

Na prawach rękopisu

INSTYTUT CYBERNETYKI TECHNICZNEJ
POLITECHNIKI WROCŁAWSKIEJ

Raport serii PREPRINTY nr 4/91

ZASTOSOWANIE SIECI PETRIEGO
DO MODELOWANIA I OCENY EFEKTYWNOŚCI
FUNKCJONOWANIA ESP

(Praca doktorska)

Elżbieta Roszkowska

Promotor:

doc. dr hab. inż. Zbigniew Banaszak

Słowa kluczowe: sieci Petriego, elastyczny system produkcyjny,
modelowanie, złożoność obliczeniowa, sterowanie,
procesy współbieżne, system komputerowo wspoma-
ganego projektowania

Wydawnictwo Politechniki Wrocławskiej
WROCŁAW 1991

Promotorowi - Panu Docentowi
dr hab. inż. Zbigniewowi Banaszakowi,
Panu Profesorowi dr hab. inż.
Józefowi Grabowskiemu, który wpro-
wadził mnie w problematykę dyskret-
nych procesów produkcyjnych, oraz
Koleżankom i Kolegom z Zakładu
Podstaw Cybernetyki i Robotyki
serdecznie dziękuję za inspirację
oraz cenne uwagi w trakcie przygo-
wania tej pracy.

SPIS TREŚCI

	str.
1. WPROWADZENIE	1
1.1. Elastyczna automatyzacja produkcji	1
1.2. Problemy projektowania i eksploatacji ESP	7
1.2.1. Projektowanie struktury ESP	7
1.2.2. Planowanie produkcji w ESP	11
1.2.3. Sterowanie przebiegiem procesów produkcyjnych w ESP	13
1.3. Modelowanie ESP	14
1.3.1. Rola modeli	14
1.3.2. Modele teorii kolejek	16
1.3.3. Dyskretne modele optymalizacyjne	18
1.3.4. Komputerowe modele symulacyjne	20
1.3.5. Modele sieci Petriego	21
2. SFORMUŁOWANIE CELU PRACY	22
2.1. Zintegrowany model ESP	22
2.2. Konflikty zasobowe	27
2.3. Sformułowanie problemu	36
2.4. Układ pracy	42
3. SYSTEMY SIECI PETRIEGO	45
3.1. Sieci	45
3.2. Elementarne systemy sieciowe	47
3.3. Systemy typu miejsce/przejsięcie	49
3.3.1. Podstawowe określenia	49
3.3.2. Ograniczenia i rozszerzenia systemów P/T	51
3.4. Sieci wyższego rzędu	54
3.4.1. Sieci predykatowe	55
3.4.2. Sieci kolorowane	57

III

3.5. Czasowe systemy sieciowe	59
3.6. Własności systemów sieciowych	59
3.7. Zastosowania	63
4. MODELOWANIE I SYNCHRONIZACJA PROCESÓW PRODUKCYJNYCH W ESP	64
4.1. Systemy współbieżnych złożonych procesów potoko- wych	64
4.2. Sformułowanie problemu	65
4.3. Model dynamiki osiągalnej	72
4.4. Problemy realizowalności i powrotu	76
4.5. Model dynamiki dopuszczalnej	93
4.6. Złożoność obliczeniowa syntezy i analizy bieżącej.	117
4.7. Model etapowy	118
5. MODEL SYMULACYJNY PROCESÓW PRODUKCYJNYCH W ESP	125
5.1. Algorytm syntezy modelu	125
5.2. Model czasowy	133
5.3. Sterowanie działaniem systemu	135
6. KOMPUTEROWY SYSTEM MODELOWANIA I OCENY EFEKTYWNOŚCI FUNKCJONOWANIA ESP	139
6.1. Charakterystyka użytkowa	140
6.2. Struktura i działanie	146
6.2.1. Blok komunikacji z użytkownikiem	147
6.2.2. Bloki edycji danych	147
6.2.3. Blok syntezy modelu	149
6.2.4. Blok symulacji	150
6.2.5. Blok statystyk	152
6.3. Przykłady zastosowań	153
7. UWAGI KOŃCOWE	170
LITERATURA	177

SPIS WAŻNIEJSZYCH OZNACZEŃ

Symbol	Znaczenie	Wprowadzono w definicjach
1	2	3
PN	system sieci Petriego typu P/T (miejsce/przejście)	3.6
M	markowanie sieci P/T	3.7
$[M_0\rangle$	zbiór markowań osiągalnych z markowania początkowego	3.7
σ	sekwencja paleń sieci P/T	3.7
PH	system typu P/T z łukami wzbraniającymi	3.10
S_π	specyfikacja procesu π	4.1
S	specyfikacja systemu procesów	4.2
H	graf przepływu procesów (graf detalooperacji)	4.2
α	relacja przepływu	4.2
α^*	tranzytywne domknięcie α	4.2
T	zbiór operacji, zbiór wierzchołków sieci reprezentujących operacje	4.2, 4.3
E	zbiór przedmiotów pracy (typów detali), zbiór wierzchołków sieci reprezentujących przedmioty	4.2, 4.3
R	zbiór realizatorów	4.2, 4.3
G	zbiór magazynów	4.2, 4.3
β, γ	funkcje wymagań zasobowych procesów	4.2
C	funkcja pojemności magazynów	4.3
MO_{SC}	model dynamiki osiągalnej systemu procesów o parametrach S, C	4.3
T^v	zbiór operacji węzłowych	4.8
T^r	zbiór operacji prostych	4.8
G^v	zbiór magazynów węzłowych	4.9

1	2	3
G^r	zbiór magazynów przejściowych	4.9
G^K	zbiór magazynów konfliktowych	4.10
G^N	zbiór magazynów niekonfliktowych	4.10
ε	funkcja minimalnej pojemności magazynów	4.11
Z_t	lokalne sąsiedztwo operacji t (zbiór stref wzbronień przejścia t reprezentującego daną operację)	4.12
MD_{SC}	model dynamiki dopuszczalnej systemu procesów o parametrach S, C	4.13
ω	funkcja istotnego stanu magazynów	4.14
Ω	funkcja następnego stanu istotnego magazynów	4.14
PE	system etapowy zbudowany na systemie typu P/T	4.17
Ms	stan systemu PE	4.18
$[Ms_0\rangle$	zbiór stanów osiągalnych w PE ze stanu początkowego	4.18
$\tilde{\sigma}$	sekwencja zdarzeń w PE	4.18
$\Sigma(Ms)$	zbiór wszystkich sekwencji zdarzeń prowadzących w systemie PE od Ms_0 do Ms	4.18
$\Sigma(PE)$	zbiór wszystkich sekwencji zdarzeń w systemie PE	4.18
ME_{SC}	model etapowy zbudowany na systemie MD_{SC}	5.1
$ME_{SC\tau}$	model czasowy systemu procesów o parametrach S, C, τ	5.1
σ_u	sekwencja czasowa w systemie ME_{SC}	5.2
$\Sigma(ME_{SC\tau})$	zbiór wszystkich sekwencji czasowych systemu $ME_{SC\tau}$	5.2

1. WPROWADZENIE

1.1. Elastyczna automatyzacja produkcji

Zmiany w sytuacji rynkowej krajów wysokouprzemysłowionych w pierwszej połowie lat siedemdziesiątych wynikające ze zwiększonych kosztów wytwarzania wywołanych kryzysem energetycznym oraz zalew wyrobami z krajów o taniej sile roboczej (zwłaszcza Dalekiego Wschodu) spowodowały znaczne zaostrzenie konkurencji i wzrost wymagań w stosunku do producentów. Dla utrzymania się na rynku koniecznym stało się zwiększenie atrakcyjności produktów poprzez szybkie wdrażanie coraz to nowych, nowoczesnych wyrobów, natychmiastowe reagowanie na pojawiające się zapotrzebowania odbiorców, dostosowanie się do wymagań rynku, tj. produkowanie szerokiego asortymentu wyrobów w partiach o niewielkiej liczności, a przy tym wszystkim zapewnienie wysokiej jakości oferowanych produktów i zachowanie konkurencyjności ceny [153, 163].

Spełnienie powyższych założeń wymagało pojawienia się nowej jakościowo koncepcji systemów produkcyjnych - systemów zautomatyzowanych, osiągających wydajność i poziom wykorzystania urządzeń zbliżony do uzyskiwanych w produkcji masowej, a jednocześnie zachowujących elastyczność wytwarzania właściwą klasycznym warsztatom wytwórczym [147, 194].

Podstawowym czynnikiem umożliwiającym realizację tejże koncepcji stał się postęp w dziedzinie programowalnych skomputeryzowanych środków produkcji, pozwalający na zastąpienie automatyzacji sztywnej środkami automatyzacji elastycznej (programowanej). Obok rozwoju technik NC (sterowania numerycznego) i CNC (sterowania komputerowego) następowało wprowadzanie

licznych usprawnień i całkowicie nowych rozwiązań w konstrukcji urządzeń.

W pierwszym rzędzie zmiany te dotyczyły obrabiarek i polegały na [163]:

- rozszerzeniu zakresu zabiegów bądź rodzaju obróbki wykonywanych na jednej obrabiarce,
- poszerzeniu zakresu funkcji obrabiarek o czynności pomocnicze: manipulacyjne (mocowanie i odmocowanie detalu, wymiana narzędzi, głowic itp.), diagnostykę procesu, narzędzi i maszyn, kontrolę przedmiotu obrabianego i korekcję jego położenia, komunikację informacyjną z otoczeniem, proste funkcje diagnozowania awarii i reagowania na nie, usuwanie odpadów itp.,
- zapewnianiu równoległości zabiegów obróbkowych i pomocniczych,
- wzroście wydajności.

Inny ważny kierunek postępu w technologii dotyczył wprowadzenia robotów do automatycznego montażu, spawania, zgrzewania, malowania, kucia itp. Podstawowe znaczenie dla integracji pracy poszczególnych stanowisk roboczych miał rozwój urządzeń automatyzujących transport i magazynowanie przedmiotów. Opracowano konstrukcje sterowanych numerycznie manipulatorów i robotów, automatycznych wózków transportowych (tzw. AGV - od ang. Automated Guided Vehicles), przenośników, układarek, itp. Poza środkami przemieszczania rozwinięto także konstrukcje odpowiednich nośników do tworzenia jednostek ładunkowych (palet, pojemników), oraz urządzeń (najczęściej regałów wysokiego składowania) do składowania przedmiotów w formie jednostek transportowych, bądź luzem. Elastyczna automatyzacja produkcji, oparta o sterowanie numeryczne pracą urządzeń, objęła także in-

ne rodzaje pomocniczych operacji produkcyjnych. Opracowano m.in. takie urządzenia jak: automatyczne maszyny i roboty pomiarowe, oczyszczarki, myjnie, suszarnie, stanowiska do konserwacji, stabilizacji temperaturowej itp.

Drugi, obok konstrukcji urządzeń, istotny nurt rozwoju technicznego decydujący o pojawieniu się i rozwoju elastycznej produkcji dotyczył środków i metod informatyki umożliwiających kompleksowe sterowanie przebiegiem produkcji. Szczególnie szybki postęp zaznaczył się w dziedzinie urządzeń peryferyjnych (terminali graficznych), pamięci masowych, procesorów komunikacyjnych, w tym współpracujących z szerokim zestawem czujników i przetworników analogowo-cyfrowych. Dostępne stało się pozyskiwanie informacji drogą wizyjną i foniczną, coraz powszechniej dla przesyłania danych w sieciach komputerowych wykorzystuje się tanią i odporną na zakłócenia przemysłowe technikę światłowodową.

Rozwojowi sprzętu komputerowego towarzyszył także postęp w zakresie oprogramowania narzędziowego, a w szczególności także jego odmiany jak:

- systemy operacyjne umożliwiające realizację interakcyjnych procesów współbieżnych i wielodostępu na mikrokomputerach,
- rozproszone bazy danych i generatory baz danych,
- pakiety komunikacji i przesyłania danych w sieciach lokalnych,
- pakiety grafiki komputerowej,
- oprogramowanie do tworzenia programów symulacyjnych oraz systemów sztucznej inteligencji.

