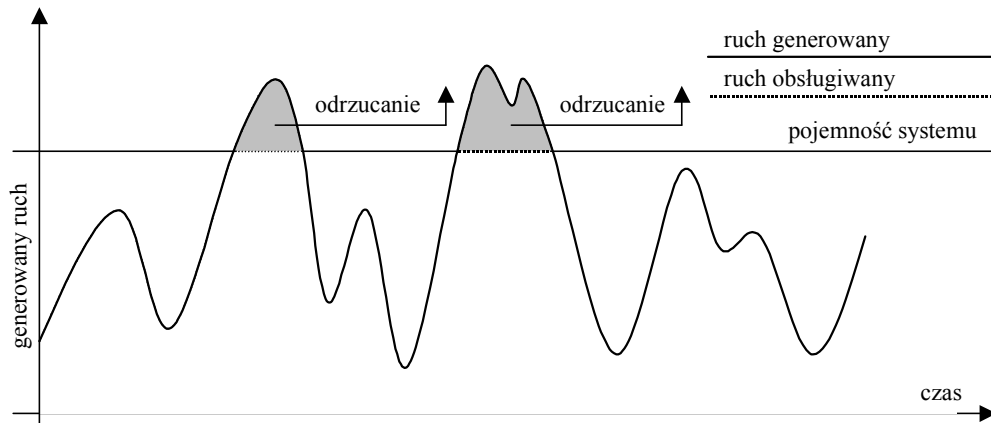
W systemach ze stratami nadmiarowe zapotrzebowanie na zasoby sieci, przekraczające maksymalną pojemność sieci, jest odrzucane na brzegach sieci, tzn. ilość ruchu wprowadzanego do sieci jest dostosowana do chwilowych możliwości obsługi wprowadzanego ruchu (rysunek 8.1b). Na ogół odrzucanie ruchu nadmiarowego nie powoduje zmian ustalonych wartości wskaźników jakości usług sieci (rysunek 8.1c) dla tej części ruchu, która jest przyjmowana do obsługi.

Konceptję usług dostarczanych bez gwarancji jakości charakteryzują:

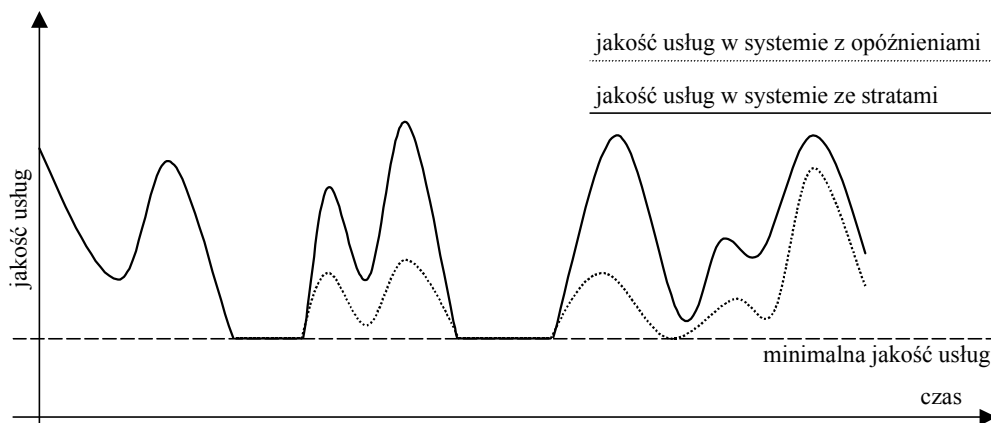
- ograniczona możliwość różnicowania ruchu i jakości usług, ze względu na przyjęcie statystycznego sposobu zwielokrotniania przepływów,
- rozkłady wartości parametrów jakości usług jako miary jakości usług; w większości przypadków wartości średnie parametrów jakości usług,
- wzrost jakości usług możliwy zarówno poprzez ograniczanie ilości ruchu współzawodniczącego o zasoby sieci, jak i implementację mechanizmów przeciwdziałania przeciążeniom oraz sterowania przepływem.



a) ruch generowany i obsługiwany w systemie z opóźnieniem



b) ruch generowany i obsługiwany w systemie ze stratami



c) jakość usług w systemie z opóźnieniem i w systemie ze stratami

Rysunek 8.1. Kompensacja chwilowego nadmiaru ruchu w systemie z opóźnieniami

8.1.2. KONCEPCJA USŁUG ZINTEGROWANYCH

Podstawą koncepcji usług zintegrowanych jest założenie, że efektywna alokacja skończonych zasobów sieci dla generowanych strumieni ruchu, a tym samym ich obsługa, jest możliwa wtedy, gdy obsługa odbywa się z wykorzystaniem wcześniej zarezerwowanych zasobów sieci. Aby maksymalizować stopień wykorzystania zasobów przyjęto, że zasoby są przydzielane na możliwie najniższym poziomie granulacji, tzn. na poziomie pojedynczych przepływów. Implementacja tej koncepcji oznacza realizację systemu z odrzuceniami. Brak wolnych zasobów w chwili żądania dostępu do sieci przez wywołanie powoduje odrzucenie wywołania. Koncepcja ta przypomina koncepcję udostępniania zasobów sieci właściwą dla systemów z komutacją kanałów.

Koncepcja usług zintegrowanych (opracowana przez *Integrated Service Working Group*) wykorzystuje tzw. podejście sesyjne (ang. *per-conversation traffic handling*) do jakości obsługi ruchu, tzn. opiera się na założeniu, że adresatem zasobów i przedmiotem sterowania jest pojedynczy przepływ (ang. *per-flow resource reservation*). Przeznaczanie zasobów dla pojedynczych przepływów daje możliwość uzyskania wysokich wartości wskaźników wykorzystania zasobów w sieci. Przydział zasobów sieci dla konkretnej aplikacji wymaga wcześniejszej rezerwacji zasobów; wymaga poprzedzenia transferu danych aplikacji w sieci wykonaniem ustalonej procedury rezerwacji zasobów.

Procedura rezerwacji zasobów dla każdego nowego wywołania jest wielokrokowa. Realizacja procedury rezerwacji, którą można interpretować jako procedurę prewencyjnego sterowania przepływem, wymaga:

1. Podania przez aplikację charakterystyk źródła ruchu oraz żądanych zasobów. Z charakterystyk tych korzystają protokoły wyboru trasy (ang. *routing protocol*) do znalezienia odpowiedniej, dla żądanych zasobów, trasy w sieci (wybór trasy jest rozwiązaniem wielokryterialnego zadania optymalizacji).

2. Znalezienie odpowiedniej trasy oznacza rozpoczęcie zadania rezerwacji zasobów na trasie przez odpowiedni protokół rezerwacji (ang. *reservation protocol*). Każdy element trasy jest sprawdzany ze względu na dostępność zasobów pozwalających na przyjęcie wywołania – dostępność żądanych zasobów oznacza ustanowienie połączenia i akceptację wywołania.

3. Ustanowienie połączenia umożliwia rozpoczęcie obsługi ruchu generowanego w ramach zaakceptowanego wywołania. Obsługa ruchu odbywa się z wykorzystaniem zasobów udostępnianych zaakceptowanemu wywołaniu na zasadach wyłączności, a udostępnianie zasobów jest wymuszane przez mechanizmy klasyfikacji i harmonogramowania pakietów.

Użytkownik sieci komputerowej, dostarczającej usługi QoS zgodnie z koncepcją (modelem) usług zintegrowanych, może wybrać jeden z dwóch modeli usług:

1. Model gwarantowanej usługi GS (ang. *Guaranteed Service*). Jakość usługi jest gwarantowana przez dostarczanie określonej ilości zasobów sieci oraz przyjęcie, na potrzeby sterowania dostępem i szeregowania przepływu, deterministycznego, górnego ograniczenia wartości opóźnienia. Ten model usługi został zaprojektowany dla

aplikacji wymagających absolutnych gwarancji górnego ograniczenia wartości opóźnienia. Usługa GS zapewnia brak strat jednostek danych oraz ogranicza maksymalne opóźnienie ich przejścia przez sieć.

2. Model sterowanego obciążenia usługi CLS (ang. *Controlled Load Service*). Akceptacja i obsługa nowych wywołań odbywa się na zasadach zbliżonych do koncepcji rezerwacji zasobów, z tym że na podstawie średnich wartości parametrów charakteryzujących generowany ruch. Ten model usługi nie daje absolutnych gwarancji wartości opóźnienia. Usługa CLS zapewnia obsługę jednostek danych z jakością, jaka byłaby dostarczana w sieci nieobciążonej, niezależnie od aktualnego stanu sieci. Z tego względu usługa GLS zapewnia jakość dostarczaną w sieci bez gwarancji jakości usług wtedy, gdy obciążenie sieci jest małe.

Wymienione dwa modele usług są komplementarne w tym znaczeniu, że realizują deterministyczny i statystyczny sposób zwielokrotniania przepływów. Dostępność dwóch modeli usług w architekturze usług zintegrowanych oznacza, że definicja usługi jest dwupoziomowa: w pierwszej kolejności definiowany jest model usługi, a następnie specyfikowane są wartości parametrów usługi.

Implementacja koncepcji usług zintegrowanych wymaga standardów dotyczących m.in. architektury alokacji zasobów, modeli usług oraz protokołu rezerwacji zasobów.

Sposoby alokacji i monitorowania zasobów, zaproponowane w koncepcji usług zintegrowanych, mają wiele ograniczeń, wynikających z:

1. Dużej granulacji monitorowanych i sterowanych w sieci procesów (przepływów). Dedykowanie zasobów i procedur sterowania indywidualnym przepływom w sieci powoduje konieczność jednoczesnej obsługi bardzo dużej, zwłaszcza w dużych sieciach, liczby przepływów i zarządzania rozdrobnionymi zasobami.

2. Zorientowania rozwiązań alokujących zasoby w celu zagwarantowania jakości usług na aplikacje charakteryzujące się względnie długim czasem wywołań i wrażliwych na opóźnienia.

3. Niewielkiej przydatności do obsługi względnie krótkich wywołań, ze względu na duże obciążenie nadmiarem organizacyjnym niezbędnym do rezerwacji zasobów i ich utrzymywania w czasie przyjmowania i obsługi wywołania.

4. Małej skalowalności. Przyjęcie zasady, że rezerwacja i sterowanie dotyczą z osobna każdego wywołania (przepływu) powoduje, że w każdym węźle trasy konieczne jest implementowanie mechanizmów klasyfikacji i szeregowania pakietów. Wątpliwa jest możliwość jednoczesnej obsługi dużej liczby przepływów z dużą szybkością transferu.

5. Złożoności systemu autoryzacji, potwierdzania i rozliczania obsługiwanych przepływów. Duża granulacja obsługiwanego ruchu wymaga rozbudowanego systemu pomiaru oraz weryfikacji źródeł i wielkości obsługiwanego ruchu zwłaszcza wtedy, gdy zwiększa się liczba dostawców usług w sieci.

Koncepcja usług zintegrowanych została zdominowana przez rozwiązania właściwe dla aplikacji odznaczających się względnie długim czasem trwania wywołań (ang. *long-lasting call*) i aplikacji wrażliwych na opóźnienia (ang. *delay-sensitive*). Tak zorientowana koncepcja jakości usług może być przydatna do obsługi aplikacji wy-

magających gwarantowanych przepustowości, takich jak telefonia IP, czy wideokonferencja w sieciach intranetowych, a protokoły rezerwacji zasobów mogą być wykorzystane do rezerwacji zasobów i sterowania dostępem dla ruchu wchodzącego do sieci rozległych.

Ograniczenia funkcjonalności i zakresu stosowalności koncepcji usług zintegrowanych wskazują na możliwość stosowania omawianej koncepcji w sieciach o ograniczonej i stabilnej liczbie użytkowników sieci, ograniczonej liczbie aplikacji tożsamej z ograniczoną liczbą przepływów i dostarczanych klas jakości usług oraz administrowanej przez jednego operatora.

8.1.3. KONCEPCJA USŁUG ZRÓŻNICOWANYCH

W koncepcji usług zróżnicowanych przyjęto, że alokacja skończonych zasobów sieci do obsługi różnych klas ruchu jest możliwa przez ograniczanie ilości ruchu wprowadzanego do sieci. Sterowanie ilością ruchu wprowadzanego do sieci i obsługiwanego dotyczy zarówno ilości ruchu w ramach wyróżnianych klas, jak i całkowitej sumy ruchu. Mechanizmy sterowania przepływem w koncepcji usług zróżnicowanych mają charakter mechanizmów reakcyjnych.

Sterowanie, polegające na ograniczaniu ilości wprowadzanego ruchu, eliminuje potrzebę stosowania procedur rezerwacji zasobów sieci. Koncepcję usług zróżnicowanych różni ponadto od koncepcji usług zintegrowanych to, że przedmiotem sterowania są klasy ruchu, tzw. agregaty strumieni ruchu, a nie indywidualne przepływy generowane dla połączeń źródło–ujście (ujścia), tzn. pomiędzy węzłami końcowymi sesji. Ograniczona i stała (lub wolnozmienna) liczba klas ruchu w sieci ogranicza ilościowo zadanie sterowania przepływem, uniezależniając rozmiar zadania sterowania od liczby aktualnie obsługiwanego przepływów, co oznacza dużą skalowalność tej koncepcji.

Przyjęcie założenia o ograniczaniu ilości wprowadzanego ruchu oznacza limitowanie dostępu do zasobów sieci na jej brzegu i przyjęcie schematu prewencyjnego sterowania przepływem. Koncepcja ta jest rozwinięciem koncepcji sterowania przepływem w sieciach pakietowych, opartych na kredytowaniu, tzn. wykorzystujących z góry ograniczoną liczbę przepustek. Rozwinięcie polega na tym, że system kredytowania jest dwupoziomowy: całkowity kredyt jest podzielony na składniki, które są przypisywane poszczególnym, wyróżnianym w sieci, klasom ruchu.

Podstawą koncepcji usług zróżnicowanych jest założenie o powoływaniu w sieci ograniczonej, względnie małej, w porównaniu z liczbą indywidualnych przepływów pomiędzy systemami końcowymi, liczby klas ruchu, tzw. klas spedycji ruchu (ang. *forwarding classes*). Ograniczenie liczby jednocześnie obsługiwanego klas ruchu upraszcza zadania monitorowania oraz sterowania dostępem do sieci i przepływem w sieci. Jednocześnie mniejsza liczba klas przepływów, dla tej samej ilości ruchu w sieci, ogranicza możliwość bardzo efektywnego wykorzystywania zasobów sieci.

Ilość ruchu, jaką użytkownicy mogą wprowadzić do sieci, w ramach każdej z wyróżnionych klas ruchu, jest ograniczona. Ograniczenie to jest egzekwowane na brzegu

sieci (ang. *network edge*), tzn. w routerach dostępowych do sieci (ang. *edge router*) lub w systemach końcowych. Udostępnianie zasobów sieci jej użytkownikom oraz sterowanie stopniem zaspokajania zapotrzebowania użytkowników na zasoby odbywa się przez zmiany ilości całkowitego ruchu dopuszczalnego do obsługi w sieci. Realizacja zadania dopuszczenia ruchu (ang. *traffic mapping*), generowanego w sesjach pomiędzy systemami końcowymi i należącego do ograniczonego zbioru klas ruchu, tylko na brzegu sieci (ang. *differentiated services network*), ogranicza m.in. zakres zadań realizowanych w węzłach szkieletowych (wewnętrznych) sieci (ang. *core router*) do względnie prostych zadań klasyfikacji i transmisji ruchu. Ograniczenie to pozwala na uproszczenie budowy węzłów szkieletowych sieci i uzyskiwanie większych szybkości obsługi ruchu.

Kryterium klasyfikacji ruchu są warunki negocjowanych uzgodnień poziomu usług, tzn. kontraktów SLA (ang. *Service Level Agreement*) zawieranych pomiędzy użytkownikiem sieci a dostawcą usług sieci. W kontrakcie SLA podawana jest liczba klas ruchu, poziom jakości usług i wartości parametrów ruchowych. Kontrakty SLA mogą być kontraktami statycznymi i dynamicznymi. Do zawierania kontraktów dynamicznych stosowane są protokoły rezerwacji zasobów sieci.

W węzłach dostępowych (routerach brzegowych) do sieci, odpowiadających m.in. za klasyfikację ruchu, kontroluje się także ruch, w celu zabezpieczenia sieci przed ruchem nadmiarowym, generowanym przez źródła z naruszeniem warunków zawartych kontraktów. Terminem kontrolowanie (ang. *policing*) określane jest proces dyskryminacji (traktowania) ruchu spoza profilu kontraktu SLA, tzn. nadmierowej części ruchu wobec ruchu opisanego wartościami parametrów kontraktu SLA. Jednostki danych należące do części nadmierowej ruchu są, po odpowiednim oznakowaniu, obsługiwane bez gwarancji jakości, tzn. mogą być odrzucane lub opóźniane w pierwszej kolejności w razie wystąpienia przeciążenia lub symptomów przeciążenia sieci.

Zasady wyróżniania klas ruchu (agregatów strumieni) mogą być różne i dotyczyć zwłaszcza:

- priorytetów odrzucania, gdzie wyższy priorytet klasy ruchu oznacza, że w razie wystąpienia przeciążenia ruch należący do klasy spedycyjnej z wyższym priorytetem odrzucania jest dyskryminowany w pierwszej kolejności,

- priorytetów dostępu do zasobów, gdzie wyższy priorytet klasy ruchu oznacza, że w razie wystąpienia konfliktu współzawodnictwa o zasoby sieci w pierwszej kolejności obsługiwany jest ruch należący do klasy z wyższym priorytetem dostępu do zasobów.

Przynależność jednostek danych ruchu do danej klasy spedycyjnej, realizowana na brzegu sieci, jest kodowana (np. w nagłówku pakietu warstwy sieciowej). Kod w nagłówku jednostki danych jest stosowany do klasyfikacji ruchu w węzłach wewnętrznych sieci, pozwalającej na różnicowanie sposobu obsługi węzłów należących do różnych klas.

Koncepcja zróżnicowanych usług ma, w porównaniu z koncepcją zintegrowanych usług, wiele zalet, do których należą m.in.:

1. Koncentracja złożonych zadań klasyfikacji przepływów, generowanych pomiędzy systemami końcowymi, do różnych klas ruchu na brzegu sieci. Podział i przydział ruchu do klas odbywa się tylko na brzegu sieci, tzn. w węzłach dostępowych, tzn. tam, gdzie szybkości strumieni ruchu są względnie małe w porównaniu z szybkościami transferu w węzłach wewnętrznych sieci.

2. Brak czasochłonnej i zasobochłonnej procedury rezerwacji zasobów. Dostęp do określonej klasy ruchu jest przedmiotem specyfikacji w kontrakcie użytkownika sieci i dostawcy usług.

3. Mniejsza liczba jednocześnie obsługiwanych strumieni. Procedury sterowania dostępem do sieci i sterowania przepływem nie dotyczą poszczególnych przepływów, a przedmiotem sterowania są zagregowane strumienie ruchu i przepływy.

4. Większa skalowalność. Zmniejszanie lub zwiększanie ilości ruchu obsługiwanego w sieci nie jest związane z koniecznością dużych zmian (jeżeli w ogóle) w liczbie powoływanych i obsługiwanych klas ruchu.

Podejście do zadania alokacji zasobów, proponowane w koncepcji usług zróżnicowanych, sprowadza się do takiego udostępniania zasobów sieci, aby wcześniej gwarantowane w kontraktach zasoby były dostarczane ograniczonej liczbie zróżnicowanych klas ruchu. Jakość gwarancji zależy od tego, jak realizowane jest udostępnianie (ang. *provisioning*) zasobów sieci oraz od tego, jak zasoby w sieci są zarządzane. Udostępnianie zasobów sieci dotyczy wszystkich zagadnień związanych z przyjmowaniem ruchu i szczególnie uważnego traktowania ruchu generowanego z naruszeniem kontraktów.

Precyzyjne udostępnianie sieci jest zadaniem trudnym i kosztownym ze względu na dynamikę przepływów w sieci. Z tego powodu udostępnianie zasobów sieci, jako sposób na gwarantowanie użytkownikom sieci deterministycznego dostępu do zasobów na wcześniej ustalonym poziomie jakości usług (koncepcja usług zróżnicowanych), jest zadaniem trudniejszym i droższym od realizacji zadania udostępniania zasobów poprzez ich rezerwację (koncepcja usług zintegrowanych).

Uproszczenie koncepcji usług zróżnicowanych, w porównaniu z koncepcją usług zintegrowanych, odbywa się za cenę gwarancji jakości usług. Gwarancja jakości usług w sensie deterministycznym, możliwa w koncepcji usług zintegrowanych, jest w koncepcji usług zróżnicowanych zastępowana gwarancją jakości usług w sensie statystycznym. Jeżeli żądana przez użytkownika jakość usług nie może być dostarczona przez sieć, to wywołanie jest odrzucane. Odrzucanie wywołań, które nie mogą być obsłużone na żądanym poziomie jakości, nie oznacza możliwości renegocjacji warunków obsługi na niższym poziomie jakości.

Podejście do różnicowania jakości usług w sieciach teleinformatycznych, reprezentowane w koncepcji różnicowania jakości usług (ang. *differentiated services*), opiera się na wyrafinowanych metodach sterowania dostępem i rezerwacji zasobów, dających gwarancję, w sensie deterministycznym lub statystycznym, uzyskania żądanych wartości bezwzględnych miar jakości (ang. *absolute performance measures*) usług sieci. Między innymi z tego powodu koncepcja ta jest określana mianem koncepcji bezwzględnego różnicowania jakości usług (ang. *absolute differentiated services*).

8.1.4. KONCEPCJA WZGLĘDNEGO RÓŻNICOWANIA JAKOŚCI USŁUG

Koncepcja bezwzględnego różnicowania jakości usług jest uzupełniana przez koncepcję względnego różnicowania jakości usług (ang. *relative differentiated services*). Istotą tego uzupełnienia jest zmiana definicji jakości usług dostarczanych przez sieć: gwarancje jakości usług dla klasy ruchu zastępuje możliwość jej obsługi w jednej z niewielu wyróżnianych klas jakości usług. Klasy jakości usług są uporządkowane według względnej jakości usług świadczonych w poszczególnych klasach. Zakresy wartości wybranych miar jakości usług, opisujące jakość usług dostarczanych aktualnie w poszczególnych klasach, są wprost proporcjonalne do wartości parametrów różnicowania klas ruchu (ang. *class differentiation parameters*) wybieranych do celów sterowania i zarządzania różnicowaniem jakości usług.

Koncepcja sieci ze względnym różnicowaniem jakości usług jest rozwiązaniem łączącym koncepcje świadczenia usług bez gwarancji jakości i bezwzględnego różnicowania usług. Łączenie polega na tym, że użytkownicy sieci mogą dynamicznie wybierać klasę jakości usług spełniającą ich wymagania dotyczące jakości usług i ograniczenia kosztu usług.

Podstawą koncepcji względnego różnicowania jakości usług jest przekonanie, że różnicowanie jakości usług dla wyróżnianych w sieci klas ruchu jest możliwe albo przez przewymiarowanie sieci (ang. *network overprovision*), albo przez implementację w sieci wyrafinowanych mechanizmów udostępniania zasobów sieci zgodnie z koncepcją usług zintegrowanych lub koncepcją usług różnicowanych.

Przewymiarowanie sieci, tzn. udostępnienie użytkownikom sieci nadmiaru zasobów w ilościach pozwalających na obsługę wyróżnianych klas ruchu na żądanym poziomie jakości w tzw. najgorszym przypadku, jest najprostszym sposobem przeciwdziałania przeciążeniom sieci, tj. unikania strat jednostek danych w czasie transmisji w sieci oraz oczekiwania jednostek danych w kolejkach na obsługę w węzłach sieci. Przewymiarowanie sieci pozwala na rezygnację ze złożonych mechanizmów udostępniania zasobów i nadzorowania ich wykorzystania, tzn. na dostarczanie usług według modelu bez gwarancji jakości. W przewymiarowanej sieci, zorientowanej na obsługę ruchu w tzw. najgorszym przypadku, jakość obsługi wszystkich klas ruchu jest, niezależnie od żądanej jakości usług, taka jak jakość obsługi klasy ruchu o najwyższych wymaganiach. Przewymiarowanie sieci, tj. realizacja tzw. modelu „grubej rury” (ang. *fat-dumb-pipe*) jest, oczywiście, skrajnie nieefektywnym rozwiązaniem ze względu na koszty realizacji sieci, zarządzanie zasobami i stopień wykorzystania zasobów.

Zasadnicza różnica pomiędzy koncepcjami usług zróżnicowanych i względnego różnicowania jakości usług polega na sposobach zarządzania przyjmowaniem wywołań użytkowników i zasobami sieci:

➤ w sieci usług zróżnicowanych jakość usług jest gwarantowana; brak możliwości zrealizowania usługi na żądanym poziomie jakości oznacza odrzucenie wywołania,

➤ w sieci ze względnym różnicowaniem jakości usług ilość i jakość usług dostarczanych danej klasie ruchu zależy od aktualnego obciążenia sieci; użytkownik sieci może albo zredukować żądania dotyczące jakości usług, albo korzystać z obsługi w wyższej klasie usług za wyższą cenę.

Centralne miejsce w koncepcji względnego różnicowania jakości usług zajmuje uporządkowany zbiór klas usług. Podstawą uporządkowania klas usług jest jakość transmisji pakietów: i -ta klasa usług jest lepsza (lub co najwyżej nie gorsza) od $(i + 1)$ -szej klasy usług w sensie wartości lokalnych wskaźników jakości dotyczących opóźnień kolejkowania i strat pakietów. Termin „nie gorsza” odnosi się do przypadku, kiedy względnie niewielkie obciążenie sieci powoduje, że jakość obsługi we wszystkich wyróżnionych klasach usług może być taka sama. W zależności od implementacji klasyfikacja pakietów do różnych klas może być realizowana albo przez aplikację w węzle źródłowym, albo przez router brzegowy sieci.

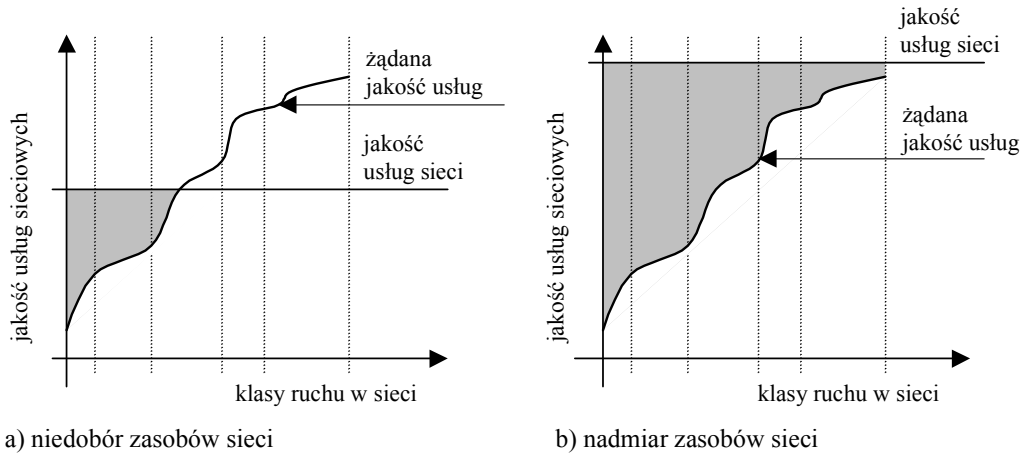
Realizacja koncepcji ze względnym różnicowaniem jakości usług jest i musi być ściśle związana z taryfikacją usług, zgodnie z którą wyższa klasa usług oznacza wyższy koszt dostarczania usług. Taryfikacja usług może być albo ryczałtowa (użytkownik abonuje pewną klasę usług i w rezultacie wszystkie generowane pakiety należą do tej klasy usług), albo zależna od ilości usług otrzymanych w poszczególnych klasach (wysokość opłaty zależy od ilości ruchu obsługiwanego w poszczególnych klasach usług). Wybranie jednego z dwóch sposobów taryfikacji usług ma wpływ na definiowanie klas usług, zarządzanie przyjmowaniem ruchu do obsługi w tych klasach oraz spodziewane zachowania użytkowników sieci.

8.1.5. JAKOŚĆ USŁUG I ZARZĄDZANIE RUCHEM W SIECI

Omawiane koncepcje dostarczania jakości usług dla ruchu w sieci teleinformatycznej charakteryzują różne sposoby zarządzania wykorzystaniem zasobów w sieci.

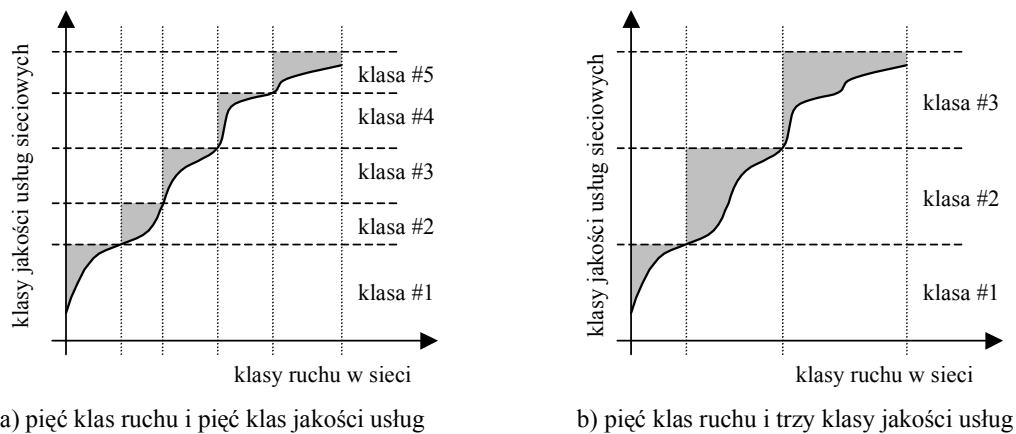
Obsługa ruchu w sieci bez gwarancji jakości oznacza, że całkowity ruch wprowadzany do sieci jest obsługiwany z taką samą jakością. Jakość dostarczanych usług zależy od ilości zasobów dostępnych w sieci oraz ilości i wymagań jakościowych ruchu wprowadzanego do sieci. Obsługa bez gwarancji jakości charakteryzuje się tym, że pewnym wyróżnionym klasom ruchu są dostarczane usługi o jakości przekraczającej żadaną jakość, innym natomiast o jakości niższej od żądanej (rysunek 8.2a).

Dostarczanie pewnym klasom ruchu jakości usług wyższej od żądanej można interpretować jako marnotrawienie zasobów sieci. Miarą marnotrawienia zasobów jest ilość zasobów zużyta na dostarczenie jakości usług większej od żądanej. Na rysunku 8.2 miarą marnotrawienia zasobów sieci jest powierzchnia obszaru położonego pomiędzy krzywą żądanej jakości usług a prostą oznaczającą jakość usług sieci. Rozwiązaniem problemu dostarczania usług o jakości niższej od żądanej jest przewymiarowanie sieci, tzn. udostępnienie zasobów w ilościach pozwalających na spełnienie żądań jakościowych najbardziej wymagającej klasy ruchu (rysunek 8.2b). Z rozwiązaniem takim wiąże się wzrost ilości marnotrawionych zasobów sieci.



Rysunek 8.2. Obsługa ruchu w sieci bez gwarancji jakości w zależności od ilości zasobów sieci

Rozwiązaniem problemu występującego w nieprzewymiarowanych sieciach bez gwarancji jakości usług (rysunek 8.2.a) i polegającego na jednoczesnym braku możliwości spełnienia wymagań jakościowych niektórych klas ruchu oraz marnotrawieniu zasobów sieci na obsługę innych klas ruchu z jakością wyższą od żądanej, jest wprowadzenie w sieci różnicowania obsługi klas ruchu, czyli klas jakości usług. Różnicowanie obsługi klas ruchu wyróżnianych w ruchu całkowitym polega na takiej dystrybucji zasobów, która pozwala spełnić wymagania jakościowe poszczególnych klas ruchu i ogranicza ilości marnotrawionych zasobów sieci (rysunek 8.3). Ilość marnotrawionych zasobów maleje, co oznacza że rośnie wartość współczynnika wykorzystania zasobów sieci wraz ze wzrostem liczby klas jakości usług, z tym że zwiększenie liczby klas ruchu wiąże się z rozbudową zarządzania ruchem.



Rysunek 8.3. Obsługa ruchu w sieci z różnicowaniem jakości usług

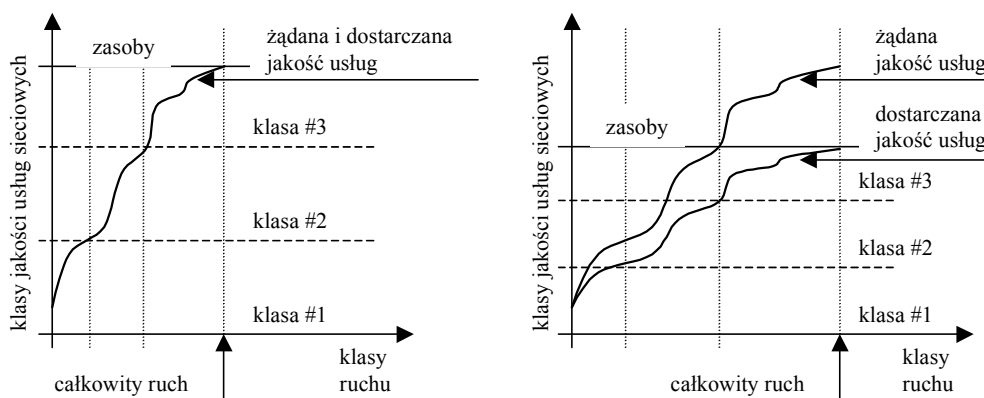
W przypadku realizacji koncepcji sieci usług integrowanych liczba klas ruchu jest równa aktualnej liczbie przepływów w sieci. Ze względu na to, że jakość obsługi każdego z przepływów może być inna, liczba klas ruchu jest praktycznie równa liczbie przepływów.

Agregacja przepływów w klasy ruchu, charakterystyczna dla koncepcji usług zróżnicowanych, oznacza zmniejszanie liczby dostępnych w sieci sposobów obsługi ruchu, tzn. liczby klas jakości usług (rysunek 8.3b).

Wraz ze zmniejszaniem liczby klas jakości usług następuje zwiększenie zakresów wartości miar jakości stosowanych do definiowania jakości obsługi ruchu w sieci. W przypadku realizacji w sieci koncepcji bezwzględnego różnicowania jakości usług zakresy wartości parametrów definiujących klasy obsługi ruchu są stałe.

Podczas realizacji w sieci koncepcji względnego różnicowania jakości usług, jakość usług w poszczególnych klasach usług zależy od aktualnego obciążenia sieci. Jeżeli ilość zasobów w sieci jest stała, zmiana ilości całkowitego ruchu w sieci wpływa na zmianę jakości usług dostarczanych poszczególnym klasom ruchu (rysunek 8.4). Na rysunku przyjęto, że ilość zasobów sieci jest stała, udział poszczególnych klas ruchu w całkowitej ilości ruchu jest niezmienny oraz że żądana jakość usług poszczególnych klas ruchu jest niezmienna.

Jeżeli ilość całkowitego ruchu w sieci jest dopasowana do ilości zasobów w sieci, to jakość usług dostarczanych poszczególnym klasom ruchu jest zgodna z żadaną jakością, tzn. krzywe żądanej i dostarczanej jakości usług pokrywają się (rysunek 8.4a). Gdy ilość ruchu wzrasta i nie jest możliwe spełnienie żądań jakościowych, następuje obniżenie jakości usług we wszystkich wyróżnionych klasach usług (rysunek 8.4b).



a) jakość dostarczanych usług zgodna z żadaną

b) jakość dostarczanych usług poniżej żądanej

Rysunek 8.4. Obsługa ruchu w sieci z względnym różnicowaniem jakości usług

Krzywa dostarczanej jakości usług jest proporcjonalna do krzywej żądanej jakości usług, tzn. jakość usług w każdej z wyróżnionych klas ulega obniżeniu, ale względna

jakość usług dostarczanych w poszczególnych klasach nie ulega zmianie (rys. 8.4b). Względne różnicowanie jakości usług oznacza także, że w razie zwiększenia ilości ruchu w sieci maleją zakresy wartości parametrów definiujących jakość usług dostarczanych w danej klasie jakości usług. W takich przypadkach użytkownik sieci może żądać obsługi w wyższej klasie jakości usług lub obniżyć żadaną jakość usług dla generowanego ruchu.

8.2. OPTIMALIZACJA WYKORZYSTANIA ZASOBÓW

Przyjęcie określonej koncepcji dostarczania jakości usług sieci, definiującej architekturę alokacji zasobów sieci, i obowiązujących w tej architekturze modeli usług pozwala na formułowanie zadań optymalizacji wybranych wskaźników jakości działania sieci (ang. *network performance*). Implementację określonej architektury alokacji zasobów sieci można utożsamiać z istnieniem pośrednich mechanizmów sterowania przepływami (ang. *implicit flow control*), realizacja zadania optymalizacji działania sieci wymaga natomiast uzupełnienia sieci o mechanizmy bezpośredniego sterowania przepływami (ang. *explicit flow control*).

Optymalizacja stopnia wykorzystania zasobów jest zadaniem zdecydowanie trudniejszym w sieciach z koncepcją usług zróżnicowanych, niż w sieciach stosujących koncepcję usług zintegrowanych.

W koncepcji usług zróżnicowanych przedmiotem sterowania są klasy ruchu będące grupami przepływów, których sumaryczne chwilowe zapotrzebowanie na zasoby sieci może być większe od zasobów dostępnych w danej chwili. Złożoność obliczeniowa zadania sterowania wynika ze złożoności sumarycznego przepływu i jego zapotrzebowania na zasoby oraz konieczności jego rozwiązywania w czasie rzeczywistym ze względu na dominację sterowania o charakterze reakcyjnym. Rozmiar zadania optymalizacji jest względnie mały z powodu ograniczonej liczby klas ruchu.

W sieciach korzystających z koncepcji usług zintegrowanych zadanie sterowania przepływem jest rozwiązywane na etapie rezerwacji zasobów. Zadanie sterowania jest w tym przypadku o tyle łatwiejsze, że dotyczy względnie wolnozmiennych, indywidualnych przepływów, odbywa się na brzegu sieci i polega na niedopuszczeniu do przekroczenia potencjału sieci; jest zadaniem sterowania prewencyjnego. Rozmiar zadania optymalizacji jest w tym przypadku duży i wynika z implementacji procedur sterowania na najniższym możliwym poziomie granulacji przepływów, tj. na poziomie przepływów pomiędzy systemami końcowymi.

Optymalizacja stopnia wykorzystania zasobów sieci najczęściej sprowadza się do poszukiwania takiej organizacji zasobów sieci, która dla zadanych procedur udostępniania zasobów realizuje dwa sprzeczne cele: maksymalizuje wartość prawdopodobieństwa realizacji zawartych kontraktów, jednocześnie minimalizując koszt dostarczenia usług spełniających ilościowe i jakościowe warunki kontraktów ruchowych.

Optymalizacja jakości działania sieci w ramach ustalonej architektury alokacji zasobów jest istotnym elementem wdrażania określonej polityki QoS. Implementacja polityki QoS oznacza nie tylko implementację mechanizmów klasyfikacji, znakowania, szeregowania, monitorowania, identyfikacji i dyskryminacji ruchu, ale także możliwość kreowania nowych usług. Koszt uzyskania tych usług, a zatem koszt ich udostępniania użytkownikom, jest różny. Zadaniem procedur optymalizacji wykorzystania zasobów jest m.in. wspieranie procesów kreowania nowych usług i ich wartościowania.

Celem optymalizacji mogą być m.in.: maksymalizacja stopnia wykorzystania zasobów sieci, minimalizacja prawdopodobieństwa powstania przeciążeń w sieci, minimalizacja prawdopodobieństwa strat jednostek danych. Optymalne wartości różnych wskaźników jakości są zwykle uzyskiwane wtedy, gdy ruch wprowadzany do sieci jest rozdzielany równomiernie na poszczególne zasoby. Stopień równomierności rozdzielania ruchu w sieci zależy od stopnia granulacji (agregacji) ruchu sterowanego. Im większy stopień agregacji ruchu, tym zadanie równomiernego rozdzielania ruchu jest trudniejsze, a stopień równomierności rozdzielania ruchu jest mniejszy.

Aby uzyskać możliwie równomierny rozdział ruchu i jednocześnie spełnić warunki formułowane przez architekturę alokacji zasobów, wymagane jest stosowanie wyrafinowanych algorytmów wyznaczania (doboru) tras w sieci dla przepływów o różnej granulacji oraz sposobów kierowania ruchem zdolnych wykorzystać pojemności wyznaczonych tras.

Zadania optymalizacji jakości działania sieci są z reguły realizowane przez złożenie rozwiązań wzajemnie zależnych i rozłącznych zadań: wyznaczania tras i kierowania ruchem. Rozłączność tych zadań powoduje, że rozwiązania zadań wyznaczania tras są niezależne od przyjętych sposobów kierowania ruchem, a przepływy na wyznaczonych trasach mogą być sterowane wieloma różnymi protokołami. Niezależność zadania wyznaczania tras od protokołów kierowania przepływem pozwala na stosowanie wielokryterialnych metod wyboru tras dla przepływów o różnej granulacji, tzn. metod właściwych dla zadań inżynierii ruchu (ang. *traffic engineering*). Dwupoziomowe rozwiązywanie omawianych zadań wymusza konieczność stosowania technik sterowania przepływem mających zdolność koordynacji różnych protokołów. Wymagania takie spełnia m.in. technika komutacji etykiet (ang. *label switching*).

8.3. WYZNACZANIE TRAS I STEROWANIE PRZEPLÝWAMI

Realizacja zadań optymalizacji wykorzystania zasobów wymaga wbudowania w sieć mechanizmów bezpośredniego sterowania przepływem w sieci i jej fragmentach tak, aby istniała możliwość takiej organizacji przepływów, która maksymalizuje zarówno stopień wywiązywania się ze zobowiązań zawartych w kontraktach, jak i stopień wykorzystania sieci.

Zadaniem metod inżynierii ruchu korzystających z zaawansowanych algorytmów wyboru tras jest takie udostępnianie zasobów sieci (zwłaszcza pojemności kanałów transmisyjnych) do organizowania przepływów, które maksymalizuje efektywność sieci. Ogólnie stosowane podejście polega na tym, że na podstawie znajomości charakterystyk strumieni ruchu, tzn. dla klas ruchu i (lub) przepływów, wyznaczane są trasy i obliczane ich pojemności. Tak wyznaczone trasy są udostępniane, jako trasy bezpośrednie (ang. *explicit route*), jednocześnie różnym protokołom kierowania ruchem.

Zadanie inżynierii ruchu, w jego podstawowym sformułowaniu, jest następujące: dla danej sieci oraz znanych wymagań użytkowników znaleźć taką organizację przepływów w sieci, która optymalizuje wybrane wskaźniki jakości. Ogólne zadanie inżynierii ruchu można traktować jako uszczegółowienie klasycznego zadania wyznaczania optymalnych tras (ang. *routing*) dla wieloskładnikowych przepływów w sieci (ang. *multicommodity flow*) oraz dla różnych wskaźników jakości działania sieci.

Algorytmy wyznaczania tras w sieciach gwarantujących jakość usług, z powodu stawianych przed nimi wymagań, znacznie różnią się od klasycznych algorytmów wyznaczania tras pomiędzy źródłem a ujściem (ang. *source-destination based routing*) stosowanych w protokołach sieciowych (np. protokoły IP). Małą przydatność tych ostatnich, pozwalających na znajdowanie najkrótszych tras do ujścia na podstawie bardzo prostych metryk (odległość, opóźnienie), wynika z ich jednokryterialności, względnie małej elastyczności (ściśle ograniczone właściwości funkcji kryterialnych), braku zdolności znakowania i identyfikacji zasobów, ograniczonych zdolności rozdzielania ruchu itp. Trasy, wyznaczane przez omawiane algorytmy, nie gwarantują pożądaných w sieciach, zorientowanych na jakość usług, właściwości: małych wartości prawdopodobieństwa strat pakietów, wysokiego stopnia wykorzystania zasobów i zapobiegania powstawaniu przeciążeń.

Wymagania stawiane algorytmom wyznaczania tras w sieciach gwarantujących jakość usług dotyczą m.in. możliwości wyznaczania tras:

- dla przepływów o dużej skali zmienności zarówno wolumenu ruchu, jak i zakresu przestrzennego,
- w ograniczonych fragmentach sieci, tzn. nie tylko tras pomiędzy systemami końcowymi (źródłami a ujściami) zlokalizowanymi na brzegu sieci,
- gwarantujących równomierność rozkładu obciążenia zasobów sieci dla różnego stopnia granulacji przepływów,
- zawierających ściśle określone żądane zasoby.

Ze względu na ograniczenia formułowane dla zadań wyznaczania tras w sieciach z różnym stopniem gwarancji jakości usług zadania te są rozwiązywane przez tzw. algorytmy wyznaczania tras z ograniczeniami (ang. *constraint-based routing*). Efektywność, elastyczność i skalowalność algorytmów wyznaczania tras z ograniczeniami jest praktycznie możliwa tylko w systemach scentralizowanych pozwalających na dostęp do danych o zasobach całej sieci i wymagań całego ruchu przyjmowanego do obsługi w sieci.

8.4. HIERARCHIA ZARZĄDZANIA PRZEPIYWAMI W SIECIACH

Efektywne zarządzanie przepływami w trasach, wyznaczanych niezależnie od protokołów kierowania ruchem w sieci, wymaga stosowania rozwiązań pozwalających koordynować działania różnych protokołów kierowania ruchem, jednocześnie wykorzystywanych w sieci, tj. rozwiązań nadrzędnych nad protokołami kierowania ruchem. Zadaniem nadrzędnego poziomu kierowania ruchem nie jest zastępowanie protokołów sterowania przepływami, ale uzupełnienie ich o mechanizmy pozwalające korzystać z alternatywnych rozwiązań dostarczanych przez metody inżynierii ruchu.

Cechy wymaganego w takim celu rozwiązania ma technika komutacji etykiet (ang. *label switching*), integrująca dwa różne tryby obsługi ruchu w sieciach pakietowych: trybu datagramowego (ang. *datagram*) i połączeń wirtualnych (ang. *virtual circuit*). Technikę komutacji etykiet można traktować jako nadrzędną nad techniką komutacji pakietów, pozwalającą na wprowadzenie dodatkowego stopnia swobody w zadaniach kierowania ruchem przez możliwość agregowania na trasie komutacji etykiet przepływów kierowanych przez różne protokoły warstw sieciowej i niższych.

Porównywanie działania technik komutacji etykiet i komutacji pakietów w sieciach jest *de facto* porównywaniem zalet i wad systemów z implementacją technik komutacji (ang. *switching*) i technik wyboru tras (ang. *routing*), które to techniki są kontynuacją dwóch klasycznych podejść do zadania komunikacji w sieciach, odpowiednio, komutacji łączy i komutacji pakietów.

Podstawową zaletą techniki komutacji etykiet jest możliwość jej stosowania do uproszczenia procesów kierowania ruchem i uniezależnienia ich od protokołów sieciowych. Uniwersalność i elastyczność techniki komutacji etykiet jest osiągnięta za cenę konieczności ustanowienia trasy komutacji etykiet przed rozpoczęciem transferu pakietów. Z kolei ceną za uproszczenie transferu pakietów jest konieczność implementacji protokołu sygnalizacyjnego lub protokołu dystrybucji etykiet.

Cechy techniki komutacji etykiet wykorzystywane są m.in. do realizacji następujących zadań:

- eliminacji zjawiska polegającego na możliwym dublowaniu obsługi tras w protokołach należących do różnych warstw sieci,
- jednoczesnej obsługi wielu logicznych podsieci opartych na różnych schematach adresowania i komutacji; na trasie komutacji etykiet możliwe jest współistnienie przepływów wieloprotokołowych,
- uproszczenia zadania integracji sieci heterogenicznych, tzn. różniących się protokołami warstwy sieciowej i wykorzystujących rozwiązania homogeniczne w warstwach niższych niż sieciowa; uproszczenie polega na możliwości zastąpienia, w zadaniach kierowania ruchem, hierarchii protokołów warstwowych kooperacją protokołów,
- uproszczenia zadania translacji i integracji rozwiązań gwarantujących jakość usług dostępnych w różnych warstwach architektury sieci,

➤ oddzielenia zadań kierowania ruchem i zadań wyboru trasy; implementacja różnych algorytmów wyznaczania tras jest możliwa bez zmiany reguł kierowania ruchem, a z kolei ten sam schemat wyboru trasy może być wykorzystywany przez różne schematy kierowania ruchem,

➤ redukcji czasu podejmowania decyzji w zadaniach klasyfikacji pakietów na potrzeby kierowania ruchem; tym samym ograniczenia problemu skalowalności sieci oraz wzrostu szybkości działania sieci,

➤ wielokryterialnej granulacji przepływów w sieciach; jakkolwiek granulację przepływów pod kontrolą pakietowych protokołów sieciowych charakteryzuje wysoka skalowalność, to potrzeba innego podziału ruchu wymaga abstrahowania od adresów ujęcia – podziały takie mogą być formowane przez złożone reguły filtracji ruchu realizowane na brzegu sieci i przekazywane w zestawianych trasach komutacji etykiet,

➤ równoważenia obciążeń; równomierny rozdział ruchu klasyfikowanego według adresów ujęć jest możliwy tylko w przypadku stosowania do wyboru tras złożonych systemów metryk; równomierny rozdział znacznie łatwiej można uzyskać przez podział tak klasyfikowanego ruchu na części obsługiwane przez różne trasy komutacji etykiet (przez różne etykietowanie fragmentów tak klasyfikowanego ruchu).

Przykładem rozwiązywania zadania optymalizacji jakości działania sieci, polegającego na wykorzystaniu metod inżynierii ruchu oraz techniki komutacji etykiet, jest realizacja tzw. modelu pokrycia (ang. *overlay model*). Zadaniem metod inżynierii jest w tym przypadku wyznaczenie, przy użyciu algorytmów wyznaczania tras z ograniczeniami, zbiorów dopuszczalnych tras dla każdego połączenia logicznego w pełnej sieci wirtualnej (ang. *mesh network*). Kierowanie przepływami w połączeniach logicznych, korzystającymi z wcześniej wyznaczonych tras, jest realizowane z użyciem techniki komutacji etykiet. Rozwiązanie to integruje korzyści powstałe w wyniku rozdzielenia zadań wyboru tras dla przepływów i kierowania przepływami.

Implementację modelu pokrycia realizuje się w dwóch krokach:

➤ dla danego zbioru węzłów brzegowych budowana jest pełna sieć wirtualna, zawierająca wszystkie możliwe połączenia logiczne pomiędzy wszystkimi węzłami brzegowymi sieci,

➤ dla tak zbudowanej sieci połączeń logicznych, danej macierzy wymagań użytkowników (macierzy ruchu) oraz dla wszystkich możliwych połączeń logicznych wyznaczane są, przy użyciu algorytmów wyznaczania tras z ograniczeniami, zbiory tras; dostępność więcej niż jednej trasy pozwala na taki podział ruchu, który maksymalizuje stopień wykorzystania zasobów sieci.

Zaletą omawianego podejścia jest większa, w porównaniu z koncepcją obsługi przepływu przez najkrótszą z tras, elastyczność. Do obsługi przepływu może być wykorzystana każda z dopuszczalnych, wcześniej wyznaczonych, tras. Decyzja o wyborze jednej z dopuszczalnych tras jest podejmowana na podstawie analizy aktualnie obsługiwanego ruchu i aktualnego obciążenia zasobów sieci.

Implementacja modelu pokrycia i zarządzanie ruchem w tym modelu jest zadaniem trudnym i nie zawsze praktycznie realizowalnym, m.in. z następujących powodów:

➤ skalowalności – rosnąca liczba węzłów brzegowych powoduje znaczący przyrost liczby wszystkich połączeń logicznych w sieci pełnej, z czym związane jest zwiększenie narzutu organizacyjnego, m.in. w postaci wydłużenia nagłówek jednostek danych,

➤ realizowalności – zwiększenie liczby węzłów brzegowych powoduje znaczący wzrost liczby wymaganych portów w każdym z urządzeń,

➤ niezawodności – obsługa awarii w dwupoziomowej sieci pełnej jest zadaniem złożonym ze względu na to, że awaria dowolnego łącza fizycznego obsługującego połączenia logiczne może powodować zerwanie wielu z tych połączeń.

8.5. PODSUMOWANIE

Podstawową zaletą sieci zgodnej z modelem obsługi ruchu teleinformatycznego bez gwarancji jakości jest względnie prosta realizacja zarządzania dostępem do sieci i zasobami sieci. Ze względu na to, że jakość usług w sieci bez gwarancji jakości jest taka sama dla wszystkich użytkowników, spełnienie żądań jakościowych wszystkich użytkowników sieci jest możliwe tylko wtedy, gdy możliwe jest spełnienie wymagań jakościowych najbardziej wymagających użytkowników. Przypadek taki zachodzi tylko wtedy, gdy ilość dostępnych zasobów w sieci pozwala na obsługę tzw. najgorszego przypadku, czyli wtedy, gdy zasoby sieci są przewymiarowane. Przewymiarowanie zasobów sieci jest związane z niskim stopniem wykorzystania zasobów.

Jedynym sposobem pozwalającym na osiągnięcie kompromisu pomiędzy dostarczaniem żądanej jakości usług dla przyjmowanego do sieci ruchu i efektywnym wykorzystaniem zasobów sieci jest różnicowanie jakości usług. Różnicowanie jakości usług polega na dostarczaniu różnym klasom ruchu różnej jakości usług za cenę rozbudowy – w porównaniu z modelem usług bez gwarancji jakości – sterowania i zarządzania dostępem ruchu do sieci i rozdziałem zasobów.

Dwie podstawowe koncepcje różnicowania jakości usług w sieciach wiążą się z różnym poziomem granulacji obsługiwanego ruchu i różnymi sposobami definiowania jakości usług sieci. W przypadku koncepcji usług zintegrowanych jednostką sterowania ruchem jest indywidualny przepływ, w koncepcji usług zróżnicowanych jednostką sterowania ruchem są natomiast agregaty przepływów indywidualnych.

Koncepcja usług zintegrowanych ma wiele właściwości decydujących o korzyściach wynikających z jej implementacji:

➤ model usług daje możliwość gwarancji jakości; model funkcjonalny jest specyfikowany z takim stopniem szczegółowości, który umożliwia dedykowanie zasobów dla każdego z przepływów oraz monitorowanie i zarządzanie ruchem na poziomie indywidualnych przepływów,

➤ do realizacji zadań rezerwacji i utrzymywania zasobów wykorzystywane są istniejące protokoły routingu,

➤ schemat rezerwacji zasobów umożliwia obsługę przepływów różnych trybów komunikacji, tzn. punkt – punkt oraz punkt – wielopunkt.

Ograniczenia koncepcji usług zintegrowanych wynikają z zakładanych, dla tej koncepcji obsługi ruchu w sieci, elastyczności, uniwersalności i dużej szczegółowości rozdziału zasobów i sterowania ruchem, które powodują, że:

➤ aktywne elementy sieci są obciążone dużym nadmiarem organizacyjnym, wynikającym ze szczegółowości rozwiązań i dużej granulacji ruchu,

➤ rozwiązania są niepraktyczne dla krótkotrwałych przepływów; efektywność czasochłonnych procedur rezerwacji i utrzymywania zarezerwowanych zasobów zwiększa się wraz z wydłużeniem czasu trwania przepływów,

➤ rozwiązania są wrażliwe na zmiany architektury sieci oraz ilościowe i jakościowe zmiany wymagań użytkowników,

➤ złożoność systemu rozliczania jakości usług sieci zwiększa się wraz ze wzrostem elastyczności i uniwersalności usług i ich jakości.

Większość istotnych właściwości koncepcji usług zróżnicowanych wynika z założenia, że sposobem redukcji problemu skalowalności koncepcji usług zintegrowanych jest ograniczenie liczby sposobów obsługi całości ruchu w sieci do względnie niewielkiej liczby klas ruchu. Przyjęcie takiego założenia oznacza, że:

➤ wymagania dotyczące jakości usług nie są negocjowane pomiędzy systemami końcowymi; zbędne są procedury rezerwacji i utrzymywania rezerwacji zasobów,

➤ efektywność wykorzystania zasobów i jakości obsługi ruchu nie zależy od czasu trwania przepływów,

➤ przewidywalność stanów, zmian stanów oraz jakości usług w sieci jest ograniczona; zmienność ruchu i potencjalne przeciążenia obniżają zdolność sieci do gwarantowania jakości usług.

Koncepcja usług zróżnicowanych powstała m.in. na bazie krytycznej analizy i rozwinięcia koncepcji usług zintegrowanych. Obecnie jest najbardziej rozwiniętą i dojrzałą koncepcją dostarczania jakości usług i efektywnego wykorzystania zasobów sieci. Głównym ograniczeniem stosowalności tej koncepcji jest złożoność mechanizmów pozwalających na jej implementację.

Koncepcja usług zróżnicowanych występuje w dwóch wersjach: bezwzględnego i względnego różnicowania jakości usług. W pierwszej z wymienionych wersji jakość usług poszczególnych klas jakości usług jest definiowana za pomocą deterministycznych zakresów wartości wskaźników jakości usług. W wersji ze względnym różnicowaniem jakości usług zakres wartości wskaźników jakości definiujących klasy usług zależy od aktualnego obciążenia sieci. Koncepcję tę można – w pewnym sensie – uznać za rozwiązanie łączące cechy modelu obsługi bez gwarancji jakości i koncepcji usług zróżnicowanych. Łączenie to ma na celu uzyskanie korzyści z różnicowania jakości usług i ograniczenie złożoności mechanizmów sterowania przepływami i zarządzania zasobami sieci. Atrakcyjność koncepcji względnego różnicowania jakości usług wynika przede wszystkim z rosnącej ilości aplikacji tolerujących zmienność jakości usług sieci (ang. *adaptive applications*).

LITERATURA

- [1] BHARATH-KUMAR K., JAFFE J.M., *Routing to multiple destinations in computer networks*, IEEE Transactions on Communications, Vol. COM-31. No. 3, March 1983, s. 343–351.
- [2] BURAKOWSKI W., BĄK A., KOPERTOWSKI Z., WYKROTA A., *Mechanizmy QoS w sieci IP*, II ogólnopolska konferencja „Internet”, Wrocław 2000, s. 25–37.
- [3] CAMPBELL A., COULSON G., GARCIA F., HUTCHISON D., LEOPOLD H., *Integrated quality of service for multimedia communications*, Lancaster University (www.itpapers.com).
- [4] CHAO H.J., GUO X., *Quality of Service control in High-Speed Networks*, Wiley, 2002.
- [5] DECASPER D., DITTIA Z., PARULKAR G., PLATTNER B., *Router plugins: a software architecture for next generation routers*, IEEE/ACM Transactions on Networking, Vol. 8, No. 1, February 2000, s. 2–16.
- [6] DeWITT T., GROSS T., LOWEKAMP B., MILLER N., STEENKISTE P., SUBHLOK J., SUTHERLAND D., *ReMoS: A resource monitoring system for network-aware applications*, CMU-CS-97-194, Carnegie Mellon University, December 1997.
- [7] DOVROLIS C., RAMANATHAN P., *A case for Relative Differentiated Services and the Proportional Differentiation Model*, IEEE Network, September/October 1999, s. 26–34.
- [8] DOVROLIS C., STILIADIS D., RAMANATHAN P., *Proportional differentiated services: delay differentiation and packet scheduling*, ACM Communications Review, Vol. 29, No. 4, October 1999, s. 109–120.
- [9] FELDMAN A., REXFORD J., CACERES R., *Efficient policies for carrying web traffic over flow-switched networks*, IEEE/ACM Transactions on Networking, Vol. 6, No. 6, December 1998, s. 673–685.
- [10] FENG Q.C., KANDLUR D.D., SAHA D., SHIN K.G., *Adaptive packet marking for maintaining end-to-end throughput in a Differentiated Services Internet*, IEEE/ACM Transactions on Networking, Vol. 7, No. 5, June 1993, s. 685–697.
- [11] FROST V.S., MELAMED B., *Traffic modelling for telecommunications networks*, IEEE Communications Magazine, Vol. 32, No. 3, March 1994, s. 70–81.
- [12] FULP W.E., REEVES D.S., *Optimal provisioning and pricing of differentiated services using QoS class promotion*, N.C. State University (www.itpapers.com).
- [13] GEORGIDIS L., GUERIN R., PERIS V., SIVARAJAN K., *Efficient network QoS provisioning based on per node traffic shaping*, IEEE/ACM Transactions on Networking, Vol. 4, No. 4, August 1996.
- [14] GŁOWACKI M., *Measurement of QoS parameters of packet-switched Wide Area Networks*, Proceedings of 6th Polish Teletraffic Symposium, Szklarska Poręba 1999, s. 128–142.
- [15] GOLMIE N., NDOUSSE T.D., SU D.H., *A differentiated optical services model for WDM networks*, National Institute of Standards and Technology, 2001 (www.itpapers.com).
- [16] GRIFFITHS J.M., *ISDN explained*, Wiley, Chichester 1992.
- [17] GRZECH A., *Techniki komutacji w podsystemach komunikacyjnych sieci komputerowych*, III polskie sympozjum „Ruch telekomunikacyjny”, t. III-3, Kraków 1996, s. 1–6.

- [18] GRZECH A., *Performance measures of intelligent computer network nodes*, Proceedings 16th World Computer Congress, Vol. 1, IEEE Press 2000, s. 454–462.
- [19] GUERIN R.A., ORDA A., WILLIAMS D., *QoS routing mechanisms and OSPF extensions*, IEEE Global Internet Mini-Conference (Globecom '97), November 1997.
- [20] GUERIN R.A., ORDA A., *QoS routing in networks with inaccurate information: theory and algorithms*, IEEE/ACM Transactions on Networking, Vol. 7, No. 3, June 1999, s. 350–364.
- [21] GUERIN R.A., PERIS V., *Quality-of-Service in packet networks basic mechanisms and directions*, Computer Networks, Vol. 31, No. 3, February 1999, s. 169–179.
- [22] GUERIN R.A., PLA V., *Aggregation and conformance in Differentiated Service networks: A case study*, Computer Communication Review, Vol. 31, No. 1, January 2001, s. 21–32.
- [23] HALSALL F., *Multimedia communications: applications, networks, protocols and standards*, Addison-Wesley, 2000.
- [24] HEGERING H.G., ABECK S., *Integrated network and system management*, Addison-Wesley, Wokingham 1994.
- [25] HURLEY P., LE BOUDEC J.Y., THIRAN P., KARA M., *ABE: providing a low-delay service within best effort*, IEEE Network, May/June 2001, s. 60–69.
- [26] KHASNABISH B, SARACCO R., *Intranets: Technologies, services and management*, IEEE Communication Magazine, October 1997, s. 84–91.
- [27] KOŚCIELNIAK D., *ISDN – Cyfrowe sieci zintegrowane usługowo*, WKŁ, Warszawa 1996.
- [28] KRUNZ M., *Bandwidth allocation strategies for transporting Variable-Bit-Rate video traffic*, IEEE Communications Magazine, January 1999, s. 40–46.
- [29] KUMMAR V.P., LAKSHMAN T.V., STILIADIS D., *Beyond Best Effort: router architectures for the differentiated services of tomorrow's Internet*, IEEE Communications Magazine, May 1988, s. 152–164.
- [30] LEE B.G., KANG M., LEE J., *Broadband telecommunications technology*, Artech House, Boston 1993.
- [31] LORENZ D.H., ORDA A., *QoS routing in networks with uncertain parameters*, IEEE/ACM Transactions on Networking, Vol. 6, No. 6, December 1998, s. 768–778.
- [32] LORENZ D.H., *QoS routing in networks with uncertain parameters*, University of Haifa, 1998.
- [33] MEER H., PULIAFITO A., RICHTER J.P., TOMARCHIO O., *QoS-adaptation by software agents in the presence of defective reservation mechanisms in the Internet*, University of Hamburg, 2000.
- [34] ORDA A., *Routing with end-to-end QoS guarantees in broadband networks*, IEEE/ACM Transactions on Networking, Vol. 7, No. 3, June 1999, s. 365–374.
- [35] OZAKI T., KIM J.B., SUDA T., *Bandwidth-efficient multicast routing for multihop, ad-hoc wireless networks*, California State University, 2001 (www.itpapers.com).
- [36] PACH A. (red.), *Sieci dostępowe dla usług szerokopasmowych*, t. II, Wydawnictwo Fundacji Postępu Telekomunikacji, Kraków 1997.
- [37] PACH A., LASOŃ A., WAJDA K., *Współpraca sieci ATM z innymi systemami telekomunikacyjnymi*, Wydawnictwo Fundacji Postępu Telekomunikacji, Kraków 1995.
- [38] PAPIR Z. (red.), *Sieci z komutacją pakietów – od X.25 do Frame Relay i ATM*, Wydawnictwo Fundacji Postępu Telekomunikacji, Kraków 1997.

- [39] PAPIR Z., *Ruch telekomunikacyjny i przeciążenia sieci pakietowych*, WKŁ, Warszawa 2001.
- [40] PEREKH A.K., GALLAGER R.G., *A generalized processor sharing approach to flow control in integrated services networks: the single-node case*, IEEE/ACM Transactions on Networking, Vol. 1, No. 3, June 1993, s. 344–357.
- [41] PRYCKER M., *Asynchronous Transfer Mode: Solution for Broadband ISDN*, Ellis Horwood, New York 1993.
- [42] RAJAN R., VERMA D., KAMAT S., FELSTAIN E., HERZOG S., *A policy framework for Integrated and Differentiated Services in the Internet*, IEEE Network, September/October 1999, s. 34–41.
- [43] SAITO H., *Call Admission Control in an ATM network using upper bound of Cell Loss Probability*, IEEE Transactions on Communications, Vol. 40, No. 9, September 1992, s. 1512–1521.
- [44] SCHMIT J., KARSTEN M., WOLF L., STEINMETZ R., *Aggregation of guaranteed service flows*, Darmstadt University of Technology, 2001 (www.itpapers.com).
- [45] SIEGEL E., PASSMORE L. D., *Network Quality of Service – What Good Enough?*, July 1997 (www.netrefence.com).
- [46] TABATABAE V., GEORGIADIS L., TASSIULAS L., *QoS provisioning and tracking fluid policies in input queueing switches*, IEEE/ACM Transactions on Networking, Vol. 9, No. 5, October 2001, s. 605–617.
- [47] TEITELBAUM B., HARES S., DUNN L., NEILSON R., NARAYAN V., REICHMEYER F., *Internet2 Qbone building a testbed for Differentiated Services*, IEEE Network, s. 8–16, September/October 1999.
- [48] WAICHEE A.F., *Delay-optimal quorum consensus for distributed systems*, IEEE Transactions on Parallel and Distributed Systems, Vol. 8, No. 1, January 1997, s. 59–69.
- [49] WAJDA K., *Sieci szerokopasmowe*, Wydawnictwo Fundacji Postępu Telekomunikacji, Kraków 1995.
- [50] WAJDA K. (red.), *Budowa sieci komputerowych w technologii ATM*, Wydawnictwo Fundacji Postępu Telekomunikacji, Kraków 1996.
- [51] WAJDA K. (red.), *Wybrane zagadnienia budowy i eksploatacji sieci korporacyjnych*, Wydawnictwo Fundacji Postępu Telekomunikacji, Kraków 1999.
- [52] WARKHEDE P., SURI S., VARGHESE G., *Fast packet classification for two-dimensional conflict-free filters*, Cisco Systems, 2001.
- [53] WANG Z., *Internet QoS: Architectures and mechanisms for Quality of Service*, Academic Press, London 2001.
- [54] WEISS W., *QoS with Differentiated Services*, Bell Labs Technical Journal, October–December 1998, s. 48–62.
- [55] XIAO X., NI L.M., *Internet QoS: A big picture*, IEEE Network, March/April 1999, s. 8–18.
- [56] YEOM I., REDDY A.L.N., *Modelling TCP behaviour in a Differentiated Services network*, IEEE/ACM Transactions on Networking, Vol. 9, No. 1, February 2001, s. 31–46.
- [57] ZHANG L., DEERING S., ESTRIN D., SHENKER S., ZAPPALA D., *RSVP: A new resource Reservation Protocol*, IEEE Network, September 1993, s. 8–18/.
- [58] ZHU K., ZHUANG Y., VINIOTIS Y., *Achieving end-to-end delay bounds by EDF scheduling without traffic shaping*, North Carolina State University, Alcatel, Amber 2001 (www.itpapers.com).

9. KONCEPCJA USŁUG ZINTEGROWANYCH

Realizację koncepcji usług zintegrowanych charakteryzują następujące założenia:
➤ gwarancja jakości usług sieci jest możliwa dzięki rezerwacji zasobów sieci dla każdego z przepływów przyjętych do obsługi w sieci,

➤ efektywne wykorzystanie zasobów sieci jest możliwe przez przydzielanie zasobów elementarnym przepływom pomiędzy źródłami i ujściami sieci, tj. przepływom na najniższym możliwym poziomie granulacji,

➤ dostępność zasobów jest warunkiem koniecznym, ale nie wystarczającym, gwarancji jakości usług sieci.

Przyjęcie założenia o alokacji zasobów do każdego z elementarnych przepływów umożliwi maksymalizację stopnia wykorzystania zasobów sieci za cenę konieczności implementacji w niej rozbudowanego funkcjonalnie i przestrzennie systemu monitorowania i zarządzania każdym z elementarnych przepływów w każdym węźle sieci.

Głównym powodem ograniczonych możliwości implementacji omawianej koncepcji dostarczania jakości usług jest duża i zmienna liczba elementarnych przepływów w sieci.

9.1. PODSTAWOWE ZAŁOŻENIA

Koncepcja usług zintegrowanych opiera się m.in. na założeniu, że uzyskanie gwarancji jakości usług pakietowej sieci komputerowej jest możliwe przez rezerwację zasobów sieci dla każdego z generowanych przepływów.

Rezerwacja zasobów polega na wyborze i ustanowieniu trasy w sieci, umożliwiającej dostęp do żądanych zasobów, przed rozpoczęciem transferu ruchu. Realizacja tego założenia oznacza rezygnację z podstawowej zasady koncepcji sieci datagramowych, zorientowanych na maksymalizację efektywności wykorzystania sieci, że trasy poszczególnych jednostek danych, generowanych w jednym źródle, są różne i niezależne. Jakość usług sieci datagramowych jest konsekwencją obsługi pakietów bez gwarancji jakości (ang. *best-effort*).

Sposoby obsługi jednostek danych bez gwarancji jakości oraz według koncepcji usług zintegrowanych reprezentują różne koncepcje obsługi przepływów: implemen-

towane w sieci mechanizmy obsługi przepływów są w pierwszym przypadku zorientowane na poszczególne jednostki danych przepływające w sieci, w drugim natomiast na poszczególne przepływy w sieci.

Wprowadzenie możliwości rezerwacji tras i ich utrzymywania na czas trwania aktywności źródła przypomina realizację usług sieci z komutacją łączy. Z tego punktu widzenia realizację koncepcji usług zintegrowanych można interpretować jako łączenie zalet techniki komutacji pakietów w formie datagramów oraz korzyści wynikających ze stosowania techniki komutacji łączy. Celem wprowadzanego w ten sposób nadmiaru organizacyjnego w sieci jest regramentacja jej zasobów przed przyjęciem do obsługi w sieci ruchu generowanego w źródłach. Rezerwacja zasobów oznacza, że rozwiązywanie konfliktów dostępu do zasobów sieci na etapie obsługi ruchu przyjętego, charakterystyczne dla sieci pakietowych, jest zastępowane rozwiązaniem zadania współzawodnictwa źródeł ruchu o zasoby na etapie ich rezerwacji, tzn. na brzegu sieci.

Podstawowym efektem rezerwacji zasobów sieciowych dla określonego wywołania jest możliwość gwarantowania żądanych lub założonych wartości parametrów jakości usług sieci, przede wszystkim opóźnienia wnoszonego przez sieć (wartości średniej, wartości maksymalnej i rozkładu zmienności wartości opóźnienia) oraz wartości prawdopodobieństwa strat. Zmiany wartości prawdopodobieństwa strat w aplikacjach wrażliwych na straty przekładają się na zmiany opóźnienia wynikające z konieczności retransmisji traconych jednostek danych. Z kolei zmiany wartości opóźnienia jednostek danych w aplikacjach wrażliwych na opóźnienia wpływają na zmiany wartości prawdopodobieństwa strat z powodu przekraczania górnego ograniczenia czasu oczekiwania aplikacji na jednostki danych.

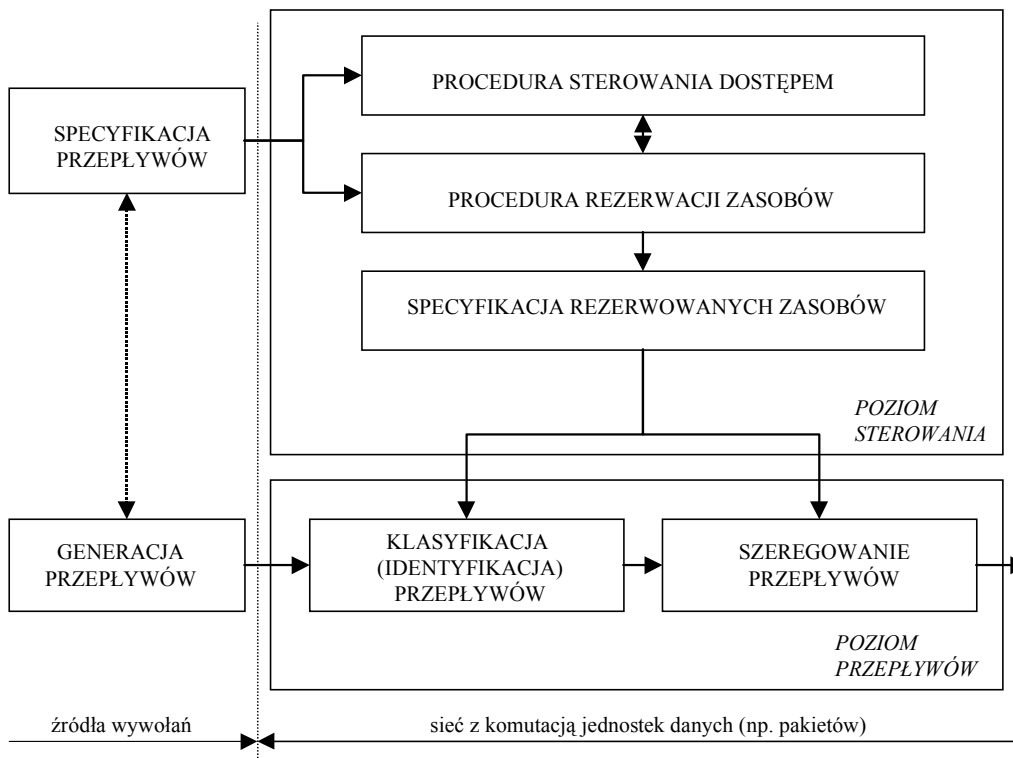
Jednym z powodów atrakcyjności koncepcji komutacji łączy jest przyjmowanie założenia o takich zmianach kosztów instalacji i udostępniania zasobów sieci (zwłaszcza pojemności), które pozwalają na rezygnację z optymalizacji ich wykorzystania, a tym samym na rezygnację ze złożonych mechanizmów wyznaczania tras (ang. *routing*) i sterowania przepływem (ang. *flow control*).

Obsługa wywołania w rozwiązaniach z komutacją polega na realizacji kilkietapowej procedury:

- rezerwacji zasobów w wyniku ustanawiania trasy dla wywołania; warunkiem przyjęcia wywołania do obsługi jest skuteczna rezerwacja zasobów w celu zagwarantowania żądanej jakości usług określonej w kontrakcie użytkownika i sieci,
- obsłudze ruchu generowanego w ramach przyjętego wywołania,
- zwalnianiu trasy po zakończeniu obsługi wywołania.

9.2. MODEL FUNKCJONALNY OBSŁUGI WYWOŁAŃ

Założenie o etapowej realizacji procedury udostępniania zasobów sieci dla przepływów w sieci z implementacją koncepcji usług zintegrowanych oznacza istnienie dwóch poziomów w hierarchicznym modelu obsługi wywołań (rysunek 9.1):



Rysunek 9.1. Model funkcjonalny sieci realizującej koncepcję usług zintegrowanych

1. Poziomu sterowania (ang. *control plane*), obejmującego:

a) przygotowanie żądania ustanowienia rezerwacji (ang. *reservation setup request*) na podstawie:

- charakterystyk wywołania i specyfikacji przepływu (ang. *flow specification*) dla wywołania żądającego dostępu do zasobów sieci,
- charakterystyk stosowanej procedury sterowania dostępem wywołań (ang. *admission control*);

b) obsługę żądania ustanowienia rezerwacji zasobów w sieci, polegającą na:

- sprawdzeniu możliwości wyboru trasy spełniającej wymagania wywołania i obsługującej to wywołanie procedury sterowania dostępem,
- wyznaczeniu optymalnej trasy (optymalizacja jedno- lub wielokryterialna),
- rezerwacji zasobów na wyznaczonej trasie,

c) konfigurację procedury sterowania dostępem, polegającą na wyznaczeniu wartości parametrów procedury sterowania na potrzeby monitorowania i pomiaru udostępnianych zasobów sieciowych w dwóch alternatywnych wersjach: parametrycznej lub pomiarowej,

d) konfigurację parametrów modułów identyfikacji (w ogólnym przypadku klasyfikacji) jednostek danych przepływów (ang. *flow identification*) i szeregowania jednostek danych przepływu (ang. *flow scheduling*).

2. Poziomu przepływu (ang. *data plane*), którego zadaniem jest realizacja procedur:

a) identyfikacji przepływów, polegającej na klasyfikacji (filtrowaniu) jednostek danych należących do obsługiwanego wywołania,

b) szeregowania przepływów jednostek danych, polegającego na wymuszaniu alokacji zarezerwowanych zasobów do obsługiwanego przepływu.

Zadaniem modułów zlokalizowanych na nadrzędnym poziomie sterowania jest ustalenie wartości parametrów procedur klasyfikacji parametrów ruchowych dla wywołania przyjętego do obsługi i rezerwacja zasobów, zadaniem natomiast modułów wykorzystywanych na podrzędnym poziomie jest realizacja obsługi przepływów na warunkach ustalonych na poziomie nadrzędnym. Różnice funkcjonalności węzłów dostępowych (końcowych) oraz węzłów pośrednich (routerów) w architekturach realizujących koncepcję usług zintegrowanych wynikają z tego, że zakres i złożoność zadań sterowania przyjmowaniem wywołań, klasyfikacji jednostek danych przepływów, wyboru tras, rezerwacji zasobów itp. są znacznie większe w węzłach końcowych niż w węzłach pośrednich.

9.2.1. CHARAKTERYSTYKI WYWOŁAŃ

Charakterystyka wywołania żądającego dostępu do sieci obejmuje m.in. opisy ruchu generowanego przez źródło i wymagań dotyczących żądanych ilości zasobów (formułowanych bezpośrednio lub pośrednio). Wartości parametrów charakteryzujących wywołanie są wykorzystywane do podejmowania decyzji o przyjęciu lub odrzuceniu wywołania do obsługi w sieci. Decyzja jest podejmowana przez odpowiedni algorytm sterowania dostępem na etapie rezerwacji zasobów.

Specyfikacja przepływu jest zbiorem warunków kontraktu dotyczącego ilości ruchu, który źródło zamierza przesłać, oraz – żądanej od sieci i dostarczanej przez nią – jakości obsługi tego ruchu. Specyfikacja przepływu jednoznacznie określa także warunki naruszenia kontraktu. Zapobieganie przypadkom naruszenia kontraktów jest zadaniem procedur nadzorowania zgodności ruchu rzeczywistego z deklarowanym w specyfikacji źródła (ang. *policing procedure*). Przypadek naruszenia kontraktu przez źródło oznacza zwykle, że jakość obsługi ruchu nadmiarowego nie jest gwarantowana, jeżeli w ogóle ruch nadmiarowy jest obsługiwany przez sieć.

Specyfikacja przepływu może być różna, przy czym pomiędzy dokładnością specyfikacji a możliwościami jej wykorzystania występują następujące związki:

1. Specyfikację ilościową przepływu determinuje aplikacja generująca przepływ. Opis przepływów generowanych przez różne aplikacje wymaga korzystania z różnych charakterystyk ilościowych.

2. Specyfikacja żądanej jakości usług dla przepływu zależy od aplikacji. Specyfikację taką determinuje przede wszystkim charakter wrażliwości aplikacji na wartości różnych wskaźników jakości działania sieci.

3. Zwiększenie liczby różnych charakterystyk przepływów (generowanych przez różne aplikacje) stosowanych w sieciach oraz parametrów opisujących te charakterystyki powoduje wzrost złożoności przetwarzania realizowanego przez algorytmy sterowania dostępem, procedury rezerwacji, procedury nadzorowania zgodności ruchu oraz procedury identyfikacji i szeregowania przepływów.

Do podstawowych parametrów ruchu i żądanej jakości obsługi, wykorzystywanych w specyfikacjach przepływów w koncepcji usług zintegrowanych, zalicza się:

1. Parametry ilościowe przepływów:

➤ szybkość szczytowa jednostek danych (ang. *peak rate*) – maksymalna szybkość pakietów, z jaką źródło może generować pakiety,

➤ szybkość średnia (ang. *sustainable rate*) – szybkość transferu uśredniona w danym przedziale czasu,

➤ rozmiar paczki (ang. *burst size*) – maksymalna liczba jednostek danych (np. pakietów, komórek) wprowadzanych do sieci z szybkością szczytową.

2. Parametry jakościowe przepływów:

➤ minimalna pojemność (ang. *minimum bandwidth*) – minimalna wartość pojemności (pasma) łącza transmisyjnego (trasy) wymagana do obsługi przepływu generowanego w ramach wywołania,

➤ opóźnienie (ang. *delay*) – opóźnienie jednostek danych wnoszone przez sieć i wyrażane albo jako średnie opóźnienie (ang. *average delay*), albo jako maksymalne opóźnienie (ang. *worst-case-delay*),

➤ zmienność opóźnienia (ang. *delay jitter*) – maksymalna różnica pomiędzy największym i najmniejszym opóźnieniami jednostek danych wnoszonymi przez sieć,

➤ prawdopodobieństwo strat (ang. *loss rate*) – udział jednostek danych traconych w ogólnej liczbie obsługiwanych jednostek danych.

9.2.2. STEROWANIE DOSTĘPEM WYWOŁAŃ

Istotne znaczenie dla efektywności obsługi transferu jednostek danych przez wcześniej zarezerwowane zasoby sieci ma implementowana na brzegu sieci procedura sterowania dostępem wywołań. Podstawowym zadaniem procedury sterowania dostępem, niezależnie od przyjętej wersji, jest podejmowanie decyzji o przyjęciu lub odrzuceniu nowego wywołania.

Procedury sterowania dostępem, które determinują i są determinowane przez przyjęty sposób charakteryzowania źródła ruchu, spełniają dwie zasadnicze funkcje:

➤ określają sposób opisu i wartości parametrów opisu źródła, na podstawie których są rezerwowane zasoby,

➤ określają sposoby monitorowania i pomiaru zasobów udostępnianych na potrzeby obsługi wywołania.

Spełnianie wymienionych funkcji powoduje, że omawiane procedury stosowane są zarówno na etapie rezerwacji zasobów dla wywołania, jak i na etapie zarządzania obsługą ruchu generowanego w ramach wywołania przyjętego do obsługi w sieci.

Procedury sterowania dostępem wykorzystywane w zadaniach rezerwacji zasobów sieciowych mogą być dwojakiego rodzaju:

➤ parametryczne (ang. *parameter-based admission control*) – stosowane w przypadkach (rzadszych), gdy możliwe jest precyzyjne scharakteryzowanie źródła przepływów, przede wszystkim do obsługi przepływów statycznych,

➤ pomiarowe (ang. *measurement-based admission control*) – stosowane w przypadkach (częstszych), gdy decyzje o dostępie są wyznaczane na podstawie obserwacji generowanego ruchu, w szczególności do obsługi przepływów dynamicznych.

Niezależnie od rodzaju procedury sterowania dostępem wywołań rezerwacja zasobów w sieci może być realizowana w jednej z dwóch wersji:

1. Zwiłokrotnianie statyczne, gdzie ilość zasobów rezerwowanych dla wywołania żądającego dostępu do sieci jest nie mniejsza od ilości żądanych zasobów. Rozwiązanie gwarantuje jakość obsługi ruchu za cenę zmniejszenia efektywności wykorzystania zasobów.

2. Zwiłokrotnianie statystyczne, gdzie sumaryczna ilość zasobów rezerwowanych dla obsługiwanych wywołań może być większa od ilości zasobów dostępnych w sieci. Rozwiązanie to zwiększa efektywność wykorzystania zasobów za cenę obniżenia gwarancji jakości usług.

Algorytmy sterowania dostępem wywołań, ze względu na sposób podziału zasobów, a zwłaszcza ze względu na podział pojemności wyjściowego kanału transmisyjnego, są zaliczane do następujących grup algorytmów:

➤ prostego sumowania (ang. *simple sum algorithm*) – nowe wywołanie jest akceptowane tylko wtedy, gdy suma żądanych zasobów dla wszystkich obsługiwanych i nowego wywołania jest mniejsza od dostępnych zasobów (algorytmy konserwatywne),

➤ mierzonej sumy (ang. *measured sum algorithm*) – nowe wywołanie jest akceptowane wtedy, gdy suma mierzonego aktualnego zapotrzebowania obsługiwanych wywołań na zasoby i zapotrzebowania żądanego przez nowe wywołanie nie przekracza ilości dostępnych zasobów (suma mierzonego, aktualnego zapotrzebowania na zasoby jest zwykle mniejsza od sumy żądanych zasobów),

➤ obszarów akceptacji (ang. *acceptance region algorithm*) – akceptacja nowego wywołania jest podejmowana na podstawie rozwiązania zadania jednoczesnej optymalizacji stopnia wykorzystania zasobów i prawdopodobieństwa straty jednostek danych, tj. prawdopodobieństwa odrzucenia jednostek danych generowanych w ramach zaakceptowanych wywołań,

➤ równoważnych zasobów (ang. *equivalent resource algorithm*) – warunkiem akceptacji nowego wywołania jest wartość prawdopodobieństwa, że sumaryczne zapotrzebowanie na zasoby po przyjęciu nowego wywołania jest większe od dostępnych zasobów i jest mniejsze od założonej wartości prawdopodobieństwa.

9.2.3. IDENTYFIKACJA JEDNOSTEK DANYCH PRZEPEŁYWU

Zadaniem procedur identyfikacji jednostek danych przepływów jest rozpoznawanie i klasyfikacja jednostek danych obsługiwanych w węzłach sieci, ze względu na przynależność do poszczególnych wywołań oraz na uprawnienia do korzystania z zasobów rezerwowanych dla tych wywołań. Identyfikacja jednostek danych przepływu jest szczególnym przypadkiem zadania klasyfikacji jednostek danych przepływu. Zadanie klasyfikacji jest zadaniem alokacji zasobów dla przepływów, zadanie identyfikacji dotyczy natomiast rozstrzygnięcia o przynależności jednostki do jednego z obsługiwanych przepływów.

Szczegółowe zadania identyfikacji jednostek danych przepływu są następujące:

➤ klasyfikacja każdej generowanej jednostki danych i wywołania (przepływu), do którego jednostka danych należy,

➤ sprawdzanie zgodności wartości parametrów rzeczywistego przepływu danego wywołania z wartościami parametrów ustalonych w czasie akceptacji wywołania.

Duża sprawność procedur identyfikacji jednostek danych przepływów jest warunkiem jakości obsługi zaakceptowanych wywołań. Jednostki danych generowane w ramach wywołań poza warunkami kontraktów określonych na etapie przyjmowania wywołania do obsługi, są traktowane w sposób specjalny, tzn. mogą być odrzucane lub przyjmowane do obsługi w sieci bez gwarancji jakości obsługi.

9.2.4. SZEREGOWANIE JEDNOSTEK DANYCH PRZEPEŁYWÓW

Zadaniem procedur szeregowania jednostek danych przepływów jest wymuszenie alokacji zarezerwowanych zasobów. Implementowany sposób alokacji zasobów determinuje bezpośrednio opóźnienie jednostek danych wnoszone przez sieć oraz zasady ich odrzucania wówczas, gdy stopień wykorzystania zasobów jest bliski ich nasycenia.

Podstawowym zadaniem szeregowania jest wyznaczanie kolejności obsługi jednostek danych współdzielących zasoby udostępniane w sieci dla poszczególnych wywołań i generowanych przepływów.

9.3. MODELE USŁUG W KONCEPCJI USŁUG ZINTEGROWANYCH

Realizacja koncepcji usług zintegrowanych polega na stosowaniu dwóch modeli usług: gwarantowanej usługi GS (ang. *Guaranteed Service*) oraz sterowanego obciążenia CLS (ang. *Controlled Load Service*). Szczególnym przypadkiem modelu sterowanego obciążenia jest obsługa bez gwarancji jakości usług.

9.3.1. MODEL GWARANTOWANEJ USŁUGI

9.3.1.1. CHARAKTERYSTYKA GWARANCJI OPÓŹNIENIA

Model gwarantowanej usługi GS jest odpowiedni do obsługi ruchu generowanego przez aplikacje, których jakość zależy od:

- dysponowania w sieci trasą (łączem) o stałej pojemności (pasmie), gwarantującej utrzymanie założonej wartości prawdopodobieństwa strat jednostek danych,
- wnoszenia przez sieć opóźnień jednostek danych nie większych od ustalonej wartości krytycznej.

Dostarczanie gwarantowanej usługi przez sieć oznacza, że dla akceptowanego wywołania i przepływu jednostek danych związanego z tym wywołaniem:

- przydzielane są zasoby w ilościach zapewniających stały, w czasie trwania wywołania, dostęp do określonej, definiowanej jako minimalna, pojemności sieci,
- konfigurowane są procedury sterowania dostępem i szeregowania jednostek danych przepływu, gwarantujące, że wartości opóźnienia jednostek danych wnoszone przez sieć są ograniczone z góry.

Podane warunki, tzn. minimalna gwarantowana pojemność i gwarantowane maksymalne opóźnienie nie są zamiennie, tzn.:

- dostęp do stałej pojemności łącza (trasy) nie gwarantuje utrzymania opóźnienia poniżej określonej wartości,
- stała dostępność określonej pojemności łącza (trasy) nie jest warunkiem wystarczającym do tego, aby wnoszone przez sieć opóźnienia jednostek danych były nie większe od zadanej wartości.

Dostarczanie usługi sieciowej, polegającej na gwarancji:

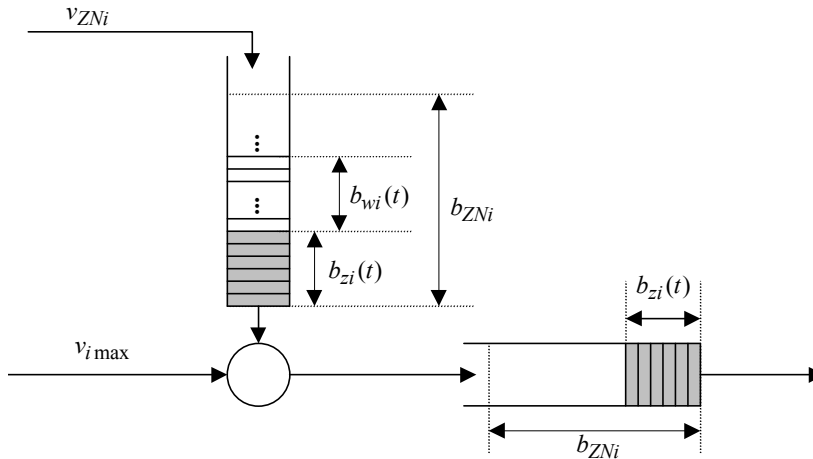
- dostępu do ustalonej minimalnej pojemności,
- opóźnienia wszystkich jednostek danych przepływu nie większego od ustalonego maksymalnego opóźnienia,

oznacza konieczność użycia konserwatywnych sposobów rezerwacji zasobów sieci, w ilościach niezbędnych do obsługi tzw. najgorszego przypadku (ang. *worst case*). Ten sposób udostępniania zasobów, jakkolwiek zapewnia możliwość uzyskania żądanej jakości usług, powoduje jednak niewielką efektywność wykorzystania zasobów sieci, a tym samym wzrost cen usług.

Gwarantowanie przez sieć maksymalnej wartości opóźnienia każdej z jednostek danych przepływu jest możliwe tylko przez ograniczenie z góry szybkości ich wprowadzania do sieci. Konieczność stosowania regulatora szybkości wprowadzania jednostek danych przepływu do sieci oznacza, że szybkość ich wprowadzania, nie większa od szybkości generowania jednostek danych w źródle przepływu, jest wielkością sterowaną. Stosowanie mechanizmu ograniczającego z góry szybkość wprowadzania jednostek danych do sieci odpowiada implementacji pomiędzy źródłem przepływu a siecią sterownika ograniczającego i „wygładzającego” szybkość nadawania jednostek danych.

9.3.1.2. STEROWANIE OPÓŹNIENIEM W MODELU GWARANTOWANEJ USŁUGI

Jednym z prostszych modeli gwarantowanej usługi, tzn. udostępniania generowanemu przepływowi łącza o stałej pojemności i gwarantowania górnego ograniczenia opóźnienia jednostek danych, jest model przepływowy ze sterownikiem przepływu, opartym na algorytmie ciekącego wiadra (ang. *leaky bucket*). Zadaniem sterownika przepływu jest zapobieganie wprowadzaniu do sieci, w określonym przedziale czasu, większej od dopuszczalnej liczby jednostek danych. Jeżeli szybkość obsługi jednostek danych jest stała (co wynika z udostępniania łącza o stałej pojemności), to ograniczenie liczby jednostek danych wprowadzanych do sieci w przedziale czasu o zadanej długości jest warunkiem wystarczającym istnienia górnego ograniczenia opóźnienia każdej z jednostek danych.



Rysunek 9.2. Sterowanie szybkością wprowadzania jednostek danych do sieci z wykorzystaniem algorytmu ciekącego wiadra

Do szacowania opóźnień jednostek danych przepływu, wnoszonych przez system obsługi ze sterownikiem opartym na algorytmie ciekącego wiadra, stosuje się następujące parametry (rysunek 9.2):

1. Źródła i -tego przepływu:
 - szczytowa szybkość nadawania jednostek danych (o stałej, jednakowej długości) przez źródło: $v_{i \max}$ (jednostek/s).
2. Sterownika i -tego przepływu:
 - pojemność bufora znaczników i -tego przepływu: b_{ZNi} ,
 - liczba buforów zajętych w pamięci buforowej znaczników i -tego przepływu w chwili t , tzn. $b_{zi}(t)$, gdzie $b_{zi}(t) \leq b_{ZNi}$,
 - liczba buforów wolnych w pamięci buforowej znaczników i -tego przepływu w chwili t , tzn. $b_{wi}(t)$, gdzie: $b_{wi}(t) \leq b_{ZNi}(t)$ i $b_{zi}(t) \leq b_{ZNi} - b_{zi}(t)$,

➤ szybkość napływu znaczników do bufora znaczników i -tego przepływu: v_{ZNi} (znaczników/s).

3. Kanału transmisyjnego (stanowiska obsługi przepływu):

➤ stała szybkość obsługi jednostek danych (pojemność kanału transmisyjnego) i -tego przepływu: C_i (jednostek/s).

W sterowniku działającym zgodnie z algorytmem ciekącego wiadra liczba buforowanych jednostek danych oczekujących na transmisję jest ograniczona pojemnością pamięci buforowej znaczników b_{ZNi} oraz szybkością wprowadzania nowych znaczników v_{ZNi} , spełniającą ograniczenie: $v_{ZNi} < C_i$. Jeżeli wszystkie jednostki danych mają jednakową długość, to maksymalne opóźnienie jednostek danych i -tego przepływu, oznaczone przez $d_{i\max}$, wnoszone przez system obsługi z omawianym sterownikiem, jest sumą trzech składowych:

➤ d_{iq} – czasu oczekiwania jednostki danych i -tego przepływu na obsługę, zależnego od wartości parametrów źródła, sterownika i stanowiska obsługi,

➤ $d_{i\text{pac}}$ – opóźnienia paketyzacji, zależnego od długości jednostki danych i szybkości obsługi,

➤ $d_{i\text{pro}}$ – opóźnienia przetwarzania, zależnego od procedur identyfikacji ruchu, i określone jest równaniem:

$$d_{i\max} = d_{iq\max} + d_{i\text{pac}} + d_{i\text{pro}} = d_{iq\max} + \frac{1}{C_i} + d_{i\text{pro}},$$

w którym maksymalny czas oczekiwania jednostki danych i -tego przepływu na obsługę $d_{iq\max}$ jest dany wyrażeniem:

$$d_{iq\max} = \begin{cases} \frac{b_{ZNi}(v_{i\max} - C_i)}{C_i(v_{i\max} - v_{ZNi})} & \text{gdy } v_{i\max} > C_i \geq v_{ZNi} \\ 0 & \text{gdy } v_{ZNi} \leq v_{i\max} \leq C_i \end{cases}.$$

Maksymalny czas oczekiwania jednostki danych i -tego przepływu na obsługę jest czasem oczekiwania na obsługę wtedy, gdy – w chwili rozpoczęcia obsługi jednostek danych przez łącze o stałej pojemności C_i – bufor znaczników jest pełny.

Wartość $d_{iq\max}$ jest zawsze mniejsza od czasu oczekiwania w tzw. najgorszym przypadku, tzn. wtedy, gdy w chwili t :

➤ szybkość wprowadzania jednostek danych do sterownika jest znacznie większa od szybkości obsługi jednostek danych, tzn. gdy $v_{i\max} \gg C_i$,

➤ liczba znaczników wolnych w pamięci buforowej znaczników jest równa pojemności bufora znaczników, tzn. gdy $b_{wi}(t) = b_{ZNi}$ i $b_{-i}(t) = 0$,

➤ liczba jednostek danych oczekujących w kolejce na obsługę przez łącze o stałej pojemności C_i jest równa zeru.

Wystąpienie najgorszego przypadku oznacza, że – ze względu na dużą różnicę pomiędzy szybkością generowania jednostek danych a szybkością obsługi jednostek

danych – wszystkie znaczniki zostaną zajęte w dowolnie krótkim przedziale czasu δt ($\delta t \rightarrow 0$), tzn. w chwili $t + \delta t$ długość kolejki jednostek danych jest równa pojemności pamięci buforowej znaczników. Czas oczekiwania jednostki danych na obsługę, zajmującą ostatnie miejsce w kolejce, jest górnym ograniczeniem opóźnienia wnoszonego przez omawiany system obsługi; jest nie większy od czasu obsługi b_{ZNi} jednostek danych przez łącze o stałej pojemności C_i , tzn.:

$$d_{iq\max} < \frac{b_{ZNi}}{C_i}.$$

Zależność górnego ograniczenia opóźnienia jednostek danych, wnoszonego przez omawiany system obsługi, tylko od pojemności pamięci buforowej znaczników i pojemności łącza (trasy) oznacza, że sterowanie opóźnieniem odbywa się przez zmianę wartości parametrów algorytmu ciekącego wiadra i stanowiska obsługi.

9.3.1.3. ILOŚĆ RUCHU I OPÓŹNIENIA W OBSŁUDZE PRZEPIYWU

Model sterownika z algorytmem ciekącego wiadra może być użyty do szacowania wartości parametrów charakteryzujących stan wykorzystania zasobów sieci, w tym średniego i maksymalnego opóźnienia wnoszonego przez sterownik oraz zaległości (ang. *backlogged*) w obsłudze przepływu, wynikających z proporcji ilości przydzielonych i żądanych zasobów. Jeżeli $a_i(\tau, t)$ oznacza ilość ruchu i -tego przepływu, wyrowadzaną ze sterownika do pamięci buforowej jednostek danych oczekujących na obsługę w sieci, w przedziale czasu $(\tau, t]$, to:

$$a_i(\tau, t) \leq \min\{(t - \tau)C_i, b_{ZNi} + v_{ZNi}(t - \tau)\},$$

a i -ty przepływ jest (b_{ZNi}, v_{ZNi}, C_i) -zgodnym przepływem.

Funkcja $a_i(0, t)$ jest ciągłą i niemalejącą funkcją czasu t . Założenie, że $a_i(\tau, t)$ jest ilością ruchu, oznacza przyjęcie przepływowego modelu ruchu, tzn. wartość $a_i(0, t)$ jest ciągłą aproksymacją dyskretnej liczby jednostek danych i -tego przepływu, wyrowadzanych ze sterownika w przedziale czasu $(0, t]$.

Ograniczenia szybkości wyrowadzania jednostek danych i -tego przepływu ze sterownika do sieci, dotyczą:

- średniej szybkości jednostek danych przepływu; ograniczeniem jest szybkość napływu znaczników (v_{ZNi}),
- szybkości szczytowej jednostek danych przepływu; ograniczeniem jest stała szybkość obsługi jednostek danych (C_i),
- wybuchowości źródła przepływu; ograniczeniem jest pojemność pamięci buforowej znaczników (b_{ZNi}).

Ilość ruchu i -tego przepływu, wyrowadzaną ze sterownika do sieci w przedziale czasu $(\tau, t]$, można wyrazić:

- liczbą znaczników wprowadzonych do pamięci buforowej znaczników w przedziale czasu $(\tau, t]$,

➤ liczbą wolnych znaczników w pamięci buforowej znaczników na początku i na końcu przedziału czasu $(\tau, t]$.

Jeżeli $b_i(t)$ oznacza całkowitą liczbę znaczników przyjętych do pamięci buforowej znaczników w przedziale czasu $(0, t]$ i liczba ta nie zawiera znaczników znajdujących się w pamięci buforowej znaczników w chwili początkowej i znaczników nieprzyjętych z powodu przepełnienia pamięci buforowej, to spełniony jest warunek:

$$b_i(t) = \min_{0 \leq \tau \leq t} \{a_i(0, \tau) + v_{ZNi}(t - \tau)\},$$

co oznacza, że:

$$b_i(t) - b_i(\tau) \leq v_{ZNi}(t - \tau).$$

Liczba wszystkich znaczników w pamięci buforowej, w dowolnej chwili t dla i -tego przepływu jest sumą liczb znaczników wolnych $b_{wi}(t)$ i znaczników zajętych $b_{zi}(t)$:

$$(b_{wi}(t) + b_{zi}(t) \leq b_{ZNi}).$$

Liczba znaczników zajętych jest równa liczbie jednostek danych przyjętych do obsługi i oczekujących w pamięci buforowej na obsługę w pamięci buforowej jednostek danych.

Liczba znaczników wolnych w pamięci buforowej znaczników dla i -tego przepływu w dowolnej chwili t jest równa sumie liczby znaczników w pamięci buforowej znaczników w chwili początkowej (dla uproszczenia liczba znaczników w pamięci buforowej w chwili początkowej jest równa pojemności pamięci b_{ZNi}) i całkowitej liczby znaczników przyjętych do pamięci buforowej znaczników w przedziale $(0, t]$ pomniejszonej o liczbę znaczników wykorzystanych w przedziale czasu $(0, t]$ (ilość ruchu wyprowadzonego przez sterownik do sieci w przedziale czasu $(0, t]$):

$$b_{wi}(t) = b_{ZNi} + b_i(t) - a_i(0, t).$$

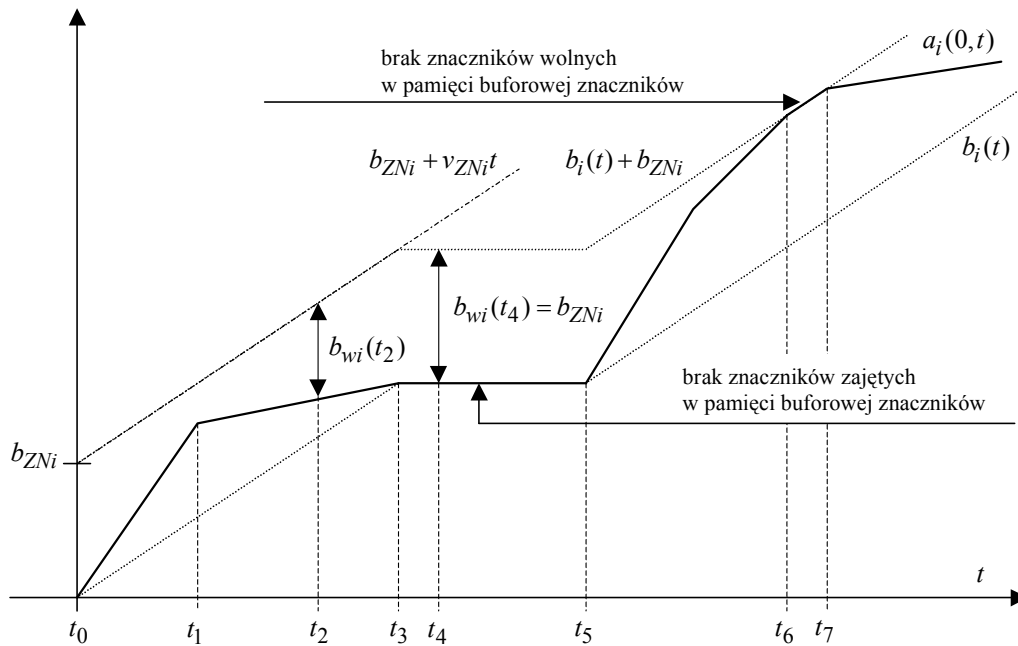
Z zależności tej wynika, że ilość ruchu i -tego przepływu, wyprowadzonego ze sterownika w przedziale czasu $(\tau, t]$ można wyrazić nierównością (rysunek 9.3):

$$\begin{aligned} a_i(\tau, t) = a_i(0, t) - a_i(0, \tau) &= [b_{ZNi} + b_i(t) - b_{wi}(t)] - [b_{ZNi} + b_i(\tau) - b_{wi}(\tau)] \leq \\ &\leq b_{wi}(\tau) + v_{ZNi}(t - \tau) - b_{wi}(t). \end{aligned}$$

Generowanie jednostek danych i -tego przepływu rozpoczyna się w chwili t_0 , kiedy liczba znaczników wolnych jest równa pojemności pamięci buforowej znaczników b_{ZNi} .

W przedziale (t_0, t_1) szybkość generowania jednostek danych i -tego przepływu przez źródło jest większa od szybkości generowania znaczników – liczba wolnych znaczników w pamięci buforowej maleje.

W przedziale (t_1, t_3) szybkość generowania jednostek danych i -tego przepływu przez źródło jest mniejsza od szybkości generowania znaczników – liczba wolnych znaczników w pamięci buforowej rośnie.



Rysunek 9.3. Zależności pomiędzy liczbą znaczników wolnych, całkowitą liczbą znaczników przyjętych i ilością ruchu wyprowadzonego ze sterownika

W przedziale (t_3, t_5) źródło nie generuje nowych jednostek danych – liczba znaczników wolnych jest równa pojemności pamięci buforowej znaczników i nowe znaczniki są odrzucane z powodu przepełnienia pamięci buforowej.

W chwili t_5 źródło ponownie rozpoczyna generowanie jednostek danych – liczba znaczników wolnych zmniejsza się. W przedziale (t_6, t_7) szybkość napływu jednostek danych jest ograniczona szybkością napływu znaczników.

Maksymalna ilość ruchu, wyprowadzana przez sterownik w przedziale $(0, t]$, jest nie większa od sumy pojemności pamięci buforowej znaczników b_{ZNi} i liczby znaczników generowanych w tym przedziale czasu $v_{ZNi}t$, tzn.:

$$a_i(0, t) \leq b_{ZNi} + v_{ZNi}t, \text{ gdy } tC_i \geq b_{ZNi} + v_{ZNi}t.$$

W takim przypadku średnia szybkość generowania jednostek danych i -tego przepływu przez źródło:

$$\bar{v}_i = v_{ZNi} + \frac{b_{ZNi}}{t}.$$

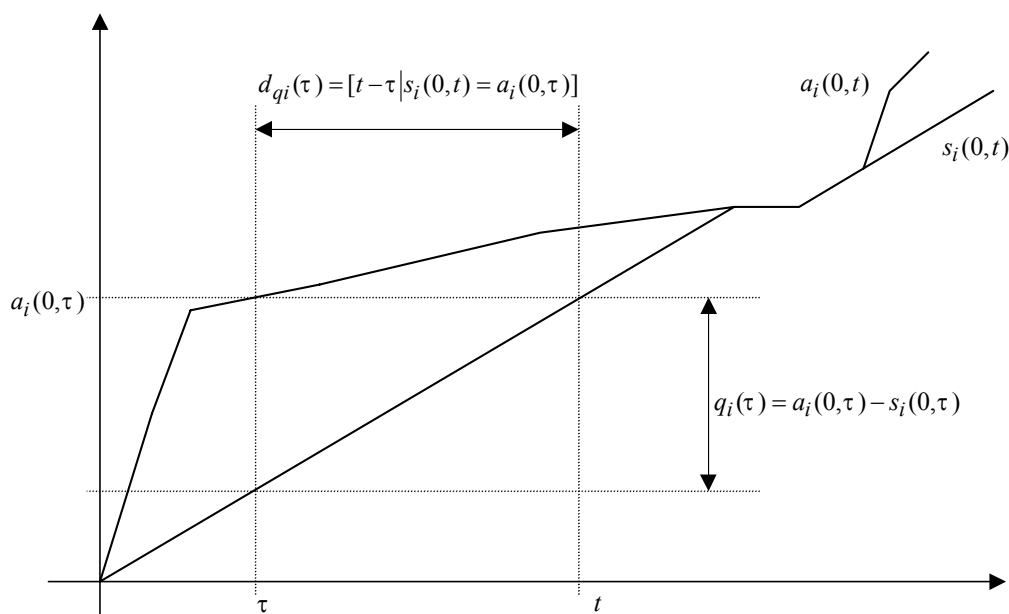
Opóźnienia jednostek danych i -tego przepływu w chwili czasu t , wnoszone przez działanie sterownika, zależą od różnicy pomiędzy ilością ruchu wyprowadzonego przez sterownik do chwili czasu t i ilości ruchu obsłużonego przez sieć do chwili czasu t .

Jeżeli przez $s_i(0, t)$ oznaczyć ilość ruchu i -tego przepływu, wyprowadzonego ze sterownika i obsłużonego przez sieć w przedziale czasu $(0, t]$, to możliwe jest zdefiniowanie dwóch miar jakości obsługi generowanego ruchu:

- zaległości w obsłudze ruchu i -tego przepływu,
- opóźnienia ruchu wnoszonego przez sieć.

Zaległość w obsłudze ruchu i -tego przepływu w chwili τ , oznaczona przez $q_i(\tau)$, jest definiowana jako różnica ilości ruchu wyprowadzonego ze sterownika $a_i(0, \tau)$ i ilości ruchu obsłużonego przez sieć $s_i(0, \tau)$ w przedziale czasu $(0, \tau]$ (rysunek 9.4):

$$q_i(\tau) = a_i(0, \tau) - s_i(0, \tau).$$



Rysunek 9.4. Zaległości i opóźnienia w obsłudze ruchu

Maksymalna wartość zaległości w obsłudze ruchu i -tego przepływu jest ograniczona z góry maksymalną długością kolejki jednostek danych oczekujących na obsługę w sieci, tj. pojemnością pamięci buforowej znaczników:

$$q_{i\max}(\tau) = \max[a_i(0, \tau) - s_i(0, \tau)] = b_{ZNi}.$$

Występowanie zaległości w obsłudze ruchu i -tego przepływu w chwili τ oznacza konieczność oczekiwania na obsługę przez sieć, tzn. powstanie zaległości oznacza

opóźnienie ruchu i -tego przepływu. Opóźnienie ruchu i -tego przepływu w chwili τ , oznaczone przez $d_{qi}(\tau)$, jest definiowane jako długość przedziału czasu potrzebnego do likwidacji powstałej zaległości:

$$d_{qi}(\tau) = \inf\{t \geq \tau : s_i(0, t) = a_i(0, \tau)\} - \tau .$$

Miarą opóźnienia obsługi ruchu i -tego przepływu $d_{qi}(\tau)$, w chwili τ jest odległość w poziomie wykresów funkcji $a_i(0, t)$ i $s_i(0, t)$ dla wartości rzędnej $a_i(0, \tau)$ (rysunek 9.4). Długość tego przedziału czasu jest równa, z dokładnością do różnicy pomiędzy modelem dyskretnym i przepływowym ruchu i -tego przepływu, czasowi oczekiwania na obsługę ostatniej jednostki danych i -tego przepływu wyprowadzonego ze sterownika i oczekującego na obsługę przez sieć.

Jeżeli maksymalna wartość zaległości w obsłudze ruchu i -tego przepływu jest równa pojemności pamięci buforowej znaczników b_{ZNi} , a jednostki danych oczekujące w kolejce są obsługiwane ze stałą szybkością C_i , to maksymalna wartość opóźnienia:

$$d_{qi \max} = \frac{b_{ZNi}}{C_i} .$$

9.3.1.4. FORMOWANIE PRZEPLYWU W MODELU GWARANTOWANEJ USŁUGI

Sterownik realizujący algorytm ciekącego wiadra, zlokalizowany pomiędzy źródłem przepływu a siecią, pełni jednocześnie dwie, wzajemnie zależne, funkcje:

1. Bezpośrednio ogranicza z góry liczbę jednostek danych przepływu wprowadzanych w przedziale czasu o ustalonej długości. Liczba ta zależy od wartości dwóch parametrów algorytmu ciekącego wiadra, tzn. szybkości generowania znaczników i pojemności pamięci buforowej znaczników oraz wartości jednego parametru źródła przepływu, tzn. maksymalnej szybkości generowania jednostek danych.

2. Pośrednio ogranicza z góry wartość opóźnienia każdej jednostki danych wnoszonego przez stanowisko obsługi. Wartość ta, dla ustalonych wartości parametrów algorytmu realizowanego przez sterownik, zależy od szybkości obsługi, tzn. pojemności łącza.

Pierwsza z wymienionych funkcji może być stosowana w zadaniach nadzorowania lub kształtowania ruchu wprowadzonego do obsługi przez źródło przepływu.

Konsekwencją przekroczenia ustalonych warunków kontraktu ruchowego, dotyczących liczby i szybkości generowanych jednostek danych i -tego przepływu, jest konieczność podejmowania decyzji dotyczących obsługi jednostek danych wygenerowanych z naruszeniem warunków kontraktu.

W razie naruszenia warunków kontraktu ruchowego przez źródło jednostek danych i -tego przepływu stosowane są dwa różne mechanizmy:

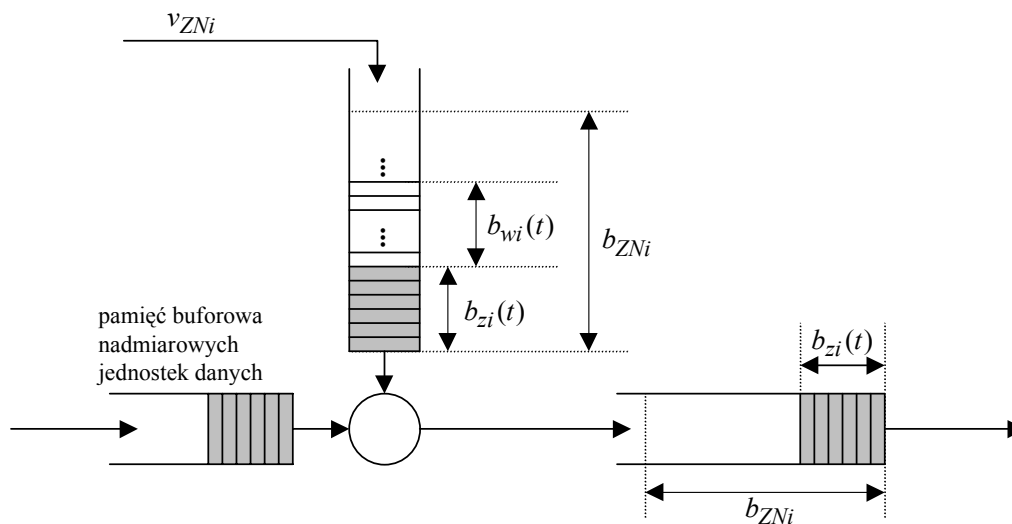
- nadzorowania przepływu (ang. *policing*),
- formowania przepływu (ang. *shaping*).

Procedura nadzorowania i -tego przepływu jednostek danych polega na tym, że na brzegu sieci porównywane są wartości parametrów oczekiwanego (determinowanego kontraktem ruchowym) i rzeczywiście generowanego strumienia jednostek danych. Jednostki danych niespełniające warunków kontraktu są odpowiednio znakowane i przyjmowane do obsługi. Oznakowane jednostki danych są obsługiwane zgodnie z zasadą najwyższej staranności, a w przypadku zaobserwowania symptomów przeciążenia sieci są odrzucane w pierwszej kolejności.

Procedura formowania i -tego przepływu jednostek danych polega na tym, że jednostki danych, wygenerowane przez źródło w ramach przyjętego do obsługi wywołania i niespełniające warunków kontraktu ruchowego wyznaczanych przez parametry sterownika przepływem, tzn. wtedy, gdy:

- maksymalna długość paczek jednostek danych przepływu i -tego przepływu jest większa od pojemności pamięci buforowej znaczników,
- szybkość napływu jednostek danych i -tego przepływu jest większa od szybkości napływu znaczników,

przyjmowane są do pamięci buforowej, z której są wyprowadzane z szybkością wyznaczaną przez wartości parametrów sterownika. Realizacja procedury formowania oznacza utworzenie w sterowniku przepływem dodatkowego bufora nadmiarowych jednostek danych. Procedura formowania przepływu może być realizowana przez sterownik, w którym implementowana jest rozszerzona wersja algorytmu ciekącego wiadra (rysunek 9.5).



Rysunek 9.5. Procedura formowania przepływu w przypadku naruszenia warunków kontraktu

Procedury nadzorowania i formowania ruchu są aktywowane wtedy, gdy naruszone zostają warunki dotyczące charakterystyk źródła ustalonych w kontrakcie rucho-

wym. Różnica działania omawianych procedur polega na tym, że procedura nadzorowania ruchu jest zorientowana na przestrzeganie warunków kontraktu ruchowego, natomiast procedura formowania ruchu – na efektywność wykorzystania zasobów sieci i niwelowanie różnic pomiędzy warunkami kontraktu ruchowego a rzeczywistym przepływem generowanym przez źródło.

9.3.2. MODEL STEROWANEGO OBCIĄŻENIA

9.3.2.1. JAKOŚĆ USŁUG

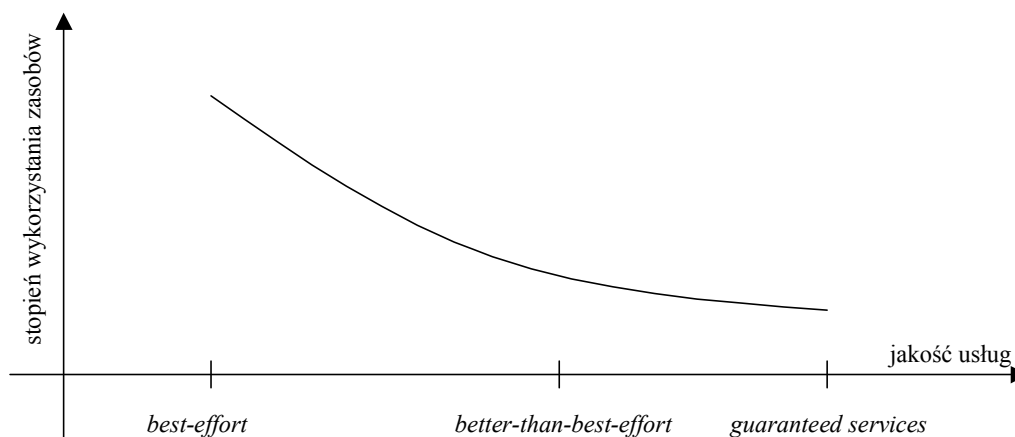
Udostępnianie zasobów sieci na potrzeby realizacji modelu gwarantowanej usługi CLS wiąże się z przydziałem zasobów przewidywanych na obsługę tzw. najgorszego przypadku. Jest to rozwiązanie uniemożliwiające na ogół efektywne wykorzystanie zasobów i podnoszące koszt dostarczania takiej usługi. Efektywność wykorzystania zasobów można poprawić jedynie przez dokładne oszacowanie wartości parametrów charakterystyk źródeł ruchu.

W przypadku aplikacji niewymagających jakości usług charakterystycznych dla modelu gwarantowanej usługi GS podobną jakość usług można uzyskać w modelu sterowanego obciążenia CLS. Ten model usługi nie gwarantuje uzyskania określonych wartości wskaźników jakości usług, takich jak opóźnienie czy przepustowość sieci. Model sterowanego obciążenia jest definiowany na potrzeby obsługi ruchu elastycznego, niewrażliwego na przeciążenia.

Cechą charakterystyczną koncepcji usług zintegrowanych jest możliwość jednoczesnej implementacji modeli gwarantowanej usługi i sterowanego obciążenia. Co więcej, zasoby niewykorzystywane do obsługi z gwarancją jakości, tzn. do realizacji koncepcji integracji usług, mogą być przeznaczone do obsługi ruchu bez gwarancji jakości usług. Oznacza to możliwość współistnienia w sieci różnych koncepcji dostarczania jakości usług.

Względnie wysoką jakość usług sieci w modelu sterowanego obciążenia uzyskuje się dzięki utrzymywaniu względnie małego obciążenia sieci, które pozwala na wzrost i kontrolowanie jakości usług dostarczanych zgodnie z zasadą najwyższej staranności (największego wysiłku). Utrzymywanie względnie małego obciążenia sieci jest tożsame z realizacją obsługi ruchu w sieci bez przeciążeń (bez natłoku). Jakość usług w modelu sterowanego obciążenia, dostarczanych zgodnie z zasadą najwyższej staranności w sieci bez przeciążeń, jest lokalizowana pomiędzy jakością usług w modelu gwarantowanej usługi i w modelu obsługi bez gwarancji jakości; stąd nazwa usługi – lepsza niż w modelu obsługi zgodnie z zasadą najwyższej staranności (ang. *better-than-best-effort*) (rysunek 9.6).

Przyjęcie modelu sterowanego obciążenia oznacza stosowanie modelu statystycznego zwielokrotniania przepływów i pozwala na zwiększenie efektywności wykorzystania zasobów w porównaniu z efektywnością możliwą do uzyskania w modelu gwarantowanej usługi.



Rysunek 9.6. Jakość usług w modelach: bez gwarancji jakości, sterowanego obciążenia i gwarantowanej usługi

9.3.2.2. OPISOWE CHARAKTERYSTYKI JAKOŚCI USŁUG

W modelu sterowanego obciążenia, zamiast podawania dokładnych wartości parametrów jakości usług, stosuje się opisowe charakterystyki oczekiwanej lub dostarczanej jakości usług.

Ze względu na to, że pojęcia, takie jak: względnie małe obciążenie i usługa lepsza niż, są mało precyzyjne i subiektywne, zarówno wymagania aplikacji co do jakości usług sieci, jak i jakość usług dostarczanych aplikacjom przez sieć, mogą być scharakteryzowane przez deklarowanie intencji dotyczących:

1. Oczekiwanych zachowań sieci, takich jak np.:

- udział liczby jednostek danych nadawanych w źródłach i dostarczanych do ujęć w całkowitej liczbie nadawanych jednostek danych będzie bardzo duży, tzn. liczba jednostek danych przyjętych do obsługi w sieci i odrzucanych w czasie obsługi w sieci z powodu przeciążeń sieci będzie bardzo mała,

- opóźnienie bardzo dużej części jednostek danych dostarczanych do ujęć nie będzie znacząco większe od ustalonej wartości opóźnienia, tzn. różnica opóźnień bardzo dużej części jednostek danych dostarczanych do ujęć nie będzie znacząco większa od ustalonej wartości.

2. Częstości zdarzeń, których zachodzenie nie jest oczekiwane, takich jak np.:

- częstość tego, że średnie opóźnienie jednostki danych, w czasie obsługi maksymalnych paczek jednostek danych, będzie znacząco większe od średniego opóźnienia jednostki danych, jest niewiele większa lub taka sama jak częstość występowania w przepływie paczek jednostek danych o maksymalnej długości,

- częstość tego, że straty jednostek danych, w czasie obsługi paczek jednostek danych o maksymalnej długości (w następstwie powstania przeciążeń sieci), będą

znacząco większe od strat jednostek danych w czasie trwania obsługi przepływu, jest niewiele większa lub taka sama jak częstość obsługi paczek jednostek danych o maksymalnej długości.

Akceptacja żądania obsługi, formułowanego w postaci oczekiwanych zachowań sieci lub zdarzeń, których zajście nie jest oczekiwane, oznacza odpowiednio, że prawdopodobieństwo powstania przeciążenia zasobów wykorzystywanych do obsługi żądania lub zajścia zdarzeń degradujących jakość usług są bardzo małe, ale różne od zera. Wartości tych prawdopodobieństw można interpretować następująco: udział przedziałów czasu, w których zasoby sieci są przeciążone i zachodzą zdarzenia degradujące oczekiwaną lub deklarowaną jakość usług w całkowitym czasie obsługi żądania jest bardzo mały.

9.3.2.3. PROCEDURA OBSŁUGI WYWOŁANIA

Obsługa przepływu jednostek danych w modelu sterowanego obciążenia jest realizowana podobnie jak w modelu gwarantowanej usługi.

Obsługa wywołania wymaga podania specyfikacji przepływu. Akceptacja wywołania następuje tylko wtedy, gdy możliwe jest ustanowienie trasy z dostępem do zasobów pozwalających na obsługę wywołania.

Decyzja o akceptacji lub odrzuceniu wywołania w modelu sterowanego obciążenia zależy od implementacji; ze względu na przybliżone i opisowe charakterystyki, żądanej przez wywołanie i możliwej do dostarczenia przez sieć jakości usług, nie jest możliwe dokładne sprecyzowanie warunków przyjęcia lub odrzucenia wywołania.

Podobnie jak w modelu usługi gwarantowanej, także w modelu sterowanego obciążenia niezbędne jest stosowanie na brzegu sieci mechanizmów monitorowania i nadzorowania zgodności generowanego przepływu z przepływem deklarowanym w specyfikacji źródła przepływu jednostek danych. W razie naruszenia przez źródło warunków kontraktu nadmiarowe jednostki danych mogą być przyjęte do obsługi tylko wtedy, gdy ich akceptacja nie spowoduje:

- zmiany jakości obsługi jednostek danych przepływu o parametrach deklarowanych w kontrakcie,
- zmiany jakości usług dostarczanych pozostałym przepływom, obsługiwanym równocześnie w sieci.

Jeżeli któryś z wymienionych wyżej warunków nie jest spełniony, to nadmiarowe jednostki danych są odrzucane.

9.4. REZERWACJA ZASOBÓW

Jakość usług ma zasadnicze znaczenie dla systemów transmisji danych, zwłaszcza dla systemów multimedialnych. Charakterystyczne dla aplikacji multimedialnych jest

jednocześnie duże i zmienne zapotrzebowanie na zasoby oraz konieczność dostarczenia mniej lub bardziej restrykcyjnych gwarancji jakości usług. Duże zapotrzebowanie na zasoby prowadzi zwykle do ekstensywnego podziału zasobów, powodując tym samym konieczność efektywnego zarządzania zasobami, gwarancję jakości usług można natomiast uzyskać za pomocą różnego rodzaju mechanizmów rezerwacji zasobów. Oznacza to, że jednakowo ważne dla obsługi ruchu multimedialnego zadania efektywnego wykorzystania zasobów i gwarantowania jakości usług są sprzeczne, to znaczy, że w rzeczywistych systemach konieczne jest podejmowanie decyzji prowadzących do kompromisu w ramach istniejących ograniczeń: ilości dostępnych zasobów oraz żądań jakości usług.

Podstawową właściwością koncepcji usług zintegrowanych jest poprzedzanie przyjęcia i obsługi generowanego przez źródła ruchu rezerwacją zasobów. Rezerwacja zasobów wymaga odpowiedniego protokołu rezerwacji, pozwalającego na przydział zasobów wzdłuż wcześniej ustalonych tras łączących systemy końcowe, tj. źródła i ujścia przepływów. Sposób rezerwacji zasobów oraz ich ilości zależą od definiowanej jakości usług żądanych od sieci i dostarczanych przez nie.

Rezerwacja zasobów w sieciach wymaga stosowania odpowiednich protokołów rezerwacji, których podstawowym zadaniem jest sprawdzenie dostępności żądanych ilości zasobów pomiędzy systemami końcowymi przepływu oraz ich rezerwacji w razie stwierdzenia ich dostępności. Przykładem protokołu rezerwacji, spełniającego warunki wymagane do realizacji koncepcji usług zintegrowanych, jest protokół RSVP (ang. *Resource Reservation Protocol*).

9.4.1. WŁAŚCIWOŚCI PROTOKOŁU REZERWACJI ZASOBÓW

Protokół RSVP, zaprojektowany jako protokół dodany do protokołu IP, ma cechy rozwiązania determinowanego zarówno wspieraną przez ten protokół architekturą TCP/IP, jak i klasą obsługiwanych aplikacji. Protokół RSVP charakteryzują następujące właściwości:

- kierunkowość, tzn. rezerwacja zasobów jest możliwa tylko w jednym kierunku (ang. *simplex reservation*); w przypadku aplikacji działających jednocześnie jako źródło i ujście przepływów, rezerwacja dokonywana jest oddzielnie dla dwóch kierunków,
- zorientowanie na ujście, tzn. inicjowanie rezerwacji zasobów odbywa się w ujściu przepływu; rozwiązanie takie pozwala na jednakowy sposób obsługi różnych typów ruchu (np. punkt–punkt, punkt–wielopunkt),
- wieloetapowość rezerwacji, tzn. rezerwacja zasobów rozpoczyna się w ujściu (ujściach) przepływów, a w wyniku propagowania informacji o rezerwowanych zasobach w kierunku do źródła stopniowo budowane jest drzewo rezerwacji (ang. *reservation tree*),
- niezależność od protokołów wyboru tras, tzn. do przekazywania informacji o rezerwowanych zasobach wykorzystywane są trasy wybierane niezależnie od

protokołu rezerwacji zasobów, co oznacza, że proces rezerwacji zasobów można interpretować jako proces potwierdzania ustanowienia trasy z dodatkową funkcją rezerwacji,

- ogólność, tzn. mechanizm rezerwacji zasobów jest uniwersalnym mechanizmem kreowania i utrzymywania stanu rezerwacji zasobów w drzewie punkt–wielopunkt lub na trasie punkt–punkt; parametry mechanizmu rezerwacji zasobów, jako modułu, mogą być stosowane przez inne mechanizmy sterowania (np. procedury sterowania dostępem do sieci),

- niezależność, tzn. protokół rezerwacji jest wyposażony we własne parametry sterowania, przekazywane w nieprzeźroczystych, z punktu widzenia protokołu, wiadomościach; parametry te mogą być dostępne dla innych mechanizmów sterowania, współpracujących z protokołem rezerwacji,

- elastyczność, tzn. stan rezerwacji zasobów utrzymuje się w przedziale czasu o długości ustalonej dla danego stanu; podtrzymanie danego stanu rezerwacji po zakończeniu ustalonego przedziału czasu wymaga odświeżenia stanu rezerwacji (ponownego ustawienia zegara); w innym przypadku dany stan rezerwacji jest automatycznie znoszony,

- wariantowość modeli rezerwacji, tzn. ogólność protokołu pozwala na implementację różnych stylów rezerwacji zasobów, różniących się dokładnością szacowania zapotrzebowania na zasoby w czasie budowy drzewa rezerwacji.

9.4.2. OGÓLNE ZASADY DZIAŁANIA PROTOKOŁU REZERWACJI ZASOBÓW

W protokole RSVP wykorzystywane są dwa typy wiadomości:

1. Wiadomości typu Path, wysyłane przez źródła przepływów i przekazywane do ujść po trasach, ustalanych zgodnie z odpowiednim protokołem routingu typu punkt–punkt lub punkt–wielopunkt, spełniają następujące zadania:

- przenoszą informacje o żądanych parametrach ruchowych źródła ruchu do wywoływanego ujścia (systemu docelowego),

- przenoszą charakterystyki trasy,

- znakują węzły pośrednie, na trasie ze źródła do ujścia, w sposób umożliwiający transmisję wiadomości typu Resv, generowanych w ujściach w odpowiedzi na wiadomości typu Path.

2. Wiadomości typu Resv, generowane w ujściach po otrzymaniu wiadomości typu Path, są wiadomościami:

- przekazywanymi z ujść do źródeł po trasach, którymi były przekazywane wiadomości typu Path, których odebranie w ujściu spowodowało wygenerowanie wiadomości Resv,

- przenoszącymi żądania rezerwacji zasobów na trasie z ujścia do źródła,

- przenoszącymi specyfikację żądanych zasobów,

- ustanawiającymi stany rezerwacji zasobów w węzłach pośrednich na trasach łączących ujście ze źródłem,
- inicjującymi, po osiągnięciu odpowiedniego źródła, rozpoczęcie fazy transferu przepływu przez źródło.

Po stwierdzeniu niedostępności zasobów w żądanych ilościach, w ujściu generowana jest wiadomość typu ResErr, której odebranie w źródle może powodować powtórzenie wysłania wiadomości Path. Nowa wiadomość Path zawiera zmieniony, w porównaniu z wysłanym w poprzedniej wiadomości Path, zestaw żądanych parametrów ruchowych. Zmiana wartości parametrów ruchowych wynika z ograniczenia ilości żądanych zasobów i ma na celu ustanowienie połączenia o niższej, w porównaniu z początkowo żadaną, jakości usług. Brak możliwości rezerwacji zasobów, w ilościach wynikających z generowanego zestawu parametrów ruchowych, oznacza obsługę generowanego przepływu zgodnie z zasadą najwyższej staranności.

Wiadomości protokołu RVSP przesyłane są od węzła do węzła (skokowo) jako datagramy IP lub jako wiadomości enkapsulowane w pakietach UDP (ang. *User Datagram Protocol*). Gdy wiadomości są dłuższe od pola danych datagramu IP, wiadomości protokołu RVSP podlegają fragmentacji, a w ujściu są odtwarzane.

Wiadomości Path znakują kolejne węzły (routery) na trasie od źródła do ujścia. Ze znakowania tego korzystają wiadomości typu Resv podczas ich transferu z ujścia do źródła. W skrajnym przypadku znakowanie w węzle pośrednim polega na pozostawieniu w nim adresu poprzedniego węzła na trasie pomiędzy węzłami końcowymi.

Wiadomość typu *Path* zawiera informacje, wykorzystywane w zadaniach znakowania, identyfikacji trasy i przepływu oraz przenoszenia informacji o żądanych zasobach, m.in.:

1. Adres węzła poprzedzającego dany węzeł na trasie ze źródła do ujścia (ang. *previous hop*).
2. Identyfikator przepływu, dla którego inicjowana jest procedura ustalenia trasy i rezerwacji zasobów; w szczególnym przypadku identyfikatorem przepływu może być np. adres źródła (ang. *sender template*).
3. Zestaw żądanych parametrów ruchowych, tzn. informacje charakteryzujące przepływ, który będzie generowany przez źródło żądające ustanowienia trasy i przydzielenia zasobów; zestaw parametrów ruchowych jest wykorzystywany do ustalania ilości żądanych i rezerwowanych zasobów oraz sterowania dostępem do sieci (informacja ta nie jest zmieniana w czasie transferu wiadomości typu *Path*), (ang. *sender Tspec*).
4. Zestaw parametrów zapewnianych przez sieć, tzn. informacje określające aktualne parametry transmisyjne trasy prowadzącej od źródła (ang. *AdSpec*) zawarte w trzech grupach parametrów: parametrów ogólnych (ang. *default general parameters*), parametrów usługi gwarantowanej (ang. *guaranteed service*) i parametrów usługi sterowanego obciążenia (ang. *controlled load service*), w tym m.in.:

- sumę stałych opóźnień na trasie (ang. *minimum path latency*); opóźnienie to jest wnoszone przez trasę niezależnie od opóźnień spowodowanych oczekiwaniem na

obsługę w kolejkach, a jego wartość jest wykorzystywana w ujściach do szacowania górnych wartości dopuszczalnego opóźnienia oczekiwania na obsługę w węzłach na trasie punkt–punkt,

- minimalną pojemność (pasmo) trasy (ang. *path bandwidth*),
- całkowitą liczbę węzłów (skoków) (ang. *integrated services hop count*), która określa zasięg działania usługi zintegrowanej,
- informację o braku wsparcia dla koncepcji usług zintegrowanych w co najmniej jednym węźle trasy (ang. *global break bit*),
- maksymalną długość pola danych pakietu (ang. *path maximum transmission unit*).

Wiadomości typu Rvsp, które można interpretować jako żądania rezerwacji na trasie z ujścia do źródła, są stosowane do rezerwacji zasobów i przenoszenia informacji o rezerwowanych zasobach w postaci dwóch specyfikacji, noszących nazwę deskryptora przepływu (ang. *flow descriptor*) i zawierających m.in.:

- specyfikację żądanej jakości usług (ang. *flow spec*), której częścią jest zestaw przyjętych parametrów ruchowych Rspec (ang. *receiver specification*),
- zestaw informacji pozwalających na filtrowanie jednostek danych wewnątrz przepływu (ang. *filter spec*); jednostki danych, które spełniają warunki filtru zawarte w polu *service class*, są obsługiwane jako jednostki danych deklarowanej klasy jakości usług, podczas gdy jednostki nie spełniające warunków filtru są obsługiwane zgodnie z zasadą najwyższej staranności,
- informację o stosowanym stylu rezerwowania zasobów.

9.4.3. MODELE REZERWACJI ZASOBÓW

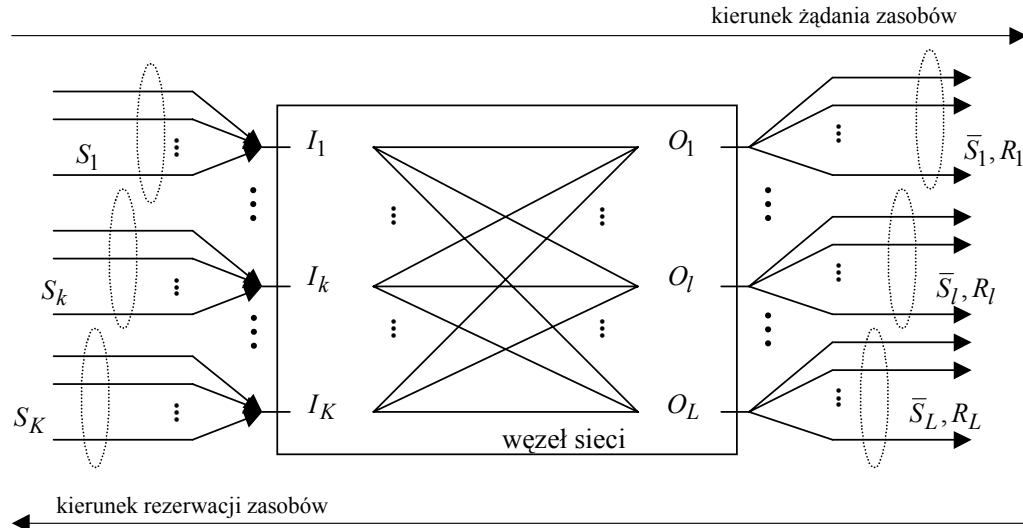
Sprawdzenie i stwierdzenie dostępności żądanych, dla przepływów pomiędzy systemami końcowymi, ilości zasobów uruchamia procedurę rezerwacji zasobów realizowanej w kierunku od ujścia do źródła. W protokole RSVP żądanie rezerwacji zasobów sieci może być obsługiwane zgodnie z różnymi stylami rezerwacji (ang. *style reservation*), które różnią się:

- sposobami agregacji ilości zasobów, rezerwowanych dla poszczególnych przepływów, w węzłach sieci,
- sposobami przekazywania informacji o zarezerwowanych zasobach w kierunku od ujścia do źródła przepływów żądających rezerwacji zasobów,
- ilością i szczegółowością informacji o zasobach rezerwowanych dla przepływów i przekazywanych pomiędzy węzłami pośrednimi i końcowymi trasy.

Żądania rezerwacji zasobów, w danym dowolnym węźle (router, przełącznik) sieci na trasie pomiędzy systemami końcowymi, są żądaniami zasobów $c(s_n, r_m)$ dla przepływów w sieci pomiędzy:

- źródłami przepływów ze zbioru
- $$S = \{s_1, s_2, \dots, s_N\},$$
- ujściami przepływów ze zbioru
- $$R = \{r_1, r_2, \dots, r_m, \dots, r_M\}.$$

Dla uproszczenia ilustracji działania różnych modeli rezerwacji przyjęto, że jedynym rezerwowanym zasobem sieci jest pojemność łączy danych na trasie przepływu (rysunek 9.7).



$$S = \{s_1, s_2, \dots, s_n, \dots, s_N\} = \bigcup_{k=1}^K S_k = \bigcup_{l=1}^L \bar{S}_l$$

$$R = \{r_1, r_2, \dots, r_m, \dots, r_M\} = \bigcup_{l=1}^M R_l$$

Rysunek 9.7. Schemat węzła sieci wykorzystywany w modelach rezerwacji zasobów

Wartość $c(S_n, S_m)$ jest równa zero wtedy, gdy nie ma żądania przepływu pomiędzy n -tym ($n \in \{1, 2, \dots, N\}$) źródłem i m -tym ($m \in \{1, 2, \dots, M\}$) ujściem; w pozostałych przypadkach wartość $c(S_n, S_m)$ jest różna od zera.

Podzbiory R_l zbioru R spełniają warunki:

$$R_1 \cup R_2 \cup \dots \cup R_L = R,$$

$$R_l \cap R_i = \emptyset \quad \text{dla } l \neq i, \text{ gdzie } l, i \in \{1, 2, \dots, L\}.$$

Podzbiory S_k zbioru S spełniają warunki:

$$S_1 \cup S_2 \cup \dots \cup S_K = S,$$

$$S_k \cap S_j = \emptyset \quad \text{dla } k \neq j, \text{ gdzie } k, j \in \{1, 2, \dots, K\}.$$

Z każdym k -tym wejściem węzła I_k ($k = 1, 2, \dots, K$), charakteryzowanym podzbiorem źródeł przepływów S_k , związana jest określona ilość żądanych zasobów $c(S_k)$.

Podobnie dla danego węzła sieci zbiór ujść przepływów można podzielić na zbiór niepustych podzbiórów zbioru ujść przepływów $R = \{R_1, R_2, \dots, R_l, \dots, R_L\}$, gdzie R_l ($l \in \{1, 2, \dots, L\}$) jest podzbiorem ujść przepływów ze zbioru R , przechodzących przez l -te wyjście węzła.

Z każdym l -tym wyjściem węzła O_l ($l = 1, 2, \dots, L$), charakteryzowanym podzbiorem zbioru wszystkich źródeł przepływów \bar{S}_l ($\bar{S}_1 \cup \bar{S}_2 \cup \dots \cup \bar{S}_l \cup \dots \cup \bar{S}_L = S$) oraz podzbiorem zbioru ujść przepływów R_l , związana jest określona ilość żądanych zasobów $c(R_l)$. Dla uproszczenia można założyć, że ilość zasobów żądanych przez l -te ujście jest całkowitą wielokrotnością jednostkowego zapotrzebowania na zasoby.

Dla tak opisanego węzła sieci oraz zbioru jego wejść i wyjść, zdefiniowane dla protokołu RSVP style rezerwacji, są następujące:

- rezerwacja sumaryczna WF (ang. *Wildcard Filter*),
- rezerwacja indywidualna FF (ang. *Fixed Filter*),
- rezerwacja hybrydowa SE (ang. *Shared Explicit*).

Wymienione style rezerwacji różnią się zakresami rezerwacji i sposobami kontroli wyboru nadawców (źródeł). Rezerwacja może dotyczyć każdego źródła (ang. *distinct*) lub grupy źródeł (ang. *shared*), przez które jest współdzielona, oraz bezpośrednio wskazanych źródeł lub być niezależna od liczby źródeł korzystających z tej rezerwacji.

Style rezerwacji WF i SE są dedykowane aplikacjom rozgłoszeniowym, dla których mało prawdopodobne jest zdarzenie, że wszystkie źródła, rezerwujące wcześniej zasoby, generują i transmitują swoje przepływy równocześnie. Przykładem takich aplikacji są asymetryczne multimedialne aplikacje interaktywne, w których w danej chwili aktywna jest zwykle tylko jedna z komunikujących się stron.

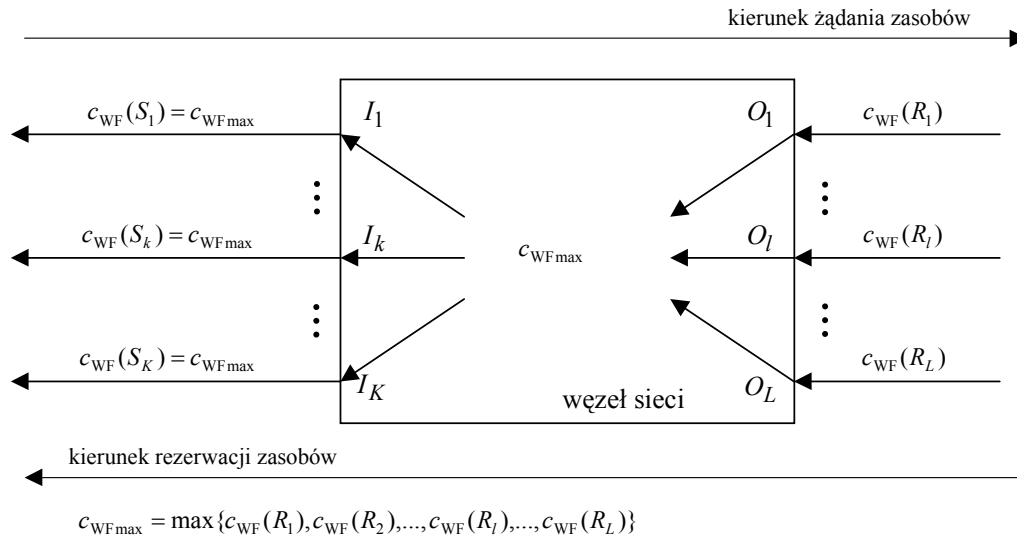
Wyróżnione style rezerwacji zasobów charakteryzują różne sposoby szacowania zapotrzebowania na ilość rezerwowanych zasobów, różne sposoby łączenia zapotrzebowania na zasoby oraz różna ilość informacji przenoszonych w kierunku do źródeł i dotyczących udziału rezerwacji zasobów dla poszczególnych przepływów w rezerwacji dla przepływów sumarycznych.

Omawiane trzy style rezerwacji zasobów sieciowych są różnymi sposobami łączenia żądań zasobów i propagowania informacji o rezerwowanych zasobach z ujść do źródeł. Łączenie rezerwowanych zasobów dokonuje się zarówno na interfejsach węzłów sieci wyjściowych, jak i wejściowych. Generacja ilości rezerwowanych zasobów na wejściach węzłów jest funkcją ilości zasobów rezerwowanych na wyjściach węzłów oraz sposobu łączenia charakterystycznego dla danego modelu. Z powodu różnic stylów nie jest możliwe ich współistnienie w jednej sieci.

9.4.3.1. STYL REZERWACJI SUMARYCZNEJ

W modelu WF (ang. *Wildcard Filter*) na wyjściach węzła dokonywana jest (przez ujścia przepływów) wspólna rezerwacja zasobów dla wszystkich ujść przepływów dołączonych do wyjść węzła sieci. Ilość rezerwowanych zasobów na wyjściu węzła jest równa największej wartości żądanych zasobów przez poszczególne ujścia przepływów, obsługiwane przez dane wyjście węzła. Zarezerwowane zasoby są współdzielone przez wszystkie źródła przepływów kierowanych do ujść obsługiwanych przez dane wyjście węzła. Podobnie rezerwowane są zasoby na wejściach węzła: zasoby dla wszystkich

źródeł korzystających z tych wejść są wspólne. Ten sposób rezerwacji zasobów charakteryzuje wspólna rezerwacja zasobów dla źródeł przepływów na wejściach i wyjściach węzła sieci (ang. *shared reservation*) bez wskazywania źródeł, którym zasoby będą udostępniane (ang. *wildcard sender selection*). Omawiany styl rezerwacji zasobów charakteryzuje nadmiarowa rezerwacja zasobów na wejściach węzła, która jest, w czasie procesu rezerwacji, propagowana w kierunku od ujść do źródeł (rysunek 9.8).



Rysunek 9.8. Styl rezerwacji zasobów w modelu rezerwacji sumarycznej

Zgodnie z omawianym stylem zasoby są rezerwowane i udostępniane według następującej procedury:

➤ ilość zasobów rezerwowanych dla każdego l -tego wyjścia węzła, oznaczana przez $c_{WF}(R_l)$, jest równa maksymalnej ilości zasobów rezerwowanych przez poszczególne ujścia:

$$c_{WF}(R_l) = \max_{r_m \in R_l} c_l(r_m),$$

gdzie:

$$c_l(r_m) = \sum_{s_n \in \bar{S}_l} [c(s_n, r_m) | r_m \in R_l],$$

➤ spośród wszystkich wartości $c_{WF}(R_l)$ ($l \in \{1, 2, \dots, L\}$) wybierana jest wartość maksymalna:

$$c_{WFmax} = \max \{c_{WF}(R_1), c_{WF}(R_2), \dots, c_{WF}(R_l), \dots, c_{WF}(R_L)\},$$

➤ ilość zasobów rezerwowanych dla każdego k -tego wejścia węzła, tj. dla podzbioru zbioru źródeł przepływów dołączonych do k -tego wejścia węzła, oznaczona przez $c_{WF}(I_k)$, jest taka sama i równa c_{WFmax} :

$$c_{WF}(S_k) = c_{WFmax} \text{ dla } k \in \{1, 2, \dots, K\},$$

➤ wszystkim źródłom przepływów należących do podzbioru S_k , tj. źródłom przepływów dołączonym do k -tego wejścia węzła, udostępniana jest taka sama ilość zasobów $c_{WF}(S_k)$; zasoby te są współdzielone przez te źródła.

W omawianym stylu rezerwacji poszczególne wejścia i wyjścia węzła są opisane sumaryczną ilością zasobów zarezerwowanych dla tych wejść $c_{WF}(S_k)$ oraz wyjść $c_{WF}(R_l)$. Oznacza to, że nie ma możliwości klasyfikacji przepływów ze względu na ich zgodność z wcześniejszą rezerwacją zasobów.

Styl rezerwacji WF tworzy pojedynczą rezerwację współdzieloną pomiędzy przepływy od wszystkich źródeł. Ilość rezerwowanych zasobów jest większa od żądanej i niezależna od liczby korzystających z niej źródeł. Rezerwacja w stylu WF jest propagowana „w górę” w kierunku źródeł i jest rozszerzana o rezerwacje dla kolejnych źródeł w miarę pojawiania się nowych żądań.

9.4.3.2. STYL REZERWACJI INDYWIDUALNEJ

W modelu FF (ang. *Fixed Filter*) na wyjściach węzła dokonywana jest (przez ujścia) oddzielna rezerwacja zasobów dla poszczególnych źródeł. Zarezerwowane zasoby, kierowane do poszczególnych źródeł, są współdzielone przez wszystkie ujścia, które dokonały rezerwacji. Podobnie na wejściach węzła – rezerwacja jest dokonywana oddzielnie dla poszczególnych źródeł przepływów. Ten sposób rezerwacji charakteryzuje oddzielna rezerwacja na wejściach i wyjściach węzła sieci dla poszczególnych źródeł (ang. *distinct reservation*) oraz bezpośrednie wskazywanie źródeł, którym zasoby są udostępniane (ang. *explicit sender selection*). Z efektywnym rozdziałem zasobów w węzłach sieci pomiędzy poszczególne źródła przepływów wiąże się konieczność propagacji (w kierunku od ujść do źródeł) informacji zarówno o ilości rezerwowanych zasobów, jak i o źródłach, dla których te ilości zasobów są rezerwowane (rysunek 9.9).

Zgodnie z omawianym modelem zasoby są rezerwowane i udostępniane według następującej procedury:

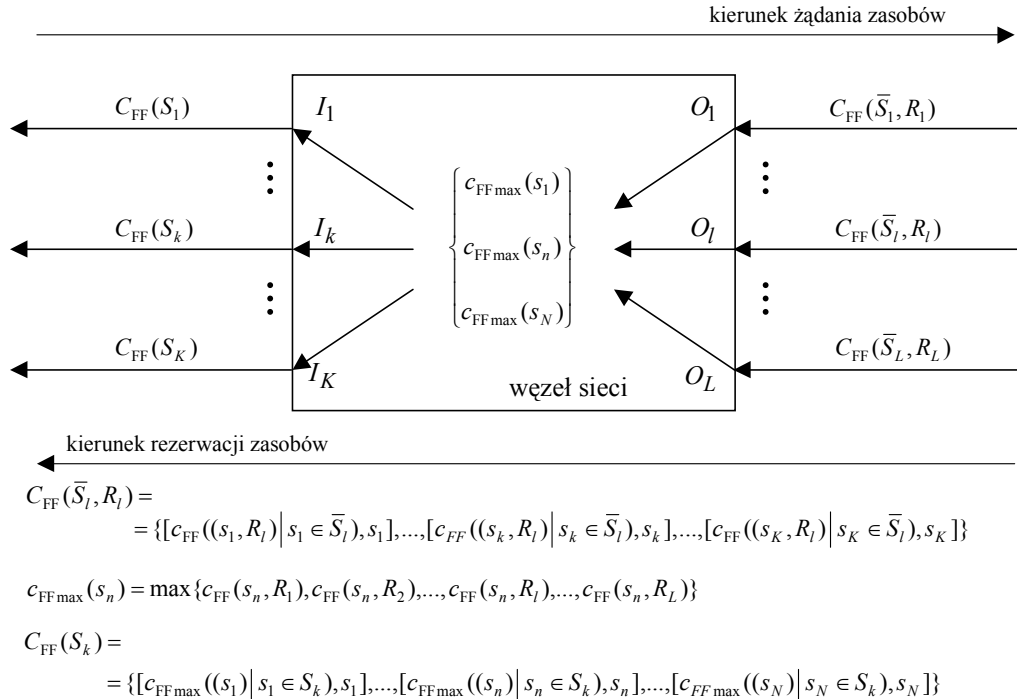
➤ ilość zasobów rezerwowanych dla każdego l -tego wyjścia węzła, oznaczona przez $c_{FF}(R_l)$, jest sumą ilości zasobów w zbiorze zasobów $C_{FF}(\bar{S}_l, R_l)$:

$$C_{FF}(\bar{S}_l, R_l) = \\ = \left\{ \left[c_{FF}((s_1, R_l) | s_1 \in \bar{S}_l), s_1 \right], \dots, \left[c_{FF}((s_n, R_l) | s_n \in \bar{S}_l), s_n \right], \dots, \left[c_{FF}((s_N, R_l) | s_N \in \bar{S}_l), s_N \right] \right\},$$

gdzie:

$$c_{FF}((s_n, R_l) | s_n \in \bar{S}_l) = \begin{cases} 0 & \text{gdy } s_n \notin \bar{S}_l \\ \sum_{r_m \in R_l} c(s_n, r_m) & \text{gdy } s_n \in \bar{S}_l \end{cases}$$

jest ilością zasobów dla przepływów ze źródła s_n do ujść r_m przekazywanych przez l -te wyjście węzła.



Rysunek 9.9. Styl rezerwacji zasobów w modelu indywidualnej rezerwacji

Ilości rezerwowanych zasobów dla źródła s_n ($n = 1, 2, \dots, N$) są różne od zera wtedy i tylko wtedy, gdy $s_n \in \bar{S}_l$:

$$c_{FF}((s_n, R_l) | s_n \in \bar{S}_l) = \sum_{r_m \in R_l} c(s_n, r_m),$$

➤ zapis dotyczący elementu $[c_{FF}((s_n, R_l) | s_n \in \bar{S}_l), s_n]$ ($n \in \{1, 2, \dots, N\}$, $l \in \{1, 2, \dots, L\}$) należącego do zbioru $C_{FF}(\bar{S}_l, R_l)$ oznacza, że zasoby są rezerwowane dla poszczególnych źródeł przepływów, a informacja o zarezerwowanych zasobach dla źródła przepływu jest przekazywana wraz z indeksem źródła,

➤ ilości zasobów rezerwowanych dla każdego l -tego wyjścia węzła są dane w postaci zbioru wartości $C_{FF}(\bar{S}_l, R_l)$; każdy element tego zbioru odpowiada ilości zasobów rezerwowanych dla różnych źródeł przepływów,

➤ zasoby rezerwowane dla każdego źródła przepływu na l -tym wyjścia węzła w ilości równej $c_{FF}[(s_n, R_l) | s_n \in \bar{S}_l]$, są udostępniane źródłu s_n ($s_n \in \bar{S}_l$) indywidualnie, tzn. zasoby udostępniane przepływowi przechodzącym przez l -te wyjście węzła są rezerwowane i zarządzane jako zasoby dla sumy przepływów identyfikowanych indeksami źródeł przepływów,

➤ ilość zasobów rezerwowanych w węźle dla każdego n -tego źródła (dołączonego do k -tego wejścia węzła) jest równa maksymalnej ilości zasobów rezerwowanych dla n -tego źródła w jednym z L wyjść węzła sieci:

$$c_{\text{FFmax}}(s_n) = \max \{c_{\text{FF}}[(s_n, R_1) | s_n \in \bar{S}_1], \dots, c_{\text{FF}}[(s_n, R_2) | s_n \in \bar{S}_2], \dots, c_{\text{FF}}[(s_n, R_L) | s_n \in \bar{S}_L]\},$$

➤ ilość zasobów rezerwowanych dla każdego k -tego wejścia węzła – $c_{\text{FF}}(S_k)$, tj. dla podzbioru zbioru źródeł przepływów dołączonych do k -tego wejścia węzła, jest sumą ilości zasobów rezerwowanych dla poszczególnych źródeł i dana jest w postaci zbioru rezerwacji zasobów:

$$C_{\text{FF}}(S_k) = \\ = \{[c_{\text{FFmax}}((s_1) | s_1 \in S_k), s_1], [c_{\text{FFmax}}((s_2) | s_2 \in S_k), s_2], \dots, [c_{\text{FFmax}}((s_N) | s_N \in S_k), s_N]\},$$

➤ zasoby udostępniane w k -tym wejściu węzła, dane zbiorem zasobów $C_{\text{FF}}(S_k)$, w którym ilości zasobów rezerwowane poszczególnym źródłom przepływów, indeksowane numerami źródeł, są udostępniane każdemu ze źródeł przepływów s_n , przechodzących przez k -te wejście węzła sieci ($s_n \in S_k$) z osobna, w ilości $c_{\text{FFmax}}(s_n)$.

Styl FF tworzy oddzielną rezerwację dla jednostek danych każdego przepływu, nie dzieląc jej z jednostkami danych innych przepływów.

Omawiany styl rezerwacji zasobów daje gwarancję, że przepływ, generowany przez dowolne źródło, zostanie zawsze obsłużony. Co więcej, rezerwacja zasobów dla każdego ze źródeł przepływów z osobna oraz przenoszenie informacji zarówno o ilościach zarezerwowanych zasobów, jak i źródle, dla którego zasoby zostały zarezerwowane, umożliwi indywidualne wymuszanie alokacji zasobów dla każdego ze źródeł oraz indywidualne monitorowanie stopnia wykorzystania zarezerwowanych zasobów przez każde ze źródeł.

9.4.3.3. STYL REZERWACJI HYBRYDOWEJ

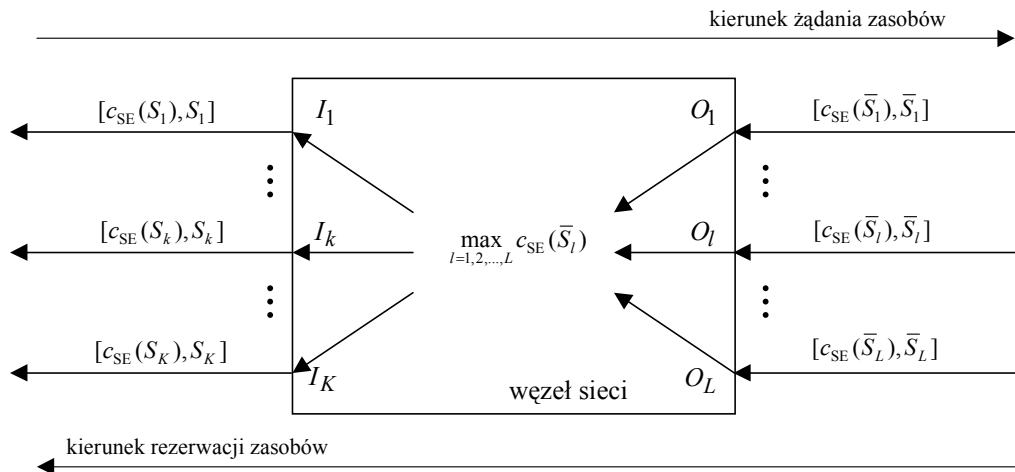
W modelu SE (ang. *Shared Explicit*) na wyjściach węzła dokonywana jest (przez ujścia) rezerwacja zasobów dla wszystkich źródeł żądających dostępu przez dane wyjście węzła. Zarezerwowane zasoby wyjścia są współdzielone przez wszystkie ujścia, które dokonały rezerwacji. Podobnie na wejściach węzła – zasoby są rezerwowane dla wszystkich źródeł wykorzystujących dane wejście węzła. W obydwu przypadkach źródła współdzielące rezerwowane zasoby są wskazywane bezpośrednio. Omawiany styl rezerwacji charakteryzuje rezerwacja dla wszystkich źródeł współdzielących zasoby (ang. *shared reservation*) oraz bezpośrednie wskazywanie źródeł, którym zasoby są udostępniane (ang. *explicit sender selection*); jest to rozwiązanie łączące cechy modeli sumarycznej i indywidualnej rezerwacji zasobów (rysunek 9.10).

Zgodnie z omawianym stylem, zasoby są rezerwowane i udostępniane według następującej procedury:

➤ ilość zasobów rezerwowanych dla każdego, l -tego wyjścia węzła, oznaczana przez $c_{SE}(\bar{S}_l)$, jest sumą zasobów rezerwowanych dla wszystkich źródeł, z których przepływy przechodzą przez l -te wyjście węzła:

$$c_{SE}(\bar{S}_l) = \sum_{s_n \in \bar{S}_l} \sum_{r_m \in R_l} c(s_n, r_m),$$

➤ zasoby w ilości $c_{SE}(\bar{S}_l)$, zarezerwowane dla l -tego wyjścia węzła, współdzielone są przez wszystkie źródła należące do podzbioru źródeł przepływów \bar{S}_l ; podzbiór \bar{S}_l jest przekazywany razem z sumaryczną ilością rezerwowanych zasobów, tj. w postaci: $[c_{SE}(\bar{S}_l), \bar{S}_l]$,



$$[c_{SE}(\bar{S}_l), \bar{S}_l] = \left[\sum_{s_n \in \bar{S}_l} \sum_{r_m \in R_l} c(s_n, r_m), \bar{S}_l \right]$$

$$[c_{SE}(S_k), S_k] = \left[\max_{l=1,2,\dots,L} (c_{SE}(\bar{S}_l) \mid S_k \cap \bar{S}_l \neq \emptyset, S_k) \right]$$

Rysunek 9.10. Styl rezerwacji zasobów w modelu hybrydowej rezerwacji

➤ do rezerwacji zasobów dla każdego k -tego wejścia węzła wykorzystywana jest ilość zasobów rezerwowanych na wyjściach węzła $c_{SE}(\bar{S}_l)$ ($l = 1, 2, \dots, L$) oraz zbiór indeksów źródeł \bar{S}_l , tj. wszystkich źródeł, których przepływy przechodzą przez l -te wyjście węzła,

➤ ilość zasobów rezerwowanych dla każdego, k -tego wejścia węzła, oznaczana przez $c_{SE}(S_k)$, jest równa maksymalnej ilości zasobów zarezerwowanych dla tych

wszystkich wyjść węzła, których podzbiory źródeł przepływów \bar{S}_l mają co najmniej jeden element wspólny ze zbiorem S_k :

$$c_{SE}(S_k) = \max_{l=1,2,\dots,L} [c_{SE}(\bar{S}_l) | S_k \cap \bar{S}_l \neq \emptyset],$$

➤ zasoby w ilości $c_{SE}(S_k)$, zarezerwowane dla k -tego wejścia węzła, są współdzielone przez wszystkie źródła należące do podzbioru źródeł przepływów S_k ; informacja o sumarycznej ilości zasobów zarezerwowanych dla źródeł przepływów $c_{SE}(S_k)$ jest podawana wraz z informacją o źródłach należących do podzbioru źródeł S_k , tj. w postaci $[c_{SE}(S_k), S_k]$.

Rezerwacja w stylu SE tworzy pojedynczą rezerwację współdzieloną przez wskazaną jawnie grupę źródeł.

W stylu hybrydowym rezerwacji zasobów wraz z informacją o sumarycznej ilości zasobów przekazywaną z węzła do węzła podawana jest informacja o podziorach źródeł przepływów uprawnionych do korzystania z tych zasobów. Informacja o źródłach przepływów pozwala na alokację zarezerwowanych zasobów dla wskazywanych podzbiorów źródeł oraz grupowe monitorowanie stopnia wykorzystania zarezerwowanych zasobów przez wskazane podzbiory źródeł.

9.5. IDENTYFIKACJA PRZEPLÝWU

Zadaniem identyfikacji jednostek danych przepływów, realizowanych w węzłach na trasach pomiędzy systemami końcowymi, jest rozpoznawanie i kwalifikowanie jednostek danych do przepływów, dla których w sieci zostały zarezerwowane zasoby. Jednostki danych identyfikuje się na podstawie zawartości wszystkich lub wybranych pól nagłówek jednostek danych (np. pakietów).

Podstawowym celem identyfikacji jest rozstrzygnięcie o przynależności jednostek danych do poszczególnych przepływów, to znaczy rozstrzygnięcie o możliwości użytkowania zarezerwowanych zasobów sieci. Efektywność identyfikacji jest jednym ze sposobów gwarantowania jakości usług ustalonych w kontrakcie ruchowym. Błędna identyfikacja jednostek danych przepływu oznacza albo dodatkowe obciążenie zasobów, albo pogorszenie jakości usług dostarczanych przez sieć.

Zadanie identyfikacji jednostek danych przepływów jest szczególnym przypadkiem klasyfikacji (filtrowania) jednostek danych. Klasyfikacja jednostek danych jest zadaniem ogólniejszym od zadania identyfikacji, w tym sensie, że reguły filtrowania jednostek danych mogą być wyrażane jako zakresy wartości. Posługiwanie się różnymi zakresami wartości daje większą elastyczność mechanizmów filtrowania jednostek danych, ale wymaga także stosowania bardziej złożonych algorytmów.

Mechanizmy identyfikacji jednostek danych przepływów zalicza się do grupy mechanizmów umożliwiających implementację koncepcji usług zintegrowanych.

9.5.1. PODSTAWOWE WYMAGANIA W ODNIESIENIU DO MECHANIZMU IDENTYFIKACJI PRZEPIŁYWU

Realizacja zadań kierowania ruchem (routing, przełączanie) w węzłach (routerach) sieci z komutacją jednostek danych oznacza konieczność implementacji w sieci mechanizmów umożliwiających kolejno:

- pobranie informacji sterującej z nagłówka jednostki danych,
- porównanie tej informacji z informacjami przechowywanymi w danym węźle (np. tablica rezerwacji zasobów, tablica routingu),
- podjęcie decyzji o dalszej obsłudze jednostki danych (odrzućcie, przyjęcie, wybór kolejnego węzła na trasie do ujścia).

Proces sprawdzania i przygotowania decyzji o przynależności jednostki danych do przepływu określa się mianem identyfikacji jednostek danych przepływu. Ze względu na to, że nagłówek jednostki danych zwykle nie zawiera informacji o jego przynależności do takiego czy innego przepływu, identyfikacja przepływu musi być wykonywana dla każdej napływającej jednostki danych. Podstawowym ograniczeniem implementacji różnych mechanizmów identyfikacji jednostek danych przepływu jest czas konieczny na porównanie i podjęcie decyzji; który jest ograniczony z góry przez górne ograniczenie czasu przetwarzania jednostki danych w węźle. Dodatkowym ograniczeniem czasu realizacji zadania identyfikacji jest konieczność jednoczesnej identyfikacji jednostek danych przepływu równocześnie dla bardzo dużej liczby przepływów.

Realizacja różnych koncepcji identyfikacji jednostek danych przepływu jest wyborem kompromisu pomiędzy szybkością działania procedury identyfikacji a ilością zasobów (zwłaszcza pamięci) niezbędnych do realizacji identyfikacji. Jedną z efektywniejszych technik identyfikacji jednostek danych przepływów, ze względu na złożoność obliczeniową i zajętość pamięci, jest technika funkcji mieszającej (ang. *hashing-based schemes*).

Mechanizmy identyfikacji jednostek danych przepływów w węzłach sieci, wykorzystujące techniki adresowania mieszającego, są względnie proste, tzn. wymagają:

- obliczenia wartości funkcji mieszającej (funkcji skrótu) dla napływającej jednostki danych,
- porównania obliczonej wartości funkcji skrótu dla jednostki danych z wartością funkcji skrótu dla przepływu, przechowywaną w węźle,
- i ewentualnie dodatkowych porównań obliczonej i przechowywanej wartości funkcji w razie wystąpienia kolizji.

Do skutecznych sposobów identyfikacji jednostek danych przepływu zalicza się także rozwiązania wykorzystujące właściwości wielomianów; w tym rozwiązania stosujące sumy kontrolne.

9.5.2. TECHNIKA FUNKCJI SKRÓTU

W wielu procesach przetwarzania informacji, zwłaszcza do przetwarzania wszelkiego rodzaju ewidencji, stosuje się tzw. struktury rangi 1, w których wartościami

prostymi są np. nazwy lub adresy danych. Strukturą rangi 1 jest ciąg danych rangi 1, uszeregowany zgodnie z uszeregowaniem zbioru nazw tych danych. Struktury takie rozmieszczane są często z zastosowaniem tzw. kodowania mieszającego (ang. *hash-coding*). Nazwy danych, tradycyjnie nazywane w tym kontekście kluczami, są przekształcane przez funkcję skrótu w liczby całkowite z danego przedziału; liczby te mogą być interpretowane jako adresy komórek, w których są zapisane dane struktury.

Podstawową ideą adresowania mieszającego jest umieszczanie każdego wystąpienia danej w miejscu, którego adres można obliczyć za pomocą pewnej funkcji. Wartość funkcji skrótu oblicza się dla wartości, która znajduje się w danym wystąpieniu i jest zazwyczaj wartością klucza głównego. Podstawowa korzyść ze stosowania adresowania mieszającego wynika z tego, że zwykle zakres wartości klucza jest znacznie szerszy od zakresu dostępnych adresów. Funkcja skrótu ogranicza straty pamięci w wyniku redukcji dowolnej liczby z zakresu zmienności klucza do zakresu zmienności adresu.

Podstawową zaletą adresowania mieszającego jest efektywne wykorzystanie pamięci; zwykle funkcję skrótu dobiera się tak, aby redukować długość adresu.

Redukcja długości adresu, będąca konsekwencją sposobu obliczania adresu, jest przyczyną kilku wad adresowania mieszającego: kolejność wystąpień rekordów z reguły nie jest zgodna z kolejnością klucza głównego, istnieje zatem możliwość wystąpienia kolizji, tzn. zdarzenia, że dwa różne wystąpienia danych mają ten sam adres. Z tego powodu w zastosowaniach funkcji skrótu konieczne jest uwzględnienie takich zdarzeń. Jednym ze sposobów ograniczania konsekwencji kolizji jest stosowanie hierarchii funkcji skrótu, zgodnie z założeniem, że pierwszy wyznaczony adres jest punktem startowym do poszukiwania wolnego adresu.

Jedną z prostszych, przykładowych metod konstrukcji funkcji skrótu, jest metoda Bella–Kamana.

Po oznaczeniu:

n – wartość liczbowa rozpatrywanego klucza,

m – liczba komórek przeznaczonych na rozmieszczenie struktury za pomocą funkcji skrótu (zakłada się, że m jest liczbą pierwszą),

i przyjęciu, dla uproszczenia, że adresy komórek przydzielanych rozpatrywanej strukturze, z której pochodzi dana, tworzą ciąg $\{1, 2, \dots, m - 1\}$, danej tej przydzielane jest słowo o adresie:

$$h_0 = r = n - mq,$$

gdzie:

$$q = \begin{cases} \left\lfloor \frac{n}{m} \right\rfloor & \text{gdy } \left\lfloor \frac{n}{m} \right\rfloor \neq 0 \\ 1 & \text{gdy } \left\lfloor \frac{n}{m} \right\rfloor = 0 \end{cases}.$$

Jeżeli komórka h_0 jest zajęta, tzn. gdy występuje konflikt adresów, rozpatrywane są kolejno komórki o adresach obliczanych rekurencyjnie według zasady:

$$h_{i+1} = (h_i + q) \bmod m.$$

Ze względu na to, że dla $q = 0$ zależność rekurencyjna definiuje zawsze ten sam adres, wprowadzane jest uzupełnienie:

$$q' = \begin{cases} q & \text{gdy } q \neq 0 \\ 1 & \text{gdy } q = 0 \end{cases}$$

i zamiast wyrażenia $h_{i+1} = (h_i + q) \bmod m$ stosuje się wyrażenie:

$$h_{i+1} = (h_i + q') \bmod m.$$

Jakość metody Bella–Kamana, mierzona średnią liczbą prób potrzebnych do odnalezienia adresu odpowiadającego zadanemu kluczowi w zależności od współczynnika zajętości, jest dość niska. Za użytecznością tej metody przemawia prostota obliczania poszczególnych wartości h_i .

9.5.3. FUNKCJA SKRÓTU W ZADANIU IDENTYFIKACJI JEDNOSTEK DANYCH PRZEPLYWU

Ustanowienie przez protokół RSVP rezerwacji zasobów w węźle sieci dla przepływu pozwala na obliczenie wartości funkcji skrótu dla tego przepływu. Jej argumentem jest zawartość nagłówka jednostek danych przepływu przyjętego do obsługi. Jeżeli wartość funkcji skrótu obliczona dla przepływu nie jest używana przez przepływy wcześniej przyjęte do obsługi, to wartość funkcji skrótu przepływu jest opisem stanu tego przepływu w tabeli rezerwacji. Jeżeli ta sama wartość funkcji skrótu została wcześniej przypisana innemu przepływowi, to występuje kolizja funkcji mieszającej (ang. *hash collision*).

Tak obliczona wartość funkcji skrótu dla przepływu jest następnie stosowana do identyfikacji jednostek danych napływających do węzła w czasie trwania przepływu przy założeniu, że wszystkie nagłówki jednostek danych tego samego przepływu są takie same. Obsługa kolizji wartości funkcji skrótu, obliczanej dla kolejno napływających jednostek danych, polega na odpowiednim ich oznakowaniu i obsłudze w tabeli rozwiązywania kolizji, w której przechowywane są nagłówki dla wszystkich przyjętych do obsługi przepływów o tej samej wartości funkcji skrótu. Każdy z nagłówków, przechowywanych w tabeli rozwiązywania kolizji, ma inny opis (wskaźnik) stanu przepływu w tabeli rezerwacji.

Identyfikacja jednostek danych danego przepływu, realizowana w węźle sieci, polega więc na zastosowaniu tej samej funkcji skrótu do obliczenia jej wartości dla nagłówków kolejnych jednostek danych przepływu. W zadaniu identyfikacji funkcja skrótu (adresowanie mieszające) realizuje zadania klasyfikacji jednostek danych z dokładnością do kolizji adresów.

Realizacja mechanizmu identyfikacji jednostek danych przepływu z wykorzystaniem funkcji skrótu jest dwuetapowa: pierwszy etap to obliczanie wartości funkcji skrótu, drugi natomiast to rozwiązywanie ewentualnych kolizji. Efektywność omawianego mechanizmu jest konsekwencją kompromisu pomiędzy szybkością obliczania wartości funkcji skrótu i rozstrzygnięcia kolizji. Zwiększanie rozmiarów tablicy wartości funkcji skrótu zmniejsza prawdopodobieństwo występowania kolizji, ich liczbę oraz szybkość obliczania wartości funkcji skrótu.

Efektywność funkcji skrótu, w zadaniach identyfikacji jednostek danych przepływu, mierzy się wartościami dwóch wskaźników:

1. Współczynnika kolizji, definiowanego jako procent przepływów z co najmniej jedną kolizją, określającego średnią jakość działania mechanizmu identyfikacji jednostek danych przepływu.

2. Maksymalnej liczby kolidujących przepływów, definiowanej jako maksymalna liczba przepływów, które mogą mieć taką samą wartość funkcji skrótu, określającego jakość działania identyfikacji jednostek danych przepływu w skrajnie niekorzystnym przypadku.

Przy założeniu, że n jest liczbą przepływów, tzn. wartością liczbową rozpatrywanego klucza, a m jest rozmiarem tablicy adresów, tzn. liczbą komórek przeznaczonych na rozmieszczenie struktury za pomocą funkcji skrótu, oraz że rozdział przepływów przez funkcję skrótu do tablicy adresów jest równomierny, wartość współczynnika kolizji (ang. *collision coefficient*) jest określona wyrażeniem:

$$w_{\text{kol}} = 1 - \frac{m \left[1 - \left(\frac{m-1}{m} \right)^n \right]}{n}$$

Charakter zależności szybkości kolizji od wartości parametrów n i m oznacza, że zwiększenie liczby przepływów, dla danego rozmiaru tablicy adresowej, powoduje wykładniczy wzrost wartości współczynnika kolizji. W praktyce działanie funkcji skrótu odbiega od teoretycznego ze względu na to, że rozkład ciągów binarnych w nagłówkach jednostek danych nie jest równomierny. W rzeczywistej sieci większość przepływów jest realizowana pomiędzy źródłami i ujściami, należącymi do ograniczonych podzbiorów zbiorów wszystkich źródeł i ujść sieci. Ograniczenie podzbioru aktywnych źródeł i ujść przepływów powoduje, że następuje koncentracja przepływów, w tym znaczeniu, że zawartości nagłówków jednostek danych tych przepływów są ciągami binarnymi o zbliżonych wartościach. Konsekwencją koncentracji przepływów jest obniżenie wartości teoretycznej skuteczności (jakości) działania funkcji skrótu.

W przypadku wyboru takiej funkcji skrótu, która gwarantuje ekstremalnie małą wartość współczynnika kolizji adresów, możliwa jest rezygnacja z implementacji mechanizmów obsługi kolizji. Rezygnacja z obsługi kolizji adresów jest jednym z możliwych rozwiązań, w których ceną prostoty i szybkości przetwarzania jednostek danych jest utrata precyzji identyfikacji jednostek danych przepływów.

W zadaniach identyfikacji przepływu mogą być stosowane różne rozwiązania adresowania mieszającego, które różnią się długością ciągów bitów nagłówka oraz konstrukcją funkcji skrótu. W przypadku pakietów IP do identyfikacji przepływu mogą być stosowane albo wszystkie bity nagłówka (adres źródła (32), adres ujścia (32), port źródła (16), port ujścia (16), identyfikator (8)), albo wybrane jego fragmenty. Gdy do identyfikacji przepływów wykorzystywane są wartości wszystkich pól nagłówka, funkcja skrótu jest obliczana dla $n = 104$.

W zależności od aplikacji i właściwości przepływów w sieci możliwa jest rezygnacja z niektórych elementów nagłówka, co oznacza zmniejszenie wartości n . Ograniczenie się do adresów sieciowych źródła i ujścia oznacza, że funkcja skrótu jest obliczana dla $n = 64$. Przyjęcie założenia, że do identyfikacji przepływów wystarczająca jest znajomość adresu sieciowego ujścia i adresu portu ujścia oznacza obliczanie wartości funkcji skrótu dla $n = 48$.

Zgodnie z oczekiwaniami, jakość działania funkcji skrótu, mierzona średnią wartością współczynnika kolizji, zmniejsza się wraz ze wzrostem wartości ilorazu długości ciągu binarnego do rozmiaru pola adresowego. Zasada ta jest zachowana pod warunkiem, że ciąg binarny jednoznacznie określa wszystkie możliwe przepływy.

Ograniczenie długości ciągów binarnych, przez rezygnację z reprezentowania w tych ciągach wszystkich pól nagłówka, pogarsza jakość działania funkcji skrótu. Rezygnacja z wybranych fragmentów nagłówka skraca czas wyznaczania wartości funkcji skrótu i zmniejsza rozmiar tablic adresowych, ale za cenę wzrostu niejednoznaczności reprezentacji tych przepływów, które różnią się wartościami pola nagłówka niereprezentowanymi w ciągu binarnym, przetwarzanym przez funkcję skrótu.

9.5.4. SUMA KONTROLNA JAKO FUNKCJA SKRÓTU

W celu identyfikacji przepływu można stosować sumę kontrolną CRC (ang. *Cyclic Redundancy Checking*). Jest to jeden z częściej stosowanych, bardzo skutecznych algorytmów detekcji błędów w systemach transmisji danych. Suma kontrolna (CRC-8, CRC-16, CRC-32) ma właściwości konwersji i redukcji długości adresów, tzn. właściwości funkcji skrótu. Właściwościami sumy kontrolnej CRC, szczególnie przydatnymi w zadaniach identyfikacji jednostek danych przepływu, są: względnie prosta realizacja układowa, minimalne opóźnienie wnoszone przez układowo realizowane zadania identyfikacji i niewielka nadmiarowość.

Istotą sumy kontrolnej CRC jest to, że suma ta jest resztą z procesu dzielenia. Wielomian, odpowiadający wybranej sekwencji bitów, jest dzielony przez wielomian generacyjny; iloraz jest odrzucany, a reszta z dzielenia jest obcinana do 32 bitów (w przypadku CRC-32). Dzielenie nie jest realizowane bezpośrednio; reszta z dzielenia jest wynikiem działania układów cyfrowych zawierających bramki EX-OR i rejestry przesuwne.

Ogólna zasada działania algorytmu CRC w zadaniach detekcji błędów polega na tym, że:

➤ wielomian generujący ciąg danych binarnych $P(x)$ jest dzielony przez wielomian generacyjny $G(x)$ algorytmu CRC (dla CRC-32 wielomian generacyjny jest następujący:

$$G(x) = x^{32} + x^{26} + x^{23} + x^{22} + x^{16} + x^{12} + x^{11} + x^{10} + x^8 + x^7 + x^5 + x^4 + x^2 + x + 1);$$

wynikiem dzielenia jest reszta z dzielenia, tzn. CRC_1 , a liczba bitów w reszcie z dzielenia CRC_1 jest równa największemu wykładnikowi w wielomianie generacyjnym $G(x)$,

➤ ciąg danych binarnych $P(x)$, uzupełniony o obliczoną w źródle sumę kontrolną CRC_1 , jest w odbiorniku ponownie dzielony przez wielomian generacyjny $G(x)$ i obliczana jest wartość sumy kontrolnej w odbiorniku, tzn. CRC_2 ,

➤ jeżeli wartość sumy kontrolnej CRC_2 obliczonej w odbiorniku jest równa sumie kontrolnej CRC_1 odebranej przez odbiornik $O(CRC_1)$, to podejmowana jest decyzja o bezbłędnej transmisji; w innym przypadku podejmowana jest decyzja o wystąpieniu błędu,

➤ wynik dzielenia jest równy przekazanej sumie kontrolnej.

Wykorzystanie sumy kontrolnej CRC w zadaniu identyfikacji jednostek danych przepływu polega na porównywaniu wartości sum kontrolnych, obliczanych dla ciągów binarnych, będących zawartością wszystkich lub wybranych pól nagłóweków napływających jednostek danych z adresami, będącymi wartościami sum kontrolnych zarezerwowanych dla przepływów przyjętych wcześniej do obsługi.

Algorytm CRC, ze względu na stochastyczne właściwości odwzorowywania, minimalizujące prawdopodobieństwo kolizji adresów, jest skutecznym algorytmem o właściwościach funkcji skrótu. Zakres zastosowań algorytmów CRC jest ograniczony dużą złożonością obliczeniową, która jest znacznie większa od złożoności obliczeniowej zwykłej funkcji skrótu dla tych samych wartości parametrów, tj. długości nagłówka jednostki danych i rozmiaru tablicy adresów.

Jakość działania algorytmu CRC jako funkcji skrótu, mierzona wartością współczynnika kolizji, jest zdecydowanie wyższa, w porównaniu z jakością działania prostych funkcji skrótu, i bliska teoretycznej. Algorytmy CRC gwarantują przede wszystkim dużo wyższą jakość działania dla tzw. najgorszego przypadku, tj. wtedy, gdy liczba kolidujących przepływów jest maksymalna.

9.5.5. PODWÓJNA FUNKCJA SKRÓTU

Rozwiązanie polegające na stosowaniu dwóch różnych funkcji skrótu, jest rozwiązaniem redukującym wartość współczynnika kolizji. Istotą rozwiązania jest to, że jeżeli pierwsza z funkcji skrótu generuje kolizję, to obliczana jest nowa wartość z wykorzystaniem drugiej funkcji skrótu. W przypadku podwójnej funkcji skrótu prawdopodobieństwo zdarzenia polegającego na kolizji adresów jest równe prawdopodobieństwu kolizji wartości drugiej funkcji skrótu.

W rozwiązaniu tym redukcję wartości współczynnika kolizji uzyskuje się za cenę dodatkowej złożoności obliczeniowej.

Jakość działania podwójnej funkcji skrótu, mierzona wartością współczynnika kolizji adresów, jest zbliżona i wyższa od jakości działania algorytmu CRC. Co więcej, złożoność obliczeniowa podwójnej funkcji skrótu jest znacznie mniejsza od złożoności obliczeniowej algorytmu CRC.

9.6. SZEREGOWANIE JEDNOSTEK DANYCH PRZEPIYWÓW

Zadaniem modułu szeregowania jednostek danych przepływu jest wymuszanie alokacji zasobów dla indywidualnych przepływów. Skutkiem różnego opóźnienia jednostek danych, wnoszonych przez sieć i jej elementy, możliwe jest występowanie przeciążeń węzłów, których konsekwencją są kolejki jednostek danych w węzłach pośrednich (routerach) na trasach pomiędzy systemami końcowymi. Wymuszanie alokacji zasobów polega na tym, że możliwa jest zmiana kolejności obsługi jednostek danych, należących do różnych przepływów, w porównaniu z kolejnością ich napływu do obsługi. Celem szeregowania jednostek danych jest podejmowanie decyzji o tym, które z jednostek danych, należących do różnych przepływów i oczekujących w kolejce, oraz w jakiej kolejności powinny uzyskać dostęp do zasobów.

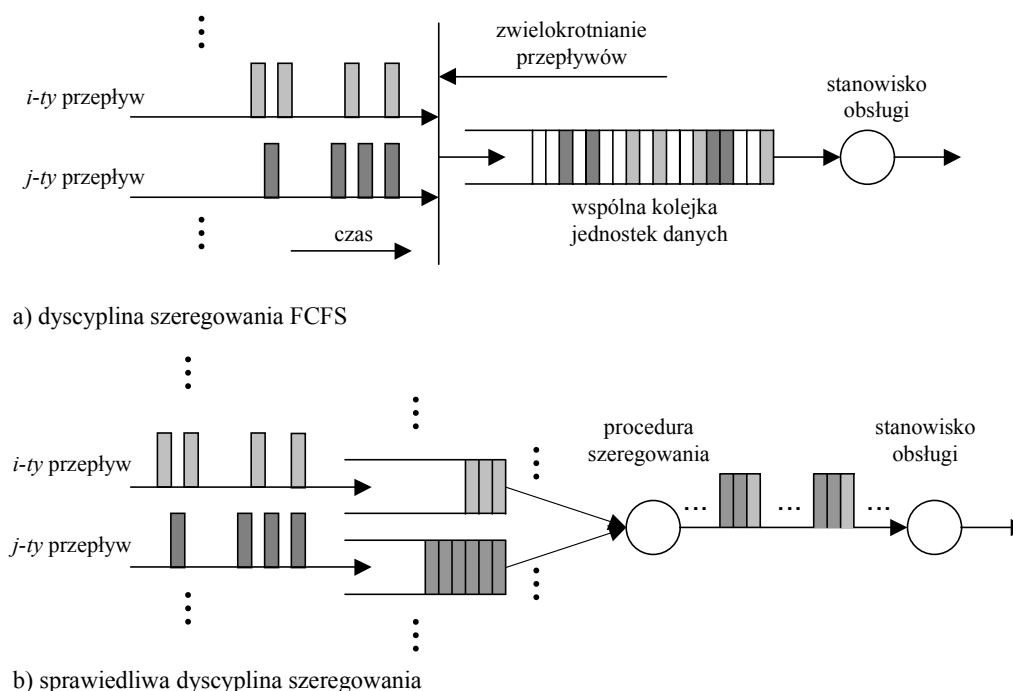
Wymuszanie alokacji zasobów oznacza konieczność stosowania różnych dyscyplin szeregowania jednostek danych, a w szczególności takich dyscyplin szeregowania, które nie zachowują kolejności napływu jednostek danych do obsługi.

Przykładem dyscypliny obsługi, która zachowuje kolejność napływu jednostek danych, jest dyscyplina FCFS (ang. *First Come First Served*). Ograniczenia dyscypliny szeregowania FCFS, ze względu na potrzebę różnicowania jakości obsługi różnych przepływów, jednocześnie obsługiwanych w węzle, są następujące:

- brak możliwości organizacji obsługi jednostek danych przepływów zgodnie z różnymi priorytetami przepływów,
- losowy charakter rozkładów czasów oczekiwania jednostek danych na obsługę,
- niesprawiedliwy sposób organizacji dostępu do zasobów, tzn. brak mechanizmu przeciwdziałania zawłaszczaniu kolejki, a tym samym stanowiska obsługi, przez przepływy o dużej szybkości.

Wymienione wady dyscypliny szeregowania FCFS, wynikające ze sposobu kolejowania jednostek danych, ograniczają bezpośrednią przydatność tej dyscypliny szeregowania i dyscyplin o podobnym sposobie działania, do zastosowań w sieciach z różnicowaniem klas ruchu i jakości usług. Brak możliwości obsługi z priorytetami, niedeterministyczny charakter opóźnień wnoszonych przez kolejkę FCFS oraz możliwość zawłaszczania zasobów są sprzeczne z właściwościami stanowiska obsługi pozwalającymi na różnicowanie jakości obsługi dla przepływów współdzielących zasoby sieci.

Eliminacja podanych ograniczeń dyscypliny FCFS, wynikających m.in. z utrzymywania tylko jednej kolejki dla jednostek danych wszystkich przepływów obsługiwanych w węźle, jest możliwa m.in. w wyniku organizacji oddzielnych kolejek jednostek danych dla każdego z przepływów, implementację różnych strategii kolejkowania jednostek danych oraz stosowania różnych algorytmów sprawiedliwego kolejkowania (szeregowanie) (ang. *fair queueing*), udostępniających zasoby węzła jednostkom danych zajmujących miejsca na czele poszczególnych kolejek (rysunek 9.11).



Rysunek 9.11. Organizacja obsługi przepływów dla dyscypliny szeregowania FCFS i różnych odmian sprawiedliwych dyscyplin szeregowania

W systemie obsługi z jedną kolejką i dyscypliną szeregowania FCFS (rysunek 9.11a) jednostki danych, należące do różnych przepływów, zajmują miejsca we wspólnej kolejce w kolejności napływu do węzła, tj. w kolejności rosnących czasów napływu do węzła. Zastosowanie dyscypliny FCFS oznacza, że kolejność wyprowadzania jednostek danych z węzła jest kolejnością wyprowadzania jednostek danych z kolejki i jest determinowana kolejnością napływu i zajmowania miejsca w kolejce.

W przypadku systemu obsługi z wieloma kolejkami o kolejności wyprowadzania jednostek danych z węzła decyduje procedura szeregowania implementowana w węźle sieci (rysunek 9.11b). Jednostki danych są wyprowadzane z każdej z kolejek zgodnie z kolejnością napływu do nich, ale procedura szeregowania decyduje

o względnych ilościach jednostek danych wyprowadzanych z poszczególnych kolejek i kolejności ich obsługi. W ten sposób procedura szeregowania może powodować, że kolejność wyprowadzania jednostek danych z węzła jest różna od kolejności ich napływu.

Działanie modułu szeregowania jednostek danych przepływów, wymagane w sieciach realizujących różne koncepcje jakości usług, jest zbliżone w skutkach do działania modułu dyspozytora (ang. *dispatcher*) i polega na:

- monitorowaniu liczby jednostek danych, nadsyłanych do węzła w ramach poszczególnych przepływów, i porównywaniu ich liczby z ilością zasobów zarezerwowanych na potrzeby obsługi danego przepływu, tzn. szacowanie możliwości ich obsługi,

- wysyłaniu jednostek danych, należących do poszczególnych przepływów, z węzła tylko wtedy, gdy nie nastąpiło wyczerpanie zasobów zarezerwowanych dla poszczególnych przepływów.

9.6.1. PODSTAWOWE WYMAGANIA DOTYCZĄCE MECHANIZMU SZEREGOWANIA JEDNOSTEK DANYCH

9.6.1.1. SEPARACJA PRZEPŁYWÓW I WSPÓLDZIELENIE ZASOBÓW

Sieci z komutacją łączy i z komutacją jednostek danych to dwa skrajne rozwiązania jednoczesnego zadania separacji przepływów i współdzielenia zasobów.

W sieci z komutacją łączy poszczególne przepływy, obsługiwane w sieci, są całkowicie separowane; każde z połączeń, a tym samym każdy z przepływów, korzysta z dedykowanych zasobów. Jeżeli dedykowane zasoby nie są w pełni wykorzystane, to są tracone.

W sieci z komutacją jednostek danych, dostarczających usługi bez gwarancji jakości, wszystkie zasoby sieci są współdzielone na zasadzie współzawodnictwa. Separacja przepływów nie występuje w żadnej formie. Brak jakiegokolwiek separacji przepływów umożliwia równomierny rozdział zasobów pomiędzy przepływy i maksymalizację efektywności wykorzystania zasobów, ale nie pozwala na różnicowanie jakości dostarczanych usług.

Głównym zadaniem szeregowania jest efektywne, kontrolowane współdzielenie zasobów sieci polegające na osiąganiu kompromisu pomiędzy separacją przepływów, niezbędną do indywidualizowania jakości usług, a współdzieleniem zasobów, wymaganym do maksymalizacji efektywności wykorzystania zasobów.

Rozwiązania realizujące omawiany kompromis pomiędzy separacją przepływów a współdzieleniem zasobów charakteryzują możliwości jednoczesnego:

- emulowania separacji przepływów,
- udostępniania zasobów, zarezerwowanych i niewykorzystywanych przez jedno przepływy, innym przepływowom.

9.6.1.2. OGRANICZENIA OPÓŹNIEŃ

W sieciach dostarczających usługi o ustalonej jakości konieczna jest implementacja mechanizmów umożliwiających sterowanie wartościami opóźnień wnoszonych przez sieci.

Dwa skrajne rodzaje ograniczeń opóźnienia w sieciach, tzn. deterministyczne i statystyczne, są konsekwencją przyjęcia rozwiązań opartych na separacji przepływów lub współdzieleniu zasobów.

Rozwiązania gwarantujące deterministyczne ograniczenia opóźnień jednostek danych wprowadzają maksymalną separację przepływów za cenę znacznej redukcji możliwości stosowania zwielokrotniania statystycznego przepływów. W razie konieczności zapewnienia deterministycznych ograniczeń opóźnienia jednostek danych konieczne jest stosowanie nadmiarowej rezerwacji zasobów, w ilościach gwarantujących zachowanie żądanej jakości usług dla tzw. najgorszego przypadku. Co więcej, zachowanie deterministycznego ograniczenia opóźnienia jednostek danych wnoszonego przez sieć wymusza stosowanie mechanizmów, których zadaniem jest ograniczenie „wybuchowości” źródła ruchu.

Statystyczne ograniczenia opóźnienia jednostek danych są charakterystyczne dla rozwiązań opartych na współdzieleniu zasobów; w tym przypadku ograniczenia opóźnienia jednostek danych są obarczone konsekwencjami wad i zalet różnych sposobów współdzielenia zasobów.

9.6.1.3. ALOKACJA ZASOBÓW

Gdy ilość dostępnych zasobów jest mniejsza niż suma żądanych zasobów, rozwiązaniem jest albo odrzucanie przepływów, albo sprawiedliwe współdzielenie (alokacja) zasobów (ang. *fairly-sharing*) pomiędzy przepływy rywalizujące o dostęp do zasobów. W przypadku odrzucania przepływów możliwe jest zachowanie jakości usług dla przepływów pozostawionych do obsługi w sieci i całkowita degradacja jakości usług dla przepływów odrzucanych. W przypadku stosowania algorytmów sprawiedliwego podziału zasobów, polegającego na gwarancji dostępu do zasobów wszystkim przepływowi w ilościach zależnych od udziału zapotrzebowania danego przepływu w sumarycznym zapotrzebowaniu wszystkich przepływów na zasoby, obsługiwane są wszystkie przepływy, ale średnia jakość usług jest niższa.

Jednym z najprostszych przykładów algorytmu sprawiedliwego szeregowania (kolejkowania), a tym samym sprawiedliwego rozdziału zasobów pomiędzy przepływy, jest algorytm obsługi cyklicznej RR (ang. *Round Robin*). Sprawiedliwość algorytmu obsługi cyklicznej wynika z tego, że w każdym cyklu każdy z przepływów, którego jednostki danych są obsługiwane zgodnie z dyscypliną FCFS, otrzymuje dostęp do zasobów umożliwiający wysłanie jednej (lub wielokrotności jednej) jednostki danych.

Sprawiedliwość działania algorytmu RR zależy od różnic w długości jednostek danych w różnych przepływach; im większe różnice długości, tym mniejsza sprawie-

dliwość algorytmu. Różne modyfikacje algorytmu RR, których celem jest poprawa sprawiedliwości działania, sprowadzają się do stosowania rozwiązań wymuszających zrównanie długości porcji danych wprowadzanych do obsługi.

Rozwiązaniem idealnie sprawiedliwym jest algorytm BRFQ (ang. *Bit-Round Fair Queueing*), zgodnie z którym w każdym kolejnym cyklu obsługiwany jest jeden bit jednostki danych z każdej niepustej kolejki, utożsamianej z jednym z przepływów. Skrócenie obsługiwanych porcji danych do jednego bitu oznacza, że dyskretny strumień jednostek danych jest aproksymowany ciągłym strumieniem bitów. Algorytm BRFQ jest praktyczną implementacją dyscypliny szeregowania z podziałem czasu procesora (ang. *processor sharing*).

Algorytm RR zapobiega zawłaszczeniu stanowiska obsługi przez przepływy o względnie dużym zapotrzebowaniu na zasoby, w porównaniu z zapotrzebowaniem innych przepływów, współdzielących te zasoby. Zapobieganie zawłaszczeniu zasobów przez zachłanne przepływy objawia się wzrostem długości kolejek jednostek danych przepływów zachłannych, a tym samym zwiększeniem opóźnienia kolejkowania, które można interpretować jako wartość odpowiedniej funkcji kary.

Innym przykładem sprawiedliwego rozdziału zasobów pomiędzy współzawodniczące przepływy jest metoda alokacji maksymalno-minimalnej (ang. *max-min fair sharing*). Idea tej metody to próba maksymalizacji minimalnej alokacji zasobów dla przepływów, których zapotrzebowanie na zasoby nie jest zaspokajane w całości. Podstawowe zasady implementacji metody alokacji maksymalno-minimalnej są następujące:

- zasoby są alokowane dla przepływów uporządkowanych w kolejności rosnącego zapotrzebowania na zasoby,
- zasoby są alokowane dla przepływów w ilościach nie większych niż zapotrzebowanie przepływów,
- przepływy, których zapotrzebowanie na zasoby nie jest zaspokajane w całości, otrzymują jednakowe części zasobów.

Części zasobów (np. pojemności), alokowanych dla przepływów zgodnie z metodą alokacji maksymalno-minimalnej, wyznacza się następująco:

1. Początkowy sprawiedliwy podział zasobów jest ilorazem całkowitej ilości zasobów i całkowitej liczby przepływów.
2. Wszystkim przepływom, których zapotrzebowanie na zasoby jest:
 - równe lub mniejsze od początkowego sprawiedliwego podziału zasobów, przydzielane są zasoby w ilościach równych zapotrzebowaniu,
 - większe od początkowego sprawiedliwego podziału zasobów, przydzielane są zasoby w ilościach równych początkowemu sprawiedliwemu podziałowi.
3. Całkowita ilość zasobów i całkowita liczba przepływów są pomniejszane, odpowiednio, o sumę przydzielonych zasobów i liczbę przepływów, którym zasoby zostały przydzielone w ilościach równych zapotrzebowaniu.
4. Bieżący sprawiedliwy podział zasobów jest ilorazem pozostałej (po pomniejszeniu) ilości zasobów i pozostałej (po pomniejszeniu) liczby przepływów.

5. Wszystkim przepływom, których zapotrzebowanie na zasoby jest:

- równe lub mniejsze od sumy bieżącego i wcześniejszych sprawiedliwych podziałów zasobów, przydzielane są zasoby równe zapotrzebowaniu,
- większe od sumy bieżącego i wcześniejszych sprawiedliwych podziałów zasobów, przydzielane są zasoby w ilościach równych sumom bieżącego i wcześniejszych sprawiedliwych podziałów zasobów.

6. Kolejne iteracje wyznaczania bieżącego sprawiedliwego podziału zasobów są powtarzane dopóty, dopóki:

- zapotrzebowanie wszystkich pozostałych (po wcześniejszych alokacjach) przepływów jest większe od sumy wcześniejszych i aktualnego, bieżącego podziału zasobów; w takim przypadku wszystkim pozostałym przepływom przydzielane są zasoby w ilości równej sumie wcześniejszych i bieżącego sprawiedliwych podziałów zasobów.

Zgodnie z omawianą metodą wszystkim przepływom o zapotrzebowaniu równym lub mniejszym od sprawiedliwego podziału zasobów przydzielane są zasoby w żądanych ilościach, podczas gdy zasoby przyznane w ramach sprawiedliwego podziału i niewykorzystane przez te przepływy są alokowane dla pozostałych przepływów, których zapotrzebowanie na zasoby jest większe od ilości wynikającej ze sprawiedliwego podziału.

W omawianej wersji metody alokacji maksymalno-minimalnej wszystkie rywalizujące o zasoby przepływy są równoprawne.

Metoda ta, po odpowiednich modyfikacjach, może być stosowana do alokacji zasobów dla przepływów o różnych priorytetach (wagach).

9.6.1.4. PODSTAWOWE ZASADY SZEREGOWANIA JEDNOSTEK DANYCH PRZEPLÝWÓW

W zadaniach szeregowania jednostek danych przepływów, wspierających różne koncepcje jakości usług, stosuje się różne sposoby wymuszania dostępu jednostek danych do zasobów.

9.6.1.4.1. SZEREGOWANIE ZACHOWUJĄCE PRACĘ I NIEZACHOWUJĄCE PRACY

W większości systemów obsługi jednostek danych przepływów stanowisko obsługi jest wolne tylko wtedy, gdy kolejka jednostek danych oczekujących na obsługę (transmisję) jest pusta. Algorytmy szeregowania o takiej właściwości są określane mianem algorytmów szeregowania zachowujących pracę (ang. *work conserving*). Większość algorytmów szeregowania i regulaminów obsługi należy do tej grupy algorytmów.

Inną grupą algorytmów szeregowania są algorytmy niezachowujące pracy (ang. *non-work conserving*). W systemach obsługi z takimi algorytmami szeregowania jednostki danych są obsługiwane (transmitowane) w węzłach wtedy, gdy zostaną wybrane do obsługi, tzn. gdy wymuszona zostanie alokacja zasobów. Jeżeli żadna z jednostek danych oczekujących na obsługę w kolejce nie zostanie wybrana do obsługi, to stanowisko obsługi pozostaje wolne, mimo że w kolejce znajdują się jednostki danych oczekujące na obsługę.