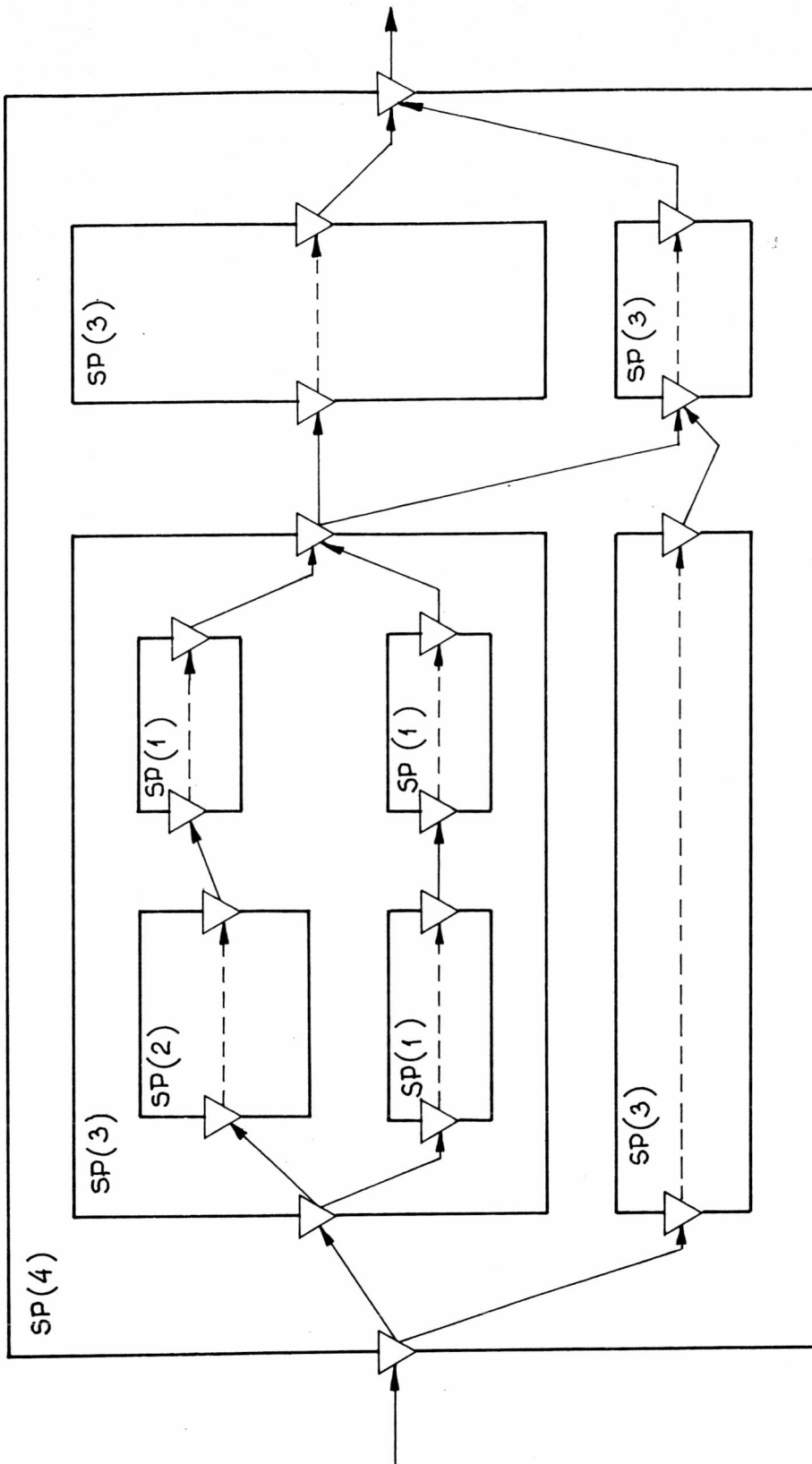
Wzmiankowane osiągnięcia w zakresie postępu technicznego umożliwiły powstanie elastycznych systemów produkcyjnych (ESP)

[9, 16, 22, 66, 67, 113, 117, 128, 147, 153, 163, 179, 182, 194, 223], tj. zintegrowanych komputerowo kompleksów programowalnych maszyn i urządzeń technologicznych, urządzeń przemieszczania materiałów i pomocy warsztatowych oraz urządzeń pomiarowych i diagnostycznych, z minimalną ręczną obsługą i krótkimi czasami przebiegów, mogące produkować dowolne wyroby, w ramach swojego przeznaczenia i w dowolnej kolejności, także współbieżnie.

W chwili obecnej większość obiektów tego typu to systemy, w których dominującą rolę odgrywa obróbka skrawaniem. Coraz powszechniej koncepcja elastycznej automatyzacji wkracza także w inne techniki wytwarzania. Dotyczy to zwłaszcza: obróbki blach, procesów odlewniczych, spawania, montażu.

W szerokim znaczeniu słowa ESP mieszczą się systemy produkcyjne różniące się zakresem automatyzacji, stopniem integracji funkcji i rozmieszczeniem elementów. Zgodnie z podziałem przedstawionym w pracy [163] i podobnie w [22, 113], wyróżnia się cztery typy elastycznych systemów produkcyjnych, granice których wyznacza położenie magazynów wejściowo-wyjściowych (Rys. 1.1).

- Elastyczny moduł produkcyjny (SP(1) - system produkcyjny pierwszego stopnia) stanowi jednostkę wyposażenia technologicznego, wyposażoną w układ sterowania numerycznego, pracującą autonomicznie i przystosowaną do wbudowania w elastyczne systemy produkcyjne wyższego stopnia. Jest nim zwykle ASO (autonomiczna stacja obróbkowa), centrum frezarsko-wytaczarskie, zrobotyzowana stacja montażowa itp., wyposażone w magazyn/-y detali oraz automatyczne urządzenia zmiany przedmiotów i narzędzi.



Rys. 1.1. Hierarchiczna budowa elastycznych systemów produkcyjnych [163]

- Elastyczne gniazdo produkcyjne (SP(2) - system produkcyjny drugiego stopnia) składa się zazwyczaj z kilku stanowisk roboczych przeznaczonych dla samodzielnego wykonywania zbioru operacji zależnych od wytwarzanego asortymentu przedmiotów, i wyposażony jest w niezależne urządzenie do automatycznej wymiany przedmiotów, zwykle typu robot przemysłowy, obsługujące wszystkie stanowiska. Funkcję nadrzędnego sterowania gniazdem pełni zazwyczaj sterownik robota, niekiedy także układ sterowania jednej z obrabiarek lub samodzielny mikrokomputer.

- Elastyczny oddział produkcyjny (SP(3) - elastyczny system produkcyjny trzeciego stopnia) obejmuje szereg elastycznych modułów i/lub gniazd produkcyjnych oraz urządzeń pomocniczych realizujących operacje kontroli, mycia, smarowania, usuwania wiórów itp., połączonych ze sobą zautomatyzowanymi elastycznie urządzeniami transportu i magazynowania oraz sterowanych komputerem nadrzędnym. Elastyczne moduły, gniazda i oddziały produkcyjne mogą być tworzone dla różnych technik wytwarzania: obróbki skrawaniem, obróbki plastycznej, montażu i in. Z punktu widzenia ich struktury wyróżnia się dwa rodzaje elastycznych oddziałów produkcyjnych:

- elastyczne linie produkcyjne, w których przepływ materiałów odbywa się zgodnie z taktem linii i w kolejności rozmieszczenia stanowisk, a ich elastyczność dotyczy łatwości przebrojenia przy przejściu na produkcję nowych wyrobów;
- elastyczne systemy produkcyjne o strukturze gniazdowej składające się z szeregu stanowisk roboczych działających w sposób asynchroniczny i sprzężonych asynchronicznym transportem, umożliwiającym przekazywanie ładunków między różnymi stanowiskami i w różnych kierunkach; brak synchronizacji czasów obróbki

a także obróbka wielu wyrobów o różnych marszrutach technologicznych powodują, że w systemach tych istotną rolę dla zapewnienia możliwie ciągłej pracy stanowisk spełniają magazyny detali - bufory przystanowiskowe i/lub magazyn centralny.

- Elastyczny wydział produkcyjny (SP(4) - elastyczny system produkcyjny czwartego stopnia) tworzy zbiór elastycznych systemów produkcyjnych niższego stopnia umożliwiając pełną produkcję określonego asortymentu wyrobów obejmującą różne fazy procesu produkcyjnego (np. produkcja przygotówek, obróbka detali, montaż, wykończanie, pakowanie) wg różnych technik i metod wytwarzania.

Elastyczna automatyzacja urządzeń produkcyjnych i komputerowa integracja ich działania umożliwiając niską pracochłonność produkcji, a jednocześnie jej dostosowanie do potrzeb rynku, tworzą potencjał wysokiej efektywności ekonomicznej ESP, którego wyzyskanie jest jednak ściśle uzależnione od przyjętych rozwiązań projektowych i sposobu eksploatacji systemu.

1.2. Problemy projektowania i eksploatacji ESP

1.2.1. Projektowanie struktury ESP

Ze stosowaniem ESP wiążą się m.in. następujące problemy projektowe:

- wybór grupy wyrobów do produkcji w ESP [72, 109, 110, 111],
- wybór podsystemu wytwarzania (wyposażenia technologicznego) [35, 38, 48, 71, 147, 164, 202, 214],
- wybór podsystemu transportowo-magazynowego [12, 36, 77, 86, 98, 114, 171, 220],
- dobór uchwytów i palet [122, 136, 160],

- wybór odpowiedniego systemu komputerowego (sprzętu i oprogramowania) [45,79,196],
- rozmieszczenie i integracja powyższych podsystemów [2, 3,49,50,78,118,135].

Jakkolwiek przeznaczeniem ESP jest wytwarzanie szerokiej gamy wyrobów, różnorodność ta ograniczona jest zawsze pewnym podobieństwem ich konstrukcji oraz technologii wytwarzania. W odróżnieniu od systemów produkcji masowej, gdzie punktem wyjścia do ich projektowania jest precyzyjna lista pożądanych wyrobów, klasa produktów budowanego ESP może być zdefiniowana jedynie poprzez określenie cech podobieństwa jej elementów i ewentualne wyróżnienie pewnych jej reprezentantów. Wytwarzanie wyrobów podobnych konstrukcyjnie i technologicznie umożliwia typizację stanowisk, urządzeń transportowych, magazynowych, manipulacyjnych, pozwala zmniejszyć różnorodność narzędzi, uchwyty, palet, prowadząc w konsekwencji do zwiększenia ograniczenia kosztów wdrożenia i eksploatacji ESP.

Drogą do osiągnięcia wysokiej efektywności systemów elastycznych jest w dużej mierze uproszczenie procesów produkcyjnych, w szczególności poprzez koncentrację prac wykonywanych na jednym stanowisku. Uproszczenia te umożliwiając ograniczenie liczby operacji transportowych, manipulacyjnych, magazynowych, a także w zdecydowany sposób redukując powiązania kooperacyjne stanowisk i związane z tym procesy informacyjno-decyzyjne, wymagają jednak opracowania nowych procesów technologicznych specjalnie dla ESP.

Dobór wyposażenia technologicznego dla projektowania systemu wymaga skojarzenia zbioru przewidywanych do produkcji wyrobów (opisanych poprzez ich charakterystyki konstrukcyjne

i wymagane procesy technologiczne) ze zbiorem podstawowych urządzeń technologicznych (np. obrabiarek) możliwych do zastosowania w ESP. Kojarzenie to dokonywane jest z punktu widzenia zgodności charakterystyk wyrobów z możliwościami technologicznymi urządzeń i prowadzi najczęściej do wytypowania ich wariantowych zestawów różniących się wydajnością, stopniem automatyzacji, ceną itp. Wybór zestawu urządzeń podstawowych decyduje w znacznym stopniu o doborze wyposażenia pomocniczego - rodzaju i liczbie stanowisk zamocowania i odmocowania detali, stanowisk mycia i kontroli, a także założeń dla doboru sprzętu transportowego - rodzaju (wózki, suwnice, układarki, przenośniki, obrotnice) i liczby środków transportowych, oraz urządzeń magazynowych - rodzaju (magazyn centralny, bufory przystanowiskowe), przeznaczenia (zapasy międzyoperacyjne, kompensacyjne, rezerwowe, pomoce warsztatowe, części zamienne), liczby i pojemności magazynów.

Czynnikiem decydującym w sposób istotny o efektywności ESP jest również projekt podsystemu sterowania, tj. środków komputerowych zaangażowanych w realizację wszelkich procesów przepływu informacji i podejmowania decyzji, jakie zachodzą w systemie. Punktem wyjściowym jest tu specyfikacja funkcji, jakie powinien pełnić podsystem sterowania oraz dobór na tej podstawie sprzętu komputerowego (składu i struktury - np. scentralizowanej/zdecentralizowanej, jedno/wielopoziomowej) i oprogramowania.

Ze względu na dominującą cechę omawianych systemów produkcyjnych jaką jest ich elastyczność, trudno jest do końca rozgraniczyć problemy projektowania od problemów eksploatacji. O ile można uznać, że architektura ESP jest ustalana przez jego

projektantów, to sposób sterowania systemem jest wynikiem decyzji podejmowanych zarówno w trakcie tworzenia systemu jak i jego użytkowania, przy czym granica w ich zakresach może być różna w różnych ESP. Stąd, obok pakietów podstawowych, wykorzystywanych w niezmienniej formie w długich okresach eksploatacji systemu (takich jak oprogramowanie transmisji danych między sterownikami urządzeń a komputerami sterującymi lub terminalami dyspozytorskimi, oprogramowanie zarządzające pozyskiwaniem informacji i bazą danych itp.) konieczne jest uwzględnienie programów narzędziowych służących generacji pakietów wymiennych lub skonstruowanych w sposób parametryczny i pozwalających przez to na elastyczną eksploatację ESP. Dotyczy to zwłaszcza oprogramowania związanego z podejmowaniem decyzji operacyjnych (np. sterowania przepływem materiałów), którego konstrukcja zarówno pod względem zakresu podejmowanych decyzji jak i sposobu ich wypracowywania powinna być dostosowana do specyfiki realizowanych zadań.

Projekt struktury przestrzennej ESP jest w dużej mierze uzależniony od wcześniej dokonywanych ustaleń projektowych - klasy procesów produkcyjnych, dla jakiej system ten jest konstruowany i doboru urządzeń technologicznych (określających przewidywane drogi przepływu przedmiotów pracy i pomocy warsztatowych) oraz planowanych rozwiązań fizycznej integracji stanowisk (poprzez podsystemy transportu, magazynowania i manipulacji). Czynniki te decydując o wymaganiach dotyczących sposobu powiązań elementów systemu rzutują tym samym na ogólny sposób ich rozmieszczenia. Zazwyczaj wyróżnia się tu trzy typy systemów [163]:

1. systemy z brakiem powiązań stanowisk produkcyjnych - przepływ materiałów realizowany jest w układzie magazyn zewnętrzny → bufor przystanowiskowy → stanowisko produkcyjne → bufor przystanowiskowy → magazyn zewnętrzny;

2. systemy o bezpośrednim powiązaniu stanowisk produkcyjnych - przepływ materiałów obejmuje transport międzystanowiskowy i magazynowanie w buforach przystanowiskowych, z wyróżnieniem struktur:

- a) liniowych - przepływ materiałów jednokierunkowy przez kolejne stanowiska i ewentualne bufora kompensacyjne,
- b) gniazdowych - przepływ materiałów wielokierunkowy obejmujący pomijanie stanowisk, nawroty;

3. systemy o pośrednich powiązaniach stanowisk - obejmujące centralny magazyn produkcyjny i ewentualnie bufora przystanowiskowe - przepływ materiałów jedno- lub wielokierunkowy, przekazywanie materiałów między stanowiskami najczęściej pośrednie, poprzez magazyn centralny, niekiedy także bezpośrednio pomiędzy buforami stanowisk.