Możliwe zastosowania algorytmów niezachowujących pracy to zadania:

- redukcji zmienności opóźnienia jednostek danych wnoszonych przez sieć,
- redukcji pojemności pamięci buforowych w węzłach sieci,
- wygładzania przepływów jednostek danych.

W przypadku zastosowania algorytmów szeregowania niezachowujących pracy do redukcji zakresu zmienności (fluktuacji) opóźnienia działanie algorytmu polega na tym, że kolejna jednostka danych, oczekująca w kolejce na obsługę, jest wybierana po upływie pewnego czasu od chwili obsłużenia poprzedniej jednostki danych. Zmniejszanie fluktuacji opóźnienia jednostek danych odbywa się za cenę zmniejszenia stopnia wykorzystania zasobów (np. pojemności łącza) i zwiększenia średniego opóźnienia jednostek danych przepływu.

9.6.1.4.2. PRIORYTETOWANIE

Priorytetowanie jest rozwiązaniem powszechnie stosowanym w zadaniach szeregowania i obsługi jednostek danych przepływów w sieciach.

W przypadku stosowania schematu prostego priorytetowania (ang. *simple priority*) wprowadzana jest pewna liczba poziomów priorytetu; wyższy priorytet zwykle poprzedza priorytet niższy. Jeżeli możliwa jest transmisja (obsługa) jednostek danych, to zastosowanie prostego priorytetu do szeregowania jednostek danych oznacza, że w pierwszej kolejności będą obsługiwane jednostki danych o najwyższym priorytecie. Jednostki danych o niższym priorytecie będą obsługiwane tylko wtedy, gdy w kolejce nie ma jednostek danych o priorytetach wyższych.

Implementacja schematu prostego priorytetowania w zadaniach szeregowania jest bardzo prosta, ale jego stosowanie musi być poprzedzone wnikliwą analizą. Rozbudowany system prostego priorytetowania może bowiem doprowadzić do sytuacji, kiedy przepływy o względnie niskich priorytetach mogą być praktycznie pozbawione dostępu do zasobów, co może prowadzić do znacznej degradacji jakości ich obsługi.

Proste priorytetowanie stosowane jest zatem rzadko, zwykle do priorytetowania przepływów generowanych przez aplikacje krytyczne (ang. *mission-critical application*), takie jak np. zadania zarządzania siecią. W przypadku stosowania schematu prostego priorytetowania jego zakres jest ściśle reglamentowany przez procedury sterowania dostępem. Celem reglamentacji jest ograniczenie udziału ruchu o najwyższym priorytecie w całkowitym ruchu obsługiwanym w sieci.

9.6.1.5. RODZAJE ALGORYTMÓW SZEREGOWANIA JEDNOSTEK DANYCH

Algorytmy szeregowania jednostek danych przepływów w sieciach z komutacją jednostek danych dzieli się na trzy grupy:

1. Algorytmy sprawiedliwego szeregowania (ang. *fair queueing algorithms*).
2. Algorytmy z gwarantowanym czasem obsługi (ang. *deadline based algorithms*).
3. Algorytmy warunkowego szeregowania (ang. *rate based algorithms*).

9.6.1.5.1. ALGORYTMY SPRAWIEDLIWEGO SZEREGOWANIA

Działanie algorytmów sprawiedliwego szeregowania jednostek danych przepływu opiera się na założeniu, że część zasobów (np. pojemności łącza) przydzielanych danemu przepływowi jest reprezentowana przez liczbę rzeczywistą oznaczającą wartość współczynnika wagowego (ang. *weight coefficient*).

Algorytm sprawiedliwego szeregowania przydziela zasoby przepływowi aktywnym proporcjonalnie do wartości ich wag. Każdy przepływ jest upoważniony do dostępu do zasobów proporcjonalnie do przypisanej mu wagi. Ilość przydzielonych przepływowi zasobów, w porównaniu z zapotrzebowaniem na nie, jest podstawą kwalifikacji przepływów do dwóch grup:

1. Przepływów bez zaległości (ang. *non-backlogged flow*), tzn. takich, którym zostały przydzielone zasoby większe lub równe od aktualnego zapotrzebowania; skutkiem tego kolejka jednostek danych przepływu oczekujących na obsługę jest pusta.

2. Przepływów z zaległościami (ang. *backlogged flow*), tzn. przepływów, którym przydzielone zostały zasoby mniejsze od aktualnego zapotrzebowania na zasoby; skutkiem tego wartość długości kolejki jednostek danych przepływu oczekujących na obsługę jest różna od zera.

Jeżeli przepływ bez zaległości nie wykorzystuje w pełni przydzielonych mu zasobów, to niewykorzystane (nadmiarowe) zasoby są przydzielane przepływowi z zaległościami, proporcjonalnie do wag wszystkich przepływów z zaległościami. Zgodnie z algorytmem sprawiedliwego szeregowania każdy z przepływów upoważnionych do współdziałania w korzystaniu z zasobów (w procesie rezerwacji) proporcjonalnie do przypisanej mu wagi, ma gwarancję otrzymania zarezerwowanych zasobów i możliwość uzyskania dodatkowych. Gwarancja dostępu do zarezerwowanych zasobów w każdym węźle, znajdującym się na trasie od źródła do ujścia, jest równoznaczna z możliwością gwarantowania ograniczonych z góry opóźnień jednostek danych przepływów na trasach, tzn. możliwością gwarantowania podstawowej właściwości jakości usług w koncepcji usług zintegrowanych.

9.6.1.5.2. ALGORYTMY SZEREGOWANIA Z NIEPRZEKRACZALNYM CZASEM OBSŁUGI

W algorytmach szeregowania z nieprzekraczalnym czasem obsługi kolejność obsługi jednostek danych jest zgodna z uporządkowaniem jednostek danych według niemalejących wartości nieprzekraczalnego czasu obsługi EDF (ang. *Earliest Deadline First*), tzn. jako pierwsza jest obsługiwana jednostka danych z najmniejszą wartością nieprzekraczalnego czasu obsługi. Do działania algorytmu wymagane jest przypisanie każdej jednostce danych jednoznacznej wartości parametru, jakim jest nieprzekraczalny (ograniczony z góry) czas obsługi (ang. *deadline*).

Cechą charakterystyczną działania omawianego algorytmu jest rozłączność górnego ograniczenia opóźnienia jednostek danych i alokacji zasobów, tzn. ilość zasobów przydzielonych dla przepływu nie determinuje jednoznacznie wartości górnego ograniczenia opóźnienia jednostek danych wnoszonego przez sieć. Cecha ta powoduje

m.in. to, że omawiane algorytmy na ogół nie są algorytmami sprawiedliwego szeregowania jednostek danych, tzn. nadmiarowe zasoby nie są rozdzielane sprawiedliwie pomiędzy przepływy.

Możliwość przypisywania jednostkom danych każdego z przepływów wartości nieprzekraczalnego czasu obsługi, niezależnie od ilości przypisanych danemu przepływowi zasobów powoduje, że kolejności napływu i wypływu jednostek danych do i z węzła, należących do tych samych i różnych przepływów, mogą być różne.

W algorytmach, w których nie ma możliwości zmiany kolejności napływających i obsługiwanych jednostek danych, względna ilość zasobów (udział w całkowitej ilości zasobów) przyznanych przepływowi determinuje wartość górnego ograniczenia opóźnienia jednostek danych. Zmniejszenie ilości zasobów powoduje wzrost wartości górnego ograniczenia opóźnienia.

Zastosowanie algorytmu z nieprzekraczalnym czasem obsługi, tzn. algorytmu szeregowania jednostek danych wnoszącego niezależność górnego ograniczenia opóźnienia od ilości przyznanych zasobów, wymaga względnie złożonych systemów sterowania dostępem. Złożoność ta wynika z konieczności implementacji dwukryterialnego sterowania dostępem, tzn. sterowania, które jednocześnie:

- zapobiega przydzielaniu przepływowi zasobów w ilościach większych od całkowitej ilości zasobów,
- nadzoruje realizowalność obsługi jednostek danych przepływów zgodnie z przypisywanymi wartościami nieprzekraczalnego czasu obsługi.

9.6.1.5.3. ALGORYTMY SZEREGOWANIA WARUNKOWEGO

Algorytmy szeregowania warunkowego spełniają jednocześnie dwa zadania: sterowania przepływem jednostek danych i właściwego ich szeregowania.

Sterownik służy do wyznaczania warunków czasowych dla napływających jednostek danych przepływu, które następnie są wybierane do obsługi zgodnie z implementowanym algorytmem szeregowania. W ten sposób sterownik jednostkami danych przepływu realizuje zadanie formowania (kształtowania) strumienia jednostek danych wprowadzanych do obsługi.

Różne sterowniki umożliwiają realizację zadań kształtowania ruchu, w tym m.in.:

- regulację szybkości napływu jednostek danych, a tym samym kompensowanie różnic szybkości napływu i wypływu jednostek danych z węzła,
- ograniczenie maksymalnej szybkości jednostek danych, polegające na spowalnianiu wyprowadzania jednostek danych w tych przedziałach czasu, w których jednostki danych napływają (są generowane) z maksymalną szybkością,
- ograniczenie zakresu zmienności (fluktuacji) opóźnienia wprowadzanych jednostek danych, pozwalające na ograniczenie różnic opóźnień jednostek danych wnoszonych przez te elementy trasy przepływu, które poprzedzają dany węzeł na trasie.

Złożenie różnych algorytmów sterowania przepływem jednostek danych oraz różnych algorytmów szeregowania jednostek danych wyprowadzanych ze sterownika daje możliwość komponowania algorytmów szeregowania warunkowego o różnych

właściwościach, dla szerokiego zakresu zmienności wartości parametrów źródeł przepływów i jakości usług. W szczególności istnieje możliwość budowy algorytmów szeregowania warunkowego zachowujących pracę i niezachowujących jej.

9.7. ALGORYTMY WAŻONEGO SPRAWIEDLIWEGO SZEREGOWANIA

Wynikiem działania algorytmów sprawiedliwego szeregowania FQ (ang. *Fair Queueing*) w węźle jest równomierny rozdział zasobów, w ilościach odwrotnie proporcjonalnych do liczby aktywnych przepływów (połączeń) w tym węźle.

W sieciach, w których możliwa jest zróżnicowana jakość usług dla różnych przepływów, podstawowym zadaniem dyscyplin szeregowania stosowanych w węzłach sieci jest przydział różnych ilości zasobów (pojemności) różnym przepływowi, umożliwiającym zarówno realizację żądanej jakości usług, jak i różnicowanie jakości usług dla różnych przepływów (połączeń) w sieci.

Niezależnie od różnego rozumienia gwarancji jakości usług w sieciach, naturalne jest wymaganie, aby jakość usług sieci była przewidywalna. W przypadku obsługi w sieci aplikacji czasu rzeczywistego przewidywalność jakości usług dotyczy np. znajomości górnego ograniczenia wartości opóźnienia wnoszonego przez sieć w czasie transferu jednostek danych na trasie od źródła do ujścia. Do podstawowych i najczęściej stosowanych dyscyplin sprawiedliwego szeregowania, dających możliwość przewidywania jakości usług komunikacyjnych dla ruchu w sieciach z komutacją jednostek danych, w tym dla ruchu w czasie rzeczywistym, zalicza się dyscypliny: z uogólnionym podziałem czasu procesora GPS (ang. *Generalized Processor Sharing*) oraz z nieprzekraczalnym czasem obsługi EDF (ang. *Earliest Deadline First*).

W zadaniach alokacji stałych pojemności dla przepływów i gwarantowania górnego ograniczenia opóźnienia jednostek danych przepływów stosowane są różne algorytmy ważonego sprawiedliwego szeregowania (kolejkowania) WFQ (ang. *Weighted Fair Queueing*), które umożliwiają przydział różnych ilości zasobów dla różnych aktywnych przepływów.

Algorytmy WFQ są praktycznymi realizacjami dyscypliny szeregowania z uogólnionym podziałem czasu procesora GPS. Praktyczna realizacja dyscypliny szeregowania GPS przez algorytm WFQ polega na tym, że WFQ emuluje dyscyplinę szeregowania GPS, ale do obsługi wybierane są jednostki danych. Algorytmy WFQ charakteryzują różną złożoność obliczeniową i dokładność otrzymywanych rozwiązań.

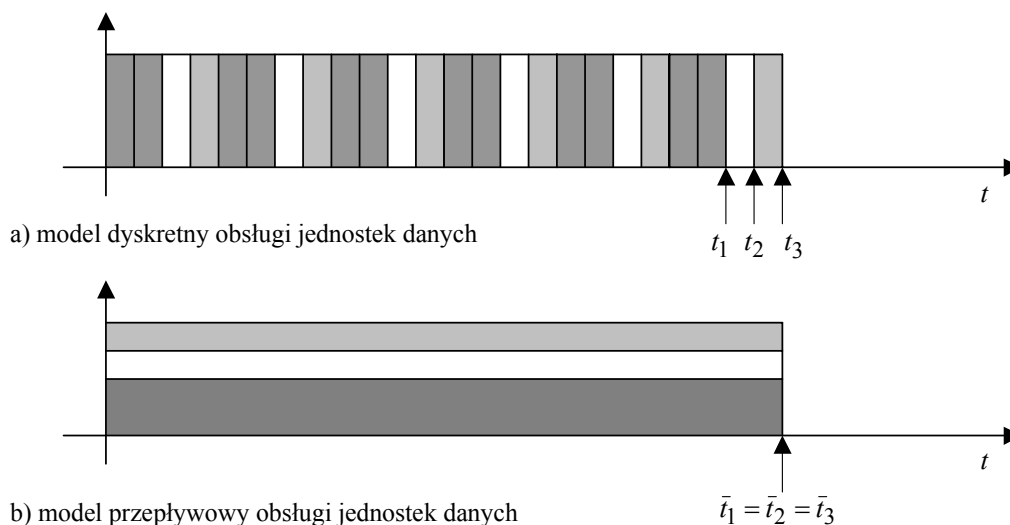
9.7.1. MODEL PRZEPLÝWOWY

W rzeczywistych sieciach komputerowych obsługa jednostek danych przepływu wiąże się z ich indywidualnym przetwarzaniem, tzn. jeden po drugim. W takim przypadku system obsługi jednostek danych przepływu na trasie jest potokowym syste-

mem obsługi (ang. *flow shop*) – wszystkie jednostki danych przepływu muszą być transmitowane w tej samej kolejności przez wszystkie łącza transmisyjne składające się na trasę źródło–ujście. Naturalnym modelem takiego systemu obsługi jednostek danych jest model dyskretny.

Ze względu na ograniczone możliwości stosowania dokładnych modeli wielu różnych systemów obsługi, powszechnie używa się przybliżonych opisów systemów. Jednym z najbardziej użytecznych i najczęściej stosowanych jest model przepływowy. Przyjęcie przybliżonego opisu ruchu w modelu przepływowym pozwala na uproszczenie opisu i analizy systemów obsługi za cenę dokładności opisu zdarzeń zachodzących w czasie.

Podstawą tej przybliżonej metody jest rezygnacja z opisu przepływów jako strumieni indywidualnych jednostek danych. Odpowiada to przyjęciu założenia, że dyskretny model napływu jednostek danych do systemu obsługi można zastąpić ciągłym procesem dyfuzji o podobnych właściwościach. Ruch w modelu przepływowym sieci jest strumieniem przepływów charakteryzowanych szybkością przepływów, a opis przepływów w postaci procesów zliczania zdarzeń zastępuje się opisem w postaci wolumenu ruchu. Pozwala to dalej na przyjęcie założenia, że przepływy w sieci są nieskończenie podzielne oraz że w węźle jednocześnie może być obsługiwanych wiele przepływów.



Rysunek 9.12. Model dyskretny i ciągły stanowiska obsługi jednostek danych

Przyjmowanie przybliżonego opisu przepływów wymusza zmianę interpretacji niektórych wyników analizy systemów obsługi przepływów i jednostek danych tych przepływów. Jednym z przykładów takiej zmiany jest interpretacja czasu zakończenia obsługi jednostek danych, należących do różnych przepływów, w modelu dokładnym (modelu zliczania) i modelu przybliżonym (modelu przepływowym).

W pierwszym z wymienionych modeli czas zakończenia obsługi różnych przepływów w tym samym węźle sieci jest czasem zakończenia obsługi ostatnich jednostek danych tych przepływów. Czasy te są różne: $t_1 \neq t_2 \neq t_3$ (rysunek 9.12a). W drugim natomiast czas zakończenia obsługi wielu różnych przepływów w tym samym węźle sieci może być taki sam, tzn. czasy zakończenia obsługi ostatnich jednostek danych różnych przepływów są takie same ($\bar{t}_1 = \bar{t}_2 = \bar{t}_3$ na rysunku 9.12b). Oznacza to, że czasy zakończenia obsługi jednostek danych w modelu dyskretnym są nie większe niż czasy zakończenia obsługi jednostek danych w modelu przepływowym.

9.7.2. SPRAWIEDLIWY ROZDZIAŁ ZASOBÓW

Jednym z najtrudniejszych zadań sterowania przepływami w sieciach jest tzw. sprawiedliwe traktowanie (ang. *fairly treating*) wszystkich użytkowników sieci wtedy, gdy zachodzi konieczność odrzucania:

- wywołań z wykorzystaniem mechanizmów sterowania prewencyjnego na etapie żądania dostępu do zasobów sieci,
- jednostek danych przepływów z wykorzystaniem mechanizmów sterowania reakcyjnego na etapie obsługi jednostek danych przepływów generowanych w ramach wywołań wcześniej przyjętych do obsługi.

Jeżeli, ze względu na ograniczone zasoby sieci, całkowity ruch teleinformatyczny generowany jednocześnie przez wiele źródeł musi być ograniczony, to istotne jest, aby ograniczanie było przeprowadzone sprawiedliwie, tzn. aby ograniczenia dotyczyły wszystkich składowych całkowitego ruchu. Wprowadzenie sprawiedliwych ograniczeń poszczególnych składowych ruchu całkowitego jest związane z obniżeniem wartości wskaźników wykorzystania zasobów. Zadania maksymalizacji np. przepustowości sieci, tj. maksymalizacji wartości współczynnika wykorzystania pojemności są zwykle sprzeczne z zadaniami sprawiedliwego rozdziału ograniczonej pojemności pomiędzy przepływy współzawodniczące o te zasoby.

Problem wyboru pomiędzy optymalnym wykorzystaniem zasobów sieci, a sprawiedliwym rozdziałem zasobów sieci powstaje zwykle wtedy, gdy zasoby sieci są przeciążone i konieczne jest ograniczenie obciążenia zasobów. Ograniczenie może być konsekwencją działania mechanizmów sterowania prewencyjnego lub reakcyjnego. Rozwiązanie zadania sprawiedliwego rozdziału zasobów oznacza zwykle znalezienie kompromisu pomiędzy sprawiedliwym rozdziałem zasobów i ograniczaniem dostępu do zasobów tych przepływów, które ponoszą największą odpowiedzialność za powstanie przeciążenia. Uzyskanie rozwiązań zadań sprawiedliwego rozdziału zasobów o podanych właściwościach jest możliwe m.in. przez zastosowanie różnych, praktycznych implementacji dyscyplin: z uogólnionym podziałem czasu procesora GPS oraz z nieprzekraczalnym czasem obsługi EDF.

W warunkach deterministycznych jakości działania dyscyplin szeregowania GPS i EDF są porównywalne, w warunkach statystycznych dyscyplina EDF zapewnia wyższy, w porównaniu z dyscypliną GPS, stopień wykorzystania zasobów.

9.7.2.1. DYSCYPLINA SZEREGOWANIA Z UOGÓLNIONYM PODZIAŁEM CZASU PROCESORA

Podstawowa zasada działania dyscypliny szeregowania GPS polega na tym, że każdy i -ty przepływ, przechodzący przez serwer GPS, obsługujący jednocześnie n przepływów i działający ze stałą szybkością C , jest charakteryzowany dodatnią liczbą rzeczywistą ϕ_i , taką, że dla każdego przedziału czasu $(\tau, t]$, w którym i -ty przepływ jest przepływem z zaległościami, spełniony jest warunek:

$$\frac{s_i(\tau, t)}{s_j(\tau, t)} \geq \frac{\phi_i}{\phi_j} \text{ dla } j = 1, 2, \dots, n,$$

gdzie $s_j(\tau, t)$ jest ilością ruchu j -tego przepływu obsługiwanego przez serwer GPS w przedziale czasu $(\tau, t]$.

Podstawowe właściwości dyscypliny szeregowania GPS to:

- gwarancja minimalnej jakości obsługi dla każdego z obsługiwanych przepływów,
- sprawiedliwy rozdział zasobów pomiędzy przepływy,
- możliwość wyprowadzenia prostych reguł sterowania dostępem w przypadku deterministycznych ograniczeń całkowitego opóźnienia jednostek danych w sieci.

Ze względu na podane właściwości dyscyplina szeregowania GPS jest podstawą wielu różnych algorytmów szeregowania, różniących się jakością działania, złożonością obliczeniową, złożonością implementacji i stopniem wykorzystywania zasobów.

9.7.2.2. DYSCYPLINA SZEREGOWANIA Z NIEPRZEKRACZALNYM CZASEM OBSŁUGI

Zasada działania dyscypliny szeregowania EDF polega na tym, że w każdym serwerze EDF każdemu i -temu przepływowi przypisana jest lokalna wartość górnego ograniczenia opóźnienia jednostek danych d_i . Jednostki danych i -tego przepływu, napływające do serwera EDF w chwilach t , są znakowane nieprzekraczalnymi (ang. *deadline*) wartościami opóźnienia $t + d_i$, a następnie obsługiwane w kolejności niemalejących nieprzekraczalnych wartości opóźnienia.

Dyscyplina szeregowania EDF jest optymalną dyscypliną, gwarantującą zachowanie górnego ograniczenia opóźnienia, którego różne wersje wykazują wyższą jakość działania od algorytmu GPS. Ograniczeniem zastosowania dyscypliny EDF w zadaniach gwarantowania jakości usług jest złożoność, odpowiadającej algorytmowi EDF, procedury sterowania dostępem do sieci.

Znaczenie dyscyplin szeregowania GPS i EDF w sieci z gwarancją jakości usług wynika z ich właściwości oraz jakości działania w warunkach:

- deterministycznych, kiedy gwarantują zachowanie górnego ograniczenia opóźnienia jednostek danych dla tzw. najgorszego przypadku,

➤ statystycznych, kiedy gwarantują zachowanie zadanej wartości prawdopodobieństwa zdarzenia, że opóźnienie jednostek danych jest nie większe od zadanego górnego ograniczenia wartości opóźnienia.

9.7.3. MIARY JAKOŚCI DYSCYPLIN SPRAWIEDLIWEGO SZEREGOWANIA

Serwer dyscypliny szeregowania jest tzw. serwerem zachowującym pracę, tzn. jest zawsze zajęty wtedy, gdy kolejka jednostek danych oczekujących na obsługę nie jest pusta. Innymi słowy, jeżeli i -ty przepływ ($i = 1, 2, \dots, n$) nie jest aktywny, tzn. nie korzysta z przyznanego mu zasobów (np. pojemności) w ilości C_i , to serwer alokuje te niewykorzystane zasoby sprawiedliwie do innych, aktywnych w tym czasie przepływów dopóty, dopóki i -ty przepływ jest nieaktywny. Sprawiedliwe kolejkowanie oznacza, że zasoby otrzymywane przez każdy aktywny przepływ są proporcjonalne do ilości zasobów przyznanego każdemu przepływowi. Działanie każdego algorytmu szeregowania wnosi określone opóźnienie w procesie obsługi przepływów. Z tego względu naturalnymi miarami jakości dyscyplin szeregowania są: sprawiedliwość szeregowania, opóźnienie i fluktuacja opóźnienia wnoszonego przez dyscyplinę szeregowania.

9.7.3.1. SPRAWIEDLIWOŚĆ DYSCYPLINY SZEREGOWANIA

Jedną z wielu definicji sprawiedliwości szeregowania (ang. *scheduling fairness*) jest następująca. Jeżeli:

➤ obsługiwane w sieci przepływy są nieskończenie podzielne, a ich obsługa jest obsługą dowolnie małych kwantów ruchu,

➤ i -ty i j -ty ($i, j = 1, 2, \dots, n; i \neq j$) są dwoma dowolnymi przepływami, które w przedziale czasu $(t_1, t_2]$ są przepływami z zaległościami,

➤ $s_i(t_1, t_2)$ i $s_j(t_1, t_2)$ są, odpowiednio, ilościami ruchu i -tego i j -tego przepływów obsłużonego przez serwer w przedziale czasu $(t_1, t_2]$,

to dyscyplina szeregowania realizowana przez serwer, której rezultatem działania jest przydział pojemności C_i i C_j , odpowiednio dla przepływów i -tego i j -tego, jest sprawiedliwa wtedy i tylko wtedy, gdy dla $\forall t_1, t_2$ i $\forall i, j$, takich, że i -ty i j -ty przepływy są aktywne w przedziale czasu $(t_1, t_2]$ spełniony jest warunek:

$$\left| \frac{s_i(t_1, t_2)}{C_i} - \frac{s_j(t_1, t_2)}{C_j} \right| = 0.$$

Interpretacja tego warunku jest następująca: stopień wykorzystania zasobów przydzielonych poszczególnym przepływom z zaległościami jest taki sam. Warto zwrócić uwagę, że wartości ilorazów $s_i(t_1, t_2)/C_i$ i $s_j(t_1, t_2)/C_j$ są jednakowe i mogą być mniej-

sze, równe lub większe od jedności. Wartości ilorazów są mniejsze od jedności, gdy wszystkie przepływy są przepływami z zaległościami, i są większe od jedności, gdy nie wszystkie przepływy są przepływami z zaległościami.

Podana definicja sprawiedliwości dyscypliny szeregowania jest spełniana tylko w przypadku idealnego, nieskończonego podzielnego przepływu. W rzeczywistych sieciach, w których obsługa ruchu polega na obsłudze dyskretnych jednostek danych o zmiennej długości, definicja sprawiedliwości jest modyfikowana do następującej: realizowana przez serwer dyscyplina szeregowania jest sprawiedliwa wtedy i tylko wtedy, gdy dla $\forall t_1, t_2$ i $\forall i, j$, takich, że i -ty i j -ty przepływy są aktywne w przedziale czasu $(t_1, t_2]$, spełniony jest warunek:

$$\left| \frac{s_i(t_1, t_2)}{C_i} - \frac{s_j(t_1, t_2)}{C_j} \right| = \beta,$$

gdzie β jest dolnym ograniczeniem sprawiedliwości rozdziału zasobów, tzw. indeksem sprawiedliwości alokacji zasobów (ang. *fairness index*).

Im mniejsza wartość indeksu sprawiedliwości, tym bardziej sprawiedliwa jest dyscyplina szeregowania.

Z oczywistych względów w sieciach z komutacją jednostek danych (np. pakietów) najmniejsza wartość indeksu sprawiedliwości dowolnej dyscypliny szeregowania nie może być mniejsza od czasu transmisji jednostki danych o największej długości. Jeżeli:

- i -ty oraz j -ty przepływy są rzeczywistymi przepływami z zaległościami, aktywnymi w przedziale czasu $(t_1, t_2]$,
- maksymalne długości jednostek danych tych przepływów są, odpowiednio, równe $l_{i \max}$ oraz $l_{j \max}$,

to dla $\forall t_1, t_2$ wartość ograniczenia indeksu sprawiedliwości alokacji zasobów β jest równa maksymalnej różnicy sumy czasów obsługi jednostek danych o maksymalnych długościach w przepływach współdzielących zasoby:

$$\max \left| \frac{s_i(t_1, t_2)}{C_i} - \frac{s_j(t_1, t_2)}{C_j} \right| = \left| \frac{s_i(t_1, t_2) \pm l_{i \max}}{C_i} - \frac{s_j(t_1, t_2) \pm l_{j \max}}{C_j} \right| \leq \frac{l_{i \max}}{C_i} + \frac{l_{j \max}}{C_j} = \beta.$$

9.7.3.2. OPÓŹNIENIE WNOSZONE PRZEZ DYSCYPLINĘ SZEREGOWANIA

Każdą dyscyplinę szeregowania, poza sprawiedliwością, charakteryzują gwarantowane wartości górnych ograniczeń opóźnień i fluktuacji opóźnienia.

Omawiane górne ograniczenia na ogół są opóźnieniami dla tzw. najgorszego przypadku, zachodzącego wtedy, gdy charakterystyki ruchu, generowanego przez źródło, są wymuszane przez sterownik z implementacją algorytmu ciekącego wiadra. Wówczas całkowite opóźnienie wnoszone przez sieć jest definiowane jako suma czasów

transmisji, czasów propagacji i czasów oczekiwania w kolejkach w każdym węźle sieci na trasach pomiędzy systemami końcowymi. Ostatni z wymienionych składników opóźnienia ma największe znaczenie – od dobrej dyscypliny szeregowania wymagane jest gwarantowanie górnego ograniczenia czasu oczekiwania na obsługę w kolejkach.

Wprowadzenie do sieci i transferowanie przez sieć, od węzła do węzła, strumienia jednostek danych przepływu wiąże się ze zniekształceniem początkowego profilu przepływu. Z tego powodu charakteryzowanie ruchu wewnątrz sieci jest bardzo trudnym zadaniem.

Do szacowania całkowitego opóźnienia dla tzw. najgorszego przypadku niezbędne jest uwzględnienie opóźnień wprowadzanych przez wszystkie kolejne węzły sieci, przez które dany przepływ jest przekazywany. Wprowadzany przez źródło do sieci przepływ jest formowany przez algorytm ciekącego wiadra, którego jednym z parametrów jest szybkość generowania znaczników v_{ZN} . W każdym serwerze, zlokalizowanym w k -tym węźle na trasie przepływu, wymagana jest rezerwacja pojemności C_k spełniająca warunek: $C_k \geq v_{ZN}$. Taka alokacja zasobów daje możliwość oszacowania maksymalnego opóźnienia i wyrażenia jego wartości wartościami parametrów algorytmu ciekącego wiadra oraz ilością rezerwowanych zasobów w tzw. najgorszym przypadku.

Do szacowania gwarantowanego, przez dyscyplinę szeregowania, górnego ograniczenia całkowitego opóźnienia wykorzystywane są następujące definicje:

1. Definicja zgodności ruchu z parametrami algorytmu ciekącego wiadra:

Jeżeli $a_i(t_1, t_2)$ jest ilością ruchu generowanego w ramach i -tego przepływu w przedziale czasu $(t_1, t_2]$, to i -ty przepływ jest definiowany jako zgodny z algorytmem ciekącego wiadra, opisanym parametrami (b_{ZNi}, v_{ZNi}) , dla każdego t_1 i t_2 , gdy spełniony jest warunek:

$$a(t_1, t_2) \leq b_{ZNi} + v_{ZNi}(t_2 - t_1).$$

2. Definicja wykładniczo ograniczonej wybuchowości ruchu:

Wybuchowość źródła ruchu, generowanego w ramach i -tego przepływu, jest ograniczona wykładniczo (ang. *exponentially bounded burstiness process*) przez proces o parametrach $(v_{ZNi}, \alpha_i, \gamma_i)$ wtedy, gdy dla każdego t_1 i t_2 ($t_2 \geq t_1$) oraz dowolnego x spełniony jest warunek:

$$P\{a(t_1, t_2) \leq x + v_{ZNi}(t_2 - t_1)\} \leq \alpha_i \exp(-\gamma_i x).$$

3. Definicja gwarantowanej szybkości obsługi jednostek danych:

Gwarantowana szybkość obsługi jednostek danych (ang. *guaranteed rate clock*) w k -tym węźle dla każdego i -tego przepływu jest wartością, charakteryzującą każdą jednostkę danych przepływu, spełniającą warunki:

$$\text{GRC}_k(i, 0) = 0,$$

$$\text{GRC}_k(i, r) = \max\{t_k(i, r), \text{GRC}_k(i, r-1)\} + \frac{l_{ir}}{C_i},$$

gdzie:

$t_k(i, r)$ – czas napływu r -tej jednostki danych i -tego przepływu do k -tego węzła,

l_{ir} – długość r -tej jednostki danych i -tego przepływu,

C_i – gwarantowana szybkość dla i -tego przepływu.

4. Definicja przynależności dyscypliny obsługi do klasy gwarantowanej szybkości obsługi jednostek danych:

Dyscyplina szeregowania w k -tym węźle należy do klasy gwarantowanej szybkości obsługi jednostek danych dla jednostek danych i -tego przepływu, gdy dana dyscyplina szeregowania gwarantuje, że r -ta jednostka danych i -tego przepływu będzie obsłużona (przekazana) w k -tym węźle do chwili upływu czasu:

$$t_{kir} = \text{GRC}_k(i, r) + \omega_k,$$

gdzie ω_k jest wartością stałą, charakteryzującą implementację danej dyscypliny szeregowania w k -tym węźle (np. maksymalny czas oczekiwania danej jednostki na zakończenie obsługi jednostki danych, poprzedzającego daną jednostkę danych w k -tym węźle trasy).

Zgodnie z podanymi definicjami, jeżeli dyscyplina szeregowania należy do klasy gwarantowanej szybkości obsługi jednostek danych, to dyscyplina ta gwarantuje, że całkowite opóźnienie r -tej jednostki danych i -tego przepływu d_{ir} , przekazywanego przez m kolejnych węzłów sieci, spełnia ograniczenia:

➤ dla ruchu ograniczonego wartościami parametrów algorytmu ciekącego wiadra:

$$d_{ir} \leq \frac{b_{ZNi} + (m-1) \max_{s \in [1, 2, \dots, r]} l_{is}}{v_{ZNi}} + \sum_{k=1}^m \omega_k + \sum_{k=1}^{m-1} \tau_k,$$

➤ dla przepływu o wykładniczo ograniczonej wybuchowości ruchu:

$$P \left\{ d_{ir} \geq x + (m-1) \frac{\max_{s \in [1, 2, \dots, r]} l_{is}}{v_{ZNi}} + \sum_{k=1}^m \omega_k + \sum_{k=1}^{m-1} \tau_k \right\} \leq \alpha_i \exp(-\gamma_i x v_{ZNi}),$$

gdzie:

$(m-1)l_{is}/v_{ZNi}$ – suma opóźnień pakietyzacji, wynikających z konieczności skompletowania jednostki danych przed rozpoczęciem kolejnej fazy jej obsługi w każdym z $(m-1)$ węzłach sieci, tzn. wszystkich węzłach trasy z wyłączeniem węzła ujścia,

τ_k ($k = 1, 2, \dots, m$) – czas propagacji w łączu pomiędzy węzłami k -tym a $(k+1)$ -szym.

9.7.4. DYSCYPLINA SZEREGOWANIA

Z UOGÓLNIONYM PODZIAŁEM CZASU PROCESORA

Wybór odpowiednich dyscyplin szeregowania jednostek danych, należących do różnych przepływów jednocześnie obsługiwanych w węzłach sieci, jest podstawą efektywnego sterowania przepływem w tych węzłach. Dobra, z tego punktu widzenia,

dyscyplina obsługi powinna umożliwiać różny sposób obsługi jednostek danych należących do przepływów o różnej żądanej jakości usług. Elastyczność dyscyplin szeregowania (ang. *scheduling flexibility*) w omawianych zadaniach nie może oznaczać rezygnacji z zasad sprawiedliwej (ang. *fairness*) obsługi, tzn. sposób obsługi wybranych kilku przepływów nie może prowadzić do pogorszenia jakości usług pozostałych przepływów i naruszenia warunków gwarancji jakości usług. Co więcej, warunkiem możliwości gwarancji jakości usług jest przewidywalność (ang. *analyzable*) działania dyscypliny szeregowania jednostek danych.

Przykładem elastycznej, sprawiedliwej i przewidywalnej dyscypliny szeregowania dla modelu przepływowego strumieni jednostek danych jest GPS. Dyscyplina szeregowania GPS, którą można interpretować jako ciągły odpowiednik dyskretnej dyscypliny szeregowania, pozwala na uzyskiwanie dokładnego rozwiązania zadania ważonego maksymalno-minimalnego sprawiedliwego szeregowania (ang. *weighted max-min flow*).

Serwer dyscypliny obsługi GPS definiuje się jako stanowisko obsługi ruchu:

- zachowujące pracę, tzn. serwer jest zawsze zajęty, gdy w kolejce do obsługi oczekują jednostki danych,
- obsługujące jednocześnie wiele przepływów o nieskończonej podzielności,
- pracujące ze stałą szybkością C .

Jeśli przyjmiemy, że:

1. $a_i(\tau, t)$ – ilość ruchu należącego do i -tego przepływu i napływającego do serwera GPS w przedziale czasu $(\tau, \tau + t]$ ($a_i(\tau, t) = a_i(t)$ (dla $\tau = 0$) – charakteryzuje zapotrzebowanie i -tego przepływu na obsługę w przedziale czasu $(\tau, \tau + t]$;

2. $s_i(\tau, t)$ – ilość ruchu należącego do i -tego przepływu i obsłużonego przez serwer GPS w przedziale czasu $(\tau, \tau + t)$ ($s_i(\tau, t) = s_i(t)$ dla $\tau = 0$) – charakteryzuje obciążenie serwera przez i -ty przepływ w przedziale czasu $(\tau, \tau + t]$;

3. i -ty przepływ jest przepływem:

- bez zaległości, gdy jest spełniony warunek: $s_i(\tau, t) = a_i(\tau, t)$,
- z zaległościami, gdy jest spełniony warunek: $s_i(\tau, t) < a_i(\tau, t)$;

4. n – liczba wszystkich przepływów obsługiwanych przez serwer GPS;

5. $\phi_1, \phi_2, \dots, \phi_i, \dots, \phi_n$ – zbiór dodatnich liczb rzeczywistych przypisanych wszystkim przepływom, obsługiwanych w danym węźle przez serwer GPS – liczba rzeczywista ϕ_i ($i \in \{1, 2, \dots, n\}$) może być interpretowana jako współczynnik wagowy;

6. C – całkowita pojemność kanału transmisyjnego, tzn. szybkość obsługi wszystkich przepływów w węźle przez serwer GPS;

7. C_i – pojemność gwarantowana (gwarantowana ilość zasobów) do obsługi i -tego przepływu ($i \in \{1, 2, \dots, n\}$);

8. \bar{C}_i – pojemność wykorzystana (wykorzystana ilość zasobów) do obsługi i -tego przepływu ($i \in \{1, 2, \dots, n\}$);

9. I – zbiór wszystkich n przepływów jednocześnie obsługiwanych w węźle przez serwer GPS,

10. $I_1(\tau, t)$ – podzbiór zbioru wszystkich przepływów, zawierający przepływy bez zaległości w przedziale czasu $(\tau, \tau + t]$: $(I_1(\tau, t) \subset I = \{1, 2, \dots, n\})$,

11. $I_2(\tau, t)$ – podzbiór zbioru wszystkich przepływów, zawierający przepływy z zaległościami w przedziale czasu $(\tau, \tau + t]$; $I_2(\tau, t) \subset I, I_1(\tau, t) \cup I_2(\tau, t) = I$,

serwer GPS jest definiowany jako stanowisko obsługi, dla którego obsłużone w przedziale $(\tau, \tau + t]$ ilości ruchu każdego przepływu z zaległościami spełniają warunek:

$$\frac{s_i(\tau, t)}{s_j(\tau, t)} \geq \frac{\phi_i}{\phi_j} \text{ dla } i \in I_2(\tau, t), j \in I, i \neq j.$$

Wynik sumowania ilości ruchu wszystkich przepływów, obsługiwanych przez algorytm GPS w przedziale $(\tau, \tau + t]$, jest równy:

$$s_i(\tau, t) \sum_{j \in I} \phi_j \geq \sum_{j \in I} s_j(\tau, t) \phi_j = t C \phi_i,$$

co oznacza, że i -temu przepływowi z zaległościami gwarantowana jest stała szybkość obsługi (stała przepustowość kanału transmisyjnego) o wartości:

$$C_i = \frac{\phi_i C}{\sum_{j \in I} \phi_j}.$$

Dyscyplina szeregowania GPS jest dyscypliną zachowującą pracę, tzn. jeżeli i -ty przepływ ($i \in \{1, 2, \dots, n\}$) jest pusty i nie wykorzystuje przydzielonej pojemności C_i , to serwer przydziela sprawiedliwie (alokuje) pojemność C_i innym aktywnym przepływowom dopóty, dopóki i -ty przepływ jest pusty.

Działanie dyscypliny szeregowania GPS można także interpretować jako schemat zwielokrotniania przepływów, o licznych praktycznie użytecznych właściwościach:

➤ jeżeli v_i jest średnią szybkością napływu jednostek danych i -tego przepływu, to dopóki spełniony jest warunek $v_i \leq C_i$, dopóty i -temu przepływowi gwarantowana jest przepustowość v_i , a i -ty przepływ jest przepływem bez zaległości, niezależnie od szybkości napływu jednostek danych należących do innych przepływów,

➤ niezależnie od gwarancji przepustowości, zaległości i -tego przepływu będą zawsze usuwane z szybkością nie mniejszą niż C_i ,

➤ wartość górnego ograniczenia opóźnienia jednostek danych i -tego przepływu jest funkcją długości kolejki jednostek danych i -tego przepływu i jest niezależna od długości kolejek jednostek danych pozostałych przepływów,

➤ możliwości zmiany wartości liczb rzeczywistych ϕ_j ($j \in I$) pozwalają na elastyczną obsługę przepływów, np. równość wartości wszystkich współczynników wagowych ϕ_j redukuje omawiany system obsługi do systemu z równomiernym podziałem zasobów (w takim przypadku działanie dyscypliny GPS jest równoważne działaniu dyscyplin FCFS i BRFQ),

➤ gdy suma średnich szybkości napływu jednostek danych wszystkich n przepływów jest mniejsza od ilości zasobów (pojemności), dowolny zbiór dodatnich liczb

rzeczywistych ϕ_j ($j \in I$), przypisanych przepływowi, zachowuje stabilność systemu, np. i -temu przepływowi, o dużej szybkości szczytowej napływu pakietów $v_{i \max}$ i małej wrażliwości na opóźnienia można przydzielić pojemność C_i , bliską wartości średniej szybkości napływu, i tym samym podnieść jakość obsługi innych przepływów,

➤ gdy szybkość napływu jednostek danych ze źródła jest sterowana zgodnie z algorytmem ciekącego wiadra, gwarantowana jest wartość maksymalnego opóźnienia jednostek danych przepływów w tzw. najgorszym przypadku.

Jeżeli dla każdego i -tego przepływu ($i \in \{1, 2, \dots, n\}$) napływ jednostek danych do stanowiska obsługi o pojemności C , przydzielonej poszczególnym przepływowi zgodnie z algorytmem GPS, jest ograniczony wartościami parametrów algorytmu ciekącego wiadra, tzn:

- szybkością napływu znaczników v_{iZN} spełniającą warunek: $v_{iZN} \leq C_i$,
- pojemnością pamięci buforowej znaczników (rozmiarami maksymalnej paczki jednostek danych) b_{iZN} ,

to maksymalne opóźnienia jednostek danych, należących do poszczególnych przepływów, wnoszone przez dyscyplinę szeregowania GPS są ograniczone z góry wartościami

$$d_{i \max} = \frac{b_{iZN}}{C_i} \leq \frac{b_{iZN}}{v_{iZN}}.$$

Całkowite maksymalne opóźnienie jednostki danych i -tego przepływu przekazywanego przez m kolejnych węzłów sieci, którego ruch jest ograniczony wartościami parametrów algorytmu ciekącego wiadra (b_{iZN} , v_{iZN}), wnoszone przez dyscyplinę szeregowania GPS i ciąg $(m - 1)$ łączy na trasie złożonej z m łączy, jest dane wyrażeniem:

$$d_{\text{GPS}i \max} \leq \frac{b_{iZN}}{v_{iZN}} + \sum_{k=1}^{m-1} \tau_k,$$

w którym τ_k ($k = 1, 2, \dots, m$) jest czasem propagacji w łączy pomiędzy k -tym węzłem a $(k + 1)$ -szym węzłem.

Z właściwości algorytmów sprawiedliwego szeregowania wynika, że ilość zasobów wykorzystanych do obsługi i -tego przepływu, obsługiwanego przez węzeł zgodnie z dyscypliną szeregowania GPS, jest nie mniejsza od gwarantowanej ilości zasobów:

1. Jeżeli wszystkie przepływy obsługiwane w węźle są przepływami z zaległościami ($I_1(\tau, t) = \emptyset$), tzn. jeżeli $s_i(\tau, t) = C_i < a_i(\tau, t)$ dla każdego $i = 1, 2, \dots, n$, to pojemność wykorzystana przez przepływy jest równa pojemności gwarantowanej dla przepływów; $\bar{C}_i = C_i$ dla każdego $i = 1, 2, \dots, n$.

2. Jeżeli co najmniej jeden spośród obsługiwanych przepływów jest przepływem bez zaległości ($I_1(\tau, t) = \emptyset$), tzn. $s_i(\tau, t) = a_i(\tau, t) \leq C_i$ dla $i \in I_1(\tau, t)$, to:

➤ pojemności wykorzystane przez przepływy bez zaległości są nie większe niż pojemności gwarantowane, tzn. $\bar{C}_i \leq C_i$ dla $i \in I_1(\tau, t)$,

➤ pojemności wykorzystane przez przepływy z zaległościami są nie mniejsze niż pojemności gwarantowane, tzn. $\bar{C}_i \geq C_i$ dla $i \in I_2(\tau, t)$.

Gdy wszystkie obsługiwane przepływy są przepływami z zaległościami, wówczas dyscyplina szeregowania GPS realizuje tylko zadanie separacji przepływów, natomiast wtedy, gdy co najmniej jeden przepływ jest przepływem bez zaległości, dyscyplina szeregowania GPS może jednocześnie realizować zadania separacji przepływów i współdzielenia zasobów.

Wszystkie jednostki danych należące do przepływów bez zaległości, tzn. do przepływów, dla których liczby jednostek danych $a_j(\tau, t)$ napływających w przedziale czasu $(\tau, \tau + t]$ są nie większe od gwarantowanej w przedziale $(\tau, \tau + t]$ pojemności C_j ($a_j(\tau, t) \leq tC_j, j \in I_1(\tau, t)$), będą obsługiwane bez oczekiwania w kolejce wejściowej. Suma pojemności niewykorzystanych przez przepływy bez zaległości może być przydzielona, w ilościach proporcjonalnych do wartości współczynników wagowych przypisanych poszczególnym przepływom, przepływom z zaległościami, tzn. przepływom należącym do $I_2(\tau, t)$ -tego podzbioru przepływów. Pojemności wykorzystane przez przepływy z zaległościami w przedziale czasu $(\tau, \tau + t]$ są określone zależnościami:

$$\begin{aligned} \bar{C}_i &= \frac{\phi_i}{\sum_{l \in I} \phi_l} C + \frac{\phi_i}{\sum_{k \in I_2(\tau, t)} \phi_k} \left(\sum_{j \in I_1(\tau, t)} (C_j - \bar{C}_j) \right) = \\ &= C_i + \frac{\phi_i}{\sum_{k \in I_2(\tau, t)} \phi_k} \left(\sum_{j \in I_1(\tau, t)} (C_j - \bar{C}_j) \right) \geq C_i, \end{aligned}$$

dla każdego $i \in I_2$.

Zależność $\bar{C}_i \geq C_i$, dla $i \in I_2(\tau, t)$, oznacza, że ilości zasobów wykorzystanych przez i -ty przepływ z zaległościami – \bar{C}_i oraz dowolny inny j -ty przepływ – \bar{C}_j spełniają warunek:

$$\frac{\bar{C}_i}{\bar{C}_j} = \frac{s_i(\tau, t)}{s_j(\tau, t)} \geq \frac{\phi_i}{\phi_j} \quad \text{dla } i \in I_2(\tau, t), j \in I, i \neq j,$$

który charakteryzuje jednocześnie wykonywane przez algorytm GPS zadania separacji przepływów i współdzielenia zasobów (pojemności).

Zgodnie z tym warunkiem względna ilość ruchu należącego do przepływu z zaległościami, obsługiwana przez system obsługi z dyscypliną szeregowania GPS, może być większa niż względna wartość liczby rzeczywistej (wartość współczynnika wagowego) przyznanej temu przepływowi.

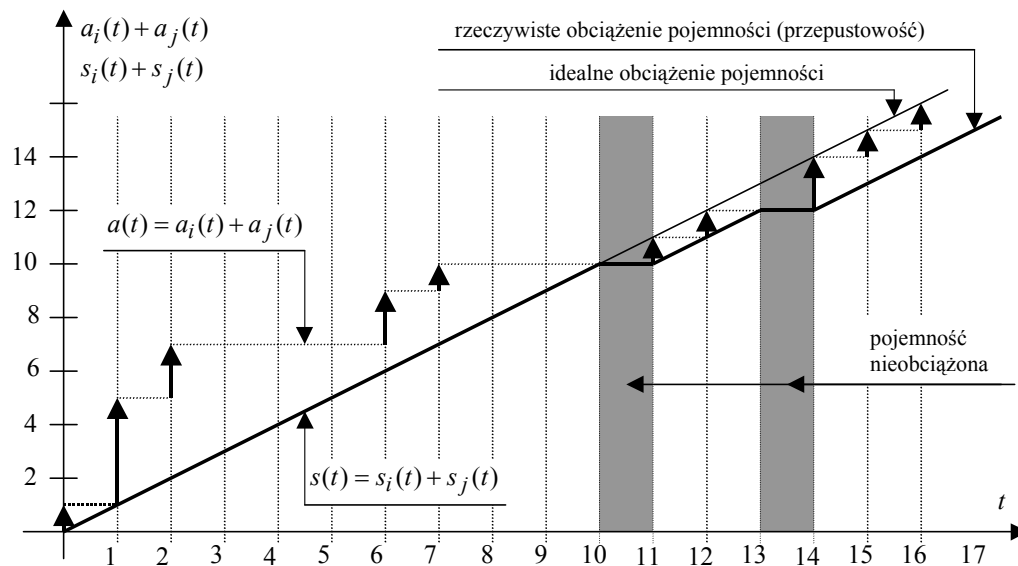
O właściwościach obsługi poszczególnych i -tych przepływów ($i \in I$), w szczególności czasów obsługi jednostek danych, można wnioskować na podstawie przebiegu krzywych:

➤ $a_i(\tau, t)$ – obciążenia stanowiska obsługi o gwarantowanej pojemności C_i przez jednostki danych i -tego przepływu w przedziale czasu $(\tau, \tau + t]$,

➤ $s_i(\tau, t)$ – wykorzystania stanowiska obsługi o gwarantowanej pojemności C_i przez jednostki danych i -tego przepływu w przedziale czasu $(\tau, \tau + t]$,

➤ $a_i(\tau, t) - s_i(\tau, t)$ – różnicy obciążenia i wykorzystania stanowiska obsługi o gwarantowanej pojemności C_i przez jednostki danych i -tego przepływu w przedziale czasu $(\tau, \tau + t]$.

Przykładowy przebieg zmian obciążenia i wykorzystania stanowiska obsługi, wykorzystywanego do obsługi dwóch przepływów: i -tego i j -tego, tzn. odpowiednio $a(\tau, t) = a_i(\tau, t) + a_j(\tau, t)$ i $s(\tau, t) = s_i(\tau, t) + s_j(\tau, t)$, dla $\tau = 0$, przedstawiono na rysunku 9.13.



Rysunek 9.13. Przebieg obciążenia i wykorzystania stanowiska obsługi

Podobnie, jak w przypadku obsługi dwóch przepływów (rysunek 9.13), odbywa się obsługa wielu przepływów jednocześnie. Jakość obsługi jednostek danych zbioru n przepływów, przez stanowisko obsługi o stałej pojemności C , zależy od różnicy sumarycznego obciążenia i wykorzystania pojemności:

$$\sum_{i \in I} a_i(\tau, t) - \sum_{i \in I} s_i(\tau, t).$$

Wyznaczanie rzeczywistego czasu zakończenia obsługi kolejnych jednostek danych, szeregowanych zgodnie z dyscypliną GPS, jest na ogół zadaniem trudnym. Wy-

nika to z tego, że praktyczna realizacja dyscypliny szeregowania GPS polega na tym, że algorytm emulujący dyscyplinę szeregowania GPS wybiera do obsługi nie nieskończone małe porcje ruchu, ale jednostki danych o określonej długości.

Podstawowa trudność w obliczaniu czasów zakończenia obsługi kolejnych jednostek danych poszczególnych przepływów, a zatem także czasu zakończenia obsługi tych przepływów, jest powodowana tym, że z każdą napływającą jednostką danych może być związana zmiana stanu przepływu, który z przepływu bez zaległości może zmienić się na przepływ z zaległościami i odwrotnie (rysunek 9.13). Zmiany stanów przepływów pociągają za sobą zmiany ilości zasobów wykorzystywanych do obsługi różnych, współbieżnych przepływów. Zmienne w czasie ilości zasobów, wykorzystywanych do obsługi tych samych przepływów, powodują:

- zmiany oczekiwanych czasów zakończenia obsługi jednostek danych należących do różnych przepływów,
- zmiany oczekiwanej kolejności zakończenia obsługi jednostek danych w porównaniu z kolejnością wynikającą z czasów rozpoczęcia obsługi tych jednostek danych,
- zdarzenia, że zasoby oczekują na jednostki danych.

Oczekiwanie przez zasoby na jednostki danych powoduje wydłużenie średnich czasów obsługi jednostek danych w poszczególnych przepływach. Występowanie losowych czasów oczekiwania zasobów na jednostki danych pomiędzy dwiema kolejnymi jednostkami danych jest przyczyną trudności szacowania czasów zakończenia obsługi kolejnych jednostek danych.

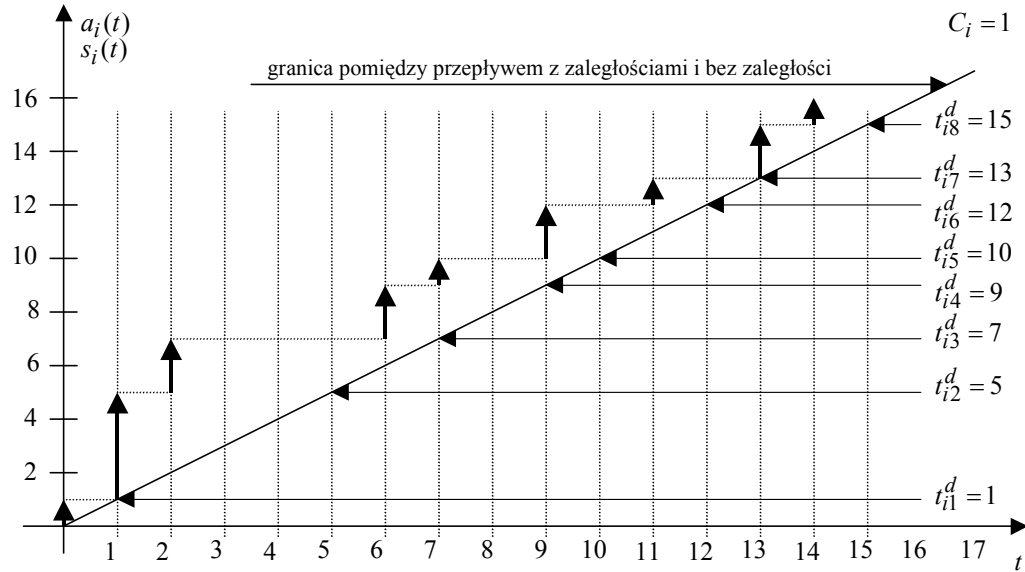
Względnie łatwo jest wyznaczyć czasy zakończenia obsługi jednostek danych wówczas, gdy wszystkie przepływy obsługiwane w danym węźle są przepływami z zaległościami, tzn. wtedy, gdy $a_i(\tau, t) \geq s_i(\tau, t) = C_i$ dla każdego $i \in I$, gdzie C_i jest ilością zasobów wynikających ze sprawiedliwego rozdziału całkowitej ilości zasobów C . Zasoby przydzielone poszczególnym przepływowi są więc w pełni wykorzystane i nie następuje przekazywanie zasobów pomiędzy poszczególnymi przepływami. W omawianym przypadku dyscyplina szeregowania GPS realizuje tylko zadanie separacji przepływów.

Obsługę i -tego przepływu, realizowaną jednocześnie z obsługą innych $(n - 1)$ przepływów, można traktować jako obsługę niezależnego przepływu przez dedykowane temu przepływowi zasoby (rysunek 9.14).

Gdy wszystkie przepływy są przepływami z zaległościami, obsługę jednostek danych przepływów z zaległościami charakteryzuje to, że czas oczekiwania stanowiska obsługi na kolejną, k -tą jednostkę danych ($k = 1, 2, \dots$) przepływu $w_{k,k-1}$ jest zawsze równy zeru, tzn. zasoby stanowiska obsługi, przeznaczone dla tego przepływu, są w całości wykorzystane. Zgodnie z tym czas zakończenia obsługi k -tej jednostki danych w i -tym przepływie, oznaczony przez t_{ik}^d , jest sumą czasu zakończenia obsługi $(k - 1)$ -szej jednostki danych w i -tym przepływie $t_{i,k-1}^d$ oraz czasu obsługi k -tej jednostki danych o długości l_{ik} , tzn.:

$$t_{ik}^d = t_{i,k-1}^d + \frac{l_{ik}}{C_i} = t_{i,k-1}^d + \frac{l_{ik}}{C_i},$$

gdzie l_{ik} jest długością jednostki danych (wielokrotność długości wzorcowej jednostki danych), a pojemności C i C_i są wyrażone w jednostkach danych na jednostkę czasu.



Rysunek 9.14. Przebieg obciążenia i wykorzystania stanowiska obsługi przez przepływy z zaległościami

Gdy dany i -ty przepływ jednostek danych jest na przemian przepływem z zaległościami i bez zaległości, czas oczekiwania na jednostkę danych przez stanowisko obsługi może być różny od zera; możliwe jest zdarzenie, że stanowisko obsługi, po obsłużeniu jednostki danych, oczekuje na kolejną jednostkę danych (rysunek 9.15). Gdy następuje zmiana stanu przepływu z przepływu z zaległościami na przepływy bez zaległości, czas zakończenia obsługi k -tej jednostki danych i -tego przepływu t_{ik}^d jest sumą czasu zakończenia obsługi $(k-1)$ -szej jednostki danych i -tego przepływu $t_{i,k-1}^d$, czasu oczekiwania stanowiska obsługi na k -tą jednostkę danych $w_{k,k-1}(C_i)$ oraz czasu obsługi k -tej jednostki danych o długości l_{ik} przez stanowisko obsługi o pojemności C_i , tzn.:

$$t_{ik}^d = t_{i,k-1}^d + w_{k,k-1}(C_i) + \frac{l_{ik}}{C_i},$$

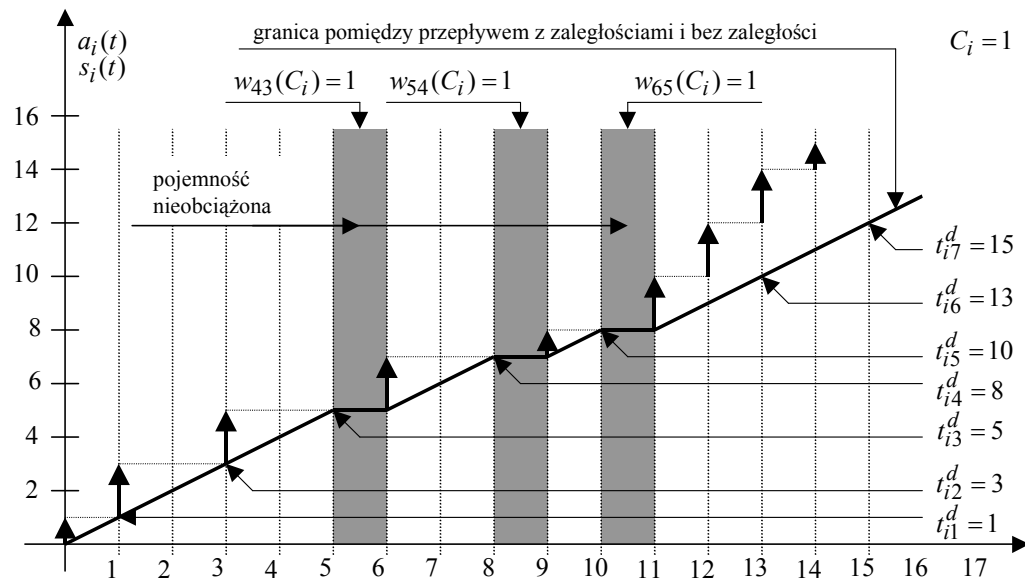
gdzie $w_{k,k-1}(C_i) = 0$ lub $w_{k,k-1}(C_i) \neq 0$.

Czas oczekiwania stanowiska obsługi na jednostki danych i -tego przepływu wpływa na czas zakończenia obsługi poszczególnych jednostek danych tego przepływu. Wpływ czasu oczekiwania przez stanowisko obsługi na jednostki danych na czasy

zakończenia obsługi poszczególnych jednostek danych jest różny w zależności od sposobu udostępniania zasobów poszczególnym przepływom, tzn. od funkcji realizowanych przez dyscyplinę szeregowania jednostek danych:

➤ jeżeli przepływy, niezależnie od tego, czy w danej chwili są przepływami bez zaległości czy przepływami z zaległościami, wykorzystują tylko zasoby dedykowane tym przepływom: $\bar{C}_i = C_i$, to znaczy że dyscyplina szeregowania jednostek danych realizuje tylko zadanie separacji przepływów,

➤ jeżeli przepływy z zaległościami mogą wykorzystywać zasoby aktualnie nie wykorzystywane przez przepływy bez zaległości: $\bar{C}_i > C_i$, to znaczy, że dyscyplina szeregowania realizuje zarówno zadanie separacji przepływów, jak i zadanie współdzielenia zasobów.

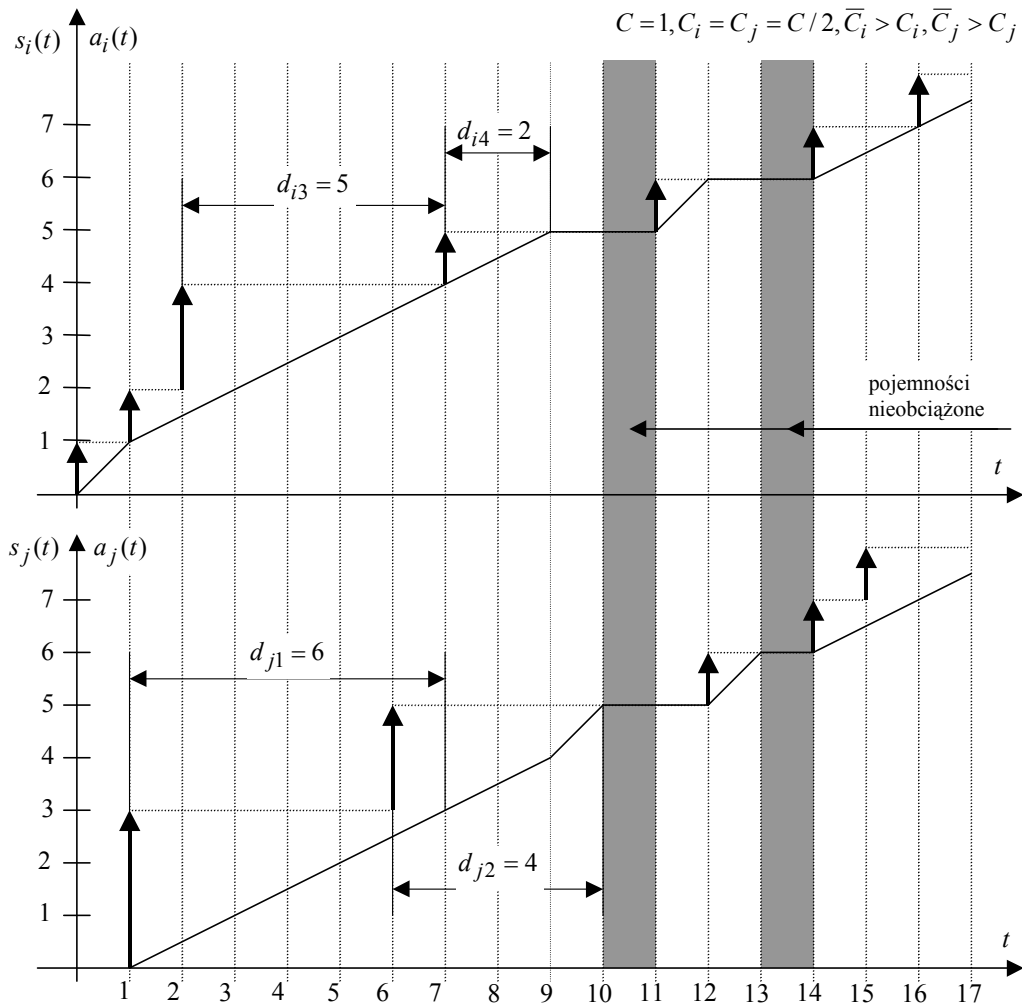


Rysunek 9.15. Oczekiwanie stanowiska obsługi na jednostki danych

Jeśli dyscyplina szeregowania nie realizuje sprawiedliwego rozdziału zasobów, tzn. gdy zasoby (pojemność) stanowiska obsługi są przeznaczone tylko dla danego przepływu (np. na rysunku 9.15 pojemność $C_i = 1$ jest dedykowana tylko i -temu przepływowi), czas zakończenia obsługi k -tej jednostki danych (liczony od chwili rozpoczęcia obsługi pierwszej jednostki danych) jest sumą czasów obsługi k jednostek danych przez stanowisko obsługi o pojemności C_i i sumą czasów oczekiwania dedykowanego stanowiska obsługi na kolejne jednostki danych przepływu $w_{l,l-1}(C_i)$ dla $l = 2, 3, \dots, k$:

$$t_{ik}^d(C_i) = \sum_{l=1}^k \frac{l_{il}}{C_i} + \sum_{l=2}^k w_{l,l-1}(C_i).$$

Jeśli dyscyplina szeregowania zapewnia sprawiedliwy rozdział zasobów, tzn. gdy zasoby mogą być wykorzystywane do obsługi wszystkich przepływów, zasoby gwarantowane dla danego przepływu mogą być, w razie ich niewykorzystywania, przekazywane do obsługi innych przepływów – zasoby gwarantowane dla danego przepływu oczekują na jednostki danych tylko wtedy, gdy nie wszystkie przepływy są przepływami bez zaległości (rysunek 9.16).



Rysunek 9.16. Przekazywanie zasobów do obsługi przepływów
w przypadku stosowania dyscypliny sprawiedliwego rozdziału zasobów

Na przykład (rysunek 9.16), jeżeli pojemności gwarantowane przepływowi i -temu oraz j -temu są jednakowe, tzn. $C_i = C_j = C/2$ i zasoby te mogą być współdzielone

przez wymienione przepływy, to np. wówczas, gdy brak jest jednostki danych i -tego przepływu na stanowisku obsługi, pojemność C_i może być przydzielana chwilowo do obsługi pakietów j -tego przepływu i odwrotnie. Średnia wartość pojemności przydzielonej do obsługi j -tego przepływu \bar{C}_j jest różna od gwarantowanej pojemności C_j .

Czas zakończenia obsługi k -tej jednostki danych, liczony od chwili rozpoczęcia obsługi pierwszej jednostki danych, jest sumą czasów obsługi k pakietów przez stanowisko obsługi o pojemności \bar{C}_i (różnej od wartości gwarantowanej pojemności C_i) i sumą czasów oczekiwania dedykowanego stanowiska obsługi na kolejne pakiety danego przepływu $w_{l,l-1}(\bar{C}_i)$ dla $l = 2, 3, \dots, k$:

$$t_{ik}^d(\bar{C}_i) = \sum_{l=1}^k \frac{l_{il}}{\bar{C}_i} + \sum_{l=2}^k w_{l,l-1}(\bar{C}_i).$$

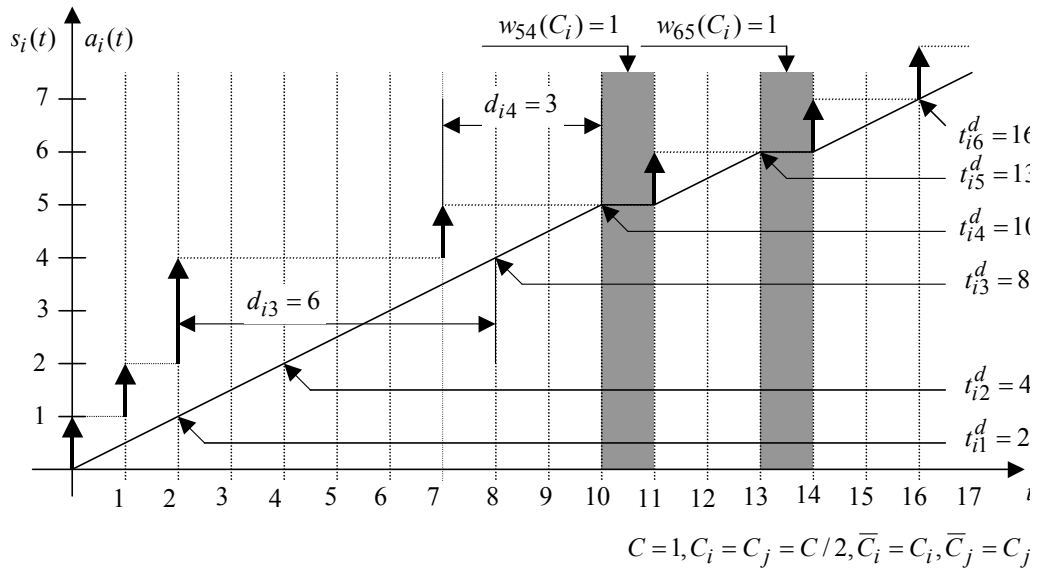
W podanych wyżej wyrażeniach użyto oznaczeń $w_{k,k-1}(C_i)$ i $w_{k,k-1}(\bar{C}_i)$ w celu zaznaczenia, że czasy oczekiwania stanowisk obsługi o różnej pojemności na kolejne jednostki danych zależą od pojemności stanowisk obsługi jednostek danych. Różna pojemność stanowisk obsługi nie ma wpływu na czasy napływu jednostek danych na stanowiska obsługi, tj. do kolejki jednostek danych oczekujących na obsługę lub bezpośrednio na stanowisko obsługi, ma natomiast wpływ na czas oczekiwania jednostki danych na obsługę w kolejce i czas obsługi jednostki danych. Skrócenie lub wydłużenie czasu obsługi danej jednostki danych może powodować, odpowiednio, wydłużenie lub skrócenie czasu oczekiwania stanowiska obsługi na kolejną jednostkę danych (rysunek 9.17).

Jeśli $\bar{C}_i > C_i$, czasy obsługi, a tym samym opóźnienia, niektórych jednostek danych mogą być krótsze od czasów obsługi i opóźnienia tych samych jednostek danych wtedy, gdy $\bar{C}_i = C_i$. Do sporządzenia wykresów na rysunku 9.17 przyjęto, że do obsługi tej samej sekwencji jednostek danych zastosowano dwie różne dyscypliny szeregowania jednostek danych przepływów współdzielących pojemności kanału transmisyjnego przydzielonych przepływowom:

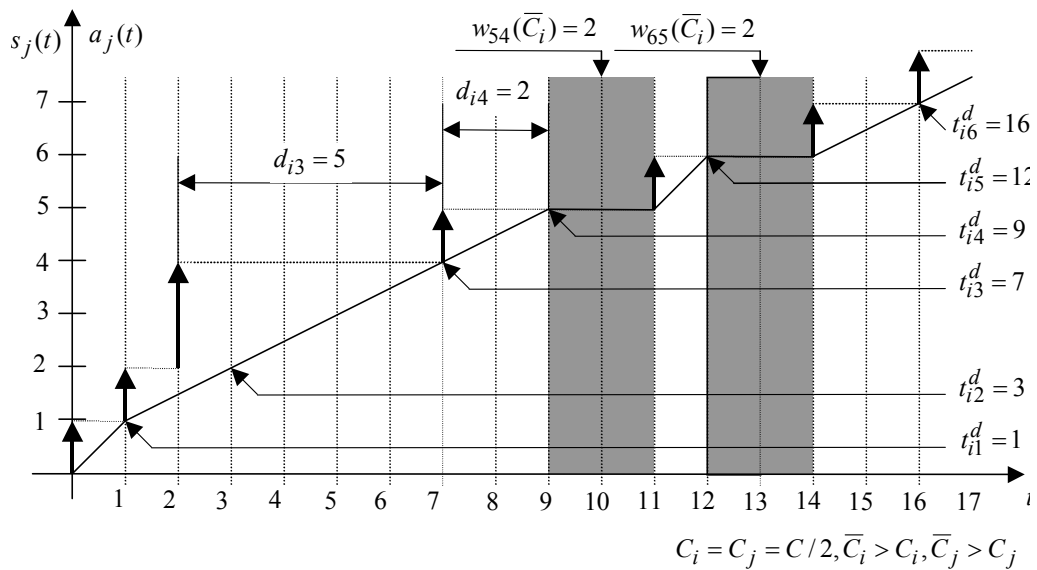
➤ jeżeli pojemność C_j ($C_j = C_i = C/2$) nie może być wykorzystana do obsługi przepływu innego niż j -ty przepływ, tzn. gdy dyscyplina szeregowania nie realizuje sprawiedliwego rozdziału zasobów, to pojemność wykorzystywana do obsługi pakietów i -tego przepływu jest stała i równa pojemności C_i ($\bar{C}_i = C_i$) gwarantowanej temu przepływowi (rysunek 9.17a),

➤ jeżeli pojemność C_j ($C_j = C_i = C/2$) może być wykorzystana do obsługi przepływu innego niż j -ty przepływ, tzn. gdy dyscyplina szeregowania realizuje sprawiedliwy rozdział zasobów, a pojemność C_j nie jest wykorzystywana do obsługi pakietów j -tego przepływu, może być wykorzystana do obsługi pakietów i -tego przepływu; w przedziałach $[0, 1)$ i $[11, 12)$ pojemność \bar{C}_i , wykorzystana przez i -ty przepływ, jest dwa razy większa od pojemności C_i ($\bar{C}_i = C_i + C_j = C$), powodując skrócenie czasu

obsługi odpowiednich pakietów (pierwszego i piątego) i -tego przepływu (rysunek 9.17b).



a) czasy oczekiwania na jednostkę danych – separacja przepływów



b) czasy oczekiwania na jednostkę danych – separacja przepływów i współdzielenie zasobów

Rysunek 9.17. Wpływ dyscypliny szeregowania na czasy obsługi jednostek danych przepływów i czasy oczekiwania stanowiska obsługi na jednostki danych przepływów

Gdy $\bar{C}_i = C_i$, następuje zmniejszenie (w porównaniu z przypadkiem, gdy $\bar{C}_i > C_i$) sumy czasów oczekiwania na jednostki danych przez stanowisko obsługi (tym samym średniego czasu oczekiwania na jednostkę danych przez stanowisko obsługi), jest to konsekwencją większej sumy czasów obsługi jednostek danych (tym samym średniego czasu obsługi jednostek danych i średniego opóźnienia jednostki). Zależność ta potwierdza cechy sprawiedliwego rozdziału zasobów jako korzystniejszego od rozdziału proporcjonalnego do wag przypisanych poszczególnym przepływom.

Dla jednostek danych należących do i -tego przepływu zachodzą zależności:

$$\sum_{l=1}^k \frac{l_{ik}}{\bar{C}_i} \leq \sum_{l=1}^k \frac{l_{ik}}{C_i},$$

tzn. suma czasów obsługi k jednostek danych na stanowisku obsługi o zasobach \bar{C}_i jest nie większa niż suma czasów obsługi k pakietów na stanowisku obsługi o zasobach C_i ($\bar{C}_i \geq C_i$),

$$\sum_{l=2}^k w_{l,l-1}(\bar{C}_i) \leq \sum_{l=2}^k w_{l,l-1}(C_i),$$

tzn. suma czasów oczekiwania na jednostki danych przez stanowisko obsługi o zasobach \bar{C}_i jest nie mniejsza niż suma czasów oczekiwania na jednostki danych przez stanowisko obsługi o zasobach C_i ($\bar{C}_i \geq C_i$),

$$t_{ik}(\bar{C}_i) \leq t_{ik}(C_i),$$

tzn. czas zakończenia obsługi k -tej jednostki danych na stanowisku obsługi o zasobach \bar{C}_i jest nie krótszy niż czas zakończenia obsługi tej jednostki danych na stanowisku obsługi o zasobach C_i ($\bar{C}_i \geq C_i$).

Realizacja dyscypliny szeregowania GPS, będącej idealną dyscypliną szeregowania dla przepływowego modelu obsługi strumienia danych o nieskończonej podzielności, powoduje, że czas zakończenia obsługi k -tej jednostki danych, należącej do i -tego przepływu nie jest rzeczywistym czasem zakończenia obsługi k -tej jednostki danych i rozpoczęcia kolejnego etapu obsługi na trasie ze źródła do ujścia. Wartość czasu t_{ik}^d można interpretować jako liczbę reprezentującą kolejność zakończenia obsługi (transmisji) k jednostek danych i jednocześnie k -tej jednostki danych na danym stanowisku obsługi.

Dyscyplina szeregowania GPS zachowuje kolejność obsługi jednostek danych, należących do dowolnego i -tego ($i = 1, 2, \dots, n$) przepływu, nie zachowuje natomiast kolejności obsługi jednostek danych należących do różnych, jednocześnie obsługiwanych przepływów.

Zachowywanie przez algorytm GPS kolejności obsługi jednostek danych, należących do tego samego i -tego ($i = 1, 2, \dots, n$) przepływu, oznacza, że ciąg czasów rozpoczęcia i zakończenia obsługi kolejnych jednostek danych jest ciągiem niemalejącym:

$$\dots \leq t_{i,k-1}^d < t_{i,k-1}^d \leq t_{ik}^a < t_{ik}^d \leq t_{i,k+1}^d < t_{i,k+1}^d \leq \dots$$

dla dowolnego k -tego pakietu i -tego przepływu, gdzie t_{ik}^a i t_{ik}^d są – odpowiednio – czasami rozpoczęcia i zakończenia obsługi k -tego pakietu i -tego przepływu.

Niezachowywanie przez algorytm GPS kolejności obsługi jednostek danych należących do różnych przepływów oznacza, że możliwe jest wystąpienie następujących, charakterystycznych zależności (kolizji) pomiędzy jednostkami danych, należącymi do różnych przepływów:

$$t_{ik}^d < t_{jl}^d, \text{ gdy } t_{ik}^a > t_{jl}^a$$

czas zakończenia obsługi k -tej jednostki danych i -tego przepływu jest krótszy od czasu zakończenia obsługi l -tej jednostki danych i -tego przepływu; czas rozpoczęcia obsługi k -tej jednostki danych i -tego przepływu jest natomiast dłuższy od czasu rozpoczęcia obsługi l -tej jednostki danych j -tego przepływu,

$$t_{ik}^d > t_{jl}^d, \text{ gdy } t_{ik}^a > t_{jl}^a \text{ i } t_{ik}^a < t_{jl}^d$$

czas zakończenia obsługi k -tej jednostki danych i -tego przepływu jest dłuższy od czasu zakończenia obsługi l -tej jednostki danych j -tego przepływu; czas rozpoczęcia obsługi k -tej jednostki danych i -tego przepływu jest natomiast krótszy od czasu zakończenia obsługi l -tej jednostki danych j -tego przepływu,

$$t_{ik}^d = t_{jl}^d, \text{ gdy } t_{ik}^a > t_{jl}^a \text{ lub } t_{ik}^a \leq t_{jl}^a$$

czas zakończenia obsługi k -tej jednostki danych i -tego przepływu jest równy czasowi zakończenia obsługi l -tej jednostki danych niezależnie od zależności pomiędzy czasami rozpoczęcia obsługi tych jednostek danych.

Możliwość generowania m.in. wyżej wymienionych zdarzeń przez dyscyplinę szeregowania GPS, niewystępujących w rzeczywistych systemach obsługi (transmisji) jednostek danych, jest konsekwencją przyjęcia w dyscyplinie szeregowania GPS przepływowego modelu obsługi strumieni jednostek danych, zakładającego nieskończoną podzielność strumienia danych.

Możliwość zachodzenia omawianych zdarzeń w serwerze GPS wymaga stosowania w praktyce takich emulacji dyscypliny szeregowania GPS, w których implementowane są mechanizmy pozwalające na rozstrzygnięcie o kolejności kolidujących jednostek danych, tzn. na aproksymację wyników działania serwera GPS. Celem aproksymacji działania serwera GPS są takie zmiany czasów zakończenia obsługi jednostek danych, uzyskanych w serwerze GPS, które pozwalają na uszeregowanie jednostek danych zgodnie z niemalejącym czasem zakończenia obsługi.

9.7.5. EMULACJA DYSCYPLINY SZEREGOWANIA Z UOGÓLNIONYM PODZIAŁEM CZASU PROCESORA

Zastosowanie modelu przepływowego ruchu w dyscyplinie sprawiedliwego szeregowania GPS powoduje, że serwer GPS generuje zdarzenia charakteryzujące się kolizją jednostek danych należących do poszczególnych, jednocześnie obsługiwanych

przepływów, tzn. algorytm GPS nie zachowuje kolejności napływu jednostek danych do obsługi. Konieczność odtwarzania czasów zakończenia obsługi jednostek danych różnych przepływów, szeregowanych zgodnie z dyscypliną GPS, wymaga jego emulacji, równoważnej dyskretnej aproksymacji przepływowej szeregowania przepływu ciągłego.

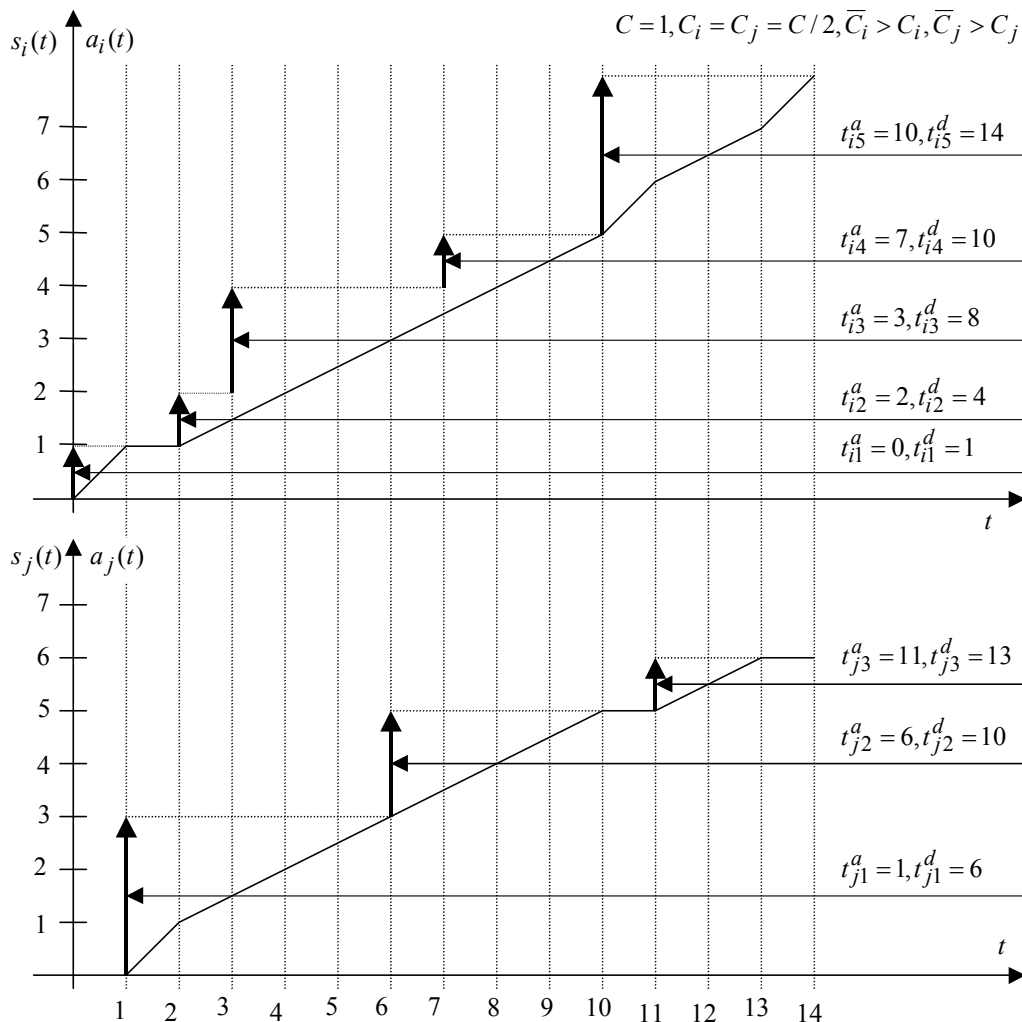
Usunięcie kolizji jednostek danych jest zadaniem algorytmu aproksymacji serwera GPS. Celem aproksymacji jest zbudowanie z wielu ciągów jednostek danych, należących do jednocześnie obsługiwanych przepływów, jednego ciągu jednostek danych uporządkowanych zgodnie z niemalejącym czasem zakończenia ich obsługi.

Jednym ze schematów dyskretnej aproksymacji dyscypliny szeregowania GPS jest schemat PGPS (ang. *Packet-by-packet GPS*), który generuje bardzo dobre przybliżenia wyników działania serwera GPS nawet wtedy, gdy jednostki danych mają zmienną długość. Algorytm szeregowania PGPS, lub inaczej WFQ (ang. *Weighted Fair Queueing*), jest wersją dyskretną emulatora przepływowej dyscypliny szeregowania GPS.

Emulacja dyscypliny szeregowania GPS przez PGPS polega na tym, że dyscyplina PGPS wymusza względną kolejność obsługi wszystkich jednostek danych należących do różnych przepływów, która jest zgodna z kolejnością napływu jednostek danych do stanowiska obsługi.

Niech t_k^d będzie przewidywanym czasem zakończenia obsługi k -tej jednostki danych, obsługiwanej zgodnie z dyscypliną szeregowania GPS. Aproksymacja działania serwera GPS przez serwer PGPS opiera się na założeniu, że dobrą aproksymacją jego działania jest schemat obsługi, który zachowuje pracę i obsługuje pakiety zgodnie z rosnącymi wartościami czasów t_k^d . Przy założeniu, że w chwili τ ($t_k^d > \tau$) serwer PGPS nie jest zajęty, możliwe jest zdarzenie, że w chwili τ następna w kolejności, ze względu na czas zakończenia obsługi przez serwer GPS, jednostka danych jeszcze nie nadeszła do systemu obsługi. Ponieważ czas napływu tej jednostki danych jest nieznanym w chwili τ , nie jest więc możliwe, aby serwer aproksymujący PGPS był jednocześnie serwerem zachowującym pracę i serwerem obsługującym jednostki danych w kolejności niemalejących czasów zakończenia obsługi przez algorytm GPS. Z tego powodu serwer aproksymujący PGPS pobiera do obsługi pierwszą w kolejności jednostkę danych, która zostałaby obsłużona jako pierwsza przez serwer GPS, gdyby żadne dodatkowe jednostki danych nie miały napłynąć do serwera GPS po czasie τ . Oznacza to, że serwer PGPS obsługuje jednostki danych w kolejności niemalejących czasów napływu. Tak zdefiniowany serwer działa zgodnie z algorytmem PGPS i odtwarza kolejność wprowadzania pakietów do systemu obsługi GPS.

Różnice działania algorytmów GPS i PGPS wynikają z ich właściwości. Algorytm GPS nie zachowuje kolejności obsługi jednostek danych należących do różnych przepływów; algorytm PGPS zachowuje natomiast kolejność obsługi jednostek danych należących do wielu, jednocześnie obsługiwanych przez to samo stanowisko obsługi, różnych przepływów.



Rysunek 9.18. Brak zachowania kolejności obsługi jednostek danych oraz kolizje jednostek danych wnoszone przez algorytm szeregowania GPS

Niezachowanie kolejności obsługi jednostek danych oraz kolizje jednostek danych wnoszone przez algorytm szeregowania GPS zilustrowano na rysunku 9.18:

➤ pierwsze pięć jednostek danych i -tego przepływu mają – odpowiednio – następujące czasy napływu i zakończenia obsługi:

$$(t_{i1}^a = 0, t_{i1}^d = 1), (t_{i2}^a = 2, t_{i2}^d = 4), (t_{i3}^a = 3, t_{i3}^d = 8),$$

$$(t_{i4}^a = 7, t_{i4}^d = 10), (t_{i5}^a = 10, t_{i5}^d = 14),$$

➤ pierwsze trzy jednostki danych j -tego przepływu mają – odpowiednio – następujące czasy napływu i zakończenia obsługi:

$$(t_{j1}^a = 1, t_{j1}^d = 6), (t_{j2}^a = 6, t_{j2}^d = 10) \text{ oraz } (t_{j3}^a = 11, t_{j3}^d = 13),$$

gdzie:

t_{ik}^a i t_{jl}^a , dla $k = 1, 2, 3, 4, 5$ i $l = 1, 2, 3$ – czas napływu do obsługi kolejnych jednostek danych i -tego i j -tego przepływu,

t_{ik}^d i t_{jl}^d , dla $k = 1, 2, 3, 4, 5$ i $l = 1, 2, 3$ – czas zakończenia obsługi kolejnych jednostek danych i -tego i j -tego przepływu.

W omawianym przykładzie charakterystyczne dla działania algorytmu szeregowania GPS są zależności pomiędzy czasami napływu i zakończenia obsługi następujących jednostek danych:

➤ pierwszej z j -tego przepływu i drugiej z i -tego przepływu, dla których zachodzą zależności: $t_{j1}^a < t_{i2}^a$ i $t_{j1}^d > t_{i2}^d$,

➤ trzeciej z j -tego przepływu i piątej z i -tego przepływu, dla których zachodzą zależności: $t_{j3}^a > t_{i5}^a$ i $t_{j3}^d < t_{i5}^d$,

powodujących ich kolizje w czasie opuszczania serwera GPS.

Działanie algorytmu PGPS, porządkującego kolejność czasów zakończenia obsługi jednostek danych, polega na tym, że w serwerze PGPS obsługiwana jest w danej chwili tylko jedna jednostka danych, a jednostki danych są obsługiwane w kolejności ich napływu do serwera. Różnica pomiędzy działaniem algorytmów GPS i PGPS sprowadza się do tego, że:

➤ jednoczesna obsługa jednostek danych, należących do przepływów separowanych przez algorytm GPS, jest zastąpiona sekwencyjną obsługą pojedynczych jednostek danych przez algorytm PGPS,

➤ obsługa jednostek danych w algorytmie GPS przez dedykowane współdzielone zasoby jest zastąpiona w algorytmie PGPS obsługą jednostek danych przez wszystkie zasoby.

Różnice pomiędzy wynikami działania algorytmów PGPS i GPS dla przykładu z rysunku 9.18, tzn. pomiędzy czasami zakończenia obsługi kolejnych jednostek danych, przedstawiono w tabeli 9.1.

Tabela 9.1. Porównanie czasów zakończenia obsługi jednostek danych w algorytmach GPS i PGPS dla przykładu z rysunku 9.18

Numer jednostki danych	i -ty przepływ					j -ty przepływ		
	1	2	3	4	5	1	2	3
Czas napływu	0	2	3	7	10	1	6	11
Rozmiar jednostki danych	1	1	2	1	3	3	2	1
Algorytm GPS	1	4	8	10	14	6	10	13
Algorytm PGPS	1	5	7	10	13	4	9	14

Algorytm PGPS porządkuje czasy zakończenia obsługi jednostek danych, eliminując kolizje jednostek danych wnoszone przez algorytm GPS. W omawianym przykładzie dwie sekwencje jednostek danych, należących do dwóch separowanych przepływów i -tego oraz j -tego, generowanych przez algorytm GPS:

➤ dla i -tego przepływu:

$$(t_{i1}^a = 0, t_{i1}^d = 1), (t_{i2}^a = 2, t_{i2}^d = 4), (t_{i3}^a = 3, t_{i2}^d = 8),$$

$$(t_{i4}^a = 7, t_{i4}^d = 10) (t_{i5}^a = 10, t_{i5}^d = 14),$$

➤ dla j -tego przepływu:

$$(t_{j1}^a = 1, t_{j1}^d = 6), (t_{j2}^a = 6, t_{j2}^d = 10) \text{ oraz } (t_{j3}^a = 11, t_{j3}^d = 13),$$

są zastępowane przez jedną sekwencję jednostek danych, generowaną przez algorytm PGPS:

$$(t_{i1}^a = 0, \hat{t}_{j1}^d = 1), (t_{j1}^a = 1, \hat{t}_{j1}^d = 4), (t_{i2}^a = 2, \hat{t}_{i2}^d = 5), (t_{i3}^a = 3, \hat{t}_{i2}^d = 7),$$

$$(t_{j2}^a = 6, \hat{t}_{j2}^d = 9), (t_{i4}^a = 7, \hat{t}_{i4}^d = 10), (t_{i5}^a = 10, \hat{t}_{i5}^d = 13) \text{ oraz } (t_{j3}^a = 11, \hat{t}_{j3}^d = 14),$$

gdzie:

t_k^d – czas zakończenia obsługi k -tej jednostki danych w algorytmie GPS,

\hat{t}_k^d – czas zakończenia obsługi k -tej jednostki danych w algorytmie PGPS.

W algorytmie GPS czas zakończenia obsługi drugiej jednostki danych i -tego przepływu jest krótszy od czasu zakończenia obsługi pierwszej jednostki danych j -tego przepływu ($t_{i2}^d < t_{j1}^d$), mimo że czas napływu tej ostatniej jest krótszy od czasu napływu drugiej i -tego przepływu ($t_{j1}^a < t_{i2}^a$).

W algorytmie PGPS obsługa drugiej jednostki danych i -tego przepływu rozpoczyna się nie wcześniej niż kończy się obsługa pierwszej jednostki danych j -tego przepływu. Czas zakończenia obsługi drugiej jednostki danych i -tego przepływu jest dłuższy od czasu zakończenia obsługi pierwszej jednostki danych z j -tego przepływu ($\hat{t}_{j2}^d > \hat{t}_{j1}^d$).

Kolejność obsługi tych jednostek danych jest zgodna z kolejnością ich napływu do obsługi – $t_{j1}^a < t_{i2}^a$. Podobne do wymienionych zmiany algorytm PGPS wprowadza do obsługi jednostek danych: trzeciej j -tego przepływu i piątej i -tego przepływu.

Efektom działania algorytmu PGPS, w porównaniu z działaniem algorytmu GPS, jest zmiana czasów zakończenia obsługi jednostek danych oraz zmiana opóźnień jednostek danych. Zmiana czasów zakończenia obsługi jednostek danych może polegać zarówno na zmniejszeniu, jak i zwiększeniu czasu zakończenia obsługi. Warto zwrócić uwagę na to, że zmiana czasów zakończenia obsługi poszczególnych jednostek danych oraz zmiana ich opóźnień nie musi wpływać na czas obsługi sekwencji jednostek danych należących do poszczególnych przepływów.

9.7.6. PORÓWNANIE DZIAŁANIA DYSCYPLIN SZEREGOWANIA GPS I PGPS

Sposób aproksymacji działania dyscypliny szeregowania GPS przez algorytm PGPS oznacza na ogół zmianę czasów zakończenia obsługi jednostek danych. Naturalną miarą jakości omawianej aproksymacji jest wydłużenie czasu obsługi jednostki danych w algorytmie PGPS w porównaniu z czasem obsługi jednostki danych przez algorytm GPS, tj. maksymalny przyrost czasu zakończenia obsługi jednostki danych wnoszony przez algorytm PGPS w porównaniu z czasem zakończenia obsługi jednostki danych przez algorytm GPS.

Gdy obsługiwane są dwa separowane przepływy jednostek danych, wartość maksymalnej różnicy pomiędzy czasami zakończenia obsługi dowolnej k -tej jednostki danych, należącej do dowolnego separowanego przepływu, przez algorytm PGPS i algorytm GPS jest dana wyrażeniem:

$$\hat{t}_k^d - t_k^d \leq \frac{l_{\max}}{C},$$

gdzie:

t_k^d – czas zakończenia obsługi k -tej jednostki danych przez algorytm GPS,

\hat{t}_k^d – czas zakończenia obsługi k -tej jednostki danych przez algorytm PGPS,

l_{\max} – maksymalna długość jednostki danych,

C – stała szybkość obsługi zagregowanego przepływu jednostek danych (pojemność kanału transmisyjnego).

Wartość maksymalnej różnicy czasów zakończenia obsługi zależy od maksymalnego czasu trwania kolizji jednostek danych, należących do dwóch różnych separowanych przepływów i obsługiwanych przez algorytm GPS.

Maksymalny czas trwania kolizji jednostek danych występuje wtedy, gdy k -ta jednostka danych i -tego przepływu, o minimalnej długości l_{\min} , napływa do serwera GPS tuż po napływie l -tego pakietu j -tego przepływu o maksymalnej długości l_{\max} , tzn. spełniony jest warunek: $t_{jl}^a < t_{ik}^a$ (rysunek 9.19).

W takim przypadku:

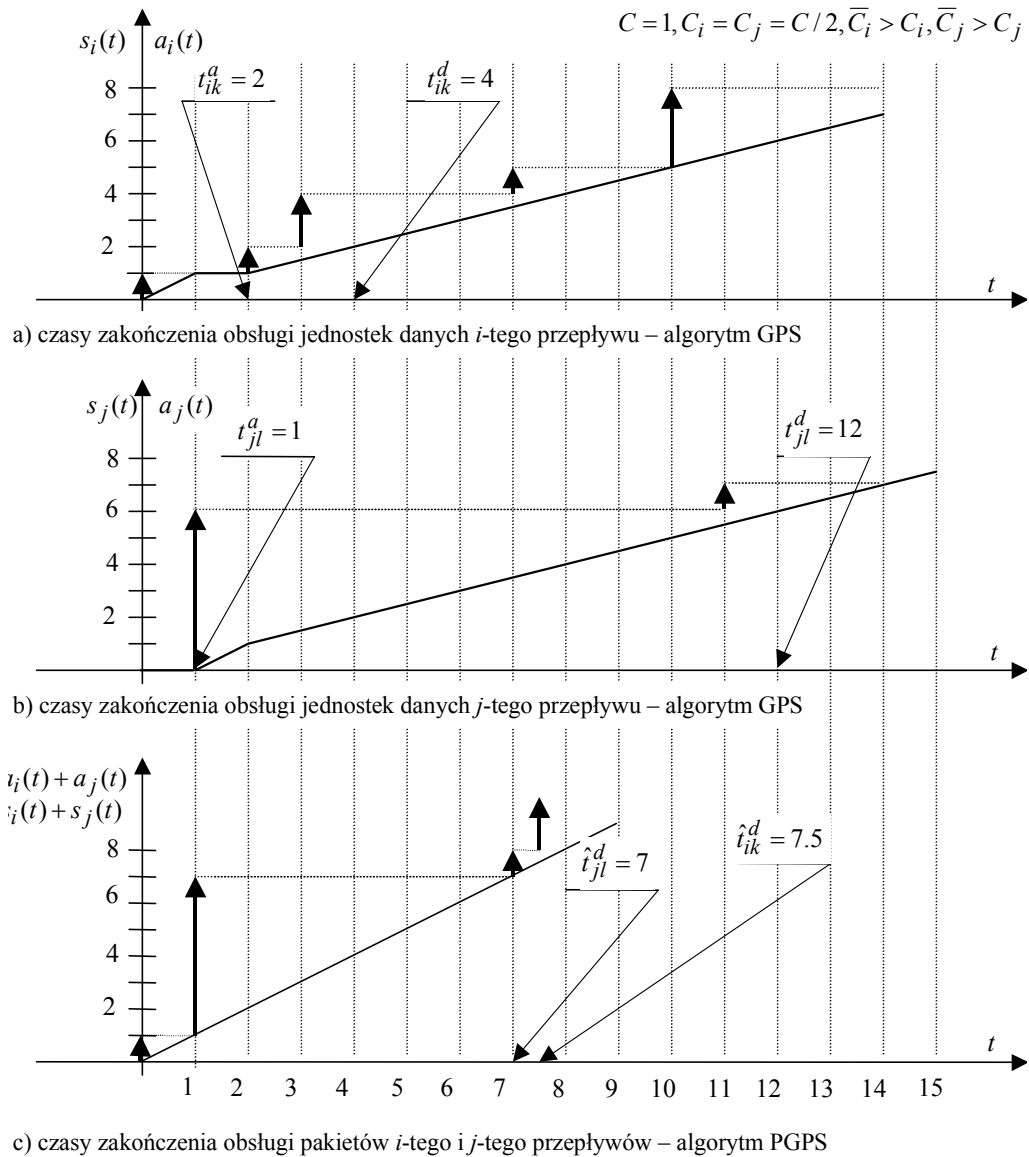
➤ czas zakończenia obsługi l -tej jednostki danych j -tego przepływu przez algorytm PGPS jest równy sumie czasu napływu jednostki danych i czasu jej obsługi przez serwer PGPS:

$$\hat{t}_{jl}^d = t_{jl}^d + \frac{l_{\max}}{C}, \text{ a stąd: } \hat{t}_{jl}^d - t_{jl}^d = \frac{l_{\max}}{C},$$

➤ czas zakończenia obsługi k -tej jednostki danych i -tego przepływu przez algorytm PGPS jest równy sumie czasu zakończenia obsługi l -tej jednostki danych j -tego

przepływu \hat{t}_{jl}^d oraz czasu obsługi k -tego pakietu i -tego przepływu przez algorytm PGPS:

$$\hat{t}_{ik}^d = \hat{t}_{jl}^d \frac{l_{\min}}{C} = t_{jl}^a + \frac{l_{\max}}{C} + \frac{l_{\min}}{C} \leq t_{ik}^a + \frac{l_{\max}}{C} + \frac{l_{\min}}{C} \leq t_{ik}^d + \frac{l_{\max}}{C}.$$



Rysunek 9.19. Różnice czasów zakończenia obsługi jednostek danych przez algorytmy GPS i PGPS

Przykład 9.1

W przykładzie prezentowanym na rysunku 9.19 przyjęto, że: czasy napływu dwóch jednostek danych, należących do dwóch różnych przepływów, oraz długości jednostek danych są następujące:

$$t_{jl}^a = 1, t_{ik}^a = 2, l_{ik} = l_{\min} = 1, l_{jl} = l_{\max} = 6 \text{ i } C = 1.$$

Obsługę tych jednostek danych przez serwer GPS, przy założeniu, że $C_i = C_j = C/2$, charakteryzują wartości czasów zakończenia ich obsługi: $t_{ik}^d = 4$ ($d_{ik} = t_{ik}^d - t_{ik}^a = 2$) i $t_{jl}^d = 12$ ($d_{jl} = t_{jl}^d - t_{jl}^a = 11$) oraz fakt, że obsługa k -tej jednostki danych, napływającej później niż l -ta jednostka danych ($t_{ik}^a = 2 > t_{jl}^a = 1$) jest zakończona przed zakończeniem obsługi l -tej jednostki danych ($t_{ik}^d = 4 < t_{jl}^d = 12$).