1.2.2. Planowanie produkcji w ESP

Wykorzystanie ESP zgodnie z jego przeznaczeniem, tj. wytwarzanie różniących się wyrobów, także współbieżnie i w krótkich seriach, wymaga równie częstych zmian planu działania systemu. Ogólne wytyczne dla konstrukcji takiego planu wyznacza aktualny pakiet zamówień na wyroby systemu oraz pożądane terminy ich realizacji, określane w trakcie tworzenia ogólnozakładowego planu produkcji (w sytuacji kiedy ESP stanowi jedynie jeden z wydziałów produkcyjnych przedsiębiorstwa) lub w wyniku bezpośrednich negocjacji z klientem (w sytuacji kiedy

ESP działa jako jednostka samodzielna). Planowanie produkcji to ustalenie takiej organizacji jej przebiegu, aby przy zachowaniu ww. wytycznych osiągnąć jednocześnie wysoką efektywność działania ESP w kontekście takich miar ilościowych, jak wydajność systemu, stopień wykorzystania urządzeń itp.

Analiza teoretyczna jak i dotychczasowe doświadczenia wdrożeń ESP prowadzą do wyodrębnienia szeregu zagadnień występujących w toku planowania produkcji ESP, stanowiących kompleks powiązanych ze sobą problemów [4, 55, 96, 99, 112, 131, 140, 150, 157, 163, 169, 175, 109, 210, 215]. Są to problemy:

- wyboru asortymentu wyrobów przeznaczonych do produkcji współbieżnej [37, 72, 87, 109, 110, 111],
- planowania procesów technologicznych dla poszczególnych wyrobów [119, 129, 143, 170, 178, 201, 203, 213],
- grupowania stanowisk, polegający na wyodrębnieniu zespołów stanowisk zamiennych dla realizacji tego samego zbioru operacji [69, 144, 190],
- mieszanki, tj. ustalania proporcji wytwarzania poszczególnych wyrobów [163, 209],
- przydziału zasobów pomocniczych, sprowadzający się do rozdziału pomiędzy wyselekcjonowane rodzaje wyrobów pomocy warsztatowych (chwyteków, narzędzi, palet, zasobników), które są dostępne w ograniczonej liczbie [211],
- kolejności wprowadzania wyrobów do systemu [134],
- przydziału operacji i niezbędnych do ich wykonania pomocy warsztatowych stanowiskom lub grupom stanowisk zamiennych [186, 190],
- harmonogramowania [7, 17, 21, 24, 33, 41, 56, 58, 65, 101, 115, 139, 146, 165, 166, 176, 183, 184, 187, 188, 189, 191, 192, 193, 197, 216, 221].

Podstawowe trudności występujące w rozwiązywaniu powyższych problemów wynikają z ich złożoności i współzależności, wielorakości koniecznych do uwzględnienia ograniczeń, dużej liczby podejmowanych decyzji, często braku ścisłych kryteriów oceny rozwiązań itp.

1.2.3. Sterowanie przebiegiem procesów produkcyjnych w ESP

Mówiąc o problemach sterowania przebiegiem produkcji będziemy mieć na myśli tę grupę zagadnień, które dotyczą podejmowania decyzji o elementarnych działaniach ESP na bieżąco, bezpośrednio przed ich wykonaniem, w uwarunkowaniu od aktualnego stanu systemu i realizowanych zadań. O ile planowanie produkcji polega na ustaleniu wartości pewnego układu parametrów określających ogólne wytyczne dla pożądanego działania ESP, to celem sterowania jest zarządzanie i nadzór nad pracą poszczególnych urządzeń, tak aby wspólnym wynikiem ich działania była zgodna z ustaleniami i możliwie efektywna realizacja zadanego planu.

Pomijając problemy syntezy programów technologicznych, których rozwiązanie określa dość jednoznacznie konstrukcja sterowników robotów, stacji obróbkowych, itp. oraz ustalona technologia procesów, zasadnicze znaczenie dla działania ESP jako całości ma sterowanie nadrzędne, którego podstawowym celem jest koordynacja pracy wszystkich elementów systemu.

Współdziałanie stacji i urządzeń jednostkowych z nadzorującymi ich działaniem systemami komputerowymi zachodzi poprzez asynchroniczną, obustronną wymianę komunikatów precyzujących m.in. rodzaj operacji przeznaczonych do wykonania na danym urządzeniu i informacje zwrotne od urządzeń sygnalizujące wystą-

pienie określonych zdarzeń w ich działaniu. Zarządzanie w czasie rzeczywistym przez system sterowania funkcjonowaniem podlegających mu jednostek wymaga procedury decyzyjnej pozwalającej na wypracowywanie decyzji sterujących (zleceń dla urządzeń) w uzależnieniu od aktualnego stanu ESP (którego obraz kształtują wcześniejsze decyzje i występujące w systemie zdarzenia) i realizowanych zadań. Konstrukcja takiej procedury łączy się z dokonywaniem ustaleń dotyczących:

- zakresu zdarzeń podlegających kontroli systemu sterowania,
- zakresu decyzji podejmowanych przez system sterowania,
- sposobu reprezentacji stanu systemu i jego zmian,
- modelu podejmowania decyzji zapewniającego prawidłowy i efektywny przebieg realizowanych procesów, np. w warunkach występowania konfliktów zasobowych.

Problemy sterowania procesami produkcyjnymi w ESP są przedmiotem znacznej liczby prac, w tym [7, 13, 15, 19, 20, 45, 53, 54, 59, 100, 105, 106, 121, 123, 124, 151, 159, 161, 196, 200, 207, 214].

1.3. Modelowanie ESP

1.3.1. Rola modeli

Podstawową metodą upraszczającą rozwiązywanie wielu problemów pojawiających się w toku projektowania i eksploatacji elastycznych systemów produkcyjnych jest posługiwanie się odpowiednimi modelami. Trudnością samą w sobie jest dobór sposobu modelowania i konstrukcja właściwego modelu, który z jednej strony powinien ujmować te wszystkie cechy ESP, które stanowią o specyfice rozwiązywanego problemu, będąc z drugiej strony na

tyłe nieskomplikowaną reprezentacją rzeczywistości, aby możliwe było na jej podstawie opracowanie praktycznych metod prowadzących do rozwiązania rozważanego zagadnienia.

Problemy rozwiązywane w oparciu o konstruowane modele dotyczą trzech głównych grup zagadnień:

- predykcji wartości wyróżnionej zmiennej zależnej od parametrów modelu,
- identyfikacji związku funkcyjnego łączącego parametry modelu z pewną wyróżnioną zmienną,
- optymalnego doboru wybranej zmiennej ze względu na określone kryterium jakości pracy systemu.

Dotychczasowe badania związane z modelowaniem ESP koncentrują się wokół czterech typów modeli:

- modeli teorii kolejek [25-31, 57-60, 73, 83, 90, 116, 154, 173, 186, 211, 217-221],
- dyskretnych modeli optymalizacyjnych [12, 21, 49, 65, 69, 73, 87, 101, 109, 118-120, 144, 146, 172, 176, 184, 187, 188-190, 211, 216],
- komputerowych modeli symulacyjnych [3, 6, 8, 17, 34, 64, 127, 130, 137, 181, 212],
- modeli sieciowych konstruowanych w oparciu o aparat teorii sieci Petriego [1, 5, 10, 11, 14, 23, 32, 43, 46, 52, 64, 74, 85, 103, 105, 107, 125, 133, 157, 162, 177, 204, 205, 180].

Ze względu na różną podstawę grupowania (aparat opisu - dla modeli teorii kolejek i sieci Petriego, rodzaj rozwiązywanych problemów - dla modeli optymalizacyjnych, aparat opisu i metoda badań - dla modeli symulacyjnych) podział powyższy nie jest w pełni rozłączny. Przykładowo, problemy formułowane w oparciu o teorię kolejek lub sieci Petriego mogą w szczególności stanowić zagadnienie optymalizacyjne, a symulacja komputerowa

wa może być jedną z metod ich przybliżonego rozwiązywania. Tym nie mniej, istnieją pewne specyficzne cechy każdej z powyższych grup modeli świadczące o ich wadach lub zaletach w kontekście określonych zastosowań.

1.3.2. Modele teorii kolejek

ESP jest tu traktowany jako otwarty (o wyróżnionym rozkładzie prawdopodobieństwa napływu zadań do systemu) lub zamknięty (o stałej liczbie zadań w systemie) system kolejek zadań obsługiwanych przez skończoną liczbę stanowisk. Parametrami modelu są liczba i struktura obiektów obsługujących oraz charakterystyki procesów w nich realizowanych. Przykładowo, w jednym z modeli podstawowych (model Jacksona [90]) uwzględnia się takie zmienne jak:

- parametr u_i wykładniczego rozkładu prawdopodobieństwa czasu obsługi zadań na stanowisku i -tym,
- prawdopodobieństwo q_i rozpoczęcia realizacji zadań od ich obsługi na stanowisku i -tym,
- prawdopodobieństwo p_{ij} przejścia zadania obsługiwanego na stanowisku i -tym do kolejki zadań oczekujących na obsługę na stanowisku j -tym,
- parametr λ wykładniczego rozkładu prawdopodobieństwa długości interwału czasu między napływem kolejnych zadań do systemu.

Zakłada się tu ponadto, że długość kolejek nie jest ograniczona oraz, że zadania obsługiwane są zgodnie z regułą FIFO (w kolejności napływu do kolejki).

W innych modelach uwzględnia się również parametry charakteryzujące ESP pod względem liczby środków transportowych, po-

jemności buforów przystanowiskowych, liczby palet, uchwytów itp. Rozważa się także inne reguły kolejności (dyscypliny) obsługi zadań.

Przedmiotem analizy jest predykcja działania systemu w kontekście takich miar jak maksymalna przepustowość systemu, średni czas przepływu zadania, średnia liczba zadań w systemie i poszczególnych kolejkach, średni czas oczekiwania w kolejce itp.

W przypadku prostszych modeli, np. [90,180], możliwe jest ustalenie algebraicznych związków pomiędzy parametrami wejściowymi a wyznaczanymi. W innych przypadkach opracowuje się właściwe dla danego typu modelu algorytmy obliczeniowe (w zależności od złożoności modelu - ścisłe, np. [83,154], lub przybliżone, np. [31]). Ustalenie powyższych związków pozwala także, w oparciu o te same modele, na wyznaczanie algorytmicznych metod rozwiązywania problemów optymalizacyjnych, np. [186].

Probabilistyczny charakter modeli teorii kolejek i przyjmowane w nich uproszczenia (w szczególności założenie o istnieniu w systemie stanu ustalonego, przyjmowane jako podstawa wszystkich obliczeń) powodują, że otrzymane rozwiązania mają zawsze charakter przybliżony. Ze względu jednak na możliwość względnie łatwego i szybkiego otrzymania charakterystyk modelowanych obiektów, znajdują one zastosowanie w wyznaczaniu, ocenie i selekcji wstępnych rozwiązań wielu problemów projektowania i planowania produkcji w ESP.

1.3.3. Dyskretne modele optymalizacyjne

Modele te tworzone są jako zbiory formuł opisujących parametry rozważanego problemu i łączące je związki, ograniczenia jakie powinno spełniać rozwiązanie dopuszczalne problemu oraz

kryterium oceny rozwiązań. Przedmiotem poszukiwań jest rozwiązanie dopuszczalne ekstremalizujące wartość zadanego kryterium.

Formalnie wyróżnia się tu dwie klasy modeli, z których obie są reprezentatywne dla ESP. W pierwszym przypadku mamy do czynienia z problemami optymalizacji statycznej, w których kryterium optymalności stanowi funkcja n zmiennych modelu, a elementy zbioru rozwiązań dopuszczalnych należą do przestrzeni R^n . Przypadek drugi dotyczy problemów optymalizacji dynamicznej, w których kryterium optymalności jest w ogólności funkcjonałem, a rozwiązania dopuszczalne są funkcjami czasu.

Modele pierwszego typu spotyka się często np. w zagadnieniach rozmieszczania elementów ESP w oparciu o przyjętą macierz powiązań transportowych. Charakterystycznym przykładem może tu być zadanie dotyczące lokalizacji stanowisk wokół przenośnika o ruchu okrężnym formułowane jako [163]:
zminimalizować

$$f(G) = \sum_{p=1}^n \sum_{q=1}^n V_{g(p)q(p)} L_{pq}, \quad (1.1)$$

gdzie:

- $G = g(1), g(2), \dots, g(n)$ jest dowolną permutacją n kolejnych liczb naturalnych,
- $g(p), q(p)$ są numerami stanowisk stojących odpowiednio na miejscach p i q ,
- $V_{g(p)g(q)}$ jest elementem znanej macierzy $[V_{ij}]$ wyznaczającej powiązania transportowe między stanowiskami o numerach $i, j \in \overline{1, n}$,
- $L_{pq} = \begin{cases} q - p, & \text{dla } p \leq q \\ n + q - p, & \text{dla } p > q \end{cases}$ jest odległością między miejscami p i q .

Problemy optymalizacji dynamicznej ESP formułowane są w znacznej większości przypadków w oparciu o modele różnego rodzaju systemów obsługi typu: job shop, flow shop, permutation flow-shop, parallel shop i inne [120]. ESP jest tu reprezentowany przez zbiór urządzeń służących współbieżnej realizacji zbioru zadań, z których każde opisywane jest ciągiem operacji wykonywanych w określonym czasie przez określone urządzenie.

Zakłada się na ogół, że operacje są wykonywane w sposób nieprzerywany oraz, że dowolne urządzenie może być w każdej z chwil swojego działania zaangażowane w wykonywanie jednej tylko operacji. Przedmiotem poszukiwań są harmonogramy (uszeregowania) operacji optymalizujące takie kryteria jak średnie wykorzystanie urządzeń, średni czas cyklu, całkowity czas realizacji zadań itp.

Ze względu na wysoką złożoność formułowanych tu problemów nie jest na ogół możliwe skonstruowanie efektywnych algorytmów prowadzących do wyznaczania rozwiązań optymalnych; stąd praktyczne znaczenie posiadają głównie metody suboptymalne (np. metoda podziałów i ograniczeń z ograniczonym cofaniem, lokalne przeszukiwanie sąsiedztwa itp.) oraz metody heurystyczne.

Wiarygodność otrzymanych wyników ściśle zależy od możliwości predykcji wartości parametrów zadania lub zgodności zachowań systemu rzeczywistego z modelowanym. Przykładowo, wartości elementów macierzy powiązań transportowych we wspomnianym wyżej modelu optymalizacji rozmieszczenia stanowisk, mają zawsze jedynie charakter szacunkowy. Podobnie uszeregowania uznane za optymalne (suboptymalne) w oparciu o badania modelowe mogą ulegać znacznej dezaktualizacji na skutek odstępstw rzeczywistych czasów realizacji operacji od zakładanych lub występowania

nia zakłóceń, uszkodzeń urządzeń itp., których modele te nie przewidują.

1.3.4. Komputerowe modele symulacyjne

W odróżnieniu od wcześniej omówionych modeli tworzonych jako podstawa dla badań teoretycznych, modele symulacyjne konstruowane są z przeznaczeniem dla badań eksperymentalnych. W symulacji komputerowej modelem jest program napisany w określonym języku programowania - specjalizowanym jak SIMSCRIPT, SIMULA, SLAM, TESS, GPSS, lub ogólnego użytku jak C, PASCAL itp.

Główną korzyścią stosowania modeli symulacyjnych jest możliwość całościowej i szczegółowej reprezentacji ESP z ukierunkowaniem na konkretny istniejący bądź projektowany system. Eksperyment symulacyjny pozwala na wyznaczenie wybranych (badanych) charakterystyk przepływu materiałów, przez co umożliwia dobór struktury i reguł sterowania ESP pod kątem optymalizacji efektywności jego działania. Uwzględniane tu kryteria są analogiczne, jak w przypadku innych modeli - wydajność systemu, stopień wykorzystania urządzeń, czasy realizacji całości zadań i cyklu produkcyjnego, zapasy robót w toku itp.

Modele symulacyjne dają zatem realną możliwość uzyskania ilościowej oceny działań systemu, której osiągnięcie, ze względu na złożoność modelu, nie jest możliwe na drodze badań teoretycznych. Z drugiej strony, modelowanie bezpośrednio w językach programowania nie stwarza wystarczających podstaw dla syntezy modeli o określonych, pożądanych cechach jakościowych oraz powoduje, że proces konstrukcji modelu wymaga znacznych nakładów czasu i pracy, a jego użyteczność ogranicza się do konkretnego ESP i konkretnych zadań analizy, dla których model ten jest tworzony.

1.3.5. Modele sieci Petriego

Modele te opisują ESP jako dynamiczny system dyskretny złożony ze skończonego zbioru obiektów, których stan składa się na stan całości systemu oraz skończonego zbioru zdarzeń, których występowanie powoduje zmiany stanu. Możliwość wystąpienia określonego zdarzenia zależy od stanu systemu, a stan następny jest jednoznacznie zależny od stanu wcześniejszego i zdarzenia powodującego jego zmianę. Uzależnienia powyższe mają charakter lokalny, tj. zarówno możliwość wystąpienia zdarzenia jak i zmiana spowodowana jego wystąpieniem zależy od/ dotyczy stanu jedynie pewnego podzbioru obiektów systemu.

Jak dotąd główne znaczenie sieci Petriego w projektowaniu czy planowaniu produkcji ESP wiąże się raczej z potencjałem, jaki zawiera w sobie ten aparat, niż modelami, które zostały już opracowane. Modele te służą rozwiązywaniu pewnych wyróżnionych, szczególnych problemów, np. synchronizacji procesów o konkretnie ustalonej strukturze, co nie pozwala na ich wykorzystanie dla innych celów i w konsekwencji tworzy konieczność konstrukcji dalszych modeli. Spośród prac, w których zagadnienia syntezy i analizy sieciowych modeli ESP są postawione w sposób ogólny, dostarczając w ten sposób rozwiązań obowiązujących w szerokiej klasie struktur systemów i procesów produkcyjnych można wymienić [1, 13, 14, 105, 157, 158].

Podstawową zaletą sieci Petriego jest możliwość ujęcia immanentnego dla ESP zjawiska asynchronizmu działań elementów systemu. Formalizm ten ułatwia w znacznym stopniu zarówno analizę zachowań współbieżnych procesów produkcyjnych, jak i syntezę mechanizmów wymuszających poprawność ich przebiegu i współ-

przebiegu. Wiążąc zdarzenia systemowe z czasem, na co pozwalają czasowe sieci Petriego, możliwe jest także tworzenie modeli optymalizacyjnych stosujących kryteria ilościowe.

Oprócz wyżej wymienionych zalet znaczne korzyści dają także:

- prosta interpretacja graficzna i bezpośrednia interpretacja pragmatyczna wszystkich składowych modelu, ułatwiająca ich skojarzenie ze składowymi rozważanego obiektu rzeczywistego,
- możliwość hierarchicznej syntezy i analizy modeli ujmujących kolejne poziomy abstrakcji,
- możliwość reprezentacji zarówno zdarzeń zaplanowanych jak i zakłóceń losowych,
- możliwość zastosowania technik analizy należących do ogólnej teorii sieci Petriego.

2. SFORMUŁOWANIE CELU PRACY

2.1. Zintegrowany model ESP

W poprzednich podrozdziałach staraliśmy się dokonać syntetycznego przeglądu głównych problemów występujących w projektowaniu i eksploatacji ESP oraz modeli i metod wspomagających ich rozwiązywanie. Opracowanie powyższe, ze względu na obszerność tematu, z pewnością nie jest w pełni wyczerpujące, tym nie mniej daje podstawy do sformułowania pewnych wniosków ogólnych.

Budowa systemów produkcyjnych charakteryzujących się wysokim stopniem automatyzacji, elastyczności i integracji funkcjonalnej elementów łączy się ze znacznym, w stosunku do systemów

konwencjonalnych, zwiększeniem nakładów inwestycyjnych. W sytuacji takiej szczególnego znaczenia nabiera postulat dążenia do osiągnięcia możliwie wysokiej efektywności ESP, realizowany poprzez staranny dobór wyposażenia produkcyjnego i struktury systemu oraz właściwe planowanie procesów produkcyjnych i sterowanie ich przebiegiem.

Wymagania stawiane procesom podejmowania decyzji dotyczących tak projektu jak i sposobu wykorzystania ESP są zatem znacznie wyższe niż w tradycyjnych systemach produkcyjnych. Co więcej, w tych ostatnich bazą dla większości ustaleń jest model oparty o założenie, że realizowane w nich procesy produkcyjne mają charakter stacjonarny, tzn. system znajduje się w stanie ustalonym, po długim okresie pracy lub też może być traktowany statycznie, w oparciu o wielkości uśrednione, przy zaniechaniu wzajemnych oddziaływań podsystemów [163]. W odniesieniu do ESP założenie takie jest niedopuszczalne ze względu na wysoki poziom integracji elementów decydujący o ich współzależnym, a jednocześnie różnym, ze względu na częste zmiany zadań produkcyjnych, wpływie na działanie systemu.

Zapewnienie wysokiej efektywności ESP wymaga zatem kompleksowego i dynamicznego podejścia do rozwiązywania występujących tu problemów, tj. podejmowania decyzji projektowych, planistycznych i operacyjnych z uwzględnieniem ich wypadkowego wpływu na przebieg procesów produkcyjnych.

Biorąc pod uwagę wysoką złożoność ESP nie można oczywiście uniknąć konieczności rozwiązywania szeregu problemów w sposób zindywidualizowany. Postępowanie kompleksowe i dynamiczne w projektowaniu oznacza, że dobór wielu parametrów takich jak struktura systemu, liczba i wydajność urządzeń technolo-

gicznych i transportowych, pojemność magazynów itp. oraz ogólnych zasad funkcjonowania i kooperacji poszczególnych podsystemów, musi uwzględniać globalną ocenę postulowanych rozwiązań w oparciu o kryteria efektywności działania ESP jako całości. Oznacza to także, że ocena taka musi być dokonywana w kontekście pozostałych czynników posiadających wpływ na wartości tychże kryteriów - przewidywanych wariantów procesów produkcyjnych i mechanizmów sterowania. Analogicznie, synteza planów produkcji czy konstrukcja algorytmów sterowania powinna być dokonywana z uwzględnieniem możliwości elastycznego wyboru, w granicach jakie stwarza projekt ESP, a wiarygodna ocena dopuszczalności i efektywności rozwiązań musi uwzględniać wpływ wszystkich czynników rzutujących na dynamikę systemu.

Aczkolwiek istotność takiego podejścia jest często podkreślana w literaturze przedmiotu (np. [163,209]), to zauważa się jednocześnie niedostatek rozwiązań wspomagających jego zastosowanie w praktyce.

Możliwości takich nie dostarczają w pełni zarówno modele masowej obsługi jak i modele stosowane w badaniach operacyjnych. Aparat ich opisu nie odwołuje się do takich pojęć jak stan systemu i jego zmiany, nie pozwalając w ten sposób na uwzględnienie mechanizmów podejmowania decyzji w czasie rzeczywistym, ocenę niezawodności i elastyczności działania ESP. Ponadto, wypracowanie odpowiednich rozwiązań analitycznych (formuł algebraicznych, efektywnych algorytmów obliczeniowych) wymaga zawsze ograniczenia złożoności modelu. Wynikiem takiego podejścia jest oddzielna analiza problemów rozmieszczania, transportu, magazynowania, obróbki itp. Uzyskiwane w ten sposób rozwiązania lokalne często ulegają dezaktualizacji w warunkach rzeczy-

wistych ze względu na oddziaływanie na siebie wszystkich podsystemów ESP.

Ograniczona przydatność do badań kompleksowych cechuje także dotychczasowe konstrukcje programów symulacyjnych. Modele te stanowią często dość dokładną, całościową reprezentację istniejących lub projektowanych systemów produkcyjnych. Umożliwiają zatem wiarygodną analizę ich zachowań, ale jedynie w kontekście tej struktury ESP i tych procesów produkcyjnych, dla których tworzony był dany program. Sytuacja taka utrudnia znacznie analizę powiązań różnych struktur ESP z jego szeroko rozumianym działaniem i powoduje, że nie jest możliwe zastosowanie modelu dedykowanego jednemu systemowi dla analizy innych.

W odróżnieniu od modeli kolejkowych i optymalizacyjnych podstawą programów symulacyjnych jest zazwyczaj model sterowania. Brak jednak jak dotąd, wystarczających podstaw teoretycznych konstrukcji mechanizmów synchronizacji procesów współbieżnych występujących w ESP powoduje, że problemy związane z ich syntezą są rozwiązywane w sposób selektywny, tj. obowiązujący tylko dla danego modelu symulacyjnego. Konstrukcje te tworzone są bezpośrednio w językach programowania i na ogół bez dowodów poprawności rozwiązań.

Analiza symulacyjna pozwala często na eksperymentalne wyszukiwanie ich niedostatków, nie umożliwia natomiast, ze względu na zawsze ograniczony zakres prowadzonych eksperymentów, pełnej weryfikacji ich poprawności pod względem określonych kryteriów jakościowych. Badania symulacyjne nie dają zatem gwarancji, że obiekt rzeczywisty, zbudowany w oparciu o analizowany model wykaże poprawność w całej dziedzinie swoich zachowań. Zadania takie muszą być rozwiązywane w drodze analitycznej,

na etapie syntezy modelu i przy zastosowaniu bardziej przejrzystego i wygodnego aparatu opisu niż ten, jaki dają języki programowania.

W rozważanym kontekście szczególnie przydatnym wydaje się być formalizm sieci Petriego pozwalający na rozwiązywanie problemów sterowania w czasie rzeczywistym na wyższym poziomie abstrakcji, niż ten jaki dają typowe języki programowania współbieżnego. Podejście takie ułatwia wnioskowanie analityczne, a uzyskane wyniki mogą być w prosty sposób sprowadzone do poziomu niższego, tj. zaimplementowane w postaci programów komputerowych. Podobnie jednak jak w przypadku modeli symulacyjnych, aparat ten był w większości wykorzystywany dla reprezentacji wyróżnionych obiektów i analizy ich dynamiki. Uzyskane tu rezultaty dotyczą zatem przypadków szczególnych, a tym samym nie mogą być w ogólności zastosowane w projektowaniu struktury i funkcjonowania ESP.