Obsługę tych jednostek danych przez serwer PGPS charakteryzują wartości czasów zakończenia ich obsługi: $\hat{t}_{ik}^d = 7,5$ ($\hat{d}_{ik} = \hat{t}_{ik}^d - t_{ik}^a = 5,5$) i $\hat{t}_{jl}^d = 7$ ($\hat{d}_{jl} = \hat{t}_{jl}^d - t_{jl}^a = 6$) oraz fakt, że obsługa k -tej jednostki danych, napływającej później niż l -ta jednostka danych ($t_{ik}^a = 2 > t_{jl}^a = 1$) jest zakończona po zakończeniu obsługi l -tej jednostki danych ($\hat{t}_{ik}^d = 7,5 > \hat{t}_{jl}^d = 7$).

Z porównania czasów zakończenia obsługi k -tej i l -tej jednostek danych przez serwery GPS i PGPS, wynika, że:

$$\left| \hat{t}_{jl}^d - t_{jl}^d \right| = 5 \leq \frac{l_{\max}}{C} = 6 \text{ oraz } \left| \hat{t}_{ik}^d - t_{ik}^d \right| = 3,5 \leq \frac{l_{\max}}{C} = 6,$$

tzn. skrócenie czasu zakończenia obsługi l -tej jednostki danych odbywa się kosztem wydłużenia czasu zakończenia obsługi k -tej jednostki danych, a wartość zmiany, polegającej na wydłużeniu czasu obsługi w serwerze PGPS, jest ograniczona z góry przez wartość czasu obsługi jednostki danych o maksymalnej długości.

Jeżeli serwer GPS obsługuje jednocześnie n separowanych przepływów jednostek danych i jeżeli w serwerze GPS następuje kolizja jednostek danych o maksymalnej długości, to wartość maksymalnej różnicy czasów zakończenia obsługi dowolnej z kolidujących jednostek danych, obsługiwanych w dowolnej kolejności przez algorytm PGPS i algorytm GPS, jest dana wyrażeniem:

$$\hat{t}_k^d - t_k^d \leq \frac{(n-1)l_{\max}}{C}.$$

Konsekwencją istnienia górnego ograniczenia różnicy czasów zakończenia obsługi dowolnej k -tej jednostki danych przez algorytm PGPS i przez algorytm GPS jest górne ograniczenie różnicy ilości ruchu (obciążenia serwera) i -tego przepływu obsługowanego zgodnie z algorytmem PGPS ($s_i(0, t)$) i zgodnie z algorytmem GPS ($\hat{s}_i(0, t)$) w przedziale czasu $(0, t]$.

Niech t będzie czasem, w którym rozpoczyna się obsługa k -tej jednostki danych i -tego przepływu o długości l_{ik} przez algorytm PGPS. Czas zakończenia obsługi tej jednostki danych przez algorytm PGPS jest równy $\hat{t}_{ik}^d = t + l_{ik}/C$, czas zakończenia obsługi tej jednostki danych przez serwer GPS jest natomiast równy t_{ik}^d . Ze względu na to, że algorytmy PGPS i GPS zachowują kolejność obsługi jednostek danych należących do tego samego przepływu, prawdziwa jest zależność:

$$s_i(0, t_{ik}^d) = \hat{s}_i(0, \hat{t}_{ik}^d) = \hat{s}_i(0, t) + \frac{l_{ik}}{C} = \hat{s}_i(0, t) + l_{ik}.$$

Ponieważ różnica czasów zakończenia obsługi tej samej jednostki danych przez algorytmy PGPS i GPS jest ograniczona z góry, tzn.:

$$t_{ik}^d \geq \frac{t + l_{ik}}{C} - \frac{l_{i \max}}{C} = t + \frac{l_{ij} - l_{i \max}}{C},$$

z nierówności:

$$s_i(t_{ik}^d) \geq s_i\left(0, t + \frac{l_{ij} - l_{i \max}}{C}\right)$$

wynika więc, że:

$$\hat{s}_i(0, t) + l_{ik} \geq s_i(0, t) + l_{ik} - l_{i \max} \Rightarrow s_i(0, t) - \hat{s}_i(0, t) \leq l_{i \max} \leq l_{\max},$$

gdzie l_{\max} jest maksymalną długością jednostek danych wszystkich przepływów.

Oznacza to, że różnica ilości ruchu i -tego przepływu (w bitach), obsługiwanego w przedziale czasu $(0, t]$ przez algorytmy PGPS i GPS jest ograniczona maksymalną długością pakietu.

Kolejną konsekwencją istnienia górnego ograniczenia różnicy czasów zakończenia obsługi dowolnej k -tej jednostki danych i -tego przepływu przez algorytmy PGPS i przez GPS jest górne ograniczenie różnicy ilości ruchu oczekującego na obsługę. Ponieważ liczba napływających do obsługi jednostek danych jest niezależna od algorytmu szeregowania, tzn. $a_i(\tau, t) = \hat{a}_i(\tau, t)$, więc:

$$\hat{q}_i(0, t) - q_i(0, t) = [\hat{a}_i(0, t) - \hat{s}_i(0, t)] - [a_i(0, t) - s_i(0, t)] = s_i(0, t) - \hat{s}_i(0, t) \leq l_{\max},$$

gdzie $q_i(0, t)$ i $\hat{q}_i(0, t)$ są ilościami ruchu i -tego przepływu oczekujących na obsługę, odpowiednio przez algorytmy GPS i PGPS.

Tak szacowane czasy zakończenia obsługi jednostek danych są stosowane do znakowania poszczególnych jednostek danych, na podstawie którego jest ustalana kolejność dalszej obsługi jednostek danych. Jednostki danych są szeregowane, tzn. są wybierane do transmisji w kolejności niemalejących czasów zakończenia obsługi.

Zastąpienie dyscypliny szeregowania GPS przez jego emulację PGPS, tzn. zastąpienie modelu przepływowego modelem dyskretnym, powoduje, że w zadaniu szacowania czasu zakończenia obsługi jednostki danych konieczne jest uwzględnienie:

- opóźnienia pakietyzacji w każdym z węzłów na trasie źródło–ujście,
- czasu oczekiwania jednostki danych na zakończenie przetwarzania jednostki danych poprzedzającej daną jednostkę danych w węzłach na trasie źródło–ujście.

Jeżeli strumień generowanych przez źródło jednostek danych jest ograniczony wartościami parametrów algorytmu sterowania dostępem (algorytm ciekącego wiadra), to po przyjęciu, że:

- C_k – całkowita pojemność łącza pomiędzy k -tym węzłem a $(k+1)$ -szym węzłem ($k = 1, 2, \dots, m-1$),
- C_{ik} i \bar{C}_{ik} – odpowiednio – pojemności gwarantowane i wykorzystywane do obsługi i -tego przepływu w łączu pomiędzy k -tym węzłem a $(k+1)$ -szym węzłem takie, że $v_{ZNi} \leq C_{ik} \leq \bar{C}_{ik}$ dla każdego $k = 1, 2, \dots, m-1$,
- m – długość trasy wyrażona liczbą wszystkich węzłów na trasie pomiędzy systemami końcowymi (liczba węzłów pośrednich plus węzeł ujściowy),
- $l_{i \max}$ – maksymalna długość jednostki danych i -tego przepływu,
- n_k – liczba wszystkich różnych przepływów obsługiwanych w k -tym węźle węzłem ($k = 1, 2, \dots, m$) trasy i -tego przepływu,
- $L_{k \max}$ – maksymalna długość jednostek danych wszystkich przyływów obsługiwanych w k -tym węźle trasy i -tego przepływu,

górne ograniczenie opóźnienia, wnoszonego przez algorytm WFQ dla jednostek danych należących do i -tego przepływu, jest dane wyrażeniem:

$$\begin{aligned}
 d_{\text{WFQ}i \max} &\leq \frac{b_{ZNi}}{\bar{C}_{ik}} + \frac{l_{i \max}}{v_{ZNi}} + \sum_{k=2}^{m-1} \frac{l_{i \max}}{C_{ik}} + \sum_{k=1}^m \frac{(n_k - 1)L_{k \max}}{C_k} + \sum_{k=1}^{m-1} \tau_k \leq \\
 &\leq \frac{b_{ZNi}}{C_{ik}} + \frac{l_{i \max}}{v_{ZNi}} + \sum_{k=2}^{m-1} \frac{l_{i \max}}{C_{ik}} + \sum_{k=1}^m \frac{(n_k - 1)L_{k \max}}{C_k} + \sum_{k=1}^{m-1} \tau_k \leq \\
 &\leq \frac{b_{ZNi}}{v_{ZNi}} + \sum_{k=1}^{m-1} \frac{l_{i \max}}{v_{ZNi}} + \sum_{k=1}^m \frac{(n_k - 1)L_{k \max}}{C_k} + \sum_{k=1}^{m-1} \tau_k \leq \\
 &= \frac{b_{ZNi} + (m-1)l_{i \max}}{v_{ZNi}} + \sum_{k=1}^m \frac{(n_k - 1)L_{k \max}}{C_k} + \sum_{k=1}^{m-1} \tau_k,
 \end{aligned}$$

gdzie:

- b_{ZNi}/v_{ZNi} – górne ograniczenie czasu oczekiwania jednostki danych wnoszonego przez algorytm ciekącego wiadra, którego wartość jest równa czasom transmisji b_{ZNi} jednostek danych na trasie o pojemności nie mniejszej od v_{ZNi} ,

- $l_{i \max}/v_{ZNi}$ – górne ograniczenie opóźnienia kompletowania (pakietyzacji), tj. czasu oczekiwania na skompletowanie jednostki danych i -tego przepływu w dowolnym z węzłów trasy z wyłączeniem węzła ujściowego,
- $l_{i \max}/C_{ik}$ – opóźnienia kompletowania tj. czasu oczekiwania na skompletowanie jednostki danych i -tego przepływu w dowolnym z węzłów trasy z wyłączeniem węzła źródłowego i ujściowego, do którego jednostki danych transmitowane są przez łącze o pojemności C_{ik} , takiej że $C_{ik} \geq v_{ZNi}$ dla $k = 1, 2, \dots, m - 1$,
- $[(m - 1)l_{i \max}]/v_{ZNi}$ – suma górnych ograniczeń opóźnień kompletowania jednostek danych i -tego przepływu we wszystkich węzłach, z wyłączeniem węzła ujściowego, trasy złożonej z m łączy o pojemnościach C_{ik} , takich że $C_{ik} \geq v_{ZNi}$ dla $k = 1, 2, \dots, m - 1$,
- $L_{k \max}/C_k$ – górne ograniczenie opóźnienia wnoszonych przez algorytm PGPS, tj. algorytm odtwarzania kolejności jednostek danych we wszystkich kolejnych m węzłach trasy,
- τ_k ($k = 1, 2, \dots, m - 1$) – czas propagacji w łączu pomiędzy k -tym węzłem a $(k + 1)$ -szym węzłem.

Jeżeli $n_k = 2$ oraz $L_{k \max} = l_{\max}$ dla każdego $k = 1, 2, \dots, m$, to całkowite opóźnienie jednostek danych i -tego przepływu, którego ruch:

$$d_{WFQ i \max} \leq \frac{b_{ZNi} + (m - 1)l_{i \max}}{v_{ZNi}} + \sum_{k=1}^m \frac{l_{\max}}{C_k} + \sum_{k=1}^{m-1} \tau_k.$$

Różnica całkowitych, maksymalnych opóźnień jednostek danych i -tego przepływu przekazywanego przez m kolejnych węzłów sieci, którego ruch jest ograniczony wartościami parametrów algorytmu ciekącego wiadra (b_{ZNi} , v_{ZNi}), wnoszone przez dyscyplinę szeregowania GPS i WFQ (PGPS) jest następująca (przy założeniu, że i -ty przepływ jest jednym z dwóch przepływów obsługiwanych w węzłach trasy):

$$d_{WFQ i \max} - d_{GPS i \max} = \frac{(m - 1)l_{i \max}}{v_{ZNi}} + \sum_{k=1}^m \frac{l_{\max}}{C_k}.$$

9.7.7. UPROSZCZONE WERSJE WAŻONYCH SPRAWIEDLIWYCH ALGORYTMÓW SZEREGOWANIA

Ze względu na to, że obliczanie czasów zakończenia obsługi pakietów w algorytmie PGPS (WFQ), będących praktycznymi emulacjami dyscypliny szeregowania GPS, jest zadaniem złożonym i czasochłonnym, praktyczna przydatność czasów zakończenia obsługi jednostek danych jest ograniczona. Przydatność ta maleje wraz ze wzrostem szybkości transferu jednostek danych.

Z powodu dużej złożoności obliczeniowej emulatorów idealnej dyscypliny szeregowania GPS, ograniczającej możliwości ich stosowania do sterowania ruchem w sieciach, praktycznie wykorzystywane są różne wersje przybliżone emulatora PGPS. Zadaniem tych przybliżonych wersji jest szybkie szacowanie czasów zakończenia obsługi jednostek danych, stosowane do ustalania kolejności, w jakiej jednostki danych różnych, jednocześnie obsługiwanych, przepływów są kierowane do dalszej obsługi na trasach w sieci.

9.7.7.1. ALGORYTM WAŻONEGO SPRAWIEDLIWEGO SZEREGOWANIA DLA NAJGORSZEGO PRZYPADKU

Algorytm sprawiedliwego w najgorszym przypadku ważonego sprawiedliwego kolejkowania W2FQ (ang. *worst-case fair – Weighted Fair Queueing*) jest rozszerzoną wersją algorytmu sprawiedliwego szeregowania WFQ, w którym do ustalania kolejności transmisji (obsługi) jednostek danych wykorzystywane są zarówno czasy rozpoczęcia, jak i czasy zakończenia obsługi jednostek danych przez algorytm GPS. Jednostka danych, wybierana jako pierwsza do transmisji, to jednostka danych o najkrótszym czasie zakończenia obsługi spośród wszystkich jednostek danych, obsługiwanych w danej chwili zgodnie z algorytmem GPS.

Podstawą algorytmu szeregowania W2FQ jest idea szeregowania zadań w kolejności niemalejących czasów przetwarzania (ang. *shortest processing time*), minimalizująca średni czas przepływu. Do uszeregowania optymalnego prowadzi przydzielanie najkrótszego zadania do aktualnie wolnego stanowiska obsługi.

Algorytm W2FQ, w porównaniu do algorytmu WFQ, zapewnia: taką samą wartość górnego ograniczenia opóźnienia jednostki danych, bardziej sprawiedliwe wykorzystanie zasobów, wygładzony przeplot jednostek danych oraz skrócenie średniego opóźnienia jednostek danych. Praktyczne znaczenie omawianego algorytmu szeregowania jest ograniczone złożonością zadania obliczania czasów zakończenia obsługi pakietów w systemie GPS.

9.7.7.2. HIERARCHICZNY ALGORYTM WAŻONEGO SPRAWIEDLIWEGO SZEREGOWANIA

Algorytm WFQ można stosować do budowy hierarchicznych algorytmów ważonego, sprawiedliwego szeregowania HWFQ (ang. *Hierarchical Weighted Fair Queueing*), tzn. takich algorytmów, w których sprawiedliwe algorytmy szeregowania są stosowane na kilku różnych poziomach agregacji przepływów jednostek danych.

Stosowanie hierarchicznych algorytmów szeregowania oznacza, że rozdział zasobów i dystrybucja niewykorzystanych zasobów realizowane są hierarchicznie. W pierwszej kolejności zasoby są przydzielane wyróżnionym grupom przepływów, a te z kolei wyróżnionym podgrupom danej grupy itd. W odwrotnym kierunku odbywa się redystrybucja nadmiarowych, aktualnie niewykorzystywanych zasobów. Nad-

miarowe zasoby podgrupy są rozdzielane w pierwszej kolejności w ramach danej podgrupy. W razie niewykorzystania nadmiarowych zasobów podgrupy przez przepływy należące do tej podgrupy, nieprzydzielone nadmiarowe zasoby danej podgrupy są przekazywane do rozdziału w grupie przepływów na wyższy poziom hierarchii.

Stosowanie algorytmów hierarchicznego rozdziału zasobów i hierarchicznego szeregowania przepływów jest uzasadnione tylko wtedy, gdy rozdzielanie zasobów pomiędzy poszczególne przepływy nie jest możliwe ze względu na ich dużą liczbę lub agregacja przepływów nie wpływa na jakość dostarczanych usług.

Hierarchiczny rozdział zasobów jest elastycznym schematem zarządzania zasobami, pozwalającym na dzielenie zasobów na kilku poziomach agregacji. Hierarchiczne algorytmy rozdziału zasobów dzielą pojemność dla hierarchii klas ruchu, gdzie każda z klas ruchu jest definiowana jako zbiór przepływów o pewnych wspólnych cechach charakterystycznych (np. przepływy z tym samym protokołem sieciowym, przepływy pomiędzy tymi samymi podsieciami). Liczba definiowanych poziomów hierarchii klas ruchu, a tym samym poziomów rozdziału zasobów, może być różna, zwłaszcza na najniższym poziomie hierarchii mogą występować przepływy identyfikowane poszczególnymi aplikacjami.

Podstawowa idea hierarchicznego rozdziału pojemności kanału transmisyjnego polega na tym, że pojemność jest rozdzielana pomiędzy różne domeny administracyjne, a w każdej z domen możliwy jest dalszy podział ze względu na rodzaje aplikacji. Z kolei pojemność, przydzielona dla każdego z rodzajów aplikacji, może być rozdzielana pomiędzy poszczególne przepływy generowane przez tę samą aplikację.

Algorytmy hierarchicznego rozdziału zasobów pozwalają na stosowanie tych samych struktur danych i algorytmów do realizacji wielu różnych usług sieci dla różnych domen administracyjnych i różnych aplikacji.

Cechą szczególną algorytmu HWFQ jest to, że klasy ruchu, znajdujące się na tym samym poziomie hierarchii, mogą współdzielić między sobą niewykorzystaną wcześniej przyznaną (początkową) pojemność lub obsługiwać ruch w ilościach zgodnych z wcześniej przyznaną (początkową) pojemnością. Zasada ta dotyczy klas ruchu na każdym ze zdefiniowanych poziomów hierarchii. Pojemności początkowe, przyznawane klasom ruchu konkurującym o zasoby, są wyznaczane zgodnie z algorytmem sprawiedliwego rozdziału zasobów, zwłaszcza początkowy rozdział pojemności może być proporcjonalny do zapotrzebowania klas ruchu na pojemność.

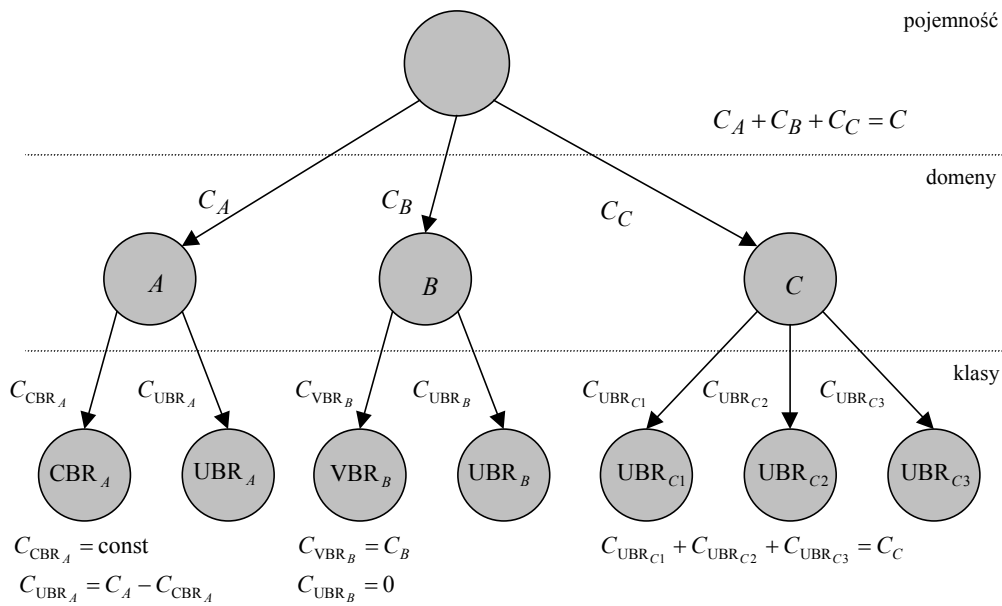
Przykład 9.2

Przykładem działania hierarchicznego, sprawiedliwego algorytmu rozdziału zasobów jest rozdział zasobów dla klas ruchu w trzech domenach administracyjnych (rysunek 9.20):

1. Domena A – dwie klasy ruchu:

- CBR_A – ruch wymagający obsługi ze stałą szybkością (ang. *Constant Bit Rate*),
- UBR_A – ruch o niezidentyfikowanej szybkości transmisji, wymagający obsługi zgodnie z zasadą najwyższej staranności (ang. *Unspecified Bit Rate*).

2. Domena B – dwie klasy ruchu:
- rt-VBR $_B$ – ruch o zmiennej szybkości, wymagający obsługi w czasie rzeczywistym (ang. *real-time Variable Bit Rate*),
 - UBR $_B$ – ruch o niezidentyfikowanej szybkości transmisji, wymagający obsługi bez gwarancji jakości usług.
3. Domena C – trzy klasy ruchu o niezidentyfikowanej szybkości transmisji, wymagające obsługi bez gwarancji jakości: UBR $_{C1}$, UBR $_{C2}$ oraz UBR $_{C3}$.



Rysunek 9.20. Rozdział zasobów dla klas ruchu w domenach administracyjnych jako przykład działania hierarchicznego, sprawiedliwego algorytmu rozdziału zasobów

Początkowy rozdział pojemności kanału transmisyjnego oznacza wyznaczenie pojemności dla każdej z trzech domen: C_A , C_B i C_C .

Pojemność C_A jest współdzielona przez dwie klasy ruchu: CBR $_A$ i UBR $_A$. Konieczność udostępnienia, w ramach pojemności przydzielonej domenie A , stałej pojemności dla ruchu CBR $_A$ ($C_{CBR_A} < C_A$; $C_{CBR_A} = \text{const}$), oznacza, że pojemność udostępniana dla ruchu UBR $_A$ jest stała ($C_{UBR_A} = C_A - C_{CBR_A} = \text{const}$).

Pojemność C_B jest współdzielona przez dwie klasy ruchu: VBR $_B$ i UBR $_B$. Jeżeli cała pojemność C_B , przydzielona domenie B , zostanie przyznana ruchowi VBR $_B$ ($C_{VBR_B} = C_B$), to w przypadku zastosowania algorytmu sprawiedliwego szeregowania przepływów ruch UBR $_B$ będzie obsługiwany tylko wtedy, gdy źródło ruchu VBR $_B$ nie będzie aktywne ($C_{UBR_B} = 0$).

Pojemność C_C jest współdzielona przez trzy klasy ruchu: UBR_{C1} , UBR_{C2} oraz UBR_{C3} . Jeżeli tym klasom ruchu zostaną przypisane współczynniki wagowe o różnych wartościach (np. $\alpha_1 < \alpha_2 < \alpha_3$), to pojemność C_C , przydzielona domenie C , zostanie rozdzielona pomiędzy klasy ruchu proporcjonalnie do wartości współczynników wagowych (np. $C_{UBR_{C1}} < C_{UBR_{C2}} < C_{UBR_{C3}}$). W przypadku zastosowania algorytmu sprawiedliwego szeregowania przepływów, pojemność niewykorzystana przez jedną z klas ruchu może być wykorzystana przez inną klasę ruchu tej samej domeny.

9.7.7.3. UPROSZCZENIA ALGORYTMU WAŻONEGO SPRAWIEDLIWEGO SZEREGOWANIA

Szacowanie czasu zakończenia obsługi jednostek danych przez algorytm PGPS, aproksymujący działanie algorytmu GPS, wymaga znajomości czasu zakończenia obsługi jednostki danych przez algorytm GPS. Obliczanie czasu zakończenia obsługi (transmisji) jednostki danych w systemie GPS jest zadaniem względnie łatwym wtedy, gdy wszystkie obsługiwane przepływy jednostek danych są przepływami z zaległościami, ponieważ:

- nie występuje przemieszczanie zasobów pomiędzy różnymi przepływami; wszystkie jednostki danych każdego przepływu są obsługiwane przez te same i ilościowo takie same zasoby,

- zasoby nie oczekują na jednostki danych; czasy zakończenia obsługi dwóch kolejnych jednostek danych różnią się czasem obsługi drugiej z jednostek danych.

Gdy wszystkie obsługiwane przepływy są przepływami z zaległościami, czas zakończenia obsługi k -tej jednostki danych i -tego przepływu jest równy sumie czasu zakończenia obsługi poprzedniego ($k - 1$)-szej jednostki danych i -tego przepływu oraz czasu obsługi k -tej jednostki danych, tzn.:

$$t_{ik}^d = t_{i,k-1}^d + \frac{l_{ik}}{C_i},$$

gdzie C_i jest częścią całkowitej pojemności C , gwarantowaną i -temu przepływowi przez algorytm sprawiedliwego rozdziału zasobów.

Szczególnym przypadkiem, kiedy zadanie szacowania czasu zakończenia obsługi jednostek danych jest także względnie łatwe, jest przepływ bez zaległości. W tym przypadku jednostki danych i -tego przepływu nie oczekują w kolejkach, a czas zakończenia obsługi k -tej jednostki danych i -tego przepływu jest sumą czasu generacji k -tej jednostki danych i czasu obsługi (transmisji) k -tej jednostki danych:

$$t_{ik}^d = t_{i,k-1}^a + \frac{l_{ik}}{C_i}.$$

W każdym innym przypadku szacowanie czasów zakończenia obsługi jednostek danych w systemie GPS jest zadaniem trudnym ze względu na to, że każda jednostka danych napływająca do systemu GPS i każda jednostka danych wpływająca z syste-

mu GPS mogą powodować zmianę stanów przepływów: z przepływów bez zaległości na przepływy z zaległościami i odwrotnie. Zmiana stanu przepływów powoduje zmiany czasów zakończenia obsługi jednostki danych należących do przepływów, których stan uległ zmianie w następstwie napływu nowej jednostki danych. Zmiana stanów przepływów wiąże się z:

- występowaniem przemieszczeń zasobów pomiędzy przepływami bez zaległości a przepływami z zaległościami, powodującymi zmiany czasów obsługi jednostek danych o takiej samej długości i należących do tych samych przepływów,

- występowaniem stanów oczekiwania zasobów na jednostki danych, powodującym występowanie przerw pomiędzy czasami zakończenia obsługi kolejnych jednostek danych tych samych przepływów.

Czas zakończenia obsługi jednostek danych można szacować za pomocą zależności:

$$t_{ik}^d = \max \left[t_{i,k-1}^d, w(t) \right] + \frac{l_{ik}}{C_i},$$

w której $w(t)$ jest liniową funkcją czasu (gdy wszystkie przepływy są przepływami z zaległościami $w(t) = 0$) aproksymującą wartość czasu rozpoczęcia obsługi k -tej jednostki danych i -tego przepływu.

Złożoność zadania szacowania czasu zakończenia obsługi jednostek danych polega na wyznaczeniu czasu zakończenia obsługi jednostki danych, której nadejście i obsługa są poprzedzone oczekiwaniem zasobów na jednostki danych, tzn. wyznaczeniu wartości funkcji $w(t)$.

Różne, przybliżone wersje algorytmu WFQ charakteryzują różne, uproszczone sposoby szacowania wartości funkcji $w(t)$.

Przykładem rozwiązania zadania szacowania wartości funkcji $w(t)$ jest algorytm SCFQ (ang. *Self-Clocked Fair Queueing*), którego działanie polega na szacowaniu wartości funkcji $w(t)$ za pomocą wartości czasu zakończenia obsługi jednostki danych, należącej do tego samego lub dowolnego innego przepływu. Oszacowaniem wartości funkcji $w(t)$, tzn. czasu zakończenia okresu oczekiwania zasobów na kolejną jednostkę danych należąca do danego przepływu, jest czas zakończenia obsługi aktualnie obsługiwanej jednostki danych, należącej do jednego z równolegle obsługiwanych przepływów.

Jeżeli:

- I_m jest podzbiorem zbioru wszystkich przepływów, których jednostki danych są obsługiwane w chwili zakończenia m -tej przerwy w obsłudze jednostek danych i -tego przepływu i rozpoczęcia obsługi k -tej jednostki danych i -tego przepływu,

- t_m jest czasem zakończenia i -tej przerwy w obsłudze jednostek danych i -tego przepływu i rozpoczęcia obsługi k -tej jednostki danych i -tego przepływu,

to oszacowaniem wartości funkcji $w(t)$ jest czas zakończenia obsługi l -tej, aktualnie obsługiwanej jednostki danych należącej do j -tego przepływu z podzbioru I_m , takiego że:

$$t_{jl}^d - t_m = \min_{r \in I_m} (t_{rs}^d - t_m).$$

Tak wyznaczone oszacowanie wartości funkcji $w(t)$ jest stosowane do określenia czasu zakończenia obsługi k -tej jednostki danych i -tego przepływu według zależności:

$$t_{ik}^d = \max \left[t_{i,k-1}^d, w(t) \right] + \frac{l_{ik}}{C_i} = \max \left[t_{i,k-1}^d, t_{jl}^d \right] + \frac{l_{ik}}{C_i}.$$

Aproksymacja wartości funkcji $w(t)$ dla danego przepływu, wartością czasu zakończenia obsługi jednej z aktualnie obsługiwanych jednostek danych innego przepływu, oznacza zwiększenie opóźnienia wnoszonego przez system obsługi. Wzrost opóźnienia jest liniową funkcją liczby jednocześnie obsługiwanych przepływów.

9.7.8. DYSCYPLINY SPRAWIEDLIWEGO SZEREGOWANIA Z PODZIAŁEM CZASU

Ze względu na złożoność obliczeniową i trudności praktycznych implementacji algorytmów emulujących dyscyplinę szeregowania GPS opracowano algorytmy, których jakość działania, w sensie sprawiedliwości rozdziału zasobów i opóźnienia jednostek danych wnoszonego przez algorytm, jest porównywalna z emulatorami dyscypliny GPS, a złożoność obliczeniowa jest znacznie mniejsza. Do algorytmów takich należą m.in. algorytmy sprawiedliwego szeregowania oparte na technice podziału czasu RR (ang. *Round Robin*), w których średni czas oczekiwania na obsługę zależy liniowo od wartości czasu obsługi jednostek danych.

9.7.8.1. ALGORYTM WRR

Działanie algorytmu cyklicznego ważonego WRR (ang. *Weighted Round Robin*) polega na tym, że liczba jednostek danych wybieranych do transmisji z cyklicznie obsługiwanych przepływów jest proporcjonalna do wagi przepływów.

W algorytmie WRR każdemu i -temu przepływowi przypisany jest współczynnik wagowy, tzn. liczba całkowita w_i , gdzie $i = 1, 2, \dots, n$, a n jest liczbą wszystkich przepływów współdzielących serwer.

Poszczególne przepływy wybierane są do obsługi sekwencyjnie; przepływowi przydzielone są szczeliny czasu ramki, a względna szybkość obsługi i -tego przepływu jest równa $w_i C / (w_1 + w_2 + \dots + w_n)$. Jeżeli szczelina ramki nie jest wykorzystana, to serwer przechodzi do obsługi następnego przepływu. Obsługa jednostki danych i -tego przepływu, napływającej tuż po zakończeniu szczeliny i -tego przepływu, jest możliwa dopiero w następnej szczeliny i -tego przepływu. Maksymalny czas oczekiwania jednostki danych i -tego przepływu na obsługę jest równy długości ramki pomniejszonej o długość szczeliny i -tego przepływu. Jeżeli obciążenie systemu obsługi jest duże, tzn. jeżeli prawie wszystkie szczeliny ramki są wykorzystywane, to czas oczekiwania jednostek danych na obsługę jest bliski czasowi trwania n szczelin, gdzie n jest liczbą wszystkich przepływów współdzielących serwer.

Algorytm WRR jest algorytmem sprawiedliwego podziału zasobów, jeżeli jednostki danych poszczególnych przepływów mają jednakową długość lub jeżeli znana jest średnia długość jednostek danych poszczególnych przepływów. Działanie algorytmu WRR jest dobrą aproksymacją algorytmu GPS wtedy, gdy liczba obsługiwanych przepływów jest mała lub długości jednostek danych są małe. Zwiększenie liczby przepływów lub wzrost długości jednostek danych oraz różnic ich długości pogarsza jakość działania algorytmu WRR jako algorytmu aproksymującego dyscyplinę szeregowania GPS.

Algorytm WRR jest, ze względu na efektywność sprawiedliwego podziału zasobów, wrażliwy na zwiększenie: liczby jednocześnie obsługiwanych przepływów, długości jednostek danych oraz różnicy długości jednostek danych.

Algorytm WRR jest gorszym emulatorem dyscypliny szeregowania GPS od algorytmu PGPS, który jest niewrażliwy na zwiększenie: liczby jednocześnie obsługiwanych przepływów, długości jednostek danych oraz różnicy długości jednostek danych. Obsługa jednostek danych o różnej długości przez algorytm PGPS jest, w porównaniu z algorytmem WRR, bardziej systematyczna, w tym znaczeniu, że aproksymacja dyscypliny szeregowania GPS przez algorytm PGPS jest niezależna od charakterystyk przepływów.

9.7.8.2. ALGORYTM DRR

Zasada działania algorytmu cyklicznego z deficytem DRR (ang. *Deficit Round Robin*) jest zbliżona do działania algorytmu WRR, z tym że w algorytmie DRR jest uwzględniana zmienna długość jednostek danych. Na potrzeby algorytmu DRR każdemu cyklicznie obsługiwanemu przepływowi przypisane są wartości:

- licznika deficytu (ang. *deficit counter*), którego wartość początkowa wynosi zero,
- kwoty (liczby) bitów (ang. *quantum*).

Transmisja jednostki danych przepływu oznacza zmniejszenie wartości kwoty, przypisanej przepływowi, o długość jednostki danych. Suma długości jednostek danych przepływu, obsługiwanych w jednym cyklu, jest nie większa od wartości kwoty ustalonej dla danego przepływu. W razie niewykorzystania ustalonej kwoty reszta kwoty jest zapisywana w liczniku deficytu przepływu. W kolejnym cyklu obsługi kwota, która jest stała dla danego przepływu, jest powiększana o stan licznika deficytu danego przepływu.

Jakość działania algorytmu DRR zależy od sposobu wyznaczania wartości kwot dla wielu jednocześnie obsługiwanych przepływów. Do emulacji algorytmu sprawiedliwego szeregowania wartości kwot są proporcjonalne do wartości wag przepływów, a kwota niewykorzystana przez przepływ w cyklu obsługi, z powodu braku jednostek danych do obsługi, jest tracona.

9.7.8.3. ALGORYTM CRR

Działanie algorytmu cyklicznego z kredytem CRR (ang. *Credit Round Robin*) jest podobne do algorytmu DRR, z tą różnicą, że suma długości jednostek danych prze-

pływu, obsługiwanych w jednym cyklu, może być większa od kwoty przyznanej dla przepływu. W razie przekroczenia kwoty różnica jest zapisywana w liczniku kredytu (ang. *credit counter*). W kolejnym cyklu obsługi stała kwota przydzielona dla danego przepływu jest pomniejszana o zawartość licznika kredytu.

9.7.9. DYSCYPLINY SZEREGOWANIA W SIECIACH Z JAKOŚCIĄ USŁUG

Zadania zapobiegania przeciążeniom i sterowania przepływami są formułowane w każdym sieciowym systemie teleinformatycznym.

W procedurach zapobiegania przeciążeniom oraz sterowania przepływami, które są implementowane w klasycznych sieciach teleinformatycznych obsługujących ruch bez gwarancji jakości, stosowane są przede wszystkim różne odmiany metody okna. Ich podstawowym zadaniem jest zabezpieczenie systemu końcowego (ujścia) przed przepełnieniem, powodowanym nadmierną ilością ruchu generowanego przez system początkowy (źródło). O cechach charakterystycznych takich rozwiązań decydują:

- względnie długi czas transmisji jednostki danych (np. pakietu), w porównaniu ze zwłoką czasową potwierdzenia wnoszoną przez sieć (ang. *round-trip delay*), uzasadniająca stosowanie reakcyjnych metod zapobiegania przeciążeniom, tzn. małą bezwładność sieci,

- przeznaczenie sieci do obsługi tylko jednej klasy ruchu, tzn. danych generowanych przez systemy komputerowe, dla której nie są wymagane gwarancje jakości usług,

- zorientowanie mechanizmów sterowania na optymalizację jakości obsługi ruchu i stopnia wykorzystania zasobów sieci w sensie średnim.

Ze względu na jednakowy sposób obsługi wszystkich jednostek danych generowanego ruchu naturalnym sposobem szeregowania jednostek danych w węzłach komutacyjnych sieci jest kolejkowanie jednostek danych zgodnie z dyscypliną obsługi FCFS w celu ich transmitowania w kolejności napływu do węzłów.

Stosowanie szybkich sieci z wieloma klasami ruchu o różnych wymaganiach co do jakości usług wymusza stosowanie innych metod sterowania przepływami w sieciach, w szczególności z następujących powodów:

- sieci są wykorzystywane do jednoczesnej obsługi różnych klas ruchu, wymagających różnych usług i różnej jakości usług,

- jednostki danych są nadawane i obsługiwane z dużymi szybkościami, co zwiększa bezwładność sieci, która dyskwalifikuje przydatność większości reakcyjnych metod zapobiegania przeciążeniom.

Duża szybkość nadawania i transmisji jednostek danych, przy których czas ich transmisji jest porównywalny (lub nawet mniejszy) od opóźnienia wnoszonego przez sieć, powoduje, że wystąpienie przeciążenia wiąże się ze stratami dużych ilości ruchu. Czas upływający od chwili wykrycia stanu przeciążenia do chwili powiadomienia

źródła i zdławienia szybkości nadawania jest, w porównaniu z czasem transmisji jednostek danych, na tyle duży, że ilość ruchu wprowadzona w tym czasie do sieci i tracona z powodu przeciążenia powoduje nieprzydatność metod reakcyjnych. Przydatność metod reakcyjnych maleje wraz ze wzrostem bezwładności sieci, mierzonej ilorazem czasu reakcji sieci do czasu transmisji jednostki danych w sieci.

Duża i rosnąca, wraz z szybkością transmisji, bezwładność sieci powoduje, że kolejowanie jednostek danych w węzłach sieci zgodnie z dyscypliną FCFS jest nieuzasadnione ze względu na to, że FCFS jest dyscypliną szeregowania:

- uniemożliwiająca efektywne różnicowanie przepływów, klas ruchu, usług i gwarantowanie jakości usług sieci,

- niesprawiedliwą w przypadku wystąpienia przeciążenia, tzn. jakość usług dla ruchu zgodnego z deklaracjami (kontraktami ruchowymi) (ang. *well-behaved traffic*) nie jest zabezpieczona przed skutkami wystąpienia ruchu niezgodnego z deklaracjami (kontraktami ruchowymi) (ang. *misbehaving traffic*).

Od dyscyplin szeregowania, implementowanych w węzłach sieci szerokopasmowych, wymaga się, aby cechowały je:

- prostota implementacji i realizacji (ang. *simplicity*), tzn. czas przetwarzania musi być znacznie mniejszy od średniego czasu transmisji jednostki danych,

- odporność na zmiany ilościowe ruchu (ang. *robustness*), tzn. w przypadku wystąpienia nieoczekiwanych zmian w ilości ruchu wprowadzanego do sieci i wystąpienia przeciążenia, ruch generowany przez źródła dotrzymujące kontraktów ruchowych musi być chroniony przed skutkami nadmiarowego ruchu, wprowadzanego przez źródła niedotrzymujące kontraktów ruchowych,

- sprawiedliwość w dostępie do zasobów (ang. *fairness*), tzn. zasoby (np. pojemność) niewykorzystywane przez pasywne przepływy powinny być współdzielone pomiędzy aktywne przepływy, proporcjonalnie do ich żądań zasobów i – co więcej – źródła ruchu nie mogą być „karane” za korzystanie z dodatkowych zasobów.

Dyscyplinami obsługi spełniającymi w przybliżeniu podane wymagania dla mechanizmów szeregowania są różne odmiany dyscypliny obsługi cyklicznej RR, w szczególności ze względu na ich właściwości dotyczące:

- oferowania stabilnej gwarancji szybkości transmisji dla przepływów ruchu i tym samym sprawiedliwego rozdziału zasobów,

- wbudowanej zdolności pasywnego sterowania przepływem, wynikającej ze sposobu zwielokrotniania ruchu; o ile w przypadku dyscypliny FCFS zwiększenie zajętości zasobów przez dany przepływ może nastąpić w wyniku zwiększenia wybuchowości źródła ruchu (ang. *traffic source burstiness*), o tyle w przypadku dyscypliny obsługi cyklicznej jest gwarancja, że udział poszczególnych przepływów w przepływie sumarycznym nie zależy od wybuchowości źródeł przepływów składowych.

Opracowano wiele różnych dyscyplin szeregowania jednostek danych ruchu i odpowiadających im algorytmów obsługi, których celem jest spełnienie wymagań, wynikających z potrzeby obsługi w węzłach sieci ruchu o różnych wymaganiach dotyczących ilości zasobów i jakości usług sieci.

Jedną z grup dyscyplin obsługi o wymaganych właściwościach jest grupa algorytmów, których zasada działania polega na tym, że każda jednostka danych w sieci jest znakowana czasem wspólnego wirtualnego zegara (ang. *virtual-time-based*). Jednostki danych w węzłach sieci są porządkowane zgodnie z rosnącymi wartościami znakowania: im mniejsza jest wartość znakowania jednostki danych, tym wyższy jest jej priorytet w dostępie do zasobów (np. do pojemności łącza transmisyjnego). Istnieje wiele algorytmów, opartych na koncepcji znakowania, gwarantujących: szybkość transmisji, górne ograniczenie opóźnienia, górne ograniczenie fluktuacji opóźnienia, sprawiedliwość w dostępie do zasobów itd.

Większość algorytmów sprawiedliwego kolejkowania opiera się na koncepcji idealnej sprawiedliwej dyscypliny szeregowania, jaką jest szeregowanie z uogólnionym podziałem czasu procesora GPS.

Dyscyplina szeregowania GPS jest algorytmem teoretycznym, nazywanym także dyscypliną sprawiedliwego przepływowego kolejkowania FFQ (ang. *Fluid Fair Queueing*), ze względu na to, że opiera się na założeniu, zgodnie z którym ruch jest nieskończenie podzielny i może być obsługiwany w nieskończenie małych ilościach, a wszystkie aktywne przepływy mogą być obsługiwane równolegle.

W rzeczywistych sieciach, w których jednostki danych są niepodzielne i obsługa ruchu prowadzona jest szeregowo, tzn. jednostka po jednostce, nie jest możliwe uzyskanie sprawiedliwego rozdziału zasobów wynikającego z algorytmu GPS. Wyniki działania algorytmu GPS można interpretować jako górne ograniczenie jakości szeregowania jednostek danych. Implementacja dyscypliny GPS do obsługi rzeczywistego ruchu jednostek danych wymaga takich aproksymacji tej dyscypliny obsługi, które uwzględniają dyskretny charakter strumieni jednostek danych, ich niepodzielność oraz sekwencyjny sposób obsługi.

Opracowano wiele różnych algorytmów, które emulują działanie idealnej dyscypliny sprawiedliwego szeregowania GPS. Chronologicznie pierwszą jest propozycja algorytmu ważonego, sprawiedliwego kolejkowania WFQ, nazywanego także algorytmem pakietowego, uogólnionego podziału czasu procesora PGPS. Celem algorytmu WFQ jest uzyskanie poziomu sprawiedliwości rozdziału zasobów, zbliżonego do zapewnianego przez dyscyplinę GPS, przez transmisję pakietów (jednostek danych) w kolejności zgodnej z rosnącymi czasami zakończenia ich obsługi w systemie GPS. Algorytm WFQ, zapewniający sprawiedliwy rozdział zasobów i górne ograniczenie opóźnienia wnoszonego przez algorytm w porównaniu do czasu obsługi zgodnie z GPS, wymaga symulacji systemu GPS w czasie rzeczywistym. Celem symulacji jest uzyskanie wspólnej dla sieci podstawy czasu, niezbędnej do szacowania czasów zakończenia obsługi jednostek danych i ich porządkowania. Konieczność symulacji systemu GPS i złożoność tego zadania przesądza o małej praktycznej przydatności algorytmu WFQ. W celu uproszczenia algorytmu WFQ zaproponowano wiele różnych rozwiązań, będących modyfikacjami algorytmu WFQ.

Jedną z propozycji uproszczenia algorytmu WFQ polega na zastąpieniu wirtualnego zegara sieci zegarem wewnętrznym algorytmu, tzn. do szacowania czasów zakoń-

czenia obsługi jednostek danych wykorzystywane są czasy zakończenia obsługi wcześniejszych jednostek danych w tych samym lub innych przepływach obsługiwanych przez serwer WFQ. Omawianą modyfikację algorytmu WFQ, noszącą nazwę SCFQ, charakteryzuję – w porównaniu z algorytmem PGPS – większe górne ograniczenie opóźnienia wnoszonego przez algorytm w porównaniu do czasu obsługi zgodnie z GPS, ale także mniejsza złożoność obliczeniowa.

Opracowano wiele różnych propozycji modyfikacji i uproszczeń algorytmu SCFQ, których celem jest zwiększenie efektywności układowych implementacji algorytmu w przełącznikach sieci (zwłaszcza w przełącznikach sieci ATM).

Inną propozycją aproksymacji dyscypliny szeregowania GPS jest algorytm ważnego sprawiedliwego kolejkowania dla najgorszego przypadku W2FQ. Zadaniem tego algorytmu jest minimalizacja rozbieżności pomiędzy jakością sprawiedliwego rozdziału zasobów dyscypliny GPS i wersji dyskretnej tej dyscypliny. Poprawę jakości rozdziału zasobów w algorytmie W2FQ, w porównaniu z algorytmem WFQ, uzyskuje się za cenę zwiększenia złożoności obliczeniowej.

Problemy związane z implementacją wymienionych aproksymacji dyscypliny GPS wynikają z dużej złożoności obliczeniowej i rozmiarów zadań sortowania jednostek danych. Praktyczna implementacja tych aproksymacji wymaga stosowania bardzo szybkich układów, które podnoszą koszty i ograniczają skalowalność rozwiązań.

Jednym ze sposobów redukcji złożoności rozwiązań zadania sprawiedliwego rozdziału zasobów są m.in. rozwiązania oparte na algorytmach obsługi cyklicznej, takich jak DRR. Algorytmem obsługi cyklicznej jest algorytm DRR, w którym w każdym z cykli każdemu obsługiwanemu przepływowi przypisywany jest kwant zasobów; niewykorzystanie lub przekroczenie kwantu oznacza – odpowiednio – zwiększenie lub zmniejszenie kwantu zasobów w kolejnych cyklach obsługi. Celem tego algorytmu jest implementacja zasady sprawiedliwego kolejkowania jednostek danych i uzyskania akceptowalnego poziomu złożoności obliczeniowej za cenę pogorszenia jakości obsługi mierzonej wskaźnikiem sprawiedliwości rozdziału zasobów i opóźnieniem wnoszonym przez algorytm. Ze względu na to, że rozdział zasobów przez algorytm DRR zależy nie tylko od charakterystyk ruchu, ale także od wyboru parametrów systemu obsługi cyklicznej, jakość działania algorytmu DRR jest nieporównywalna *a priori* z jakością działania algorytmów emulujących działanie dyscypliny szeregowania GPS.

Jednym z rozwiązań zwiększających, w porównaniu z algorytmem DRR, jakość obsługi przepływów oraz zmniejszających złożoność obliczeniową zadania sprawiedliwego szeregowania, w porównaniu z algorytmami emulującymi dyscyplinę GPS, jest algorytm sprawiedliwego szeregowania z gwarancją szybkości obsługi CBFQ (ang. *Credit-Based Fair Queueing*). W algorytmie tym do każdego przepływu przypisany jest licznik, którego wartość specyfikuje ilość zasobów (kredyt) dostępnych dla danego przepływu. Decyzja o kolejności transmisji jest podejmowana na podstawie decyzji wynikającej z analizy wielkości zgromadzonego kredytu, żądanej ilości zasobów oraz rozmiaru jednostki danych przepływu. Algorytm CBFQ pozwala na uzyska-

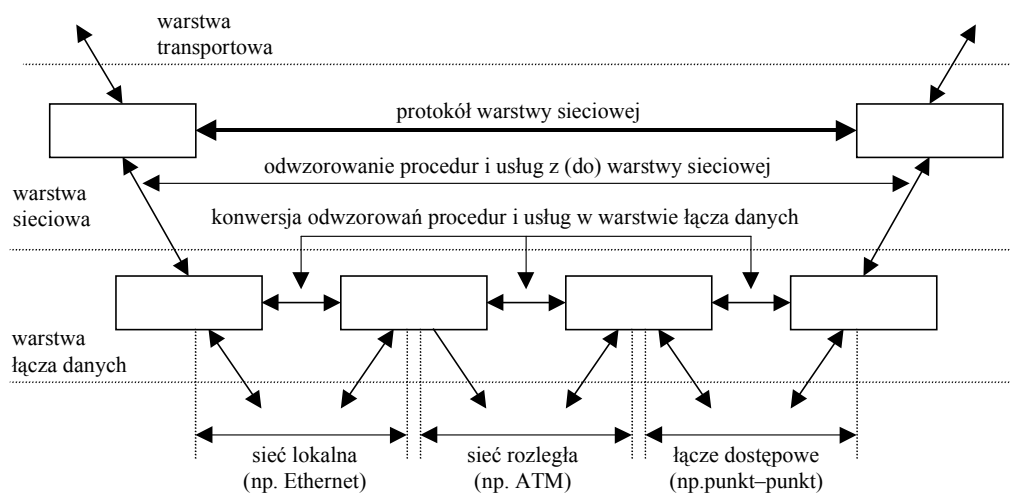
nie jakości obsługi, w sensie sprawiedliwości rozdziału zasobów i opóźnienia wnoszonego przez algorytm, porównywalnej z algorytmem SCFQ przy znacznie mniejszej złożoności obliczeniowej.

9.8. ODWZOROWANIE JAKOŚCI USŁUG W SIECIACH Z INTEGRACJĄ USŁUG

Koncepcja sieci z integracją usług jest ogólną koncepcją dostarczania gwarantowanej jakości usług; dotyczy zwłaszcza rozszerzenia zdolności świadczenia (dostarczania) jakości usług w architekturze TCP/IP, tj. protokół warstwy transportowej i sieciowej.

Ze względu na to, że w modelu warstwowym realizacja usług w warstwie transportowej i sieciowej odbywa się z wykorzystaniem usług warstwy łącza danych, konieczne są procedury odwzorowywania działania mechanizmów gwarantujących jakość usług warstwy sieciowej w mechanizmy implementowane w warstwie łącza danych.

Złożoność zadania odwzorowania usług warstwy sieciowej w usługi warstwy łącza danych wynika m.in. z istnienia wielu, różnych koncepcyjnie i technologicznie, rozwiązań warstwy łącza danych jednocześnie lub zamiennie wspierających usługi warstwy sieciowej w jednej sieci. W sieciach heterogenicznych z zadaniem odwzorowania usług i procedur warstwy sieciowej wiążą się zadania konwersji odwzorowań pomiędzy różnymi rozwiązaniami warstwy łącza danych (rysunek 9.21).



Rysunek 9.21. Ogólna koncepcja odwzorowań procedur i usług w architekturach warstwowym

W koncepcji sieci z integracją usług standardy odwzorowań procedur i usług warstwy sieciowej w procedury i usługi warstwy łącza danych dotyczą dwóch grup odwzorowań:

1. Odwzorowania usług (ang. *service mappings*) – definicje sposobów, w jakie rozwiązania warstwy łącza danych są wykorzystywane do realizacji usług dla modeli usług definiowanych w warstwie sieciowej, tzn. modelu gwarantowanej usługi i modelu sterowanego obciążenia.

2. Odwzorowanie procedury rezerwacji (ang. *setup protocol mappings*) – opisy sposobów implementacji lub odwzorowania protokołów rezerwacji zasobów warstwy sieciowej w procedury i protokoły warstwy łącza danych.

Złożoność zadania odwzorowania, realizowanego na styku warstwy sieciowej i łącza danych, zależy od sposobów rezerwacji i udostępniania zasobów w konkretnych standardach warstwy łącza danych. W przypadku łącza dostępowego typu punkt–punkt, zadanie odwzorowania jest względnie proste; wynika to z prostoty i właściwości takiego łącza. Zadanie odwzorowania jest znacznie bardziej złożone w sieciach lokalnych, wykorzystujących różne metody dostępu do zasobów (medium transmisyjnego); w szczególności w sieciach korzystających w sposób zdecentralizowany z rywalizacyjnych metod dostępu (np. Ethernet). W przypadku sieci ATM zadanie odwzorowania polega na translacji procedur i usług warstwy sieciowej w standardy jakości usług ATM.

9.9. PODSUMOWANIE

Opracowanie koncepcji usług zintegrowanych podyktowane było ograniczonymi możliwościami mechanizmów priorytetowania i znakowania ruchu, stosowanymi w sieciach bez gwarancji usług do rozwiązywania zadań zapewniania jakości usług, zwłaszcza dla ruchu wrażliwego na opóźnienia. W koncepcji usług zintegrowanych używane są mechanizmy kontroli dopuszczania, wyznaczania tras, szeregowania i nadzorowania (odrzućcia).

Obsługę ruchu w sieciach realizujących koncepcję usług zintegrowanych determinują podstawowe jej założenia dotyczące sposobów gwarantowania jakości usług, sposobów i granulacji alokowanych zasobów sieci oraz mechanizmów zarządzania alokacją zasobów i ich wykorzystaniem.

W koncepcji usług zintegrowanych formułowane są dwa modele usług: usługi gwarantowanej i usługi sterowanego obciążenia.

Realizacja usługi gwarantowanej, przeznaczonej dla aplikacji wymagających zachowania ustalonego górnego ograniczenia opóźnienia wnoszonego przez sieć, wymaga gwarantowanego dostępu do określonych zasobów sieci – dostęp do zasobów jest warunkiem koniecznym. Dostarczanie gwarantowanej jakości usług wymaga, oprócz dostępności zasobów, implementacji w sieci mechanizmów wymuszających alokację przydzielonych zasobów. Wartości miar jakości dostarczanych usług są w tym przypadku bezwzględne.

Sposoby realizacji usługi sterowanego obciążenia są zbliżone do sposobów realizacji usługi gwarantowanej w tym znaczeniu, że stosowane są te same koncepcje udostępniania i zarządzania zasobami. O różnicy omawianych modeli usług decyduje różne formułowanie miar jakości. W przypadku usług sterowanego obciążenia korzysta się ze statystycznych miar jakości, co powoduje, że usługę sterowanego obciążenia można interpretować jako lepszą od usługi bez gwarancji jakości (ang. *better-than-best effort*).

Naczelna idea koncepcji usług zintegrowanych wywodzi się z założenia, że bezwzględna gwarancja jakości usług w sieci jest możliwa tylko przez rezerwację zasobów dla strumieni jednostek danych użytkownika sieci.

Rezerwacja zasobów sieci dla poszczególnych przepływów wymaga implementacji w sieci protokołu sygnalizacyjnego niezależnego od protokołu sieciowego. Działanie protokołu rezerwacji zasobów w sieci z integracją usług dla przepływu żądającego zasobów jest zbliżone do zasad działania większości protokołów sygnalizacyjnych. Wymiana wiadomości protokołu na trasie od źródła do ujścia i z powrotem pozwala na sprawdzenie dostępności żądanych zasobów i ich rezerwację; w przypadku niepowodzenia rezerwacji żądanie dostępu do sieci nie jest przyjmowane.

Realizacja usług zintegrowanych wiąże się z implementacją w sieci czterech grup mechanizmów: rezerwacji, sterowania dostępem do sieci, klasyfikacji i szeregowania. Warunkiem obsługi aplikacji żądającej usługi jest ustanowienie trasy i zarezerwowanie zasobów na tej trasie przed rozpoczęciem transmisji. Zadaniem procedur sterowania dostępem jest podejmowanie decyzji o możliwości przyjęcia żądania obsługi do realizacji w sieci. Klasyfikacja jest procesem uruchamianym w każdym węźle trasy, a jej wyniki są wykorzystywane do kształtowania ruchu, dyskryminacji nadmiarowego ruchu, identyfikacji praw dostępu do zarezerwowanych zasobów itd. Zadaniem szeregowania jest wymuszanie dostępu do zarezerwowanych zasobów, czyli do realizacji warunku wystarczającego bezwzględnej lub względnej gwarancji jakości usług. Złożenie działań wymienionych czterech grup procedur przeciwdziała globalnym i lokalnym przeciążeniom sieci pozwala alokować wcześniej zarezerwowane zasoby sieci dla ruchu przyjętego do obsługi w sieci. Implementacja tych procedur umożliwia sterowanie jakością usług w sieci, tzn. różnicowanie jakości obsługi przepływów współdzielących zasoby sieci.

Tym, co w zasadniczy sposób wpływa na właściwości usług zintegrowanych jest poziom granulacji różnicowania jakości usług; przedmiotem sterowania jest jakość usług dostarczanych elementarnym przepływom w sieci. Taki poziom granulacji sterowania przepływami, jakkolwiek pozwala na dużą granulację alokowanych zasobów i tym samym efektywne wykorzystanie zasobów sieci, wymaga jednak znacznego rozbudowania systemu monitorowania i zarządzania ruchem w sieci, niwelującego korzyści wynikające z dużej granulacji obsługiwanego ruchu.

Zorientowanie koncepcji usług zintegrowanych na elementarne przepływy w sieci jest jej podstawowym ograniczeniem. Skalowalność takiej sieci jest ograniczona przyrostem ilości danych niezbędnych do zarządzania siecią, który jest proporcjonalny do liczby elementarnych przepływów w sieci.

LITERATURA

- [1] BENSOU B., TSANG D.H.K., CHAN K.T., *Credit-based Fair Queueing (CBFQ): A simple service-scheduling algorithm for packet-switched networks*, IEEE/ACM Transactions on Networking, Vol. 9, No. 5, October 2001, s. 591–604.
- [2] BŁAŻEWICZ J., CELLARY W., SŁOWIŃSKI R., WĘGLARZ J., *Badania operacyjne dla informatyków*, WNT, Warszawa 1983.
- [3] BURAKOWSKI W., BAŁ A., KOPERTOWSKI Z., WYKROTA A., *Mechanizmy QoS w sieci IP*, II ogólnopolska konferencja „Internet”, Wrocław 2000, s. 25–37.
- [4] CHAO J., *SATURN: a terabit packet switch using Dual Round-Robin*, IEEE Communications Magazine, December 2000, s. 78–84.
- [5] CHAO H.J., GUO X., *Quality of Service control in High-Speed Networks*, Wiley, 2002.
- [6] CHEUNG S.Y., PENCEA S., *Pipelined sections: a new buffer management discipline for scalable QoS provision*, Emory University (www.itpapers.com).
- [7] COURTOIS P.J., *The M/G/1 finite capacity queue with delays*, IEEE Transactions on Communications, Vol. COM-28, February 1982, s. 165–172.
- [8] DECASPER D., DITTIA Z., PARULKAR G., PLATTNER B., *Router plugins: a software architecture for next generation routers*, IEEE/ACM Transactions on Networking, Vol. 8, No. 1, February 2000, s. 2–16.
- [9] FELDMAN A., REXFORD J., CACERES R., *Efficient policies for carrying web traffic over flow-switched networks*, IEEE/ACM Transactions on Networking, Vol. 6, No. 6, December 1998, s. 673–685.
- [10] FENG Q.C., KANDLUR D.D., SAHA D., SHIN K.G., *Adaptive packet marking for maintaining end-to-end throughput in a Differentiated Services Internet*, IEEE/ACM Transactions on Networking, Vol. 7, No. 5, June 1993, s. 685–697.
- [12] GEVROS P., CROWCROFT J., KIRSTEIN P., BHATTI S., *Congestion control mechanisms and the best effort service model*, IEEE Network, May/June 2001, s. 16–26.
- [13] GUERIN R.A., ORDA A., WILLIAMS D., *QoS routing mechanisms and OSPF extensions*, IEEE Global Internet Mini-Conference (Globecom'97), November 1997.
- [14] KESHAV S., *On the efficient implementation of Fair Queueing*, Internetworking: Research and Experiences, Vol. 2, 1991, s. 157–173.
- [15] KUMMAR V.P., LAKSHMAN T.V., STILIADIS D., *Beyond Best Effort: router architectures for the differentiated services of tomorrow's Internet*, IEEE Communications Magazine, May 1988, s. 152–164.
- [16] LORENZ D.H., ORDA A., *QoS routing in networks with uncertain parameters*, IEEE/ACM Transactions on Networking, Vol. 6, No. 6, December 1998, s. 768–778.
- [17] LORENZ D.H., *QoS routing in networks with uncertain parameters*, University of Haifa 1998.
- [18] MEER H., PULIAFITO A., RICHTER J.P., TOMARCHIO O., *QoS-adaptation by software agents in the presence of defective reservation mechanisms in the Internet*, University of Hamburg, 2000.
- [19] MINKENBERG C., ENGBERSON D., *A combined input and output queued packet-switched system based on PRIZMA switch-on-a-chip technology*, IEEE Communications Magazine, December 2000, s. 70–77.

- [20] NONG G., HAMDI M., *On the provision of Quality-of-Service guarantees for input queued switches*, IEEE Communications Magazine, December 2000, s. 62–69.
- [21] PAREKH A.K., GALLAGER R.G., *A Generalized Processor Sharing approach to flow control in Integrated Services networks: the single node case*, IEEE/ACM Transactions on Networking, Vol. 1, No. 3, June 1993, s. 344–357.
- [22] PEREKH A.K., GALLAGER R.G., *A Generalized Processor Sharing approach to flow control in integrated services networks: the multiple node case*, IEEE/ACM Transactions on Networking, Vol. 2, No. 2, April 1994, s. 137–150.
- [23] RAJAN R., VERMA D., KAMAT S., FELSTAIN E., HERZOG S., *A policy framework for Integrated and Differentiated Services in the Internet*, IEEE Network, September/October 1999, s. 36–41.
- [24] RUIZ-SANCHEZ M., ANTIPOLIS S., BIRSACK E.W., DABBOUS W., *Survey and taxonomy of IP address lookup algorithms*, IEEE Network, March/April 2001, s. 8–23.
- [25] SCHMIT J., KARSTEN M., WOLF L., STEINMETZ R., *Aggregation of guaranteed service flows*, Darmstadt University of Technology, 2001 (www.itpapers.com).
- [26] SIVARAMAN V., CHIUSI F.M., GERLA M., *End-to-end statistical delay service under GPS and EDF scheduling: A comparison study* (www.itpapers.com).
- [27] STOICA I., ZHANG H., NG T.S.E., *A hierarchical fair service curve algorithm for link-sharing, real-time, and priority services*, IEEE/ACM Transactions on Networking, Vol. 8, No. 2, April 2000, s. 185–211.
- [28] WAICHEE A.F., *Delay-optimal quorum consensus for distributed systems*, IEEE Transactions on Parallel and Distributed Systems, Vol. 8, No. 1, January 1997, s. 59–69.
- [29] WARKHEDE P., SURI S., VARGHESE G., *Fast packet classification for two-dimensional conflict-free filters*, Cisco Systems, 2001.
- [30] WANG Z., *Internet QoS: Architectures and mechanisms for Quality of Service*, Academic Press, London 2001.
- [31] XIAO X., NI L.M., *Internet QoS: A big picture*, IEEE Network, March/April 1999, s. 8–18,
- [32] ZHANG L., DEERING S., ESTRIN D., SHENKER S., ZAPPALA D., *RVSP: A new resource ReSerVation Protocol*, IEEE Network, September 1993, s. 8–18.
- [33] ZHU K., ZHUANG Y., VINIOTIS Y., *Achieving end-to-end delay bounds by EDF scheduling without traffic shaping*, North Carolina State University, Alcatel, Amber 2001 (www.itpapers.com).

10. KONCEPCJA USŁUG ZRÓŻNICOWANYCH

Podstawę koncepcji usług zróżnicowanych stanowi założenie, że przedmiotem sterowania jest ograniczona, w porównaniu z liczbą wszystkich przepływów elementarnych, liczba klas ruchu w sieci. Sterowanie, polegające na ograniczaniu ilości wprowadzanego ruchu, eliminuje potrzebę stosowania procedur rezerwacji zasobów sieci. Ograniczona i stała lub wolnozmienna liczba klas ruchu uniezależnia rozmiar zadania sterowania ruchem w sieci od liczby aktualnie obsługiwanych przepływów elementarnych. Architektura usług zróżnicowanych opiera się na prostym modelu, w którym ruch wprowadzany do sieci jest odpowiednio klasyfikowany i następnie obsługiwany według określonego kontraktu ruchowego w każdym z węzłów sieci. Niezależne strumienie ruchu obsługiwane w tej samej klasie są agregowane i otrzymują ten sam poziom jakości usług.

10.1. PODSTAWOWE ZAŁOŻENIA KONCEPCJI USŁUG ZRÓŻNICOWANYCH

Koncepcja usług zróżnicowanych jest odpowiedzią na oczekiwania dotyczące względnie prostych, wielopoziomowych i zgrubnych metod udostępniania zasobów dla ruchu generowanego przez aplikacje o różnych wymaganiach jakościowych, zwłaszcza przez aplikacje internetowe. Koncepcja usług zróżnicowanych jest rozwiązaniem uzupełniającym ograniczenia koncepcji usług zintegrowanych.

Podejście zastosowane w koncepcji usług zróżnicowanych opiera się na założeniu, że różnej jakości usługi w sieci mogą być dostarczane za pomocą niewielkiej liczby dobrze zdefiniowanych modułów funkcjonalnych, z których możliwe jest budowanie różnorodnych, złożonych i zagregowanych funkcjonalności odpowiadających potrzebom użytkowników.

Sieci oferujące usługi zróżnicowane są zorientowane na obsługę i dostarczanie jakości usług zagregowanym strumieniom ruchu (ang. *aggregate traffic handling*). W koncepcji usług zróżnicowanych całkowity ruch generowany przez użytkowników sieci jest dzielony na niewielką liczbę klas ruchu, określanych także mianem klas spedycji (ang. *forwarding class*) lub agregatów strumieni (ang. *behaviour aggregate*).

Przynależność jednostki danych do klasy spedycji jest odpowiednio kodowana w nagłówku pakietu. Podczas klasyfikacji pakietu w jego nagłówku zapisywany jest unikatowy kod klasy ruchu. Każda z wyróżnionych klas ruchu charakteryzuje się predefiniowanym sposobem obsługi w sieci, zwłaszcza priorytetem odrzucania jednostek danych i sposobem alokacji zasobów (pojemności).

Warunkami efektywnej implementacji koncepcji usług zróżnicowanych jest rozwiązywanie, w czasie obsługi jednostek danych należących do wyróżnionych klas ruchu, następujących zadań:

- klasyfikacji jednostek danych – kwalifikowanie jednostek danych do agregatów ruchu oznacza konieczność identyfikacji każdej jednostki danych (zadanie klasyfikacji),
- wymuszania ograniczeń na ilość ruchu w sieci dla każdej z klas – jakość usług dla każdej z klas ruchu zależy od ilości ruchu danej klasy w sieci oraz od całkowitej ilości ruchu w sieci (zadania monitorowania, kształtowania i odrzucania ruchu),
- różnicowania sposobów obsługi klas różnych klas ruchu w różnych domenach administracyjnych sieci – zadania klasyfikacji i nadzorowania ruchu różnią się w zależności od miejsca w sieci, w którym są realizowane.

W zadaniach zwielokrotniania ruchu rozróżnia się pojęcia agregacji ruchu (ang. *flows aggregation*) i grupowania ruchu (ang. *flows grouping*). Terminem agregacja określa się ogólne zadanie koncentracji różnych przepływów w wyróżnionym obszarze sieci, terminem grupowanie natomiast określa się zadanie, w którym obszarem koncentracji ruchu jest cała sieć, tzn. agregowane są przepływy pomiędzy systemami końcowymi. Przyjęcie takiego podziału oznacza, że grupowanie jest specjalnym przypadkiem agregacji.

Zorientowanie koncepcji usług zróżnicowanych na klasy ruchu jest podstawowym źródłem różnic pomiędzy tą koncepcją a koncepcją usług zintegrowanych, która jest przeznaczona do obsługi ruchu na niższym poziomie granulacji, tzn. na obsługę i dostarczanie jakości usług przepływowi w sieci (ang. *per-flow traffic handling*).

Orientacja na zagregowane strumienie ruchu oznacza, że przedmiotem procedur wyznaczania tras, zapobiegania przeciążeniom i sterowania przepływem w sieci są klasy ruchu. Ograniczona liczba klas ruchu, znacznie mniejsza i bardziej stabilna od liczby wszystkich przepływów w sieci, rozwiązuje problem skalowalności istotnie ograniczający implementację koncepcji usług zintegrowanych.

Rozwiązania przyjęte w koncepcji usług zróżnicowanych są pośrednie pomiędzy koncepcją usług udostępnianych bez gwarancji jakości (zgodnie z zasadą najwyższej staranności) oraz koncepcją usług zintegrowanych. Dostarczanie usług w modelu bez gwarancji jakości usług opiera się na współzawodnictwie przepływów o współdzielone zasoby sieci po przyjęciu przepływów do obsługi, koncepcja usług zintegrowanych natomiast jest oparta na rezerwacji i reglamentacji zasobów przed przyjęciem ruchu do obsługi. Współzawodnictwo o zasoby praktycznie wyklucza możliwość różnicowania jakości usług, maksymalizując stopień wykorzystania zasobów, podczas gdy rezerwowanie zasobów gwarantuje jakość usług, zmniejszając efektywność wykorzystania zasobów. Koncepcja usług zróżnicowanych jest rozwiązaniem pośrednim, w tym sensie, że proponowane rozwiązania są kompromisem pomiędzy pożądaną (przez użyt-

kownika sieci) jakością usług i oczekiwaną (przez operatora sieci) efektywnością wykorzystania zasobów. Jakość usług dostarczanych w sieci z implementacją koncepcji usług zróżnicowanych jest określana mianem: lepsza niż w modelu bez gwarancji jakości (ang. *better-than-best-effort*).

Jednym ze źródeł efektywności tej koncepcji jest także założenie, że sterowanie przepływami klas ruchu odbywa się na brzegu sieci. Z punktu widzenia realizacji koncepcji usług zróżnicowanych istotny jest podział węzłów sieci na węzły brzegowe (ang. *boundary, edge nodes*) i wewnętrzne (szkieletowe) (ang. *interior, core nodes*). Zakres ich odpowiedzialności i kompetencji w realizacji omawianej koncepcji realizacji jakości usług jest różny; większość zadań istotnych dla realizacji koncepcji usług zróżnicowanych jest lokalizowana na brzegu sieci.

Obsługa ruchu napływającego do węzła brzegowego sieci obejmuje realizację dwóch grup zadań: klasyfikację jednostek danych (ang. *classification*) i kondycjonowanie ruchu (ang. *traffic conditioning*), obejmujących m.in. przypisywanie pakietów do klas ruchu, sprawdzanie zgodności charakterystyk ruchu z warunkami kontraktów ruchowych na świadczenia usług (nadzorowanie) i dławienie przepływu pakietów nadmiarowych.

Rola węzłów wewnętrznych ogranicza się do przekazywania ruchu według zasad przewidzianych dla poszczególnych klas pakietów.

10.2. DEFINIOWANIE USŁUG W SIECI USŁUG ZRÓŻNICOWANYCH

Dwa zasadnicze i różne pojęcia – podobne, ale nie tożsame – w koncepcji usług zróżnicowanych, dotyczące definiowania, realizacji i weryfikacji jakości usług w sieciach, to:

- usługa sieci (ang. *network service*),
- sposób spedycji ruchu w sieci (ang. *forwarding treatment*).

Do opisu usług sieci, na etapie ich definiowania lub weryfikacji przez użytkownika, stosowane są wartości obiektywnych, mierzalnych wskaźników jakości działania sieci, takich jak: opóźnienie, zakres zmienności opóźnienia, prawdopodobieństwo strat. Określony sposób spedycji ruchu w sieci nie jest tożsamy z określoną jakością usługi sieci; definicja jakości usług w określonym sposobie spedycji ruchu ma charakter jakościowy.

W koncepcji usług zróżnicowanych usługa sieci polega na złożeniu:

- wybranego sposobu spedycji klasy ruchu, oraz
- mechanizmu reglamentacji ruchu dystrybuowanego z wykorzystaniem tego sposobu spedycji,

tzn. żadaną lub deklarowaną jakością usługi dostarczaną użytkownikowi sieci uzyskuje się przez sterowanie ilością ruchu przyjmowanego do sieci i obsługiwanego wybranym sposobem obsługi ruchu.

Zgodnie z tym jakość usług zróżnicowanych jest jakościowo definiowana przez wybrany sposób spedykcji ruchu, a utrzymanie zadanych ilościowych charakterystyk jakości usług jest zadaniem różnych procedur zarządzania ruchem (ang. *traffic engineering*) i (lub) procedur wyznaczania tras z ograniczeniami (ang. *constrained-based routing*).

Z przyjętego w koncepcji usług zróżnicowanych sposobu definiowania usług wynika, że:

- dany sposób spedykcji ruchu w sieci jest warunkiem koniecznym, ale nie wystarczającym do uzyskania określonej jakości usługi,
- dany sposób spedykcji może być wykorzystany do konstrukcji usługi i definiowania warunków, po spełnieniu których usługa ma określoną jakość,
- dany sposób spedykcji ruchu w sieci może być wykorzystany do budowy różnych usług i gwarantowania różnej ich jakości,
- dana usługa może być konstruowana za pomocą różnych sposobów spedykcji ruchu oraz różnymi mechanizmami sterowania ilością ruchu w sieci.

Przykładem omawianych zależności jest związek pomiędzy transmisją bez strat w sieci (usługa) a transmisją z priorytetem (sposób spedykcji ruchu). Realizacja omawianej usługi, tj. transmisji bez strat w sieci, dla danej klasy ruchu oznacza, że:

- zaliczany do danej klasy ruch jest odpowiednio znakowany i obsługiwany zgodnie z przyznanym priorytetem (być może najwyższym),
- ilość obsługiwanego ruchu z przyznanym priorytetem w sieci jest ograniczona, co oznacza, że zarówno ilość tego ruchu w sieci, jak i ilość zasobów udostępnianych dla obsługi tego ruchu są przedmiotem sterowania.

W ten sposób transmisja ruchu bez strat jest wynikiem złożenia sposobu spedykcji ruchu (transmisja z priorytetem) i reglamentacji ilości ruchu korzystającego z tego sposobu spedykcji (sterowanie dostępem do sieci).

Innym przykładem konstrukcji usługi, powstającej przez złożenie sposobu spedykcji ruchu i ograniczania ilości ruchu wprowadzanego do sieci, jest usługa polegająca na gwarancji górnego ograniczenia opóźnienia. W tym przypadku usługa polegająca na dostępie do ustalonej, stałej pojemności w sieci może być złożeniem sposobu spedykcji ruchu (priorytetowanie klas ruchu) i reglamentacji liczby obsługiwanych przepływów w ramach poszczególnych klas ruchu (sterowanie dostępem do sieci).

Przyjęcie założenia, że usługa i jej jakość powstają przez złożenie sposobu spedykcji ruchu i sterowanie dostępem oraz ich parametryzacją powoduje, że ograniczona liczba sposobów spedykcji ruchu i mechanizmów sterowania w sieci pozwala na budowę względnie różnych usług i względnie duże różnicowanie ich jakości. Implementacja takiej koncepcji udostępniania zasobów powoduje, że dla sieci z koncepcją usług zróżnicowanych charakterystyczne są:

- modularność koncepcji budowania usług,
- duża zmienność usług przy wykorzystaniu małej liczby stabilnych sposobów spedykcji ruchu oraz mechanizmów sterowania przeciążeniami i przepływami,
- uniwersalność sposobu kreowania i zmian jakości usług,

➤ skalowalność usług sieci w dużym zakresie zmienności generowanego ruchu i udostępnianych zasobów sieci.

10.3. OGÓLNE ZASADY SPEDYCJI RUCHU

Różne sposoby spedycji ruchu mogą być definiowane w postaci dogodnej do implementacji wybranego mechanizmu sterowania lub w postaci ogólnej, niezależnej od implementacji jakiegokolwiek mechanizmu. Niezależny od implementacji ogólny opis sposobu spedycji ruchu jest uniwersalny, gdyż ten sam oczekiwany sposób spedycji ruchu można uzyskać w wyniku implementacji różnych mechanizmów.

Klasyfikacja strumieni generowanych pomiędzy systemami końcowymi, opisanych odpowiednimi punktami kodowymi DSCP (ang. *Differentiated Service Code Point*), do klas ruchu BA (ang. *Behaviour Aggregate*) jest realizowana na brzegu sieci, tzn. w routerach brzegowych (ang. *edge router*). Sposób spedycji wszystkich pakietów, posiadających ten sam kod DSCP, tzn. należących do jednego agregatu BA, przez routery szkieletowe (ang. *core router*) jest taki sam. Sposoby spedycji klasy ruchu BA w pojedynczych węzłach wewnętrznych sieci, tzn. w routerach szkieletowych, są określane mianem reguły przesyłania PHB (ang. *Per-Hop Behaviour*). Sposoby (zasady, klasy, reguły) spedycji ruchu obejmują różne procedury obsługi zagregowanych strumieni pakietów (agregatów) polegające m.in. na ich kolejkowaniu, priorytetowaniu, szeregowaniu, odrzucaniu.

Różne reguły przesyłania ruchu w węzłach wewnętrznych sieci, tj. różne PHB, są podstawowymi modułami stosowanymi do alokacji zasobów dla różnych agregatów BA. Podział zasobów pomiędzy współzawodniczące agregaty ruchu BA jest wynikiem jednoczesnego działania różnych sposobów spedycji. Usługi typu źródło–ujście (ang. *end-to-end*) dostarczane użytkownikom sieci są budowane przez złożenie różnych reguł przesyłania ruchu PHB oraz odpowiednich mechanizmów: klasyfikowania ruchu (ang. *traffic classification*), kontrolowania ruchu (ang. *traffic policing*), kondycjonowania ruchu (ang. *traffic conditioning*), kształtowania ruchu (ang. *traffic shaping*) i udostępniania zasobów (ang. *resource provisioning*). W pewnym uproszczeniu reguły przesyłania ruchu można utożsamiać z klasami ruchu.

Lokalne (węzłowe) wersje różnych reguł przesyłania ruchu, tj. różne lokalne reguły przesyłania PHB, można charakteryzować bezpośrednio (ang. *absolute term*) lub pośrednio (ang. *relative term*). W przypadku przydzielania różnym agregatom ruchu BA zasobów (np. pojemności) w danym węźle sieci, zasoby mogą być charakteryzowane albo wartościami bezwzględnymi, tzn. bezpośrednio, albo wartościami względnymi, opisującymi udział w dostępnej, całkowitej ilości zasobów, tzn. pośrednio.

Zbiór reguł przesyłania ruchu PHB może tworzyć grupę PHB (ang. *PHB group*). Tworzenie grupy PHB jest uzasadnione wtedy, gdy zbiór lokalnych wersji reguł przesyłania (sposobów spedycji) ruchu PHB jest implementowany jednocześnie jako grupa korzystająca z współdzielonych zasobów w ramach jednego wspólnego ogranicze-

nia. Pojedynczą regułę przesyłania ruchu PHB można interpretować jako szczególny przypadek grupy reguł PHB. Grupa PHB jest opisem alokacji zasobów dla poszczególnych PHB w grupie. Alokacja może być opisana pośrednio i dotyczyć np. alokacji pojemności, priorytetu odrzucania pakietów itp. Przykładem grupy PHB jest zbiór priorytetów odrzucania dla każdego PHB w grupie.

Z opisanej funkcjonalności reguł przesyłania ruchu PHB wynika, że różne reguły PHB charakteryzują różne sposoby buforowania i szeregowania pakietów.

10.4. USŁUGI SIECI Z RÓŻNICOWANIEM USŁUG

Terminem usługi określa się różne sposoby obsługi ruchu użytkownika w domenie sieci lub w połączeniach pomiędzy systemami końcowymi (parami źródło–ujście). Usługa jest widocznym dla użytkownika sposobem obsługi jego ruchu w sieci, reguła przesyłania ruchu PHB natomiast jest ukryta w węzłach sieci.

Kreowanie usługi jest procesem składania różnych współpracujących ze sobą komponentów, realizujących m.in. zadania: kondycjonowania ruchu w węzłach brzegowych sieci, klasyfikacji ruchu do poszczególnych lokalnych wersji sposobów spedykcji ruchu, udostępniania zasobów sieci i spedykcji ruchu wewnątrz sieci.

W koncepcji usług zróżnicowanych, zapewniana jakość usług jest definiowana w postaci kontraktów usługowych SLA (ang. *Service Level Agreement*), zawieranych pomiędzy użytkownikiem sieci a dostawcą usług sieciowych. W kontrakcie SLA specyfikowane są szczegóły usługi żądanej przez użytkownika i dostarczanej użytkownikowi, w tym: liczba klas ruchu obsługiwanych w domenie, poziom jakości usług klas ruchu obsługiwanych w domenie, zakresy wartości parametrów ruchowych itd.

Jednym z istotniejszych, ze względu na jakość usługi, elementów kontraktu SLA jest specyfikacja wartości parametrów ruchowych, tzn. kontrakt ruchowy TCA (ang. *Traffic Conditioning Agreement*). Kontrakt ruchowy TCA specyfikuje parametry i wartości parametrów profili ruchu i sposobów nadzorowania zgodności rzeczywistego ruchu z profilem ruchu deklarowanym w kontrakcie: kontrakt TCA dotyczy przede wszystkim:

- profili ruchu, definiowanych wartościami parametrów algorytmów realizowanych przez sterowniki przepływów (np. szybkości napływu znaczników i pojemności bufora znaczników algorytmu ciekącego wiadra),
- miar jakości działania (np. przepustowość, opóźnienie, priorytety odrzucania),
- sposobów obsługi pakietów generowanych z naruszeniem warunków kontraktu (np. obsługa bez gwarancji jakości),
- dodatkowych usług dostarczanych przez dostawcę usług, np. dotyczących znakowania i kształtowania ruchu.

Kontrakt usługowy SLA, oprócz kontraktu ruchowego TCA, może zawierać także inne charakterystyki usługi oraz sposobów i warunków jej dostarczania, takich jak np. dostępność i niezawodność usługi, sposoby monitorowania i audytu ruchu, system rozliczeń.

Kontrakty usługowe SLA, których warunki są przedmiotem negocjacji pomiędzy użytkownikiem sieci i dostawcą usług przed rozpoczęciem obsługi ruchu, mogą być kontraktami statycznymi lub dynamicznymi. Na ogół kontrakty SLA są kontraktami statycznymi. Obsługa kontraktów dynamicznych wymaga implementacji systemów zarządzania automatyzujących obsługę zgłaszanych żądań zmian warunków kontraktów i renegocjacji kontraktów.

Różne usługi, definiowane, realizowane i dostarczane w sieciach z koncepcją usług zróżnicowanych, różnią się:

1. Opisem wartości parametrów usług:

➤ ilościowym (ang. *quantitative*) – wartości parametrów usług są podawane w wartościach bezwzględnych (np. wartość maksymalnego opóźnienia pakietu),

➤ jakościowym (ang. *qualitative*) – parametry usług są charakteryzowane w formie opisowej (np. mniejsze opóźnienie pakietu).

2. Lokalizacją źródeł i ujść:

➤ źródło i ujście w jednej domenie usług – kontrakt SLA negocjowany pomiędzy użytkownikiem i dostawcą usług,

➤ źródło i ujście w różnych domenach – kontrakt SLA negocjowany pomiędzy jednym użytkownikiem i dostawcą usług w domenie użytkownika–źródła oraz dostawcami usług w domenach: użytkownika–źródła, użytkownika–ujścia oraz pośrednich.

3. Sposobem dystrybucji ruchu:

➤ dla ruchu z danego węzła źródłowego do danego węzła ujściowego (tryb punkt–punkt),

➤ dla ruchu z danego węzła źródłowego do podzbioru zbioru wszystkich węzłów ujściowych (tryb punkt–wielopunkt),

➤ dla ruchu z danego węzła źródłowego do wszystkich węzłów ujściowych (tryb rozgłoszeniowy).

4. Miejscem implementacji procedur sterowania przepływem:

➤ ruch ze źródła (generowany przez użytkownika) do sieci – wymuszanie przestrzegania warunków kontraktu SLA jest realizowane w węźle brzegowym sieci, do którego dołączony jest użytkownik generujący ruch,

➤ ruch z sieci do ujścia (odbierany przez użytkownika sieci) – wymuszanie przestrzegania warunków kontraktu SLA jest realizowane dla wyróżnionego ruchu we wszystkich węzłach pośrednich na trasie od węzła źródłowego do ujściowego.

10.5. KLASYFIKACJA I KONDYCJONOWANIE RUCHU

Realizacja koncepcji usług zróżnicowanych oznacza m.in., że w węzłach brzegowych sieci wykonywane są zadania:

➤ klasyfikacji ruchu, polegające na przypisywaniu nadchodzących pakietów do klas ruchu (klas spedykcji ruchu) obsługiwanych przez sieć,

➤ kondycjonowania ruchu, którego celem jest wymuszanie zgodności parametrów generowanego ruchu z parametrami specyfikowanymi w kontraktach SLA dla każdego użytkownika.

Jakość realizacji zadań klasyfikacji i kondycjonowania ruchu ma podstawowe znaczenie dla jakości usług sieci, ze względu na to, że alokacja zasobów sieci odbywa się tylko na podstawie ilości ruchu poszczególnych klas ruchu wprowadzonych do sieci. Nieprawidłowa klasyfikacja lub nieprzestrzeganie warunków kontraktu oznacza marnotrawienie zasobów sieci oraz degradację jakości usług jednej lub wszystkich klas ruchu.

10.5.1. KLASYFIKACJA RUCHU

Zadanie klasyfikacji polega na kategoryzacji jednostek danych (pakietów) w przepływy. Przepływy w routerze są specyfikowane przez reguły przetwarzania napływających pakietów. Przetwarzanie wszystkich pakietów należących do tego samego przepływu odbywa się w routerze według tych samych zasad.

Klasyfikacja jest wymagana do realizacji praktycznie każdej funkcji związanej ze sterowaniem ruchem. Znaczenie klasyfikacji pakietów rośnie wraz ze wzrostem funkcjonalności routerów, zwłaszcza w sieciach z różnicowaniem jakości usług. Klasyfikacja pakietów jest wymagana m.in. do sterowania dostępem do sieci, kontroli zgodności przepływów z kontraktami ruchowymi, kierowania ruchem, kształtowania ruchu, taryfikacji ruchu i usług sieci.

Zadania klasyfikacji ruchu w sieciach są różnie formułowane, w zależności od tego, na podstawie jakich informacji odbywa się klasyfikacja i co jest oczekiwanym jej wynikiem.

Wyróżniane są następujące zadania szczegółowe:

➤ klasyfikacja (ang. *classification*) – na podstawie zawartości różnych wybranych pól nagłówka jednostek danych podejmowana jest decyzja o zaliczeniu jednostki danych do jednej z wielu klas ruchu; tym samym do jednego z wielu możliwych sposobów obsługi,

➤ identyfikacji (ang. *identification*) – na podstawie zawartości różnych wybranych pól nagłówka jednostek danych podejmowana jest decyzja o przynależności, lub nie, danej jednostki danych do ustalonych klas ruchu i (lub) klas obsługi ruchu,

➤ wyznaczania węzła trasy – na podstawie zawartości pola adresu systemu końcowego (ujścia), polegającego na sprawdzaniu zawartości odpowiedniej tablicy adresów (ang. *table lookup*), wyznaczany jest adres kolejnego węzła na trasie.

Wymienione zadania są formułowane podobnie, ale do ich rozwiązywania w czasie rzeczywistym, ze względu na ich złożoność obliczeniową i pamięciową, stosuje się różne algorytmy.

Zadaniem klasyfikacji ruchu, realizowanym przez klasyfikatory ruchu (ang. *traffic classifier*), jest przypisywanie napływających pakietów, na podstawie wcześniej zdefiniowanych zasad, do jednej z wielu możliwych klas ruchu, tzn. agregatów ruchu.

Przypisanie pakietów do jednej z klas ruchu wiąże się z odpowiednim ich znakowaniem (cechowaniem).

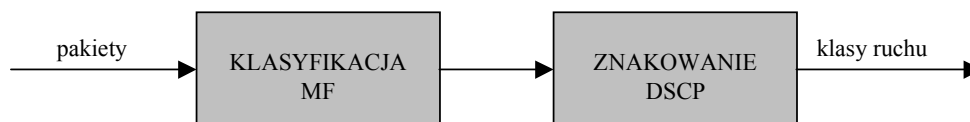
Złożoność zadania klasyfikacji pakietów zależy od tego, czy dotyczy ono klasyfikacji pakietów generowanych przez jedno czy wiele źródeł. W drugim z wymienionych przypadków pakiety są klasyfikowane po wcześniejszym zidentyfikowaniu źródła pakietu i wyborze (ustaleniu) odpowiednich, dla zidentyfikowanego źródła, warunków kondycjonowania ruchu TCA.

Moduł klasyfikacji pakietów można interpretować jako zbiór reguł i zbiór działań (akcji) odpowiadających tym regułom. Zbiór reguł to najczęściej różne możliwe kombinacje zawartości pól nagłówka pakietu; zbiór działań natomiast to różne sposoby obsługi pakietów spełniających warunki reguł, takie jak np. znakowanie kodami DSCP, wybór odpowiednich sposobów kondycjonowania czy alokacja odpowiednich zasobów dla klas spedycji ruchu.

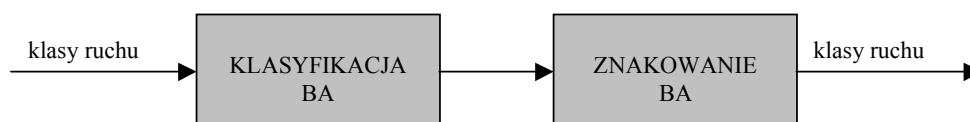
Istnieją różne sposoby klasyfikowania pakietów. W koncepcji usług zróżnicowanych zadanie klasyfikacji pakietów jest realizowane przez (rysunek 10.1):

- klasyfikatory (wielopolowe) pól nagłówków pakietów MF (ang. *Multifield Classifier*), tzn. pakiety są klasyfikowane ze względu na zawartość wielu wybranych pól nagłówka pakietów,

- klasyfikatory agregatów strumieni BAC (ang. *Behaviour Aggregate Classifier*), tzn. pakiety są klasyfikowane ze względu na przynależność do określonego strumienia zagregowanego.



a) klasyfikacja i znakowanie w routerze brzegowym



b) klasyfikacja i znakowanie w routerze szkieletowym

Rysunek 10.1. Klasyfikowanie i znakowanie pakietów w routerach

Do klasyfikacji pakietów na podstawie zawartości pól nagłówka pakietu MF stosuje się różne kombinacje zawartości jednego lub więcej pól nagłówka, takich jak np. adres źródła, adres ujścia, port źródła, port ujścia, identyfikator pakietu. Różne kombinacje pól i ich zawartości pozwalają m.in. na wyróżnianie i odpowiednie znakowanie generowanych pakietów identyfikatorami np. różnych typów aplikacji, źródeł, ujść, sieci, podsieci, przepływów. Klasyfikacja pakietów na podstawie zawartości pól

nagłówka jest, ze względu na dużą liczbę możliwych kombinacji, zadaniem względnie złożonym, ale za to daje dużą możliwość kreowania różnych klas ruchu oraz formułowania różnych sposobów alokacji zasobów.

Klasyfikacja ze względu na zawartość pól nagłówka jest w koncepcji usług zróżnicowanych tożsama z rozdziałem pakietów do różnych klas ruchu. Klasyfikacja ta jest realizowana w węzłach brzegowych (routerach brzegowych) sieci.

W ogólnym przypadku wynikiem klasyfikacji pakietu, na podstawie zawartości pól nagłówka pakietu, jest zakwalifikowanie pakietu do jednej z klas ruchu, tzn. wyznaczenie punktu kodowego DSCP. Wykonywane są jednocześnie zadania klasyfikacji i znakowania pakietu.

Klasyfikacja ze względu na przynależność do strumienia zagregowanego BA jest, w porównaniu z zadaniem klasyfikacji ze względu na zawartość pól nagłówka, zadaniem względnie prostym. Klasyfikacja polega na wyznaczeniu wartości kodu DSCP, tzn. przynależności pakietu do jednej z klas ruchu. W danym węźle wewnętrznym sieci (routerze szkieletowym) reguła przesyłania klasy ruchu BA, tzn. wszystkich pakietów mających ten sam kod DSCP, jest taka sama. Liczba sposobów spedycji realizowanych w danym węźle jest równa liczbie różnych kodów DSCP.

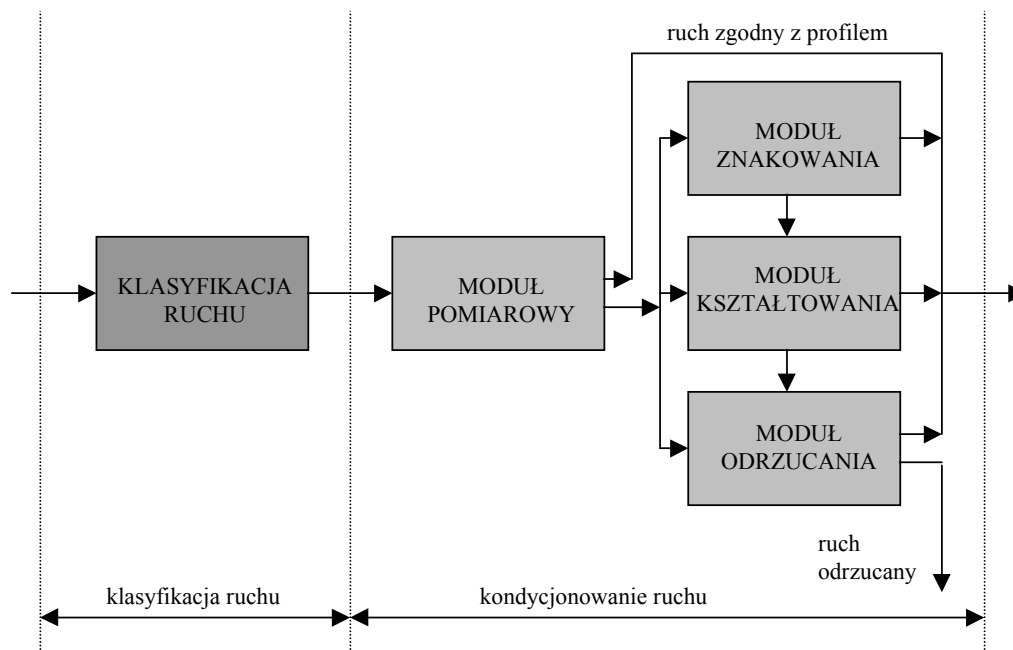
Klasyfikacja ze względu na kod klasy ruchu DSCP jest w koncepcji usług zintegrowanych tożsama ze sprawdzeniem przynależności pakietów do klas ruchu obsługiwanych w danym węźle. Klasyfikacja ta jest realizowana w węzłach wewnętrznych (routerach szkieletowych) sieci.

Wynikiem wykonania zadania klasyfikacji jest odpowiednie, właściwe dla użytego klasyfikatora, znakowanie klasyfikowanych pakietów. W przypadku klasyfikatora BA celem znakowania jest m.in. cechowanie pakietów nie należących do klas ruchu obsługiwanych w danym węźle sieci.

Klasyfikacja na podstawie analizy zawartości pól nagłówka (klasyfikacja MF), w porównaniu z klasyfikacją dotyczącą przynależności do klasy ruchu (klasyfikacja BA), pozwala na większą granulację ruchu, większą równomierność obciążania zasobów sieci i tym samym lepszą efektywność wykorzystania zasobów sieci. Klasyfikacja MF jest klasyfikacją pierwotną pakietów, klasyfikacja BA natomiast jest klasyfikacją wtórną.

10.5.2. KONDYCJONOWANIE RUCHU

Zadaniem kondycjonowania (kontrolowania) ruchu jest wymuszanie przestrzegania warunków kontraktu TCA. Zadanie kondycjonowania ruchu jest realizowane przez moduły: pomiarowy (ang. *traffic meter module*), znakowania ruchu (ang. *traffic marker module*), kształtowania ruchu (ang. *traffic shaper module*) i odrzucania jednostek danych (ang. *dropper module*). Zadaniem pierwszego z wymienionych modułów jest monitorowanie i pomiar strumieni pakietów. Zadania trzech pozostałych modułów wykonawczych są realizowane wtedy, gdy monitorowany i mierzony strumień pakietów nie spełnia warunków kontraktu TCA (rysunek 10.2).



Rysunek 10.2. Moduły funkcjonalne i zasady kondycjonowania ruchu

10.5.2.1. MODUŁ POMIAROWY RUCHU

Moduł pomiarowy jest przeznaczony do pomiaru zgodności rzeczywistego profilu ruchu generowanego przez źródła z profilem ustalonym w kontrakcie. Wynikiem działania modułu pomiarowego jest podział generowanego strumienia pakietów na dwa podstrumienie jednostek danych:

- zgodnego z profilem (ang. *profile conformant*) – należące do tego podstrumienia jednostki danych są wprowadzane do sieci po przejściu lub z pominięciem modułu znakowania,
- niezgodnego z profilem (ang. *profile nonconformant*) – obsługa jednostek danych należących do tego podstrumienia zależy od działań realizowanych przez moduły znakowania jednostek danych, kształtowania ruchu i odrzucania jednostek danych zgodnie z warunkami zdefiniowanymi w kontrakcie TCA.

Jedno z częściej stosowanych rozwiązań modułu pomiarowego polega na implementacji sterownika przepływu, w którym wykorzystywany jest algorytm ciekącego wiadra. Wartości parametrów algorytmu mają wartości wynikające z profilu ruchu ustalonego w kontrakcie TCA.

Wszystkie, należące do podstrumienia ruchu niezgodnego z ustalonym w kontrakcie profilem ruchu i wyselekcjonowane w module pomiarowym, pakiety są kierowane do dalszego przetwarzania w module znakowania albo – z pominięciem modułu znakowania – do modułów kształtowania ruchu i odrzucania pakietów.

10.5.2.2. MODUŁ ZNAKOWANIA RUCHU

Pakiety napływające do modułu znakowania mogą być wcześniej oznakowane lub nie mieć znaku (pakiety po raz pierwszy wprowadzane do sieci). W pierwszym przypadku zmienia się oznakowanie pakietów, a w drugim – znakuje (cehuje).

Zmiany znakowania nie wymagają pakietów należące do ruchu zgodnego z profilem, których warunki obsługi w danym węźle nie ulegają żadnym zmianom.

Zmiana znakowania pakietów jest potrzebna w następujących przypadkach:

➤ przejścia pakietu przez granicę oddzielającą dwie różne domeny usług – zachodzi konieczność zmiany kodu DSCP, polegającej na translacji różnych kodów DSCP obowiązujących w sąsiednich domenach,

➤ naruszenia przez pakiety warunków kontraktu – zmiana znakowania oznacza zwykle przypisanie kodu DSCP, oznaczającego przydział „gorszego” sposobu spedycji (reguły przesyłania),

➤ konieczności zmiany sposobu spedycji z „lepszego” na „gorszy” (ang. *PHB demotion*),

➤ możliwości zmiany sposobu spedycji z „gorszego” na „lepszy” (ang. *PHB promotion*).

Znakowaniem lub zmianą znakowania objęte są zawsze wszystkie te pakiety, które należą do ruchu niezgodnego z ustalonym profilem. W tym przypadku celem znakowania jest potwierdzenie faktu niezgodności rzeczywistego napływu pakietów z warunkami kontraktu; pakiety te będą odrzucane w pierwszej kolejności w razie wystąpienia przeciążenia sieci.

10.5.2.3. MODUŁ KSZTAŁTOWANIA RUCHU

Do modułu kształtowania ruchu kierowane są wszystkie pakiety należące do tego podstrumienia pakietów, który nie jest zgodny z ustalonym w kontrakcie profilem ruchu. Celem modułu kształtowania ruchu jest wymuszenie zgodności tej części ruchu z uzgodnionym profilem ruchu.

Istnieje zasadnicza różnica pomiędzy zadaniami modułów pomiaru ruchu i kształtowania ruchu. W module pomiaru następuje wydzielenie, z całkowitego strumienia pakietów, podstrumienia pakietów zgodnego z profilem. Podstrumień ten jest wyprowadzany do sieci. Podstrumień pozostałych pakietów, odrzuconych przez moduł pomiaru, jest kierowany do modułu kształtowania ruchu, w którym dostosowuje się go do wymaganego profilu ruchu. Dostosowanie oznacza spowalnianie lub odrzucanie pakietów. Spowalnianie polega na chwilowym przechowywaniu pakietów w pamięci buforowej, z której wyprowadzane są do sieci jako strumień pakietów zgodny z ustalonym profilem. Przepelnienie skończonej pojemności pamięci buforowej powoduje, że niektóre z pakietów kierowanych do modułu kształtowania ruchu są odrzucane.

Kształtowanie ruchu jest zadaniem znacznie bardziej złożonym od pomiaru ruchu. Ze względu na to, że kształtowanie ruchu zmienia wartości parametrów charakteryzu-

jących przepływ pakietów (m.in. wzrost opóźnienia, zmniejszanie szybkości transferu, zwiększenie wartości współczynnika strat), algorytmy stosowane w tych modułach i wartości ich parametrów zależą od specyfikacji realizowanych usług. W niektórych przypadkach zadania kształtowania ruchu są realizowane przez restrykcyjne, prewencyjne algorytmy sterowania dostępem.

Jednym z możliwych sposobów realizacji modułu kształtowania ruchu jest użycie sterownika przepływu z implementowanym algorytmem ciekącego wiadra; wartości parametrów algorytmu są wyznaczane na podstawie wartości parametrów kontraktu TCA dla realizowanych usług.

Zadanie kształtowania ruchu jest zwykle wymagane w węzłach brzegowych leżących na granicach domen, tzn. tam, gdzie zachodzi potrzeba dostosowania profilu rzeczywistego, napływającego do węzła strumienia pakietów, do profilu obowiązującego w sąsiedniej domenie sieci.

10.5.2.4. MODUŁ ODRZUCANIA RUCHU

Strumienie pakietów odrzucanych przez moduł pomiarowy, jako niezgodne z uzgodnionym profilem ruchu, mogą być kierowane do modułu kształtowania ruchu lub do modułu odrzucania pakietów. Kierowanie pakietów bezpośrednio z modułu pomiarowego do modułu odrzucania pakietów, z pominięciem modułów znakowania i kształtowania ruchu, dotyczą usług, w których kształtowanie nadmiarowego, w porównaniu z profilem ruchu, strumienia pakietów jest nieuzasadnione. Jednym z powodów może być nieprzydatność dla aplikacji pakietów dodatkowo opóźnianych w module kształtowania ruchu.

10.5.3. LOKALIZACJA MODUŁÓW KLASYFIKOWANIA I KONDYCJONOWANIA RUCHU

Moduły klasyfikacji i kondycjonowania ruchu są zazwyczaj lokalizowane w węzłach wejściowych i wyjściowych, tzn. węzłach brzegowych domen usług sieci. W węzłach wyjściowych domen strumienie pakietów są przekazywane do innych domen. Z różnych powodów omawiane moduły mogą być lokalizowane zarówno w węzłach wewnętrznych domen usług, jak i w domenach nie będących domenami usług zróżnicowanych.

10.5.3.1. LOKALIZACJA W DOMENACH ŹRÓDŁOWYCH USŁUG ZRÓŻNICOWANYCH

Zadaniem modułów klasyfikacji i kondycjonowania ruchu, lokalizowanych w węzłach źródłowych i pośrednich domen źródłowych (ang. *source domain*), tj. domen, w których pakiety są generowane, jest wstępna klasyfikacja i znakowanie jednostek danych (ang. *premarking*) przed wysłaniem pakietów poza domenę źródłową.

Wstępne klasyfikowanie i znakowanie pakietów, poprzedzające podobne zadania realizowane w węzłach brzegowych, jest wykonywane z następujących powodów:

- pozwala klasyfikować i znakować pakiety (różnicować pakiety) w sposób charakterystyczny dla lokalnych wymagań dotyczących jakości usług; celem może być zwłaszcza limitowanie ilości pakietów w definiowanych klasach priorytetów,

- klasyfikacja i znakowanie pakietów w systemie zdecentralizowanym ułatwia wprowadzenie i skuteczne zarządzanie większą liczbą różnych schematów klasyfikacji i znakowania niż w systemie scentralizowanym,

- zadania klasyfikacji i znakowania pakietów w domenach źródłowych są prostsze niż w strumieniach zagregowanych, złożonych ze strumieni generowanych przez wiele źródeł,

- klasyfikacja i znakowanie pakietów w domenach źródłowych dotyczy ograniczonej liczby klas i pakietów w tych klasach; mniejsza liczba schematów klasyfikacji i znakowania w domenie źródłowej upraszcza zarządzanie tymi zadaniami.

Ze względu na to, że wstępną klasyfikację i znakowanie pakietów można wykonywać w kilku węzłach domeny źródłowej, konieczne jest stosowanie mechanizmów zapewniających zgodność zagregowanego ruchu dla wyróżnionych klas spedycji, generowanego w domenie źródłowej, z warunkami kontraktu TCA.

10.5.3.2. LOKALIZACJA W WĘZŁACH BRZEGOWYCH DOMEN USŁUG ZRÓŻNICOWANYCH

Strumienie jednostek danych mogą być klasyfikowane, znakowane i kondycjonowane w węzłach na granicy domen różnych usług: węzłach wejściowych kolejnej (ang. *downstream*) domeny usług i węzłach wyjściowych poprzedniej (ang. *upstream*) domeny usług. To, która z dwóch domen jest odpowiedzialna za znakowanie jednostek danych kodami usług zróżnicowanych, definiowane jest przez kontrakt SLA. Niezależnie od przyjętego sposobu znakowania obowiązuje założenie, że pakiety napływające do węzła wejściowego kolejnej domeny mogą nie spełniać warunków kondycjonowania ruchu TCA. Węzeł wejściowy kolejnej domeny musi być wyposażony w mechanizmy wymuszania spełnienia warunków TCA przez wprowadzany strumień pakietów.

Jeżeli w kolejnej domenie usług nie ma możliwości klasyfikacji i znakowania pakietów, to wszystkie zadania z tego zakresu muszą być wykonane przez węzeł wejściowy tej domeny. Węzeł ten może także realizować zadania klasyfikacji i znakowania pakietów dla źródeł niewyposażonych w odpowiednie moduły, wymaga to stosowania klasyfikacji pierwotnej, tj. klasyfikacji na podstawie zawartości odpowiednich pól nagłówek pakietów.

Osobnym zadaniem jest zadanie klasyfikacji i znakowania pakietów wtedy, gdy dwie sąsiednie domeny usług wykorzystują różne kody DSCP dla tych samych klas spedycji ruchu PHB. W takim przypadku konieczna jest zmiana znakowania, realizowana przez odpowiednie mechanizmy odwzorowywania kodów DSCP dla dwóch

przeciwnych kierunków przepływów pakietów. Za zmianę kodu odpowiada węzeł wejściowy kolejnej domeny lub węzeł wyjściowy poprzedniej domeny.

10.5.3.3. LOKALIZACJA W WĘZŁACH WEWNĘTRZNYCH DOMEN USŁUG ZRÓŻNICOWANYCH

Klasyfikacja i kondycjonowanie ruchu, realizowane w węzłach brzegowych domen usług zróżnicowanych, nie wyklucza implementacji odpowiednich modułów w węzłach wewnętrznych domen. Ich implementacja jest tam niezbędna wtedy, gdy w węzłach wewnętrznych domen następuje agregacja ruchu, która może powodować przeciążenia. W takich przypadkach konieczne jest kształtowanie ruchu nie tylko na brzegu domen, ale także w ich węzłach wewnętrznych sieci.

10.6. STANDARDY SPEDYCJI RUCHU

10.6.1. GRUPY SPEDYCJI RUCHU

Zaproponowane i standaryzowane przez IETF (ang. *Internet Engineering Task Force*) dwie grupy reguł przesyłania ruchu, oprócz domyślnego PHB (ang. *Per Hop Behaviour*) odpowiadającego usłudze bez gwarancji usług, zostały wywiedzione z klasycznych technik szeregowania przepływów w sieciach z komutacją jednostek danych, tzn. AF PHB (ang. *Assured Forwarding Per Hop Behaviour*) i EF PHB (ang. *Expedited Forwarding Per Hop Behaviour*).

Zakres usług oferowanych z wykorzystaniem grup reguł przesyłania ruchu wynika z kontraktu SLA, określającego zakres usług, liczbę obsługiwanych klas, ich jakość usług oraz ich parametry ruchowe. Kontrakt SLA określa parametry algorytmów sterowania ruchem. W zależności od charakterystyk ruchu kontrakty SLA mogą być statyczne (długookresowe) lub dynamiczne. Do ustalania kontraktów dynamicznych stosowany jest protokół rezerwacji (sygnalizacyjny) i jednostki zarządzające zasobami oraz dostępem do tych zasobów – brokery pasma BB (ang. *Bandwidth Broker*).

Obydwie wymienione grupy reguł przesyłania ruchu są wykorzystywane do realizacji usług sieci, według następującej zasady:

- jakość usług sieci, w sensie jakościowym, wynika z reguły przesyłania ruchu, tj. sposobu obsługi pakietów,
- jakość usług sieci, w sensie ilościowym, wynika z uzupełnienia reguły przesyłania ruchu o procedury zarządzania ruchem i wyboru tras.

10.6.1.1. GRUPA AF PHB

Podstawowa idea grupy AF PHB wynika z koncepcji, zgodnie z którą w profilu (kontrakcie) usługi SLA dla źródła strumienia pakietu specyfikowana jest oczekiwana

pojemność (przepływność) sieci (ang. *expected capacity*). W węźle brzegowym sieci strumień generowanych pakietów jest monitorowany, a kolejno napływające pakiety są znakowane jako: spełniające warunki profilu (ang. *in profile*) lub jako niespełniające warunków profilu (ang. *out of profile*); bity *in/out* oznaczają priorytety odrzucania. Od sieci oczekiwane jest udostępnienie przepływności na poziomie (w ilościach) umożliwiającym obsługę wszystkich pakietów spełniających warunki profilu usług, tzn. pakietów o wyższym priorytecie, oraz pakietów nie spełniających warunków profilu, tzn. pakietów o niższym priorytecie tylko wtedy, gdy pojemność sieci nie jest wykorzystywana przez pakiety o wyższym priorytecie.

Omawiana idea znakowania pakietów jest w standardzie AF PHB rozszerzona na cztery klasy spedycji (obsługi) ruchu, w których wyróżnia się po trzy poziomy odrzucania jednostek danych (ang. *drop precedences*). Z tego powodu do znakowania pakietów potrzebnych jest dwanaście różnych punktów kodowych DSCP. Alokacja zasobów sieci, tzn. pojemności pamięci buforowych i łączy, w ilościach minimalnych dla każdej z wyróżnionych klas ruchu jest niezależna. Zasoby sieci są kierowane do poszczególnych klas AF. Niezależność alokacji zasobów dla wyróżnionych grup spedycji ruchu powoduje, że charakterystyki usług, definiowanych za pomocą tej grupy spedycji ruchu, są zbliżone do charakterystyk usług sterowanego obciążenia CLS w koncepcji usług zintegrowanych.

Jakość usług poszczególnych klas AF nie jest definiowana ilościowo, ale jakościowo, tzn. dwie różne klasy AF powinny oferować usługi różnej jakości. Ten sposób definiowania jakości usług jest określany mianem względnego różnicowania jakości usług. Cechą obsługi w grupie jest zachowywanie kolejności jednostek danych w ramach grupy bez względu na poziom odrzucania i stosowane algorytmy sterowania przepływem.

Warunki gwarantujące przekazanie jednostki danych w grupie AF PHB są określone w zależności od przydzielonych zasobów dla klasy AF, chwilowego obciążenia w klasie AF oraz danego prawdopodobieństwa (poziomu) odrzucania jednostek danych. W razie powstania natłoku jednostki danych o wyższym poziomie odrzucania są usuwane przed tymi, których poziom odrzucania jest niższy.

Trzy poziomy priorytetu odrzucania pakietów w każdej z czterech klas spedycji ruchu dają minimalną elastyczność sterowania przepływem w stanie przeciążenia. Odrzucanie pakietów sumarycznego przepływu pakietów danej klasy jest aktywowane trzema różnymi wartościami progowymi zajętości pamięci buforowych, utrzymywanych niezależnie (ang. *assured queue*) dla każdej z klas spedycji: przekroczenie najniższej wartości progowej powoduje odrzucanie pakietów o najwyższym priorytecie odrzucania itd.

Niezależność alokacji zasobów od klas spedycji ruchu oznacza także, że realizacje odrzucania pakietów w każdej z klas spedycji ruchu mogą być różne. Odrzucanie występuje tylko w tej grupie spedycji ruchu, w której nastąpiło przekroczenie ustalonych wartości progowych wykorzystania zasobów.

Implementacja standardu AF PHB w węzłach sieci wymaga użycia wszystkich czterech klas reguł przesyłania ruchu i co najmniej dwóch poziomów priorytetów od-

rzucania pakietów. Standard AF PHB nie precyzuje sposobu podziału zasobów sieci, które pozostają w niej po udostępnieniu grupom spedycji wymaganych minimalnych ilości zasobów.

Niezależność alokacji zasobów dla wyróżnionych klas spedycji ruchu powoduje, że grupa spedycji ruchu AF PHB jest stosowana do definiowania usług, których jakość zależy od obciążenia klas spedycji ruchu. Jedną z takich usług jest usługa AS (ang. *Assured Service*), której charakterystyki są zbliżone do charakterystyk usługi sterowanego obciążenia CLS w koncepcji usług zintegrowanych. Wartości parametrów jakości usług dla AS są specyfikowane w statycznym kontrakcie SLA.

Innym rodzajem usługi korzystającej z grupy spedycji AF PHB jest trzyklasowa usługa OS (ang. *Olympic Service*). Trzy klasy usług OS korzystają z trzech różnych klas spedycji i różnią się ilością alokowanych zasobów w porównaniu do ilości obsługiwanego ruchu. Najwyższą klasę usług, tzn. złotą (ang. *gold*), charakteryzuje najwyższy poziom udostępniania zasobów, definiowany jako iloraz ilości ruchu dozwolonego w tej klasie i ilości zasobów rezerwowanych w tym celu – im wyższy poziom udostępniania zasobów, tym mniejsza wartość ilorazu. W normalnych warunkach obsługi obciążenie ruchem poszczególnych klas usług, tzn. złotej, srebrnej (ang. *silver*) i brązowej (ang. *bronze*), jest tym mniejsze, im wyższa jest klasa usług.

10.6.1.2. GRUPA EF PHB

Grupa spedycji ruchu EF PHB jest definiowana przez formułowanie dwóch wymagań:

- szybkość obsługi pakietów zagregowanego ruchu w każdym z węzłów sieci musi być nie mniejsza od szybkości ustalonej w kontrakcie,
- szybkość obsługi pakietów klasy ruchu w węzle sieci jest niezależna od szybkości obsługi dowolnej innej klasy ruchu współdzielącej zasoby węzła.

Gwarancja ilości zasobów dla grupy spedycji ruchu EF PHB i niezależność alokacji zasobów dla wyróżnionych klas ruchu powoduje, że charakterystyki usług, definiowanych z wykorzystaniem tej grupy spedycji, są zbliżone do charakterystyk gwarantowanej usługi GS definiowanej w koncepcji usług zintegrowanych. W grupie EF PHB jakość usług nie jest definiowana ilościowo, a jedynie jakościowo.

Implementacja grupy obsługi ruchu EF PHB w węzłach sieci wymaga stosowania różnych sposobów priorytetowania ruchu (zadanie dyscypliny szeregowania) o ograniczonej z góry szybkości napływu pakietów (zadanie sterowników przepływu).

Zadaniem dyscyplin szeregowania jest z jednej strony zachowanie właściwości tego sposobu spedycji ruchu, tzn. obsługi z priorytetami, a z drugiej strony ochrona innych klas ruchu współdzielących zasoby przed majoryzowaniem stopnia wykorzystania zasobów przez klasy ruchu obsługiwanego zgodnie ze standardem EF PHB. Wymagania stawiane dyscyplinom szeregowania spełniają m.in. algorytmy kolejkowania z priorytetami oraz algorytmy ważonego sprawiedliwego kolejkowania.

Zadaniem sterowników przepływu jest kształtowanie ruchu, polegające na ograniczeniu z góry szybkości ruchu napływającego do obsługi; właściwości takie mają m.in. sterowniki przepływu korzystające z różnych odmian algorytmu ciekącego wiadra.

Grupa spedycji ruchu EF PHB jest stosowana m.in. do organizacji usług charakteryzujących się małym prawdopodobieństwem odrzucania pakietów, małymi opóźnieniami pakietów wnoszonymi przez węzły sieci i małymi fluktuacjami opóźnienia pakietów, tj. usług o jakości wymaganej dla obsługi ruchu wrażliwego na opóźnienia, zwłaszcza ruchu wymagającego obsługi w czasie rzeczywistym. Warunkiem realizacji tych wymagań jest brak kolejkowania jednostek danych w węzłach sieci. Przykładem usługi, opartej na standardzie grupy spedycji ruchu EF PHB, jest usługa Premium (ang. *Premium Service*) dla bezstratnej transmisji pakietów o niewielkim opóźnieniu tranzytowym i ograniczonych z góry fluktuacjach opóźnienia.

10.6.2. MODELE USŁUG PHB

Ze względu na to, że sieci pakietowe bez gwarancji jakości usług (np. Internet) nie spełniają wymagań dotyczących obsługi ruchu strumieniowego, zaproponowano modele usług, które mogą współistnieć w sieci z usługami bez gwarancji jakości usług. W przypadku sieci Internet takimi modelami usług są m.in. usługa AS (ang. *Assured Service*) zaliczana do grupy reguł przesyłania ruchu AF PHB i usługa Premium wykorzystująca właściwości grupy reguł przesyłania ruchu EF PHB.

Zaproponowane dla sieci Internet usługi są przykładem innego podejścia do świadczenia usług w sieciach bez gwarancji jakości usług. Cechą charakterystyczną tych usług jest: stosowanie w routerach sieci jednego wspólnego mechanizmu obsługi ruchu, realizacja różnicowania jakości usług w routerach brzegowych oraz zarządzanie przepustowością sieci jako żądanym i alokowanym zasobem sieci.

Ograniczenie ilości zasobów sieci wykorzystywanych do realizacji usług AS i Premium pozwala na jednoczesne udostępnianie w sieci wymienionych dwóch usług oraz obsługi ruchu bez gwarancji jakości.

10.6.2.1. USŁUGA AS

Usługa AS (ang. *Assured Service*) polega na udostępnianiu użytkownikowi sieci żądanej pojemności trasy lub sieci, ale bez ścisłych gwarancji, że pojemność ta będzie zawsze dostępna. Realizacja usługi AS nie oznacza rezerwacji pojemności, ale udostępnienie określonej oczekiwanej pojemności (ang. *expected capacity*) i zapewnienie jej dostępności w sensie statystycznym dla generowanego ruchu. Gwarancja, jaką otrzymuje użytkownik usługi, polega na tym, że jest mało prawdopodobne, aby generowany ruch był odrzucany dopóty, dopóki profil generowanego ruchu jest zgodny z oczekiwanym profilem ruchu (ang. *expected capacity profile*). Dokładne znaczenie „mało prawdopodobne” zależy od jakości alokacji zasobów dla usługi. Ruch genero-

wany przez użytkownika usługi AS może nie być zgodny z oczekiwanym profilem ruchu, ale gwarancje obsługi ruchu nadmiarowego są inne niż gwarancje obsługi ruchu zgodnego z profilem.

Opracowanie algorytmu alokacji pojemności dla usługi AS, który zapewnia jednocześnie wysoką jakość obsługi ruchu i wysoki stopień wykorzystania zasobów sieci jest zadaniem bardzo trudnym, jeżeli w ogóle możliwym. Trudność ta rośnie wraz ze zwiększeniem liczby użytkowników usługi, granulacji ruchu i rozproszenia przestrzennego sieci. Realizacja usługi AS wymaga stosowania w routerach mechanizmów kotwiczenia tras.

Realizacja usługi AS jest sposobem na różnicowanie usług bez gwarancji jakości (ang. *providing different levels of best-effort services*) w przypadku przeciążenia sieci; różnicowanie usług bez gwarancji jakości jest możliwe przez odpowiednie zarządzanie pojemnością pamięci buforowych w węzłach sieci. Dostarczanie usługi AS w sieci może być interpretowane jako względne różnicowanie jakości usług dla różnych użytkowników w sieci. Jakość usługi gwarantowanej nie jest definiowana przez jakiejkolwiek parametry ilościowe, natomiast jakość usługi AS jest wyższa od jakości usług bez gwarancji jakości (ang. *better than best-effort*).

Implementacja usługi AS oznacza, że w routerze dostępowym LR (ang. *Leaf Router*), do którego bezpośrednio podłączone jest źródło generowanego ruchu, odbywa się klasyfikacja i kształtowanie ruchu. Jeżeli generowany ruch nie przekracza warunków zdefiniowanych w kontrakcie SLA, to ruch jest zgodny z profilem (ang. *in profile*); w innym przypadku jest poza profilem (ang. *out of profile*). Klasyfikacja pakietów do jednej z wymienionych grup oznacza oznakowanie pakietów. Wszystkie oznakowane pakiety, zarówno zgodne z profilem, jak i spoza profilu ruchu, są umieszczane w kolejce AQ (ang. *Assured Queue*), która jest zarządzana zgodnie z mechanizmem RIO (ang. *Random Early Detection with In and Out*).

Mechanizm zarządzania kolejką RIO jest rozszerzeniem mechanizmu RED, tzn. jest złożeniem dwóch algorytmów RED: jednego dla pakietów zgodnych z profilem i drugiego dla pakietów spoza profilu. Do zarządzania długością kolejki mechanizm RIO wykorzystuje dwie wartości progowe. Jeżeli długość kolejki jest:

- mniejsza od pierwszej wartości progowej, to żaden z pakietów nie jest odrzucany,
- większa od pierwszej wartości progowej i mniejsza od drugiej wartości progowej, to losowo odrzucane są tylko pakiety spoza profilu ruchu,
- większa od drugiej wartości progowej, to losowo odrzucane są zarówno pakiety zgodne z profilem i spoza profilu, z tym że odrzucanie pakietów spoza profilu ruchu jest bardziej agresywne.

10.6.2.2. USŁUGA PREMIUM

Usługa Premium, nazywana także usługą VLL (ang. *Virtual Leased Line*), polega na emulacji w sieci pakietowej łącza dzierżawionego o stałej przepustowości pomiędzy systemami końcowymi. Użytkownik sieci korzystający z usługi Premium otrzy-

muje gwarancję dostępu do nominalnej pojemności ustalonej trasy w sieci i tym samym gwarancję minimalnego opóźnienia kolejkowania i strat pakietów na trasie pomiędzy systemami końcowymi.

Realizacja usługi Premium polega na udostępnianiu użytkownikom maksymalnego profilu pojemności (ang. *peak capacity profile*). Ruch generowany przez użytkownika usługi jest zawsze nie większy od maksymalnej pojemności i jest obsługiwany w routerach z wykorzystaniem osobnych kolejek o wysokim priorytecie. Ruch wchodzący do domeny musi spełniać warunki określone w kontrakcie SLA odnośnie maksymalnej pojemności; pakiety niespełniające warunków kontraktu są odrzucane w routerze brzegowym.

Ruch generowany przez źródło, do obsługi którego stosuje się usługę Premium, jest kształtowany na brzegu sieci. Kształtowanie polega na ograniczaniu ilości wprowadzanego ruchu do wartości maksymalnej pojemności tak, że „wybuchy” źródła ruchu nie są wprowadzane do sieci. Nadmiarowe pakiety, generowane w trakcie „wybuchów” źródła ruchu, mogą być kolejkowane w modułach kształtowania ruchu zlokalizowanych na brzegu sieci. Udostępnienie dla generowanego ruchu żądanej maksymalnej pojemności i ograniczanie ilości wprowadzanego do sieci ruchu pozwala na kreowanie usługi o małych odchyleniach opóźnienia i małych opóźnieniach kolejkowania, której realizacja nie wymaga zarządzania kolejkami.

Zasoby udostępniane użytkownikowi usługi Premium są niezależne od obciążenia trasy i sieci generowanego przez pozostałych użytkowników sieci. Poziom gwarancji jakości usług jest zbliżony do gwarancji w przypadku realizacji modelu usługi gwarantowanej GS w koncepcji sieci z integracją usług.

Realizacja usługi Premium jest dwuetapowa i wymaga implementacji mechanizmów:

- rezerwacji pojemności przez protokół udostępniania pojemności BB (ang. *Bandwidth Broker*) lub program agencki (ang. *agent*) ustanawiający rezerwację w domenie adresowej sieci,
- kotwiczenia tras (ang. *route pinning*) w celu utrzymywania ruchu w zarezerwowanych łączach sieci, niezależnie od dynamicznych zmian tras, dokonywanych w sieci IP.

Implementacja usługi Premium oznacza, że w routerze dostępowym LR (ang. *Leaf Router*), do którego bezpośrednio podłączone jest źródło ruchu, dokonuje się klasyfikacji i kształtowania ruchu. Wynikiem klasyfikacji pakietów ruchu i kształtowania ruchu jest oznakowanie pakietów należących do klasy Premium. Pakiety, które nie należą do tej klasy ruchu, są obsługiwane za pomocą innych dostępnych usług sieci (np. AS lub bez gwarancji jakości). W kolejnych routerach, zlokalizowanych na trasie od źródła do ujścia, kontrolowana jest zgodność ruchu z profilem. We wszystkich routerach trasy pakiety klasy Premium są umieszczane w kolejkach PQ (ang. *Premium Queue*), których priorytet obsługi jest najwyższy. Ponieważ ruch w klasie Premium jest zawsze zgodny z profilem ruchu, kolejki PQ nie wymagają więc zarządzania.

Podstawą alokacji zasobów w usłudze Premium jest maksymalna szybkość transmisji, co powoduje, że transmisja ruchu w klasie Premium wprowadzanego do sieci może odbywać się bez żadnych ograniczeń, żądana bowiem pojemność jest zawsze dostępna. Ten sposób alokacji zasobów jest kosztowny ze względu na ilość udostępnianej pojemności i nieefektywny ze względu na wykorzystanie pojemności.

Jednym ze sposobów zapobiegania nieefektywności wykorzystania pojemności gwarantowanej dla ruchu w usłudze Premium jest stosowanie różnych technik alokacji pojemności BAM (ang. *Bandwidth Allocation Method*) wykorzystujących m.in. wygładzanie ruchu (ang. *traffic smoothing*), adaptacyjne metody odrzucania (ang. *intelligent discarding*), adaptacyjne metody alokacji pojemności.

Podstawowe różnice pomiędzy usługami AS i Premium są konsekwencją różnicy w sposobie dostępu do zasobów sieci (tabela 10.1).

Tabela 10.1. Porównanie usług: Assured Service i Premium Service

Kryterium	Assured Service	Premium Service
Grupa spedykcji ruchu	AF PHB	EF PHB
Udostępnianie zasobów	dostęp do oczekiwanej pojemności	dostępu do maksymalnej pojemności
Rodzaj gwarancji jakości usług	statystyczny	deterministyczny
Sposób definiowania jakości dostarczanych usług	jakościowy	jakościowy
Jakość usług	minimalne opóźnienie kolejkowania, minimalna fluktuacja opóźnienia, minimalne straty	względne różnicowanie jakości usług
Rezerwacja zasobów	nie wymagana	wymagana
Kotwiczenie tras w sieci	wymagane	wymagane
Operacje wykonywane w węzłach brzegowych (dostępowych)	znakowanie jednostek danych przepływów	kształtowanie ruchu

10.6.2.3. PRZYKŁADOWE REALIZACJE USŁUG ASSURED SERVICE I PREMIUM SERVICE

Realizacja każdej usługi AS i (lub) Premium oznacza realizację określonego kontraktu ruchowego SLA, opisującego wymagania dla ruchu generowanego przez źródło. Ze względu na to, że wszystkie źródła danej domeny współdzielą ograniczone zasoby sieci specyfikowane w kontraktach ruchowych, niezbędna jest ich kooperacja w postaci alokacji usług.

Możliwe są dwa sposoby alokacji usług (ang. *service allocation*) sieci precyzowanych kontraktami ruchowymi SLA:

- każde ze źródeł podejmuje decyzję o usłudze niezależnie od innych,

➤ decyzja o usługach dla ruchu generowanego przez wszystkie źródła danej domeny jest podejmowana przez sterownik zasobów (ang. *resource controller*), jakim jest broker pojemności (pasma) BB.

Implementacja wspólnego dla domeny sterownika zasobów jest rozwiązaniem korzystnym ze względu na możliwość centralizacji zarządzania zasobami.

Korzystanie z usługi przez źródło ruchu oznacza, że przed rozpoczęciem transmisji pakietów wybierany jest typ usługi. Decyzja taka jest podejmowana przez źródło niezależnie lub z wykorzystaniem sterownika zasobów. W przypadku korzystania ze sterownika zasobów proces transmisji rozpoczyna się od wysłania wiadomości sygnalizacyjnej ze źródła do sterownika zawierającej żądanie usługi. Zadaniem sterownika zasobów, w odpowiedzi na żądanie usługi, jest generowanie żądania dostępu do określonych zasobów sieci oraz ustalenie i przekazanie zbioru reguł dotyczących zasad klasyfikacji, znakowania i kształtowania ruchu. Informacje te są przekazywane do routera wejściowego źródła ruchu, tj. routera, do którego źródło ruchu jest podłączone bezpośrednio; w ten sposób możliwe jest znakowanie pakietów na wyjściu źródła generowanego ruchu. Przekazanie zbioru reguł wymaga zastosowania protokołu sygnalizacyjnego, za pomocą którego wymieniane są informacje pomiędzy źródłami, routerami i sterownikami zasobów w danej domenie sieci. Protokołem sygnalizacyjnym może być np. protokół rezerwacji RSVP.

Działanie sterownika zasobów zależy od typu kontraktu ruchowego SLA pomiędzy użytkownikiem a siecią ISP (ang. *Internet Service Provider*).

W przypadku gdy kontrakt ruchowy SLA jest statyczny (ang. *static SLA*), zadanie wyznaczania zbioru reguł dla routera wejściowego źródła jest procesem jednorazowym. Wynikiem działania jest statyczna alokacja zasobów dla każdego z użytkowników usług sieci. Zasoby nie wykorzystane mogą być współdzielone przez innych użytkowników.

Gdy kontrakt ruchowy jest dynamiczny (ang. *dynamic SLA*), realizacja usługi wymaga wielokrotnej wymiany informacji pomiędzy routerami sieci a sterownikiem zasobów. Każdorazowa zmiana warunków kontraktu ruchowego SLA wymaga użycia protokołu sygnalizacyjnego do wysłania żądania dostępu do zasobów sieci ISP. Zadaniem routera brzegowego sieci ISP jest sprawdzenie możliwości obsługi żądanego kontraktu SLA i potwierdzenie lub odrzucenie żądania dostępu do zasobów. Obsługa żądania sterownika zasobów dostępu do zasobów sieci w routerze brzegowym sieci ISP zależy od funkcjonalności routera. Zbiór reguł klasyfikowania, kontrolowania, znakowania i kształtowania ruchu może być ustalany przez router potwierdzający dostępność żądanych zasobów lub przez sterownik zasobów po otrzymaniu z routera potwierdzenia dostępności żądanych zasobów sieci.

10.6.2.3.1. ASSURED SERVICE ZE STATYCZNYM KONTRAKTEM SLA

Źródło S , zlokalizowane w Sieci1, żąda zrealizowania usługi AS w sieci ISP. Ujściem przepływu, opisanego statycznym kontraktem ruchowym, jest ujście D , zlokalizowane w Sieci2.

Proces obsługi żądania źródła ruchu jest następujący (rysunek 10.3):

1. Źródło *S* wysyła wiadomość PATH protokołu sygnalizacyjnego RSVP do lokalnego sterownika zasobów BB1 z żądaniem skonfigurowania usługi AS dla generowanego ruchu.

2. Akceptacja żądania skonfigurowania usługi AS przez sterownik zasobów BB1 oznacza, że konfigurowany jest router dostępowy LRS źródła *S*; w innym przypadku sterownik zasobów BB1 powiadamia źródło o odrzuceniu żądania skonfigurowania usługi AS. Konfiguracja routera LRS oznacza, że znany jest w nim zbiór reguł dotyczących znakowania, klasyfikowania i kształtowania ruchu.

3. Sterownik BB1 powiadamia źródło *S* o przyjęciu żądania skonfigurowania usługi.

4. Źródło *S* wysyła pakiety do routera dostępowego LRS.

5. Skonfigurowany przez sterownik zasobów BB1 router LRS znakuje nadchodzące pakiety, tzn. np. ustawia (lub nie) określony bit np. pola ToS (ang. *Type of Service*) w nagłówku pakietu (pakiety z nieustawionym bitem są pakietami spoza profilu ruchu precyzowanego w kontrakcie SLA). Tak oznakowane pakiety są wysyłane w kierunku routera brzegowego Sieci1, tzn. routera ER1 – routera wyjściowego Sieci1. Ruch wychodzący z routera LRS jest ruchem klasy AS.

6. W każdym routerze na trasie od routera dostępowego LRS do routera brzegowego ER1 na styku Sieci1 i sieci ISP, z wyłączeniem routera LRS, wszystkie pakiety są klasyfikowane, tzn. sprawdzana jest ich przynależność do klasy AS. W każdym z routerów organizowana jest kolejka AQ, do sterowania długością której stosowany jest mechanizm RIO.

7. W routerze brzegowym BR1 sieci ISP, zlokalizowanym na styku Sieci1 i sieci ISP, dokonywana jest klasyfikacja i kontrola (ang. *policing*) pakietów przepływu. Kontrola polega na tym, że:

➤ oznakowanie wszystkich pakietów, oznakowanych wcześniej przez router RS jako pakiety spoza profilu ruchu, nie ulega zmianie,

➤ jeżeli szybkość transmisji pakietów oznakowanych wcześniej jako pakiety zgodne z profilem ruchu jest większa od szybkości podanej w profilu ruchu, to ustawienie bitów pakietów zaliczanych do ruchu nadmiarowego jest znoszone (ang. *remarking*).

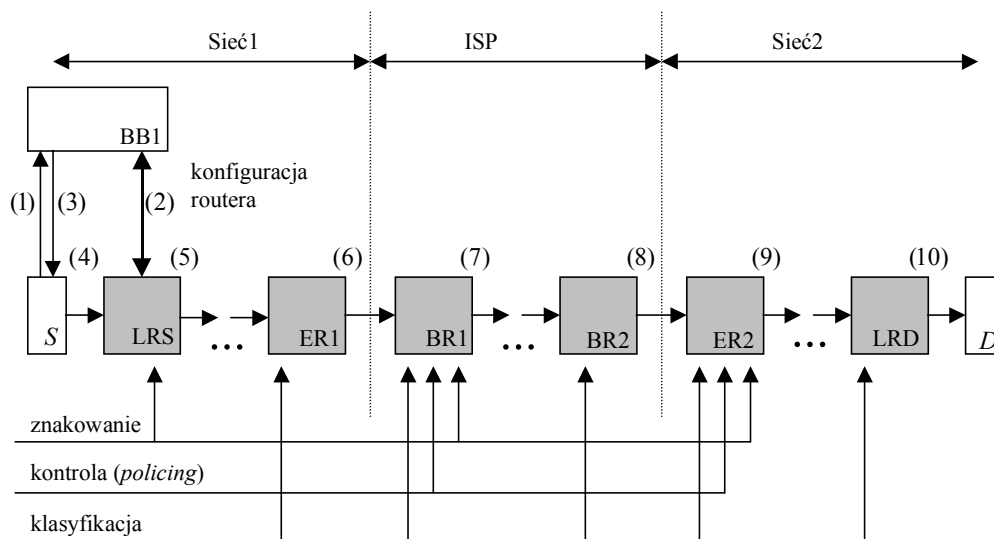
Tak oznakowane pakiety są kolejgowane w kolejce AQ z mechanizmem RIO, a następnie transmitowane przez sieć ISP do wyjściowego routera brzegowego sieci ISP, tzn. BR2.

8. W każdym routerze sieci ISP na trasie od routera BR1 do routera brzegowego BR2 na styku ISP i Sieci2, z wyłączeniem routera BR1, wszystkie pakiety są klasyfikowane i umieszczane w kolejkach AQ z mechanizmem RIO.

9. W routerze brzegowym ER2, należącym do Sieci2 i zlokalizowanym na styku sieci ISP, dokonywana jest klasyfikacja i kontrola pakietów przepływu, podobnie jak w routerze brzegowym BR1.

10. W każdym routerze na trasie od routera brzegowego ER2 do routera wyjściowego LRD dla ujścia *D*, z wyłączeniem routera ER2, wszystkie pakiety są klasyfiko-

wane. W każdym z routerów organizowana jest kolejka AQ, której długość jest regulowana zgodnie z algorytmem RIO.



Rysunek 10.3. Przykładowa realizacja usługi Assured Service

Jakość usługi AS jest wynikiem względnego różnicowania jakości usług w sieci; do jej realizacji wystarczy skonfigurowanie routera wejściowego źródła generowanego ruchu. Ponieważ straty pakietów oznakowanych jako zgodne z profilem są względnie małe nawet w przypadku wystąpienia przeciążeń, jakość usług jest więc przewidywalna, jeżeli generowany ruch jest zgodny z kontraktem ruchowym. W przypadku braku przeciążeń wszystkie generowane pakiety będą dostarczone do ujścia.

Dostępność usługi AS w sieci pakietowej jednoznacznie określa wymagania stawiane poszczególnym routerom na trasach pomiędzy źródłami i ujściami ruchu. W omawianej, przykładowej realizacji usługi AS, wymagana funkcjonalność routerów jest następująca:

- routery dostępowe źródła LR (ang. *Leaf Router*) – klasyfikacja i znakowanie pakietów,
- routery brzegowe ER (ang. *Edge Router*) domen sieci źródła i ujścia – klasyfikacja pakietów,
- routery brzegowe BR (ang. *Boundary Router*) sieci ISP – klasyfikacja pakietów, kontrola (ang. *policing*) i kształtowanie ruchu oraz znakowanie pakietów,
- wszystkie inne routery – klasyfikacja pakietów ruchu AS i obsługa kolejek AQ zgodnie z algorytmem RIO.

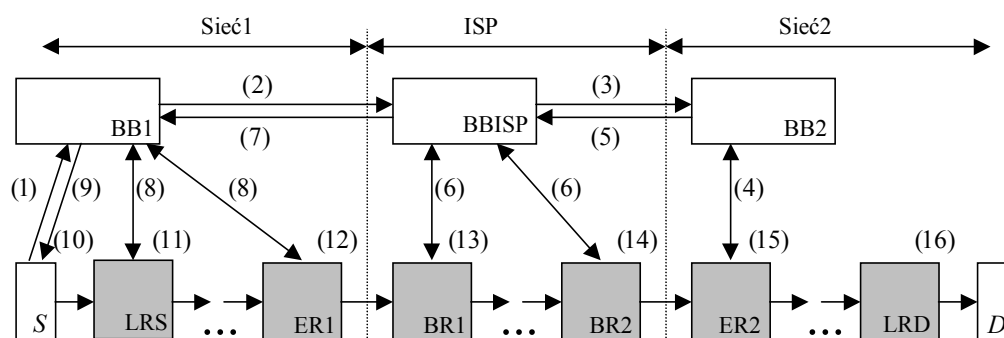
10.6.2.3.2. PREMIUM SERVICE Z DYNAMICZNYM KONTRAKTEM SLA

Zlokalizowane w Sieci1 źródło *S* żąda zrealizowania usługi Premium w sieci ISP. Ujściem przepływu, opisanego dynamicznym kontraktem ruchowym, jest ujście *D*,

zlokalizowane w Sieci2. Ponieważ usługa Premium jest równoznaczna z emulacją łącza dzierżawionego na poziomie pakietów, rozpoczęcie transmisji pakietów przez źródło wymaga więc rezerwacji zasobów na trasie od źródła do ujścia. Realizacja usługi Premium składa się z dwóch etapów: rezerwacji zasobów i transmisji pakietów.

Etap I – rezerwacja zasobów

Rezerwacja zasobów w celu realizacji usługi Premium (np. z wykorzystaniem protokołu RSVP) ma następujący przebieg (rysunek 10.4):



Rysunek 10.4. Przykładowa realizacja usługi Premium Service

1. Źródło S , zlokalizowane w Sieci1, wysyła do lokalnego sterownika zasobów BB1 wiadomość RSVP PATH z żądaniem skonfigurowania usługi Premium.

W lokalnym sterowniku zasobów BB1 podejmowana jest decyzja o przyjęciu lub odrzuceniu żądania. Jeżeli żądanie jest odrzucane, to wysyłane jest powiadomienie do źródła S i procedura rezerwacji zostaje zakończona.

2. Akceptacja żądania przez sterownik zasobów BB1 powoduje wysłanie wiadomości RSVP PATH z żądaniem udostępnienia zasobów do sterownika zasobów BBISP sieci ISP. Jeżeli żądanie jest odrzucane, to wysyłane jest powiadomienie do sterownika BB1, który powiadamia źródło S i procedura rezerwacji zostaje zakończona.

3. Akceptacja żądania przez sterownik zasobów BBISP powoduje wysłanie wiadomości RSVP PATH z żądaniem udostępnienia zasobów do sterownika zasobów BB2 w Sieci2. Jeżeli żądanie jest odrzucane, to z BB2 wysyłane jest powiadomienie do sterownika BBISP, skąd kolejno do sterownika BB1 i źródła S ; procedura rezerwacji zostaje zakończona.

4. Akceptacja żądania przez sterownik zasobów BB2 powoduje wysłanie wiadomości RSVP konfigurujących reguły klasyfikacji i kontroli ruchu w routerze ER2 w Sieci2.

5. Akceptacja żądania przez sterownik zasobów BB2 i skonfigurowanie routera ER2 powoduje wysłanie wiadomości RSVP do sterownika zasobów BBISP.

6. Odebranie wiadomości RSVP przez sterownik BBISP powoduje wysłanie wiadomości konfigurujących reguły klasyfikacji i kontroli ruchu w routerze BR1 oraz reguły klasyfikacji i kształtowania ruchu w routerze BR2.

7. Skonfigurowanie routerów BR1 i BR2 powoduje wysłanie przez sterownik BBISP wiadomości RSVP RSVP do sterownika zasobów BB1.

8. Odebranie wiadomości RSVP RSVP przez sterownik BB1 powoduje wysłanie:

➤ wiadomości konfigurujących reguły klasyfikacji i kontroli ruchu w routerze LRS,

➤ wiadomości konfigurujących reguły klasyfikacji i kształtowania ruchu w routerze BR2.

9. Skonfigurowanie routerów LRS i ER1 powoduje wysłanie przez sterownik BB1 do źródła *S* wiadomości RSVP RSVP. Odebranie przez źródło *S* tej wiadomości jest powiadomieniem o zarezerwowaniu zasobów zdefiniowanych w kontrakcie ruchowym SLA i dostępności usługi Premium; powoduje rozpoczęcie transmisji pakietów.

Etap II – transmisja pakietów

Usługa Premium dla generowanego ruchu pakietów jest wykonywana w następujący sposób (rysunek 10.4):

10. Źródło *S* wysyła pakiety do routera dostępowego LRS.

11. Router realizuje klasyfikację MF; jeżeli generowany ruch jest zgodny z profilem danym kontraktem SLA, to każdy z pakietów jest znakowany; w innym przypadku odbywa się kształtowanie ruchu, wynikiem którego jest strumień pakietów klasy Premium zgodny z profilem. Wszystkie pakiety są kierowane do kolejki PQ. Ruch wychodzący z routera LRS jest ruchem klasy Premium.

12. W każdym routerze na trasie od routera dostępowego LRS do routera brzegowego ER1 na styku Sieci1 i ISP, z wyłączeniem routera LRS, wszystkie pakiety są klasyfikowane, tzn. sprawdzana jest ich przynależność do klasy Premium. Pakiety są wprowadzane do kolejek PQ. W routerze ER1 oprócz klasyfikacji sprawdzana jest zgodność profilu ruchu z kontraktem SLA. Dodatkowo w routerze wykonywane jest ponowne kształtowanie ruchu (ang. *reshaping*), z tym że nie dotyczy ono indywidualnych przepływów, ale zagregowanego ruchu kierowanego z routera ER1 do routera brzegowego BR1 sieci ISP.

13. W routerze BR1 pakiety są klasyfikowane i kontrolowane; pakiety nadmiarowe są odrzucane.

14. W każdym z routerów, leżących na trasie od routera BR1 do routera BR2, pakiety są klasyfikowane, z tym że w routerze BR2 dodatkowo odbywa się kolejne kształtowanie ruchu.

15. W routerze ER2 wykonywane są działania takie jak w routerze BR1, tzn. pakiety są klasyfikowane i kontrolowana jest zgodność ruchu z profilem. W razie wykrycia pakietów nadmiarowych, są one odrzucane.

16. W każdym z routerów, zlokalizowanych na trasie od routera ER2 do routera LRD, wykonywana jest procedura klasyfikacji pakietów. Router LRD przekazuje pakiety ruchu Premium z kolejki PQ do ujścia *D*.

Dostępność usługi Premium w sieci pakietowej jednoznacznie definiuje wymagania stawiane poszczególnym routerom na trasach pomiędzy źródłami i ujściami ruchu. Wymagana funkcjonalność routerów jest następująca:

- routery dostępowe źródeł LR (ang. *Leaf Router*) – klasyfikacja pakietów, kształtowanie ruchu Premium i znakowanie pakietów,
- routery brzegowe ER (ang. *Edge Router*) domen sieci źródła i ujścia – klasyfikacja pakietów,
- routery brzegowe BR (ang. *Boundary Router*) sieci ISP – klasyfikacja pakietów oraz kontrola (ang. *policing*) i kształtowanie ruchu Premium,
- wszystkie inne routery – klasyfikacja pakietów ruchu Premium i obsługa kolejek PQ z najwyższym priorytetem.

10.7. WYMAGANIA SIECI Z KONCEPCJĄ USŁUG ZRÓŻNICOWANYCH

10.7.1. WYMAGANIA Z ZAKRESU RÓŻNICOWANIA RUCHU

Na ogół prawdziwe jest stwierdzenie, że obsługa ruchu w sieci bez gwarancji jakości usług (zwana także *send and pray model*), pozwala uzyskać żadaną jakość usług pod warunkiem implementacji w sieci odpowiedniego systemu udostępniania zasobów (ang. *resource provisioning*) sieci. Zawsze jednak niezbędne są mechanizmy różnicowania ruchu i obsługi ruchu, które muszą być aktywowane w razie powstawania przeciążeń. Powstanie przeciążenia zawsze oznacza, że brak jest zasobów w ilościach umożliwiających dostarczenie żądanej lub deklarowanej jakości usług.

Wprowadzenie dowolnego mechanizmu różnicowania zawsze wymaga gromadzenia, utrzymywania i dystrybucji informacji o stanie sieci, co z kolei zwiększa zapotrzebowanie na pojemność łącza transmisyjnych, pamięć i moc przetwarzania w węzłach sieci (routerach).

Implementacja wyrafinowanych koncepcji i architektur sieci z różnicowaniem usług jest możliwa przez lokalizowanie różnych mechanizmów i procedur albo w systemach końcowych, albo w węzłach sieci. Uzasadnieniem dla mało efektywnego lokalizowania mechanizmów różnicowania w systemach końcowych jest ograniczona funkcjonalność klasycznych architektur sieci, zwłaszcza architektur routerów. Lokalizacja mechanizmów wspierających różnicowanie ruchu i usług w węzłach sieci wymaga zmian tradycyjnych sposobów organizacji routerów m.in. w zadaniach organizacji, komutacji, zwielokrotniania, szeregowania, kierowania i klasyfikowania przepływów w sieciach.

Likwidacja ograniczeń tradycyjnych architektur routerów wymaga implementacji w tych ostatnich m.in. następujących mechanizmów:

- klasyfikacji pakietów, których zadaniem jest wielokryterialne rozróżnianie pakietów i grup pakietów,
- zarządzania pamięciami buforowymi, których zadaniem jest podejmowanie decyzji o ilości pamięci przyznawanych różnym klasom ruchu i zasadach odrzucania ruchu w warunkach przeciążenia,

➤ szeregowania pakietów, których zadaniem jest wymuszanie alokacji zasobów umożliwiającej spełnianie gwarancji dotyczących przepływności i opóźnień.

10.7.2. WYMAGANIA Z ZAKRESU OBSŁUGI RUCHU W CZASIE RZECZYWISTYM

W sieci z koncepcją usług zróżnicowanych wymagane jest przede wszystkim, aby maksymalne opóźnienie przetwarzania nagłówka pakietu nie było większe od opóźnienia pakietu z klasy ruchu o najniższym dopuszczalnym opóźnieniu pakietu. Wynika to z tego, że przetwarzanie nagłówka dowolnego pakietu, które jest potrzebne do podjęcia decyzji determinującej jakość obsługi pakietu w sieci, nie może odbywać się z naruszeniem warunków obsługi najbardziej wymagającej (w sensie jakości usług) klasy ruchu w sieci.

Z wymagania tego wynika główne ograniczenie architektury routera sieci z różnicowaniem usług: przetwarzanie nagłówka pakietu nie może być poprzedzane opóźnieniem kolejkowania, tzn. kolejkowanie pakietów na wejściach routera nie jest dopuszczalne.

Zniesienie kolejkowania pakietów na wejściu routera oznacza, że przetwarzanie nagłówków odbywa się z szybkością równą szybkości transmisji (ang. *wire speed*) oraz że szybkość przetwarzania nie zależy od charakterystyk ruchu.

Zachowanie zasady, zgodnie z którą pakiety w węźle sieci (routerze) nie są kolejkowane przed przetwarzaniem nagłówka, jest istotne ze względu na to, że dopiero przetworzenie nagłówków umożliwi routerowi ustalenie poziomu jakości usług żądanych dla pakietów różnych klas ruchu. Powstawanie w routerze kolejek, w których pakiety oczekują na przetwarzanie nagłówków oznacza, że dolne ograniczenie jakości usług jest determinowane wystąpieniem tzw. najgorszego przypadku, tzn. wartością maksymalnego czasu oczekiwania pakietu na przetwarzanie nagłówka.

W tradycyjnych rozwiązaniach routerów do przekazywania jednostek danych stosuje się różne rodzaje buforowania dynamicznego (ang. *cache-based methods*). Rozwiązania takie opierają się na założeniu, że:

➤ połączenie jest sekwencją pakietów przekazywanych pomiędzy tymi samymi źródłem i ujściem, w czasie trwania krótszym niż kilkadziesiąt sekund,

➤ nadchodzące do routera pakiety są skorelowane, tzn. jeżeli napływający pakiet należy do nowego połączenia (przepływu), to prawdopodobne jest, że kolejne pakiety należą do tego samego połączenia.

Dla tak definiowanych połączeń, generowanych przez różne aplikacje, zasadne jest stosowanie metod z dynamicznym buforowaniem, zgodnie z którym przetwarzanie nagłówka pierwszego pakietu polega na analizie całego nagłówka. Tak przetworzony nagłówek wraz z opisem akcji, właściwej dla kolejnych pakietów o takim samym nagłówku, jest buforowany. W przypadku napływu pakietu o takim samym nagłówku, tzn. kolejnego pakietu połączenia, decyzja o sposobie obsługi tego pakietu jest podejmowana na podstawie zawartości bufora.

Ze względu na szybkość przetwarzania nagłówków metody dynamicznego buforowania mają następujące ograniczenia:

- zaprojektowanie efektywnej funkcji skrótu bez apriorycznej znajomości wartości nagłówków jest zadaniem trudnym,
- nie jest możliwe zaprojektowanie jednakowo efektywnej funkcji skrótu dla różnych profili ruchu,
- nie jest możliwa całkowita eliminacja kolejkowania z uwagi na wrażliwość metod dynamicznego buforowania na długość połączeń oraz profile ruchu.

10.7.3. ARCHITEKTURY ROUTERÓW

Przydatność architektur routerów różnych generacji do realizacji zadań, istotnych dla funkcjonalności sieci z różnicowaniem usług, wynika z zastosowanych rozwiązań dotyczących obsługi wejść i wyjść, sposobów przekazywania danych w routerze pomiędzy procesorami i interfejsami, wewnętrznych rozwiązań transmisji danych, ilości przekazywanych danych, sekwencji działań oraz możliwości zrównoleglenia wybranych operacji.

Podstawą klasyfikacji routerów do kolejnych generacji jest wieloprocessorowość, zrównoleglenie przetwarzania i techniki komutacji.

10.7.3.1. SCENTRALIZOWANE PRZETWARZANIE I WSPÓLDZIELENIE MAGISTRALI

Architektura tzw. pierwszej generacji routerów, wykorzystujących pojedynczy, uniwersalny procesor i współdzieloną magistralę SPSB (ang. *Single Processor – Shared Bus*) jest zorientowana na obsługę wielu różnych protokołów, gdzie zagwarantowanie połączeń (ang. *connectivity*) jest ważniejsze od efektywności przekazywania (ang. *forwarding capacity*) jednostek danych.

Routerzy pierwszej generacji stosują uniwersalny procesor i współdzieloną magistralę do łączenia różnych interfejsów. Wszystkie funkcje związane z wyznaczaniem tras i sterowaniem przepływem są realizowane przez centralny procesor.

Napływające do poszczególnych interfejsów wejściowych pakiety są przekazywane do centralnego procesora, w którym wyznaczany jest adres kolejnego węzła (routera) sieci, po czym pakiety są przekazywane do odpowiedniego interfejsu wyjściowego routera. Ten sposób transmisji i przetwarzania oznacza, że każdy pakiet jest transmitowany w magistrali co najmniej dwa razy – z wejścia do procesora oraz z procesora do wyjścia.

Jakość działania routera w dużym stopniu zależy od przepustowości magistrali oraz szybkości procesora. Ze względu na wielofunkcyjność centralnego procesora jakość działania routera zależy od systemu operacyjnego. Podstawowym sposobem zwiększania jakości działania routerów pierwszej generacji była budowa różnych systemów operacyjnych.

Wzrost przepustowości interfejsów wejściowych i wyjściowych jest ograniczony skalowalnością architektury, determinowanej scentralizowanym przetwarzaniem i sposobem komunikacji pomiędzy każdym z interfejsów i procesorem oraz pomiędzy procesorem a interfejsami.

10.7.3.2. WIELOPROCESOROWE PRZETWARZANIE I WSPÓLDZIELENIE MAGISTRALI

Głównym celem architektury routerów tzw. drugiej generacji, w której wykorzystywane jest rozproszone, wieloprocessorowe przetwarzanie i współdzielenie magistrali MPSB (ang. *Multiple Processors – Shared Bus*), jest ograniczenie ilości transmisji w magistrali.

Ograniczenie transmisji w magistrali wynika z rozdzielenia przetwarzania niezbędnego do kierowania ruchem (ang. *forwarding*) na przetwarzanie w centralnym procesorze i w procesorach zlokalizowanych w interfejsach wejść routera, wyposażonych w uniwersalne procesory i mechanizmy dynamicznego buforowania.

Pierwszy pakiet nowego połączenia jest przekazywany do procesora centralnego, gdzie są wyznaczane działania właściwe dla wszystkich pakietów identyfikowanych wartościami parametrów połączenia. Zastosowanie dynamicznego buforowania w interfejsach pozwala na przekazywanie kolejnych jednostek danych, należących do tych samych połączeń, bezpośrednio między interfejsami. Dynamiczne buforowanie występuje w dwóch wersjach: dla połączeń (ang. *per-connection caching*) i dla tras (ang. *per-route caching*). Rozdzielenie przetwarzania redukuje liczbę transmisji z interfejsów wejściowych do procesora centralnego do liczby połączeń.

Ograniczeniem możliwości wzrostu jakości działania routerów korzystających z omawianej architektury jest organizacja komunikacji pomiędzy interfejsami za pomocą współdzielonej magistrali oraz uniwersalność procesorów.

10.7.3.3. WIELOPROCESOROWE PRZETWARZANIE I KOMUTACJA PRZESTRZENNA

W routerach tzw. trzeciej generacji stosuje się wieloprocessorowość i komutację przestrzenną MPSS (ang. *Multiple Processors – Space Switching*). Zastąpienie magistrali, występującej w routerach wcześniejszych generacji, krosownicą (ang. *crossbar*) likwiduje ograniczenia jakości działania wynikające z konfliktów w dostępie do magistrali i ograniczonej przepustowości współdzielonej magistrali.

Jednocześnie zastosowanie pola komutacyjnego z komutacją przestrzenną oznacza rezygnację z rozdzielenia przetwarzania na przetwarzanie scentralizowane i przetwarzanie zlokalizowane w interfejsach.

Ograniczenia architektury routerów trzeciej generacji wynikają ze stosowania uniwersalnych procesorów; możliwości obsługi przepływów o dużej intensywności są ograniczone szybkością przetwarzania.

10.7.3.4. WSPÓLDZIELONE PRZETWARZANIE RÓWNOLEGŁE I KOMUTACJA PRZESTRZENNA

Zrównoleglenie przetwarzania nagłówków jest jednym z rozwiązań routerów pozwalających na skalowanie szybkości obsługi pakietów w dużych zakresach wartości. Idea architektury routerów tzw. czwartej generacji, w których implementowane jest przetwarzanie równoległe i komutacja przestrzenna SPPSS (ang. *Shared Parallel Processing – Space Switching*), wywodzi się z założenia, że prawdopodobieństwo jednoczesnego przeciążenia wszystkich interfejsów wejściowych routera jest bardzo małe. Zgodnie z tym założeniem rozproszenie i zrównoleglenie rozwiązywania zadania wyznaczania adresu dla połączeń może zwiększyć szybkość działania routera i jego skalowalność.

Do rozwiązywania zadań wyboru adresu wykorzystywane są wszystkie, chwilowo nie zajęte procesory przekazywania (ang. *forwarding engine*). W celu przyspieszenia wykonania zadania do procesorów jest przekazywany tylko nagłówek pakietu; w ten sposób eliminuje się zbędną transmisję całego pakietu. Całe pakiety są transmitowane tylko pomiędzy interfejsami wejściowymi i wyjściowymi. Wynik przetwarzania, tj. adres następnego węzła sieci dla danego pakietu jest zwracany do interfejsu, który przekazuje pakiet do wyznaczonego interfejsu wyjściowego.

Czas potrzebny na rozwiązanie zadania wyznaczenia adresu zależy m.in. od aktualnego obciążenia procesorów oraz zastosowanego mechanizmu równoważenia obciążeń. Realizacja wyznaczania adresu z wykorzystaniem przetwarzania równoległego wiąże się dodatkowo z koniecznością utrzymywania odpowiednich baz danych we wszystkich procesorach oraz implementacji mechanizmów zapewniających spójność tych baz danych.

Skalowalność omawianej architektury opiera się na założeniu, że dodanie odpowiedniej liczby procesorów pozwala dowolnie zwiększyć przepływność routera; wraz ze zwiększeniem liczby procesorów rośnie prawdopodobieństwo, że co najmniej jeden z procesorów zawsze nie jest zajęty.

10.7.4. OPTIMALIZACJA PRZETWARZANIA PAKIETÓW W ROUTERZE

Podstawą koncepcji uniwersalnego, optymalnego routera do obsługi ruchu należącego do różnych klas ruchu w sieci z różnicowaniem usług jest podział funkcjonalny procesu przetwarzania na specyficzne zadania i operacje, możliwe do wykonania z szybkością porównywalną do szybkości napływu pakietów i niezależnie od charakterystyk napływającego do routera ruchu.

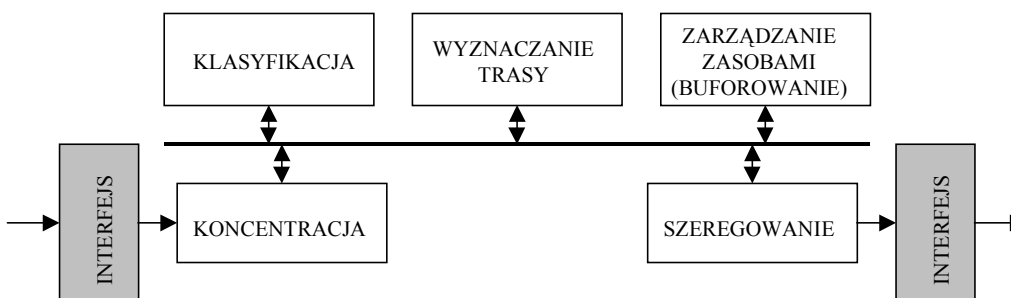
W procesie przetwarzania pakietu w routerze, którego celem jest zarówno przekazanie pakietu do kolejnego węzła sieci, jak i uzyskanie predefiniowanej jakości usług sieci, można wyróżnić następujące zadania:

- filtracji i klasyfikacji pakietów,

- wyznaczenia adresu kolejnego węzła na trasie do ujścia za pomocą tablicy routingu,
- kolejkowania pakietów w kolejce wyjściowej interfejsu wyznaczonego w zadaniach klasyfikacji i routingu,
- szeregowania pakietów w kolejkach wyjściowych routera w celu wymuszenia takiej alokacji zasobów, która jest warunkiem uzyskania żądanej jakości usług.

Rozwiązywanie wymienionych zadań w strukturze wieloprocessorowej wiąże się z dużą złożonością pamięciową zadania oraz algorytmów synchronizacji.

Dla wymienionych zadań wspólne jest to, że informacjami wejściowymi są nagłówki pakietu oraz struktury danych tworzone na potrzeby różnych protokołów. Rozłączność poszczególnych zadań oraz specyfika i rozłączność struktur danych, niezbędnych do wykonania każdego zadania, mogą być wykorzystane w rozwiązaniu, w którym specjalizowane procesory wykonują poszczególne zadania w kolejności wynikającej ze specyfiki procesu przekazywania pakietu w routerze (węźle) sieci. Każdy z procesorów wykonuje jedno z zadań, wykorzystując tylko te struktury danych, które wiążą się z zakresem funkcjonalnym tego zadania (rysunek 10.5).



Rysunek 10.5. Schemat przetwarzania nagłówka pakietu w zadaniu przekazywania pakietu w routerze

10.8. KLASYFIKACJA JEDNOSTEK DANYCH

Tradycyjne rozwiązanie zadania wyznaczenia trasy w węźle sieci z komutacją jednostek danych polega na wyborze łącza wyjściowego z danego węzła na podstawie adresu systemu końcowego (ujścia), a następnie na transferze jednostki danych. Wyznaczanie trasy jednostek danych tylko na podstawie adresu ujścia oznacza, że w danym węźle wszystkie jednostki danych o takim samym adresie ujścia są obsługiwane tak samo. Rozwiązanie takie uniemożliwia różnicowanie składowych zagregowanego strumienia ruchu obsługiwanego w danym węźle i tym samym różnicowanie jakości usług dla strumieni składowych.

Różnicowanie ruchu i jakości usług, dostarczanej dla różnych składowych zagregowanego strumienia jednostek danych, wymaga klasyfikacji (ang. *classification*) lub filtracji (ang. *filtering*) jednostek danych na podstawie zawartości wybranych pól na-

główna pakietu. Realizację klasyfikacji poprzedza wybór pól lub fragmentów pól nagłówka, których zawartość będzie wykorzystywana do klasyfikacji oraz znalezienia reguł klasyfikacji (reguły filtrowania lub filtru) odpowiadającym możliwym sposobom obsługi składowych ruchu w sieci.

Klasyfikacja jednostek danych jest podstawową operacją realizowaną w praktycznie każdym urządzeniu sieciowym, a jakość klasyfikacji ma krytyczne znaczenie dla jakości usług dostarczanych w sieci teleinformatycznej. Wynikiem klasyfikacji jednostek danych jest ich grupowanie (kategoryzacja) w przepływy. Każda z reguł klasyfikacji (ang. *classification rule*) klasyfikatora, definiowanego jako zbiór reguł klasyfikacji, określa jednoznacznie przepływ i odwrotnie. Klasyfikacja jednostek danych w węzłach sieci jest warunkiem implementacji każdej koncepcji różnicowania jakości usług w sieci.

Klasyfikacja pakietów jest jedną z podstawowych operacji wykonywanych w urządzeniach sieciowych, takich jak przełączniki i routery. Zakres zastosowań klasyfikacji pakietów jest bardzo duży i obejmuje zarówno względnie proste zadania przeszukiwania tablic tras na podstawie zawartości pola adresu ujęcia w nagłówkach pakietów, jak i złożone zadania identyfikacji pakietów na podstawie zawartości wielu różnych pól nagłówków. Ze względu na to, że:

- zawarte w polach nagłówka pakietu warstwy sieciowej informacje dotyczą zdarzeń i procesów realizowanych w warstwie transportowej (warstwa czwarta) i wyższych architektury sieci,
- klasyfikacja pakietów może być realizowana z wykorzystaniem różnych kombinacji fragmentów pól nagłówka,

klasyfikacja pakietów jest także określana mianem przełączania L4+.

Każda reguła klasyfikacji pakietu charakteryzowana jest ustalonym zakresem wartości wybranych pól nagłówka, tzn. prefiksem, których spełnienie oznacza przynależność pakietu do określonej klasy ruchu.

Klasyfikację pakietów stosuje się w wielu różnych rozwiązaniach, w tym na potrzeby: filtrowania pakietów w zaporach (ang. *firewall*), agregacji strumieni pakietów w zadaniach tunelowania ruchu, dekompozycji strumieni pakietów, różnicowania ruchu w zadaniach związanych z różnicowaniem jakości usług, równoważenia obciążeń itp.

Do najistotniejszych, pożądaných atrybutów procesu filtracji pakietów w sieciach z różnicowaniem usług należą następujące:

- zasady filtracji muszą mieć zastosowanie do szerokiego zakresu adresów, numerów portów lub protokołów,
- zasady filtracji nie powinny być ograniczone tylko do realizacji dokładnego dopasowania, tzn. powinny umożliwiać klasyfikację pakietów należących do zagregowanych przepływów; jeżeli algorytmy klasyfikacji realizowałyby tylko dokładne dopasowanie, to klasyfikacja musiałaby być poprzedzana wstępnym przetwarzaniem, którego celem byłaby translacja zakresów wartości reguł filtracji w dokładne wartości,
- zasady filtracji muszą mieć przypisane priorytety w celu rozwiązywania potencjalnych konfliktów, wynikających z nakładania się zasad filtracji.

Zadanie klasyfikacji wiąże się ze stosowaniem filtrów, rozumianych jako zestaw reguł, których zadaniem jest ograniczanie pełnego zakresu wartości wybranych pól nagłówków pakietów do pewnych określonych zakresów wartości.

10.8.1. ALGORYTMY KLASYFIKACJI RUCHU

Dla danej reguły lub danego zbioru reguł definiujących atrybuty pakietu lub zawartości pakietu (w ogólnym przypadku jednostki danych), klasyfikacja pakietów jest procesem identyfikacji reguły lub reguł z danego zbioru, którym dany pakiet odpowiada lub do których jest dopasowany.

Tabela 10.2. Przykłady zadań klasyfikacji pakietów

Warstwa	Zastosowanie klasyfikacji	Rodzaj klasyfikacji	Przykładowa reguła
Łącze danych (druga)	komutacja etykiet (<i>switching</i>)	dopasowanie etykiet pakietu i trasy	przełączanie pakietów z zadanymi wartościami etykiet
Sieciowa (trzecia)	przekazywanie pakietów (<i>forwarding</i>)	znajdowanie najdłuższego prefiksu adresu ujścia	przekazywanie pakietów o zadanym zakresie wartości nagłówka ustaloną trasą
Transportowa (czwarta)	identyfikacja przepływów (<i>IntServ</i>)	dopasowanie identyfikatorów pakietów i przepływów	obsługa pakietów o zadanych zawartościach wybranych pól nagłówka z zadanym priorytetem
Transportowa (czwarta)	filtrowanie pakietów (<i>diffServ</i>)	znajdowanie prefiksu lub zakresu prefiksów	odrzućcie pakietów o zadanych wartościach wybranych pól nagłówka
Aplikacji (siódma)	równoważenie obciążeń (<i>load balancing</i>)	skanowanie zawartości nagłówków pakietów	kierowanie pakietów o różnych zakresach wartości wybranych pól nagłówka pakietu do różnych serwerów
Aplikacji (siódma)	detekcja naruszeń (<i>intrusion detection</i>)	skanowanie zawartości pakietów	generowanie alarmu, jeżeli pakiet zawiera zadane wartości wybranych pól

Kategoryzacja reguł klasyfikacji jednostek danych (wiadomości, segmenty, pakiety, ramki, komórki) stosowanych w sieciach może być bardzo różna. Podstawą kategoryzacji mogą być m.in.: cele klasyfikacji, sposoby realizacji klasyfikacji, złożoność obliczeniowa i pamięciowa algorytmów. Jedną z możliwych kategoryzacji metod klasyfikacji jest kategoryzacja ze względu na cele stosowania klasyfikacji jednostek danych:

➤ przekazywanie jednostek danych (ang. *forwarding applications*) – celem klasyfikacji jest zwykle dopasowanie, na podstawie zawartości jednego z pól nagłówka (adresu), jednostki danych do jednej z tras w sieci danej elementami tablicy przełączania lub tras w węzle sieci,

➤ filtrowanie jednostek danych (ang. *filter-based applications*) – celem klasyfikacji jest znajdowanie na podstawie zawartości wielu pól nagłówka jednostki danych klas obsługi ruchu, do której jednostka danych należy,

➤ skanowanie jednostek danych (ang. *content-aware applications*) – celem klasyfikacji jest znajdowanie jednostek danych spełniających warunki definiowane zarówno dla pól nagłówka, jak i pola danych jednostki danych.

Przykładowe zadania klasyfikacji pakietów w zadaniach przekazywania, filtracji i skanowania podano w tabeli 10.2.

Wymienione wyżej zadania klasyfikacji jednostek danych w sieciach pakietowych różnią przede wszystkim długości sekwencji bitów, na podstawie których podejmowane są decyzje oraz liczba i rodzaje reguł klasyfikacji.

10.8.2. MIARY JAKOŚCI ALGORYTMÓW KLASYFIKACJI

Do porównywania działania algorytmów klasyfikacji jednostek danych można stosować różne miary uwzględniające właściwości: klasyfikowanych ciągów bitów, struktur danych tworzonych w celu klasyfikacji, sposobów prowadzenia klasyfikacji itd.

Głównymi miarami algorytmów klasyfikacji są:

➤ złożoność przestrzenna SC (ang. *Space Complexity*), definiowana dla danej reguły klasyfikacji jako górne ograniczenie rozmiaru pamięci niezbędnej do przechowywania bazy danych reguły klasyfikacji i związanych z nią struktur danych – mała złożoność przestrzenna zwiększa możliwości układowej realizacji klasyfikatorów,

➤ złożoność czasowa TC (ang. *Time Complexity*), definiowana jako górne ograniczenie maksymalnej liczby kroków wymaganych do przeprowadzenia klasyfikacji; złożoność czasowa maleje wraz ze wzrostem równoległości wykonywanych operacji i zmniejszaniem liczby stosowanych reguł klasyfikacji – mała złożoność czasowa jest równoważna dużej szybkości klasyfikacji; zwłaszcza w czasie rzeczywistym,

➤ złożoność obliczeniowa PC (ang. *Power Complexity*), definiowana jako górne ograniczenie iloczynu maksymalnej liczby kroków niezbędnych do wykonania klasyfikacji, maksymalnej liczby operacji wykonywanych równolegle w każdym z kroków algorytmu i maksymalnego kosztu pojedynczej operacji,

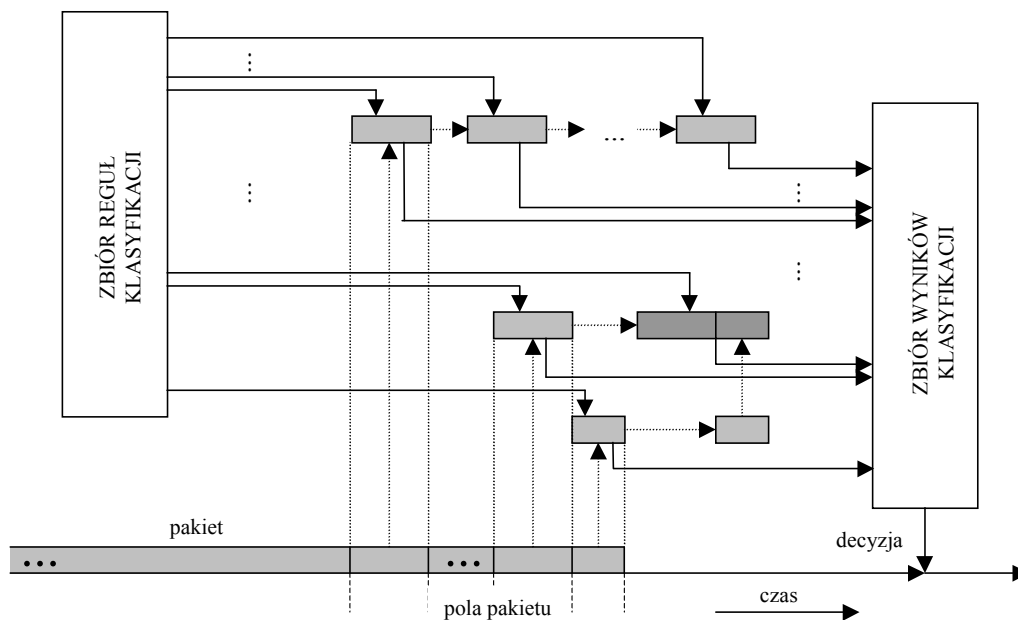
➤ złożoność aktualizacji UC (ang. *Update Complexity*), definiowana jako górne ograniczenie maksymalnej liczby kroków wymaganych do pojedynczych zmian bazy danych reguł klasyfikacji polegających na wprowadzaniu, usuwaniu lub aktualizacji reguł – mała złożoność aktualizacji decyduje o skalowalności i elastyczności zarządzania ruchem w sieci,

➤ złożoność reguł RC (ang. *Rule Complexity*), definiowana dla danej reguły klasyfikacji jako górne ograniczenie iloczynu maksymalnej liczby operacji wykonywanych na każdym z pól reguły klasyfikacji i całkowitej liczby pól uwzględnianych w danej regule klasyfikacji – duża złożoność reguł decyduje o uniwersalności i elastyczności klasyfikatora.

Oczekiwane cechy klasyfikatora jednostek danych w węzłach sieci są następujące:

- mała złożoność przestrzenna gwarantująca efektywność i skalowalność wykorzystania pamięci buforowej,
- mała złożoność czasowa determinująca dużą szybkość klasyfikacji,
- mała złożoność obliczeniowa umożliwiająca dużą skalowalność klasyfikatora,
- mała złożoność aktualizacji dająca możliwość częstych i szybkich zmian reguł i liczby klasyfikacji w bazie reguł,
- duża złożoność reguł pozwalająca na realizację różnych algorytmów klasyfikacji,
- uniwersalność i elastyczność reguł klasyfikacji pozwalających na dobór optymalnych algorytmów klasyfikacji do klasy obsługiwanego ruchu,
- szeregowe przetwarzanie klasyfikowanych jednostek danych pozwalające na wielokrotne klasyfikowanie tych samych jednostek danych według różnych reguł.

Ogólny schemat klasyfikacji pakietów na podstawie zawartości wyróżnionych pól nagłówka pakietu lub pakietu obejmuje działania polegające m.in. na jednokrotnym użyciu reguły kwalifikacji dla wybranego pola nagłówka, sekwencyjnym kwalifikowaniu na podstawie jednego z pól, równoległym klasyfikowaniu na podstawie wielu różnych pól, kwalifikacji na podstawie zawartości łączonych pól (rysunek 10.6).



Rysunek 10.6. Ogólny schemat klasyfikacji pakietów na podstawie zawartości pól pakietu

10.8.3. OGÓLNE SFORMUŁOWANIE ZADANIA KLASYFIKACJI

W ogólnym sformułowaniu zadania klasyfikacja pakietów polega na sprawdzeniu zawartości I pól nagłówka każdego pakietu należących do zbioru wszystkich wyróżnionych pól nagłówka pakietu $Z_p = \{P[1], P[2], \dots, P[i], \dots, P[I]\}$. W niektórych zaawansowanych zastosowaniach klasyfikacji pakietów zbiór analizowanych pól pakietu nie musi być ograniczony wyłącznie do pól nagłówka, poszczególne pola mogą być klasyfikowane wielokrotnie (szeregowo lub równoległe) za pomocą różnych reguł, w procesie klasyfikacji pola mogą być łączone i dzielone itd.

Filtr $F_k(I)$ jest zbiorem I zakresów wartości (prefiksów) $F_k[i]$ dla $i = 1, 2, \dots, i, \dots, I$, po jednym dla każdego spośród I wyróżnionych pól nagłówka ze zbioru Z_p , tzn. $F_k(I) = \{F_k[1], F_k[2], \dots, F_k[i], \dots, F_k[I]\}$. Filtr $F_k(I)$ przekazuje i -te pole nagłówka, jeżeli prefiks $F_k[i]$ ($F_k[i] \in F_k(I)$) przekazuje pole $P[i]$ ($P[i] \in Z_p$). Reguła filtrowania jest filtrem przekazującym pakiet P wtedy, gdy przekazuje wszystkie pola pakietu.

Na przykład: filtr, opisany dwuelementowym zbiorem prefiksów dla adresu źródła i ujścia pakietu:

$$F_k(2) = \{F_k[1], F_k[2]\} = \{[111.222.333.*], [444.555.*.666]\},$$

przekazuje pakiet P_1 o adresie $\{[111.222.333.123], [444.555.456.666]\}$, a nie przekazuje pakietu P_2 o adresie $\{[111.222.333.123], [444.500.456.666]\}$.

Zbiór filtrów ($F(I) = \{F_1(I), F_2(I), \dots, F_k(I), \dots, F_K(I)$, z których każdy zawiera I zakresów wartości (prefiksów), jest bazą danych klasyfikacji pakietów; K jest liczbą wszystkich filtrów.

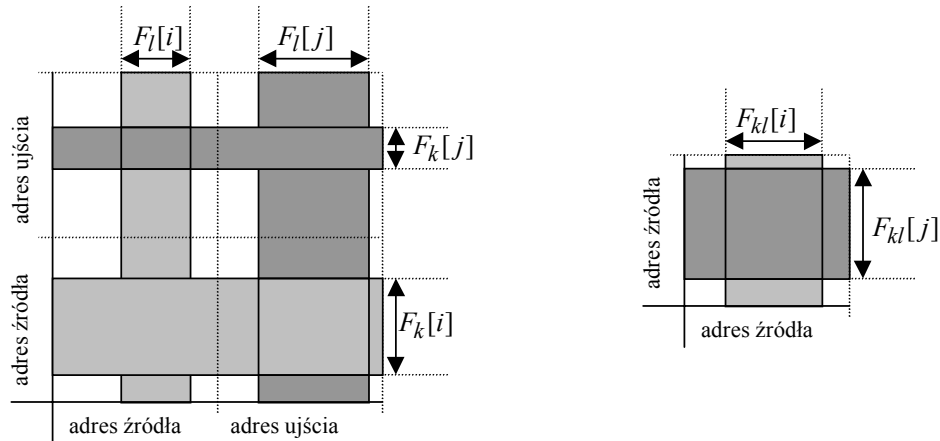
Ze względu na możliwość przekazywania tego samego danego pakietu przez więcej niż jeden z filtrów ze zbioru $F(I)$ możliwe jest porządkowanie filtrów, czyli wybór najlepszego filtra ze zbioru $F(I)$. Jeżeli:

- dwa dowolne filtry $F_k(I)$ i $F_l(I)$ ze zbioru $F(I)$ przekazują pakiet P ,
- dla dowolnego i ($i \in \{1, 2, \dots, I\}$) i -ty prefiks filtra $F_k(I)$, tzn. $F_k[i]$, jest dłuższy od i -tego prefiksu filtra $F_l(I)$, tzn. $F_l[i]$,

to filtr $F_k(I)$ jest bardziej specyficzny od filtra $F_l(I)$ w i -tym polu, z kolei filtr $F_l(I)$ jest mniej specyficzny od filtra $F_k(I)$ w i -tym polu.

Najlepszym filtrem ze zbioru $F(I)$ dla pakietu P jest taki filtr przekazujący pakiet P , który nie jest mniej specyficzny od dowolnego innego filtra ze zbioru $F(I)$ przekazującego pakiet P .

W zbiorze filtrów $F(I)$ możliwe jest wystąpienie sprzeczności filtrów; filtry $F_k(I)$ oraz $F_l(I)$ ($F_k(I), F_l(I) \in F(I)$) są sprzeczne dla pakietu P wtedy, gdy filtr $F_k(I)$ jest bardziej specyficzny od filtra $F_l(I)$ dla pakietu P w dowolnym i -tym polu ($i \in \{1, 2, \dots, I\}$) i jednocześnie mniej specyficzny od filtra $F_l(I)$ dla pakietu P w dowolnym innym j -tym polu ($j \in \{1, 2, \dots, I\}, j \neq i$) (rysunek 10.7a).



$$F_k(2) = \{F_k[i], F_k[j]\}$$

$$F_l(2) = \{F_l[i], F_l[j]\}$$

$$F_k[i] \supset F_l[i]$$

$$F_k[j] \subset F_l[j]$$

a) sprzeczność filtrów

$$F_{kl}(2) = \{F_{kl}[i], F_{kl}[j]\} = \{F_k[i], F_l[j]\}$$

$$F_{kl}[i] = F_k[i] \cup F_l[i] = F_k[i]$$

$$F_{kl}[j] = F_k[j] \cup F_l[j] = F_l[j]$$

b) rozwiązanie sprzeczności filtrów

Rysunek 10.7. Sprzeczność filtrów i rozwiązanie sprzeczności filtrów

Sprzeczność filtrów dla pakietu P oznacza niejednoznaczność w wyborze sposobu obsługi pakietu P , tzn. decyzji o tym, który ze sposobów obsługi pakietu P , odpowiadających sprzecznym filtrom, zostanie zastosowany.

Przykład 10.1

Spośród trzech filtrów $F_k(2)$, $F_l(2)$ i $F_m(2)$ ze zbioru filtrów $F(2)$, zawierających następujące prefiksy:

$$F_k(2) = \{F_k[1], F_k[2]\} = \{[111.*.*], [444.555.*.666]\},$$

$$F_l(2) = \{F_l[1], F_l[2]\} = \{[111.222.*.*], [444.555.*.*]\},$$

$$F_m(2) = \{F_m[1], F_m[2]\} = \{[111.222.333.*], [444.555.*.666]\},$$

i przekazujących pakiet P o adresie $\{[111.222.333.123], [444.555.456.666]\}$:

➤ pierwsze prefiksy filtrów $F_l(2)$ i $F_m(2)$ są dłuższe od pierwszego prefiksu filtra $F_k(2)$, tzn. filtry $F_l(2)$ i $F_m(2)$ są bardziej specyficzne od filtra $F_k(2)$ w polu adresu źródła pakietu,

➤ drugi prefiks filtra $F_l(2)$ jest krótszy od drugich prefiksów filtrów $F_k(2)$ i $F_m(2)$, tzn. filtr $F_l(2)$ jest mniej specyficzny od filtrów $F_k(2)$ i $F_m(2)$ w polu adresu ujścia pakietu,

➤ filtry $F_k(2)$ i $F_l(2)$ są sprzeczne dla danego pakietu P , ponieważ filtr $F_l(2)$ jest bardziej specyficzny od filtra $F_k(2)$ dla pakietu P w polu adresu źródła, a jednocześnie filtr $F_k(2)$ jest bardziej specyficzny od filtra $F_l(2)$ dla pakietu P w polu adresu ujścia,

➤ filtry $F_k(2)$ i $F_m(2)$ nie są sprzeczne dla danego pakietu P , ponieważ filtr $F_m(2)$ jest bardziej specyficzny od filtra $F_k(2)$ dla pakietu P w polu adresu źródła i nie jest mniej specyficzny od filtra $F_k(2)$ dla pakietu P w polu adresu ujścia,

➤ filtry $F_l(2)$ i $F_m(2)$ nie są sprzeczne dla danego pakietu P , ponieważ filtr $F_m(2)$ jest bardziej specyficzny od filtra $F_l(2)$ dla pakietu P zarówno w polu adresu ujścia, jak i w polu adresu źródła,

➤ najlepszym filtrem dla pakietu P , spośród omawianych trzech filtrów $F_k(2)$, $F_l(2)$ oraz $F_m(2)$, jest filtr $F_m(2)$, ponieważ nie jest on mniej specyficzny od pozostałych dwóch filtrów przekazujących pakiet P .

Jednym z rozwiązań sprzeczności filtrów dla pakietu P jest wprowadzenie dodatkowego filtra rozwiązującego sprzeczność adresów $F_{kl}(I)$ (ang. *conflict resolution filter*), którego prefiksy składowe są sumami prefiksów składowych sprzecznych filtrów $F_k(I)$ i $F_l(I)$, tzn.:

$$F_{kl}(I) = \{\dots, F_k[i] \cup F_l[i], \dots, F_k[j] \cup F_l[j], \dots\}.$$

Filtr $F_{kl}[I]$, budowany na podstawie filtrów sprzecznych $F_k(I)$ i $F_l(I)$, jest unikatowym, najlepszym filtrem dla wszystkich możliwych wartości pól adresów źródeł i ujść pakietu P .

Przykład 10.2

Dwa filtry, opisane dwuelementowymi zbiorami prefiksów dla adresu źródła (pierwsze pole) i ujścia pakietu (drugie pole):

$$F_k(2) = \{F_k[1], F_k[2]\} = \{[111.*.*.*], [444.555.*.666]\},$$

$$F_l(2) = \{F_l[1], F_l[2]\} = \{[111.222.*.*], [444.555.*.*]\},$$

przekazują pakiet P o adresie $\{[111.222.321.123], [444.555.623.666]\}$, z tym że:

➤ filtr $F_k(2)$ jest mniej specyficzny od filtra $F_l(2)$ w polu adresu źródła pakietu P ,

➤ filtr $F_k(2)$ jest bardziej specyficzny od filtra $F_l(2)$ w polu adresu ujścia pakietu P , co oznacza, że filtry $F_k(2)$ i $F_l(2)$ są sprzeczne dla pakietu P .

Filtrem, rozwiązującym sprzeczność filtrów $F_k(2)$ i $F_l(2)$ jest filtr:

$$F_{kl}(2) = \{F_{kl}[1], F_{kl}[2]\} = \{[111.222.*.*], [444.555.*.666]\},$$

którego prefiksy są prefiksami sprzecznych filtrów $F_k(2)$ i $F_l(2)$, ustalonymi w następujący sposób:

➤ prefiks filtra $F_{kl}(2)$ dla adresu źródła pakietu jest prefiksem filtra $F_l(2)$, tzn. filtra bardziej specyficznego dla adresu źródła pakietu P ,

➤ prefiks filtra $F_{kl}(2)$ dla adresu ujścia pakietu jest prefiksem filtra $F_k(2)$, tzn. filtra bardziej specyficznego dla adresu ujścia pakietu P .

Zadanie niesprzecznej (dwuwymiarowej) dwupolowej klasyfikacji pakietów formułowane jest następująco:

Dla danego zbioru niesprzecznych dwuwymiarowych (dwupolowych) filtrów:

$$F(2) = \{F_1(2), F_2(2), \dots, F_k(2), \dots, F_K(2)\},$$

oraz dla danego zbioru wszystkich wyróżnionych pól nagłówka pakietu:

$$Z_p = \{P[1], P[2], \dots, P[i], \dots, P[I]\},$$

wyznaczyć najlepszy filtr $F^* \in F(2)$, tzn. taki filtr przekazujący pakiet P , który nie jest mniej specyficzny od dowolnego innego filtra ze zbioru dostępnych filtrów przekazujących pakiet P .

Jeżeli filtr, należący do zbioru wszystkich filtrów, utożsamiać z jedną z wielu jednocześnie obsługiwanych klas ruchu, to znalezienie najlepszego filtra dla pakietu oznacza zakwalifikowanie tego pakietu do jednej z możliwych klas ruchu, tak aby zawartość wybranych pól pakietu odpowiadała charakterystykom klasy ruchu.

Złożoność filtrów zwiększa się wraz ze wzrostem wymiarowości zadania klasyfikacji, uzasadniając w wielu przypadkach stosowanie algorytmów heurystycznych.

10.8.4. ALGORYTMY KLASYFIKACJI

10.8.4.1. DYNAMICZNE BUFOROWANIE

Najprostszym rozwiązaniem zadania klasyfikacji pakietów jest przegląd zupełny wszystkich filtrów. Złożoność pamięciowa takich algorytmów jest $O(K)$, gdzie K jest liczbą wszystkich filtrów. Złożoność obliczeniowa takich algorytmów jest nie do zaakceptowania nawet dla filtrów o średnich rozmiarach.

Techniką, która jest zwykle stosowana w sprzętowych i programowych rozwiązaniach przyspieszających działanie algorytmów przeglądu zupełnego, jest dynamiczne buforowanie (ang. *caching*). Uzasadnieniem stosowania dynamicznego buforowania jest czas trwania każdego przepływu, zawierającego zwykle pakiety o takich samych nagłówkach, wielokrotnie dłuższy od czasu transmisji pakietu; można zakładać, że nadejście pierwszego pakietu nowego przepływu oznacza napływ, w najbliższym czasie, kolejnych pakietów tego przepływu.

Zastosowanie techniki dynamicznego buforowania polega na tym, że informacja o nowych przepływach jest składowana w buforze. Napływ nowego pakietu powoduje sprawdzenie, w pierwszej kolejności, zawartości bufora w celu stwierdzenia, czy pakiet ten należy do jednego z przepływów, o których informacja jest przechowywana w buforze. Jeżeli informacja o poszukiwanym przepływie znajduje się w buforze, to dany pakiet jest obsługiwany tzw. szybką ścieżką (ang. *fast path*). W innym przypadku pakiet jest kierowany do obsługi tzw. wolną ścieżką (ang. *slow path*); nagłówek pakietu jest sprawdzany i klasyfikowany, a informacja o nowym przepływie przekazywana jest do bufora.

Jakość działania dynamicznego buforowania silnie zależy od liczby pakietów w poszczególnych przepływach. Ponadto, jeżeli liczba jednocześnie obsługiwanych przepływów jest większa od rozmiarów bufora, to szybkość klasyfikacji gwałtownie spada.

Rozwiązania z zastosowaniem dynamicznego buforowania są wrażliwe na charakterystyki ruchu i z tego powodu ich działanie nie jest deterministyczne. W najgorszym przypadku wiele pakietów klasyfikuje się z wykorzystaniem tzw. wolnej ścieżki, co może powodować degradację jakości usług. W celu gwarantowania deterministycznych charakterystyk jakości usług działanie algorytmu klasyfikacji musi odbywać się z szybkością równą szybkości napływu pakietów i musi być niezależne od charakterystyk ruchu.

10.8.4.2. PODEJŚCIE GEOMETRYCZNE

Geometryczna interpretacja i -wymiarowego filtra to i -wymiarowa kostka w i -wymiarowej przestrzeni. Rozszerzenie zakresu j -tego wymiaru i -wymiarowej kostki odpowiada w zadaniu klasyfikacji zwiększeniu zakresu przenoszonych (dopasowanych) wartości j -tego pola nagłówka. Nagłówek pakietu odpowiada jednemu z punktów w i -wymiarowej przestrzeni. Z tego względu zadanie wielowymiarowej filtracji odpowiada zadaniu lokalizacji punktu w przestrzeni wielowymiarowej.

W dwuwymiarowej przestrzeni dany zbiór niesprzecznych dwuwymiarowych filtrów:

$$F(2) = F = \{F_1(2), F_2(2), \dots, F_k(2), \dots, F_K(2)\} = \{F_1, F_2, \dots, F_k, \dots, F_K\},$$

jest reprezentowany na płaszczyźnie przez K prostokątów, które mogą na siebie częściowo zachodzić i których wymiary są zakresami wartości dwóch pól poszczególnych filtrów. Każdy z nagłówków pakietów, jednoznacznie charakteryzowany wartościami dwóch pól (np. adresu źródła i ujścia), jest reprezentowany przez jeden punkt na tej samej co filtry płaszczyźnie.

Jeżeli punkt na płaszczyźnie, reprezentujący nagłówek pakietu, należy do prostokąta reprezentującego filtr F_k ($k = 1, 2, \dots, K$), to filtr F_k przenosi dany pakiet, tzn. dany pakiet jest dopasowany do tego filtra F_k .

Realizacja algorytmu geometrycznego (ang. *filtration geometric scheme*) klasyfikacji jest podzielona na dwa etapy:

- etap przygotowania (ang. *preprocessing step*),
- etap klasyfikacji (ang. *classification step*).

Na ogół celem etapu przygotowania algorytmu geometrycznego jest wyznaczenie, dla danego K -elementowego zbioru I -wymiarowych filtrów $F(I) = \{F_1(I), F_2(I), \dots, F_k(I), \dots, F_K(I)\}$:

➤ zbiorów rozłącznych przedziałów zakresu wartości rzutów K -elementowego zbioru I -wymiarowych kostek przestrzennych na poszczególne osie układu współrzędnych,

➤ zbiorów map bitowych dla każdego z przedziałów należących do wcześniej wyznaczonych rozłącznych zbiorów przedziałów zakresu wartości rzutów K -elementowego zbioru I -wymiarowych kostek przestrzennych,

a celem etapu klasyfikacji jest wyznaczenie, dla każdego napływającego pakietu P :

- zbioru I współrzędnych pakietu w I -wymiarowej przestrzeni,
- zbioru pojedynczych przedziałów zakresu wartości rzutów K -elementowego zbioru I -wymiarowych kostek przestrzennych na każdą z poszczególnych osi układu współrzędnych,
- iloczynu map bitowych wszystkich, wyznaczonych wcześniej, podzbiorów przedziałów zakresu wartości rzutów K -elementowego zbioru I -wymiarowych kostek przestrzennych na każdą z poszczególnych osi układu współrzędnych.

Iloczyn map bitowych pozwala na wyznaczenie zbioru filtrów, będących podzbiorem wszystkich filtrów $F(I)$, które przenoszą dany pakiet P . W przypadku uporządkowania zbioru filtrów $F(I)$, iloczyn map bitowych określa hierarchię filtrów przenoszących dany pakiet P .

Działanie algorytmu geometrycznego klasyfikacji, wymagające dużych ilości pamięci, polega na tym, że wyznacza się K najlepszych prefiksów, po jednym dla każdego z K pól filtrów. Dla każdego z filtrów utrzymywana jest wcześniej wyznaczona I -elementowa mapa bitowa. Realizacja zadania klasyfikacji pakietu polega na pobraniu z pamięci K map bitowych, z których każda zawiera I elementów, i wyznaczeniu ich iloczynu logicznego (przecięcia): I -elementowy iloczyn logiczny map bitowych wyznacza podzbiór zbioru wszystkich filtrów przenoszących klasyfikowany pakiet. Złożoność pamięciowa algorytmu geometrycznego jest $O(K^2)$, a jego realizacja każdorazowo wymaga czytania $N \times I$ bitów z pamięci.

W przypadku K -elementowego zbioru dwuwymiarowych filtrów $F = \{F_1, F_2, \dots, F_k, \dots, F_K\}$ zadaniem etapu przygotowania algorytmu geometrycznego jest wyznaczenie dwóch zbiorów rozłącznych przedziałów zakresu wartości rzutów K prostokątów, odpowiadających K filtrom ze zbioru filtrów F , na dwie osie współrzędnych. Wartościami rzutów poszczególnych filtrów F_k ($k = 1, 2, \dots, K$) na osie współrzędnych są odcinki $X(F_k)$ i $Y(F_k)$, których znajomość pozwala wyznaczyć zbiory rozłącznych przedziałów zakresu wartości rzutów wszystkich filtrów ze zbioru filtrów F (rysunek 10.8).

Algorytm geometryczny dla K -elementowego zbioru filtrów dwuwymiarowych:

1. Etap przygotowania:

1.1. Wyznacza się zbiór wszystkich rozłącznych przedziałów zakresu wartości rzutów wszystkich dwuwymiarowych filtrów ze zbioru F na:

➤ oś odciętych (odpowiadającą np. zakresowi wartości adresu źródła – rysunek 10.8):

$$X = \{X_1, X_2, \dots, X_n, \dots, X_{N-1}, X_N\},$$

spełniający warunki:

$$X_1 \cup X_2 \cup \dots \cup X_{N-1} \cup X_N = X(F_1) \cup X(F_2) \cup \dots \cup X(F_{K-1}) \cup X(F_K),$$

$X_i \cap X_j = \emptyset$ dla każdego $i, j \in \{1, 2, \dots, n, \dots, N\}$, $i \neq j$,

$$X_n \cap X(F_k) = \begin{cases} X_n & \text{dla każdego } n \in \{1, 2, \dots, N\} \text{ i } k \in \{1, 2, \dots, K\}, \\ \emptyset & \end{cases}$$

➤ oś rzędnych (odpowiadającą np. zakresowi wartości adresu ujścia – rysunek 10.8):

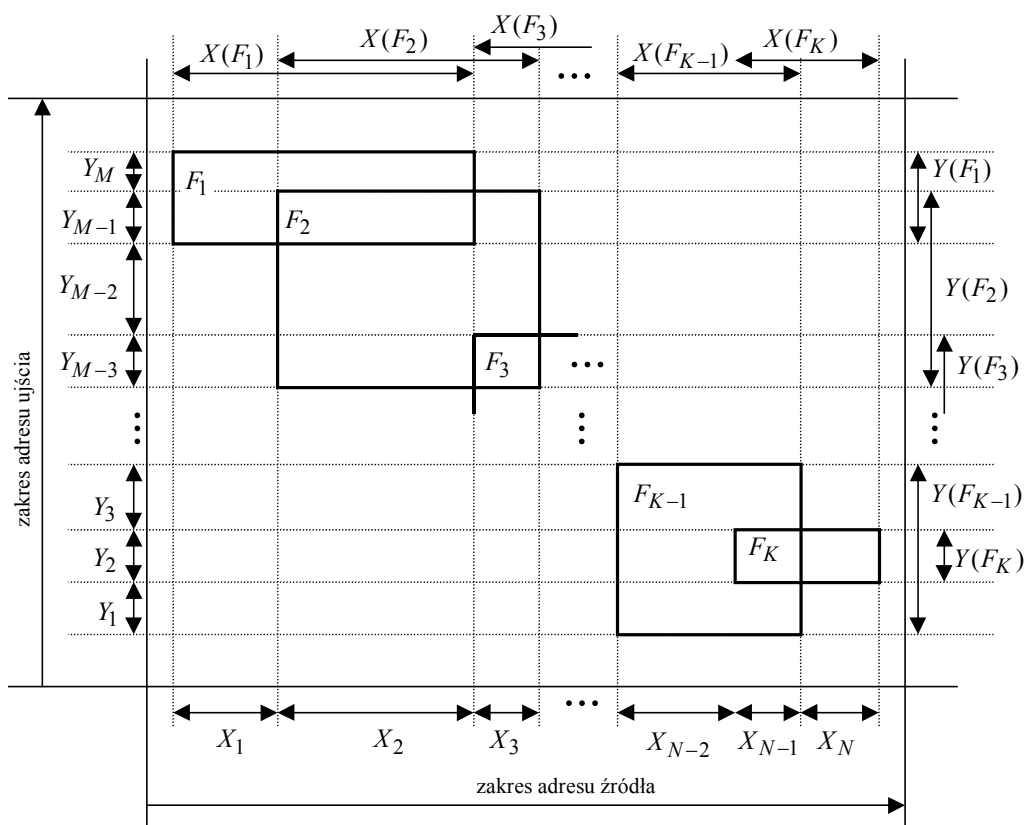
$$Y = \{Y_1, Y_2, \dots, Y_m, \dots, Y_{M-1}, Y_M\},$$

spełniający warunki:

$$Y_1 \cup Y_2 \cup \dots \cup Y_{M-1} \cup Y_M = Y(F_1) \cup Y(F_2) \cup \dots \cup Y(F_{K-1}) \cup Y(F_K),$$

$$Y_u \cap Y_v = \emptyset \text{ dla każdego } u, v \in \{1, 2, \dots, m, \dots, M\}, \quad u \neq v,$$

$$Y_m \cap X(F_k) = \begin{cases} X^m & \text{dla każdego } m \in \{1, 2, \dots, M\} \text{ i } k \in \{1, 2, \dots, K\}. \\ \emptyset & \end{cases}$$



Rysunek 10.8. Algorytm geometryczny klasyfikacji pakietów

1.2. Dla każdego z przedziałów X_n ($X_n \in X$) oraz Y_m ($Y_m \in Y$) wyznaczone są mapy bitowe:

$$B(X_n) = [x_{n1}, x_{n2}, \dots, x_{nk}, \dots, x_{nK}],$$

$$\text{gdzie: } x_{nk} = \begin{cases} 1 & \text{gdy } X_n \cap X(F_k) = X_n \\ 0 & \text{gdy } X_n \cap X(F_k) = \emptyset \end{cases}$$

$$B(Y_m) = [y_{m1}, y_{m2}, \dots, y_{mk}, \dots, y_{mK}],$$

$$\text{gdzie: } y_{mk} = \begin{cases} 1 & \text{gdy } Y_m \cap Y(F_k) = Y_m \\ 0 & \text{gdy } Y_m \cap Y(F_k) = \emptyset \end{cases}$$

Wartość $x_{nk} = 1$ nk -tego elementu mapy bitowej $B(X_n)$ oznacza, że przedział $X(F_k)$, będący rzutem prostokąta reprezentującego dwuwymiarowy filtr F_k , na oś odciętych zawiera w sobie przedział wartości X_n . Jeżeli $x_{nk} = 0$, to przedział wartości $X(F_k)$ i przedział wartości X_n są rozłączne.

2. Etap klasyfikacji:

2.1. W przypadku zbioru filtrów dwuwymiarowych wartości dwóch pól nagłówka każdego napływającego pakietu określają na dwuwymiarowej płaszczyźnie dwie współrzędne (x_p, y_p) . Dla dwóch pól nagłówka klasyfikowanego pakietu, danych wartościami współrzędnych (x_p, y_p) , wyznaczone są przedziały X_n ($X_n \in X$) i Y_m ($Y_m \in Y$), do których należą, odpowiednio, kolejne współrzędne pakietu, tzn. przedziały:

- $X_n = X_n(x_p)$ wtedy i tylko wtedy, gdy $x_p \in X_n$ i $X_n \in X$,
- $Y_m = Y_m(y_p)$ wtedy i tylko wtedy, gdy $y_p \in Y_m$ i $Y_m \in Y$.

Kolejne dwie współrzędne wybranych pól nagłówka pakietu mogą należeć tylko do jednego przedziału ze zbioru X i tylko jednego przedziału ze zbioru Y , ze względu na to, że przedziały tych zbiorów są rozłączne.

Wyznaczenie przedziałów $X_n(x_p)$ i $Y_m(y_p)$ jest równoznaczne z wyznaczeniem map bitowych $B(X_n(x_p))$ i $B(Y_m(y_p))$.