Proponowany w niniejszej pracy sposób widzenia ESP polega na jego reprezentacji w postaci systemu współbieżnych procesów produkcyjnych. Bezpośredni wpływ zarówno projektu wyposażenia technologicznego, planu produkcji jak i mechanizmów sterowania na przebieg procesów pozwala rozważać wszystkie te zagadnienia w ramach wspólnego modelu. Oznacza to możliwość wykorzystania jednego aparatu pojęciowego, np. ogólnego modelu sieciowego, dla sformułowania zadań planowania struktury procesów technologicznych, planowania i harmonogramowania produkcji oraz sterowania operatywnego, które to zadania mogą być rozważane łącznie.

Dla wykonywania każdej z kolejnych składowych (operacji) procesów potrzebna jest jedynie część zasobów ESP. Dążąc do

zapewnienia wysokiego stopnia ich wykorzystania dopuszcza się zatem naprzemienne używanie tego samego zasobu przez wiele procesów. Sytuacja taka, w warunkach asynchronicznego przebiegu realizowanych zadań produkcyjnych, prowadzi do współzawodnictwa procesów o zasoby, tzn. do konfliktu zasobowego.

Rozwiązywanie konfliktów zasobowych w ESP wymaga metod, które powinny uwzględniać specyfikę procesów produkcyjnych, różniącą je od np. współbieżnych procesów komputerowych, oraz posiadać niską złożoność obliczeniową. Istotne jest przy tym, aby uzyskiwane rozwiązania miały charakter elastyczny, zachowujący wysoki potencjał możliwości adaptacyjnych wyposażenia technologicznego systemu. Możliwość automatycznego i elastycznego rozstrzygania konfliktów zasobowych pozwala z jednej strony na ocenę różnych dopuszczalnych wariantów przebiegu modelowanych procesów, z drugiej zaś na rozważanie konstrukcji procedur sterowania logicznego procesami współbieżnymi w czasie rzeczywistym. Stanowi zatem warunek zintegrowanego rozwiązywania problemów ESP.

2.2. Konflikty zasobowe

Występowanie konfliktów zasobowych wymaga mechanizmu koordynacji działania procesów, którego celem jest zapewnienie ich poprawnego i efektywnego (względem określonych kryteriów oceny, odpowiednio jakościowej i ilościowej) współbiegu. Zadania te realizowane są w drodze podejmowania decyzji rozstrzygających pierwszeństwo przydziału konkurującym o nie procesom.

Konieczność rozstrzygania konfliktów zasobowych w czasie rzeczywistym wymusza posługiwanie się dynamicznymi modelami

przebiegu procesów. W przestrzeniach stanów generowanych przez te modele można wyróżnić podzbiór stanów bezpiecznych, tzn. posiadających tę własność, że z każdego stanu należącego do tego podzbioru istnieje możliwość przejścia do każdego innego stanu z tego podzbioru. Należy zauważyć, że zachowanie modelowanego systemu (realizacje procesów w nim zachodzących) ograniczone do podprzestrzeni stanów bezpiecznych polega na takim rozstrzygnięciu konfliktów zasobowych, że nie prowadzi to do wzajemnej blokady procesów.

Na problem syntezy dynamicznych modeli procesów, których przestrzenie stanów zawierają tylko stany bezpieczne, można spojrzeć z dwóch stron. W pierwszym podejściu (tzw. podejście od strony analizy modelu) w kolejnych iteracjach dokonywane są modyfikacje pewnego modelu wyjściowego. Ich celem jest otrzymanie modelu końcowego, którego przestrzeń stanów obejmuje jedynie stany bezpieczne. W podejściu drugim (tzw. podejście od strony syntezy modelu) wykorzystywane są algorytmy przeprowadzające specyfikację procesów (dane charakteryzujące związki występujące między procesami i zasobami) w dynamiczny model przebiegu procesów należący do zadanej, wcześniej ustalonej klasy modeli. Budowa takich algorytmów oparta jest na znajomości warunków koniecznych "bezpiecznego" rozstrzygnięcia konfliktów zasobowych w rozważanej klasie modeli.

Należy zauważyć, że racjonalne postępowanie w podejściu pierwszym (ukierunkowujące proces modyfikacji modelu) jak i wyznaczanie algorytmów syntezy modeli wymaga uwzględnienia istniejących metod rozwiązywania konfliktów zasobowych. Jest rzeczą oczywistą, że efektywność tych metod jest uwarunkowana z jednej strony specyfiką rozważanych procesów, z drugiej zaś

koniecznością budowy algorytmów o niskiej złożoności obliczeniowej. Ze względu na NP-trudny charakter zadań syntezy modeli bezblokadowych dopuszcza się rozwiązania suboptymalne. Oznacza to, że przestrzenie stanów generowanych modeli mogą nie zawierać pewnych stanów bezpiecznych, które występują w modelach reprezentujących wszystkie możliwe (w tym także blokadowe) realizacje procesów.

Poszukiwanie metod rozwiązywania problemu blokad dotyczy głównie rozstrzygania konfliktów zasobowych w systemach komputerowych. Abstrakcyjny charakter pojęciowy tworzonych dla tego celu modeli pozwala jednak rozważać możliwość ich wykorzystania także w innych obiektach, w tym ESP.

Zagadnienia te analizowane są w oparciu o model dynamiczny - system procesów i zasobów, którego stan ulega zmianom następującym w wyniku zdarzeń żądania, otrzymania (przydziału) i zwalniania zasobów przez procesy [174]. W ogólności wyróżnia się tu dwa typy zasobów: zwrotne i niezwrotne.

Zasób zwrotny charakteryzuje się stałą liczbą jednostek, z których każda może być albo wolna (dostępna) albo przydzielona określonemu procesowi. Przydział jednostki może nastąpić tylko wtedy, gdy jest ona wolna, zwolnienie - tylko wtedy gdy jest przydzielona.

Zasób niezwrotny opisuje się liczbą aktualnie wolnych (dostępnych) jego jednostek, która może być nieograniczona (w niektórych modelach zakłada się jej ograniczoność, np. pojemnością obszaru, w którym jednostki te są gromadzone). Zwiększenie stanu zasobu niezwrotnego następuje poprzez zwalnianie (wytwarzanie) jego jednostek przez pewne procesy (tzw. procesy producentów); zmniejszanie - poprzez bezzwrotny przydział jednostek (zużycie) pewnym procesom (tzw. procesom

konsumentów). Zwalnianie, w przypadku ogólnym, nie podlega żadnym ograniczeniom (w modelach ograniczonych zasobów niez-
wrotnych może nastąpić tylko wtedy, jeżeli nie spowoduje to
przekroczenia maksymalnej dopuszczalnej liczby jednostek za-
sobu); liczba jednostek przydzielanych nie może być większa
od aktualnie dostępnych jednostek zasobu.

Procesy traktowane są jako obiekty dynamiczne, wymagają-
ce, w ogólności, różnych zasobów dla realizacji różnych eta-
pów swojego działania. Proces może zażądać zasobów, jeżeli
jest w stanie aktywnym (nie posiada niezrealizowanych żądań
zasobowych), po czym przechodzi w stan pasywny (oczekiwania)
aż do czasu ich przydziału.

Określa się, że proces jest wstrzymany w stanie S, jeże-
li oczekuje na zasoby, których przydział w S jest niemożliwy
oraz jest zablokowany w stanie S, jeżeli jest wstrzymany w S
i pozostaje wstrzymany w każdym ze stanów osiągalnych z S.

Stan S jest blokadą, jeżeli istnieje proces zablokowany
w S, oraz blokadą całkowitą, jeżeli wszystkie procesy są za-
blokowane w S.

Wyróżnia się trzy ogólne podejścia do rozwiązywania prob-
lemu blokad:

- wykrywanie i likwidację blokad [68,174],
- zapobieganie blokadom [80,82],
- unikanie blokad [18,42,74,80,92].

Podejście pierwsze zezwala na wystąpienie blokady, która
po jej wykryciu zostaje zlikwidowana poprzez usuwanie pewnych
procesów z systemu. Metoda taka, jako niepraktyczna w przypad-
ku procesów o charakterze strumieni materiałowych (z jakimi
mamy do czynienia w ESP), nie będzie dalej rozważana.

Pozostałe dwa podejścia polegają na zabezpieczeniu systemu przed wystąpieniem blokady poprzez zapewnienie, że nie będzie spełniony pewien warunek konieczny dla jej powstania.

W odróżnieniu od systemów procesów korzystających jedynie z zasobów zwrotnych, dla których możliwe jest opracowanie pewnych metod ogólnych, przeciwdziałanie blokadom wśród producentów i konsumentów wymaga znajomości konkretnej lub dopuszczalnej struktury ich współpracy. Istniejące tu rozwiązania były w dużej mierze konstruowane pod kątem bezblokowej wymiany komunikatów w określonych systemach komputerowych [80], a ich wykorzystanie w sterowaniu procesami produkcyjnymi może dotyczyć jedynie bardzo specyficznych przypadków.

Stan blokady w systemie procesów korzystających z zasobów zwrotnych może powstać tylko wtedy, gdy spełnione są jednocześnie cztery warunki [42,108]:

- wzajemnego wykluczania; konkretna jednostka zasobu nie może być jednocześnie przydzielona więcej niż jednemu procesowi,
- niewywłaszczalności; konkretną jednostkę zasobu może zwolnić tylko ten proces, który ją otrzymał,
- częściowego przydziału; proces nie może zwalniać zasobów w trakcie oczekiwania na przydział innych,
- wzajemnego czekania; może wystąpić stan, w którym istnieje zamknięty łańcuch procesów, z których każdy oczekuje na zasób przydzielony jego następnikowi w łańcuchu.

Ponieważ zaprzeczenie trzem pierwszym warunkom jest w przypadku procesów produkcyjnych trudne do realizacji lub wręcz niemożliwe, uwzględnimy dalej metody, których celem jest niedopuszczenie do spełnienia ostatniego z warunków.

Przeciwdziałanie występowaniu blokad poprzez konstrukcję metod gwarantujących brak "wzajemnego czekania" może mieć charakter zarówno zapobiegania jak i unikania. Różnica pomiędzy jednym a drugim podejściem polega na tym, że w pierwszym przypadku nakłada się pewne warunki na sposób żądania zasobów przez procesy (a zatem na strukturę ich programów), podczas gdy w drugim - na sposób przydziału zasobów procesom (co pozwala rozważać dowolną strukturę ich programów). W sytuacji pierwszej, procesy które nie spełniają nałożonych warunków są usuwane z systemu.

W praktyce stosuje się dwie metody zapobiegające wzajemnemu czekaniu procesów [18,82]. W prostszej z nich każdy proces musi w chwili rozpoczęcia swego działania wystąpić z żądaniem wszystkich potrzebnych mu jednostek zasobów na cały czas przebywania w systemie. W drugiej metodzie wprowadza się uporządkowanie rodzajów zasobów i wymaga się, aby zadanie mające przydzielone jednostki zasobu R_j występowało z żądaniami dotyczącymi jedynie zasobów R_k o indeksach $k < j$ ($j, k = 1, 2, \dots, n$).

Łatwo zauważyć, że podejście zapobiegania cechuje niski stopień wykorzystania zasobów systemu. Metody te wymuszają w ogólności przydział jednostek zasobów procesom wcześniej, niż są one procesom tym potrzebne. Sytuacja taka powoduje zatem nadmierną bezczynność zasobów. Zaletą rozważanego podejścia jest natomiast bardzo niski narzut czasowy, związany jedynie ze sprawdzeniem, czy żądania procesów spełniają nałożone na nie warunki.

Zabezpieczenie systemu przed wystąpieniem sytuacji wzajemnego czekania procesów realizują także metody wykorzystujące podejście unikania. Podejście to, jak wspomniano wcześniej,

polega na nałożeniu pewnych warunków na sposób (kolejność) przydziału zasobów aktualnie ubiegającym się o nie procesom. Zezwala się tu na realizację określonego żądania zasobowego tylko wtedy, jeżeli wynikająca z tego ewentualna zmiana stanu rozdziału zasobów prowadzi do stanu bezpiecznego. Stan rozdziału zasobów uważa się za bezpieczny, jeżeli zapewnia on, że dowolne przyszłe zamówienia zasobowe procesów będą mogły być zrealizowane bez wystąpienia blokady. Stwierdzenie tego faktu wymaga oprócz znajomości aktualnego stanu systemu, także pewnych dodatkowych informacji dotyczących wymagań zasobowych procesów.

W klasycznej dla podejścia unikania metodzie bankiera (algorytmie Bankiera [68,74,80]) zakłada się, że każdy z procesów jest scharakteryzowany przez swoje maksymalne potrzeby zasobowe oraz, że przydział zasobów w wysokości tychże potrzeb pozwala na ukończenie danego procesu i zwrot wszystkich przydzielonych mu wcześniej jednostek zasobów do puli systemu. Weryfikacja bezpieczeństwa stanu wiąże się z założeniem najgorszego przypadku, tj. sytuacji, w której wszystkie procesy żądają wszystkich, brakujących im do zaspokojenia maksymalnej potrzeby jednostek zasobów, nie uwalniając przedtem przydzielonych im wcześniej.

Można pokazać [74], że dla takiego przypadku stan rozdziału zasobów jest bezpieczny wtedy i tylko wtedy, gdy istnieje tzw. bezpieczna sekwencja procesów. Jest to taka sekwencja, w której każdy proces ma zapewniony przydział jednostek zasobów brakujących mu do zaspokojenia maksymalnej potrzeby przy wykorzystaniu wolnych jednostek zasobów i jednostek zasobów przydzielonych procesom poprzedzającym go w sekwencji.

Klasyczny algorytm bankiera sprawdza istnienie takiej sekwencji w co najwyżej $mn(n+1)/2$ krokach, aczkolwiek, możliwe jest także opracowanie algorytmu stwierdzającego jednoznacznie, czy przy założeniu najgorszego przypadku żądań procesów stan rozdziału zasobów zwrotnych jest bezpieczny, w czasie proporcjonalnym do mn [80].

Unikanie blokad pozwala uzyskać w ogólności wyższy stopień wykorzystania zasobów, niż w przypadku zapobiegania ich wystąpieniu. Zysk ten osiąga się jednak poprzez zwiększenie narzutu czasowego, związanego z koniecznością wykonywania algorytmu testowania w każdym przypadku żądania przydziału lub zwolnienia jednostek zasobów.

Niedoskonałość metod rozwiązywania konfliktów zasobowych w procesach komputerowych, w przypadku odniesienia ich do procesów produkcyjnych, wynika z dwóch przyczyn: (1) ograniczonych możliwości ich wykorzystania i (2) stosowania nadmiernych ograniczeń w szeregowaniu obsługi żądań przydziału zasobów.

Pierwsza z przyczyn wynika ze specyfiki niektórych klas procesów produkcyjnych, uniemożliwiającej spełnienie założeń istniejących modeli procesów i zasobów. Modelowanie procesów produkcyjnych złożonych z częściowo uporządkowanych ciągów operacji (np. procesów montażowych) wymaga uwzględnienia faktu, że ostatnia z wykorzystywanych przez pewien podproces posiadająca następniki jednostka zasobu (np. jednostka magazynu, w której zostaje złożony produkt finalny tego podprocesu) nie zostaje zwolniona z chwilą ukończenia jego działania. Zasób ten nie może jednak zostać zwolniony przez proces zewnętrzny (co pozwoliłoby na jego wyłączenie), a jedynie przez następnik podprocesu, który go zajął. Zasób ten zostaje zatem niejako

przekazany przez proces kończący swoje działanie jego następnikowi, a nie zwrócony do puli systemu, jak tego wymaga model procesów korzystających wspólnie ze zbioru zasobów zwrotnych. Oznacza to, że metody zabezpieczeń przed blokadami, opracowane dla tego modelu nie mogą być wykorzystane w przypadku rozważanych procesów produkcyjnych.

Dla wyrażenia współpracy powyższych procesów może być natomiast zastosowany model producentów i konsumentów komunikatów połączonych przez wspólne bufory o skończonej pojemności [80]. W odróżnieniu jednak od jednostek informacji (komunikatów), których "zużycie" może łączyć się z bezwarunkowym zwolnieniem bufora, przepływ półproduktów (zasobów niezwrotnych "zużywanych" przez procesy w celu "wytworzenia" innych) wymaga zawsze uwarunkowania możliwości zwolnienia jednego bufora (magazynu, przestrzeni roboczej) wolnym miejscem w innym. Warunek ten nie jest naturalnie konieczny w ogólnym przypadku przesyłania komunikatów w systemach komputerowych. W przypadkach szczególnych, w których zakłada się, że odbiór ("zużycie") komunikatu wymaga wysłania ("wytworzenia") odpowiedzi, stosuje się rozwiązania [80] polegające np. na wykorzystaniu tego samego miejsca w buforze na komunikat i odpowiedź. Rozwiązanie takie nie jest dopuszczalne w ESP, gdzie kolejne miejsca pobytu przetwarzanych materiałów wyznacza marszruta technologiczna.

Spośród bardzo niewielu prac ukierunkowanych na rozwiązywanie konfliktów zasobowych w ESP można wymienić [1, 14, 108, 140, 195]. Prace te dotyczą liniowych struktur przepływu materiałów, a współdziałanie procesów spełnia założenia systemu procesów korzystających z zasobów zwrotnych. Wyznaczane tu metody zabezpieczeń przed blokadami nie mają zatem charakteru ogólnego

i nie znajdują zastosowania np. w przypadku scharakteryzowanych wcześniej złożonych procesów produkcyjnych.

Drugie z wcześniej wymienionych ograniczeń zastosowania istniejących metod zabezpieczeń przed blokadą wynika z silniejszych założeń dotyczących struktury i działania procesów produkcyjnych niż komputerowych. Stopień posiadanej informacji o przepływie strumieni materiałowych w ESP jest znacznie wyższy niż ten, jakiego znajomość zakładają algorytmy unikania blokad w systemach komputerowych. W szczególności znany jest tu porządek wykorzystywania zasobów przez procesy, co stwarza możliwość analizy bezpieczeństwa w ograniczeniu do tych stanów rozdziału zasobów i żądań procesów, jakie są faktycznie osiągalne z danego. Zakładanie najgorszego przypadku (jednoczesnych maksymalnych zapotrzebowań procesów) jest tu warunkiem zbyt restrykcyjnym.

Algorytm unikania blokad wyposażony w powyższe informacje pozwala w ogólnym przypadku zaakceptować żądania, których odrzucenia wymaga "bankier", a zatem prowadzi w ogólności do zwiększenia efektywności systemu.

Uwagi powyższe wskazują na potrzebę poszukiwania nowych modeli, oddających specyfikę procesów produkcyjnych w ESP, oraz nowych metod rozwiązywania występującym tam konfliktów zasobowych.

2.3. Sformułowanie problemu

Jak już wspomniano, ograniczenia przyjmowane na sposób realizacji procesów materiałowych są często silniejsze od przyjmowanych w procesach przetwarzania informacji. Czynniki ten powo-

duje, że metody rozstrzygania konfliktów zasobowych znajdujące zastosowanie w systemach komputerowych mogą w wielu przypadkach być nieprzydatne w ESP. Ponadto, brak dokładnych założeń dotyczących sposobu wykorzystywania zasobów przez procesy komputerowe wymusza ogólność rozwiązań, co obniża ich efektywność. Z uwagi na specyfikę procesów materiałowych (brak możliwości likwidacji procesów, wyłączenia zasobów czy przerywania wykonywanych operacji) pewne metody wykorzystywane w systemach komputerowych nie mogą być zastosowane w ESP. Czynniki powyższe decydują zatem o potrzebie opracowania nowych metod rozwiązywania konfliktów zasobowych, zgodnych ze specyfiką rozważanej klasy obiektów, tj. elastycznych systemów produkcyjnych.

Użyteczność takich metod jest uwarunkowana generowaniem przez nie algorytmów o niskiej złożoności obliczeniowej, umożliwiającej ich wykorzystanie w procedurach sterowania adaptacyjnego w czasie rzeczywistym. Opracowanie metody rozstrzygania konfliktów zasobowych o takiej własności stanowi z jednej strony istotny przyczynek dla rozwoju teorii automatycznego programowania (automatycznej syntezy procedur sterowania w czasie rzeczywistym), z drugiej zaś istotny element budowy systemów zintegrowanego modelowania ESP (umożliwiając automatyczną syntezę modeli symulacyjnych pozwalających na ocenę różnych dopuszczalnych wariantów sterowanych procesów, bądź to na etapie projektowania strukturalnego bądź też sterowania dyspozytorskiego w trybie on-line).

Przyjęte w pracy podejście do wyznaczania metod automatycznej syntezy dopuszczalnych procedur sterowania procesami współbieżnymi jest oparte na wspomnianym w p. 2.2 "podejściu od strony syntezy". Zgodnie z jego założeniami w sposób arbi-

tralny zostaje przyjęta klasa specyfikacji modelowanych procesów oraz klasa sieci Petriego, do której należeć będą ich modele dynamiczne. Dla zadanych klas wyznaczana jest transformacja pozwalająca w sposób algorytmiczny przyporządkować elementom zbioru specyfikacji odpowiadające im modele sieciowe spełniające zadane własności (w rozważanym przypadku bezblokadowość realizacji modelowanych procesów). Gwarancja poprawności transformacji wynika z warunków wystarczających dla spełnienia rozważanej własności, które są podstawą algorytmu jej realizacji.

Główny problem niniejszej pracy sformułowany jest przy następujących założeniach:

(1) Dany jest skończony zbiór złożonych procesów potokowych. Każdy z procesów opisany jest grafem detalooperacji wyznaczającym plan przepływu jednostek materiałów związanych z realizacją pojedynczego wyrobu. Potokowy charakter procesu oznacza, że wytwarzanie kolejnych wyrobów tego samego typu rozpoczyna się w miarę dostępności zasobów jeszcze przed ukończeniem rozpoczętych wcześniej, w wyniku czego w systemie mogą znajdować się półfabrykaty tego samego typu wyrobu w różnych fazach swojej produkcji.

(2) Wykonanie określonej operacji wymaga określonego realizatora (zasobu zwrotnego systemu). Przechowanie określonego detalu na określonym etapie zaawansowania procesu produkcyjnego (po wykonaniu jednej, a przed poddaniem go następnej operacji) wymaga określonego magazynu międzyoperacyjnego (zasobu zwrotnego systemu o skończonej liczbie jednostek).

Dla zadanej klasy specyfikacji procesów **SG** należy wyznaczyć model sieciowy dopuszczalnych realizacji procesów, tzn. wyznaczyć algorytm wykonujący transformację:

$$\text{TR} : \text{SC} \rightarrow \text{PHF}$$

dla dowolnie zadanego $\text{SC} \in \text{SC}$, gdzie:

(1.1) SC jest zbiorem wszystkich szóstek $\text{SC} = (\text{R}, \text{G}, \text{H}, \beta, \gamma, \text{C})$,

gdzie: R jest skończonym zbiorem realizatorów,

G jest skończonym zbiorem magazynów,

$\text{H} = (\text{T} \cup \text{E}, \alpha)$ jest grafem detalooperacji,

$\beta : \text{T} \rightarrow \text{R}$ jest funkcją wymagań zasobowych operacji,

$\gamma : \text{E} \rightarrow \text{G}$ jest funkcją wymagań zasobowych detali (obiektów operacji),

$\text{C} : \text{G} \rightarrow \{1, 2, \dots\}$ jest funkcją pojemności magazynów,

takich, że graf detalooperacji jest dwudzielny, acykliczny i złożony ze skończonej liczby słabospójnych podgrafów oraz stopień wejściowy i stopień wyjściowy każdego wierzchołka $e \in \text{E}$ wynosi jeden.

(1.2) PHF jest klasą modeli sieciowych o elementach $\text{PH}^\uparrow = (\text{PH}, \uparrow)$,

gdzie: $\text{PH} = (\text{PT}, \text{D})$ jest siecią Petriego z łukami wzbraniającymi, $\text{PH} \in \text{PH}$,

$\text{PT} = (\text{P}, \text{T}, \text{F}, \text{K}, \text{W}, \text{M}_0)$ jest siecią Petriego typu miejsce/przejscie, $\text{PT} \in \text{PT}$,

$\text{D} \subseteq \text{P} \times \text{T}$ jest relacją wzbraniającą,

$\uparrow$ jest warunkiem ograniczającym możliwość palenia przejść w uzależnieniu od markowania miejsc połączonych z danym przejściem relacją D .

Przyjmuje się, że transformacja ta powinna zapewniać, że uzyskiwany model odzwierciedla dopuszczalne tzn. bezblokowe realizacje rozważanych procesów.

Przedstawiony powyżej problem poszukiwania transformacji TR rozwiązywany jest w wyniku realizacji jego podproblemów. Dekompozycja ta wyróżnia:

(1) Problem modelowania: wyznaczyć transformację

$$TR1 : SC \longrightarrow PT,$$

taką, że $MO = \{ MO_{SC} \mid MO_{SC} = TR1(SC) \ \& \ SC \in SC \}$ jest klasą modeli sieciowych osiągalnych (możliwych) realizacji procesów,

Rozwiązaniem tego problemu jest algorytm o liniowej złożoności $O(n)$, gdzie n jest liczbą wierzchołków grafu detalooperacji H , przekształcający daną specyfikację SC w standardową postać struktury sieci PT [158]. Uzyskany model nie jest w ogólnym przypadku modelem dynamiki bezpiecznej.

(2) Problem synchronizacji: wyznaczyć transformację

$$TR2 : MO \longrightarrow PH$$

oraz warunek φ określony dla elementów klasy $MD = \{ MD_{SC} \mid MD_{SC} = TR2(MO_{SC}) \ \& \ SC \in SC \}$, takie, że $MD = \{ MD_{SC}^{\varphi} \mid SC \in SC \}$ jest klasą modeli sieciowych dopuszczalnych realizacji procesów.

Ponieważ problem taki nie posiada ogólnego rozwiązania ze względu na to, że istnieją modele MO_{SC} nierealizowalne (tzn. takie, w których każda realizacja prowadzi do blokady), zostaje on zdekomponowany na dwa podproblemy:

(2.1) Problem realizowalności: wyznaczyć warunek W wystarczający dla realizowalności modelu $MO_{SC} \in \mathbf{MO}$ oraz wykazać jego prawdziwość.

(2.2) Problem zabezpieczeń: wyznaczyć warunki wystarczające (strukturę relacji D i warunek ϕ) dla zabezpieczenia modelu $MO_{SC} \in \mathbf{MO}'$ (gdzie $\mathbf{MO}'$ podzbiór $\mathbf{MO}$ zawierający wszystkie modele MO_{SC} , dla których spełniony jest warunek W) przed występowaniem blokad oraz wykazać ich prawdziwość.

Rozwiązaniem rozważanego problemu synchronizacji są algorytmy wielomianowe [158]:

- testowania spełnialności warunku W przez zadany model $MO_{SC} \in \mathbf{MO}$,
- transformacji zadanego modelu $MO_{SC} \in \mathbf{MO}$ do $MD_{SC} \in \mathbf{MD}$,
- testowania spełnialności warunku ϕ dla zadanego przejścia t i markowania M zadanego modelu $MD_{SC} \in \mathbf{MD}$.