2.2. Dla znanych map bitowych $B(X_n)$ i $B(Y_m)$ obliczany jest ich iloczyn logiczny:

$$B(x_p, y_p) = \text{AND}[B(X_n(x_p)), B(Y_m(y_p))] = [b_1(x_p, y_p), b_2(x_p, y_p), \dots, b_K(x_p, y_p)],$$

gdzie:

$$b_k(x_p, y_p) = \begin{cases} 1 & \text{gdy } x_{nk} \wedge y_{mk} = 1 \\ 0 & \text{gdy } x_{nk} \wedge y_{mk} = 0 \end{cases}$$

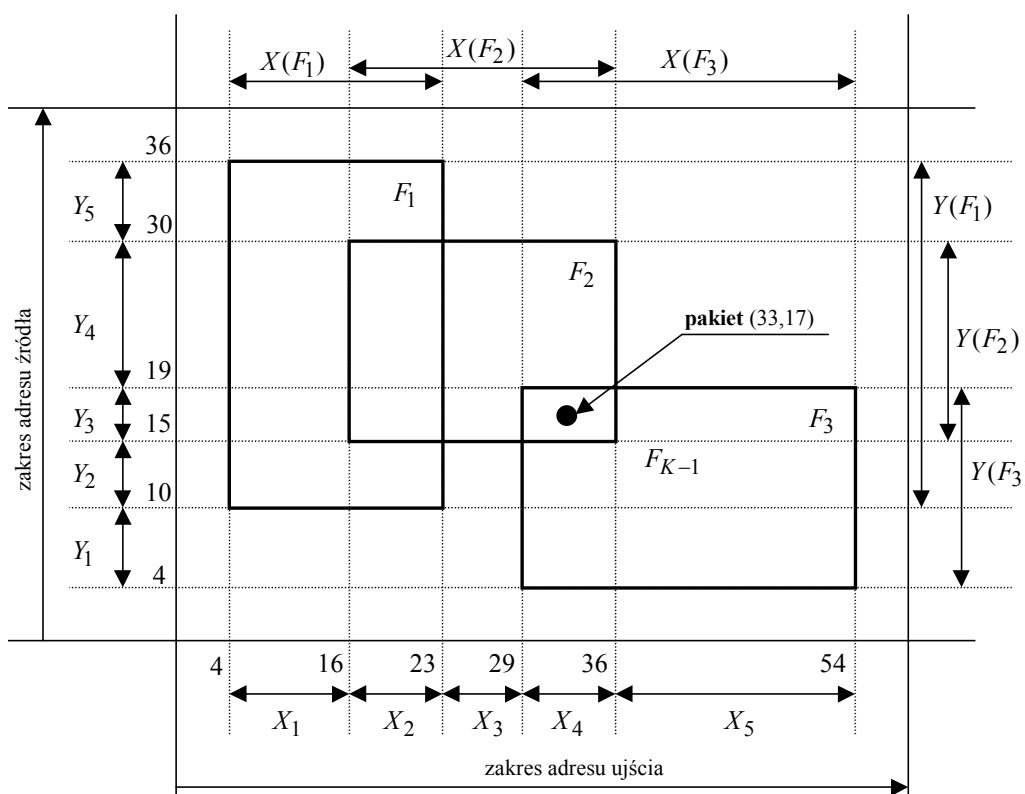
Jeżeli wartość k -tego elementu iloczynu logicznego $B(x_p, y_p)$ map bitowych $B(X_n(x_p))$ i $B(Y_m(y_p))$ jest równa $b_k(x_p, y_p) = 1$, to filtr F_k przenosi pakiet o współrzędnych (x_p, y_p) ; w innym przypadku filtr F_k nie przenosi pakietu o podanych współrzędnych.

Liczba jedynek w iloczynie logicznym $B(x_p, y_p)$ jest liczbą wszystkich filtrów ze zbioru F , które przenoszą pakiet o współrzędnych (x_p, y_p) , natomiast jedynka na k -tej pozycji iloczynu $B(x_p, y_p)$ wskazuje k -ty filtr jako filtr przenoszący dany pakiet.

Jeżeli filtry F_k ($k = 1, 2, \dots, K$), znajdujące się w zbiorze filtrów F , są uporządkowane np. według priorytetów przenoszenia pakietu, to położenie pierwszej jedyinki (licząc od lewej) w iloczynie $B(x_p, y_p)$ wskazuje filtr o najwyższym priorytecie przenoszący dany pakiet.

Przykład 10.3

Przykład działania algorytmu geometrycznego dla trzejelementowego zbioru filtrów dwuwymiarowych.



Rysunek 10.9. Filtracja pakietu przez trzy filtry dwuwymiarowe

Dla trzech filtrów dwuwymiarowych i pakietu, takich że (rysunek 10.9):

➤ filtry są reprezentowane na płaszczyźnie przez prostokąty F_1 , F_2 i F_3 o wymiarach, odpowiednio:

$$F_1\{[4,23], [10,36]\}, F_2\{[16,36], [15,30]\}, F_3\{[29,54], [4,19]\},$$

➤ pakiet jest reprezentowany na płaszczyźnie punktem o współrzędnych (33,17), oraz przy założeniu, że priorytet filtru F_3 jest najwyższy, a priorytet filtru F_2 jest najniższy, działanie algorytmu geometrycznego jest następujące:

1. Etap przygotowania:

1.1. Wyznacza się zbiór wszystkich przedziałów zakresu wartości rzutów trzech dwuwymiarowych filtrów na:

➤ oś odciętych (odpowiadającą adresom źródła), tzn.:

$$X = \{X_1[4,16], X_2[16,23], X_3[23,29], X_4[29,36], X_5[36,54]\},$$

➤ oś rzędnych (odpowiadającą adresom ujścia), tzn.:

$$Y = \{Y_1[4,10], Y_2[10,15], Y_3[15,19], Y_4[19,30], \dots X_5[30,36]\}.$$

1.2. Po uwzględnieniu priorytetów filtrów, tzn. kolejności F_3 , F_1 i F_2 , wyznacza się mapy bitowe dla:

➤ przedziałów X_1, X_2, X_3, X_4, X_5 , tzn.:

$$B(X_1) = [0,1,0], B(X_2) = [0,1,1], B(X_3) = [0,0,1], B(X_4) = [1,0,1], B(X_5) = [1,0,0],$$

oraz Y_1, Y_2, Y_3, Y_4, Y_5 , tzn.

$$B(Y_1) = [1,0,0], B(Y_2) = [1,1,0], B(Y_3) = [1,1,1], B(Y_4) = [0,1,1], B(Y_5) = [0,1,0].$$

2. Etap klasyfikacji

2.1. Z porównania:

➤ wartości współrzędnych adresów źródła i ujścia pakietu, tj. $x_p = 33$ oraz $y_p = 17$,

➤ z zakresami wartości przedziałów zbiorów X oraz Y ,

wynika, że $x_p \in X_4[29,36]$ oraz $y_p \in Y_3[15,19]$.

2.2. Dla map bitowych:

$B(X_4) = [1,0,1]$ oraz $B(Y_3) = [1,1,1]$ oblicza się ich iloczyn logiczny:

$$B(33,17) = AND[B(X_4), B(Y_3)] = [1,0,1] \wedge [1,1,1] = [1,0,1],$$

co oznacza, że:

➤ nagłówek pakietu o adresach źródła i ujścia, danych współrzędnymi (33,17) jest przenoszony przez filtry F_3 i F_2 , a nie jest przenoszony przez filtr F_1 ,

➤ ze względu na priorytety filtrów, tzn. kolejność F_3 , F_1 i F_2 , dopasowanie pakietu do filtru F_3 ma najwyższy priorytet.

10.8.4.3. PRZESTRZEŃ KROTEK

Podstawowa idea przestrzeni krotek wynika ze spostrzeżenia, że jakkolwiek baza danych filtrów może zawierać wiele różnych filtrów o różnych długości prefiksach, to – ze względu na ograniczoną długość pól nagłówek pakietów – liczba różnych długości prefiksów pól nagłówek w filtrach jest względnie mała. Podobnie względnie mała, w porównaniu z liczbą wszystkich możliwych filtrów, jest liczba różnych kombinacji długości prefiksów w filtrach.

Przeźren krotek zawiera wszystkie rozłączne kombinacje różnych długości prefiksów, nie większych niż maksymalna długość poszczególnych pól, w zbiorze wszystkich możliwych filtrów.

Na przykład w routerze sieci szkieletowej może być ponad 500 000 prefiksów adresów ujściowych, ale dla nagłówka pakietu IPv4 liczba różnych długości prefiksów filtrów $w = 32$. Oznacza to, że wszystkie prefiksy filtrów routera pakietów IPv4 mogą być podzielone na 32 grupy, po jednej dla każdej z możliwych długości prefiksu pola adresu ujścia pakietu. Ze względu na to, że wszystkie prefiksy w danej grupie mają tę samą długość, długość ciągu bitów prefiksu może być stosowana do utworzenia schematu skrótu do sprawdzania obecności najdłuższego prefiksu w bazie danych.

Możliwość grupowania prefiksów filtrów o jednakowej długości wykorzystano w koncepcji przestrzeni krotek dla wielowymiarowych filtrów. Ze względu na ograniczoną i względnie małą liczbę różnych długości prefiksów pól filtrów możliwe jest zdefiniowanie krotki dla każdej z możliwych kombinacji długości pól. Ponieważ znana jest długość prefiksu (liczba bitów) każdego z pól nagłówka pakietu każdej tak definiowanej krotki, kaskadowe powiązanie tych liczb bitów pozwala więc utworzyć schemat skrótu, który może być następnie zastosowany do odwzorowania filtrów danej krotki w tablicę skrótu.

Na przykład dwa dwuwymiarowe filtry $F_1 = [100*,1*]$ i $F_2 = [111*,0*]$ są odwzorowywane w krotkę $(3,1)$. Do poszukiwania krotki $(3,1)$ potrzebny jest schemat skrótu, konstruowany przez połączenie trzech bitów pola adresu źródła i jednego bitu pola adresu ujścia. Najlepsze dopasowanie filtra jest możliwe przez próbkowanie każdej krotki. Ze względu na to, że na ogół liczba krotek jest znacznie mniejsza od liczby filtrów, nawet przegląd zupełny przestrzeni krotek wiąże się ze znaczną poprawą efektywności klasyfikacji w porównaniu z przeglądem zupełnym wszystkich filtrów.

10.8.4.4. ALGORYTM RS

Przegląd zupełny dwuwymiarowej przestrzeni krotek wymaga, w najgorszym przypadku, realizacji procedury złożonej z w^2 kroków. Naturalne jest pytanie, czy możliwe jest znalezienie optymalnego filtra bez konieczności przeglądu zupełnego.

Jednym z algorytmów umożliwiających znajdowanie optymalnego filtra bez konieczności przeglądu zupełnego przestrzeni krotek o wymiarach $w \times w$, jest algorytm RS (ang. *Rectangular Search*). Algorytm RS umożliwia znalezienie najlepiej dopasowanego filtra w co najwyżej $2w + 1$ krokach.

W algorytmie RS wykorzystywane są właściwości krotek, wyznaczane w dwóch etapach: przetwarzania wstępnego (ang. *pre-computation*) i znakowania (ang. *markers*).

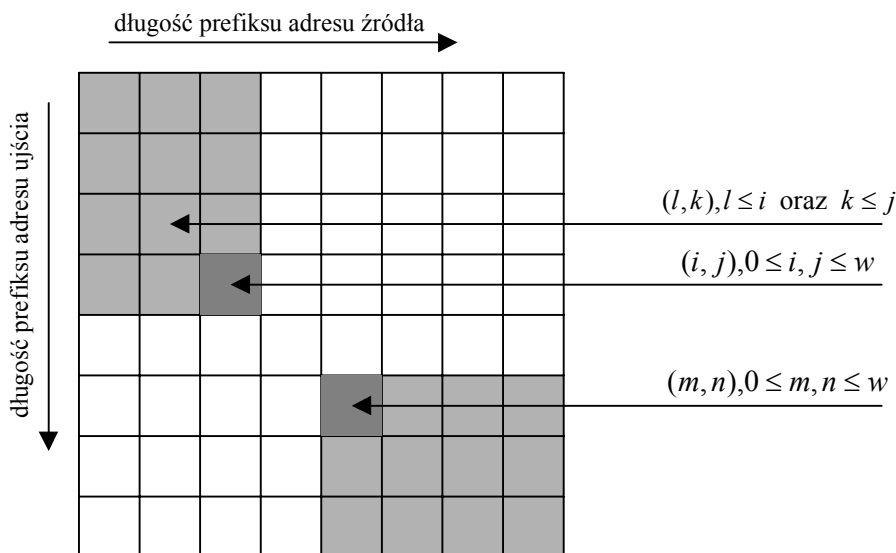
Idea przetwarzania wstępnego w algorytmie RS jest następująca:

➤ filtr F , należący do krotki (i, j) , gdzie $0 \leq i, j \leq w$, charakteryzuje i -elementowy prefiks adresu źródła i j -elementowy prefiks adresu ujścia (np. dwuwymiarowy filtr $F = ([100*], [0111*])$ należy do krotki $(i, j) = (3,4)$),

➤ wszystkie filtry, które należą do krotek (l, k) , gdzie $l \leq i$ oraz $k \leq j$ są filtrami mniej specyficznymi od filtra F , tzn. długości ich prefiksów adresów źródła i ujścia są nie dłuższe od prefiksów adresów źródła i ujścia filtra F (np. filtr $F_1 = ([10*], [01*])$ należący do krotki $(l, k) = (2,2)$ jest mniej specyficzny od filtra $F = ([100*], [0111*])$, należącego do krotki $(i, j) = (3,4)$),

➤ nagłówki wszystkich pakietów przesyłanych przez filtr F , należący do krotki (i, j) , są przesyłane przez filtry należące do wszystkich krotek (l, k) , gdzie $l \leq i$ oraz $k \leq j$ (tzn. wszystkich krotek reprezentowanych przez prostokąt na lewo i w górę od krotki (i, j) na rysunku 10.10),

➤ jeżeli, dla nagłówka danego pakietu P , filtr przesyłający dany pakiet znajduje się w krotce (i, j) , tj. filtr dopasowany do nagłówka pakietu znajduje się w krotce (i, j) , to zbędne jest przeszukiwanie krotek (l, k) , gdzie $l \leq i$ oraz $k \leq j$ (zbędne jest przeszukiwanie krotek należących do prostokąta na lewo i w górę od pola (i, j) na rysunku 10.10).



Rysunek 10.10. Schemat przetwarzania wstępnego i znakowania krotek w algorytmie RS

Idea znakowania w algorytmie RS jest stosowana wtedy, gdy w danej krotce nie zostanie znaleziony filtr dopasowany do nagłówka danego pakietu P . Procedura znakowania jest następująca:

➤ z każdym filtrem F należącym do krotki (i, j) można utożsamiać znakowanie każdej z krotek należących do prostokąta na lewo i w górę od krotki (i, j) ,

➤ znakowanie krotki (l, k) , gdzie $l \leq i$ oraz $k \leq j$, przez filtr F należący do krotki (i, j) polega na tym, że każdej krotce (l, k) przypisana jest część prefiksów filtra F

z krotki (i, j) (np. znakowanie krotki $(l, k) = (2, 1)$ przez filtr $F = ([100*], [0111*])$ należący do krotki $(i, j) = (3, 4)$ jest następujące: $(l, k)_F = (2, 1)_F = ([10*], [0*])$),

➤ filtry należące do każdej z krotek, należących do prostokąta w prawo i w dół od krotki (m, n) , gdzie $0 \leq m, n \leq w$, mogą znakować krotkę (i, j) , tzn. jeżeli nagłówek danego pakietu P jest dopasowany do filtra z krotki (m, n) , to jest także dopasowany do filtra z krotki (i, j) (rysunek 10.10),

➤ jeżeli w krotce (m, n) nie ma filtra przenoszącego nagłówek danego pakietu P , tzn. dopasowanego do nagłówka danego pakietu P , to jest pewne, że nie ma filtra o takich właściwościach w krotkach należących do prostokąta na prawo i w dół od krotki (m, n) (rysunek 10.10).

Zakładając, że długości adresu źródła i ujścia są jednakowe i równe w , to z przedstawionego opisu procedur przetwarzania wstępnego i znakowania wynika algorytm wyznaczania optymalnego filtra dopasowanego do nagłówka danego pakietu P :

➤ procedurę sprawdzania krotek, dla nagłówka danego pakietu P , rozpoczyna się od lewej dolnej krotki przestrzeni krotek, tj. krotki $(1, w)$,

➤ jeżeli w krotce $(1, w)$ znajduje się filtr dopasowany do nagłówka danego pakietu P , to kolejną przeszukiwaną krotką jest krotka $(2, w)$; wynika to z właściwości procedury przetwarzania wstępnego; każda krotka $(1, k)$, gdzie $k \leq w$, jest znakowana filtrem z krotki $(1, w)$, dopasowanym do nagłówka danego pakietu P (przejście z krotki $(1, w)$ do krotki $(2, w)$ oznacza przejście do kolejnej kolumny w prawo na rysunku 10.10),

➤ jeżeli w krotce $(1, w)$ nie ma filtra dopasowanego do nagłówka danego pakietu P , to kolejną przeszukiwaną krotką jest krotka $(1, w - 1)$; wynika to z właściwości procedury znakowania; krotka $(1, w)$ nie jest znakowana żadnym filtrem z dowolnej krotki (k, w) , gdzie $k \leq w$, dopasowanym do nagłówka danego pakietu P (przejście z krotki $(1, w)$ do krotki $(1, w - 1)$ oznacza przejście do kolejnego wiersza w górę na rysunku 10.10),

➤ każda kolejna procedura sprawdzania krotki kończy się zmianą krotki odpowiadającej zwiększeniem numeru kolumny o jeden lub zmniejszeniem numeru wiersza o jeden,

➤ w najgorszym przypadku optymalny filtr jest znajdowany po sprawdzeniu $2w - 1$ krotek.

Idea wstępnego przetwarzania i znakowania w algorytmie RS może być podstawą budowy algorytmu binarnego przeszukiwania zbioru filtrów w celu znalezienia najlepiej dopasowanego filtra.

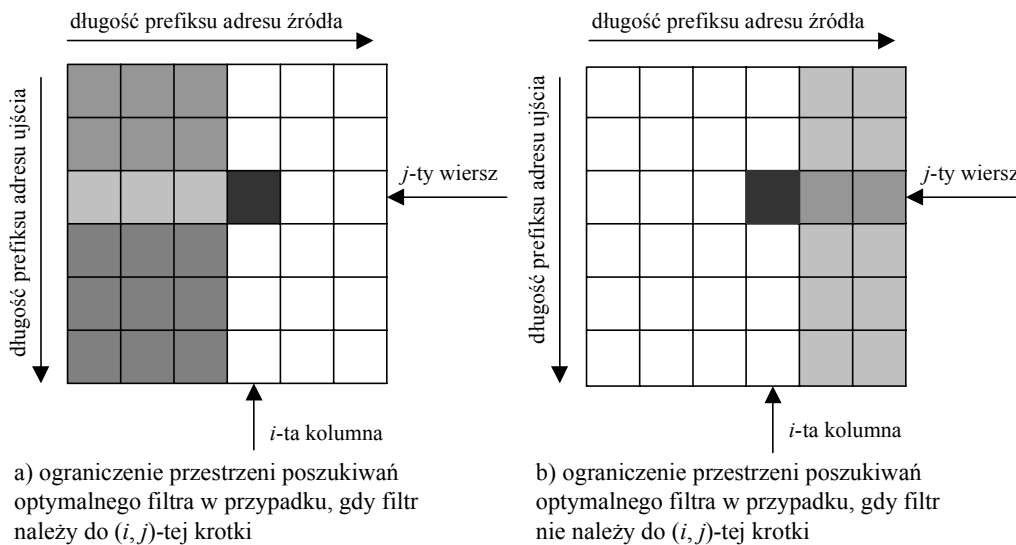
10.8.4.5. ALGORYTM BS

W algorytmie BS (ang. *Binary Search*) wykorzystywane są właściwości krotek, wynikające z procedur przetwarzania wstępnego i znakowania, określonych dla algorytmu RS oraz właściwości filtrów niesprzecznych (ang. *conflict free filters*).

Działanie algorytmu BS opiera się na następujących właściwościach przestrzeni krotek i filtrów (rysunek 10.11):

➤ i -ta kolumna krotek w przestrzeni krotek jest zbiorem tych wszystkich krotek, dla których długość prefiksu adresu źródła wynosi dokładnie i bitów; i -ta kolumna jest zbiorem krotek (i, j) , gdzie $0 \leq j \leq w$,

➤ każdy z filtrów, należących do krotki (i, j) , znakuje wszystkie krotki (k, j) , gdzie $k \leq i$, tzn. wszystkie krotki w tym samym wierszu i w kolumnach na lewo od i -tej kolumny (np. filtr z krotki $(i, j) = (4, 3)$ znakuje krotki z wiersza trzeciego i pierwszych trzech kolumn na rysunku 10.11a),



Rysunek 10.11. Schemat przetwarzania wstępnego i znakowania krotek w algorytmie BS

Jeżeli:

➤ istnieje filtr należący do krotki (i, j) i dopasowany do nagłówka danego pakietu P , lub

➤ krotka (i, j) jest znakowana filtrem z krotki (k, j) , gdzie $i < k \leq w$, dopasowanym do nagłówka danego pakietu P ,

to wszystkie kolumny $k < i$ przestrzeni krotek (tzn. kolumny na lewo od i -tej kolumny na rysunku 10.10a) są eliminowane z procesu przeszukiwania przestrzeni krotek i . Wynika to z właściwości procedury przetwarzania wstępnego i procedury znakowania dla niesprzecznych filtrów.

Jeżeli:

➤ nie istnieje filtr należący do krotki (i, j) i dopasowany do nagłówka danego pakietu P , lub

➤ krotka (i, j) nie jest znakowana filtrem z krotki (k, j) , gdzie $i < k \leq w$, dopasowanym do nagłówka danego pakietu P ,

to wszystkie krotki (k, j) (tzn. krotki j -tego wiersza na prawo od i -tej kolumny na rysunku 10.11b) są eliminowane, co oznacza, że wszystkie kolumny $k > i$ przestrzeni krotek (tzn. kolumny na prawo od i -tej kolumny na rysunku 10.11b) mogą być wyeliminowane z przestrzeni krotek w w kolejnych krokach; wynika to z procedury znakowania dla niesprzecznych filtrów.

Wymienione właściwości procesu przeszukiwania wynikają z następujących właściwości niesprzecznych filtrów:

➤ jeżeli należący do krotki (i, j) niespreczny filtr F jest dopasowany do nagłówka danego pakietu P , to do nagłówka tego pakietu są dopasowane także filtry (k, l) , gdzie $0 \leq k \leq i$ oraz $0 \leq l \leq j$ (tzn. filtry należące do krotek w prostokącie w lewo i w górę od krotki (i, j) na rysunku 10.11a) – znalezienie filtra F o podanych właściwościach w krotce (i, j) pozwala wyeliminować z dalszych poszukiwań optymalnego filtra krotki (k, l) , gdzie $0 \leq k \leq i$ oraz $0 \leq l \leq j$,

➤ istnienie należącego do krotki (i, j) niesprzecznego filtra F dopasowanego do nagłówka danego pakietu P wyklucza istnienie filtra F' dopasowanego do tego pakietu P i należącego do dowolnej z krotek (k, l) , gdzie $0 \leq k \leq i$ oraz $0 \leq j < l \leq w$ (tzn. filtry należące do krotek w prostokącie w lewo i w dół od krotki (i, j) na rysunku 10.11a); w takim przypadku filtr F' jest mniej specyficzny od filtra F dla pierwszego z pól i bardziej specyficzny od filtra F dla drugiego z pól, co przeczy niesprzeczności filtrów – znalezienie filtra F o podanych właściwościach w krotce (i, j) i założenie o niesprzeczności filtrów pozwala wyeliminować z dalszych poszukiwań optymalnego filtra krotki (k, l) , gdzie $0 \leq k \leq i$ oraz $0 \leq j < l \leq w$,

➤ jeżeli żaden z filtrów należących do krotki (i, j) nie jest dopasowany do nagłówka danego pakietu P oraz krotka (i, j) nie jest znakowana żadnym filtrem z krotek (k, j) , gdzie $0 \leq i < k \leq w$ (tzn. krotki j -tego wiersza na prawo od krotki (i, j) na rysunku 10.11b), to znaczy, że w krotkach tych nie ma filtra dopasowanego do drugiego pola nagłówka danego pakietu P – nieznanie filtra dopasowanego do nagłówka pakietu P ani znakowania krotki filtrem dopasowanym do drugiego pola nagłówka pakietu P pozwala wyeliminować z dalszych poszukiwań optymalnego filtra krotki j -tego wiersza,

➤ jeżeli w żadnej z krotek i -tej kolumny nie ma filtra dopasowanego do nagłówka pakietu P , to po wykonaniu w kroków procedury przeszukiwania przestrzeni krotek można wyeliminować z dalszego poszukiwania optymalnego filtra wszystkie krotki z kolumn (i, k) , gdzie $k \geq i$ (tzn. z prostokąta na prawo od i -tej kolumny na rysunku 10.11b).

Z podanych właściwości kolumny krotek oraz właściwości procedur wstępnego przetwarzania i znakowania niesprzecznych filtrów wynika prosty algorytm przeszukiwania przestrzeni krotek. Przeszukanie wszystkich w krotek i -tej, środkowej kolumny przestrzeni krotek pozwala na wyeliminowanie w w krokach połowy przestrzeni krotek. W najgorszym przypadku przeszukiwanie w krotek kolumny musi być powtarzane $\log w$ razy. Górne ograniczenie złożoności obliczeniowej procesu przeszukiwania całej przestrzeni krotek jest $O(w \log w)$.

Dalsza redukcja złożoności obliczeniowej procesu przeszukiwania jest możliwa pod warunkiem skorzystania z właściwości binarnych adresów pól filtrów.

10.9. WYZNACZANIE ADRESU WĘZŁA TRASY

Wyznaczanie adresu kolejnego węzła trasy na podstawie przeglądu zawartości tablicy tras (ang. *route table lookup*), utrzymywanej w węźle sieci pakietowej, jest jednym z zadań rozwiązywanych w każdym węźle trasy w procesie przekazywania pakietów, na który składają się zadania klasyfikacji przepływów i pakietów, wyznaczenia adresu węzła trasy oraz kolejkowania i szeregowania pakietów.

10.9.1. KLASYFIKACJA W JEDNYM WYMIARZE

Wyznaczanie adresu kolejnego węzła trasy jest szczególnym przypadkiem ogólnego zadania klasyfikacji pakietów. Na ogół rozwiązanie zadania wyznaczenia adresu kolejnego węzła trasy w danym węźle sieci polega na znalezieniu w bazie danych, zawierających wszystkie filtry adresów ujść routera, takiego filtra dopasowanego do nagłówka danego pakietu, którego prefiks jest najdłuższy spośród prefiksów wszystkich filtrów dopasowanych do nagłówka danego pakietu. Filtr spełniający te warunki jest filtrem optymalnym dla pola adresu ujścia w nagłówku danego pakietu.

Do rozwiązywania zadań wyznaczenia adresów węzłów, tj. wyznaczenia najdłuższych prefiksów filtrów przenoszących dany pakiet, stosuje się różne algorytmy, w tym m.in. oparte na znajdowaniu najdłuższych ścieżek drzew, tj. grafów spójnych i acyklicznych. Złożoność obliczeniowa i pamięciowa takich algorytmów jest proporcjonalna do liczby bitów pola adresu ujścia w nagłówkach pakietów.

Ze względu na dużą złożoność obliczeniową i pamięciową algorytmów opartych na przetwarzaniu grafów, w praktycznych rozwiązaniach routerów implementowane są rozwiązania wykorzystujące buforowanie dynamiczne i zorientowane na obsługę połączeń (ang. *connection caches*) lub tras (ang. *route caches*). Istotnym ograniczeniem takich rozwiązań jest nieprzewidywalność rozwiązań oraz zależność efektywności rozwiązań zarówno od charakterystyk ruchu, jak i liczby połączeń lub tras ruchu obsługiwanych w węźle.

Opracowano wiele algorytmów, w których łączone są właściwości rozwiązań polegających na przeszukiwaniu binarnym (ang. *binary search*) oraz rozwiązań wykorzystujących tablice skrótów (ang. *hash table*). Główna idea takich rozwiązań polega na tym, że w routerze utrzymuje się wcześniej wyznaczoną tablicę skrótów prefiksów dla każdej z możliwych długości prefiksu.

Znajdowanie najdłuższego prefiksu filtra, dopasowanego do danego pakietu, jest procedurą dwuetapową:

- etap pierwszy polega na sprawdzaniu zbioru wszystkich podzbiorów filtrów o ustalonej długości prefiksów; złożoność obliczeniowa i pamięciowa procedury sprawdzania podzbiorów filtrów jest determinowana właściwościami filtrów i wynikających stąd algorytmów,

- etap drugi polega na przeszukiwaniu danego podzbioru filtrów, charakteryzowanego długością prefiksów, w celu znalezienia filtra dopasowanego do nagłówka danego pakietu.

Na ogół rozwiązanie zadania wyboru optymalnego filtra dla nagłówka danego pakietu polega na znalezieniu najlepszego filtra w każdym z podzbiorów zbioru wszystkich filtrów, a następnie na wyborze filtra optymalnego spośród najlepszych filtrów w poszczególnych podzbiórach. Złożoność obliczeniowa takiej procedury wynosi $O(\log w)$, gdzie w jest długością pola adresu ujścia nagłówka pakietu.

10.9.2. KLASYFIKACJA W DWÓCH WYMIARACH

Klasyfikacja dwuwymiarowa ma coraz większe znaczenie w sieciach, w których z ustanowieniem trasy w sieci wiąże się gwarancja jakości usług.

Implementacja protokołu rezerwacji zasobów (np. RVSP), oznaczająca m.in. gwarantowanie stałej pojemności i górnego ograniczenia opóźnienia w połączeniu pomiędzy systemami końcowymi, jest praktycznie niemożliwa w bardzo szybkich sieciach szkieletowych, w których liczba przepływów jednocześnie aktywnych jest bardzo duża.

Jednym z rozwiązań omawianego problemu jest transfer zagregowanego przepływu specyficznym przygotowaną i utrzymywaną trasą kierunkową ruchu (ang. *traffic-engineered path*). Trasa taka jest definiowana nie tylko adresem ujścia pakietu, ale także adresem źródła pakietu.

Protokół rezerwacji zasobów może być stosowany do definiowania zasad agregacji i routingu. Podstawową w takich zadaniach jest dwuwymiarowa klasyfikacja pakietów, według której wyznacza się zarówno adres następnego węzła na trasie pomiędzy systemami końcowymi, jak i alokację zasobów dla każdego z pakietów w funkcji adresów źródła i ujścia.

Dwuwymiarowa klasyfikacja może być także wykorzystywana do przekazywania pakietów w trybie jeden do wielu (ang. *multicasting*), wymagającego sprawdzania zarówno adresu źródła, jak i adresów grup adresów węzłów końcowych.

10.10. MODELE RÓŻNICOWANIA USŁUG

Ruch strumieniowy, generowany m.in. przez aplikacje multimedialne, charakteryzuje się połączeniowością, dużą wrażliwością na fluktuację opóźnienia i przepustowości sieci oraz – z uwagi na redundancję informacji – w pewnym zakresie odpornością na straty jednostek danych. Obsługa ruchu strumieniowego, wykazującego znaczną zmienność szybkości (ang. *rate variability*), zależną od użytego sposobu kodowania, wymaga dostępności dużych przepustowości sieci oraz gwarancji jakości usług sieci.

W sieciach ze względnym różnicowaniem usług jedyną gwarancją dotyczącą jakości usług jest gwarancja, że jakość usług w wyższej klasie usług jest wyższa od jakości usług dostarczanych w klasie niższej. W takim przypadku jakość usług w poszczególnych klasach usług nie jest definiowana wartościami parametrów ilościowych. Ilość i jakość usług dostarczanych użytkownikom sieci w danej klasie usług zależy od bieżącego obciążenia sieci.

Do oceny schematów względnego różnicowania usług stosuje się dwa kryteria:

➤ sterowalności (ang. *controllability*) – możliwe jest różnicowanie klas usług niezależnie od obciążenia poszczególnych klas ruchem generowanym przez użytkowników sieci,

➤ przewidywalności (ang. *predictability*) – różnicowanie klas usług jest przewidywalne lub inaczej spójne (ang. *consistent*), tzn. jakość usług w klasach wyższych jest wyższa (lub co najmniej nie niższa) od jakości w klasach niższych niezależnie od długości przedziału czasu obserwacji i zmian w obciążeniu sieci.

10.10.1. SCHEMATY WZGLĘDNEGO RÓŻNICOWANIA USŁUG

Różne schematy względnego różnicowania usług charakteryzuje różna sterowalność i przewidywalność.

Względne różnicowanie usług może być realizowane z wykorzystaniem m.in. następujących schematów: całkowitego priorytetowania, różnicowania cen i różnicowania pojemności.

Schemat całkowitego priorytetowania SP (ang. *Strict Prioritization*), zgodnie z którym obsługa ruchu z wyższej klasy usług poprzedza obsługę ruchu z klas niższych, pozwala na uporządkowanie klas usług ze względu na oferowaną jakość usług. Do wad tego schematu różnicowania usług zalicza się następujące jego właściwości:

➤ w przypadku dużego obciążenia wyższej klasy usług możliwe jest zawłaszczanie zasobów przez wyższą klasę kosztem niższej klasy, co powoduje tzw. efekt wygłodzenia ruchu (ang. *traffic starvation*), tzn. wydłużenie czasu oczekiwania na obsługę w niższych klasach usług,

➤ brak sterowalności – zakresy jakości usług dostarczanych przez poszczególne klasy zależą od obciążenia poszczególnych klas usług.

Przykładem schematu różnicowania cen (ang. *Price Differentiation*) jest schemat PMP (ang. *Paris Metro Pricing*), w którym do względnego różnicowania usług stosuje się nie mechanizmy przekazywania ruchu (ang. *forwarding mechanism*), ale ceny. Działanie tego schematu opiera się na założeniu, że konsekwencją wyższych cen jest mniejsze obciążenie w wyższych klasach usług. Wadami mechanizmu różnicowania usług na podstawie ceny są:

➤ zależność efektywności schematu od czasu trwania różnicowania jakości, tzn. efektywność schematu zwiększa się wraz z wydłużaniem przedziałów czasu obserwacji skutków działania schematu i wraz ze zmniejszaniem częstotliwości modyfikacji taryf,

➤ nieprzewidywalność jakości usług, tj. zależność jakości usług od obciążenia klasy; przeciążenie klasy, powodowane m.in. wzrostem aktywności „bogaty” użytkowników może prowadzić do tego, że jakość obsługi w klasie wyższej jest niższa od jakości obsługi dostarczanej w klasie niższej.

Działanie schematu różnicowania pojemności (ang. *capacity differentiation*), jako schematu względnego różnicowania usług, polega na tym, że ilość zasobów (pojem-

ności łączy lub pojemności pamięci buforowych) alokowana do wyższych klas usług jest większa i proporcjonalna do oczekiwanego obciążenia klasy. Żądany sposób alokacji zasobów można uzyskać m.in. przez odpowiednią konfigurację sprawiedliwej dyscypliny szeregowania WFQ. Wadą omawianego schematu różnicowania jest to, że wyższe klasy usług często mogą dostarczać niższej jakości usługi niż klasy niższe, naruszając w ten sposób podstawową właściwość względnego różnicowania jakości usług. Dzieje się tak dlatego, że jakość usług w każdej z klas usług zależy od krótkotrwałych zależności pomiędzy alokowanymi zasobami i napływającym ruchem. Większe obciążenie wyższych klas usług przez długotrwałe połączenia (ruch) może prowadzić do majoryzowania zasobów i tym samym degradacji jakości obsługi połączeń (ruchu) krótkotrwałych, mierzonych np. opóźnieniem wnoszonym w danej klasie jakości usług. Możliwość wystąpienia omawianych różnic jakości usług powoduje, że schemat różnicowania pojemności nie spełnia warunku przewidywalności.

10.10.2. MODEL PROPORCJONALNEGO RÓŻNICOWANIA USŁUG

Model proporcjonalnego różnicowania usług PDM (ang. *Proportional Differentiation Model*) zaprojektowano jako schemat względnego różnicowania klas usług spełniający warunki sterowalności i przewidywalności.

Działanie schematu PDM opiera się na założeniu, że wartości miar jakości pewnej klasy usług są proporcjonalne do wartości wybranych parametrów różnicowania jakości usług (ang. *quality differentiation parameters*). Za podstawowe miary jakości klas usług w sieciach pakietowych przyjmuje się opóźnienia kolejkowania i straty pakietów; wyższa jakość usług oznacza mniejsze opóźnienia kolejkowania i mniejsze prawdopodobieństwo straty pakietów.

Przy założeniu, że:

➤ $q_i(t, t + \tau)$ jest miarą jakości dostarczanych przez i -tą ($i = 1, 2, \dots, N$) klasę usług w przedziale czasu $(t, t + \tau)$,

➤ τ ($\tau > 0$) jest długością względnie krótkiego przedziału czasu monitorowania jakości usług,

jakość usług jest proporcjonalnie względnie różnicowana, jeżeli w każdym przedziale czasu monitorowania i dla każdej pary (i, j) klas usług, gdzie $i, j = 1, 2, \dots, N$ oraz $i \neq j$, miary jakości usług $q_i(t, t + \tau)$ i $q_j(t, t + \tau)$ spełniają warunki:

$$\frac{q_i(t, t + \tau)}{q_j(t, t + \tau)} = \frac{c_i}{c_j},$$

gdzie $c_1 < c_2 < \dots < c_N$ są wartościami wybranego parametru różnicowania usług.

Podana definicja względnego różnicowania usług stanowi, że – niezależnie od zmian jakości usług powodowanych zmianą obciążenia klasy usług – wartość współczynnika różnicowania usług pomiędzy klasami pozostaje stała i jest sterowalna. Co

więcej, ponieważ różnicowanie dotyczy względnie krótkich przedziałów czasu monitorowania, względne uporządkowanie klas usług jest więc spójne i przewidywalne z punktu widzenia użytkownika sieci.

W modelu proporcjonalnego różnicowania usług mogą być stosowane różne miary jakości usług, w tym m.in. opóźnienie kolejkowania i prawdopodobieństwo strat pakietów. Podstawienie:

$$q_i(t, t + \tau) = \frac{1}{d_i(t, t + \tau)},$$

gdzie $d_i(t, t + \tau)$ jest średnim opóźnieniem kolejkowania pakietów i -tej klasy usług, których obsługa została zakończona w przedziale czasu $(t, t + \tau)$, prowadzi do modelu proporcjonalnego różnicowania opóźnienia PDD (ang. *Proportional Delay Differentiation*).

W modelu PDD dla wszystkich par (i, j) klas usług, gdzie $i, j = 1, 2, \dots, N$ i $i \neq j$, oraz dla każdego przedziału czasu monitorowania $(t, t + \tau)$, w którym zdefiniowane są wartości $d_i(t, t + \tau)$ i $d_j(t, t + \tau)$, spełnione są warunki:

$$\frac{d_i(t, t + \tau)}{d_j(t, t + \tau)} = \frac{\delta_i}{\delta_j},$$

gdzie $\delta_1 < \delta_2 < \dots < \delta_N$ są wartościami parametru różnicowania opóźnienia (ang. *delay differentiation parameters*). Wartość $d_i(t, t + \tau)$ nie jest zdefiniowana wtedy, gdy w przedziale czasu monitorowania $(t, t + \tau)$ nie została zakończona obsługa żadnego pakietu.

Z kolei podstawienie:

$$q_i(t, t + \tau) = \frac{1}{l_i(t, t + \tau)},$$

gdzie $l_i(t, t + \tau)$ jest częścią wszystkich pakietów i -tej klasy usług, których obsługa została wstrzymana w chwili t lub które napłynęły w przedziale czasu monitorowania $(t, t + \tau)$ i zostały odrzucone w tym samym przedziale czasu monitorowania, prowadzi do modelu proporcjonalnego różnicowania szybkości strat PLRD (ang. *Proportional Loss Rate Differentiation*).

W modelu PLRD dla wszystkich par (i, j) klas usług, gdzie $i, j = 1, 2, \dots, N$ i $i \neq j$, oraz dla każdego przedziału czasu monitorowania $(t, t + \tau)$, wartości $l_i(t, t + \tau)$ i $l_j(t, t + \tau)$, spełniają warunki:

$$\frac{l_i(t, t + \tau)}{l_j(t, t + \tau)} = \frac{\sigma_i}{\sigma_j},$$

gdzie $\sigma_1 < \sigma_2 < \dots < \sigma_N$ jest ciągiem uporządkowanych wartości parametru różnicowania szybkości strat (ang. *loss rate differentiation parameters*).

Model proporcjonalnego różnicowania usług PDM jest sterowalny z punktu widzenia operatora sieci przez dobór wartości parametrów różnicowania jakości usług. Model ten zapewnia także przewidywalność. Jeżeli długość przedziału czasu monitorowania τ jest wystarczająco mała, to jakość usług wyższej klasy jest lepsza od jakości usług klas niższych. Ograniczeniem modelu PDM jest wykonalność usług (ang. *feasibility*) o jakości zadanej wartościami parametrów różnicowania w określonych warunkach; zwłaszcza w przypadku dużego obciążenia klasy.

10.11. ZARZĄDZANIE ZASOBAMI

Realizacja sieci z różnicowaniem usług wiąże się z wymaganiami, bez spełnienia których nie jest możliwe różnicowanie klas ruchu i klas spedycji ruchu. Wymaganiem takim jest m.in. konieczność obsługi pakietów na wejściach routerów z szybkością równą szybkości napływu pakietów z sieci, tzn. nieorganizowanie kolejek wejściowych. Spełnienie omawianego wymagania oznacza m.in. zmianę architektur routerów na takie, których nieprzetwarzanie w routerze nie jest poprzedzane czasem oczekiwania na przetwarzanie. Innym krytycznym zasobem jest pojemność łączy wychodzących z węzłów, efektywność wykorzystania których zależy zarówno od pojemności wyjściowych pamięci buforowych, jak i dyscyplin szeregowania implementowanych na wyjściach pamięci buforowych.

10.11.1. ZARZĄDZANIE POJEMNOŚCIĄ ŁĄCZY WYJŚCIOWYCH

Zarządzanie pojemnością pamięci buforowych i pojemnością łączy wyjściowych jest zadaniem łącznym. Dyscyplina szeregowania, która dynamicznie alokuje pojemność łączy wyjściowego, jest tylko warunkiem koniecznym dostępu połączenia do pojemności łączy. Jeżeli w danym połączeniu nie ma pakietów oczekujących na transmisję, to możliwość transmisji, udostępniania przez dyscyplinę szeregowania, jest tracona (niektóre dyscypliny szeregowania przenoszą niewykorzystane alokacje pojemności w postaci tzw. kredytu). Dla połączeń adaptacyjnych szczególnie ważne jest, aby udostępnianie możliwości transmisji przez dyscyplinę szeregowania było skorelowane z odpowiednią alokacją pojemności pamięci buforowych. Korelacja dyscypliny szeregowania i procedur alokacji pojemności pamięci buforowej dla danego połączenia jest gwarancją, że dane połączenie wykorzysta możliwości dostępu do pojemności łączy i tym samym osiągnie oczekiwany poziom współdzielenia pojemności łączy wyjściowego.

Wartość oczekiwana poziomu współdzielenia pojemności łączy wyjściowego (ang. *desired level of link sharing*) określa dostęp poszczególnych klas ruchu do pojemności łączy; wartość ta jest przypisywana poszczególnym klasom ruchu administracyjnie lub w postaci rezerwacji. Wartość oczekiwana poziomu współdzielenia pojemności łączy dla danej klasy ruchu w sieci z różnicowaniem usług odpowiada gwarantowanej pojemności łączy dla danego przepływu w sieci z integracją usług.

Działanie węzła (routera) sieci z różnicowaniem usług jest następujące:

1. Algorytm filtracji (klasyfikowania) pakietów przypisuje każdy pakiet do jednej z klas ruchu.
2. Algorytm wyznaczania adresu kolejnego węzła na trasie danej klasy ruchu wyznacza kolejkę wyjściową, organizowaną w pamięci buforowej.
3. Dyscyplina szeregowania, implementowana na wejściu łącza wyjściowego, wyznacza kolejność transmisji pakietów poszczególnych klas ruchu z uwagi na:
 - wartości oczekiwane poziomu współdzielenia pojemności łącza klas ruchu,
 - zajętość pojemności pamięci buforowej przez pakiety różnych klas ruchu,
 - przebieg ostatniej sekwencji obsługi kolejek wyjściowych klas ruchu, zależnej od implementowanej dyscypliny szeregowania.

10.11.2. PRZEPIŁYWY NIEADAPTACYJNE I ADAPTACYJNE

W koncepcji sieci z różnicowaniem usług wyróżnia się dwa rodzaje przepływów, których charakterystyki wpływają na wybór dyscypliny szeregowania i działanie dyscypliny szeregowania: połączenia nieadaptacyjne i adaptacyjne.

Przepływy nieadaptacyjne (ang. *nonadaptive flow*) to takie, w których szybkość napływu jednostek danych nie zależy od poprzednich decyzji dyscypliny szeregowania. Przykładem takich przepływów są strumienie jednostek danych generowanych przez aplikacje czasu rzeczywistego, takie jak np. wideo. Żądana dla takich przepływów pojemność łącza wyjściowego jest ustalana przez mechanizmy udostępniania zasobów lub przez procedury sygnalizacyjne (ang. *signalling procedure*), do których zalicza się także procedury rezerwacji zasobów. Różnica pomiędzy udostępnianiem a sygnalizacją polega na tym, że udostępnianie dotyczy zwykle przepływu zagregowanego, efektem sygnalizacji natomiast jest rezerwacja zasobów dla przepływu, tj. na najniższym poziomie granulacji. Niezależnie od tego, czy zasoby (w tym przypadku pojemność łącza wyjściowego) są udostępniane czy też rezerwowane, w każdym z tych przypadków niezbędne jest implementowanie mechanizmów szeregowania, wymuszających alokację udostępnionych lub zarezerwowanych zasobów pakietom odpowiednich przepływów.

Przepływy adaptacyjne (ang. *adaptive flow*) to takie, w których szybkość napływu jednostek danych zależy od poprzednich decyzji dyscypliny szeregowania. Przykładem takich przepływów są przepływy, w których do sterowania szybkością napływu jednostek danych stosuje się różne mechanizmy reakcyjne, np. mechanizm okna. Przepływy takie charakteryzuje zmiana szybkości napływu jednostek danych w zależności od ilości dostępnych zasobów.

Ze względu na to, że zasoby w sieciach z różnicowaniem usług są z definicji współdzielone przez klasy ruchu o różnych charakterystykach, istotne znaczenie dla efektywności obsługi różnych klas ruchu oraz efektywności wykorzystania zasobów ma sposób zarządzania zasobami; w omawianym przypadku zarządzania pojemnością łączy wyjściowych z węzła.

10.11.3. HIERARCHICZNE ZARZĄDZANIE POJEMNOŚCIĄ ŁĄCZY

Jednym z rozwiązań pozwalających na efektywne zarządzanie pojemnością łączy wyjściowych, współdzieloną przez przepływy nieadaptacyjne i adaptacyjne, jest hierarchiczny podział pojemności łączy HLS (ang. *Hierarchical Link Sharing*).

Hierarchiczny podział pojemności jest przykładem schematu elastycznego zarządzania zasobami; całkowitą pojemność łączy dzieli się na klasy i każda z klas jest definiowana jako przepływ zagregowany lub zbiór przepływów o pewnych specyficznych właściwościach.

Hierarchiczne zarządzanie pojemnością polega na tym, że pojemności, otrzymane w wyniku podziału pojemności pomiędzy różne domeny administracyjne, mogą być rozdzielane dalej pomiędzy klasy ruchu z danej domeny administracyjnej. Wzrost szczegółowości podziału pojemności odpowiada zwiększeniu liczby poziomów hierarchii; zgodnie z tym na najniższym poziomie hierarchii pojemność jest rozdzielana pomiędzy indywidualne przepływy.

Uzupełnienie hierarchicznego schematu zarządzania rozdziałem pojemności przez sprawiedliwe dyscypliny szeregowania pozwala na efektywne wykorzystanie pojemności. Sprawiedliwa dyscyplina szeregowania gwarantuje przekazywanie pojemności pomiędzy przepływami należącymi do jednej domeny administracyjnej na jednym z wyróżnionych poziomów hierarchii.

W razie niewykorzystania pojemności przez przepływy należące do jednej domeny administracyjnej na jednym z poziomów hierarchii, niewykorzystana pojemność może być przekazana do wykorzystania na wyższy poziom hierarchii podziału pojemności.

Zaletą hierarchicznego zarządzania pojemnością w sieci z różnicowaniem usług jest możliwość organizowania wielu różnych sposobów obsługi przepływów za pomocą ograniczonych zbiorów struktur danych i algorytmów. Uniwersalność omawianego schematu zarządzania zasobami polega na tym, że pozwala, w skrajnych przypadkach, na efektywną organizację:

- zarówno obsługi dużej liczby izolowanych indywidualnych przepływów, tzn. realizację koncepcji sieci z integracją usług (ang. *per-flow queueing*),

- jak i małej liczby przepływów zagregowanych, tzn. realizację koncepcji sieci z różnicowaniem usług (ang. *per-aggregate queueing*).

Źródłem korzyści, wynikających z możliwości alokacji pojemności łączy dla indywidualnych przepływów, jest rozłączne kolejkowanie jednostek danych każdego z przepływów i pełna separacja każdego z przepływów.

W przypadku niektórych przepływów połączenie korzyści wynikających z rozłącznego kolejkowania i separacji przepływów oraz możliwości zastosowania sposobu buforowania dopasowanego do charakterystyk przepływu może prowadzić do znacznego wzrostu jakości obsługi przepływu w sieci.

10.11.4. SKALOWALNOŚĆ OBSŁUGI INDYWIDUALNYCH PRZEPEŁYWÓW

Przyjęcie schematu zarządzania pojemnością, zgodnie z którym obsługiwane są indywidualne przepływy, jest tożsame z rozłącznym kolejkowaniem jednostek danych każdego z przepływów. W takim przypadku w routerze muszą być gromadzone i utrzymywane dane o wszystkich aktywnych przepływach (przebieg jest definiowany jako przepływ aktywny, gdy czas międzyzgłoszeniowy pomiędzy kolejnymi jednostkami danych przepływu, identyfikowanego tymi samymi adresami źródła i ujścia, jest krótszy niż 60 s) dotyczące w szczególności:

- wartości współczynników wagowych dla poszczególnych przepływów,
- jakości usług dostarczanych poszczególnym przepływom.

10.11.4.1. DYSCYPLINY WAŻONEGO SPRAWIEDLIWEGO KOLEJKOWANIA

Znajomość wartości wymienionych parametrów indywidualnych przepływów jest warunkiem wystarczającym i koniecznym implementacji algorytmów sprawiedliwego rozdziału pojemności łącza wyjściowego pomiędzy indywidualne przepływy, tzn. implementacji idealnych i nieidealnych dyscyplin sprawiedliwego kolejkowania.

10.11.4.1.1. ALGORYTMY IDEALNEGO SZEREGOWANIA

Algorytmy idealnego szeregowania IWFQ (ang. *Ideal Weighted Fair Queueing*) emulują przepływowy model ruchu w ograniczeniach modelu dyskretnego. W modelu przepływowym ruchu wszystkie jednostki danych są z założenia nieskończenie podzielne, a obsługa wszystkich przepływów z zaległościami odbywa się z szybkością proporcjonalną do wartości współczynników wagowych poszczególnych przepływów.

Idealne szeregowanie oznacza, że nieprzerwana obsługa danego przepływu z zaległościami jest tożsama z przydzielaniem przepływowi pojemności, której ilość zależy tylko od wartości współczynnika wagowego tego przepływu oraz wartości współczynników wagowych wszystkich pozostałych przepływów z zaległościami. W takim przypadku nie jest konieczna znajomość stanów żadnego z przepływów, który nie jest przepływem z zaległościami. Ponieważ liczba przepływów w routerze jest ograniczona przez system zarządzania pamięcią buforową, tzn. przez maksymalną liczbę organizowanych kolejek, liczba przepływów, dla których konieczne jest utrzymywanie informacji o ich stanie, jest zatem równa maksymalnej liczbie kolejek.

10.11.4.1.2. ALGORYTMY DYSKRETNEGO SZEREGOWANIA

Realizacja algorytmów nieidealnego (dyskretnego) ważonego sprawiedliwego kolejkowania NWFQ (ang. *Non-ideal Weighted Fair Queueing*), emulujących idealne ważne sprawiedliwe kolejkowanie, wymaga utrzymywania informacji zarówno o stanach emulowanych przepływach idealnych, jak i o stanach przepływów dyskretnych.

nych. Wynika to z tego, że rzeczywiste szeregowanie pakietów nie emuluje modelu przepływowego, ponieważ pakiety różnych przepływów nie mogą być obsługiwane jednocześnie. W najgorszym przypadku zbiory przepływów z zaległościami w modelu przepływowym i w modelu dyskretnym mogą być całkowicie różne. Oznacza to, że utrzymywana dla algorytmu dyskretnego szeregowania maksymalna ilość informacji o stanie przepływów jest dwa razy większa od ilości informacji utrzymywanej dla algorytmu idealnego szeregowania.

Ważone sprawiedliwe kolejkowanie pakietów w routerze wymaga zastosowania mechanizmu uzyskiwania informacji o stanie idealnego ważonego kolejkowania na podstawie informacji o stanie dyskretnego ważonego kolejkowania. Informacja ta jest wykorzystywana do szeregowania pakietów w systemie dyskretnym. Jednym z rozwiązań jest stosowanie funkcji tzw. potencjału przepływu (ang. *flow potential*) i potencjału systemu (ang. *system potential*). Algorytm szeregowania przypisuje każdemu przepływowi wartość potencjału przepływu, równą wartości znormalizowanej jakości obsługi przepływu wtedy, gdy przepływ ten jest przepływem aktywnym. Wartość znormalizowanej jakości obsługi przepływu jest wartością ilorazu szybkości przepływu i pojemności alokowanej dla tego przepływu. Potencjał systemu obsługi jest wykorzystywany do śledzenia wartości znormalizowanej jakości usług dla wszystkich przepływów w przedziale czasu aktywności systemu. Aktualna wartość potencjału systemu jest różnicą maksymalnego potencjału systemu i sumy wartości potencjału przepływów aktywnych. Jeżeli dany przepływ jest na nowo przepływem z zaległościami, to aktualna wartość potencjału danego przepływu jest obliczana jako maksimum dwóch wartości: poprzedniej wartości potencjału tego przepływu i aktualnej wartości potencjału systemu.

Utrzymywana w routerze ilość informacji o stanie przepływów zależy od algorytmu dyskretnego ważonego sprawiedliwego kolejkowania. Jeżeli np. potencjał systemu jest zawsze większy od potencjału każdego z przepływów bez zaległości, to niezbędne jest utrzymywanie informacji tylko o przepływach z zaległościami.

Ilość informacji gromadzonych i utrzymywanych dla algorytmu dyskretnego ważonego sprawiedliwego kolejkowania zależy od maksymalnej liczby przepływów aktywnych w odpowiednim algorytmie idealnego kolejkowania; w najgorszym przypadku liczba przepływów aktywnych w algorytmie idealnego kolejkowania jest równa liczbie jednostek danych (pakietów) w systemie. Jeżeli np. router obsługuje łącze OC-3 (STS-3) o przepływności 155,52 Mb/s, średnia długość pakietu jest równa 2048 bitów, a szybkość obsługi kolejek jest równa 5 pakietom na sekundę, to maksymalna liczba przepływów z zaległościami jest mniejsza od 16 000.

10.11.4.2. JAKOŚĆ OBSŁUGI SEPAROWANYCH PRZEPLÝWÓW

W przypadku sieci z różnicowaniem usług rezultatem filtracji pakietów w routerze jest różnicowanie klas ruchu, tzn. przyporządkowanie do zaledwie kilku przepływów nielicznych klas ruchu. Ograniczona liczba klas ruchu, w porównaniu z liczbą indywi-

dualnych przepływów, oznacza także ograniczenie liczby kolejek, których organizacja jest warunkiem koniecznym izolowania różnych klas ruchu i różnicowania jakości usług dla poszczególnych klas ruchu. Efektywna obsługa ograniczonej liczby, w porównaniu z liczbą indywidualnych przepływów, separowanych zagregowanych przepływów wymaga gromadzenia i utrzymywania danych o klasach ruchu w postaci: wartości współczynników wagowych dla klas ruchu i jakości usług dostarczanych klasom ruchu.

Stosowanie systemów zarządzania zasobami w routerze pozwala na połączenie korzyści wynikających z separacji przepływów, a stosowanie odpowiedniego buforowania przepływów może powodować wzrost jakości obsługi przepływów.

Przykładem klasy ruchu, dla którego separacja ruchu i buforowanie przepływów w routerze może powodować wzrost jakości usług sieci, jest ruch TCP w sieci Internet. Przepływy TCP mają m.in. następujące cechy charakterystyczne:

1. Wybuchowość ruchu (ang. *traffic burstiness*).

Przyczyną wybuchowości przepływów TCP jest stosowanie metody okna jako techniki sterowania przepływem; szybkość transmisji zależy od szybkości napływu potwierdzeń oraz sposobów zmian rozmiaru okna, zwłaszcza w czasie rozpoczynania generowania przepływu (podwajanie rozmiaru okna).

2. Wrażliwość na straty (ang. *loss sensitivity*).

Przepływ TCP jest bardzo wrażliwy na straty, ze względu na to, że każda strata jest traktowana jako strata wywołana przeciążeniem (ang. *congestion loss*) i powoduje znaczną redukcję szybkości generowanego w źródle przepływu. Dla stałej wartości współczynnika strat spadek przepustowości następuje z odwrotnością kwadratu iloczynu pojemności i opóźnienia. Wrażliwość na straty przepływu TCP jest dodatkowo zwiększona przez spowolnienie napływu potwierdzeń. Przyrost wrażliwości jest proporcjonalny do wzrostu wartości ilorazu opóźnienia potwierdzeń do opóźnienia pakietów.

3. Nierównomierność obciążenia połączenia (ang. *unfairness*).

Wrażliwość przepływu TCP na wartości parametrów mechanizmu okna powoduje, że zmiany szybkości przepływu mają charakter oscylacyjny (tzw. efekt kompresji potwierdzeń).

Wymagania stawiane systemom zarządzania zasobami w routerach sieci z różnicowaniem usług, a dotyczące obsługi przepływów TCP, są formułowane następująco:

- utrzymywanie minimalnej różnicy pomiędzy rzeczywistą i oczekiwaną wartością poziomu współdzielenia pojemności łącza wyjściowego z routera dla względnie długich przepływów TCP (przepływ TCP jest długi, gdy czas transmisji przepływu jest znacznie dłuższy od iloczynu opóźnienia i rzeczywistej pojemności łącza wykorzystywanej przez przepływ),
- zapewnienie małego opóźnienia dla krótkich przepływów TCP,
- zabezpieczenie jakości obsługi przepływów TCP przed wpływem ruchu innego niż TCP, tzn. zabezpieczenie przed zawłaszczaniem zasobów,
- zapewnienie maksymalnego poziomu wykorzystania zasobów; zwłaszcza pojemności łącza wyjściowego.

Ze względu na cechy charakterystyczne przepływów TCP oraz wymagania dotyczące obsługi przepływów TCP w sieci z różnicowaniem usług system zarządzania zasobami w routerze powinien spełniać następujące funkcje:

1. W zakresie klasyfikacji pakietów przepływów:
 - separowanie długich i krótkich przepływów TCP w celu minimalizacji opóźnienia dla krótkich przepływów.
2. W zakresie zarządzania pojemnością pamięci buforowej:
 - buforowanie paczek pakietów generowanych w przepływach TCP, zwłaszcza w czasie rozpoczynania przepływu, w celu minimalizacji wartości współczynnika strat,
 - buforowanie nierównomierności ruchu o charakterze oscylacyjnym w celu zwiększenia efektywności wykorzystania pojemności łącza.
3. W zakresie zarządzania pojemnością łącza wyjściowego (dyscyplina szeregowania):
 - gwarancja minimalnego opóźnienia dla krótkich przepływów TCP,
 - gwarancja maksymalnej przepustowości dla długich przepływów TCP,
 - sprawiedliwość rozdziału zasobów pomiędzy współzawodniczące przepływy TCP,
 - zabezpieczenie przez zawłaszczaniem pojemności przez przepływy inne niż przepływy TCP.

Realizacja w routerze wymienionych wyżej funkcji oznacza, że algorytmy zarządzania zasobami oraz wartości ich parametrów zależą od właściwości przepływów TCP. Obsługa przepływów TCP przez algorytmy sprawiedliwego ważonego kolejkowania, implementowane w routerze, daje możliwość sprawiedliwej obsługi tych przepływów. Jeżeli przepływ TCP nie jest przepływem z zaległościami (np. z powodu zbyt małego rozmiaru okna), to możliwość sprawiedliwej obsługi przepływu nie oznacza sprawiedliwego współdzielenia pojemności łącza wyjściowego. Warunkiem sprawiedliwego (lub niesprawiedliwego, ale sterowanego) ważonego współdzielenia pojemności łącza przez przepływ TCP są wzajemnie dopasowane: system zarządzania pojemnością pamięci buforowej i dyscyplina szeregowania.

Zadaniem systemu zarządzania pojemnością pamięci buforowej, dopasowanego do dyscypliny sprawiedliwego ważonego szeregowania, dla każdego z przepływów TCP jest rezerwacja nominalnej pojemności pamięci buforowej.

Wartość nominalnej pojemności dla zbioru przepływów TCP jest zbiorem wartości pojemności proporcjonalnych do wag poszczególnych przepływów, tzn. nominalnych długości kolejek dla poszczególnych przepływów. Gdy przekroczona jest założona wartość progowa całkowitej zajętości nominalnej pojemności pamięci buforowej i gdy napływa nowy pakiet, wówczas stosowany jest mechanizm wypychania (ang. *push-out*), tzn. napływ nowego pakietu wymusza usunięcie jednego z pakietów znajdującego się w pamięci buforowej.

Parametrem mechanizmu wypychania jest kryterium wyboru pakietu do usunięcia z pamięci buforowej. Zwykle kryterium tym jest wartość współczynnika przepełnienia nominalnych długości kolejek; z pamięci usuwany jest pakiet należący do przepływu, dla którego współczynnik przepełnienia kolejki jest największy.

10.11.5. ZARZĄDZANIE USŁUGAMI

Dostarczanie zróżnicowanej jakości usług w sieci z różnicowaniem usług wymaga, oprócz procedur obsługi transferu danych i utrzymywania tras, także odpowiednich procedur zarządzania.

Funkcje zarządzania w sieciach z różnicowaniem usług spełniają brokery pasma BB (ang. *Bandwidth Broker*). Do głównych ich zadań zalicza się sterowanie dostępem, zarządzanie zasobami sieci oraz konfigurowanie routerów dostępowych i brzegowych sieci.

Gdy działania wszystkich brokerów pasma BB w sieci są koordynowane, można mówić o istnieniu w sieci zintegrowanego środowiska zarządzania usługami (ang. *integrated policy services environment*).

Podstawowymi komponentami zintegrowanego środowiska zarządzania usługami w sieci z różnicowaniem usług są moduły realizujące zadania związane z zarządzaniem:

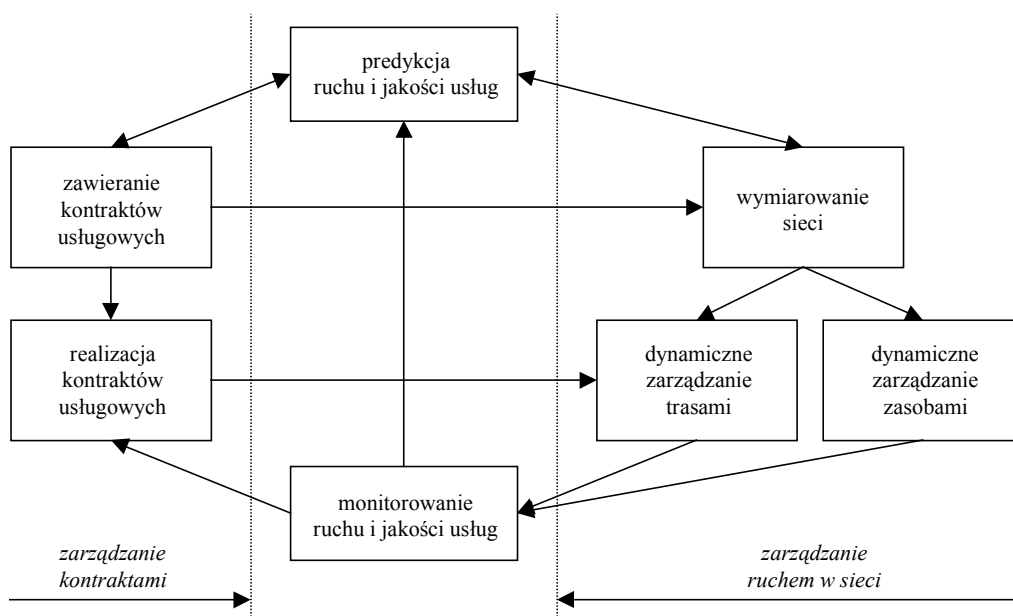
- kontraktami usługowymi SLA (ang. *Service Level Agreement*),
- ruchem w sieci (ang. *traffic engineering*).

Podział na zarządzanie kontraktami i inżynierią ruchu jest uzasadniony zależnościami pomiędzy tymi grupami zadań. Zawierane kontrakty decydują o ilości ruchu obsługiwanego w sieci, a zarządzanie ruchem w sieci danymi zasobami decyduje o jakości dostarczanych usług. Zmiana jakości i ilości kontraktów ma wpływ na jakość usług, stopień obciążenia zasobów ma natomiast wpływ na proces zawierania kontraktów (rysunek 10.12).

Wśród zadań dotyczących zarządzania kontraktami SLA można wyróżnić dwie podgrupy zadań związanych z procesem przyjmowania żądań dostępu do sieci, tzn. negocjowaniem kontraktów SLA oraz procesem dynamicznego zarządzania przepływami, tzn. realizacją zawartych kontraktów SLA. Negocjowanie kontraktów obejmuje wszystkie działania, których celem jest uzgodnienie warunków przyjęcia do obsługi w sieci żądania użytkownika. Dynamiczne zarządzanie przepływami obejmuje działania związane z realizacją ustalonych kontraktów i dotyczy m.in. sterowania dostępem, znakowania, kształtowania ruchu. Zarządzanie kontraktami jest realizowane w węzłach brzegowych sieci.

Zadania inżynierii ruchu (zarządzania ruchem) są realizowane na dwóch poziomach: na poziomie wyższym odbywa się wymiarowanie sieci (ang. *network dimensioning*), a na poziomie niższym realizowane jest sterowanie zasobami przydzielonymi przez moduł udostępniania zasobów. Zadaniem modułu wymiarowania sieci jest translacja (odwzorowanie) żądań użytkowników na fizyczne zasoby sieci i taka ich alokacja, która pozwala na obsługę oczekiwanej ilości ruchu. Zadaniem sterowania udostępnionymi zasobami jest zarządzanie wcześniej udostępnionymi zasobami w trakcie obsługi przepływów w czasie rzeczywistym, polegające na przeciwdziałaniu następstwom fluktuacji ruchu. Sterowanie udostępnionymi zasobami obejmuje zadania dynamicznego sterowania trasami i zasobami. Dynamiczne sterowanie trasami – realizowane w węzłach brzegowych sieci – dotyczy wyboru tras i ewentualnego

równoważenia obciążenia ruchem alternatywnych tras, sterowanie zasobami – realizowane w każdym z węzłów sieci – dotyczy natomiast zarządzania pamięciami buforowymi i szeregowania ruchu.



Rysunek 10.12. Ogólna architektura zarządzania w koncepcji usług zróżnicowanych

Warunkiem współdziałania modułów zarządzania kontraktami i zarządzania ruchem w sieci jest wymiana informacji przekazywanych z modułów zarządzania ruchem do modułów zarządzania kontraktami. Takie oddziaływanie modułów ma charakter sprzężenia zwrotnego – na podstawie informacji o jakości dostarczanych usług i wykorzystaniu zasobów możliwe są zmiany zasad zawierania kontraktów. Wymianę informacji pomiędzy modułami zarządzania kontraktami i zarządzania ruchem umożliwiają moduły monitorowania i przewidywania ruchu i jakości usług w sieci.

10.12. PORÓWNANIE KONCEPCJI SIECI Z INTEGRACJĄ USŁUG I SIECI Z RÓŻNICOWANIEM USŁUG

Dogodną formą przedstawiania koncepcji usług zróżnicowanych (model usług zróżnicowanych) jest jej porównanie z koncepcją usług zintegrowanych (model usług zintegrowanych) ze względu na sposób, szczegółowość i zakres organizacji ruchu w sieci oraz mechanizmy sterowania tym ruchem (tabela 10.3).

Tabela 10.3. Porównanie koncepcji sieci z integracją usług i sieci z różnicowaniem usług

Kryterium	Koncepcja sieci z integracją usług	Koncepcja sieci z różnicowaniem usług
1	2	3
Zorientowanie koncepcji świadczenia usług	połączenie: węzeł źródłowy–węzeł ujściowy (węzły ujściowe)	domena adresowa (sieci, podsieci, ...)
Poziom agregacji obsługiwanego ruchu (granulacja poziomu różnicowania jakości usług)	brak agregacji; przedmiotem obsługi jest ruch na najniższym możliwym poziomie granulacji ruchu	agregacja przepływów (najniższy poziom granulacji) na poziomie definiowanym liczbą i rodzajem klas spedycji ruchu
Przedmiot sterowania	indywidualny przepływ (obiektywna granulacja ruchu)	klasa spedycji (subiektywna granulacja ruchu)
Rodzaj sterowania	prewencyjne i reakcyjne	reakcyjne i prewencyjne
Alokacja zasobów	alokacja zasobów dla indywidualnych przepływów	alokacja zasobów dla klas ruchu, tzn. sum zaregrowanych przepływów
Udostępnianie zasobów	udostępnianie zasobów przez rezerwację zasobów	udostępnianie zasobów przez priorytowanie przepływów w różnych klasach ruchu
Definiowanie i realizacja jakości usług	bezpośrednio – jakość usługi jest definiowana przez podanie wartości miar jakości dostarczanych użytkownikowi dla przepływu typu: węzeł źródłowy–węzeł ujściowy	pośrednio – wynikowa jakość usługi jest obserwacją efektów działania algorytmu będącego złożeniem sposobu obsługi klasy spedycji w węzłach wewnętrznych i procedury sterowania dostępem na brzegu sieci
Gwarancja jakości usług dla indywidualnych przepływów	bezpośrednio – rezerwacja zasobów dla indywidualnych przepływów	pośrednio – priorytowanie przepływów w poszczególnych klasach spedycji i udostępnianie zasobów klasom spedycji
Wymuszanie zakontraktowanej jakości usług	w każdym węźle trasy źródło–ujście; dopasowanie przepływu do możliwości obsługi przez zarezerwowane zasoby	na brzegu sieci – sterowanie obciążeniem sieci (sterowanie dostępem) i kształtowanie profilu ruchu
Sposób różnicowania jakości usług	gwarancja jakości w sensie deterministycznym lub statystycznym	bezwzględne lub względne charakterystyki jakości usług
Liczba różnych mechanizmów sterowania	równa liczbie różnych definiowanych usług	równa liczbie wyróżnianych klas spedycji
Lokalizacja mechanizmów sterowania w sieci	zadania klasyfikacji, identyfikacji i szeregowania realizowane w każdym węźle sieci	zadania klasyfikacji, znakowania i kondycjonowania dotyczące wprowadzanych strumieni ruchu w węzłach brzegowych i dotyczące klas ruchu w węzłach wewnętrznych sieci
Sterowanie dostępem do sieci	bezwzględnie konieczne	konieczne tylko dla bezwzględnego różnicowania jakości usług
Koordinacja różnicowania jakości usług	w połączeniach pomiędzy systemami końcowymi	lokalna
Zarządzanie siecią	zbliżone do zarządzania sieciami z komutacją łączy	zarządzanie charakterystyczne dla sieci z komutacją pakietów

1	2	3
Taryfikacja jakości usług	na podstawie charakterystyk przepływów i wymagań jakościowych	na podstawie rzeczywistego wykorzystania sieci i dostarczonej jakości usług
Skalowalność	mała – liczba implementacji mechanizmów sterowania jest zmienna; zależy od liczby obsługiwanych w sieci indywidualnych przepływów	duża – liczba implementacji mechanizmów sterowania jest stała; zależy od liczby klas spedycji, a w mniejszym stopniu zależy od ilości obsługiwanego w sieci ruchu
Zdefiniowane usługi	<ul style="list-style-type: none"> ➤ usługa gwarantowana (<i>Guaranteed Service</i>); gwarancja pojemności i wartości maksymalnego opóźnienia oraz brak strat, ➤ usługa sterowanego obciążenia (<i>Controlled Load</i>); aproksymacja usług bez gwarancji jakości w warunkach bez natłoku w sieci 	<ul style="list-style-type: none"> ➤ usługa AS (<i>Assured Service</i>) w grupie AF PHB, ➤ usługa Premium (<i>Premium Service</i>) w grupie EF PHB, ➤ usługa Olympic (<i>Olympic Service</i>) w grupie AF PHB

Dla koncepcji sieci z różnicowaniem usług charakterystyczne są:

- zorientowanie koncepcji na jakość usług dla klas ruchu w domenach; różnicowanie jakości usług polega na definiowaniu domen różnicowania usług (definiowanie węzłów brzegowych), definiowanie klas spedycji w tych domenach i sterowanie ilością ruchu wprowadzanego do poszczególnych domen,
- wysoki poziom agregacji obsługiwanego ruchu; jakość usług jest określana dla klas spedycji, co determinuje odpowiednio zagregowany poziom sterowania dostępem do sieci, przeciwdziałania przeciążeniom i sterowania przepływem,
- mieszany (prewencyjny i reaktywny) charakter sterowania ruchem; zmiana jakości usług, zależna od ilości ruchu wprowadzanego do sieci w węzłach brzegowych (metody prewencyjne), jest wynikiem zmian obciążenia w przypadku degradacji jakości usług (metody reaktywne),
- alokacja zasobów dla klas ruchu; alokacja zasobów odbywa się na poziomie wynikającym z poziomu, na którym odbywa się agregacja ruchu w klasy,
- udostępnianie zasobów klasom spedycji; zasoby są udostępniane w ilościach i na zasadach właściwych dla metod ich alokacji,
- pośrednie definiowanie jakości usług, tzn. definiowanie jakości usług jako efektu przyjętego sposobu spedycji klas ruchu (ang. *forwarding treatment*); jakość usług nie jest definiowana wprost przez podawanie wartości określonych wskaźników jakości, ale jest wynikiem obserwacji rezultatów działania algorytmu spedycji ruchu, utworzonego ze złożenia sposobu obsługi klasy ruchu i ograniczeń ilości ruchu wprowadzanego do sieci,
- pośrednie gwarantowanie jakości usługi; zmiana jakości usług, dla danej alokacji zasobów, jest możliwa przez zmianę ilości ruchu wprowadzanego do sieci, tj. przez zmianę obciążenia sieci,
- wymuszanie jakości usług na brzegach sieci; jakość usług dostarczanych danej klasie spedycji, dla ilości zasobów przyznanych klasie oraz ustalonych zasad kierowania ruchem w sieci, zależy od ilości ruchu wprowadzanego do sieci na brzegu sieci,

➤ ograniczona liczba mechanizmów sterowania i ograniczona liczba lokalizacji implementacji tych mechanizmów; jakość usług jest wynikiem złożenia sposobu obsługi klasy ruchu i obciążenia sieci, co oznacza że liczba stosowanych mechanizmów sterowania, implementowanych na brzegu sieci i w węzłach wewnętrznych, jest względnie mała,

➤ duża skalowalność; złożoność implementacji koncepcji usług zróżnicowanych zależy od liczby klas spedycji, liczby miejsc wprowadzania ruchu do sieci, tzn. liczby węzłów brzegowych sieci oraz liczby miejsc kierowania ruchem, tzn. liczby węzłów wewnętrznych sieci, a w mniejszym stopniu zależy od ilości ruchu obsługiwanego w sieci; a zwłaszcza nie zależy od liczby indywidualnych przepływów w sieci.

10.13. PODSUMOWANIE

Ewolucja sieci teleinformatycznych od sieci bez gwarancji jakości usług do sieci z różnicowaniem usług wiąże się z ewolucją architektur sieci oraz wzrostem liczby różnych mechanizmów implementowanych w węzłach sieci. Zadaniem tych mechanizmów jest dostarczanie wyrafinowanych sposobów różnicowania ruchu i jego obsługi w celu spełnienia specyficznych wymagań użytkowników sieci.

Zbiór wymaganych mechanizmów dotyczy m.in. następujących funkcji:

➤ przetwarzania nagłówków pakietów z szybkością równą szybkości napływu; kolejkovanie pakietów przed przetwarzaniem może powodować degradację żądanej jakości usług,

➤ wielowymiarowego klasyfikowania nagłówków pakietów; jednowymiarowa klasyfikacja nagłówków nie spełnia warunków wyznaczania tzw. tras z ograniczeniami,

➤ wielokryterialnego klasyfikowania przepływów do separacji przepływów na różnych poziomach granulacji i różnych wymaganiach na zasoby sieci,

➤ elastycznego i sprawiedliwego szeregowania jednostek danych przepływów; efektywne wykorzystanie zasobów transmisyjnych przez przepływy o zmiennym zapotrzebowaniu na zasoby wymaga przekazywania zasobów pomiędzy przepływami współdzielącymi zasoby,

➤ elastycznego zarządzania pojemnością pamięci buforowych; efektywne wykorzystanie zasobów transmisyjnych wymaga wzajemnego dopasowania systemu zarządzania pamięcią buforową i procedur szeregowania.

Implementacja mechanizmów realizujących podane funkcje w węzłach sieci umożliwia elastyczne agregowanie indywidualnych przepływów, izolowanie przepływów ze względu na żadaną jakość usług oraz dedykowaną alokację zasobów sieci.

Implementacja modelu usług zróżnicowanych jest, w porównaniu z modelem usług zintegrowanych, zadaniem prostszym i bez ograniczeń skalowalności.

Zasadniczą rolę w modelu usług zróżnicowanych odgrywa statyczny lub dynamiczny kontrakt ruchowy użytkownika i dostawcy usług sieciowych, w którym specyfikowana jest klasa usług i dopuszczalna ilość ruchu danej klasy. Możliwość formułowania

kontraktów statycznych lub dynamicznych pozwala na dostosowanie obsługi ruchu do jego zmienności w czasie. Operacje klasyfikowania, zarządzania, kształtowania i szeregowania ruchu, wykonywane na brzegu sieci (w routerach brzegowych), są praktyczną realizacją warunków obsługi ruchu specyfikowanych w kontrakcie ruchowym.

Operacje klasyfikacji, kontrolowania, kształtowania i szeregowania pozwalają na dostarczanie różnej jakości usług, w tym obsługę ruchu wymagającego małych opóźnień i małych wariacji opóźnień (np. *Premium Service*) oraz obsługę ruchu o jakości wyższej niż bez gwarancji jakości usług (np. *Assured Service*).

Istotną cechą modelu usług zróżnicowanych jest względność jakości usług dostarczanych poszczególnym, wyróżnionym klasom ruchu. Z tego punktu widzenia obsługa ruchu w modelu usług zróżnicowanych jest obsługą ze względnymi priorytetami.

Model usług zróżnicowanych jest znacząco różny od modelu usług zintegrowanych. Przyczyny różnic wymienionych modeli różnicowania jakości usług dostarczanych w sieci są następujące:

- ograniczona liczba klas ruchu – ilość informacji o stanie sieci jest proporcjonalna do liczby klas, a nie przepływów, co znacznie zwiększa skalowalność modelu usług zróżnicowanych,

- liczba miejsc wykonywania złożonych operacji przetwarzania, tzn. klasyfikacji, kontrolowania, kształtowania i szeregowania ruchu – operacje te są wykonywane tylko na brzegu sieci, co upraszcza implementację modelu i zarządzanie ruchem w modelu usług zróżnicowanych,

- lokalizacja miejsc wykonywania operacji przetwarzania – wymagana szybkość przetwarzania na brzegu sieci jest względnie mała, czego konsekwencją jest złożoność względnie wolnych routerów brzegowych i prostota względnie szybkich routerów szkieletowych.

LITERATURA

- [1] APOSTOLOPOULOS G., GUERIN R., KAMAT S., ORDA A., TRIPATHI S.K., *Interdomain QoS Routing in IP networks: a feasibility and cost/benefit analysis*, IEEE Network, September/October 1999, s. 43–54.
- [2] BERTSEKAS D., GALLAGER R., *Data networks*, Prentice-Hall International, Englewood Cliffs 1987.
- [3] BONATTI M., CASALI F., POPPLE G., *Integrated broadband communications: network and engineering aspects*, North-Holland, Amsterdam 1991.
- [4] BRAUN T., GUENTER M., KHALIL I., *Management of Quality of Service enabled VPNs*, IEEE Communications Magazine, May 2001, s. 90–98.
- [5] CHAO H.J., GUO X., *Quality of Service control in High-Speed Networks*, Wiley, 2002.
- [6] CHEUNG S.Y., PENCEA S., *Pipelined sections: a new buffer management discipline for scalable QoS provision*, Emory University (www.itpapers.com).
- [7] CLARK D., WROCLAWSKI J., *An approach to service allocation in the Internet*, Internet Draft, July 1997.

- [8] CLARK D., FANG W., *Explicit allocation of best effort packet delivery service*, IEEE/ACM Transactions on Networking, Vol. 6, No. 4, April 1998.
- [9] DAVROLIS C., STILIADIS D., RAMANATHAN P., *Proportional Differentiated Services: delay differentiation and packet scheduling*, IEEE Transactions on Networking, Vol. 10, No. 1, February 2002, s. 12–26.
- [10] DECASPER D., DITTIA Z., PARULKAR G., PLATTNER B., *Router plugins: a software architecture for next generation routers*, IEEE/ACM Transactions on Networking, Vol. 8, No. 1, February 2000, s. 2–16.
- [11] DOVROLIS C., RAMANATHAN P., *A case for Relative Differentiated Services and the Proportional Differentiation Model*, IEEE Network, September/October 1999, s. 26–34.
- [12] FELDMAN A., REXFORD J., CACERES R., *Efficient policies for carrying web traffic over flow-switched networks*, IEEE/ACM Transactions on Networking, Vol. 6, No. 6, December 1998, s. 673–685.
- [13] FENG Q.C., KANDLUR D.D., SAHA D., SHIN K.G., *Adaptive packet marking for maintaining end-to-end throughput in a Differentiated Services Internet*, IEEE/ACM Transactions on Networking, Vol. 7, No. 5, June 1993, s. 685–697.
- [14] FLEGKAS P., TRIMINTZIOS P., PAVLOU G., *A policy-based Quality of Service management system for IP DiffServ networks*, IEEE Network, March/April 2002, s. 50–56.
- [15] FULP W.E., REEVES D.S., *Optimal provisioning and pricing of differentiated services using QoS class promotion*, N.C. State University (www.itpapers.com).
- [16] FURINI M., TOWSLEY D., *Real-time traffic transmission over the Internet*,
- [17] GEORGIDIS L., GUERIN R., PERIS V., SIVARAJAN K., *Efficient network QoS provisioning based on per node traffic shaping*, IEEE/ACM Transactions on Networking, Vol. 4, No. 4, August 1996.
- [18] GEVROS P., CROWCROFT J., KIRSTEIN P., BHATTI S., *Congestion control mechanisms and the best effort service model*, IEEE Network, May/June 2001, s. 16–26.
- [19] GIBBENS R., KEY P., *Distributed control and resource marking using best-effort routers*, IEEE Network, May/June 2001, s. 54–59.
- [20] GROSSGLAUSER M., KESHAVS., TSE D., *RCBR: A simple and efficient service for multiple time-scale traffic*, IEEE/ACM Transactions on Networking, August 1998.
- [21] GUERIN R.A., PLA V., *Aggregation and conformance in Differentiated Service networks: A case study*, Computer Communication Review, Vol. 31, No. 1, January 2001, s. 21–32.
- [22] GUPTA P., McKEOWN N., *Algorithms for packet classification*, IEEE Network, March/April 2001, s. 24–31.
- [23] IYER S., KOMPPELLA R., SHELAT A., *ClassIP: An architecture for fast and flexible packet classification*, IEEE Network, March/April 2001, s. 33–41.
- [24] JACOBSON V., NICHOLS K., PODURI K., *An expedited forwarding PHB*, Internet Engineering Task Force, Differentiated Services Working Group, Internet Draft, February 1999.
- [25] KUMMAR V.P., LAKSHMAN T.V., STILIADIS D., *Beyond Best Effort: router architectures for the differentiated services of tomorrow's Internet*, IEEE Communications Magazine, May 1988, s. 152–164.
- [26] LEUNG M.K.H., LUI J.C.S., YAU D.K.Y., *Adaptive proportional delay Differentiated Services: characterization and performance evaluation*, IEEE Transactions on Networking, Vol. 9, No. 6, December 2001, s. 801–817.

- [27] LORENZ D.H., ORDA A., *QoS routing in networks with uncertain parameters*, IEEE/ACM Transactions on Networking, Vol. 6, No. 6, December 1998, s. 768–778.
- [28] LORENZ D.H., *QoS routing in networks with uncertain parameters*, University of Haifa, 1998.
- [29] MEER H., PULIAFITO A., RICHTER J.P., TOMARCHIO O., *QoS-adaptation by software agents in the presence of defective reservation mechanisms in the Internet*, University of Hamburg, 2000.
- [30] NICHOLS K., JACOBSON V., ZHANG L., *A two-bit Differentiated Services architecture for the Internet*, Internet Draft, April 1999.
- [31] NONG G., HAMDI M., *On the provision of Quality-of-Service guarantees for input queued switches*, IEEE Communications Magazine, December 2000, s. 62–69.
- [32] QIU J., KNIGHTLY W., *Measurement-based admission control with aggregate traffic envelopes*, IEEE Transactions on Networking, Vol. 9, No. 4, April 2001, s. 199–210.
- [33] RAJAN R., VERMA D., KAMAT S., FELSTAIN E., HERZOG S., *A policy framework for Integrated and Differentiated Services in the Internet*, IEEE Network, September/October 1999, s. 36–41.
- [34] REISSLEIN M., ROSS K.W., RAJAGOPAL S., *A framework for guaranteeing statistical QoS*, IEEE Transactions on Networking, Vol. 10, No. 1, February 2002, s. 27–42.
- [35] RUIZ-SANCHEZ M., ANTIPOLIS S., BIRSACK E.W., DABBOUS W., *Survey and taxonomy of IP address lookup algorithms*, IEEE Network, March/April 2001, s. 8–23.
- [36] SCHMIT J., KARSTEN M., WOLF L., STEINMETZ R., *Aggregation of guaranteed service flows*, Darmstadt University of Technology, 2001 (www.itpapers.com).
- [37] STOICA I., ZHANG H., NG T.S.E., *A hierarchical fair service curve algorithm for link-sharing, real-time, and priority services*, IEEE/ACM Transactions on Networking, Vol. 8, No. 2, April 2000, s. 185–211.
- [38] TEITELBAUM B., HARES S., DUNN L., NEILSON R., NARAYAN V., REICHMEYER F., *Internet2 Qbone building a testbed for Differentiated Services*, IEEE Network, September/October 1999, s. 8–16.
- [39] TRIMINTZIOS P., ANDRIKOPOULOS G.P., FLEGKAS P., GRIFFIN D., GEORGATOS P., GODERIS D., T'JONES Y., GEORGIADIS L., JACQUENET Ch. EGAN R., *A management and control architecture for providing IP Differentiated Services in MPLS-based networks*, IEEE Communications Magazine, May 2001, s. 82–88.
- [40] VERMA D.C., CALO S., AMIRI K., *Policy-based management of content distribution networks*, IEEE Network, March/April 2002, s. 34–39.
- [41] WAICHEE A.F., *Delay-optimal quorum consensus for distributed systems*, IEEE Transactions on Parallel and Distributed Systems, Vol. 8, No. 1, January 1997, s. 59–69.
- [42] WANG Z., *Internet QoS: Architectures and mechanisms for Quality of Service*, Academic Press, London 2001.
- [43] WEISS W., *QoS with Differentiated Services*, Bell Labs Technical Journal, October–December 1998, s. 48–62.
- [44] XIAO X., NI L.M., *Internet QoS: A big picture*, IEEE Network, March/April 1999, s. 8–18.
- [45] YEOM I., REDDY A.L.N., *Modelling TCP behaviour in a Differentiated Services network*, IEEE/ACM Transactions on Networking, Vol. 9, No. 1, February 2001, s. 31–46.

11. INŻYNIERIA RUCHU

Różnicowanie jakości usług w sieciach teleinformatycznych, wynikające ze stosowania różnych koncepcji dostarczania jakości usług sieci, takich jak model usług zintegrowanych IS (ang. *Integrated Services*) czy model usług zróżnicowanych DS (ang. *Differentiated Services*), można interpretować jako różnicowanie degradacji jakości usług dostarczanych różnym klasom usług wtedy, gdy wzrasta obciążenie sieci. Gdy obciążenie zasobów sieci jest względnie małe, wówczas różnica jakości usług dostarczanych użytkownikom przez sieci z implementacją różnych koncepcji (modeli) jakości usług oraz sieci bez gwarancji jakości usług jest niewielka.

Dla zadanego obciążenia sieci, danego ilością i strukturą ruchu generowanego przez użytkowników, przeciążenia sieci są powodowane niedostępnością zasobów lub nierównomierną ich dystrybucją. W pierwszym przypadku wzrost jakości usług sieci jest możliwy w wyniku zwiększenia ilości zasobów udostępnianych dla ruchu użytkowników sieci. W drugim przypadku wzrost jakości usług można uzyskać przez zmianę rozkładu obciążeń zasobów sieci. Wynika stąd, że dopiero wyczerpanie możliwości przeciwdziałania przeciążeniom przez równomierne obciążenie zasobów sieci uzasadnia stosowanie takich koncepcji dostarczania jakości usług, które polegają na dostarczaniu usług o zadanej jakości jednym klasom ruchu kosztem degradacji jakości usług dostarczanych innym użytkownikom tej samej sieci.

Ograniczeniem efektywności modeli usług zintegrowanych i (lub) usług zróżnicowanych w sieciach jest zależność jakości usług sieci oraz stopnia wykorzystania zasobów sieci od metod wyznaczania tras implementowanych w sieci. Zwiększenie efektywności modeli zapewniania jakości usług uzyskuje się dzięki stosowaniu takich wspierających je rozwiązań, które prowadzą do efektywnego wykorzystania dostępnych zasobów sieci.

Terminem inżynieria ruchu (ang. *traffic engineering*) określa się metody, mechanizmy i procedury sterowania ruchem w sieci przeciwdziałające powstawaniu takich przeciążeń, których przyczyną jest nierównomierne obciążenie (a tym samym wykorzystanie) zasobów sieci teleinformatycznej. Zadania inżynierii ruchu w sieciach mogą być formułowane jako zadania badania, wyznaczania i optymalizacji wykorzystania zasobów sieci przy zachowaniu założonego poziomu jakości usług dostarczanych użytkownikom sieci lub jako maksymalizacja jakości usług sieci dla zadanych zasob-

bów sieci. Zadania te, zorientowane zarówno na równomierność, jak i efektywność wykorzystania zasobów sieci, charakteryzują różne sformułowania wskaźników jakości działania sieci i ograniczeń. Zadania inżynierii ruchu można utożsamiać z zadaniami wyznaczania tras z ograniczeniami i wymuszania przepływów przez tak wyznaczone trasy, których celem nie jest zastępowanie algorytmów różnicowania jakości usług, ale poprawa ich działania. Istotą metod inżynierii ruchu nie jest zastępowanie modeli zapewniania jakości usług w sieciach, ale dostarczanie mechanizmów ułatwiających zapewnianie jakości usług. Metody inżynierii ruchu rozszerzają możliwości modeli zapewniania jakości usług. Zastosowanie metod inżynierii ruchu wpływa na efektywniejsze wykorzystanie zasobów sieci, co dla zadanej ilości ruchu obsługiwanego w sieci o danych zasobach jest równoważne ze wzrostem jakości usług.

Przeciwdziałanie przeciążeniom, należące do zadań inżynierii ruchu, oraz sterowanie degradacją jakości usług w przypadku wystąpienia przeciążenia sieci, będące celem implementacji różnych koncepcji różnicowania jakości usług, są działaniami komplementarnymi. Modele usług zintegrowanych i usług zróżnicowanych uzupełniają więc metody inżynierii ruchu i odwrotnie. Inżynieria ruchu i modele różnicowania jakości usług są komplementarne także w tym znaczeniu, że metody inżynierii ruchu ułatwiają implementację różnych koncepcji różnicowania jakości usług w sieciach.

W praktycznych rozwiązaniach zadań inżynierii ruchu zwykle stosuje się podejście pierwsze, tzn. celem jest takie sterowanie ruchem w sieci, które minimalizuje możliwość powstawania przeciążeń, maksymalizuje równomierność wykorzystania zasobów sieci oraz ogranicza zakres oddziaływania przeciążeń na jakość dostarczanych usług.

Przykład 11.1

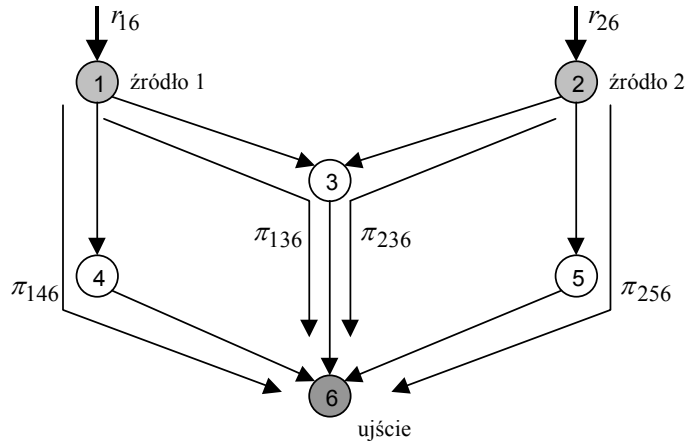
Ilustracją zapotrzebowania na metody właściwe dla inżynierii ruchu jest zadanie rozdziału zasobów i wyznaczania tras w przykładowej sieci, przedstawionej na rysunku 11.1.

Dla zadania przekazywania ruchu w sieci przyjęto, że:

- pojemności wszystkich łączy skierowanych są jednakowe i równe 10 jednostkom, tzn. $C_{13} = C_{14} = C_{23} = C_{25} = C_{36} = C_{46} = C_{56} = 10$,
- pomiędzy parami węzłów (1, 6) oraz (2, 6) istnieją po dwie rozłączne trasy, odpowiednio: π_{146} i π_{136} oraz π_{256} i π_{236} ,
- przepływy generowane w źródłach 1. oraz 2. i kierowane do ujścia są równe – odpowiednio – r_{16} i r_{26} jednostkom,
- miarą jakości obsługi ruchu generowanego w źródłach i przenoszonego przez sieć jest wartość średnia opóźnienia jednostek danych d_{ij} , której wartość dla przepływu f_{ij} w łączu o pojemności C_{ij} jest wartością wyrażenia:

$$d_{ij} = \frac{f_{ij}}{C_{ij} - f_{ij}} f_{ij}.$$

Dla omawianej sieci oraz podanych parametrów, charakteryzujących działanie tej sieci, możliwe jest m.in. wystąpienie następujących przypadków:



Rysunek 11.1. Przykładowa sieć z czterema trasami łączącymi dwa źródła z jednym ujściem

Przypadek 1

Wartość generowanych przepływów jest jednakowa i równa: $r_{16} = 5$ oraz $r_{26} = 5$.

Jeżeli przepływy $r_{16} = 5$ oraz $r_{26} = 5$ są przekazywane, odpowiednio:

➤ trasami π_{146} oraz π_{256} tak, że:

$$f_{14} = f_{46} = r_{16} = 5, \quad f_{25} = f_{56} = r_{26} = 5 \quad \text{i} \quad f_{13} = f_{23} = f_{36} = 0,$$

to opóźnienie $\bar{d} = 0,4$, tzn. żadna z tras nie jest przeciążona;

➤ trasami π_{136} i π_{236} tak, że:

$$f_{14} = f_{46} = 0, \quad f_{25} = f_{56} = 0, \quad f_{13} = r_{16} = 5, \quad f_{23} = r_{26} = 5 \quad \text{i} \quad f_{36} = r_{16} + r_{26} = 10,$$

to opóźnienie $\bar{d} = \infty$, tzn. trasy π_{136} i π_{236} są przeciążone.

Przypadek 2

Wartość generowanych przepływów jest jednakowa i równa: $r_{16} = 10$ oraz $r_{26} = 10$.

Jeżeli przepływy $r_{16} = 10$ oraz $r_{26} = 10$ są przekazywane, odpowiednio:

➤ trasami π_{146} oraz π_{256} tak, że:

$$f_{14} = f_{46} = r_{16} = 10, \quad f_{25} = f_{56} = r_{26} = 10 \quad \text{i} \quad f_{13} = f_{23} = f_{36} = 0,$$

to opóźnienie $\bar{d} = \infty$, tzn. trasy π_{146} i π_{256} są przeciążone;

➤ trasami π_{136} i π_{236} tak, że:

$$f_{14} = f_{46} = 0, \quad f_{25} = f_{56} = 0, \quad f_{13} = r_{16} = 10, \quad f_{23} = r_{26} = 10 \quad \text{i} \quad f_{36} = r_{16} + r_{26} = 20,$$

to opóźnienie $\bar{d} = \infty$, tzn. trasy π_{136} i π_{236} są przeciążone;

➤ trasami π_{146} i π_{136} oraz π_{256} i π_{236} tak, że:

$$f_{14} = f_{46} = \frac{1}{2} r_{16} = 5, \quad f_{25} = f_{56} = \frac{1}{2} r_{26} = 5, \quad f_{13} = \frac{1}{2} r_{16} = 5,$$

$$f_{23} = \frac{1}{2}r_{16} = 5 \text{ i } f_{36} = \frac{1}{2}r_{16} + \frac{1}{2}r_{26} = 10,$$

to opóźnienie $\bar{d} = \infty$, tzn. trasy π_{136} i π_{236} są przeciążone;

► trasami π_{146} i π_{136} oraz π_{256} i π_{236} tak, że:

$$f_{14} = f_{46} = \frac{3}{5}r_{16} = 6, \quad f_{25} = f_{56} = \frac{3}{5}r_{26} = 6, \quad f_{13} = \frac{2}{5}r_{16} = 4,$$

$$f_{23} = \frac{2}{5}r_{16} = 4 \text{ i } f_{36} = \frac{2}{5}r_{16} + \frac{2}{5}r_{26} = 8,$$

to opóźnienie $\bar{d} \approx 0,53$, tzn. żadna z tras nie jest przeciążona;

► trasami π_{146} i π_{136} oraz π_{256} i π_{236} tak, że:

$$f_{14} = f_{46} = 0,7r_{16} = 8, \quad f_{25} = f_{56} = 0,7r_{26} = 8, \quad f_{13} = 0,3r_{16} = 3,$$

$$f_{23} = 0,3r_{16} = 4 \text{ i } f_{36} = 0,3r_{16} + 0,3r_{26} = 6,$$

to opóźnienie $\bar{d} \approx 0,6$, tzn. żadna z tras nie jest przeciążona.

Występujące w podanym przykładzie przypadki ilustrują zależność jakości obsługi ruchu, mierzonej np. średnim opóźnieniem jednostek danych, od sposobu wyznaczenia tras i kierowania ruchu do obsługi przez poszczególne trasy. Dla takich samym wymagań użytkowników sieci (danych macierzą wymagań) oraz ustalonych zasobów sieci (danych w postaci tras) różne sposoby kierowania ruchem prowadzą do różnej jakości obsługi przepływów, w tym do rozwiązań powodujących degradację jakości usług, będącą rezultatem powstania przeciążeń w sieci.

Podany przykład ilustruje konsekwencje dwóch właściwości klasycznych metod wyznaczania tras z komutacją pakietów, tzn. zorientowanie na ujście oraz lokalność podejmowania decyzji o wyborze trasy.

Zorientowanie na ujście (ang. *destination-based routing*) protokołu routingu polega na tym, że dla wszystkich pakietów w danym węźle sieci, których adresy ujścia są takie same i które współdzielą ten sam prefiks adresu ujścia, następnym węzłem na trasie do ujścia jest ten sam węzeł. Dlatego zwykle w tablicach routingu węzłów istnieje tylko jedna trasa do węzła o danym adresie, z wyjątkiem przypadków, gdy do danego ujścia prowadzi kilka tras o jednakowym koszcie. Zorientowanie na ujście jednokryterialnego protokołu routingu nie wykorzystuje możliwości zrównoleglenia przepływu i korzystania z wielu różnych tras do obsługi jednego przepływu. Brak możliwości zrównoleglenia obsługi pojedynczych przepływów pomiędzy systemami końcowymi powoduje, że obciążenie sieci może być nierównomierne. Nierównomierność obciążenia sieci jest dodatkową przyczyną występowania przeciążeń; wzrost nierównomierności obciążenia powoduje spadek wartości całkowitego ruchu w sieci, przy którym możliwe jest powstanie przeciążenia. W przypadku występowania w sieci, pomiędzy systemami końcowymi, wielu różnych tras o zbliżonych kosztach, możliwe są także permanentne oscylacje tras.

Lokalność podejmowania decyzji o wyborze trasy polega na tym, że wybór trasy jest najlepszy z uwagi na węzeł, którego ta decyzja dotyczy. Taki sposób wyboru trasy opiera się na założeniu, że decyzja podejmowana w węźle ma niewielki lub żaden wpływ na stan całej sieci. Założenie takie, poza szczególnymi przypadkami, nie jest prawdziwe. Jeżeli kryterium wyboru trasy we wszystkich węzłach sieci jest odległość węzłów, to wybór najkrótszych tras może bardzo szybko doprowadzić do powstania przeciążenia. Bardziej bezpieczne, ze względu na powstawanie przeciążeń, są protokoły routingu, stosujące do wyboru trasy kryteria inne niż odległość. Optymalizacja wykorzystania zasobów sieci nie jest możliwa, jeżeli do podejmowania decyzji o ich alokacji nie są brane pod uwagę ani cel działania całej sieci, ani globalny stan zasobów.

Jedynym sposobem rozwiązania problemów nierównomiernego obciążenia zasobów, powodowanych protokołami routingu zorientowanymi na ujście, jest zmiana zasad i protokołów wyboru tras w sieci lub uzupełnienie protokołów routingu o mechanizmy zarządzania ruchem wewnątrz sieci.

Zmiana zasad wyznaczania tras, w porównaniu z protokołami zorientowanymi na ujście, oznacza wykorzystanie metod wielokryterialnego wyboru tras zaliczanych do metod wyznaczania tras z ograniczeniami (ang. *constraint-based routing*).

Druga grupa rozwiązań polega na wprowadzeniu do sieci niezależnego od routingu systemu zarządzania ruchem (ang. *overlay model*). Wyznaczane logiczne połączenia pomiędzy każdą parą węzłów (optymalne trasy) dają w rezultacie pełne pokrycie danej sieci (ang. *mesh network*). Tak wyznaczone połączenia logiczne można interpretować jako bezpośrednie trasy dla nadrzędnego poziomu sterowania przepływami, na którym wykorzystywane są protokoły sygnalizacji. Celem systemu zarządzania ruchem jest taka alokacja bezpośrednich tras w sieci, dla której uzyskiwane jest równoważenie obciążenia wszystkich zasobów transmisyjnych.

11.1. ZADANIA INŻYNIERII RUCHU W SIECIACH

11.1.1. MECHANIZMY JAKOŚCI USŁUG W SIECIACH Z KOMUTACJĄ PAKIETÓW

Realizacja koncepcji różnicowania jakości usług w sieciach oznacza, że w sieci są implementowane mechanizmy umożliwiające wymuszanie różnicowania ruchu oraz różnicowania jakości obsługi ruchu, a także monitorowanie jakości różnicowania. Mechanizmy wspierające różnicowanie należą do dwóch grup:

➤ mechanizmy wymuszania różnicowania usług, klasyfikowania i odwzorowywania przepływów oraz klas ruchu i sterowania ilością wykorzystywanych zasobów (ang. *data path*),

➤ mechanizmy negocjacji warunków obsługi ruchu, alokacji zasobów i identyfikacji uprawnień do korzystania z zasobów (ang. *control path*).

Innym kryterium klasyfikacji mechanizmów stosowanych do realizacji różnych koncepcji jakości usług w sieciach jest ich zdolność do realizacji założonych funkcji, którą mierzy się m.in. wartościami następujących wskaźników:

- sprawiedliwość dostępu do zasobów; sposób dostępu przepływów do zasobów nadmiarowych,
- izolowanie przepływów; zabezpieczanie przed zawłaszczaniem zasobów jednych przepływów przez inne przepływy,
- efektywność; liczba obsługiwanych przepływów dla danego poziomu jakości usług,
- złożoność; trudność implementacji i nadmiar organizacyjny.

Projektowanie sieci o zadanej zdolności do różnicowania jakości usług oznacza wybór, spośród wielu różnych dostępnych, mechanizmów charakteryzujących się różną: funkcjonalnością, skalowalnością, ilością i jakością informacji, niezbędną do efektywnego stosowania mechanizmu złożonością implementacji i utrzymywania itd.

11.1.2. CELE OPTIMALIZACJI STOPNIA WYKORZYSTANIA ZASOBÓW SIECI

Ogólnym celem stosowania mechanizmów inżynierii ruchu w sieciach jest wzrost lub utrzymanie wysokiego poziomu jakości działania sieci przez jednoczesną optymalizację jakości usług sieci oraz stopnia wykorzystania zasobów sieci. Większość zadań inżynierii ruchu jest rozwiązywana za pomocą m.in. różnych mechanizmów równoważenia obciążeń w sieci.

Szczegółowe sformułowania zadań inżynierii ruchu zależą od wielu czynników, w tym m.in. od: architektury sieci, charakterystyk ruchu generowanego przez użytkowników, wymagań dotyczących jakości usług sieci, ilości zasobów krytycznych, możliwości implementacji określonych metod optymalizacji, stosowanych systemów rozliczania jakości usług.

Do najczęściej formułowanych i wzajemnie zależnych celów stosowania metod inżynierii ruchu zalicza się następujące:

- minimalizacja częstości występowania przeciążeń i zakresu ich oddziaływania,
- minimalizacja strat jednostek danych w sieci,
- minimalizacja całkowitego opóźnienia jednostek danych wnoszonych przez sieć,
- wzrost efektywności wykorzystania zasobów, zwłaszcza zasobów transmisyjnych,
- wzrost liczby użytkowników (przepływów) obsługiwanych w sieci o danych zasobach przy zachowaniu wymaganej (założonej) jakości usług.

Zadaniem mechanizmów inżynierii ruchu w sieci, wspierających osiągnięcie wyżej wymienionych celów, jest zapobieganie powstawaniu zdarzeń powodujących degradację jakości usług lub ograniczanie konsekwencji zdarzeń powodujących degradację jakości usług. Z tego punktu widzenia zakres zadań inżynierii ruchu nie różni się od zakresu łącznych zadań wyznaczania tras, przeciwdziałania przeciążeniom i sterowania przepływem.

Minimalizacja częstości występowania przeciążeń sprowadza się do usuwania przyczyn powstawania przeciążeń, do których zalicza się:

- nadmiarowość ruchu przyjmowanego do obsługi w sieci wobec zasobów sieci,
- nierównomierność rozdziału obsługiwanego ruchu, w wyniku którego niektóre z elementów sieci są przeciążone, a inne niedociążone.

Nadmiarowość ruchu wobec zasobów może być usunięta albo przez wzrost zasobów, albo ograniczenie ilości ruchu wprowadzanego do sieci i zarządzanie ruchem wprowadzonym do sieci. Rozwiązanie pierwsze jest właściwe dla zadania projektowania lub migracji sieci, drugie natomiast jest właściwe dla zadania sterowania dostępem do sieci i sterowania przepływami w sieci.

Trudniejsze jest zadanie lokalizacji i ograniczania oddziaływania przeciążenia w sieci. Skutecznym sposobem ograniczenia konsekwencji przeciążeń jest szybkość działania mechanizmów sterowania reakcyjnego. Ograniczeniem zakresu stosowania takich mechanizmów jest bezwładność sieci.

Straty jednostek danych w sieciach są konsekwencją powstawania przeciążeń i stosowania różnych mechanizmów ich rozładowywania. Jednostki danych są tracone wtedy, gdy powstanie przeciążenia wymusza szybkie zmniejszenie ilości ruchu w sieci. W takich przypadkach zmniejszanie ilości ruchu odbywa się najczęściej przez odrzucanie odpowiednio znakowanych części obsługiwanego ruchu. Wielkość strat jednostek danych zależy od zakresu oddziaływania przeciążenia i bezwładności sieci.

Minimalizacja całkowitego opóźnienia jednostek danych wnoszonych przez sieć jest zadaniem sterowania przepływem z ograniczeniami. Opóźnienie jednostek danych w sieci zależy od metod udostępniania zasobów oraz obciążenia zasobów sieci. Zadaniem metod udostępniania zasobów jest ograniczanie zmienności opóźnienia. Zmniejszanie obciążenia sieci zwykle prowadzi do powstania warunków, umożliwiających zmniejszenie opóźnienia. Minimalizacja opóźnienia jest zadaniem minimalizacji maksymalnego stopnia wykorzystania zasobów, którego właściwości mają znaczący wpływ na jakość działania sieci: ogranicza liczbę strat, prowadzi do równoważenia obciążeń w sieci i umożliwia wzrost ilości obsługiwanego ruchu.

Uzyskanie założonego i stabilnego poziomu wykorzystania zasobów sieci oznacza, że uzyskana jest optymalna jakość działania sieci w sensie wartości współczynnika strat, całkowitego opóźnienia itd. Optymalna jakość działania sieci jest wrażliwa na zmiany obciążenia zasobów, zwłaszcza na nierównomierność obciążenia zasobów.

Zwiększenie liczby użytkowników (przepływów) obsługiwanych w sieci o danych zasobach, przy zachowaniu wymaganej (założonej) jakości usług, jest możliwe wtedy, gdy rośnie równomierność obciążenia zasobów.

11.1.3. KOMPONENTY SYSTEMU INŻYNIERII RUCHU W SIECI

Cele systemu inżynierii ruchu w sieci uzyskuje się dzięki stosowaniu metod równoważenia obciążeń zasobów sieci. W zadaniach inżynierii ruchu, zorientowanych na jednoczesną optymalizację jakości działania sieci i stopnia wykorzystania zasobów, metody

równoważenia obciążeń są uzupełnieniem konserwatywnych sposobów wykorzystywania zasobów, zorientowanych na optymalizację wskaźników jakości działania sieci.

Rozwiązanie klasycznego jednokryterialnego zadania wyznaczania optymalnych tras wiąże się z realizacją zasady zachowywania zasobów (ang. *resource conserving*), tzn. wyznaczone rozwiązanie cechuje optymalne zużycie zasobów. Przykładem takiego rozwiązania jest np. wybór najkrótszej trasy. Wybór trasy, innej niż najkrótsza, oznacza wzrost ilości wykorzystywanych zasobów. Rozwiązania zachowujące zasoby są efektywne ze względu na stopień wykorzystania zasobów w warunkach dużego obciążenia sieci. Zastosowanie metod równoważenia obciążeń oznacza m.in., że ilość zasobów wykorzystywanych do obsługi danego ruchu w sieci nie jest optymalna.

Rozwiązania zadań inżynierii ruchu charakteryzuje kompromis pomiędzy metodami zachowującymi zasoby i równoważącymi obciążenia. Przykładem kompromisu są takie algorytmy, zgodnie z którymi w warunkach dużego obciążenia sieci optymalizowane jest wykorzystanie zasobów, a w warunkach mniejszego obciążenia – jakość dostarczanych usług. Oznacza to np. przełączanie obsługi danego ruchu przez najkrótsze trasy (ang. *shortest path*) na obsługę przez trasy o maksymalnej dostępnej przepustowości (ang. *widest path*) w zależności od całkowitego obciążenia sieci.

Zrównoważenie obciążenia w sieci jest warunkiem efektywnego wykorzystania zasobów, wzrostu ilości obsługiwanego ruchu oraz stabilności jakości działania sieci, której miarą jest stałość miar jakości usług, takich jak współczynnik strat, opóźnienie itd.

Efektywność metod równoważenia obciążeń w sieci zależy od:

- zakresu i szczegółowości monitorowania stanu sieci,
- mechanizmów wyznaczania optymalnych tras z ograniczeniami korzystających z danych z systemu monitorowania,
- mechanizmów wymuszania alokacji zasobów i przepływów w sieci.

11.2. MONITOROWANIE STANÓW SIECI

Wyznaczanie tras w sieci wymaga znajomości struktury topologicznej i dostępności zasobów sieci, przede wszystkim pojemności (pasma) kanałów transmisyjnych. Zwykle zakłada się, że pojemność pamięci buforowej jest wystarczająco duża, co pozwala nie uwzględniać tego zasobu w ograniczeniach zadań wyznaczania tras.

Praktyczne realizacje algorytmów wyznaczania tras muszą uwzględniać częstotliwość aktualizacji informacji o stanie zasobów. Względnie szybkie zmiany stanu zasobów powodują, że niezbędny jest kompromis pomiędzy dokładnością informacji o stanie zasobów i ilością informacji o stanie zasobów przekazywanych w sieci. Jednym z możliwych rozwiązań jest aktualizacja informacji o stanie zasobów tylko wtedy, gdy zmiany przekraczają zadane wartości progowe.

Do ciągłego monitorowania stanu sieci i jej elementów stosuje się różne protokoły routingu, które umożliwiają wykrywanie zmian struktury topologicznej sieci i zmianę tras. Na ogół algorytmy routingu, wymieniające informację o strukturze topologicznej

sieci, tzn. o łączach w sieci i ich kosztach, opierają się na dwóch ogólnych metodach: wektora odległości (ang. *distance vector routing*) i stanu łączy (ang. *link state routing*).

11.2.1. ALGORYTMY WEKTORÓW ODLEGŁOŚCI

W algorytmach wykorzystujących wektory odległości sąsiednie węzły sieci wymieniają okresowo między sobą wektory odległości do każdego z węzłów ujściowych sieci. Algorytmy te są praktycznymi realizacjami rozproszonego asynchronicznego algorytmu Bellmana–Ford’a (ang. *distributed asynchronous Bellman-Ford algorithm*), którego podstawowymi ograniczeniami są: duża złożoność obliczeniowa oraz duża liczba wymienianych wiadomości w zadaniach aktualizacji.

Dla opartych na wektorach odległości algorytmów charakterystyczne jest to, że w każdym węźle budowana jest tablica routingu, której elementy są uaktualniane na podstawie informacji otrzymywanych z sąsiednich węzłów. Proces wymiany informacji i aktualizacji tablic routingu jest podobny do rozwiązania optymalnego.

Złożoność obliczeniowa algorytmu jest $O(n^3)$, gdzie n jest liczbą węzłów sieci.

Główną zaletą metody wektorów odległości jest jej prostota, wadą natomiast jest to, że wraz ze wzrostem rozmiaru sieci rośnie długość wiadomości z informacją o strukturze topologicznej oraz wydłuża się czas propagacji informacji o zmianach w sieci. Kolejnym ograniczeniem omawianych algorytmów jest ich skalowalność.

Istnieje wiele implementacji algorytmów wykorzystujących wektory odległości, takich jak np.:

➤ RIP (ang. *Routing Information Protocol*)

RIP jest bardzo prostym protokołem, w którym decyzja o wyborze trasy jest podejmowana na podstawie zliczania liczby skoków (ang. *hop count*), gdzie liczba skoków jest równa liczbie węzłów (routerów) na trasie. Trasa jest wyznaczana z pominięciem długości łączy pomiędzy węzłami i pojemności dostępnych w tych łączach.

➤ IGRP (ang. *Interior Gateway Routing Protocol*)

Działanie protokołu IGRP jest zbliżone do protokołu RIP, ale protokół IGRP jest lepszy w tym sensie, że w decyzjach dotyczących wyboru trasy uwzględniane są informacje o ograniczeniach pojemności łączy oraz opóźnieniach na trasie.

11.2.2. ALGORYTMY STANÓW ŁĄCZY

W algorytmach stanów łączy wymuszana jest pełna znajomość struktury topologicznej sieci, tzn. stanów łączy sieci przez każdy z węzłów sieci. Proces aktualizacji informacji o stanie łączy polega na dystrybucji informacji o zmianach, która jest realizowana przez tzw. niezawodne zalewanie (ang. *reliable flooding*). Niezawodne zalewanie polega na tym, że wiadomości są wysyłane z danego węzła do wszystkich węzłów sąsiednich i retransmitowane aż do chwili otrzymania potwierdzenia. Ze względu na właściwości zalewania, każdy z węzłów może otrzymać wiele kopii tej samej wiadomości i musi być wyposażony w mechanizm usuwania kopii nadmiarowo-

wych. W rezultacie każdy węzeł otrzymuje jedną kopię wiadomości. Trasy dystrybucji wiadomości tworzą drzewo rozpięte na grafie sieci. Proces wymiany wiadomości o zmianach stanu kończy się wtedy, gdy wszystkie węzły zgodzą się na to, że mają taką samą informację o stanie łączy sieci.

Metodę stanów łączy zaproponowano jako rozwiązującą problem ograniczonej skalowalności metody wektorów odległości.

Po aktualizacji informacji o stanie sieci w każdym z węzłów sieci tworzona jest lokalna tablica przekazywania (ang. *forwarding table*). Lokalna tablica jest optymalna lokalnie, tzn. jest najkorzystniejsza ze względu na dany węzeł. Informacje w tablicy przekazywania są informacjami o najkrótszych trasach do wszystkich węzłów ujściowych. Do obliczania długości tras stosowane są takie algorytmy, jak algorytm Dijkstry.

Złożoność algorytmów korzystających ze stanów łączy jest większa od złożoności algorytmów wektorów odległości, ale są szybciej zbieżne. W razie uszkodzenia węzła lub łącza, albo dodania nowego węzła lub łącza, konieczna jest aktualizacja informacji o stanie sieci w każdym z węzłów sieci.

Do podstawowych implementacji algorytmów wykorzystujących metodę stanów łączy należy m.in. protokół OSPF (ang. *Open Shortest Path First*), który zapewnia wszechstronne monitorowanie stanu sieci. Decyzje o wyborze tras są podejmowane na podstawie wielu parametrów sieci, w tym: długości tras, dostępnej pojemności, kosztu łącza, przeciążenia, opóźnienia. Protokół OSPF obsługuje trzy tryby transmisji: tryb punkt–punkt (ang. *point-to-point*), tryb rozgłoszeniowy (ang. *broadcast*) oraz tryb wielodostępu bez rozgłaszania (ang. *non-broadcast multiaccess*). Ponadto protokół OSPF wspiera procesy: segmentacji sieci (ang. *sub-netting*), filtracji pakietów (ang. *filtering*) oraz rozdzielania przepływów (ang. *splitting*).

Protokół OSPF, ze względu na zakres parametrów wykorzystywanych do podejmowania decyzji o wyborze trasy oraz zdolność wspierania wybranych procesów organizacji przepływów w sieci, jest stosowany jako protokół odniesienia.

Schemat aktualizacji informacji o stanie łączy sieci może być łatwo rozszerzony do przekazywania informacji wymaganych do realizacji zadań inżynierii ruchu.

11.3. ALGORYTMY WYZNACZANIA TRAS W SIECIACH Z JAKOŚCIĄ USŁUG

Zadaniem szerokopasmowych sieci z integracją usług jest obsługa ruchu generowanego przez wiele różnych aplikacji z różnymi wymaganiami dotyczącymi jakości usług. Niezależnie od koncepcji zapewniania jakości usług implementowanej w takiej sieci, głównym problemem jest udostępnianie zasobów sieci poszczególnym przepływowom tak, aby spełnione były wymagania każdego z przepływów i aby zasoby były udostępniane efektywnie w całej sieci. Do podstawowych rozwiązań architektur sieci spełniających podane wymagania należą metody wyznaczania tras, w których jest uwzględniana żądana jakość obsługi (ang. *QoS routing*).

Zasadniczym problemem ustanawiania połączeń spełniających wymagania z zakresu jakości usług jest odwzorowanie wymagań formułowanych dla połączenia pomiędzy systemami końcowymi, dotyczących wymagań lokalnych w węzłach sieci, z których wynika sposób rezerwacji zasobów na trasie.

Zdolność sieci do gwarantowania założonej wartości opóźnienia całkowitego na trasie w sieci z integracją usług zależy na przykład od implementowanych w węzłach sieci dyscyplin szeregowania, charakteryzowanych górnymi ograniczeniami maksymalnego opóźnienia, z których wynika górne ograniczenie maksymalnego opóźnienia całkowitego na trasie. Górne ograniczenie maksymalnego opóźnienia jest wskaźnikiem, który może być stosowany do pomiaru jakości tras i zdolności tras do spełnienia wymagań przepływu. Ze względu na to, że dla każdej dyscypliny szeregowania istnieje jednoznaczny związek pomiędzy gwarantowanym maksymalnym opóźnieniem a szybkością transferu, wymagania dotyczące ograniczenia maksymalnego opóźnienia mogą więc być odwzorowane w wymagania dotyczące szybkości transferu. Znajomość szybkości transferu przepływów obsługiwanych w węzłach i na trasach jest tożsama ze znajomością wolnych zasobów transmisyjnych na trasach sieci. Na podstawie podanej właściwości odwzorowania ograniczenia opóźnienia szybkości transferu możliwe jest wyznaczenie trasy spełniającej wymagania co do jakości usług, formułowane w postaci ograniczenia maksymalnego opóźnienia całkowitego. Można wykazać, że dla danego połączenia z górnym ograniczeniem maksymalnego opóźnienia całkowitego istnienie w sieci trasy spełniającej wymagania jakościowe tego połączenia jest możliwe po wykonaniu co najwyżej N iteracji algorytmu wyznaczania najkrótszej trasy, gdzie N jest liczbą wszystkich łączy w sieci.

Kolejnym problemem jest efektywność procesu wyszukiwania tras w sieci, które spełniają wymagania jakościowe połączeń, tzn. redukcja złożoności obliczeniowej i optymalizacja procesu wyznaczania tras. W podanym wcześniej przykładzie wykazanie istnienia odpowiedniej trasy wymaga wykonania liczby iteracji równej liczbie łączy w sieci. W związku z tym istotnym problemem jest takie wyszukiwanie tras spełniających wymagania, które gwarantuje nie tylko znalezienie trasy, ale także efektywność lub optymalność procesu wyszukiwania ze względu na system zarządzania zasobami sieci.

Innym ważnym problemem wyznaczania tras spełniających wymagania jakościowe jest wielokryterialny wybór tras z ograniczeniami (ang. *constrained-based routing*). Typowe sformułowanie takiego zadania to znalezienie trasy minimalizującej wartość wybranego wskaźnika jakości (np. koszt) z zachowaniem jednego lub więcej ograniczeń (np. maksymalne opóźnienie całkowite). Tak formułowane zadania są zadaniami programowania liniowego, które należą do zadań *NP*-trudnych. Przydatność i realizowalność algorytmów dostarczających dokładne rozwiązania takich zadań w czasie rzeczywistym jest bardzo ograniczona. Jednym z proponowanych rozwiązań tego problemu jest poszukiwanie takich metod wyznaczania rozwiązań przybliżonych, których złożoność może być ograniczana przez wykorzystanie różnych właściwości sieci.

11.3.1. INFORMACJA I ALGORYTMY WYZNACZANIA TRAS

W procesie selekcji trasy spełniającej wymagania z zakresu jakości usług nowego przepływu korzysta się z informacji zarówno o charakterystykach i wymaganiach nowego przepływu, jak i o dostępności zasobów w sieci. Dodatkowo, na potrzeby monitorowania efektywności, istotne jest gromadzenie informacji o ilości zasobów wykorzystywanych przez nowy przepływ. Z oczywistych względów preferowany jest wybór „najtańszych” tras spośród wszystkich dopuszczalnych. Możliwe jest odrzucenie żądania obsługi, jeżeli koszt trasy jest zbyt duży. Zadanie odrzucania żądań dostępu jest wprawdzie zadaniem sterowania dostępem, a nie wyboru trasy, ale wyposażenie algorytmu wyboru trasy w możliwość liczenia np. kosztu trasy daje możliwość odrzucania żądań na etapie wyboru trasy.

Ze względu na algorytmy wyznaczania tras istotne znaczenie mają następujące charakterystyki (metryki) sieci:

- dostępna pojemność łączy transmisyjnych; większość wymagań dotyczących jakości usług jest i może być odwzorowana w zasób, jakim jest związana z szybkością transmisji szerokość pasma (ang. *bandwidth*),

- długość trasy liczona liczbą węzłów na trasie; koszt trasy zależy od jej długości, a im krótsza jest trasa, tym mniejsza ilość wykorzystywanych zasobów,

- polityka udostępniania zasobów; oznacza to ograniczenie zasobów sieci do takich, które mogą spełniać wymagania z zakresu jakości usług; polityka udostępniania może wpływać na uproszczenie zadań optymalizacji wykonywanych na potrzeby wyznaczania trasy.

Nadrzędna zależność jakości usług w sieci od pojemności i długości trasy wynika z tego, że pierwotnymi zasobami w każdym systemie transmisji sygnałów są: pojemność (szerokość pasma), czas i moc sygnału.

11.3.2. ALGORYTMY WYZNACZANIA TRAS

Podstawowe znaczenie dla konstrukcji algorytmów wyznaczania tras i efektywność ich działania mają:

- kryteria optymalizacji stosowane do wyboru trasy; koszt trasy jest bezpośrednio funkcją długości trasy i ilości udostępnianej pojemności, a pośrednio funkcją kryteriów optymalizacji,

- struktura topologiczna sieci; długość i liczba alternatywnych tras zależy od rozmiaru sieci, tzn. liczby węzłów i liczby łączy,

- sposób wywoływania algorytmu; algorytm wyznaczania tras może być inicjowany wywołaniem lub zmianami stanu sieci.

Jednym ze sposobów wyróżniania klas algorytmów wyznaczania tras, spełniających wymagania z zakresu jakości usług, jest dokładność algorytmów i sposoby wywoływania algorytmów. Na tej podstawie można wyróżnić algorytmy:

- wyznaczania dokładnych tras z góry,

- wyznaczania przybliżonych tras z góry,
- wyznaczania dokładnych tras na żądanie.

Wyznaczanie dokładnych tras z góry polega na tym, że dla danej struktury topologicznej sieci i metryk łącz jednorazowo są wyznaczane wszystkie możliwe trasy. Kolejne iteracje algorytmu są uruchamiane zmianami w strukturze topologicznej sieci lub zmianami zasobów. Zakończenie działania algorytmu oznacza, że możliwe jest wskazanie trasy do każdego z ujść i dla każdej żądanej wartości pojemności trasy. Jeżeli do wyznaczania tras stosuje się algorytm Bellmana–Forda, to wyznaczone rozwiązanie jest rozwiązaniem dokładnym, a wyznaczone trasy są trasami o minimalnej długości, w sensie liczby węzłów na trasie. Najlepsza trasa do danego węzła ujściowego jest trasą o maksymalnej możliwej pojemności spośród wszystkich dopuszczalnych tras o minimalnej długości.

Wyznaczanie przybliżonych tras z góry jest możliwe m.in. za pomocą algorytmu Dijkstry. Przybliżone, a nie dokładne, rozwiązanie zadania wyznaczenia tras z góry dla sieci o zadanej strukturze topologicznej i zasobach jest konsekwencją działania algorytmu. Utrata dokładności wynika z konieczności kwantowania wartości pojemności, które są brane pod uwagę w czasie wyznaczania tras o minimalnej długości. Zakres wartości pojemności, które mogą być żądane przez nowe wywołania, jest odwzorowywany na ustaloną niewielką liczbę skwantowanych wartości, dla których wyznaczane są najkrótsze trasy.

Algorytmy wyznaczania dokładnych tras na żądanie są wywoływane żądaniem obsługi nowego przepływu. Uzasadnieniem wyznaczania trasy na żądanie jest albo niewielka liczba obsługiwanych przepływów, albo żądana duża dokładność rozwiązania zadania optymalizacji. Dokładność wyznaczania trasy na żądanie wynika z możliwości uwzględnienia aktualnych metryk zasobów sieci. Dodatkową korzyścią wyznaczania tras na żądanie jest zbędność utrzymywania tablic routingu w węzłach sieci.

Przydatność poszczególnych grup algorytmów w danej sieci zależy od rozmiaru sieci, dokładności informacji (metryk) o stanie zasobów, szybkości zachodzenia zmian w sieci, liczby i czasu trwania przepływów itd. Złożoność obliczeniowa algorytmów wyznaczania tras na żądanie jest uzasadniona dokładnością rozwiązania. W przypadku algorytmów wyznaczania tras z góry złożoność obliczeniowa algorytmu jest kompensowana natychmiastowym dostępem do tablic routingu w razie napływu nowego żądania obsługi przepływu.

11.3.3. ZADANIE WYZNACZANIA TRAS Z OGRANICZENIAMI

Zadania wyznaczania tras z ograniczeniami są rozszerzeniem zadań wyznaczania tras o zadanej jakości usług (ang. *QoS routing*) w tym znaczeniu, że uwzględniane są także inne niż tylko jakość usług ograniczenia. Przykładami takich ograniczeń są np. dostępność zasobów na trasie, niezawodność trasy i konieczność równoważenia obciążeń. Celem rozwiązania zadania wyznaczania tras z ograniczeniami jest znalezienie tras spełniających wymagania jakościowe i zwiększających stopień wykorzystania

zasobów sieci. Minimalizacja maksymalnego stopnia wykorzystania zasobów sieci prowadzi do optymalnego wykorzystania zasobów sieci w wyniku takiego rozłożenia ruchu w sieci, które równoważy obciążenie jej zasobów (ang. *load balancing*).

Zadanie inżynierii ruchu jest zwykle przeciwstawiane zadaniu sterowania ruchem w sieciach, w których wybór trasy jest rozwiązaniem jednokryterialnego zadania optymalizacji przy założeniu niepodzielności ruchu identyfikowanego parą węzłów: źródła i ujścia oraz nieuwzględniania różnych charakterystyk i wymagań obsługiwanego ruchu. Wyznaczenie trasy przez optymalizację wskaźnika jakości ściśle związanego ze strukturą topologiczną sieci (np. najkrótsza trasa w sensie liczby węzłów na trasie lub opóźnienia wnoszonego przez trasę) prowadzi do rozwiązań, w których trasy przekazywania jednostek danych do ustalonych węzłów ujściowych są niezmiennie. Może to prowadzić do przypadków, kiedy trasy optymalne, ze względu na wykorzystywane metryki, są silnie obciążone, gdy inne trasy pozostają nieobciążone, mimo dużego obciążenia całej sieci. Z tego powodu zadaniem inżynierii ruchu jest równoważenie obciążeń elementów sieci z uwzględnieniem różnic jakości usług żądanych dla przepływów współdzielących zasoby sieci.

11.3.3.1. OGÓLNE SFORMUŁOWANIE ZADANIA WYZNACZANIA TRAS Z OGRANICZENIAMI

Implementacja tras z ograniczeniami w sieci oznacza realizację dwuetapowej procedury: wyznaczenia optymalnych tras i wymuszania przekazywania ruchu na wyznaczonych trasach.

Dana jest sieć, dla której modelem struktury topologicznej jest graf skierowany $G = (N, \Gamma)$, gdzie N jest zbiorem węzłów sieci, a Γ jest zbiorem łuków. Liczba węzłów i łuków jest, odpowiednio, równa $N = |N|$ oraz $M = |\Gamma|$. Założenie o grafie skierowanym oznacza, że łącza sieci są skierowane, tzn. łącza (i, j) oraz (j, i) są różne.

Zadanie optymalizacji polega np. na tym, aby dla znanej macierzy ruchu $R = [r_{mn}]_{N \times N}$, gdzie r_{mn} jest ilością ruchu wymianianą pomiędzy systemami końcowymi m -tym oraz n -tym, wyznaczyć takie trasy, które pozwolą na obsługę całego generowanego ruchu i zminimalizują maksymalny stopień wykorzystania łączy sieci.

Po przyjęciu, że:

- C_{ij} – pojemność łącza skierowanego (i, j) dla każdego $(i, j) \in \Gamma$,
- k – numer składnika ruchu w sieci, gdzie $k \in \{1, 2, \dots, N \times N\}$, a każdy składnik ruchu jest identyfikowany węzłem źródłowym, ujściowym i ilością ruchu, tzn. odpowiednio: s_k ($s_k \in \Gamma$), t_k ($t_k \in \Gamma$, $t_k \neq s_k$) oraz r_k ,
- f_{ij}^k – część ilości ruchu k -tego składnika przechodzącego przez łącze (i, j) (jeżeli spełniony jest warunek: $0 \leq f_{ij}^k \leq 1$, to k -ty składnik może być przekazywany wieloma trasami, jeśli natomiast spełniony jest warunek: $f_{ij}^k = 0$ lub $f_{ij}^k = 1$, to k -ty składnik może być przekazywany tylko jedną trasą),

- każdy składnik ruchu może być przekazywany więcej niż jedną trasą,
- α_{ij} – wartość współczynnika wykorzystania pojemności łącza (i, j) ,

to sformułowanie zadania wyznaczania optymalnych tras w sieci jest następujące:

$$\alpha^* = \min \alpha = \min \left[\max_{(i,j) \in \tilde{A}} \{\alpha_{ij}\} \right],$$

przy ograniczeniach:

$$\sum_{j \in \{1, 2, \dots, N\}, j \neq i} f_{ij}^k - \sum_{j \in \{1, 2, \dots, N\}, j \neq i} f_{ji}^k = \begin{cases} 1 & \text{wtedy, gdy } i = s_k \\ 0 & \text{wtedy, gdy } i \neq s_k, i \neq t_k \\ -1 & \text{wtedy, gdy } i = t_k \end{cases}$$

dla każdego $k \in \{1, 2, \dots, N \times N\}$,

$$\sum_{k \in \{1, 2, \dots, N \times N\}} \gamma_k f_{ij}^k \leq \alpha C_{ij} \quad \text{dla każdego } (i, j) \in \Gamma,$$

$$0 \leq f_{ij}^k \leq 1 \text{ oraz } \alpha \geq 0.$$

Rozwiązaniem zadania optymalizacji jest zbiór optymalnych tras dla zapotrzebowania danego macierzą ruchu R . Ze względu na to, że poszczególne składniki ruchu mogą być przekazywane wieloma trasami, w węzłach sieci muszą być realizowane funkcje rozdziału poszczególnych składników całkowitego przepływu.

Rozdział jednego składnika ruchu na poszczególne, dostępne w sieci, trasy jest zagadnieniem złożonym. W szczególności nie jest pożądane, aby jednostki danych tego samego przepływu elementarnego, należące do danego składnika, były przekazywane różnymi trasami. W takim przypadku różne opóźnienia wnoszone przez różne trasy oznaczają konieczność odtwarzania kolejności i eliminowanie fluktuacji opóźnienia.

Rozwiązanie opisanego wyżej zadania optymalizacji nie jest równoważne rozwiązaniu zadania równoważenia obciążeń.

11.3.3.2. ZŁOŻONOŚĆ OBLICZENIOWA ZADAŃ WYZNACZANIA TRAS Z OGRANICZENIAMI

Algorytmy wyznaczania optymalnych tras z ograniczeniami, tzn. wyznaczenia elementów tablic tras (tablic routingu) w węzłach sieci (routerach) oraz ich złożoność obliczeniowa zależą od wyboru kryteriów zadania optymalizacji.

W zadaniach wyznaczania optymalnych tras do wyboru tras i szacowania skutków ich wykorzystania dla jakości usług w sieci stosowane są różne kryteria, należące najczęściej do jednej z trzech grup wskaźników jakości: addytywnych (np. opóźnienie, wariancja opóźnienia, liczba łączy na trasie, koszt transmisji), multiplikatywnych (np. niezawodność) i wklęsłych (np. przepustowość).

Złożoność obliczeniowa zadań wyznaczania optymalnych tras zależy od użytego kryterium oraz liczby i rodzaju ograniczeń uwzględnianych w zadaniu. Poza nielicznymi przypadkami są to zadania *NP-trudne*.

Względnie proste są te zadania optymalizacji, w których ograniczenia dotyczą pojemności i długości tras, wyrażanych liczbą węzłów na trasie pomiędzy systemami końcowymi. Przydatność rozwiązań tak sformułowanych zadań optymalizacji wynika m.in. z charakteru zależności innych miar jakości, takich jak opóźnienie i wariancja opóźnienia, od pojemności i długości tras w sieci oraz możliwości ich wzajemnego odwzorowania.

Rozwiązania zadań wyznaczania tras mogą być albo wyznaczone z góry dla każdej z możliwych klas ruchu w sieci, albo wyznaczone na żądanie.

Obciążenie routerów zadaniami wyznaczania tras z ograniczeniami, struktura i wielkość utrzymywanych i aktualizowanych tablic tras oraz stabilność sieci zależą od typu routingu oraz kryterium i liczby ograniczeń zadania optymalizacji.

Wybrany dla danej sieci typ routingu determinuje poziom granulacji routingu (ang. *routing granularity*). W zależności od tego, czy wyznaczane trasy są zorientowane na ujścia (ang. *destination-based*), pary źródło–ujście (ang. *source-destination-based*), przepływy (ang. *flow-based*), magistrale (ang. *traffic-trunk-based*) lub klasy ruchu (ang. *traffic-class-based*) różna jest liczba zadań routingu jednocześnie rozwiązywanych w węzłach sieci oraz efektywność wykorzystania zasobów i stabilność sieci. Największa efektywność wykorzystania zasobów jest możliwa wtedy, gdy granulacja routingu jest największa, tzn. wtedy, gdy granulacja ruchu (ang. *traffic granularity*) obsługiwanego w sieci jest największa; w takim przypadku trasa jest wyznaczana niezależnie dla każdego elementarnego przepływu w sieci. Jednocześnie wzrost granulacji routingu oznacza zwiększenie ilości niezbędnych zasobów obliczeniowych, komunikacyjnych i pamięci. Złożoność obliczeniowa zadań wyznaczania tras z ograniczeniami może być zredukowana m.in. przez zmniejszanie częstotliwości aktualizacji tablic tras oraz ograniczanie zakresu zadań optymalizacji do wybranych fragmentów sieci.

Struktura i rozmiar tablic tras w węzłach sieci zależą bezpośrednio od poziomu granulacji routingu; wzrost granulacji powoduje zwiększenie rozmiaru tablic i tym samym np. wzrost pojemności pamięci oraz wydłużenie czasu przeszukiwania tablic. Redukcja rozmiarów tablic jest możliwa przez założenie pośrednich poziomów granulacji routingu, kwantowanie wartości miar tras lub pośrednie wyznaczanie tras z ograniczeniami. Pośrednie wyznaczanie tras oznacza, że w węzłach sieci utrzymywane są tablice tras dla obsługi bez gwarancji jakości, a trasy z ograniczeniami wyznaczane są na żądanie.

11.3.4. WYZNACZANIE TRASY Z GWARANCJĄ JAKOŚCI USŁUG

Wyznaczanie tras z gwarancją jakości usług (ang. *QoS routing*) jest na ogół zadaniem wielokryterialnego wyznaczania optymalnych tras. Rozwiązanie omawianego zadania jest zwykle procesem wieloetapowym – określa się dopuszczalne trasy, spo-

śród których jest wyznaczana trasa (trasy) optymalna w sensie przyjętego (przyjętych) wskaźników jakości obsługi poszczególnych składowych ruchu w sieci. O rozwiązaniu zadania optymalizacji, a tym samym o jakości usług, decyduje liczba kryteriów i ograniczeń oraz sposób dekompozycji zadania.

Przykładami tak sformułowanych zadań są m.in. zadania wyznaczania tras najszerszych i najkrótszych (ang. *widest-shortest*), najkrótszych i najszerszych (ang. *shortest-widest*), najszybszych spośród dopuszczalnych (ang. *fastest-feasible*).

W przypadku zadania wyznaczania tras najkrótszych i najszerszych w pierwszej kolejności są wyznaczane trasy najkrótsze, a wtedy, gdy istnieje wiele takich tras, wybierana jest spośród najkrótszych taka, której dostępna przepustowość jest największa.

Rozwiązanie zadania wyboru najszerszej i najkrótszej oznacza, że wyznaczane są trasy o największej dostępnej przepustowości, w przypadku gdy takich tras jest więcej niż jedna, wybierana jest najkrótsza spośród nich.

Wybór najszybszej spośród dopuszczalnych tras oznacza, że w pierwszym etapie wyznaczane są trasy dopuszczalne, tzn. spełniające zadane ograniczenia, a następnie spośród dopuszczalnych wyznaczana jest trasa zapewniająca minimalne opóźnienie.

11.3.4.1. MODEL SIECI I TRASY W SIECI

Dana jest sieć, dla której modelem struktury topologicznej jest graf $G = (N, \Gamma)$, gdzie N jest zbiorem węzłów sieci, a Γ jest zbiorem łączy. Liczba węzłów i łączy jest, odpowiednio, równa $N = |N|$ oraz $M = |\Gamma|$. Długość najdłuższej trasy w sieci jest równa $h_{\max} \leq N - 1$.

Każde z łączy $k \in \Gamma$ jest opisane następującymi parametrami:

- \bar{C}_k – pojemność k -tego łącza, tj. maksymalna szybkość transmisji w k -tym łączy,
- C_k – dostępna pojemność k -tego łącza, tzn. maksymalna szybkość transmisji, jaka może być udostępniona dla nowego połączenia (żądania zasobów transmisyjnych); $C_k \leq \bar{C}_k$ dla każdego $k \in \Gamma$,
- d_k – opóźnienie propagacji k -tego łącza.

Każde i -te połączenie charakteryzowane jest ponadto następującymi parametrami:

- s_i, t_i – numery – odpowiednio – źródła i ujścia i -tego połączenia,
- b_i – maksymalny rozmiar paczki jednostek danych (pojemność bufora sterownika z algorytmem ciekącego wiadra),
- $l_{i \max}$ – maksymalna długość jednostki danych,
- D_i – górne ograniczenie maksymalnego opóźnienia całkowitego jednostki danych,
- $v_{i \max}$ – żądana maksymalna szybkość transmisji dla i -tego połączenia ($v_{i \max} \geq v_i$).

Jeżeli i -temu połączeniu zostanie udostępniona trasa, przechodząca przez k -te łącze, to następuje zmiana wartości dostępnej pojemności, tzn. $C_k \rightarrow C_k - v_i$, gdzie v_i jest szybkością transmisji i -tego połączenia; $v_i \leq C_k$ dla każdego $k \in \Gamma$.

Dane i -te połączenie może być przekazane jedną z tras $\pi_{ij} \in \Pi_i$ pomiędzy węzłem źródłowym s_i i węzłem ujściowym t_i , gdzie Π_i ($\Pi_i \neq \emptyset$) jest zbiorem wszystkich tras dla i -tego połączenia.

Po oznaczeniu:

$n(\pi_{ij})$ – długość trasy π_{ij} , wyrażona liczbą pojedynczych łączy na trasie π_{ij} ,

$v(\pi_{ij})$ – maksymalna szybkość transmisji dostępna na trasie π_{ij} , to znaczy

$$v(\pi_{ij}) = \min_{k \in \pi_{ij}} C_k,$$

oraz przyjęciu, że:

➤ w węzłach sieci implementowana jest sprawiedliwa dyscyplina szeregowania, odwzorowująca górne ograniczenie maksymalnego opóźnienia w maksymalną szybkość transmisji,

➤ i -te połączenie jest przekazywane trasą π_{ij} , spełniającą warunek: $v_i \leq v(\pi_{ij})$,

górne ograniczenie maksymalnego opóźnienia całkowitego jednostki danych, wnoszonego przez trasę $\pi_{ij} \in \Pi_i$ jest określone wyrażeniem:

$$D_i(\pi_{ij}, v_i) = \frac{b_i + n(\pi_{ij})l_{i \max}}{v(\pi_{ij})} + \sum_{k \in \pi_{ij}} d_k.$$

Wartość $D_i(\pi_{ij}, v_i)$ jest wartością gwarantowanego maksymalnego opóźnienia całkowitego na trasie $\pi_{ij} \in \Pi_i$ dla i -tego połączenia.

Trasa $\pi_{ij} \in \Pi_i$ jest trasą dopuszczalną (ang. *feasible path*) dla i -tego połączenia wtedy, gdy $D_i(\pi_{ij}, v_i) \leq D_i$ oraz $v(\pi_{ij}) \geq v_{i \max}$. Zbiór wszystkich tras dopuszczalnych dla i -tego połączenia $\bar{\mathcal{D}}_i$ jest podzbiorem zbioru wszystkich tras dla i -tego połączenia, tzn. $\bar{\mathcal{D}}_i \subset \mathcal{D}_i$. Trasa dopuszczalna $\pi_{ip} \in \bar{\mathcal{D}}_i$ jest trasą najszybszą (ang. *fastest path*), gdy $D_i(\pi_{ip}, v_i) \leq \min_{j \in I_i} D_i(\pi_{ij}, v_i)$, gdzie I_i jest zbiorem indeksów wszystkich tras

dla i -tego połączenia. Dane i -te połączenie jest połączeniem dopuszczalnym wtedy, gdy zbiór tras dopuszczalnych dla tego połączenia nie jest zbiorem pustym, tzn. $\bar{\mathcal{D}}_i \neq \emptyset$.

11.3.4.2. ZNAJDOWANIE TRAS DOPUSZCZALNYCH I OPTYMALNYCH

Jeżeli liczba wszystkich połączeń w sieci jest równa I , to ogólne zadanie znajdowania tras dopuszczalnych, tzn. zadanie znajdowania tras dopuszczalnych przepływu wieloskładnikowego MFP (ang. *Multicommodity Feasible Path*), jest następujące: dla każdego i -tego połączenia ($0 \leq i \leq I$) wyznaczyć takie trasy π_{ij} , które spełniają warunki:

$$\frac{b_i + n(\pi_{ij})l_{i \max}}{v(\pi_{ij})} + \sum_{k \in \pi_{ij}} d_k \leq D_i,$$

$$\sum_{\{i|k \in \pi_{ij}\}} v_i \leq C_k \text{ dla każdego } k \in \tilde{A},$$

$$v(\pi_{ij}) \geq v_{i \max},$$

$$v_i = v_{\min}(\pi_{ij}) \leq v(\pi_{ij}).$$

Interpretacja podanych warunków jest następująca:

➤ opóźnienie wnoszone przez trasy π_{ij} dla ruchu generowanego w ramach i -tego połączenia jest nie większe od zadanego górnego ograniczenia maksymalnego opóźnienia całkowitego jednostki danych D_i ,

➤ suma szybkości wszystkich połączeń przechodzących przez każde z łączy sieci jest nie większa od dostępnej pojemności tego łącza,

➤ maksymalna szybkość transmisji dostępna na trasie dla i -tego połączenia jest nie mniejsza od maksymalnej szybkości transmisji żądanej przez i -te połączenie,

➤ maksymalna szybkość transmisji dostępna na trasie dla i -tego połączenia jest nie mniejsza od szybkości transmisji i -tego połączenia.

Zadanie MFP jest zadaniem NP -trudnym.

Rozwiązanie zadania MFP nie jest rozwiązaniem zadania wyznaczenia optymalnej trasy dla zbioru połączeń. Rozwiązanie optymalne jest znajdowane w wyniku przeszukiwania zbioru tras dopuszczalnych.

11.3.4.3. KRYTERIA WYBORU TRASY OPTYMALNEJ

Wyznaczenie rozwiązania optymalnego wymaga przyjęcia takiego kryterium wyboru trasy, jak np. równoważenie obciążenia zasobów, profilowanie przepływu, współczynnik zawłaszczania szybkości transmisji.

Przykładem kryterium wyboru optymalnej trasy jest współczynnik zawłaszczania szybkości transmisji (ang. *rate consumption rate*). Wybór takiego wskaźnika wynika z obserwacji, że w przypadku zastosowania sterownika z algorytmem ciekącego wiadra szybkość transmisji przydzielona dla i -tego połączenia nie jest ustalana a priori, ale zależy od wyboru trasy i jej przepustowości.

Współczynnik zawłaszczania szybkości transmisji na dopuszczalnej trasie π_{ij} jest definiowany jako różnica pomiędzy szybkością transmisji i -tego połączenia v_i a szybkością transmisji $v(\pi_{ij})$ udostępnianą dla i -tego połączenia na trasie π_{ij} , tzn.:

$$\Delta v(\pi_{ij}) = v(\pi_{ij}) - v_i = v(\pi_{ij}) - \frac{b_i + n(\pi_{ij})l_{i \max}}{D_i - \sum_{k \in \pi_{ij}} d_k} = \min_{k \in \pi_{ij}} C_k - \frac{b_i + n(\pi_{ij})l_{i \max}}{D_i - \sum_{k \in \pi_{ij}} d_k}.$$

Ze względu na to, że szybkość transmisji na dopuszczalnej trasie π_{ij} jest równa minimalnej wartości dostępnych pojemności w łączach tworzących dopuszczalną trasę π_{ij} , współczynnik zawłaszczania szybkości transmisji na dopuszczalnej trasie π_{ij} można zastąpić sumą współczynników zawłaszczania szybkości transmisji w poszczególnych łączach dopuszczalnej trasy π_{ij} , tzn.:

$$\bar{\Delta}v(\pi_{ij}) = \sum_{k \in \pi_{ij}} \left[C_k - \frac{b_i + n(\pi_{ij})l_{i\max}}{D_i - \sum_{k \in \pi_{ij}} d_k} \right].$$

Na podstawie jednego z wymienionych kryteriów możliwe jest wybranie optymalnej trasy w zbiorze tras dopuszczalnych; p -ta trasa dopuszczalna π_{ip} ($\pi_{ip} \in \bar{\mathcal{D}}_i$) jest optymalna wtedy, gdy spełniony jest warunek:

$$\Delta v(\pi_{io}) = \min \{ \Delta v(\pi_{ij}) \mid \pi_{ij} \in \bar{\mathcal{D}}_i \} \quad (\text{lub warunek } \bar{\Delta}v(\pi_{io}) = \min \{ \bar{\Delta}v(\pi_{ij}) \mid \pi_{ij} \in \bar{\mathcal{D}}_i \}).$$

Innym możliwym kryterium wyboru trasy optymalnej spośród tras dopuszczalnych jest kryterium równoważenia obciążenia. Sposób, w jaki szybkość transmisji $v(\pi_{ij})$ trasy dopuszczalnej π_{ij} wpływa na dopuszczalną szybkość transmisji w łączach tej trasy zależy od chwilowych szybkości transmisji w tych łączach. Z tego względu lepszą miarą równoważenia obciążenia sieci jest taka, która bierze pod uwagę raczej względny niż bezwzględny wpływ nowego połączenia na dostępną pojemność każdego łącza trasy.

Przykładową miarą równoważenia obciążenia może być maksymalna wartość ilorazu minimalnej szybkości transmisji $v_{\min}(\pi_{ij})$ ($v_{\min}(\pi_{ij}) = v_i$) na dopuszczalnej trasie π_{ij} oraz wartości dostępnych pojemności w łączach tworzących dopuszczalną trasę π_{ij} , tzn.:

$$\delta(\pi_{ij}) = \max_{k \in \pi_{ij}} \frac{v_{\min}(\pi_{ij})}{C_k} = \max_{k \in \pi_{ij}} \frac{1}{C_k} \times \frac{b_i + n(\pi_{ij})l_{i\max}}{D_i - \sum_{k \in \pi_{ij}} d_k}.$$

W przypadku omawianego kryterium równoważenia obciążeń p -ta trasa dopuszczalna π_{ip} ($\pi_{ip} \in \bar{\Pi}_i$) jest optymalna wtedy, gdy na trasie π_{ip} istnieje takie łącze l ($l \in \pi_{ip}$), dla którego spełniony jest warunek:

$$C_l \times \frac{D_i - \sum_{k \in \pi_{ip}} d_k}{b_i + n(\pi_{ip})l_{i\max}} = \max_{\pi_{ij} \in \bar{\mathcal{D}}_i} \min_{k \in \pi_{ij}} \left[C_k \times \frac{D_i - \sum_{k \in \pi_{ij}} d_k}{b_i + n(\pi_{ij})l_{i\max}} \right].$$

11.4. RÓWNOWAŻENIE OBCIĄŻEŃ

Jednym z podstawowych ograniczeń efektywności sterowania przepływem jest nierównomierność obciążenia zasobów sieci.

Podstawowe przyczyny nierównomierności obciążenia zasobów sieci to:

- stosowanie algorytmów rozdziału zasobów, w których nie jest wykorzystywana informacja o aktualnym stanie zasobów sieci,
- brak mechanizmów zrównoleglenia obsługi przepływów i klas ruchu.

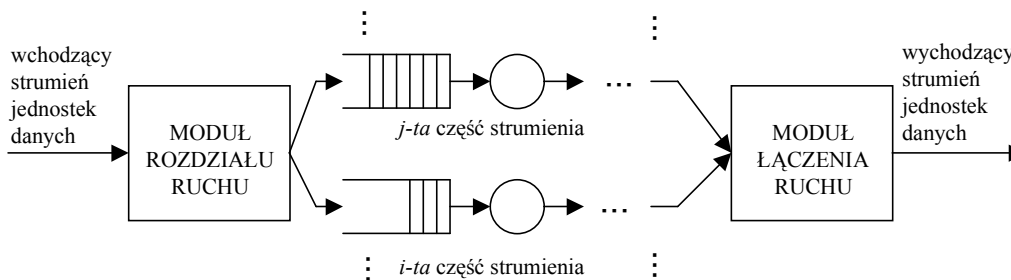
Nierównomierność obciążenia zasobów zwiększa się wraz ze wzrostem wolumenu ruchu związanego z przepływem lub klasą ruchu.

Przeciwdziałanie nierównomierności obciążenia ma szczególne znaczenie w modelu usług zróżnicowanych, ze względu na to, że obsługiwane klasy ruchu są, z założenia, agregatami pojedynczych przepływów. Udział ruchu względnie małej liczby poszczególnych klas w całości ruchu obsługiwanego w sieci jest na tyle duży, że statyczne i deterministyczne kierowanie ruchem może prowadzić do niskiej efektywności wykorzystania zasobów i degradacji jakości obsługi ruchu.

Zagadnienie równoważenia obciążeń ma mniejsze znaczenie w modelu usług zintegrowanych, gdzie przedmiotem sterowania są pojedyncze przepływy. Jeżeli maleje względny udział pojedynczego przepływu w ruchu obsługiwanym w sieci, to zaburzenia równomierności obciążenia zasobów, powodowane przyjęciem do obsługi pojedynczego przepływu, są względnie małe.

Jednym ze sposobów wymuszania równoważenia obciążeń jest podział obciążenia. Jeżeli wolumen ruchu nowo przyjmowanego do obsługi przepływu jest względnie duży, to skierowanie tego przepływu do obsługi przez tylko jedną z tras sieci może powodować znaczne dysproporcje obciążenia różnych tras pomiędzy tymi samymi węzłami końcowymi. Rozdział danego przepływu na części i skierowanie tych części przepływu do obsługi przez równoległe trasy pozwala równoważyć obciążenie za cenę rozbudowania systemu zarządzania ruchem w sieci.

Podstawowe funkcje modułu rozdzielania ruchu (ang. *traffic splitter*) to podział strumienia jednostek danych, wchodzącego do węzła sieci, na strumienie składowe wychodzące z danego węzła (rysunek 11.2).



Rysunek 11.2. Rozdział i łączenie ruchu w zadaniu równoważenia obciążeń

Kryteria rozdziału oryginalnego strumienia jednostek danych mogą być różne – od prostych, w których rozdział jest proporcjonalny do liczby strumieni składowych, do złożonych, w których podstawą rozdziału jest obciążenie poszczególnych alternatywnych tras. Podział może być statyczny, tzn. podział strumienia na części jest stały w czasie trwania przepływu, lub dynamiczny, tzn. udział poszczególnych części składowych przepływu zmienia się w czasie.

Realizacja rozdziału strumienia jednostek danych wiąże się z koniecznością uwzględnienia następujących wymagań:

- ilość informacji, na podstawie której podejmowane są decyzje, musi być minimalna, ze względu na to, że procedura rozdziału strumienia jednostek danych jest wykonywana dla każdej z jednostek danych,
- rozkład ruchu w poszczególnych strumieniach składowych musi być stabilny, ze względu na efektywność wykorzystania zasobów,
- w węzle ujściowym konieczne jest odtwarzanie oryginalnej kolejności jednostek danych, ze względu na różne opóźnienia wnoszone przez poszczególne trasy obsługujące różne części przepływu.

11.4.1. ALGORYTMY ROZDZIELANIA PRZEPLÝWÓW

Rozdział strumienia jednostek danych na strumienie składowe jest na ogół zadaniem formułowanym w celu rozwiązywania konfliktów żądań dostępu w systemach wielodostępu.

Wielodostępny system komunikacyjny można interpretować jako system kolejek. W każdym z węzłów organizowana jest kolejka jednostek danych, oczekujących na transmisję w wielodostępnym kanale transmisyjnym, który jest wspólnym serwerem. Ze względu na serwer wszystkie oczekujące na obsługę jednostki danych tworzą kolejkę obsługiwaną zgodnie z odpowiednią dyscypliną kolejkowania. W rzeczywistych przypadkach serwer nie dysponuje informacją o tym, w którym z węzłów znajdują się jednostki danych oczekujące na obsługę, oraz – podobnie – poszczególne węzły nie mają informacji o stanie innych węzłów. Oznacza to, że informacja o stanie sumarycznej kolejki jest rozproszona.

Pośród wielu są dwie skrajne strategie obsługi podanego zadania. Jedną z nich to strategia współzawodnictwa bez ograniczeń (ang. *free-for-all sharing*), zgodnie z którą wszystkie węzły transmitują nowe jednostki danych natychmiast, zakładając że nie wystąpi interferencja z jednostkami danych innych węzłów. Zasadniczym problemem w tej strategii jest sposób retransmisji jednostek danych wtedy, gdy zdarzy się kolizja jednostek danych różnych węzłów (interferencja). Drugą to strategią idealnego szeregowania (ang. *perfectly scheduled*), w której zgodnie z przyjętą regułą (np. obsługa cykliczna), poszczególnym węzłom są przydzielane rozłączne przedziały czasu w kanale transmisyjnym. W tym przypadku odpowiedzi wymagają następujące pytania: jak ustalana jest kolejność obsługi węzłów?, jak długie są przedziały czasu dla poszczególnych węzłów? oraz jak węzły są informowane o możliwości transmisji?

Realizacja strategii współzawodnictwa bez ograniczeń oznacza konieczność implementacji algorytmu rozwiązywania kolizji żądań dostępu. W szczelinowej wersji systemu obsługi istnieje wiele sposobów rozwiązywania kolizji przez węzły w niej uczestniczące, tzn. sposobów podejmowania decyzji o transmisji lub jej zaniechaniu w kolejnych szczelinach czasu. W najprostszym przypadku decyzja taka może być losowana. Innym sposobem jest wykorzystanie do rozwiązania kolizji różnic czasów

napływu kolidujących jednostek danych. Jeszcze innym sposobem jest rozstrzygnięcie kolizji na podstawie adresów węzłów; przy założeniu, że zbiór węzłów o unikatowych adresach binarnych jest skończony, wynik porównywania kolejnych bitów adresu może być kryterium wyboru węzła. Ostatnie z rozwiązań ma tę zaletę, że liczba szczelin czasowych potrzebna do rozwiązania kolizji jest ograniczona, ze względu na to, że adresy każdej pary węzłów różnią się co najmniej jednym bitem.

Wszystkie wyżej wymienione rozwiązania mają jedną wspólną właściwość: zbiór kolidujących węzłów jest rozdzielany na podzbiory, z których jeden transmituje w kolejnej szczeliny czasu. Jeżeli kolizja nie jest rozwiązana, tzn. jeżeli kolidujące węzły należą do tego samego podzbioru, to dany podzbiór jest dzielony na podzbiory. Algorytmy o takiej właściwości są określane mianem algorytmów rozdziału (ang. *splitting algorithms*).

11.4.2. ALGORYTMY ROZDZIAŁU O STRUKTURZE DRZEWA

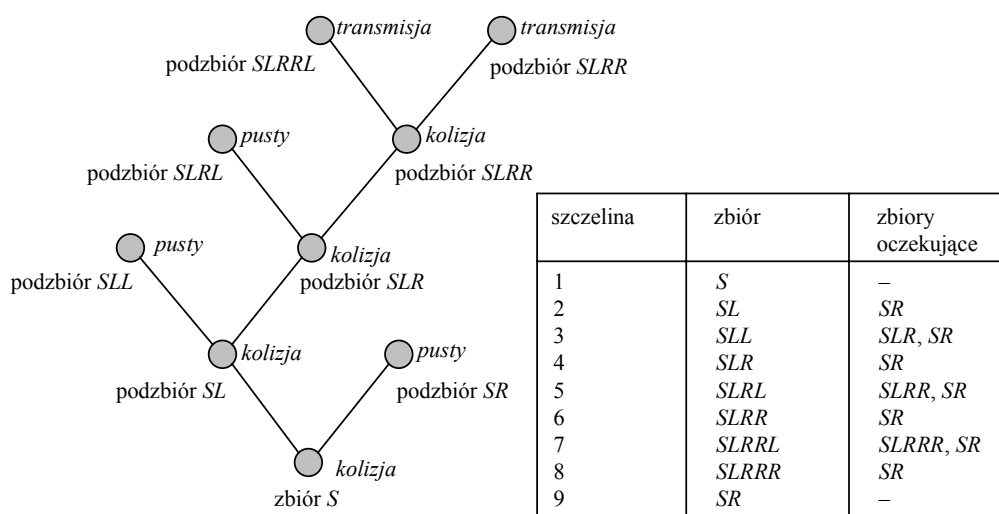
Pierwszymi algorytmami rozdziału były algorytmy o strukturze drzewa. Jeżeli zdarzy się kolizja, np. w k -tej szczeliny czasowej, to wszystkie węzły nie uczestniczące w kolizji przechodzą w stan oczekiwania, a zbiór wszystkich węzłów uczestniczących w kolizji jest dzielony na dwa podzbiory, np. na podstawie wyników losowania. Węzły pierwszego z podzbiorów transmitują w $(k + 1)$ -szej szczeliny czasowej, a jeżeli ta szczelina jest wolna (brak transmisji) lub transmisja jest poprawna, tzn. bez kolizji, to w $(k + 2)$ -giej szczeliny transmitują węzły drugiego podzbioru. Jeżeli podczas transmisji w $(k + 1)$ -szej szczeliny wystąpi kolizja, to pierwszy z dwóch podzbiorów jest dzielony ponownie, a węzły drugiego podzbioru oczekują na rozwiązanie kolizji w pierwszym podzbiorze.

Działanie algorytmu można zilustrować drzewem binarnym (każdy wierzchołek drzewa reprezentuje test o dwu możliwych wynikach), którego elementy dotyczą zdarzeń będących konsekwencją sekwencji podziałów zbioru węzłów, tzn. braku transmisji (wolna szczelina), poprawnej transmisji w szczeliny i kolizji węzłów w szczeliny (rysunek 11.3).

W drzewie tym zbiór S , będący korzeniem drzewa, jest zbiorem węzłów uczestniczących w początkowej kolizji, a zbiory SL i SR są dwoma podzbiorami, powstałymi w wyniku podziału zbioru S . Podobnie SLL i SLR są rezultatem podziału podzbioru SL po tym, jak podzbiór SL wygenerował kolizję. Każdy z wierzchołków drzewa binarnego odpowiada podzbiorowi (być może pustemu) węzłów z jednostkami danych oczekującymi na transmisję. Z wierzchołków drzewa, którym odpowiadają węzły z dwiema lub więcej jednostkami danych, odchodzą dwie gałęzie, odpowiadające podziałowi podzbioru na dwa nowe podzbiory. Wierzchołki drzewa, którym odpowiadają węzły z jedną lub bez jednostki danych są wierzchołkami wiszącymi (rysunek 11.4).

Ograniczeniem algorytmu drzewa jest sposób obsługi jednostki danych żądającej dostępu w czasie rozwiązywania kolizji, tzn. w przedziale rozwiązywania kolizji (ang. *collision resolution period*). Oczekiwanie z obsługą nowej jednostki danych na zakoń-

czenie przedziału rozwiązywania kolizji, zwłaszcza wtedy, gdy długość przedziału rozwiązywania kolizji rośnie, powoduje wydłużenie przedziałów rozwiązywania kolejnych kolizji (efekt kuli śnieżowej). Rozwiązanie polega na tym, że po zakończeniu przedziału rozwiązywania kolizji zbiór węzłów oczekujących na transmisję jest dzielony na podzbiory, dla których kolizje rozwiązywane są sekwencyjnie. Liczba podzbiorów jest dobierana tak, aby średnia liczba węzłów kolidujących, przypadających na jeden podzbiór, była niewiele większa od jednego.



Rysunek 11.3. Drzewo binarne zadania rozwiązywania kolizji i podziału zbioru *S*

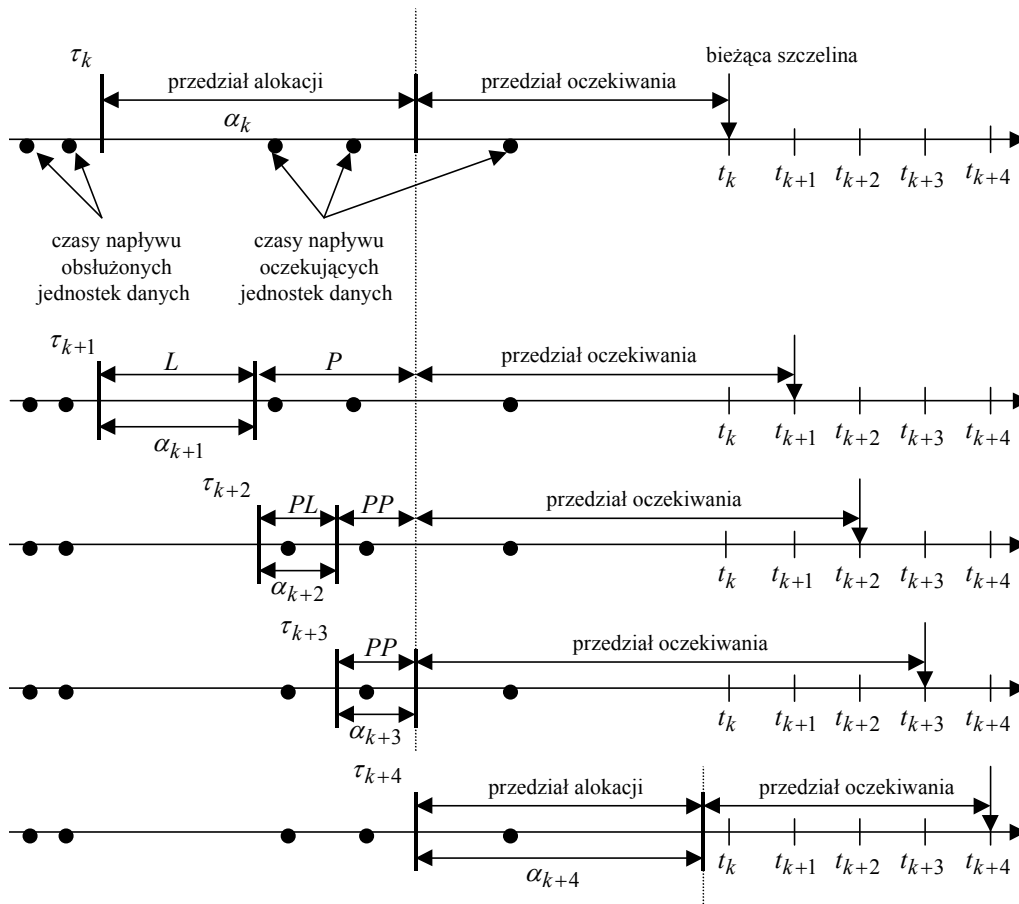
Maksymalna przepustowość, dostępna w systemie obsługi korzystającym z algorytmu drzewa, wynosi 0,43 jednostek danych w szczelinie. Co więcej, algorytm drzewa nie zachowuje kolejności napływu jednostek danych.

Wadą algorytmu drzewa jest m.in. wymaganie monitorowania kanału transmisyjnego i śledzenia czasu zakończenia obsługi każdej kolizji przez wszystkie węzły. Jednym ze sposobów ograniczenia tych niedogodności jest rozwiązanie polegające na tym, że nowe żądania dostępu są dołączane do zbioru oczekujących na transmisję. W takim przypadku zbiór węzłów monitorujących kanał transmisyjny jest ograniczony do zbioru węzłów, których jednostki danych oczekują na obsługę.

Względnie niska efektywność algorytmu drzewa wynika z tego, że rozwiązanie kolizji wymaga sprawdzenia wszystkich wierzchołków drzewa, w tym wierzchołków, którym odpowiada brak jednostek danych (np. wierzchołki *SLL* i *SLRL* na rysunku 11.3). Istnieje wiele propozycji rozwiązań zadania rozdziału przepływu, których celem jest poprawa efektywności algorytmu drzewa. Źródłem wzrostu efektywności jest wykorzystanie właściwości różnych algorytmów podziału zbiorów kolidujących węzłów na rozłączne podzbiory.

11.4.3. ALGORYTMY ROZDZIAŁU FCFS

Jednym z prostszych sposobów podziału zbioru kolidujących jednostek danych na rozłączne podzbiory jest podział na podstawie czasów napływu jednostek danych. W takim przypadku do danego podzbioru należą wszystkie jednostki danych, które napłynęły w pewnym danym przedziale czasu, a jeżeli zdarza się kolizja, to dany przedział jest dzielony na mniejsze przedziały. W każdym przypadku transmitowana jest jednostka danych o najmniejszym czasie napływu, tzn. jednostki danych są transmitowane w kolejności ich napływu; stąd nazwa algorytmu, tzn. FCFS (ang. *First Come First Served*).



Rysunek 11.4. Rozwiązywanie kolizji przez algorytm FCFS – lewa część przedziału alokacji bez kolizji

W każdej dyskretnej chwili czasu t_k , będącej wielokrotnością długości szczeliny Δ_k , algorytm FCFS wskazuje jednostki danych, które powinny być transmitowane w k -tej szczelinie czasowej, tzn. w przedziale czasu $[t_k, t_{k+1})$ i które napłynęły w pew-

nym wcześniejszym przedziale czasu $[\tau_k, \tau_k + \alpha_k)$. Jednostki danych, które napłynęły w przedziale $[\tau_k, \tau_k + \alpha_k)$, można traktować jako jednostki danych w obsłudze, jednostki danych napływające po czasie $\tau_k + \alpha_k$ tworzą natomiast kolejkę jednostek oczekujących na obsługę. Przedział czasu o długości α_k jest przedziałem alokacji (ang. *allocation interval*), a przedział o długości $t_k - (\tau_k + \alpha_k)$ jest przedziałem oczekiwania (ang. *waiting interval*). Przedziałom alokacji i oczekiwania odpowiadają zbiory węzłów: odpowiednio – zbiór alokacji i zbiór oczekiwania. Cechą charakterystyczną tak definowanej kolejki jednostek danych jest nieznaną liczbą jednostek danych w kolejce.

Algorytm rozdziału FCFS jest zbiorem reguł, zgodnie z którymi węzły wyznaczają wartości τ_k i α_k dla każdej kolejnej wartości k na podstawie znajomości poprzedniego stanu kolejki. Reguły te są modyfikacjami reguł obowiązujących w algorytmie drzewa.

Kolizja jednostek danych w k -tej szczelinie czasowej powoduje podział przedziału alokacji na dwa jednakowo długie przedziały czasu: $L = [\tau_k, \tau_k + \alpha_k/2)$ i $P = [\tau_k + \alpha_k/2, \tau_k + \alpha_k)$. Skrajny lewy przedział czasu $L = [\tau_k, \tau_k + \alpha_k/2)$ jest przedziałem alokacji dla $(k + 1)$ -szej szczeliny czasu, tzn. $\tau_{k+1} = \tau_k$ i $\alpha_{k+1} = \alpha_k/2$.

Jeżeli przedział alokacji dla $(k + 1)$ -szej szczeliny jest pusty (brak żądań dostępu), to oznacza, że w przedziale $P = [\tau_{k+1} + \alpha_{k+1}, \tau_{k+1} + 2\alpha_{k+1})$ są co najmniej dwie jednostki danych. Przedział $P = [\tau_{k+1} + \alpha_{k+1}, \tau_{k+1} + 2\alpha_{k+1})$ jest dzielony na dwa przedziały: $PL = [\tau_{k+1} + \alpha_{k+1}, \tau_{k+1} + 3\alpha_{k+1}/2)$ i $PP = [\tau_{k+1} + 3\alpha_{k+1}/2, \tau_{k+1} + 2\alpha_{k+1})$, tzn. $\tau_{k+2} = \tau_{k+1} + \alpha_{k+1}$ i $\alpha_{k+2} = \alpha_{k+1}/2$. Przedział PL jest przedziałem alokacji dla kolejnej $(k + 2)$ -giej szczeliny czasowej. Proces ten jest powtarzany do chwili rozwiązania aktualnej kolizji (rysunek 11.4).

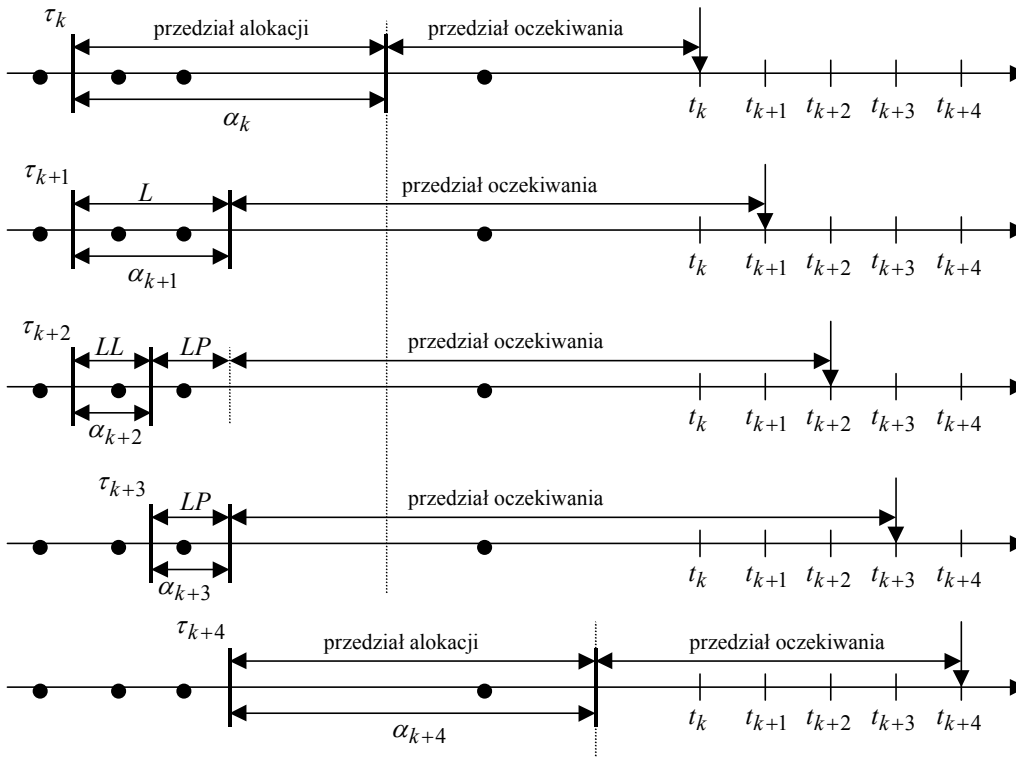
Jeżeli w przedziale alokacji dla $(k + 1)$ -szej szczeliny jest kolizja, to oznacza, że w przedziale alokacji $L = [\tau_k, \tau_k + \alpha_k/2) = [\tau_{k+1}, \tau_{k+1} + \alpha_{k+1})$ są co najmniej dwie jednostki danych. Przedział ten jest dzielony na dwa przedziały, tzn. przedział alokacji dla $(k + 2)$ -giej szczeliny $LL = [\tau_{k+1}, \tau_{k+1} + \alpha_{k+1}/2)$ oraz przedział oczekiwania $LP = [\tau_{k+1} + \alpha_{k+1}/2, \tau_{k+1} + \alpha_{k+1})$. Z podziału tego wynika, że: $\tau_{k+2} = \tau_{k+1}$ oraz $\alpha_{k+2} = \alpha_{k+1}/2$ (rysunek 11.5).

W algorytmie FCFS, w przeciwieństwie do algorytmu drzewa, zakończenie rozwiązywania kolejnej kolizji jest równoznaczne z podziałem wszystkich oczekujących jednostek danych na dwa podzbiory. Do wyznaczania tych podzbiorów wykorzystywane są wartości czasu rozpoczęcia bieżącej szczeliny czasowej i czasu zakończenia rozwiązania poprzedniej kolizji.

W omawianej wersji algorytmu FCFS przedziały alokacji i oczekiwania są sobie równe, podobnie jak podzbiory powstające w wyniku kolejnych podziałów przedziału alokacji. Zwiększenie przepustowości algorytmu FCFS jest możliwe przez wprowadzenie optymalnych podziałów podanych wyżej przedziałów.

Realizacja algorytmu rozdziału FCFS wymaga monitorowania kanału przez wszystkie węzły w każdej chwili czasu. Jedną z możliwych modyfikacji algorytmu FCFS polega na tym, że potrzeba monitorowania kanału transmisyjnego przez węzeł zachodzi tylko wtedy, gdy generowane jest żądanie transmisji jednostki danych.

Zgodnie z taką propozycją do obsługi nowej jednostki danych stosowany jest algorytm zbliżony do algorytmu LCFS (ang. *Last Come First Served*), tzn. do oszacowania czasu obsługi ostatnio napływającej jednostki danych potrzebna jest licznosc zbioru jednostek danych oczekujących na obsługę, po obsłudze którego dana jednostka danych będzie pierwszą w kolejce do obsługi.



Rysunek 11.5. Rozwiązywanie kolizji przez algorytm FCFS –prawa część przedziału alokacji bez kolizji

Jedną z odmian algorytmu FCFS jest algorytm, zgodnie z którym zbiór alokacji nie jest tworzony na podstawie wartości czasów napływu jednostek danych, ale na podstawie przynależności do danego podzbioru węzłów. Jest to algorytm obsługi cyklicznej RR (ang. *Round Robin*).

Jeżeli zbiór węzłów jest zbiorem skończonym i zawiera M węzłów, to wszystkie węzły mogą tworzyć cykl, w którym $m + 1$. węzeł jest poprzedzany przez m -ty węzeł, a M -ty węzeł poprzedza węzeł pierwszy. Zbiorem alokacji jest ciąg węzłów należących do cyklu. Po zakończeniu rozwiązywania kolizji dla danego ciągu alokacji następny zbiór alokacji jest ciągiem kolejnych węzłów cyklu. Rozmiar zbioru alokacji, inicjującego rozwiązywanie kolizji, zależy od czasu obiegu cyklu. Jeżeli obciążenie systemu jest względnie małe, to zbiór alokacji może być tożsamy ze zbiorem wszyst-

kich węzłów. Jeżeli obciążenie systemu jest względnie duże, to rozmiar zbioru alokacji może być ograniczony do jednego węzła; w tym przypadku system rozdziału przepływu jest równoważny systemowi zwielokrotniania z podziałem czasu TDM (ang. *Time Division Multiplexing*).

11.4.4. ALGORYTMY ROZDZIAŁU Z FUNKCJĄ SKRÓTU

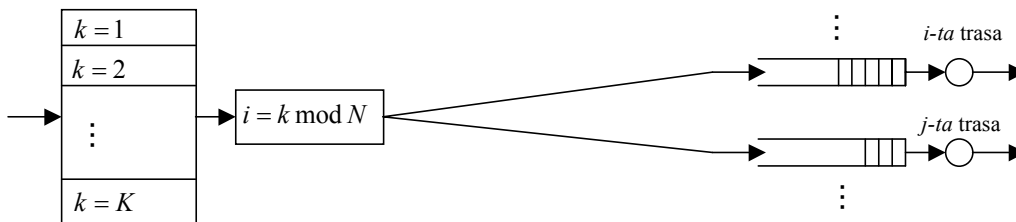
Algorytmy rozdziału strumieni jednostek danych przepływów, wykorzystujące funkcje skrótu HBTS (ang. *Hashing-based Traffic-splitting*), są łatwe w implementacji i możliwa jest ich sprzętowa realizacja.

W przypadku przepływu pakietów do obliczenia wartości funkcji skrótu mogą być stosowane wartości pól nagłówka pakietu. Wartość funkcji skrótu dla pakietów należących do tego samego przepływu jest taka sama. Ta właściwość funkcji skrótu może być wykorzystana m.in. do odtwarzania kolejności.

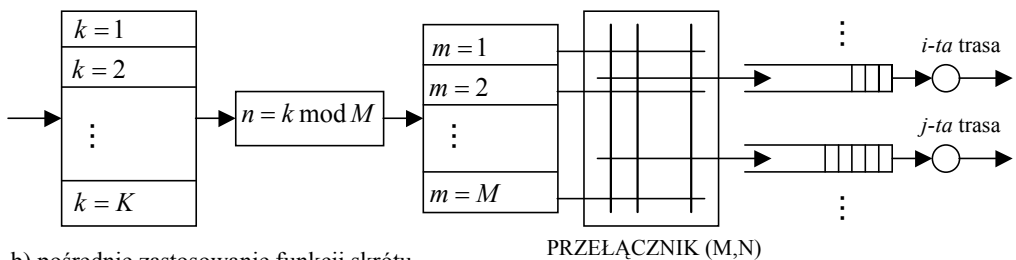
Algorytmy rozdziału przepływu klasy ruchu, wykorzystujące funkcję skrótu, dzielone są na dwie grupy (rysunek 11.6):

➤ bezpośrednie (ang. *direct hashing*); wartość funkcji skrótu, obliczana dla wybranych pól nagłówka, jest numerem trasy wyjściowej z węzła lub jest stosowana do wyznaczenia numeru trasy wyjściowej z węzła,

➤ tablicowe (pośrednie) (ang. *table-based hashing*); wartość funkcji skrótu jest stosowana do podziału strumienia wejściowego na rozłączne części, którym następnie przyporządkowywane są różne trasy wyjściowe z węzła.



a) bezpośrednie zastosowanie funkcji skrótu



b) pośrednie zastosowanie funkcji skrótu

Rysunek 11.6. Bezpośrednie i pośrednie zastosowanie funkcji skrótu w zadaniu rozdziału przepływu

11.4.4.1. PODEJŚCIE BEZPOŚREDNIE

W najprostszym algorytmie rozdziału przepływu na podstawie zawartości pola adresu wyjściowego pakietu, w którym stosowane jest podejście bezpośrednie (rysunek 11.6a), numer trasy wyjściowej jest wynikiem działania:

$$i = k \times \text{mod } N,$$

gdzie:

i – numer trasy wyjściowej z węzła ($1 \leq i \leq N$),

N – liczba wszystkich tras wyjściowych z węzła,

k – wartość pola adresowego ($1 \leq k \leq K$),

K – liczba wszystkich różnych wartości pól adresowych i jednocześnie największa wartość pola adresowego.

Innym prostym sposobem wyznaczania numeru trasy wyjściowej jest wykorzystanie do tego celu sumy kontrolnej, tzn.:

$$i = \{CRCL[k]\} \times \text{mod } N,$$

gdzie L jest długością sumy kontrolnej, np. $L = 16$ dla $CRC16$.

Stosowanie sumy kontrolnej, mimo większej złożoności obliczeniowej w porównaniu z działaniem bezpośrednim na adresie, jest uzasadnione właściwościami sumy kontrolnej; w szczególności równomiernością rozkładu wartości sumy kontrolnej dla różnych zawartości pola adresu.

Wadą podejścia bezpośredniego jest to, że rozdział strumienia wejściowego nie zależy od charakterystyk tras wyjściowych. Rozdział jest proporcjonalny do liczby różnych zawartości pól adresowych nagłówka z dokładnością do zdolności rozdzielczej funkcji skrótu.

11.4.4.2. PODEJŚCIE TABLICOWE

Podejście pośrednie z wykorzystaniem tablicy wartości funkcji skrótu pozwala na eliminację podstawowej wady podejścia bezpośredniego. Wyznaczanie numeru trasy wyjściowej jest dwuetapowe:

➤ w pierwszym kroku wszystkie jednostki danych przepływu, o różnych wartościach pola adresowego k ($1 \leq k \leq K$), są klasyfikowane do jednej z M ($M < K$) różnych grup jednostek danych,

➤ w drugim kroku jednostki danych z poszczególnych M grup są przydzielane do jednej z N ($N < M$) różnych tras wyjściowych węzła.

Odpowiedni dobór liczby grup jednostek danych M oraz metod klasyfikacji decyduje o elastyczności rozdziału przepływu na grupy. Różne metody odwzorowania numeru grupy w numery tras wyjściowych pozwalają na dostosowanie ilości ruchu kierowanego do obsługi przez różne trasy wyjściowe do zasobów dostępnych na tych trasach, tzn. na efektywne wykorzystanie zasobów tras.

11.5. TECHNIKA KOMUTACJI ETYKIET

Oryginalne i początkowe zadanie stosowania techniki wieloprotokołowej komutacji (przełączania) etykiet (etykietowej) MPLS (ang. *Multiprotocol Label Switching*) to uproszczenie mechanizmów przekazywania pakietów w węzłach sieci, a tym samym poprawa efektywności wykorzystania zasobów i przyspieszenie realizacji przekazywania (komutacji) pakietów w routerach. Przydatność tej techniki, także w zadaniach inżynierii ruchu, wynika z możliwości jej stosowania w zadaniach równoważenia obciążeń w sieciach, tzn. do wymuszania takich tras dla ruchu w sieci, które maksymalizują jakość obsługi ruchu oraz stopień wykorzystania zasobów sieci. Stopień wykorzystania zasobów jest możliwy m.in. przez zrównoleglenie obsługi elementów tego samego przepływu.

Koncepcja komutacji etykiet jest złożeniem dwóch zasadniczo różnych koncepcji obsługi ruchu w sieci, tzn. koncepcji datagramu i łącza wirtualnego. Podstawowa idea komutacji etykiet znajduje zastosowanie w różnych technologiach, technikach i rozwiązaniach sieciowych (np. FR, ATM, IP).

Zastosowanie komutacji etykiet w sieciach IP pozwala na unifikację kierowania ruchem i ułatwia integrację z różnymi rozwiązaniami warstw niższych (np. ATM). Celem stosowania komutacji etykiet w sieciach IP nie jest zastąpienie komutacji pakietów IP, ale uzupełnienie zakresu usług sieciowych oferowanych przez sieci IP o usługi dotyczące inżynierii ruchu, gwarancji jakości usług sieci itp. W szczególności istotne są następujące korzyści wynikające z zastosowania koncepcji komutacji etykiet w sieciach IP:

- uproszczenie procesu przekazywania pakietów, tzn. procesu klasyfikacji pakietu i wyznaczenia trasy za cenę ustanowienia trasy komutacji etykiet (ang. *label switched path*) przez protokół sygnalizacyjny (ang. *signalling protocol*) lub protokół dystrybucji etykiet (ang. *label distribution protocol*), poprzedzającego transfer pakietów w sieci,

- możliwość obsługi różnych protokołów przez tę samą trasę w sieci po oddzieleniu procesu przekazywania pakietów (ang. *forwarding*) od procesu wyznaczenia trasy pakietu (ang. *routing*),

- zwiększenie granulacji przepływów dzięki możliwości stosowania różnych schematów klasyfikacji pakietów.

11.5.1. WYZNACZANIE TRAS I PRZEŁĄCZANIE

Zadania wyznaczenia tras dla jednostek danych (ang. *routing*) i przełączania jednostek danych (ang. *switching*) są odzwierciedleniem dwóch koncepcji realizacji sieci pakietowych, tzn. sieci datagramowych i sieci połączeń wirtualnych.

Niezależnie od przyjętego rozwiązania proces przekazywania pakietów w węzle sieci z komutacją pakietów wiąże się z realizacją dwóch grup funkcji: sterowania przekazywaniem pakietów i przekazywaniem pakietów. Realizacja wymienionych grup

funkcji na trasie w sieci oznacza współlistnienie trasy danych (ang. *data path*) i trasy sterowania (ang. *control path*).

Trasa danych spełnia funkcje wykonawcze w zadaniu przekazywania pakietów. Przetwarzanie każdego nowo napływającego pakietu do węzła (routera) sieci polega na skorzystaniu z zawartości nagłówka pakietu oraz zawartości tablicy przekazywania (ang. *forwarding table*) do znalezienia adresu kolejnego węzła na trasie.

Trasa sterowania spełnia funkcje organizacyjne wobec trasy danych. Trasa sterowania jest wykorzystywana do wymiany informacji pomiędzy węzłami sieci, które są niezbędne do wyznaczania i aktualizacji zawartości tablicy przekazywania.

Konstrukcja tablicy przekazywania jednostek danych oraz jej wykorzystanie do podejmowania decyzji są dwiema oddzielnymi logicznie operacjami, tzn. funkcje tras danych i sterowania są rozłączne.

Różne realizacje rozwiązań węzłów sieci charakteryzują różne sposoby wykonywania operacji na trasie danych i trasie sterowania, liczba powtórzeń omawianych operacji na trasie oraz różny stopień separacji tych dwóch grup funkcji.

Zasadnicza różnica pomiędzy urządzeniami realizującymi funkcje wyznaczania tras i przełączania polega na różnej funkcjonalności trasy danych i trasy sterowania:

- w przypadku wyznaczania tras dla pakietów zadania analizy zawartości nagłówka pakietu i jego dopasowania do tablicy tras są realizowane na trasie danych, zadaniem tras sterowania jest natomiast aktualizacja zawartości tablic tras,

- w przypadku przełączania pakietów zadania analizy zawartości nagłówka pakietu są realizowane na trasie sterowania, w której generowana jest wartość indeksu wykorzystywana na trasie danych do przekazywania pakietów, zadania trasy danych są natomiast ograniczone do przekazywania i przełączania pakietów.

Źródłem efektywności przełączania jest to, że operacja analizy nagłówka pakietu i generacji wartości indeksu jest wykonywana tylko na brzegu sieci, tzn. nie jest powtarzana w każdym z węzłów trasy. Indeksowanie pakietów na brzegu sieci upraszcza proces przekazywania pakietów na trasie danych, eliminuje bowiem wykonywanie złożonych czasowo i pamięciowo zadań klasyfikacji w każdym z węzłów sieci. Uproszczenie procesu przekazywania pakietów na trasie danych odbywa się za cenę rozbudowy funkcjonalności trasy sterowania, której zadaniem jest ustanowienie tras indeksów w sieci, tzn. tras identyfikowanych wartościami indeksów i tablicami odwzorowań indeksów w węzłach sieci.

Do realizacji tras sterowania mogą być stosowane różne protokoły o cechach protokołów sygnalizacyjnych.

Technikę MPLS można interpretować jako zaawansowaną formę przekazywania pakietów, która zastępuje tradycyjne algorytmy przekazywania pakietów, wykorzystujące różne sposoby dopasowywania względnie długich adresów ujęć, bardziej efektywnymi algorytmami odwzorowywania wartości etykiet. W przypadku sieci IP, implementacja MPLS pozwala uprościć realizację pewnych funkcji i realizować funkcje trudno dostępne lub niedostępne w klasycznej sieci IP (tabela 11.1).

Tabela 11.1. Podstawowe różnice pomiędzy routingiem IP i komutacją etykiet MPLS

Funkcje	Routing	Komutacja etykiet
Analiza zawartości nagłówka pakietu	realizowana w każdym węźle trasy	realizowana tylko raz, tzn. na brzegi sieci, gdzie przydzielana jest wartość etykiety
Wspieranie różnych trybów transmisji	zorientowane na obsługę transmisji w trybie jeden-do-jednego. Złożone algorytmy obsługi transmisji w trybie jeden-do-wielu	niezależność algorytmów od trybu transmisji
Podstawa decyzji o wyborze tras w sieci	adresy; w większości przypadków na podstawie adresu ujścia	wartości różnych parametrów, w tym m.in. wartości parametrów opisujących jakość usług sieci

11.5.2. ARCHITEKTURA MPLS

Istotą MPLS (ang. *Multiprotocol Label Switching*) jest generowanie względnie krótkich etykiet o ustalonych długościach, które są reprezentantami nagłówków pakietów (ogólnie jednostek danych). Etykieta można interpretować jako stenogram nagłówka. Etykiety są wykorzystywane w procesach przekazywania pakietów. Realizacja MPLS oznacza, że pakiety są „enkapsulowane” w etykietach przez pierwsze z urządzeń MPLS, jakie pakiet napotyka na drodze ze źródła do sieci, tzn. na brzegu sieci. Urządzenie MPLS analizuje zawartość nagłówka pakietu i wybiera odpowiednią etykietę. Zaletą MPLS jest m.in. to, że analiza nagłówka pakietu nie musi być ograniczona do zawartości pola adresu ujściowego pakietu, tzn. do wyboru etykiety mogą być wykorzystane dane zawarte nie tylko w nagłówku pakietu. Do podejmowania decyzji o przekazywaniu pakietu we wszystkich kolejnych węzłach sieci wykorzystywana jest etykieta MPLS, a nie nagłówek pakietu. Jeżeli pakiet z etykietą MPLS jest wyprowadzany z sieci, to odpowiednie urządzenie, zlokalizowane na brzegu sieci, usuwa etykietę.

W terminologii MPLS każdy z węzłów i routerów przetwarzających pakiety jest nazywany routerem LSR (ang. *Label Switched Router*). Router LSR wykorzystuje etykietę, zawartą w nagłówku pakietu, jako indeks do ustalenia następnego węzła i nowej etykiety. Pakiet jest wysyłany do kolejnego węzła po tym, jak dotychczasowa wartość etykiety zostanie wymieniona na wartość przyznaną na czas przekazania pakietu do kolejnego węzła. Wykorzystywana do przekazywania pakietu trasa komutacji etykiet LSP (ang. *Label Switched Path*) jest definiowana przez odwzorowania wartości etykiet w węzłach sieci.

Architekturę MPLS charakteryzują: sposoby odwzorowywania etykiet oraz mechanizmy dystrybucji etykiet w sieci.

11.5.2.1. ETYKIETA

Etykieta (ang. *label*) jest względnie krótkim identyfikatorem o ustalonej długości. Pakiet jest etykietowany, gdy etykieta jest zakodowana w pakiecie. Sposoby kodowa-

nia etykiety w pakiecie mogą być różne – etykieta może być albo umieszczana w jednym z pól istniejącego protokołu warstwy łącza danych lub sieci, albo w polu protokołu dodatkowej warstwy.

W sieci MPLS etykieta (20 bitów) jest polem nagłówka (32 bity) dodawanego do pakietu. W skład nagłówka MPLS wchodzi, oprócz etykiety, m.in. pole Exp (3 bity), które może być stosowane do wyróżniania klas ruchu i tym samym różnicowania jakości usług.

Ze względu na to, że etykieta jest identyfikatorem stosowanym do przekazywania pakietów, zadaniem dowolnego routera LSR jest powiązanie etykiety z trasą LSP. Różne wymagania dotyczące jednoznaczności wartości etykiet zależą od trybu transmisji realizowanego w sieci. W przypadku trybu transmisji jeden-do-jednego jednoznaczność etykiety jest wymagana w tylko w połączeniach pomiędzy routerami, w trybie jeden-do-wielu jednoznaczność etykiety wymagana jest natomiast w całej sieci lub jej wyróżnionej części.

Każda wartość etykiety wiąże się z klasą równoważności przekazywania FEC (ang. *Forwarding Equivalence Class*).

W technice MPLS dopuszczalne jest etykietowanie pakietu wieloma etykietami według schematu, że ostatnio przypisana etykieta jest etykietą o najwyższym priorytecie LIFOS (ang. *Last-In First-Out Stack*). Hierarchia etykiet może być wykorzystana do organizacji zagnieżdżanych usług, takich jak np. agregacja przepływów, tunelowanie przepływów. Dopuszczalność znakowania przepływów wieloma etykietami, tożsama z możliwością dodatkowej granulacji przepływów, oznacza, że ta sama jednostka danych może należeć do różnych klas ruchu w czasie jej przekazywania w sieci.

Działaniem odwrotnym w skutkach od działania hierarchii wartości etykiet jest łączenie wartości etykiet. Skutkiem łączenia wartości etykiet jest zmniejszanie granulacji przepływów w wyróżnionych fragmentach sieci.

11.5.2.2. KLASY RÓWNOWAŻNOŚCI PRZEKAZYWANIA RUCHU FEC

Ze względu na organizację przepływów w sieciach liczba klas ruchu wyróżnianych z uwagi na wartości wszystkich lub wybranych parametrów jest zawsze mniejsza od liczby wszystkich przepływów elementarnych. W klasycznych sieciach IP liczba klas ruchu może być równa np. liczbie wszystkich ujść w sieci, co wynika z tego, że wszystkie przepływy elementarne o tym samym adresie ujścia są traktowane jednakowo.

Celem wyróżniania klas ruchu w sieci jest uproszczenie procesu przekazywania jednostek danych poszczególnych przepływów elementarnych przez zmniejszenie liczby różnych możliwych w sieci sposobów obsługi. Zmniejszenie liczby klas ruchu oznacza zmniejszenie liczby różnych wymaganych sposobów obsługi ruchu w sieci i tym samym zmniejszenie złożoności obliczeniowej i pamięciowej zadań realizowanych w węzłach sieci. Liczba klas ruchu w sieci zależy od liczby kryteriów stosowanych do podziału ruchu. Zwiększenie liczby kryteriów pozwala na wzrost poziomu granulacji (uszczegółowienia) ruchu w sieci.

W technice MPLS obsługa ruchu w sieci polega na obsłudze klas ruchu, gdzie klasa ruchu jest utożsamiana z unikatowym sposobem obsługi. Klasa ruchu FEC jest tożsama ze zbiorem reguł klasyfikacji jednostek danych, których spełnienie oznacza przynależność jednostki danych do danej klasy ruchu.

11.5.2.3. TABLICE PRZEKAZYWANIA RUCHU

Realizacja przełączania w routerach LSR odbywa się za pomocą tablic przekazywania etykiet (ang. *label-switching table*), których organizacja jest zbliżona do tablic routingu. Tablica przekazywania pozwala na realizację odwzorowania wartości etykiety jednostki danych na wejściu routera LSR w wartość etykiety przypisywanej tej jednostce danych na wyjściu routera LSR. Jeśli router realizuje zadania równoważenia obciążenia, jednemu wejściu tablicy przekazywania ruchu odpowiada kilka wyjść tej tablicy (ang. *next-hop label-forwarding entry*).

11.5.2.4. PRZYDZIELANIE ETYKIET

Korzystanie z trasy LSP wymaga wcześniejszej aktualizacji tablic routingu w routerach LSR dla każdej z możliwych wartości etykiet. Proces aktualizacji tablic przekazywania w routerach LSR polega na dystrybucji (rozsyłaniu) wartości etykiet (ang. *label distribution*) i jest tożsamy z ustanawianiem tras (ang. *path setup*). Sposoby przydzielania i rozsyłania etykiet definiuje protokół LDP (ang. *Label Distribution Protocol*).

Cechą charakterystyczną techniki MPLS jest to, że przydzielanie etykiet różnym klasom ruchu odbywa się w kierunku przeciwnym do kierunku transferu jednostek danych. Oznacza to, że kierunek transferu informacji na trasach sterowania (ang. *downstream*) jest przeciwny do kierunku wymiany jednostek danych na trasach danych (ang. *upstream*). Przydzielanie etykiety dla danej klasy FEC jest dokonywane zawsze przez router następny, a informacja o tym powiązaniu jest przesyłana do routera poprzedniego. Otrzymanie informacji o powiązaniu oznacza dla danego routera znajomość następnego routera trasy LSP. Przyjęcie przeciwnych kierunków transferu na trasach sterowania i danych prowadzi do uproszczenia reguł odwzorowywania wartości etykiet w węzłach sieci.

Przydzielanie etykiet może być realizowane w dwóch trybach:

- na żądanie (ang. *downstream on demand*); poprzedzający router LSR, inicjujący przydzielanie etykiet, żąda od sąsiedniego (następnego w kierunku przesyłania danych) routera przydzielenia etykiety; przydzielanie etykiety przez router następny jest inicjowane przez router poprzedni,

- „niezamówionym” (ang. *unsolicited downstream*); następny router LSR w kierunku przesyłania danych samodzielnie podejmuje decyzję o przydzieleniu etykiety i informuje o tym router poprzedni.

W technice MPLS definiowane są dwa style rozsyłania etykiet: niezależny (ang. *independent*) i uporządkowany (ang. *ordered*). W przypadku niezależnego rozsyłania

etykiet router może rozsyłać informację o powiązaniach do routerów poprzedzających, bez znajomości routera następnego, tzn. bez znajomości etykiety wychodzącej. W przypadku uporządkowanego rozsyłania etykiet przydzielanie ich jest możliwe tylko wtedy, gdy router ma informację o przydzielonej etykietce z węzła następnego na trasie lub jest routerem brzegowym trasy LSP.

11.5.2.5. WYZNACZANIE TRAS W SIECI

Proces dystrybucji wartości etykiet przez routery LSR jest równoważny ustanawianiu trasy LSP. W zależności od tego, w jaką informację o sieci i w jakie algorytmy przetwarzania tej informacji są wyposażone routery LSR, możliwe są dwa tryby wyznaczania tras LSP:

- krok po kroku (ang. *hop-by-hop*); w każdym kolejnym węźle ustanawianej trasy LSP wywoływany jest moduł wyznaczania trasy, którego zadaniem jest niezależne wyznaczenie kolejnego węzła na trasie,
- bezpośrednio przez wybrane węzły sieci (ang. *explicit routing*); moduł wyznaczania trasy jest wywoływany w węźle brzegowym sieci, a jego zadaniem jest jednorazowy wybór całej trasy.

Jakość ustanowionej trasy zależy od algorytmów wyznaczania tras implementowanych w poszczególnych węzłach sieci, w tym na brzegu sieci. Pełne wykorzystanie właściwości techniki MPLS jest możliwe tylko wtedy, gdy implementowane algorytmy wyznaczania tras uwzględniają zarówno wymagania ruchu dotyczące jakości usług, jak i aktualny stan sieci.

11.5.3. MODELE ZAPEWNIANIA JAKOŚCI USŁUG SIECI I KOMUTACJA ETYKIET

Ze względu na cechy techniki wieloprotokołowej komutacji etykiet MPLS, która może być stosowana m.in. do sterowania przepływami i równoważenia obciążenia w sieci, technika ta spełnia wymagania stawiane metodom inżynierii ruchu.

W technice MPLS brak jest mechanizmów zapewniania jakości usług w sieciach, ale można ją stosować do podniesienia efektywności i rozszerzenia możliwości różnych koncepcji (modeli) zapewniania jakości usług. Podniesienie efektywności modeli różnicowania jakości usług jest konsekwencją zdolności równoważenia obciążeń przez technikę komutacji etykiet, a rozszerzenie możliwości różnych koncepcji zapewniania jakości usług wynika z możliwości współpracy techniki MPLS z różnymi protokołami sieciowymi.

11.5.3.1. MODEL USŁUG ZINTEGROWANYCH I KOMUTACJA ETYKIET

Integracja modelu usług zintegrowanych i techniki komutacji etykiet pozwala na rozwiązanie niektórych ograniczeń modelu usług zintegrowanych – efektywności

wykorzystania zasobów sieci, skalowalności modelu i ograniczeń zakresu stosowania modelu do sieci z protokołem IP.