Problemem wykraczającym w zasadzie poza główny cel pracy, ale łączącym się z wcześniejszymi podproblemami przez to, że wymaga wyznaczenia jednej z dopuszczalnych realizacji procesów (tzn. realizacji odzwierciedlającej procedurę sterowania) jest:

(3) Problem optymalizacji: przy zadanym zbiorze dyscyplin obsługi R wyznaczyć dyscyplinę $r \in R$ optymalizującą zadane kryterium $k \in K$ dla zadanego modelu $MD_{SC} \in \mathbf{MD}$ przy założeniu określonych czasów wykonywania operacji (czasów palenia przejść $\tau(t)$, $t \in T$) oraz określonej serii produkcyjnej (liczby powtórzeń $d = d(i)$, $i \in \overline{1, n}$ każdego z n procesów wyznaczonych specyfikacją SC), tzn. wyznaczyć

$$\underset{r \in R}{\text{ext}} k(\text{MD}_{SC}^f, r, \tau, d),$$

gdzie $\text{ext} \in \{\max, \min\}$.

Rozwiązanie tego problemu sprowadza się do wykonania odpowiednich eksperymentów symulacji komputerowej.

Przedstawiona dekompozycja problemu ukazując sposób automatycznej syntezy procedur sterowania procesami współbieżnymi podkreśla heurystyczny charakter rozwiązywania poszczególnych podproblemów. Oznacza to, że zbiór dopuszczalnych realizacji, ujęty przez generowany model, jest w ogólnym przypadku pewnym podzbiorem wszystkich dopuszczalnych realizacji. Zatem postulowana metoda przedstawia podejście suboptymalne (w wyżej wymienionym sensie) do konstrukcji procedur sterowania współbieżnego w czasie rzeczywistym.

Przedstawiona dekompozycja problemu oraz zasygnalizowane metody rozwiązywania jego podproblemów znajdują odbicie w układzie niniejszej pracy.

2.4. Układ pracy

Przedstawione dotąd dwa rozdziały mają charakter wprowadzający.

Pierwszy z nich ma na celu przedstawienie obiektu dalszych badań - elastycznych systemów produkcyjnych, problemów związanych z ich projektowaniem i eksploatacją oraz wspomagających realizację tych zadań metod i modeli.

Rozdział drugi ukazuje potrzebę zintegrowanego podejścia do modelowania ESP i ukazując niedostatek aktualnie dostępnych

metod i technik uzasadnia przyjęte sformułowanie problemu oraz drogę prowadzącą do jego rozwiązania.

Efektem poszukiwań narzędzia przydatnego dla opisu modeli dynamicznych jest rozdział trzeci, będący skrótowym przeglądem różnych klas sieci Petriego, w tym systemów typu P/T wykorzystanych dalej jako podstawa formalna modelowania.

Zasadniczą część pracy stanowią rozdziały czwarty i piąty, które poświęcone są syntezie dynamicznych modeli ESP.

Podrozdziały 4.1 i 4.2 przedstawiają, w sposób bardziej szczegółowy niż zostało to dokonane we wstępie, sformułowanie problemu pracy. Ponadto przedstawiono tu założenia i ograniczenia wyznaczające zakres dopuszczalnej dynamiki rozważanej klasy procesów.

Podrozdział 4.3 poświęcony jest rozwiązaniu problemu modelowania. W oparciu o otrzymany model zagadnienia blokad są analizowane w części 4.4. Zostaje tu wykazane, że problemy optymalnego testowania realizowalności modeli $MO_{SC} \in MO$ oraz optymalnego zabezpieczenia przed blokadami (tj. problemy, których rozwiązanie powinno dawać jednoznaczną odpowiedź na zadane pytanie) są NP-trudne. Dla obniżenia ich złożoności przyjmuje się rozwiązanie suboptymalne, tj. stwierdzające jedynie spełnianie warunków dostatecznych. Wyznaczenie tych warunków należy do kolejnego etapu dekompozycji problemu głównego, tzn. do podproblemu synchronizacji. Zostają one sformułowane i wykazane w podrozdziale 4.5. Warunki te umożliwiają budowę algorytmu transformacji modeli MO_{SC} , które odzwierciedlają możliwe realizacje procesów, do modeli MD_{SC} , które odzwierciedlają dopuszczalne realizacje procesów.

Ponieważ dla potrzeb zapewnienia dopuszczalności realizacji procesów wystarczająca jest reprezentacja operacji przy pomocy jednego zdarzenia (co pozwala na obniżenie złożoności modelu) założenie to zostało przyjęte w fazie konstrukcji klasy modeli **MD**. W celu zapewnienia potencjalnie bardziej efektywnej gospodarki zasobami należy uwzględnić zdarzenia zwrotu pewnych zasobów także w trakcie ich realizacji. Wymaga to "rozbicia" zdarzenia złożonego na składowe zdarzenia elementarne. Transformacja taka jest dokonana w podrozdziale 4.6. Jej wynikiem jest "etapowy" model dopuszczalnych realizacji procesów (uwzględniający zdarzenia rozpoczęcia, zakończenia operacji i częściowego zwrotu zasobów). Model ten zostaje bezpośrednio wykorzystany w rozwiązaniu problemu optymalizacji, tj. konstrukcji modeli symulacyjnych (rozdział 5).

Rozdział 6 przedstawia symulacyjny system wspomagania projektowania i eksploatacji ESP. W punkcie 6.1 ujęta została charakterystyka użytkowa systemu, w punkcie 6.2 jego struktura i działanie oraz w punkcie 6.3 przykłady zastosowań.

Wnioski podsumowujące zawarto w rozdziale 7, po którym następuje spis literatury.

3. SYSTEMY SIECI PETRIEGO

Teoria sieci jest dziedziną teorii systemów, która powstała w wyniku dążenia do przyczynowo-zdarzeniowego opisu zjawisk zachodzących w systemach składających się ze skończonej liczby asynchronicznie działających podukładów. Bazowym elementem konstrukcji wszystkich modeli tworzonych w ramach tejże teorii jest abstrakcyjne pojęcie sieci, natomiast zasadnicze różnice występują w stopniu agregacji opisu modelowanych zjawisk i występujących między nimi zależności.

W niniejszym rozdziale przedstawione zostaną podstawowe klasy sieci Petriego, ich własności oraz obszary zastosowań. Celem tych rozważań jest uzasadnienie wyboru aparatu sieciowego (w szczególności systemów typu miejsce/przejście) do modelowania procesów produkcyjnych w ESP oraz podkreślenie związków zachodzących pomiędzy własnościami modelowanych obiektów, a ich modelami sieciowymi.

3.1. Sieci

Teorię systemów rozproszonych można wyróżnić poprzez sposób, w jaki formułuje się dwa powiązane ze sobą pojęcia - stanu i zmiany stanu systemu (przejścia). Podstawowe założenia teorii sieci można przedstawić następująco:

- (i) Stany i przejścia są dwoma przeplatającymi się, ale odrębnymi zjawiskami, które powinny być traktowane w sposób jednakowy.
- (ii) Zarówno stany jak i przejścia podlegają określonym rozkładom na stany i przejścia składowe.

- (iii) Zakres zmian spowodowanych przez dane przejście jest ustalony i nie zależy od stanu, w jakim zachodzi.
- (iv) Określone przejście może zajść w danym stanie wtedy i tylko wtedy, jeżeli zakres zmian związanych z tym przejściem jest w danym stanie możliwy.

Konstrukcja modeli uwzględniających powyższe założenia wykorzystuje obiekt matematyczny jakim jest sieć [199].

Definicja 3.1

Trójkę uporządkowaną $N = (P, T, F)$, gdzie P i T są zbiorami skończonymi nazywamy siecią, jeżeli

- (1) $P \cup T \neq \emptyset$, $P \cap T = \emptyset$,
- (2) $F \subseteq (P \times T) \cup (T \times P)$,
- (3) $\text{dom}(F) \cup \text{cod}(F) = P \cup T$.

Z punktu widzenia teorii grafów sieć jest zatem dwudzielnym grafem skierowanym bez izolowanych wierzchołków. W zastosowaniu do systemów sieciowych zbiór P pozwoli nam wyodrębnić składowe stanu systemu, zbiór T - składowe przejść, natomiast relacja przepływu F - zakres zmian stanu spowodowanych zaistnieniem danego przejścia. W zależności od konstruowanego systemu elementy zbioru P będą interpretowane jako warunki, miejsca, predykaty itp. i przedstawiane są na schematach w postaci okręgów. Zbiór T będziemy interpretować jako zbiór zdarzeń, procesów, przejść itp., którego elementy mają graficzną postać kwadratów lub odcinków.

W celu uproszczenia opisu omawianych dalej systemów wprowadzimy następującą notację definiującą "lokalne" fragmenty relacji przepływu F sieci $N = (P, T, F)$.

Definicja 3.2

Dla sieci $N = (P, T, F)$ wprowadzimy następujące oznaczenia

- (1) $\cdot x = \{y \mid y \in Y \ \& \ (y, x) \in F\},$
- (2) $x^{\cdot} = \{y \mid y \in Y \ \& \ (x, y) \in F\},$
- (3) $\cdot Y = \bigcup_{x \in Y} \cdot x, \quad Y^{\cdot} = \bigcup_{x \in Y} x^{\cdot},$

gdzie $x, y \in P \cup T, \quad Y \subseteq P \cup T.$

Trójka $N = (P, T, F)$ nazwana siecią stanowi podstawowe pojęcie wykorzystywane w określeniach różnych klas sieci Petriego.

3.2. Elementarne systemy sieciowe

W elementarnych systemach sieciowych zakłada się, że stany systemu można opisać poprzez podanie warunków, jakie w danej sytuacji obowiązują (konfigurację). Zmiany obowiązywania warunków (zmiany konfiguracji) spowodowane są wystąpieniem zdarzeń. Warunki i zdarzenia są ze sobą powiązane przez następujące związki [199].

Definicja 3.3

Czwórkę uporządkowaną $NE = (P, T, F, c_{in})$ nazywamy elementarnym systemem sieciowym (systemem EN) jeżeli (P, T, F) jest siecią, a $c_{in} \subseteq P$ jest początkową konfiguracją NE.

Będziemy zakładać, że elementy $p \in P$ stanowią elementarne warunki, a elementy $t \in T$ elementarne zdarzenia systemu EN. Konfigurację c_{in} (lub dowolną inną konfigurację c) systemu NE można przedstawić graficznie poprzez umieszczenie znaczników (kroppek) w wierzchołkach $p \in c_{in}$ (odpowiednio $p \in c$) sieci (P, T, F) .

Definicja 3.4

Niech $NE = (P, T, F, c_{in})$ będzie systemem EN.

- (1) Zdarzenie $t \in T$ może wystąpić w konfiguracji $c \subseteq P$ (ma koncesję w c) wtedy i tylko wtedy, jeżeli $\cdot t \subseteq c$ i $t \cap c = \emptyset$.
- (2) Zbiór zdarzeń $G \subseteq P$ może wystąpić współbieżnie w konfiguracji $c \subseteq P$ wtedy i tylko wtedy, jeżeli dowolne dwa różne zdarzenia $t', t'' \in G$ są przyczynowo niezależne, tj.
 $(\forall t', t'' \in G)(t' \neq t'' \Rightarrow \cdot t' \cap \cdot t'' = \emptyset \ \& \ t' \cap t'' = \emptyset)$
i każde zdarzenie $t \in G$ ma koncesję w c .
- (3) Wystąpienie zbioru zdarzeń $G \subseteq P$ (zdarzenia $t \in P$) w konfiguracji c powoduje zmianę konfiguracji c na $c' = (c \setminus \cdot G) \cup G$ (odpowiednio $c' = (c \setminus \cdot t) \cup t$), co będziemy oznaczać przez $c[G \rangle c'$ (odpowiednio $c[t \rangle c'$).

W niektórych zastosowaniach wadą systemów EN jest możliwość analizy jedynie przyszłych zachowań modelowanych układów (brak możliwości analizy "w tył", tj. prześledzenia wcześniejszego zachowania systemu). Powyższa niedogodność nie występuje w systemie warunkowo-zdarzeniowym będącym restrykcyjnym systemem sieciowym definiowanym następująco.

Definicja 3.5

Czwórkę uporządkowaną $CE = (P, T, F, C)$ nazywamy systemem warunkowo-zdarzeniowym (systemem CE) jeżeli:

- (1) (P, T, F) jest siecią prostą,
- (2) $C \subseteq 2^P$ jest klasą równoważności relacji osiągalności $R_{CE} = (r_{CE} \cup r_{CE}^{-1})^*$, gdzie $r_{CE} \subseteq 2^P \times 2^P$ jest dana przez:
 $c' r_{CE} c'' \iff (\exists G \subseteq T)(c'[G \rangle c'')$.
- (3) $(\forall t \in T)(\exists c \in C)(t \text{ ma koncesję w } c)$.

Uwaga

- (i) Sieć N jest prosta jeżeli $(\forall x, y \in P \cup T)(\cdot x = \cdot y \ \& \ x^{\circ} = y^{\circ} \Rightarrow x = y)$.
- (ii) Przez φ^* oznaczamy zwrotne i przechodnie domknięcie relacji φ ; 2^P oznacza zbiór potęgowy P .

Systemy CE są podstawowymi modelami elementarnych sieci Petriego, aczkolwiek inne podklasy systemów EN (S-grafy, T-grafy, sieci wolnego wyboru) mają również bogatą literaturę (np. [198]).