Najistotniejszą korzyścią łączenia modelu usług zintegrowanych i techniki komutacji etykiet jest możliwość agregacji przepływów. Realizacja omawianej integracji wymaga rozszerzenia protokołu RSVP o mechanizmy zestawiania tras komutowanych etykietowo. Takie rozszerzenie protokołu RSVP umożliwi klasyfikację pakietów nie tylko na podstawie adresu i numeru portu źródła, identyfikatora protokołu warstwy transportowej oraz adresu i numeru portu ujścia. Klasyfikowanie pakietów z wykorzystaniem innych, szerszych kryteriów pozwala na agregowanie wielu pojedynczych przepływów w jednej trasie (tunelu) LSP. Agregacja pojedynczych przepływów rozwiązuje problem skalowalności modelu usług zintegrowanych; zwiększenie liczby zarezerwowanych połączeń w sieci nie jest proporcjonalny do wzrostu liczby pojedynczych przepływów.

Rozszerzenie protokołu rezerwacji zasobów uniezależnia zestawianie tras od protokołów routingu. Możliwość zestawienia trasy zawierającej wybrane węzły sieci pozwala na wybór tras optymalnych z uwzględnieniem różnych kryteriów, w tym równoważenia obciążeń.

11.5.3.2. MODEL USŁUG ZRÓŻNICOWANYCH I KOMUTACJA ETYKIET

Warunkiem współpracy modelu usług zróżnicowanych i techniki komutacji etykiet jest określenie sposobu reprezentacji klas usług modelu usług zróżnicowanych w sieci komutacji etykiet MPLS. W modelu usług zróżnicowanych klasa usług jest definiowana odpowiednią wartością punktu kodowego DSCP, natomiast w sieci MPLS klasa usług jest określana zawartością etykiety lub innych pól nagłówka MPLS; warunkiem współpracy jest zdefiniowanie reguł odwzorowania wartości punktów kodowych w wartości pól nagłówków MPLS.

Źródłem korzyści z integracji modelu usług zróżnicowanych i techniki komutacji etykiet jest elastyczność techniki komutacji etykiet. Opracowane zasady współpracy modelu usług zróżnicowanych i techniki komutacji etykiet pozwalają na rozszerzenie funkcjonalności modelu usług zróżnicowanych o mechanizm integrowania przesyłania różnych klas ruchu w jednej trasie LSP oraz mechanizm równoważenia obciążeń, tzn. możliwość przesyłania pakietów należących do tej samej klasy usług więcej niż jedną trasą LSP.

11.5.4. PRZYKŁADOWA REALIZACJA USŁUGI PREMIUM SERVICE W ARCHITEKTURZE MPLS

Przykładem wykorzystania architektury MPLS do różnicowania jakości usług w sieci pakietowej może być realizacja usługi Premium w sieci integrującej model usług zróżnicowanych i technikę komutacji etykiet.

Do realizacji usługi Premium w architekturze MPLS mogą być wykorzystane wszystkie możliwe w sieci trasy komutacji etykiet LSP, których liczba jest równa $N(N - 1)/2$, gdzie N jest liczbą wszystkich routerów brzegowych sieci. Liczba wszystkich tras LSP może być zredukowana przez zastąpienie wszystkich tras, których węzłami początkowymi są wszystkie routery wejściowe sieci (ang. *ingress router*) i węzłem końcowym jest jeden router wyjściowy sieci (ang. *egress router*), drzewem ujścia (ang. *sink tree*). Jeżeli każdy router brzegowy sieci jest zarówno routerem wejściowym, jak i wyjściowym, to liczba drzew ujściowych tras LSP jest równa N .

Jeśli przyjmie się, że trasy komutacji etykiet LSP lub drzewa ujściowe tras LSP są utrzymywane dla każdej klasy ruchu, to całkowita liczba tras LSP jest równa $MN(N - 1)/2$, a liczba wszystkich drzew ujściowych tras LSP jest równa MN . W przypadku wykorzystywania tych samych tras LSP do transmisji różnych klas ruchu niezbędne jest zastosowanie znakowania klas ruchu.

Realizacja usługi Premium w architekturze MPLS różni się od realizacji tej usługi w modelu usług zróżnicowanych sposobem obsługi pakietów w routerach sieci ISP:

- w routerach wejściowych sieci ISP do pakietu, nadchodzącego z sieci, do której należy źródło ruchu, dołączana jest etykieta MPLS,
- w routerach szkieletowych sieci ISP do sterowania przepływem pakietów wykorzystywana jest zawartość pola CoS (ang. *Class of Service*) etykiety MPLS,
- w routerach wyjściowych sieci ISP etykieta MPLS jest usuwana.

Różnice w obsłudze ruchu są ograniczone do zasięgu sieci ISP o architekturze MPLS, gdzie konfigurowane i utrzymywane są trasy etykiet LSP.

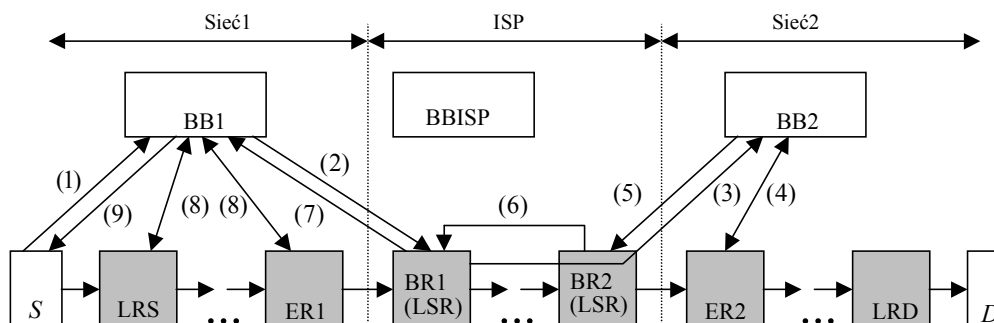
Pakiety wchodzące do sieci ISP o architekturze MPLS są klasyfikowane i przekazywane (routowane) w routerze wejściowym sieci pełniącym funkcje przełączania etykiet LSR (ang. *Label Switched Router*). Trasy LSP są używane jako tunele (ang. *tunnel*). Skonfigurowanie tras etykiet LSP w sieci ISP jest równoznaczne z tym, że trasa pakietu w sieci ISP jest determinowana etykietą dołączaną przez wejściowy router LSR sieci ISP.

Realizacja usługi Premium w sieci integrującej model usług zróżnicowanych i architekturę MPLS, podobnie jak w modelu usług zróżnicowanych, składa się z dwóch etapów: rezerwacji zasobów i transmisji pakietów. Specyfika architektury MPLS ma wpływ na sposób konfigurowania usługi Premium (etap rezerwacji zasobów), realizacja etapu drugiego (transmisja pakietów) natomiast przebiega w obydwu przypadkach tak samo.

Rezerwacja zasobów ma następujący przebieg (rysunek 11.8):

1. Źródło S , zlokalizowane w Sieci1, wysyła do lokalnego sterownika zasobów BB1 wiadomość RVSP PATH (wiadomość PATH protokołu RVSP) żądanie skonfigurowania usługi Premium.

W lokalnym sterowniku zasobów BB1 podejmowana jest decyzja o przyjęciu lub odrzuceniu żądania. Jeżeli żądanie jest odrzucane, to wysyłane jest powiadomienie do źródła S i procedura rezerwacji zostaje zakończona.



Rysunek 11.8. Przykładowa realizacja usługi Premium Service w sieci integrującej model usług zróżnicowanych i architekturę MPLS

2. Jeżeli żądanie skonfigurowania usługi Premium jest przyjęte przez sterownik zasobów BB1, to do wejściowego BR1 (LSR) sieci ISP wysyłana jest wiadomość RVSP PATH z żądaniem udostępnienia zasobów.

3. W routerze wejściowym BR1(LSR) sieci ISP, na podstawie analizy dostępnych zasobów sieci ISP, podejmowana jest decyzja o akceptacji żądania lub jej odrzuceniu:

- jeżeli żądanie jest odrzucane, to router BR1(LSR) wysyła powiadomienie do sterownika zasobów BB1, które z kolei powiadamia źródło *S* o odrzuceniu żądania skonfigurowania usługi,

- jeżeli żądanie jest akceptowane, to router BR1 (LSR) wysyła wiadomość RVSP PATH do routera wyjściowego BR2 (LSR) sieci ISP, który wiadomość przekazuje do sterownika zasobów BB2; wiadomości na trasie pomiędzy routerami sieci ISP są przekazywane trasą LSP.

4. W sterowniku BB2 podejmowana jest decyzja o możliwości realizacji żądanej usługi w Sieci2:

- jeżeli żądana usługa nie może być realizowana w Sieci 2, to sterownik BB2 generuje powiadomienia, przekazywane następnie do routera BR2 i dalej do źródła *S*,

- jeżeli żądanie jest akceptowane, to sterownik zasobów BB2 wysyła do routera ER2 wiadomość w celu konfiguracji jego zbioru reguł klasyfikacji i kontroli pakietów.

5. Skonfigurowanie routera ER2 powoduje wysłanie przez sterownik zasobów BB2 powiadomienia RVSP RVSP (wiadomość RVSP protokołu RVSP) do routera BR2(LSR).

6. Odebranie przez router BR2(LSR) powoduje konfigurację jego reguł kształtowania ruchu i wysłanie powiadomienia RVSP RVSP do routera BR1(LSR); do tego celu wykorzystywana jest odpowiednia trasa LSP.

7. W routerze BR1(LSR) konfigurowane są jego reguły klasyfikacji pakietów i kontroli (ang. *policing*) ruchu, a następnie wysyłane jest powiadomienie RVSP RVSP do sterownika zasobów BB1 w Sieci1.

8. Odebranie przez sterownik zasobów powiadomienia RVSP powoduje skonfigurowanie zbioru reguł klasyfikacji i kształtowania ruchu w routerze dostępowym LR1 oraz reguł kształtowania ruchu w routerze brzegowym ER1 w Sieci1.

9. Sterownik BBI powiadamia źródło *S* o zakończeniu konfiguracji usługi Premium, którego odebranie umożliwia rozpoczęcie obsługi ruchu.

11.6. PODSUMOWANIE

Większość klasycznych protokołów routingu, takich jak np. protokół IP, jest całkowicie przezroczysta (ang. *transparent*) dla żądań różnej jakości usług ze strony różnych przepływów w sieci. W konsekwencji decyzje dotyczące tras są podejmowane bez względu na dostępność zasobów i wymagania jakościowe użytkowników. Oznacza to, że możliwe są sytuacje, kiedy przepływy są kierowane na trasy niezdolne do ich obsługi na żądanym poziomie jakości usług, podczas gdy alternatywne trasy z żądanymi zasobami są dostępne. Nierównomierność obciążenia zasobów sieci może się objawiać znaczącym spadkiem ilości i jakości usług.

Wyznaczanie tras dostarczających żądanej jakości usług jest zadaniem algorytmów routingu, które mogą identyfikować trasy spełniające wymagania użytkowników sieci z zakresu jakości usług dla maksymalnej liczby przepływów. Podstawowe znaczenie mają takie rozwiązania, zgodnie z którym algorytmy routingu o podanych właściwościach są rozszerzeniem klasycznych protokołów routingu.

Działanie protokołów routingu, wspierających realizację różnych koncepcji udostępniania różnej jakości usług dla różnych klas ruchu w sieci, jest zbliżone do działania każdego innego protokołu routingu. Warunkiem działania takiego protokołu jest specyfikacja wymagań jakościowych, które w różny sposób mogą być udostępniane dla protokołu. Realizacja żądania dotyczącego jakości usług oznacza, że protokół routingu podaje specyfikację trasy najbardziej dopasowanej do żądań danego przepływu. W zależności od wymagań przepływów oraz procedur selekcji tras w algorytmie routingu, specyfikacja trasy może być różna: od wskazania kolejnego węzła na trasie do listy wszystkich węzłów pośrednich na trasie od źródła do ujścia. Wyznaczenie dopasowanej trasy pozwala na przypisanie przepływu do tej trasy; przepływ pozostaje przypisany trasie do chwili zwolnienia trasy lub do chwili, gdy trasa przestaje być dopasowana do przepływu.

Algorytmom routingu, pozwalającym wyznaczać trasy o żądanych właściwościach ilościowych i jakościowych, stawiane są różne warunki, wynikające m.in. z wymagań dotyczących prostoty implementacji, dokładności i złożoności obliczeniowej.

Realizacja celów inżynierii ruchu wymaga uzupełnienia algorytmów wyznaczania tras z ograniczeniami przez algorytmy równoważenia obciążeń. Znaczenie algorytmów równoważenia obciążeń jest tym większe, im mniejsza jest granulacja przepływu w sieci. Z tego m.in. wynika duże znaczenie algorytmów równoważenia obciążeń w modelu usług zróżnicowanych, gdzie liczba klas ruchu jest względnie mała.

Efektywnym sposobem realizacji podstawowego zadania inżynierii ruchu w sieciach, tzn. równoważenia obciążeń, jest m.in. stosowanie techniki komutacji etykiet. Implementacja tej techniki oznacza wprowadzenie do sieci nowego poziomu sterowania przepływem ruchu i uniezależnienie efektywności sterowania przepływami od protokołów warstw wyższych. Efektywność tej techniki jest widoczna szczególnie wtedy, gdy uzupełnia działanie protokołów o bardzo ograniczonych możliwościach różnicowania ruchu i jakości obsługi ruchu (np. IP).

Różne techniki inżynierii ruchu, takie jak wyznaczanie tras z ograniczeniami czy komutacja etykiet, nie pozwalają bezpośrednio sterować parametrami jakości usług. Techniki te dostarczają odpowiednie mechanizmy, które ułatwiają zapewnianie jakości usług, w szczególności w sieciach z implementacją modelu usług zróżnicowanych.

LITERATURA

- [1] ABOUL-MAGD O., JAMOSSI B., *QoS and service interworking using constraint-route Label Distribution Protocol (CR-LDP)*, IEEE Communications Magazine, May 2001, s. 134–139.
- [2] APOSTOLOPOULOS G., GUERIN R., KAMAT S., ORDA A., TRIPATHI S.K., *Interdomain QoS Routing in IP networks: a feasibility and cost/benefit analysis*, IEEE Network, September/October 1999, s. 43–54.
- [3] AUKIA P., KODIALAM M., KOPPOL P.V.N., LAKSHMAN T.V., SARIN H., SUTER B., *RATES: A server for MPLS traffic engineering*, IEEE Network, March/April 2000, s. 34–41.
- [4] BHARATH-KUMAR K., JAFFE J.M., *Routing to multiple destinations in computer networks*, IEEE Transactions on Communications, Vol. COM-31. No. 3, March 1983, s. 343–351.
- [5] BURAKOWSKI W., *Inżynieria ruchu w sieciach B-ISDN ATM*, Przegląd Telekomunikacyjny, nr 7, 1993.
- [6] BURAKOWSKI W., BĄK A., KOPERTOWSKI Z., WYKROTA A., *Mechanizmy QoS w sieci IP*, materiały konferencji „Internet – Wrocław 2000”, Wrocław 2000, s. 25–37.
- [7] CHAO H.J., GUO X., *Quality of Service control in High-Speed Networks*, Wiley, 2002.
- [8] DECASPER D., DITTIA Z., PARULKAR G., PLATTNER B., *Router pluggins: a software architecture for next generation routers*, IEEE/ACM Transactions on Networking, Vol. 8, No. 1, s. 2–16, February 2000.
- [9] FELDMANN A., REXFORD J., CACERES R., *Efficient policies for carrying web traffic over flow-switched networks*, IEEE/ACM Transactions on Networking, Vol. 6, No. 6, December 1998, s. 673–685.
- [10] FELDMANN A., GREENBERG A., LUND C., REINGOLD N., REXFORD J., *Netscape: traffic engineering for IP networks*, IEEE Network, March/April 2000, s. 11–19.
- [11] FENG Q.C., KANDLUR D.D., SAHA D., SHIN K.G., *Adaptive packet marking for maintaining end-to-end throughput in a Differentiated Services Internet*, IEEE/ACM Transactions on Networking, Vol. 7, No. 5, June 1993, s. 685–697.

- [12] FULP W.E., REEVES D.S., *Optimal provisioning and pricing of differentiated services using QoS class promotion*, N.C. State University (www.itpapers.com).
- [13] GEORGIDIS L., GUERIN R., PERIS V., SIVARAJAN K., *Efficient network QoS provisioning based on per node traffic shaping*, IEEE/ACM Transactions on Networking, Vol. 4, No. 4, August 1996.
- [14] GIACOMAZZI P., MUSUMECI L., *Transport of IP controlled-load service over ATM networks*, IEEE Network, January/February 1999, s. 36–47.
- [15] GRIFFIN T.G., SHEPHERD F.B., WILFONG G., *The stable path problem and interdomain routing*, IEEE Transactions on Networking, Vol. 10, No. 2, April 2002, s. 232–243.
- [16] GUERIN R.A., ORDA A., WILLIAMS D., *QoS routing mechanisms and OSPF extensions*, IEEE Global Internet Mini-Conference (Globecom '97), November 1997.
- [17] HEGERING H.G., ABECK S., *Integrated network and system management*, Addison-Wesley, Wokingham 1994.
- [18] JAJSZCZYK A., *Ewolucja sieci stacjonarnych*, Przegląd Telekomunikacyjny, rocznik LXXIV, nr 1, 2001, s. 16–20.
- [19] KACZMAREK S., NARLOCH M., *Analiza metod realizacji jakości usług w sieciach IP (IP QoS)*, materiały konferencji „Intenet – Wrocław 2000”, Wrocław 2000, s. 75–87.
- [20] KOMPELLA V.P., PASQUALE J.C., POLYZOS G.C., *Multicast routing for multimedia communication*, IEEE/ACM Transactions on Networking, Vol. 1, No. 3, June 1993, s. 286–292.
- [21] KUO G., *Multiprotocol Label Switching*, IEEE Communications, Vol. 37, No. 12, December 1999.
- [22] LORENZ D.H., ORDA A., *QoS routing in networks with uncertain parameters*, IEEE/ACM Transactions on Networking, Vol. 6, No. 6, December 1998, s. 768–778.
- [23] LEUE S., OECHSLIN P.A., *On parallelizing and optimising the implementation of communication protocols*, IEEE Transactions on Networking, Vol. 4, No. 1, February 1996, s. 55–70.
- [24] LUBACZ J., *Modele odniesienia w rozwoju infrastruktury informacyjnej*, Przegląd Telekomunikacyjny, rocznik LXXIV, nr 1, 2001, s. 48–52.
- [25] LUBACZ J., TOMASZEWSKI A., *Internet a sieci nowej generacji*, Przegląd Telekomunikacyjny, rocznik LXXIV, nr 5–6, 2001, s. 326–330.
- [26] MEER H., PULIAFITO A., RICHTER J.P., TOMARCHIO O., *QoS-adaptation by software agents in the presence of defective reservation mechanisms in the Internet*, University of Hamburg, 2000.
- [27] MISHRA P.P., SARAN H., *Capacity management and routing policies for Voice over IP traffic*, IEEE Network, March/April 2001, s. 20–27.
- [28] ORDA A., *Routing with end-to-end QoS guarantees in broadband networks*, IEEE/ACM Transactions on Networking, Vol. 7, No. 3, June 1999, s. 365–374.
- [29] OZAKI T., KIM J.B., SUDA T., *Bandwidth-efficient multicast routing for multihop, ad-hoc wireless networks*, California State University, 2001 (www.itpapers.com).
- [30] PAPIR Z., *Ruch telekomunikacyjny i przeciążenia sieci pakietowych*, WKŁ, Warszawa 2001.
- [31] RYAN J., *Multiprotocol Label Switching (MPLS)*, Ennovate Networks, 2001 (www.tecguide.com).

- [32] SAHINOGLU Z., TEKINAY S., *On multimedia networks: self-similar traffic and network performance*, IEEE Communications Magazine, January 1999, s. 48–56.
- [33] SALSANO S., VELTRI L., *QoS control by means of COPS (Common Open Policy Service) to support SIP-based applications*, IEEE Network, March/April 2002, s. 27–33.
- [34] SHAIKH A., REXFORD J., SHIN K.G., *Evaluating the impact of state link state on Quality-of-Service routing*, IEEE Transactions on Networking, Vol. 9, No. 2, April 2001, s. 162–176.
- [35] STANKIEWICZ R., JAJSZCZYK A., *Wieloprotokółowa komutacja etykietowa (MPLS) i jej rola w zapewnianiu jakości usług w sieciach IP*, Przegląd Telekomunikacyjny, rocznik LXXXV, nr 4, 2002.
- [36] STOICA I., ZHANG H., NG T.S.E., *A hierarchical fair service curve algorithm for link-sharing, real-time, and priority services*, IEEE/ACM Transactions on Networking, Vol. 8, No. 2, April 2000, s. 185–211.
- [37] TRIMINTZIOS P., ANDRIKOPOULOS G.P., FLEGKAS P., GRIFFIN D., GEORGAT-SOS P., GODERIS D., T'JONES Y., GEORGIADIS L., JACQUENET Ch., EGAN R., *A management and control architecture for providing IP Differentiated Services in MPLS-based networks*, IEEE Communications Magazine, May 2001, s. 82–88.
- [38] VISWANATHAN A., FELDMAN N., WANG Z., CALLON R., *Evolution of Multi-protocol Label Switching*, IEEE Communications, Vol. 36, No. 5, May 1998.
- [39] WAICHEE A.F., *Delay-optimal quorum consensus for distributed systems*, IEEE Transactions on Parallel and Distributed Systems, Vol. 8, No. 1, January 1997, s. 59–69.
- [40] WANG Z., CROWCROFT J., *Bandwidth-delay based routing algorithms*, Proceedings of the GLOBECOM '95 Conference, Vol. 3, November 1995, s. 2129–2133.
- [41] WANG Z., *Internet traffic engineering*, IEEE Networks, Vol. 14, No. 2, March/April 2000.
- [42] WANG Z., *Internet QoS: Architectures and mechanisms for Quality of Service*, Academic Press, London 2001.
- [43] XIAO X., NI L.M., *Internet QoS: A big picture*, IEEE Network, March/April 1999, s. 8–18.
- [44] XIAO X., HANNAN A., BAILEY B., NI L.M., *Traffic Engineering with MPLS in the Internet*, IEEE Network, March/April 2000, s. 28–33.
- [45] YUAN X., *Heuristic algorithms for multiconstrained Quality-of-Service routing*, IEEE Transactions on Networking, Vol. 10, No. 2, February 2002, s. 244–256.

12. PODSUMOWANIE

Rozwój sieci teleinformatycznych i usług dostarczanych przez te sieci, zwłaszcza związanych z ruchem multimedialnym i dostarczaniem usług na żądanie (ang. *on demand*), wymaga zapewnienia przez sieć jakości usług i ich różnicowania. Różne podejścia, proponowane i stosowane do rozwiązywania zadań gwarantowania i różnicowania jakości usług dostarczanych przez sieci, wykorzystują w różnym stopniu dwie podstawowe koncepcje zapewniania jakości usług:

- wydzielania w sieci pakietowej bez gwarancji jakości usług zasobów w ilościach pozwalających na organizowanie takich podsieci, w których limitowanie ilości ruchu przyjmowanego do obsługi i jego obsługa zgodnie z zasadą najwyższej staranności pozwalają na uzyskanie jednakowej jakości usług w wydzielanych podsieciach i różnicowanie jakości usług w różnych podsieciach,

- rezerwacji zasobów dla ruchu przyjmowanego do obsługi w sieci; rezerwacja zasobów pozwala na gwarancję jakości usług na poziomie determinowanym przez ilości rezerwowanych zasobów.

Efektywność każdej z koncepcji zależy od przyjętego poziomu granulacji ruchu oraz działania implementowanych w sieci mechanizmów zarządzania ruchem.

Zarządzanie ruchem w sieciach teleinformatycznych ma na celu osiągnięcie stanu, kiedy zasoby sieci są sterowalne i sterowane, a celem sterowania jest udostępnianie zasobów sieci oraz ich zabezpieczenie przed przeciążeniami i blokadami. Celem zarządzania ruchem jest przede wszystkim zabezpieczenie użytkowników przed degradacją jakości usług oferowanych przez sieć, której zasoby są rozdzielane pomiędzy wiele połączeń obsługujących jednocześnie wiele różnych strumieni zgłoszeń. Dlatego też niezbędne jest zarządzanie stopniem wykorzystania zasobów i sterowanie ruchem tak, aby sieć działała na określonym (założonym) poziomie jakości usług, nawet wtedy, gdy generowane przez użytkowników obciążenie sieci jest większe od jej potencjału.

W sieciach z komutacją łączy akceptacja żądania obsługi oznacza ustanowienie połączenia pomiędzy abonentami oraz przydział stałej ilości zasobów na czas trwania połączenia. Brak możliwości ustanowienia połączenia i przydziału żądanych zasobów sieci oznacza odrzucenie żądania. Ten prosty sposób zarządzania dostępem zgłoszeń do zasobów sieci z komutacją łączy w zupełności wystarcza do przeciwdziałania prze-

ciążeniom, ponieważ raz przydzielone zasoby są zawsze dostępne, a rywalizacja o przydzielone zasoby sieci jest wykluczona.

Sterowanie ruchem w sieciach pakietowych jest zadaniem znacznie bardziej złożonym ze względu na losowy charakter ruchu oraz – wpisana w ideę sieci z komutacją wiadomości – rywalizację użytkowników o zasoby sieciowe. Obsługa jednostek danych w trybie „zapamiętaj i prześlij dalej” wiąże się z organizacją kolejek w każdym węzle sieci. Jeżeli intensywność napływu jednostek danych przekracza możliwości obsługi w sieci, to długość kolejek rośnie. Przeciążenie elementów sieci, równoważne przeciążeniu zasobów sieci, oznacza zwykle odrzucanie napływających jednostek danych w węzłach ujściowych oraz konieczność ich retransmitowania przez węzły źródłowe, powodując m.in. wzrost opóźnień i wariancji opóźnień jednostek danych. Sytuacjom takim zapobiega się m.in. przez przyjęcie górnych ograniczeń stopnia wykorzystania zasobów sieci, co może ograniczać skalę zjawiska, ale nie zapobiega powstawaniu opisywanych sytuacji i nie rozwiązuje problemu przeciążenia sieci. Przyjęcie bardziej rygorystycznych ograniczeń stopnia wykorzystania zasobów sieci, dających możliwość stosowania względnie prostych mechanizmów sterowania ruchem w sieci, jest nie do przyjęcia dla operatora sieci, zainteresowanego ekstremalizacją stopnia wykorzystania zasobów. Dlatego też podstawowym problemem zarządzania i sterowania w sieciach z komutacją wiadomości jest jednoczesna maksymalizacja stopnia wykorzystania zasobów i minimalizacja prawdopodobieństwa powstawania przeciążeń prowadzących do pogorszenia jakości usług.

W sieciach z komutacją wiadomości stosowane są, wzajemnie się uzupełniające, dwa rodzaje sterowania ruchem: przeciwdziałanie przeciążeniom i sterowanie przepływem.

Celem mechanizmów przeciwdziałania przeciążeniom jest utrzymywanie całkowitej ilości ruchu obsługiwanego w sieci lub w jej fragmentach poniżej pewnego poziomu, którego zachowanie gwarantuje brak przeciążeń i tym samym jakość usług sieci.

Celem mechanizmów sterowania przepływem, adresowanych zasadniczo do pojedynczych par systemów końcowych, jest regulacja ilości ruchu obsługiwanego w sieci, co polega na dopasowaniu szybkości wprowadzania ruchu do sieci do szybkości jego wyprowadzania. Sterowanie przepływem pomiędzy systemami końcowymi, ze względu na jednoczesne współistnienie w sieci wielu takich przepływów, nie zapobiega powstawaniu przeciążeń. Stosowane techniki sterowania przepływem dzieli się na prewencyjne i reakcyjne; znaczenie tych ostatnich maleje wraz ze wzrostem szybkości sieci.

Poprawne, zgodne z oczekiwaniami użytkowników, działanie wielu aplikacji sieciowych wymaga określonych gwarancji różnej jakości usług dla różnych klas ruchu. Jakość żądanych usług sieci jest nadrzędnym kryterium podziału ruchu w sieci, wyróżniania różnych jego klas w sieci oraz wyróżniania różnych sposobów obsługi.

W klasycznych rozwiązaniach sieci teleinformatycznych z komutacją jednostek danych (wiadomości, bloków danych, segmentów, pakietów, ramek, komórek) nie ma

naturalnych mechanizmów wspierających różnicowanie klas ruchu i klas jakości usług. Wynika to z natury systemów sieciowych z komutacją jednostek danych, w których opóźnienia i straty wnoszone przez poszczególne elementy sieci nie są zdeterminowane i zależą od wielu czynników, w tym od aktualnego stanu obciążenia sieci. Z natury sieci z komutacją jednostek danych wynika także, charakterystyczna dla takich sieci, obsługa ruchu bez gwarancji jakości oraz jakościowy sposób opisu uzyskiwanej jakości usług.

Podstawowe czynniki decydujące o atrakcyjności sieci pakietowych (sieci Internet) dla użytkowników i operatorów to m.in. jej powszechność oraz względna łatwość realizacji usług i kierowania ruchem. Prostota elementów sieci i zarządzania ruchem wynika m.in. z tego, że sieć pakietowa jest zorientowana na funkcje transportowe, usługi natomiast są realizowane w dołączonych do sieciach niezależnych węzłach usługowych. Jednocześnie sieci z komutacją pakietów charakteryzuje względnie niska jakość transportu danych i niezawodność oraz ograniczone możliwości świadczenia zaawansowanych tzw. usług inteligentnych dostępnych np. w sieciach ISDN. Konsekwencją względnej prostoty sieci pakietowych jest brak gwarancji i różnicowania jakości usług. Jakość usług zależy od aktualnego obciążenia sieci i jest w sensie średnim taka sama dla wszystkich użytkowników sieci.

Uzyskanie różnicowania jakości usług w sieci bez gwarancji jakości usług wymaga uzupełnienia mechanizmów komutacji jednostek danych o rozwiązania (metody, procedury, mechanizmy) gwarantujące skuteczne dedykowanie zasobów, sterowanie dostępnymi zasobami, opóźnieniem, fluktuacją opóźnienia czy stratami jednostek danych dla wybranego prostego lub złożonego strumienia ruchu. Wdrożenie takich procedur oznacza zwykle wzrost liczby i złożoności mechanizmów sterowania ruchem implementowanych w sieci oraz zwiększenie ilości zasobów sieci kierowanych do obsługi ruchu w sieci.

Modele usług zintegrowanych i zróżnicowanych implementowane w sieciach pakietowych są sposobami obsługi ruchu teleinformatycznego, w których żadaną jakość usług osiąga się przez nałożenie na sieć bez gwarancji jakości usług dodatkowych ograniczeń dotyczących: warunków dopuszczania ruchu do obsługi, alokacji zasobów sieci, kształtowania ruchu, klasyfikacji przepływów, szeregowania przepływów itd. Nakładane ograniczenia mają różny charakter i różny zakres, w granicznym przypadku prowadzą do rozwiązań negujących ideę sieci z komutacją jednostek danych, tzn. ideę ograniczonego współzawodnictwa o zasoby sieci.

Implementacja modeli usług zintegrowanych i usług zróżnicowanych w sieci bez gwarancji jakości usług oznacza dekompozycję zasobów sieci pomiędzy trzy ogólne zbiory klas ruchu i tym samym klas usług. Podział ten jest równoważny utrzymywaniu trzech rozłącznych podsieci różniących się sposobami dostępu do zasobów i stopniem ich wykorzystania.

Realizacja modelu usług zintegrowanych wiąże się z rezerwacją zasobów w ilościach gwarantujących zapewnienie określonego poziomu przepływności, zachowanie ograniczenia opóźnienia wnoszonego przez podsieć oraz brak strat kolejkowania dla

poszczególnych przepływów elementarnych. Implementacja tego modelu polega na wydzieleniu i wymuszaniu dostępności do zasobów w ilościach wymaganych dla żądanej gwarancji jakości usług. Jakość usług dostępnych w modelu usług zintegrowanych jest okupiona praktyczną rezygnacją z podstawowych właściwości komutacji jednostek danych, tzn. współdzielenia zasobów i braku separacji przepływów w sieci.

Realizacja modelu usług zróżnicowanych jest rozszerzeniem modelu sieci bez gwarancji jakości usług, w takim znaczeniu, że różne bezwzględne lub względne priorytety różnych klas ruchu pozwalają na różnicowanie jakości obsługi. Różnice jakości usług są tym większe, im większe jest obciążenie podsieci, tzn. im większa jest różnica pomiędzy żądanymi i dostępnymi zasobami sieci. Implementacja modelu usług zróżnicowanych oznacza w praktyce utrzymywanie jednocześnie wielu podsieci bez gwarancji jakości usług z ograniczoną ilością ruchu w poszczególnych podsieciach.

Zasoby sieci kierowane do realizacji modelu bez gwarancji jakości są udostępniane dla ruchu na zasadzie współzawodnictwa; w konsekwencji jakość usług zależy od obciążenia podsieci i jest jednakowa w sensie średnim dla całkowitego ruchu w tej podsieci.

W modelach usług zintegrowanych i usług zróżnicowanych do sterowania ruchem stosuje się praktycznie te same metody, które stosowane są w sieciach bez gwarancji jakości usług. Istotna różnica pomiędzy modelem bez gwarancji jakości usług a wcześniej wymienionymi polega na tym, że implementacja modeli usług zintegrowanych i zróżnicowanych wymaga dodatkowo mechanizmów separujących wydzielone podsieci, dekomponujących zasoby sieci i zarządzające ilością ruchu obsługiwanego przez te podsieci. Wynikiem stosowania tych mechanizmów jest bezwzględna lub względna jakość usług dostarczanych użytkownikom sieci.

Analiza modeli usług zintegrowanych i usług zróżnicowanych oraz mechanizmów niezbędnych do ich realizacji pokazuje, że budowanie rozwiązań sieciowych gwarantujących jakość usług na bazie rozwiązania sieciowego, które z założenia takich gwarancji jakości usług nie zapewnia, jest zadaniem złożonym i o ograniczonych perspektywach. Koncepcje usług zintegrowanych i usług zróżnicowanych są koncepcjami zorientowanymi pakietowo, w tym znaczeniu, że są sposobami uzyskiwania jakości usług przez różnego rodzaju zawężenia lub rozszerzania koncepcji komutacji pakietów. Przyjęcie takiego punktu widzenia wyłącza z rozważań wiele różnych metod rozwiązywania problemów efektywności, jakości usług, niezawodności czy sterowania ruchem, w tym metod z powodzeniem stosowanych m.in. w sieciach telekomunikacyjnych.

SŁOWNIK

- absolute differentiated services* – bezwzględne różnicowanie jakości usług
- absolute performance measures* – bezwzględne miary jakości działania
- acknowledge* – powiadamianie
- acceptance region algorithm* – algorytm obszarów akceptacji
- acknowledge number* – numer potwierdzenia
- adaptive flow* – przepływ adaptacyjny
- adaptive pacing* – adaptacyjna metoda krokowa
- aggregate traffic* – ruch zagregowany
- aggregate traffic handling* – obsługa zorientowana na zagregowane strumienie ruchu
- aggressive flows* – przepływy zachłanne
- allocation interval* – przedział (czasu) alokacji
- application layer* – warstwa aplikacji
- assembly deadlock* – zakleszczenie odtwarzania (złożenia) jednostki danych
- assured service* – usługa zapewniona
- asynchronous transfer mode* – asynchroniczny tryb transferu
- authentication* – uwierzytelnianie
- automatic retransmission request* – procedura automatycznej retransmisji (korekcja przez retransmisje)
- backpressure effect* – efekt oddziaływania zwrotnego
- backpressure mechanism* – mechanizm oddziaływania zwrotnego
- backpressure vector* – wektor propagacji wstecznej
- backlogged flow* – przepływ z zaległościami
- bandwidth allocation* – przydział pojemności (pasma)
- behaviour aggregate* – agregaty strumieni ruchu
- best-effort* – obsługa ruchu bez gwarancji jakości
- bidirectional coding* – ramka interpolowana (ramka MPEG typu B)
- binary explicit feedback* – binarna bezpośrednia informacja zwrotna
- Bit Error Ratio (BER)* – elementarna stopa błędów
- boundary functions* – funkcje graniczne
- boundary node* – węzeł brzegowy
- bottleneck* – wąskie gardło
- broadcast mode* – tryb rozgłoszeniowy
- bucket depth* – pojemność (głębokość) bufora znaczników
- cache-based buffering* – buforowanie dynamiczne
- caching* – dynamiczne buforowanie
- call admission algorithm* – algorytm przyjmowania wywołań
- call admission control* – sterowanie przyjmowaniem wywołań
- call blocking probability* – prawdopodobieństwo odrzucania wywołań
- call control parameters* – miary jakości połączeń
- capacity differentiation* – różnicowanie pojemności
- carrier portion* – część sieciowa (zbiór funkcji trzech najniższych warstw modelu ISO/OSI)
- Cell Delay Variation (1-point CDV)* – jednopunktowa zmienność opóźnienia komórki
- Cell Delay Variation (1-point CDV)* – dwupunktowa zmienność opóźnienia komórki
- cell insertion rate* – współczynnik wtrącania komórek
- cell loss rate* – szybkość odrzucania (strat) komórek

- centralized flow control* – scentralizowane sterowanie przepływem
- choke packet* – pakiet dławiący
- choke packet scheme* – metoda z pakietem dławiącym (pakietów dławiących)
- circuit-switched network* – sieć z komutacją łączy
- class differentiation parameters* – parametry różnicowania klas ruchu
- classification* – klasyfikacja jednostek danych
- classification rule* – reguła klasyfikacji
- classifier* – klasyfikator
- coarse skew* – asymetria zgrubna
- coding delay* – opóźnienie kodowania
- collision coefficient* – współczynnik (prawdopodobieństwo) kolizji
- collision resolution period* – przedział (czasu) rozwiązywania kolizji
- communication controller* – sterownik komunikacyjny
- communication subnetwork (subnet)* – podsieć komunikacyjna
- communication subsystem* – podsystem komunikacyjny
- complex flow control* – całościowe sterowanie przepływem
- concatenation (blocking)* – łączenie (blokowanie) wielu jednostek danych
- concatenation probability* – prawdopodobieństwo rywalizacji
- conflict free filters* – filtry niesprzeczne
- conflict resolution filter* – filtr rozwiązujący konflikt adresów
- conforming connections* – połączenia nie naruszające warunków kontraktu ruchowego
- congestion* – przeciążenie
- congestion avoidance* – unikanie przeciążeń
- congestion collapse* – upadek sieci powodowany przeciążeniem
- congestion control* – przeciwdziałanie przeciążeniom
- congestion notification* – zgłoszenie przeciążenia
- congestion price* – cena przeciążenia
- congestion recovery* – regeneracja przeciążeń
- congestion signal* – sygnalizacja przeciążenia
- connection acceptance probability* – prawdopodobieństwo akceptacji połączenia
- connection admission control* – sterowanie przyjmowaniem wywołań
- connection blocking probability* – prawdopodobieństwo blokady połączenia
- connectionless protocol* – protokół bezpołączeniowy
- connection-oriented control* – sterowanie zorientowane na połączenia w sieci
- connection-oriented mode* – tryb połączeniowy
- connection-oriented protocol* – protokół połączeniowy
- connection release delay* – opóźnienie rozłączenia (znoszenia) połączenia
- constant link rate* – łącze danych o stałej szybkości
- constraint-based routing* – wyznaczanie tras z ograniczeniami
- continuous-go-back ARQ* – korekcja przez retransmisje grupowe
- contract violation probability* – prawdopodobieństwo naruszenia kontraktu ruchowego
- control function* – funkcja sterowania
- control path* – trasa sterowania
- control plane* – poziom sterowania
- controllability* – sterowalność
- controlled load service model* – model sterowanego obciążenia usługi
- core node* – węzeł szkieletowy
- core router* – router szkieletowy
- crash* – upadek systemu
- credit counter* – licznik kredytu
- credit method* – metoda kredytów
- crossbar* – krosownica
- customer depart rate* – średnia szybkość wypływu jednostek danych
- customer loss probability* – prawdopodobieństwo utraty jednostek danych
- customer portion* – część użytkownika (zbiór funkcji trzech najwyższych warstw modelu ISO/OSI)
- data flow control* – sterowanie przepływem danych
- data link control layer* – warstwa łącza danych
- data path* – trasa danych
- data plane* – poziom obsługi przepływów jednostek danych
- datagram network* – sieć z komutacją datagramów
- decision component* – moduł podejmowania decyzji
- deadline based algorithm* – algorytm szeregowania z gwarantowanym czasem obsługi

- deadlock* – zakleszczenie
- deadlock avoidance* – unikanie zakleszczeń
- decentralized flow control* – zdecentralizowane sterowanie przepływem
- decision function* – funkcja decyzyjna
- deficit counter* – licznik deficytu
- degree of anticipation* – współczynnik wyprzedzania
- degree of flow aggregation* – stopień agregacji przepływu
- delay offset* – przesunięcie (uskok) opóźnienia
- delay sensitive traffic* – ruch wrażliwy na opóźnienia
- delay variation (jitter)* – zmienność (fluktuacja) opóźnienia
- delay-systems* – systemy z opóźnieniami (w ogólnym przypadku systemy obsługi tolerujące opóźnienia)
- desired level of link sharing* – wartość oczekiwana poziomu współdzielenia pojemności łącza
- destination congestion* – przeciążenie ujścia
- destination-based routing* – routing zorientowany na ujścia przepływów
- deterministic jitter* – zmienność (fluktuacja) czasu transmisji jednostki danych
- deterministic multiplexing* – zwielokrotnianie deterministyczne
- direct hashing* – bezpośrednie zastosowanie funkcji skrótu
- direct store-and-forward deadlock* – zakleszczenie proste (bezpośrednie)
- dispatcher* – dyspozytor (muduł)
- distance vector routing* – wyznaczanie tras na podstawie wektora odległości
- distinct reservation* – rezerwacja indywidualna
- distributed algorithm* – algorytm rozproszony
- downstream on demand (label)* – rozdzielanie na żądanie (etykiet)
- drop level* – wartość progowa odrzucania (jednostek danych)
- drop precedences* – poziomy odrzucania jednostek danych
- drop probability* – prawdopodobieństwo odrzucania (jednostek danych)
- drop-from-front* – odrzucanie (pakietów) z czoła kolejki
- dropper module* – moduł odrzucania jednostek danych
- dynamic window sizing procedure* – procedura dynamicznego wymiarowania okna
- edge node* – węzeł brzegowy
- edge router* – router dostępowy (brzegowy)
- elastic traffic* – ruch elastyczny (wrażliwy na opóźnienia, fluktuację opóźnienia i straty)
- elimination of duplicates* – eliminacja duplikatów jednostek danych
- end-to-end control (source-sink control)* – jednokrokowe sterowanie przepływem
- end-to-end marking probability* – prawdopodobieństwo znakowania (pakietów) na trasie
- End-to-End QoS* – koncepcja szeregowej organizacji jakości usług
- enforcement component* – moduły wykonawcze
- engross router* – router wyjściowy sieci
- entry-to-exit level flow control* – sterowanie przepływem na poziomie trasy
- equivalent resource algorithm* – algorytm równoważnych zasobów
- equivalent capacity* – równoważna pojemność połączenia
- error control* – korekcja błędów
- excess demand probability* – prawdopodobieństwa nadmiaru żądań (popytu)
- expected capacity* – oczekiwana pojemność
- expected capacity profile* – oczekiwany profil pojemności
- explicit flow control* – bezpośrednie sterowanie przepływem
- explicit route* – trasa bezpośrednia
- explicit sender selection* – bezpośredni wybór źródła
- exponentially bounded burstiness process* – ograniczona wykładniczo wybuchowość źródła ruchu
- fair queuing* – sprawiedliwe kolejkowanie (szeregowanie)
- fair resource allocation* – sprawiedliwa alokacja współdzielonych zasobów
- fairness index* – indeks sprawiedliwości alokacji zasobów
- fairly-sharing* – sprawiedliwe współdzielenie (alokacja) zasobów
- fast path* – szybka ścieżka
- fast recovery procedure* – procedura szybkiego odtwarzania
- fast retransmission procedure* – procedura szybkiej retransmisji
- fastest path* – trasa najszybsza
- fat-dumb-pipe* – model (sieci) „grubej rury”

- feasible path* – trasa dopuszczalna
fidelity – wierność odtwarzania
filtration geometric scheme – geometryczny algorytm filtracji
filtering – filtrowanie (klasyfikacja) jednostek danych
fine skew – asymetria dokładna
flexibility factor – współczynnik elastyczności
flow – przepływ
flow control – sterowanie przepływem
flow descriptor – deskryptor przepływu
flow granulation – granulacja przepływu
flow identification – identyfikacja jednostek danych przepływu
flow potential – potencjał przepływu
flow shop – potokowy system obsługi
flow-based routing – routing zorientowany na przepływy
flow-oriented control – metody sterowania zorientowane na przepływy w sieci
flow scheduling – szeregowanie jednostek danych przepływu
flow specification – specyfikacja przepływu
flows aggregation – agregowanie przepływów
flows grouping – grupowanie przepływów
flows multiplexing – zwielokrotnianie przepływów
flows splitting – rozdział przepływów
fluid model – model przepływowy ruchu
forwarding classes – klasy spedycji ruchu
forwarding engine – procesor przekazywania jednostek danych
forwarding table – tablica przekazywania
forwarding treatment – sposoby spedycji klas ruchu
frame – ramka
free-for-all sharing – współzawodnictwo (o zasoby) bez ograniczeń
frequency of control – częstotliwość sterowania
- global congestion* – przeciążenie globalne
gradual restriction – metoda stopniowych ograniczeń
green (red) tokens – kolorowane znaczniki
guaranteed rate clock – gwarantowana szybkość obsługi jednostek danych
guaranteed service model – model gwarantowanej (jakości) usługi
- hash table* – tablica skrótów
heterogeneous network – sieć heterogeniczna
- homogeneous network* – sieć homogeniczna
host computer – komputer główny
host process – proces użytkownika
hop level flow control – sterowanie przepływem na poziomie łącza danych
hash collision – kolizja funkcji mieszającej
hash-coding – kodowanie mieszające
hashing-based scheme – technika funkcji skrótu (funkcji mieszającej)
- implicit flow control* – pośrednie sterowanie przepływem
in profile – ruch spełniający warunki profilu
indirect store-and-forward deadlock – zakleszczenie pośrednie
individual flow control – indywidualne sterowanie przepływem
inelastic traffic – ruch strumieniowy (wrażliwy na straty opóźnienia, fluktuację opóźnienia i przepustowość)
information transfer parameters – parametry miar jakości transferu
ingress router – router wejściowy sieci
input buffer limit scheme – metoda ograniczonej pojemności wejściowej pamięci buforowej
input packets – pakiety wejściowe
insertion ratio – współczynnik wtrącania jednostek danych
interarrival time – czas międzyzgłoszeniowy
interframe coding – ramka interkodowana (ramka MPEG typu I)
integrated policy services environment – zintegrowane środowisko zarządzania usługami
interior node – węzeł wewnętrzny (szkieletowy)
intermittent synchronous transfer mode – przerywany synchroniczny tryb transferu
internetworking utopia – koncepcja bezproblemowej integracji sieci
intrusion detection – detekcja naruszeń
isarithmic scheme – metoda izarytmiczna (przepustek) sterowania przepływem
isochronous transfer mode – izochroniczny tryb transferu
- jitter* – zmienność (fluktuacja) opóźnienia jednostki danych
jitter removal buffer – wyrównawcza pamięć buforowa
label – etykieta

- label distribution protocol* – protokół dystrybucji etykiet
- label switched path* – trasa komutacji etykiet
- label switching table* – tablica przekazywania etykiet
- label switching* – komutacja etykiet
- latency* – czas oczekiwania na odtwarzanie
- leading edge* – strzałka prowadząca (metoda okna)
- leaky bucket algorithm* – algorytm ciekącego wiadra
- level of statistical multiplexing* – poziom zwielokrotniania statystycznego
- link state routing* – wyznaczanie tras ze względu na stan sieci
- link-by-link control (stepwise control hop control)* – krokowe sterowanie przepływem
- link-loading factor* – współczynnik obciążenia łącza
- load balancing* – równoważenie obciążeń
- local congestion* – przeciążenie lokalne
- local marking probability* – lokalne prawdopodobieństwo znakowania (pakietów)
- long-lasting call* – wywołania o długim czasie trwania
- longer-lived congestion* – trwałe przeciążenie
- loss-systems* – systemy ze stratami (w ogólnym przypadku systemy obsługi tolerujące straty)
- loss model* – model obsługi ze stratami jednostek danych
- loss ratio* – współczynnik strat
- loss sensitive traffic* – ruch wrażliwy na straty
- loss tolerant traffic* – ruch niewrażliwy na straty
- marking algorithm* – algorytm znakowania
- maximum delay* – maksymalne opóźnienie
- max-min flow* – przepływ maksymalno-minimalny
- max-min flow control* – maksymalno-minimalne sterowanie przepływem
- mean interdeparture time* – średni czas między wychodzącymi jednostkami danych
- mean number of waiting customers* – średnia liczba oczekujących jednostek danych
- mesh virtual network* – pełna sieć wirtualna
- measured sum algorithm* – algorytm sumy mierzonej
- measurement-based admission control* – sterowanie dostępem na podstawie pomiarów
- minimum path latency* – suma stałych opóźnień na trasie
- misbehaving traffic* – ruch niezgodny z kontraktem ruchowym
- mission critical application* – aplikacje krytyczne
- multiaccess* – wielodostęp
- multicommodity flow* – przepływ wieloskładnikowy
- multiservice network* – sieć wielousługowa
- multivalued explicit feedback* – wielowartościowa bezpośrednia informacja zwrotna
- negative acknowledge* – powiadamianie negatywne
- nested modules* – hierarchia zagnieżdżonych modułów
- network edge* – brzeg sieci
- network performance* – jakość działania sieci
- network resource allocation* – alokacja zasobów sieci
- network resource management* – zarządzanie zasobami w sieci
- network-aware applications* – aplikacje dopasowane do zmienności dostępnych zasobów sieci
- network access level flow control* – sterowanie przepływem na poziomie dostępu do sieci
- network dimensioning* – wymiarowanie sieci
- network layer* – warstwa sieciowa
- network overprovision* – przewymiarowanie sieci
- network power* – moc sieci
- network service* – usługa sieci
- nonadaptive flow* – nieadaptacyjny przepływ
- nonconforming connections* – połączenia naruszające warunki kontraktu
- noncooperative flows* – przepływy nie-współpracujące
- nonpersistent CSMA* – metoda CSMA bez wymuszania transmisji
- non-backlogged flow* – przepływ bez zaległości
- non-work conserving* – szeregowanie niezachowujące pracy
- offered load* – ruch oferowany do sieci przez zbiorowość użytkowników
- on/off source* – źródło dwustanowe
- on-the-fly processing* – przetwarzanie „w locie”
- out of profile* – ruch niespełniający warunków profilu
- overflow* – przepełnienie
- overload* – przeciążenie

- overall internal congestion* – całkowite przeciążenie
- overlay model* – model pokrycia
- persistent CSMA* – metoda CSMA z wymuszaniem transmisji
- pacing method* – metoda krokowa
- pacing window* – metoda kroczącego okna
- packet disassembly* – rozformowanie pakietów
- packet discard function* – funkcja znakowania (odrzućcia) pakietów
- packet-switched network* – sieć z komutacją pakietów
- packetization delay* – opóźnienie pakietyzacji
- parameter-based admission control* – parametryczne sterowanie dostępem
- path flow control* – sterowanie przepływem na trasie
- peak capacity profile* – maksymalny profil pojemności
- penalty function* – funkcja kary
- perfect scheduling* – idealne szeregowanie
- permanent virtual circuit* – stałe połączenie wirtualne
- permit* – przepustka (pozwolenie, żeton)
- persistent CSMA* – trwała metoda CSMA
- per-aggregate queueing* – kolejkowanie zorientowane na agregaty przepływów
- per-class scheduling* – szeregowanie zorientowane na jednostki danych klas ruchu
- per-connection caching* – dynamiczne buforowanie dla połączeń
- per-conversation traffic handling* – podejście sesyjne do obsługi ruchu
- per-flow allocation* – dedykowana przepływowi alokacja pojemności pamięci buforowej
- per-flow queueing* – kolejkowanie zorientowane na pojedyncze przepływy
- per-flow resource reservation* – rezerwacja zasobów dla przepływów
- per-flow scheduling* – szeregowanie zorientowane na jednostki danych przepływów
- per-route caching* – dynamiczne buforowanie dla tras
- physical interface module* – moduł fizycznego interfejsu
- physical layer* – warstwa fizyczna
- piggybacked* – niejawnie (przepustki potwierdzenia)
- play-out buffer* – odbiorczy bufor wyrównawczy
- play-out point* – czas (przesunięcie) odtwarzania
- point-to-point connection* – połączenie typu punkt–punkt
- positive acknowledge* – powiadamianie pozytywne
- predictability* – przewidywalność
- predictive coding* – ramka z predykcją ruchu (ramka MPEG typu P)
- preemptive* – priorytet pierwszeństwa obsługi
- premarking* – wstępne znakowanie jednostek danych
- presentation layer* – warstwa prezentacji
- preventive control* – sterowanie prewencyjne
- price* – cena (miara przeciążenia)
- price differentiation* – różnicowanie cen
- priority levels* – poziomy priorytetów
- processing delay* – opóźnienie przetwarzania
- processor sharing* – współdzielenie czasu procesora
- profile conformant* – ruch zgodny z profilem
- profile nonconformant* – ruch niezgodny z profilem
- propagation delay* – opóźnienie propagacji
- push-out mechanism* – mechanizm wypychania
- quality differentiation parameters* – parametry różnicowania jakości
- QoS* – jakość usług
- QoS assurance* – gwarancja jakości usług (dla aplikacji)
- QoS budget* – budżet jakości usług
- QoS chain* – łańcuch jakości usług
- QoS metric* – miara jakości usług
- QoS parameter mapping* – odwzorowywanie (translacja) wartości parametrów jakości usług
- QoS policy* – polityka dostarczania jakości usług
- QoS system* – system jakości usług
- QoS-enabled network* – sieć umożliwiająca różnicowanie jakości usług
- quantum* – kwota (liczba) bitów
- queue mismatch* – niedopasowanie długości kolejek
- queuing delay* – opóźnienie kolejkowania
- random drop* – losowe odrzucanie (pakietów)
- rate based algorithms* – algorytmy szeregowania warunkowego
- rate consumption rate* – współczynnik zawłaszczania szybkości transmisji
- rate indication* – wskazanie szybkości transmisji

- rate mismatch* – niedopasowanie szybkości transmisji i pojemności łącza
- rate variability* – zmienność szybkości
- reactive control* – sterowanie reakcyjne
- reassembly* – desegmentacja (odtworzenie) jednostek danych
- reassembly buffers* – pamięć buforowa odtwarzania
- reassembly deadlock* – zakleszczenie kompletacji
- reassembly deadlock at the destination* – zakleszczenie kompletacji w ujściu
- reassembly delay* – opóźnienie odtwarzania
- real-time sensitive traffic* – ruch wrażliwy na obsługę w czasie rzeczywistym
- recovery from loss* – odtwarzanie utraconych jednostek danych
- relative differentiated services* – względne różnicowanie jakości usług
- reliable flooding* – niezawodne zalewanie
- reliability* – niezawodność
- requirement matrix* – macierz wymagań
- resequence* – odtwarzanie kolejności jednostek danych
- resequence deadlock* – zakleszczenie odtwarzania kolejności jednostek danych
- reservation setup request* – żądania ustanowienia rezerwacji
- reservation protocol* – protokół rezerwacji (zasobów)
- reservation tree* – drzewo rezerwacji
- resource allocation* – przydzielanie (alokacja) zasobów
- resource conserving* – zasada zachowania zasobów
- resource controller* – sterownik zasobów
- resource provisioning* – udostępnianie zasobów
- retransmission control* – sterowanie retransmisjami jednostek danych
- retransmission model* – model obsługi z retransmisjami jednostek danych
- retransmission time-out* – zwłoka czasowa retransmisji
- robustness* – odporność na zmiany ilościowe ruchu
- round-trip-delay* – opóźnienie potwierdzenia (powiadomienia)
- route pinning* – kotwiczenie trasy
- route table* – tablica tras
- routing algorithm* – algorytm wyznaczania tras
- routing granulation* – granulacja routingu
- routing protocol* – protokół wyboru (wyznaczania) tras
- scheduling fairness* – sprawiedliwość szeregowania
- scheduling flexibility* – elastyczność dyscypliny szeregowania
- segmentation* – segmentacja (dzielenie) jednostek danych
- selective ARQ* – korekcja poprzez selektywne retransmisje
- selective congestion* – przeciążenie wybranych zasobów sieci (wybiórcze)
- selective discarding* – odrzucanie wybiórcze (selektywne)
- selective flow control* – wybiórcze (selektywne) sterowanie przepływem
- sequence deadlock* – zakleszczenie sekwencji
- sequence number* – numer sekwencyjny
- Service Access Point (SAP)* – punkt dostępu do usług warstw
- service allocation* – alokacja usług
- Service Data Unit (SDU)* – jednostka danych warstwy
- session layer* – warstwa sesji
- service class* – klasa obsługi ruchu
- service mapping* – odwzorowanie usług
- setup connection delay* – opóźnienie ustanowienia połączenia
- setup protocol mappings* – odwzorowanie procedury rezerwacji
- shared buffer pool* – współdzielenie całkowitej pojemności pamięci buforowej
- shared reservation* – współdzielona rezerwacja
- shortest path* – najkrótsza trasa
- shortest processing time* – najkrótszy czas przetwarzania
- signalling procedure* – procedura sygnalizacyjna
- simple sum algorithm* – algorytm prostego sumowania
- simple traffic* – strumień prosty ruchu
- simplex reservation* – rezerwacja jednokierunkowa
- sink tree* – drzewo (tras) ujścia
- skew* – asymetria
- sliding window* – metoda przesuwającego okna
- slow path* – wolna ścieżka
- slow start procedure* – procedura spowolnionego startu
- slowdown mechanism* – mechanizm spowalniania

- source-destination based routing* – routing zorientowany na źródła przepływów
- source utilization* – współczynnik wykorzystania źródła
- speed matching* – dopasowanie szybkości transferu
- splitting algorithm* – algorytm rozdziału (ruchu)
- statistical multiplexing* – zwielokrotnianie statystyczne
- statistical bandwidth* – pojemność statystyczna
- stop-and-wait ARQ* – procedura korekcji poprzez pojedyncze retransmisje
- store-and-forward* – zapamiętaj i prześlij dalej (sposób obsługi jednostek danych)
- store-and-forward deadlock* – zakleszczenie bezpośrednio
- strict prioritization* – całkowite priorytetowanie
- strongly controlled network* – sieć silnie sterowana
- superposition of arrival streams* – superpozycja strumieni zgłoszeń
- synchronous transfer mode* – synchroniczny tryb transferu
- system potential* – potencjał systemu
- switched virtual circuit* – tymczasowe połączenie wirtualne
- switching* – komutacja
- switching delay* – opóźnienie komutacji (przełączania)
- table-based hashing* – pośrednie zastosowanie funkcji skrótu
- tail drop* – odrzucanie „ogona” (pakietów)
- terminal cluster controller* – sterownik skupienia urządzeń końcowych
- terminal device* – urządzenie końcowe
- three-way handshake* – trzyetapowa procedura potwierdzenia gotowości
- threshold function* – funkcja progowa
- threshold mechanism* – mechanizm progowy
- throughput* – przepustowość
- throughput degradation* – degradacja przepustowości
- time window flow control* – sterowanie przepływem z tzw. oknem czasowym
- time-out* – zwłoka czasowa
- token arrival rate* – szybkość napływu znaczników
- token loss probability* – prawdopodobieństwo utraty znaczników
- token pool* – zbiór (puła) znaczników
- Top-to-Bottom QoS* – koncepcja hierarchicznego zagnieżdżenia usług
- traffic* – ruch teleinformatyczny
- traffic classification* – klasyfikacja ruchu
- traffic classifier* – klasyfikatory ruchu
- traffic conditioning* – kondycjonowanie ruchu
- traffic contract* – kontrakt ruchowy
- traffic engineering* – inżynieria ruchu teleinformatycznego (zarządzanie ruchem)
- traffic granulation* – granulacja ruchu
- traffic identification* – identyfikacja ruchu
- traffic marker module* – moduł znakowania ruchu
- traffic meter module* – moduł pomiaru ruchu
- traffic policing* – kontrolowanie ruchu
- traffic shaper module* – moduł kształtowania ruchu
- traffic shaping* – kształtowanie ruchu
- traffic smoothing* – wygładzanie ruchu
- traffic splitter* – rozdzielanie ruchu
- traffic starvation* – wygłodzenie ruchu (efekt)
- traffic-class routing* – routing zorientowany na klasy ruchu
- traffic-trunk routing* – routing zorientowany na magistrale
- traffic-engineered path* – trasa kierunkowa
- trailing edge* – strzałka końcowa (metoda okna)
- transfer delay* – opóźnienie transferu jednostek danych
- transit packets* – pakiety tranzytowe
- transient congestion* – przejściowe (chwilowe) przeciążenie
- transmission channel bandwidth* – pojemność (pasmo) kanału transmisyjnego
- transmission control* – sterowanie transmisją
- transmission delay* – opóźnienie transmisji
- transport layer* – warstwa transportowa
- transport level flow control* – sterowanie przepływem na poziomie transportowym
- underutilization* – nie w pełni wykorzystane zasoby
- unfair resource allocation* – niesprawiedliwy podział zasobów
- unsolicited downstream* – przydzielanie „niezamówione” (etykiet)
- usage parameter control* – nadzorowanie kontraktów ruchowych
- virtual circuit network* – sieć połączeń wirtualnych
- virtual connection* – połączenie wirtualne

<i>virtual route</i> – trasa wirtualna	<i>well-behaved traffic</i> – ruch zgodny z kontraktem ruchowym
<i>virtual route pacing</i> – metoda krokowa sterowania przepływem	<i>widest path</i> – najszersza trasa
<i>waiting interval</i> – przedział (czasu) oczekiwania	<i>window method</i> – metoda okna
<i>weakly controlled network</i> – sieć słabo sterowana (bez sterowania przepływem na poziomie łącza danych)	<i>window size</i> – rozmiar okna
<i>weight coefficient</i> – współczynnik wagowy	<i>window size indicator</i> – wskaźnik rozmiaru okna
<i>weighted max-min flow</i> – ważony przepływ maksymalno-minimalny	<i>wire speed</i> – szybkość transmisji w medium
	<i>work conserving</i> – szeregowanie zachowujące pracę
	<i>worst-case-delay</i> – opóźnienie maksymalne (w tzw. najgorszym przypadku)

ZESTAWIENIE SKRÓTÓW

ABR	– Available Bit Rate	CLR	– Cell Loss Rate
ACK	– Acknowledge	CMR	– Cell Misinsertion Rate
AF PHB	– Assured Forwarding Per Hop Behaviour	CoS	– Class of Service
AS	– Assured Service	CP	– Complete Partitioning
ALL	– Allocation	CRC	– Cyclic Redundancy Checking
APPN	– Advanced Peer-to-Peer Networking	CRR	– Credit Round Robin
AQM	– Active Queue Management	CS	– Complete Sharing
AQMCA	– Active Queue Management for Congestion Avoidance	CSMA	– Carrier Sense Multiple Access
ARQ	– Automatic Retransmission Request	CSMA/CD	– Carrier Sense Multiple Access with Collision Detection
ATM	– Asynchronous Transfer Mode	CSMA/CA	– Carrier Sense Multiple Access with Collision Avoidance
BA	– Behaviour Aggregate	CTD	– Cell Transfer Delay
BAC	– Behaviour Aggregate Classifier	CVDT	– Cell Variation Delay Tolerance
BAM	– Bandwidth Allocation Mechanism	DCE	– Data Circuit Equipment
BB	– Bandwidth Broker	DLC	– Data Link Control
BECN	– Backward Explicit Congestion Notification	DCT	– Discrete Cosine Transform
BER	– Bit Error Ratio	diffServ	– Differentiated Services
B-ISDN	– Broadband Integrated Services Digital Network	DNA	– Digital Network Architecture
BRFQ	– Bit-Round Fair Queueing	DRR	– Deficit Round Robin
BS	– Binary Search	DS	– Differentiated Services
CAC	– Call Admission Control	DSCP	– Differentiated Service Code Point
CBFQ	– Credit-Based Fair Queueing	DT	– Drop Tail
CBR	– Constant Bit Rate	DTE	– Data Terminal Equipment
CDV	– Cell Delay Variation	DTFC	– Drop-and-Throttle Flow Control
CDW	– Cell Delay Window	DWSP	– Dynamic Window Sizing Procedure
CER	– Cell Error Rate	EC	– Equivalent Capacity
CHPS	– Choke Packet Scheme	ECN	– Explicit Congestion Notification
CIR	– Committed Information Rate	EDF	– Earliest Deadline First
CIR	– Cell Insertion Rate	EF PHB	– Expedited Forwarding Per Hop Behaviour
CLS	– Controlled Load Services	ER	– Explicit Route
		ERD	– Early Random Drop

EWMA	– Exponentially Weighted Moving Average	MCR	– Minimum Cell Rate
FA	– Flow Approximation to Loss Rate	MF	– Multifield Classifier
FBR	– Fast Buffer Reservation	MFP	– Multicommodity Feasible Path
FCFS	– First Come First Served	MPEG	– Moving Picture Expert Group
FEC	– Forwarding Equivalence Class	MMBP	– Markov Modulated Bernoulli Process
FECC	– Forward Error Correcting Coding	MPLS	– Multiprotocol Label Switching
FECN	– Forward Explicit Congestion Notification	MPSB	– Multiple Processors – Shared Bus
FFQ	– Fluid Fair Queueing	MPSS	– Multiple Processors – Space Switching
FQ	– Fair Queueing	MW	– Moving Window
FR	– Frame Relay	NA	– Non-parametric Approach
FRP	– Fast Retransmission Procedure	NACK	– Negative Acknowledge
FRP	– Fast Recovery Procedure	NCP	– Network Control Program
GA	– Gaussian Approximation	nrt-VBR	– non real time – Variable Bit Rate
GOP	– Group of Pictures	NWFQ	– Non-ideal Weighted Fair Queueing
GPS	– Generalized Processor Sharing	N-ISDN	– Narrowband Integrated Services Digital Network
GR	– Gradual Restriction	OS	– Olympic Service
GS	– Guaranteed Services	OSI	– Open Systems Interconnection
G-RED	– Gentle-Random Early Detection	OSPF	– Open Shortest Path First
HBTS	– Hashing-based Traffic-splitting	PBR	– Peak Bit Rate
HLS	– Hierarchical Link Sharing	PC	– Power Complexity
HTA	– Heavy Traffic Approximation	PCR	– Peak Cell Rate
HWFQ	– Hierarchical Weighted Fair Queueing	PDM	– Proportional Differentiation Model
IBLS	– Input Buffer Limit Scheme	PDP	– Proportional Delay Differentiation
IGRP	– Interior Gateway Routing Protocol	PGPS	– packet-by-packet GPS
intServ	– Integrated Services	PHB	– Per-Hop Behaviour
IS	– Integrated Services	PLRD	– Proportional Loss Rate Differentiation
ISO	– International Standard Organization	PMP	– Paris Metro Pricing
ISP	– Internet Service Provider	PQ	– Premium Queue
IETF	– Internet Engineering Task Force	PSE	– Packet Switching Exchange
IWFQ	– Ideal Weighted Fair Queueing	PUL	– Packet Underway Limit
JW	– Jumping Window	QMCR	– Queue Management for Congestion Recovery
LB	– Leaky Bucket	QoS	– Quality of Services
LCFS	– Last Come First Served	rt-VBR	– real time – Variable Bit Rate
LDP	– Label Distribution Protocol	RBA	– Reassembly Buffer Allocation
LIFOS	– Last-In First-Out Stack	RC	– Rule Complexity
LR	– Leaf Router	RD	– Random Drop
LSP	– Label Switched Path	RDS	– Relative Differentiated Services
LSR	– Label Switched Router	RED	– Random Early Detection
MAC	– Medium Access Control	REM	– Random Exponential Marking
MBS	– Maximum Burst Size		

REQALL	– Request for Allocation	SPSB	– Single Processor – Shared Bus
RFNM	– Ready For Next Message	SSP	– slow start procedure
RIO	– Random Early Detection (RED) with In and Out	TDM	– Time Division Multiplexing
RIP	– Routing Information First	TC	– Time Complexity
RM OSI	– Reference Model of Open Systems Interconnections	TCA	– Traffic Conditioning Agreement
RR	– Receive Ready	TCP/IP	– Transmission Control Proto- col/Internet Protocol
RR	– Round Robin	TCP	– Transmission Control Protocol
RNR	– Receive Not Ready	ToS	– Type of Service
RS	– Rectangular Search	TS	– Transport Station
RSVP	– Resource Reservation Protocol	TJW	– Triggered Jumping Window
SAP	– Service Access Point	UBR	– Unspecified Bit Rate
SBP	– Structured Buffer Pool	UC	– Update Complexity
SBR	– Sustainable Bit Rate	UDP	– User Datagram Protocol
SC	– Space Complexity	VBR	– Variable Bit Rate
SCFQ	– Self-Clocked Fair Queueing	VC	– Virtual Connection
SCR	– Sustainable Cell Rate	VC	– Virtual Channel
SDU	– Service Data Unit	VLL	– Virtual Leased Line
SECBR	– Severely Errored Cell Block Ratio	VP	– Virtual Path
SLA	– Service Level Agreement	VR	– Virtual Route
SMA	– Sharing with Minimum Allocation	WFQ	– Weighted Fair Queueing
SMAMQ	– Sharing with Minimum Allocation and Maximum Queue	W2FQ	– worst-case fair – Weighted Fair Queueing
SMQ	– Sharing with Maximum Queues	WRR	– Weighted Round Robin
SNA	– Systems Network Architecture		
SPPSS	– Shared Parallel Processing – Space Switching		

SKOROWIDZ

- Active Queue Management (AQM) 299, 303, 309
- adresowanie skrótu 437
- AQM for Congestion Avoidance (AQMCA) 298
- Advanced Peer-to-Peer Networking (APPN) 273
- agregowanie przepływów 564, 603
- agregaty strumieni ruchu 388, 497
- algorytm Bellmana–Forda 576
 - Dijkstry 580
 - ciekącego wiadra 44, 45, 351, 412, 479
 - filtracji geometryczny 537
 - klasyfikacji 530, 536
 - – odmiany 531
 - – dynamiczne buforowanie 536
 - – podejście geometryczne 537
 - – przestrzeń krotek 542
 - – BS 543
 - – RS 545
 - przeciwdziałania przeciążeniu 301
 - rozdzielania zasobów 588, 590
 - – o strukturze drzewa 590
 - – FCFS 592
 - – z funkcją skrótu 595
 - stanu łączy 576
 - sterowania dostępem:
 - – obszarów akceptacji 409
 - – prostego sumowania 409
 - – równoważnych zasobów 409
 - – sumy mierzonej 409
 - szeregowania 149, 447
 - – sprawiedliwego 447, 448
 - – – właściwości 460
 - – warunkowego 447, 449
 - – ważonego sprawiedliwego 450
 - z gwarantowanym czasem obsługi 447, 448, 453
 - znakowania pakietów 305, 311, 309
- algorytm wektorów odległości 576
 - wyznaczania przepływów maksymalno-minimalnych 131
 - wyznaczania tras QoS 577, 579
- alokacja pojemności 316
 - – adaptacyjna 517
 - usług 517
 - zasobów sieci 158, 291, 441
 - – maksymalno-minimalna 445
 - – sprawiedliwa 444
- aplikacje krytyczne 447
 - wrażliwe na obsługę w czasie rzeczywistym 36
 - wrażliwe na opóźnienia 386
 - tolerujące opóźnienia 36
- architektura routerów 525
 - MPLS 599
- asymetria 54
- Assured Forwarding Per Hop Behaviour (AF PHB) 511
- Assured Service (AS) 514
- Assured Queue (AQ) 512, 515
- Asynchronous Transfer Mode (ATM) 39, 48, 60, 287
- Automatic Retransmission Request (ARQ) 98
- Available Bit Rate (ABR) 290

- Backward Explicit Congestion Notification (BECN) 293
- Bandwidth Allocation Mechanism (BAM) 517
- Bandwidth Broker (BB) 511, 518
- Behaviour Aggregate (BA) 505
- Behaviour Aggregate Classifier (BAC) 505
- bezwładność procedur 376
 - sieci 488
- Binary Search (BS) 545
- Bit Error Ratio (BER) 39

- Bit-Round Fair Queuing (BRFQ) 445, 459
- Broadband Integrated Services Digital Network (B-ISDN) 38
- broker pasma 511, 518
- budżet QoS 67
- bufor odtwarzania 80
- buforowanie dynamiczne 536
 - wyrównawcze 52
- Call Admission Control (CAC) 315
- Carrier Sense Multiple Access (CSMA) 169
- Choke Packet Scheme (CHPS) 255
- cele sterowania przepływem na poziomie:
 - dostępu do sieci 190, 253
 - łącza danych 190, 191, 254
 - transportu 189, 268
 - trasy 190, 219, 232, 236, 254
- cele zapobiegania przeciążeniom 159
- Cell Delay Variation (CDV) 290
- Cell Error Rate (CER) 290
- Cell Insertion Rate (CIR) 41, 292
- Cell Loss Rate (CLR) 40, 290, 335
- Cell Misinsertion Rate (CMR) 290
- Cell Transfer Delay (CTD) 290
- Cell Variation Delay Tolerance (CVDT) 290
- cena przeciążenia 302, 310
- Class of Service (CoS) 604
- Complete Partitioning (CP) 196
- Complete Sharing (CS) 216
- Constant Bit Rate (CBR) 290
- Credit Round Robin (CRR) 488
- Credit-Based Fair Queueing (CBFQ) 491
- Cyclic Redundancy Checking (CRC) 439
- czas oczekiwania na odtwarzanie 54
 - na obsługę 413
 - – maksymalny 413
 - odtwarzania 52, 55
 - międzyzłozeniowy 48
 - rozłączenia połączenia 37
 - stanowiska obsługi 464
 - ustanowienia połączenia 37
- Data Circuit Equipment (DCE) 188
- Data Terminal Equipment (DTE) 188
- Deficit Round Robin (DRR) 487
- degradacja przepustowości 160
 - spowodowana niedopasowaniem zasobów 160
 - spowodowana metodą dostępu 165
- degradacja jakości usług 568
- desegmentacja 72
- detekcja naruszeń 349
- Differentiated Services (diffServ) 497
- Differentiated Service Code Point (DSCP) 505
- Discrete Cosine Transform (DCT) 82
- dławienie szybkości 113
- domeny usług 510
- dopasowanie szybkości 160
- Drop Tail (DT) 303
- Drop-and-Throttle Flow Control (DTFC) 264
- dyscyplina szeregowania CBFQ 491
 - – CRR 488
 - – DRR 487
 - – EDF 448, 453
 - – FCFS 375, 441, 459, 592
 - – FQ 450
 - – FFQ 490
 - – GPS 450, 453, 458
 - – IWFQ 556
 - – HWFQ 482
 - – PGPS 471
 - – RR 444, 594
 - – SCFQ 485
 - – WFQ 450, 471
 - – W2FQ 481
 - – WRR 486
- Early Random Drop (ERD) 303
- Earliest Deadline First (EDF) 448
- efekt propagacji wstecznej 275
- elastyczność dyscypliny szeregowania 458
- elementarna stopa błędów 39
- eliminacja duplikatów jednostek danych 268
- Equivalent Capacity (EC) 328
- etykieta 599
- Expedited Forwarding Per Hop Behaviour (EF PHB) 511, 513
- Explicit Congestion Notification (ECN) 301, 304
- Explicit Route (ER) 245
- Exponentially Weighted Moving Average (EWMA) 373
- Fair Queueing (FQ) 450
- Fast Buffer Reservation (FBR) 322
- filtr 304, 533
 - dwuwymiarowy 538
- filtracja 428, 528, 577
- First Come First Served (FCFS) 375, 441

- Flow Approximation to Loss Rate (FA) 335
- fluktuacja opóźnień 293
- Fluid Fair Queueing (FFQ) 490
- Forward Explicit Congestion Notification (FECN) 293
- Frame Relay (FR) 292
- fluktuacja opóźnień 48
- Forwarding Equivalence Class (FEC) 600
- funkcja kary 118
 - skrótu 435, 595
 - – podwójna 440
 - mieszająca 435
 - znakowania (odrzućcia) pakietów 305, 309, 311
- Gaussian Approximation (GA) 319, 334
- Generalized Processor Sharing (GPS) 450, 453, 458
- Gentle-Random Early Detection (G-RED) 306
- Gradual Restriction (GR) 255
- granulacja ruchu 583
- Hashing-based Traffic-splitting (HBTS) 595
- Heavy Traffic Approximation (HTA) 346
- Hierarchical Link Sharing (HLS) 555
- Hierarchical Weighted Fair Queueing (HWFQ) 482
- hierarchia procedur sterowania przepływem 275
 - zagnieżdżonych modułów 59
 - zarządzania przepływami 396
- Ideal Weighted Fair Queueing (IWFQ) 556
- identyfikacja jednostek danych przepływu 410, 434
- Input Buffer Limit Scheme (IBLS) 256
- Integrated Services (intServ) 404
- Interior Gateway Routing Protocol (IGRP) 576
- inżynieria ruchu 395, 568
- jakość działania sieci 394, 502
 - usług 391, 420
 - – charakterystyki 421
 - – dostarczana 391
 - – łańcuch 26
 - – miary 36
 - – parametry 502
 - – polityka 395
 - – żądana 392
- jakość sterowania przepływem 295
- Jumping Window (JW) 372
- kanał wirtualny 357
- klasa ruchu 483
 - spedykcji ruchu 387, 497
- klasy równoważności 600
- klasyfikacja jednostek danych 499
 - dwuwymiarowa 537, 549
 - jednowymiarowa 548
 - ruchu 189, 504
- klasyfikator ruchu 504
 - agregatów 505
 - wielopolowy 505
 - cechy 532
 - miary 532
- kodowanie mieszające 435
- kolejkowanie agregatów przepływów 556
- kolizja funkcji mieszającej 437
- kompensowanie opóźnień 80
- komutacja etykiet 395
 - komórek 289
 - łączy 36
 - pakietów 381, 397
 - ramek 292
 - wiadomości 36
- konceptcja usług zintegrowanych 382, 385, 404, 568
 - usług zróżnicowanych 382, 387, 497, 568
- kondycjonowanie ruchu 504, 507
- kontrakt ruchowy 289, 502, 518
 - – dynamiczny 518
 - – statyczny 518
 - usługowy 388, 502
 - wymuszanie 292
- kotwiczenie tras 516
- kryteria optymalizacji 107, 406, 579
- kształtowanie ruchu 348
- kwota (liczba) bitów 487
- Label Switched Path (LSP) 599
- Label Switched Router (LSR) 599
- Label Distribution Protocol (LDP) 601
- Last-In First-Out Stack (LIFOS) 600
- Leaky Bucket (LB) 44, 45, 351, 412, 479
- liczba poziomów priorytetów 298
- licznik deficytu 487
- macierz ruchu 108
 - wymagań 108

- Markov Modulated Bernoulli Process (MMBP) 359
- marnotrawienie zasobów 391
- Maximum Burst Size (MBS) 290
- mechanizm detekcji przeciążenia 304
 - identyfikacji jednostek danych 434
 - inżynierii ruchu 573
 - oddziaływania zwrotnego 254
 - pakietu dławiącego 255
 - progowy 303, 375
 - spowalniania 215, 254
 - sterowania przepływem 137
 - – bezpośredni 394
 - – pośredni 394
 - szeregowania 443
 - wypychania 374, 560
- mechanizm okna 135, 371
 - – czasowego 135
 - – hierarchia 151
 - – kredytowy 293
 - – przesuwne 135
- mechanizmy przeciwdziałania przeciążeniom:
 - ATM 289
 - FR 292
 - TCP/IP 293
- Medium Access Control (MAC) 62
- metoda adaptacyjna krokowa 274
 - alokacji maksymalno-minimalnej 445
 - pojemności 316
 - izarytmiczna 255
 - kredytów 268
 - krokowa 246, 268
 - ograniczonej pojemności buforów wejściowych 255, 259
 - okna 135, 252, 255
 - – czasowego 135
 - – przesuwne 135
 - – kroczącego 139, 268
 - pakietów dławiących 255, 264
 - równoważenia obciążenia 575, 581, 587
 - stopniowych ograniczeń 255
- metody dostępu:
 - CSMA 169
 - losowe 170
 - prosty schemat Aloha 165
 - szczelinowy schemat Aloha 168
- miara bezwzględna 389
 - algorytmów klasyfikacji 531
- miara jakości dyscyplin szeregowania 454
 - – połączeń 36
 - – transferu 36
 - – działania 502
 - przeciążenia 310
- Minimum Cell Rate (MCR) 290
- moc sieci 111
- model kolejkowy 107, 220
 - – połączenia bez procedury sterowania 220
 - – – z procedurą sterowania 222
 - – – z lokalną procedurą sterowania 223
 - – przepływu na trasie 226
 - mechanizmu RED 307
 - obsługi ruchu z opóźnieniem 174
 - – z retransmisjami 174
 - – ze stratami (odrzuconiami) 176
 - pokrycia 398
 - przeciążenia 309
 - przepływowy ruchu 44, 450
 - sieci i trasy w sieci 584
 - usług zintegrowanych 382, 385, 404, 602
 - – gwarantowanej usługi 385, 411, 418
 - – model funkcjonalny 405
 - – PHB 514
 - – sterowanego obciążenia 386, 420
 - usług zróżnicowanych 382, 387, 497, 603
 - – AS 513, 518
 - – Premium 514, 520, 603
- modele kolejkowe 220
 - przepływów 107
 - różnicowania usług 548
 - zapewniania jakości 602
- moduł dyspozytora 443
 - klasyfikacji ruchu 505, 528
 - kształtowania ruchu 508
 - odrzucania jednostek danych 509
 - pomiaru ruchu 508
 - znakowania ruchu 508
- monitorowanie stanu sieci 575
- Moving Picture Expert Group (MPEG) 83
- Moving Window (MW) 373
- Multicommodity Feasible Path (MFP) 586
- Multifield Classifier (MF) 505
- Multiple Processors – Shared Bus (MPSB) 526
- Multiple Processors – Space Switching (MPSS) 526
- Multiprotocol Label Switching (MPLS) 597
- nadzorowanie kontraktów ruchowych 291

- Network Control Program (NCP) 270
 niedopasowanie długości kolejek 310
 – szybkości transmisji 310
 nierównomierność obciążenia 571, 588
 niezawodność 69–71, 75, 76, 79, 80, 82, 85
 Nonparametric Approach (NA) 340
 non real time – Variable Bit Rate (nrt-VBR) 290
 Non-ideal Weighted Fair Queueing (NWFQ) 557
- obciążenie systemu obsługi 124, 557, 568
 – równomierne 396
 – nierównomierne 571
 – zasobów 396
- obsługa ruchu:
 – bez gwarancji jakości 293, 381, 499
 – – różnicowanie 515
 – z gwarancją jakości
 – – usług zintegrowanych 382, 404
 – – – zróżnicowanych 382, 497
- odrzucanie adaptacyjne 517
 – losowe 299
 – „ogonów” 299, 303
 – z czoła kolejki 299
 – selektywne 374
- odtwarzanie kolejności 103, 209, 233, 268
 – utraconych pakietów 268
 – wiadomości 54, 181, 233, 268
- odwzorowanie parametrów jakości usług 70
 – procedury rezerwacji 493
 – usług 493
- ograniczanie opóźnień 441
- Olympic Service (OS) 513
- Open Shortest Path First (OSPF) 577
- opóźnienie dyscypliny szeregowania 455
 – kodowania 42
 – kolejkowania 43
 – kompletacji 479
 – komutacji 43
 – maksymalne 73, 82, 144, 408, 460
 – – gwarantowane 411
 – na trasie 144
 – obsługi przepływu 414
 – pakietyzacji 42, 412, 479
 – potwierdzenia 307
 – propagacji 43, 307
 – przetwarzania 44, 412
 – rozłączenia (znoszenia) połączenia 37
 – średnie 407
 – transferu 41
- opóźnienie transmisji 43
 – ustanawiania połączenia 37
- optymalizacja jakości działania sieci 395
 – jednokryterialna 107
 – wielokryterialna 406
- organizacja jakości usług hierarchiczna 26
 – szeregową 26
- packet-by-packet GPS (PGPS) 471
- pakiet dławiący 265
 – tranzytowy 265
 – wejściowy 265
- pakietyzacja wiadomości 268
- pamięć buforowa odtwarzania 269
 – – wyrównawcza 49
 – – znaczników 44, 351
- parametry różnicowania jakości klas ruchu 390
- Paris Metro Pricing (PMP) 550
- Peak Cell Rate (PCR) 290
- Per-Hop Behaviour (PHB) 502
- podejście sesyjne 385
- podział pamięci buforowej całkowity 193, 195
 – proporcjonalny 194
 – proporcjonalny nadmiarowy 194
 – mieszany 194
 – równomierny 193, 196
 – z maksymalną długością kolejki 197
 – z minimalną długością kolejki 197
- pojemność (głębokość) bufora znaczników 44
- polityka jakości usług 22
 – udostępniania zasobów 579
- połączenia naruszające warunki kontraktu 349, 502
 – nienaruszające warunków kontraktu 349, 502
- poziom odrzucania jednostek danych 512
 – priorytetu odrzucania 512
 – przepływu 407
 – sterowania 406
- poziom zwielokrotniania statystycznego 318
- pojemność dostępna 579
 – minimalna 408
 – – gwarantowana 411, 479
 – oczekiwana 479, 511
 – równoważna 329
 – stała 341
 – statystyczna 318
 – współdzielenie 553
 – żądana 321
- połączenie wirtualne jednorazowe 208

- połączenie wirtualne sekwencyjne 208
 - – stałe 208
 - – tymczasowe 208
- potencjał przepływu 557
 - systemu 557
- potwierdzenie niejawne 138
- prawdopodobieństwo akceptacji połączenia 37
 - blokady połączenia 37
 - blokady przyjmowania zgłoszeń 230
 - nadmiaru żądań (popytu) 322
 - odrzucania 304
 - odrzucania wywołań 304, 376
 - naruszenia kontraktu 372
 - przekroczenia zasobów 327
 - przepełnienia pamięci buforowej 331
 - rywalizacji 325
 - utraty:
 - – jednostek danych 38, 339, 341, 407
 - – znaczników
 - znakowania pakietów 305, 311 309
 - żądania dostępu 325
- Premium Queue (PQ) 516
- Premium Service 514
- priorytetowanie 447, 500
 - proste 447
- procedura dynamicznego wymiarowania okna 294
 - formowania 419
 - krocącego okna 139
 - łączenia bloków 293
 - nadzorowania 419
 - obsługi wywołania 422
 - przyjmowania segmentów 293
 - retransmisji bloków 293
 - – grupowej 100, 249
 - – pojedynczych 99
 - – selektywnej 101
 - rezerwacji zasobów 389
 - spowolnionego startu 294
 - sterowania dostępem 406
 - szybkiego odtwarzania 294
 - szybkiej retransmisji 294
 - trzyetapowa potwierdzenia gotowości 271
 - zarządzania ruchem 500
 - wyprowadzania bloków 293
 - wyznaczania tras 395, 415
 - – – z ograniczeniami 500, 578
- procedury sterowania przepływem:
 - cele 113
 - implementacja 114, 135
- procedury sterowania przepływem:
 - hierarchia 275
 - integracja i optymalizacja 276
 - integracja ruchu 277
 - integracja trybów komutacji 277
 - zarządzanie retransmisjami 278
 - w sieciach heterogenicznych 283
 - w trybie rozgłoszeniowym 282
 - wyznaczanie tras 280
- procedury sterowania przepływem wyróżniane ze względu na:
 - czas przebywania w sieci 193, 199
 - połączenia wirtualne 193, 206
 - strumienie wyjściowe 192, 194
- procedury sterowania przepływem w połączeniach wirtualnych:
 - jednorazowe 210
 - krokowe 210
- proces użytkownika 64
- profil pojemności 516
- propagacja wsteczna 153, 192, 213
- Proportional Delay Differentiation (PDD) 552
- Proportional Differentiation Model (PDM) 553
- Proportional Loss Rate Differentiation (PLRD) 552
- protokół dystrybucji etykiet 597
 - rezerwacji zasobów 385, 423
 - – RSVP 423
 - sygnalizacyjny 423, 597
 - wyznaczania trasy 385
- próg zajętości pamięci buforowej 152
- przeciążenia:
 - likwidacja (regeneracja) 159, 297
 - hierarchia 188, 243
 - powstawanie 571
 - zapobieganie (unikanie) 159, 297
- przeciążenie 107, 159, 296
- przeciążenie całościowe 266
 - globalne 235, 253
 - lokalne 234, 253
 - konsekwencje 296
 - przejściowe 304
 - selektywne 266
 - tras 253, 265
 - trwałe 304
 - zasięg 293
- przeciwdziałanie przeciążeniom 106, 232, 268, 287, 292, 299, 312, 568
 - – fazy 297

- przeciwdziałanie przeciążeniom:
- sposoby 297
 - sposoby pozyskiwania informacji 297
 - w TCP/IP 301, 558
 - przepełnieniom 268
 - natłokom 114
 - zakleszczeniom 160, 179, 232
- przedział (czasu) alokacji 593
- oczekiwania 593
 - rozwiązywania kolizji 589
- przepełnienie 268
- przepływ adaptacyjny 554
- bez zaległości 414
 - dopuszczalny 130
 - formowanie 418
 - izolowany 573
 - maksymalno-minimalny 130
 - nadzorowanie 418
 - nieadaptacyjny 554
 - niewspółpracujący 299
 - pojedynczy 298
 - separowany 558
 - wieloprotokołowy 397
 - wielokładnikowy 586
 - zachłanny 299
 - zagregowany 298, 497
 - z zaległościami 414
- przepustowość 70, 82, 144
- przesunięcie opóźnienia 52
- przewidywalność 550
- przewymiarowanie sieci 390, 560
- przydzielanie etykiet na żądanie 601
- – „niezamówione” 601
- Queue Management for Congestion Recovery (QMCR) 298
- Quality of Services (QoS) 20, 381
- Random Drop (RD) 303
- Random Early Detection (RED) 304
- Random Early Detection with In and Out (RIO) 515
- Random Exponential Marking (REM) 309
- ramka interkodowana (MPEG typu I) 83
- interpolowana (MPEG typu B) 84
 - z predykcją ruchu (MPEG typu P) 83
- real time – Variable Bit Rate (rt-VBR) 290, 483
- Reassembly Buffer Allocation (RBA) 238
- Rectangular Search (RS) 543
- Reference Model (RM OSI) 59
- reglamentacja zapotrzebowania na zasoby 351
- zasobów 351, 371
- reguła klasyfikacji 529
- Relative Differentiated Services (RDS) 299
- Resource Reservation Protocol (RSVP) 423
- modele (style) rezerwacji 426
 - – Fixed Filter (FF) 430
 - – Shared Explicit (SE) 432
 - – Wildcard Filter (WF) 428
 - właściwości 423
 - zasady działania 424
- restrykcyjność systemu obsługi 354
- rezerwacja zasobów 406, 411, 422, 521
- dla najgorszego przypadku 411
 - pojemności 516
- router brzegowy 387, 520
- dostępowy 388, 516
 - szkieletowy 388
 - trasy 516
 - architektura 525
- Round Robin (RR) 150, 444, 594
- Routing Information Protocol (RIP) 576
- routing zorientowany na klasy ruchu 498
- – magistrale 583
 - – pary źródło–ujście 396
 - – przepływy 385, 583
 - – ujścia przepływów 571, 583
 - – źródła przepływów 583
- rozdział przepływów 577
- strumieni danych 63
 - maksymalizujący przepustowość 126
 - niesprawiedliwy 170
 - ruchu 588
 - sprawiedliwy 128, 445, 452
 - zasobów jednakowy 127
- rozmiar okna 307
- minimalny 147
 - wyznaczanie 137
 - zmiany 147, 307
- rozwiązywanie kolizji 590
- równoważenie obciążeń 396, 575
- różnicowanie cen 550
- jakości usług bezwzględne 390
 - – – proporcjonalne 551
 - – – względne 299, 394
 - obsługi 498
 - pojemności 550
 - ruchu 523

- różnicowanie usług sieci bez gwarancji jakości 515
- ruch:
 - agregacja 498
 - dopuszczanie 388
 - elastyczny 293
 - formowanie 418
 - grupowanie 498
 - identyfikacja 410, 434, 504
 - kontrola 388
 - klasyfikacja 189, 498, 504
 - kondycjonowanie 499, 504, 507
 - kształtowanie 414
 - nadzorowanie 418
 - niezgodny z profilem 508, 515
 - profil 502
 - strumieniowy 549
 - tolerujący straty 328
 - utrzymywanie 348
 - zgodny z profilem 508, 515
 - wrażliwy na opóźnienia 35
 - na obsługę w czasie rzeczywistym 37
 - na straty 35, 328
 - wygładzanie 318, 348, 517
 - zgodność 456
- segmentacja 72, 577
- Self-Clocked Fair Queueing (SCFQ) 485
- separacja przepływów 441
- Service Access Point (SAP) 66
- Service Data Unit (SDU) 68
- Service Level Agreement (SLA) 502
- Severely Errored Cell Block Ratio (SECBR) 290
- Shared Parallel Processing – Space Switching (SPPSS) 527
- Sharing with Minimum Allocation (SMA) 196
- Sharing with Maximum Queues (SMQ) 197
- Sharing with Minimum Allocation and Maximum Queue (SMAMQ) 198
- sieć datagramowa 206
 - sieć kolejek 225, 228
 - połączeń wirtualnych 206
 - silnie sterowana 259
 - słabo sterowana 258
 - z komutacją łączy 36
 - z komutacją pakietów 36, 206, 381
 - wielousługowa 381
 - wirtualna pełna 398
- sieć kolejek 225, 228
- Single Processor – Shared Bus (SPSB) 525
- skalowalność 389
- spedycja klas ruchu – sposoby 499
- specyfikacja przepływu 406
- spowolniony start 294
- sprawiedliwa alokacja współdzielonych zasobów 160
- sprawiedliwa alokacja zasobów 444
- sprawiedliwe kolejkowania 447, 556
 - szeregowania 454
 - sprawiedliwy rozdział zasobów 126, 147
 - dostęp do zasobów 573
- sprawiedliwy rozdział zasobów:
 - jednakowy 127
 - przez mechanizm okna 148
- sprężenie miary przeciążenia 312
- sterowalność 550
- sterowanie dostępem 406, 408
 - parametryczne 409
 - na podstawie pomiarów 409
 - opóźnieniem 412
 - prewencyjne 287, 312
 - przeciążeniem 297
 - reakcyjne 287
 - przepływem 113, 395, 405
 - bezpośrednio 394
 - danych 272
 - dwupoziomowe 151, 153, 210
 - indywidualne 216, 238
 - jednorazowe 210
 - jednopoziomowe 151, 153
 - krokowe 138, 210
 - maksymalno-minimalne 129
 - na poziomie:
 - dostępu do sieci 190, 253
 - trasy 151, 190, 216, 219, 225, 252
 - łącza danych 190, 191, 216
 - połączenia wirtualnego 151, 210, 216
 - transportowym 189, 268
 - na trasie 151
 - pośrednie 394
 - scentralizowane 113
 - wielokrokowe
 - wybiórcze
 - w procesie komunikacyjnym 138
 - zdecentralizowane 113
 - przyjmowaniem wywołań 291, 314, 349, 406
 - zorientowane na połączenia 385
 - przepływy 385
- sterownik przepływu 412

- sterownik zasobów 518
- stopa błędów 39
 - strat komórek 40
- Structured Buffer Pool (SBP) 200
- strumień ruchu zagregowany 497
- suma kontrolna 439
- suma miar przeciążenia 311
- superpozycja strumieni 315
- Sustainable Bit Rate (SBR) 290
- Sustainable Cell Rate (SCR) 290
- sygnalizacja przeciążenia:
 - w przód 293
 - wstecz 293
- sygnał przeciążenia 310
- system z opóźnieniami 383
 - ze stratami 383
 - obsługi potokowy 450
 - – ciągły 451
 - – dyskretny 451
- Systems Network Architecture (SNA) 138
- szeregowanie 296, 410, 441
 - dyskretnie 557
 - idealne 444, 589
 - jednostek danych 407
 - niezachowujące pracy 298, 446
 - przepływów 298
 - sprawiedliwe 447
 - – warunkowe 449
 - – ważone 450
 - – z gwarantowanym czasem obsługi 453
 - wymagania 441
 - zachowujące pracę 298, 446
 - zasady 446
- szybkość bitowa dostępna (ABR) 290
 - – niezidentyfikowana (UBR) 290, 483
 - – stała (CBR) 290
 - – zmienna (VBR) 290, 335
 - napływu znaczników 44, 351
 - strat 318
 - szczytowa 340, 408
 - średnia 340, 408
- ścieżka wirtualna 357
- tablica przekazywania 577
 - – etykiet 598, 601
 - tras (routingu) 248, 582
- Traffic Conditioning Agreement (TCA) 502
- Transmission Control Protocol/Internet Protocol (TCP/IP) 293
- trasa bezpośrednia 244, 396
 - danych 598
 - długość 479
 - dopuszczalna 585
 - – najszybsza 585
 - – optymalna 585
 - komutacji etykiet 597
 - najkrótsza 575, 584
 - najszersza 575, 584
 - najszybsza 584
 - optymalna 109
 - 244
- Triggered Jumping Window (TJW) 372
- tryb rozgłoszeniowy transmisji 66, 577
 - – asynchroniczny 61
 - – izochroniczny 61
 - – synchroniczny 61
 - – synchroniczny przerywany 61
- tryb bezpołączeniowy 63, 208
 - połączeniowy 63, 208
- upadek systemu 174
- udostępnianie zasobów 314, 389
- unikanie zakleszczeń 160
- Unspecified Bit Rate (UBR) 483
- usługi bez gwarancji jakości 116, 381, 383, 499
 - sieci 499
 - zintegrowane 382, 385, 404
 - zróżnicowane 382
- uzupełnianie jednostek danych 81
- Variable Bit Rate (VBR) 290, 335
- Virtual Leased Line (VLL) 515
- wariancja opóźnienia 48, 376
- wartość progowa odrzucania 304
- warunek zachowania ciągłości przepływu 151
- warunki aktywności procedury sterowania przepływem 139
- warunki konieczności sterowania przepływem 125
- warunki optymalności konieczne 110, 121
 - – wystarczające 110, 121
 - powstawania przeciążeń 159
 - profilu 512
 - zakleszczeń 159
- wąskie gardło 130
- Weighted Fair Queueing (WFQ) 450, 471
- Weighted Round Robin (WRR) 486
- węzeł brzegowy 499

- węzeł wewnętrzny (szkieletowy) 499
- wiadomość wewnętrzna 220
 - zewnętrzna 220
- wierność odtwarzania 54
- właściwości mechanizmu okna 141
- worst-case fair – Weighted Fair Queueing (W2FQ) 481
- współczynnik aktywności źródła 318
 - elastyczności 373
 - kolizji 438
 - maksymalnego wykorzystania łącza 108
 - obciążenia łącza 224
 - obciążenia stanowiska obsługi 462
 - sprawiedliwej alokacji zasobów 455
 - strat 39
 - wtrącania jednostek danych 40-41
 - wykorzystania stanowiska obsługi 462
 - zawłaszczania szybkości transmisji 586
 - kanału transmisyjnego 167, 168
 - wyprzedzania 100
- współdzielenie czasu procesora 445
 - kanału transmisyjnego 166
 - pamięci buforowej 175
 - pojemności 315
 - zasobów 126, 302, 441
- współzawodnictwo (o zasoby) bez ograniczeń 589
- wygładzanie ruchu 85-86,
- wykorzystanie zasobów 569, 573
- wymiarowanie sieci 560
- wypychanie 560
- wyznaczanie rozmiaru okna 137
 - węzła trasy 504, 548
- wyznaczanie tras 395, 405
 - – optymalnych 116, 396, 406
 - – pomiędzy źródłem a ujściem 116
 - – QoS 395, 584
 - – z ograniczeniami 396, 572, 577
- zadanie identyfikacji jednostek danych 434
 - inżynierii ruchu 396, 572, 581
 - łącznego wyznaczania tras i sterowania przepływem 116
 - klasyfikacji ruchu 504, 533
 - sprawiedliwego szeregowania 458
 - sterowania przepływem 113
 - wyznaczania tras 108
 - – z ograniczeniami 580
- zadanie wyznaczania przepływów 131
- zakleszczenie bezpośrednie (proste) 174, 254
 - kompletacji 182
 - – w ujściu 182, 254
 - odtwarzania (złożenia) jednostki danych 181, 233
 - odtwarzania kolejności 209, 233
 - pośrednie 174
 - sekwencji 184
- zalewanie 576
- zarządzanie aktywne kolejkami 303
 - kolejkami 298
 - kontraktami usługowymi 560
 - pamięcią buforową 197, 296
 - pojemnością łączy wyjściowych 553
 - hierarchiczne pojemnością 555
 - przepływami 396
 - retransmisjami 278
 - ruchem 391, 560, 572
 - usługami 560
 - zasobami w sieci 291, 553
- zawłaszczenie pamięci buforowej 175
- zgłoszenia końcowe 327
 - oznaczone 327
 - pojedyncze 327
 - pośrednie 327
- złożoność obliczeniowa algorytmu klasyfikacji 531
 - – – routingu 576, 582
- zmiennosc czasu transmisji 69
 - opóźnienia 47, 68, 74, 78, 84, 407
 - – dynamiczna 81
 - – statyczna 81
- znacznik 44, 351, 356
- znakowanie 305, 374
 - globalne 312
 - lokalne 312
 - przez ustawianie 305
- zwielokrotnianie deterministyczne 316
 - z podziałem czasu 135
 - statyczne 409
 - statystyczne 316, 409
 - strumieni danych 63
- zwłoka czasowa 294, 307, 488
- żądanie ustanowienia rezerwacji 406
 - obsługi rezerwacji 406
- źródło dwustanowe 346, 353
 - przepływu 412

Składam wyrazy podziękowania Recenzentom – Panu Profesorowi Zdzisławowi Papirowi z Katedry Telekomunikacji Akademii Górniczo-Hutniczej w Krakowie oraz Panu Profesorowi Józefowi Woźniakowi z Katedry Systemów Informacyjnych Politechniki Gdańskiej – za cenne uwagi, które pozwoliły na udoskonalenie tekstu niniejszej monografii

Adam Grzech