3.3. Systemy typu miejsce/przejście

3.3.1. Podstawowe określenia

Systemy miejsc i przejść (systemy P/T, sieci Petri) można traktować w pewnym stopniu jako uogólnienie systemów EN. Zasadnicza idea polega na zniesieniu ograniczenia występowania co najwyżej jednego znacznika w dowolnym wierzchołku $p \in P$ sieci opisującej system. Konsekwencją tej koncepcji jest możliwość reprezentacji kilku warunków poprzez jeden wierzchołek $p \in P$, oraz kilku zdarzeń poprzez jeden wierzchołek $t \in T$ sieci (P, T, F) . Następująca definicja opisuje statyczne aspekty systemów P/T [155, 156].

Definicja 3.6

Szóstkę uporządkowaną $PN = (P, T, F, K, W, M_0)$ nazywamy systemem P/T jeżeli

- (1) (P, T, F) jest siecią; wierzchołki $p \in P$ nazywane są miejscami a wierzchołki $t \in T$ nazywane są przejściami.

- (2) $K : P \rightarrow N^+ \cup \{\infty\}$ jest funkcją pojemności miejsc.
- (3) $W : F \rightarrow N^+$ jest funkcją krotności łuków.
- (4) $M_0 : P \rightarrow N$ jest markowaniem początkowym spełniającym $M_0(p) \leq K(p)$ dla każdego $p \in P$.

Często wygodnie jest przyjąć, że funkcja krotności łuków W dana jest w postaci kanonicznej $W : (P \times T) \cup (T \times P) \rightarrow N$ zakładając, że $W(x,y) = 0$ jeżeli $(x,y) \notin F$. Funkcję w tej postaci wykorzystujemy w poniższej definicji określającej dynamiczne aspekty systemów P/T .

Definicja 3.7

Dla systemu PN jak w Definicji 3.6 określamy:

- (1) Markowaniem PN jest funkcja $M : P \rightarrow N$ taka, że $M(p) \leq K(p)$ dla każdego $p \in P$.
- (2) Przejście $t \in T$ jest przygotowane w markowaniu M jeżeli $(\forall p \in P)(W(p,t) \leq M(p) \leq K(p) - W(t,p))$.
- (3) Jeżeli przejście t jest przygotowane w markowaniu M , to t może być odpalone powodując zmianę markowania z M na M' zgodnie z formułą $(\forall p \in P)(M'(p) = M(p) - W(p,t) + W(t,p))$.
- (4) Zmianę markowania z M na M' spowodowaną odpaleniem przejścia t będziemy oznaczać przez $M[t \rangle M'$.
- (5) Przez $[M_0 \rangle$ będziemy oznaczać najmniejszy zbiór markowań PN , zwany zbiorem markowań osiągalnych w PN taki, że
 - 1. $M_0 \in [M_0 \rangle$
 - 2. Jeżeli $M_1 \in [M_0 \rangle$ i $M_1[t \rangle M_2$ dla pewnego $t \in T$, to $M_2 \in [M_0 \rangle$.
- (6) Sekwencja $\sigma = M_0 t_1 M_1 t_2 M_2 \dots$ (skończona lub nie) taka, że $M_0[t_1 \rangle M_1, M_1[t_2 \rangle M_2, \dots$ jest nazywana sekwencją palen w PN .

Istotnym wnioskiem z powyższej definicji jest fakt, że podobnie jak w systemach EN, warunek przygotowania danego przejścia t oraz zakres zmian spowodowanych jego odpaleniem jest lokalny, tzn. jest ograniczony do najbliższego sąsiedztwa przejścia t (miejsc związanych z przejściem t poprzez relację F).

Odpalenie pojedynczego przejścia powoduje elementarną zmianę markowania w PN. Niekiedy rozważa się również zmiany stanu systemu spowodowane współbieżnym wystąpieniem pewnego zbioru zdarzeń.

Definicja 3.8

Dla systemu PN jak w Definicji 3.6 określamy:

- (1) $G \subseteq T$ jest zbiorem przejść współbieżnie przygotowanych w markowaniu M jeżeli

$$(\forall p \in P) \left(\sum_{t \in G} W(p, t) \leq M(p) \leq K(p) - \sum_{t \in G} W(t, p) \right)$$

- (2) Jeżeli $G \subseteq T$ jest zbiorem przejść współbieżnie przygotowanych w M , to przejścia te mogą być odpalone powodując zmianę markowania z M na M' zgodnie z formułą:

$$(\forall p \in P) (M'(p) = M(p) - \sum_{t \in G} W(p, t) + \sum_{t \in G} W(t, p)).$$

3.3.2. Ograniczenia i rozszerzenia systemów P/T

W wielu przypadkach problemy występujące tak w analizie systemów P/T, jak i w syntezie modeli o zadanych własnościach, okazują się być zbyt złożone obliczeniowo lub wręcz nierozstrzygalne (w sensie niemożliwości konstrukcji algorytmu prowadzącego do rozwiązania [185]). Te same problemy można niekiedy rozwiązać w odniesieniu do pewnych podklas systemu P/T powstałych poprzez nałożenie określonych ograniczeń syntaktycznych i/lub strukturalnych.

Najwięcej miejsca w literaturze poświęcono tzw. sieciom markowanym [44,141] oraz ich podklasom: maszynom stanowym i grafom markowanym, a także [51,198] sieciom wolnego wyboru i prostym sieciom Petriego. Systemy te definiuje się następująco.

Definicja 3.9

- (1) Siecią markowaną jest system P/T taki, że
 $(\forall p \in P)(K(p) = \infty) \ \& \ (\forall x, y \in F)(W(x, y) = 1).$
- (2) Maszyną stanową jest sieć markowana taka, że
 $(\forall t \in T)(\| \cdot t \| = \| t \cdot \| = 1).$
- (3) Grafem markowanym jest sieć markowana taka, że
 $(\forall p \in P)(\| \cdot p \| = \| p \cdot \| = 1).$
- (4) Siecią wolnego wyboru jest sieć markowana taka, że
 $(\forall (p, t) \in F)(\| \cdot t \| = 1 \vee \| p \cdot \| = 1).$
- (5) Prostą siecią Petriego jest sieć markowana taka, że
 $(\forall t', t'', t''' \in T)(\cdot t' \cap \cdot t'' \neq \emptyset \Rightarrow \cdot t' \cap \cdot t''' = \emptyset \ \& \ \& \ \cdot t'' \cap \cdot t''' = \emptyset \ \& \ \| \cdot t' \cap \cdot t'' \| = 1).$

Z drugiej strony, system P/T okazuje się niekiedy zbyt prostym modelem dla reprezentacji wielu systemów rzeczywistych. W celu zwiększenia jego mocy modelowania wprowadza się tzw. rozszerzone systemy P/T, których przykładami mogą być:

- sieci priorytetowe [75,76], w których wyznacza się pewien porządek na zbiorze przejść pozwalający na wybór przygotowanego przejścia, które ma być odpalone w danym markowaniu,
- sieci zerujące [75], w których odpalenie określonych przejść powoduje wyzerowanie markowania określonych miejsc,
- sieci samomodyfikujące się [208], w których krotność łuku jest zmienną, której wartość zależy od aktualnej liczby znaczników określonego miejsca związanego z danym łukiem,

- adaptacyjne sieci Petriego [46], będące szczególnym przypadkiem sieci samomodyfikujących się, w których brak znaczników w danym miejscu sieci powoduje "usunięcie" wyznaczonego łuku.

Z punktu widzenia zastosowania omawianego formalizmu w niniejszej pracy najbardziej istotnym rozszerzeniem systemów P/T są sieci z łukami wzbraniającymi. W swojej podstawowej formie [75] są to systemy P/T uzupełnione o relację $D \subseteq P \times T$, w których możliwość odpalenia danego przejścia jest dodatkowo uwarunkowana niezerowym markowaniem wszystkich miejsc związanych z tym przejściem poprzez relację D .

Dla naszych potrzeb rozszerzymy warunek przygotowania sieci z łukami wzbraniającymi zakładając, że:

Definicja 3.10

Parą uporządkowaną $PH = (PN, D)$ będziemy nazywać systemem P/T z łukami wzbraniającymi, jeżeli:

- (1) PN jest systemem P/T jak w Definicji 3.6,
- (2) $D \subseteq P \times T$ jest relacją wzbronień,
- (3) dla systemu PH obowiązuje Definicja 3.7 za wyjątkiem p. 2 formułującego regułę przygotowania przejść, która dla PH ma postać restrykcyjną w stosunku do PN , tzn.
 - (3.1) przygotowanie przejścia $t \in T$ zależy nie tylko od stanu miejsc związanych z t poprzez relację F , ale również od stanu miejsc związanych z t poprzez relację D ,
 - (3.2) przejście które nie jest przygotowane w PN w danym markowaniu M , nie jest również przygotowane w PH .

Łatwo zauważyć, że stosunkowo proste rozszerzenie systemu P/T o łuki wzbraniające znacznie rozszerza możliwości modelowa-

nia, pozwalając na pożądaną modyfikację zachowania wyjściowego modelu poprzez uzupełnienie go relacją D i odpowiednim warunkiem ograniczającym przygotowanie danego przejścia t od markowania miejsc $p \in \{p \mid (p, t) \in D\}$.

3.4. Sieci wyższego rzędu

Modelowanie rzeczywistych systemów przy wykorzystaniu formalizmu systemów EN lub P/T prowadzi często do modeli, których analiza jest bardzo utrudniona ze względu na rozmiar otrzymanych sieci. W sytuacji, gdy modele te zawierają podobnie lub identycznie ustrukturalizowane podsieci, istnieje możliwość konstrukcji systemu, którego składowe (miejsca, przejścia, łuki) będą reprezentować podzbiory składowych sieci pierwotnej.

Idąc dalej, wprowadza się indywidualizację znaczników oraz konotację łuków definiującą typ znacznika (lub n -tki znaczników), jakie będą przepływać przez dany łuk. Ponadto, jeżeli zbiór znaczników indywidualnych można ustrukturalizować przy pomocy określonych operatorów i predykatów, przypisuje się przejściom pewne logiczne formuły (zbudowane ze zmiennych reprezentujących znaczniki oraz powyższych operatorów i predykatów). Ich wartość logiczna determinuje możliwość aktywacji danego przejścia przy określonym markowaniu najbliższego sąsiedztwa.

Najbardziej reprezentatywne dla sieci wyższego rzędu są sieci predykatowe oraz kolorowane sieci Petriego, które krótko przedstawimy poniżej. Należy zaznaczyć, że definiowane były też inne modele tego typu takie jak sieci predykatów i zdarzeń oraz sieci relacyjne [155,159].

3.4.1. Sieci predykatowe

Podstawą dla budowy sieci predykatowych (sieci PrT) jest zbiór obiektów ustrukturalizowany przy pomocy funkcji i relacji [63]. Struktura ta jest częściowo statyczna i częściowo dynamiczna. Część statyczną tworzy prosta struktura relacyjna (zwana suportem systemu dynamicznego) $R=(D, f_1, f_2, \dots, f_k, r_1, r_2, \dots, r_n)$, gdzie D jest niepustym zbiorem obiektów tworzących dziedzinę R , $\{f_i\}$ jest zbiorem funkcji w D , a $\{r_i\}$ jest zbiorem relacji w D . Strukturę dynamiczną charakteryzuje zmienność niektórych relacji wynikająca ze zmian stanu modelu. Sieć PrT można przedstawić jako sieć konotowaną, w której zmienne relacje reprezentowane są przez miejsca, natomiast symbole funkcji i relacji części statycznej (operatory i predykaty) występują w konotacji łuków i przejść.

Podanie ścisłej matematycznej definicji sieci predykatowych wymaga wcześniejszego określenia formalnego języka opisu struktury relacyjnej, jego słownika, semantyki i szeregu pojęć pochodnych. Dla naszych celów (skrótowego przeglądu systemów sieci Petriego) wystarczające jest mniej formalne, opisowe ujęcie sieci PrT, którą to definicję podajemy za [61].

Definicja 3.11

Sieć PrT jest obiektem złożonym z:

- (1) sieci (P, T, F) ,
- (2) struktury relacyjnej R złożonej ze zbioru obiektów (zindywidualizowanych znaczników) D oraz operatorów i predykatów,
- (3) konotatów do wszystkich łuków $f \in F$ będących formalnymi sumami n -tek zmiennych,
- (4) konotatów do niektórych przejść $t \in T$ będących formułami

logicznymi skonstruowanymi bez użycia kwantyfikatorów z powyższych zmiennych, operatorów i predykatów,

- (5) górnego ograniczenia K liczby możliwych wystąpień tej samej n -tki obiektów w dowolnym miejscu $p \in P$,
- (6) markowania początkowego miejsc $p \in P$ przy pomocy n -tek obiektów struktury R .

Przejście $t \in T$ jest przygotowane w danym markowaniu, jeżeli możliwe jest spełnienie przypisanej do niego formuły logicznej (poprzez podstawienie w miejsce zmiennych odpowiednich stałych występujących jako składowe n -tek markujących miejsca incydentne do t) oraz miejsca wejściowe $p \in {}^*t$ zawierają odpowiednią liczbę kopii n -tek wskazywanych przez konotat łuku (p, t) , a umieszczenie odpowiednich n -tek (wskazywanych przez konotat łuku (t, p)) w miejscach wyjściowych $p \in t^*$ nie spowoduje przekroczenia górnego ograniczenia K .

Zapalenie przygotowanego przejścia t powoduje usunięcie kopii odpowiednich n -tek z miejsc wejściowych t i umieszczenie kopii odpowiednich n -tek w miejscach wyjściowych t .

Należy podkreślić, że obniżenie złożoności modelowanego systemu poprzez jego konstrukcję w postaci sieci predykatowej (w stosunku do modelu tego samego systemu zbudowanego w postaci sieci EN lub P/T) nie implikuje w ogólności zmniejszenia stopnia trudności jego analizy. Zysk ten można osiągnąć tylko w sytuacji bardzo podobnej lub identycznej struktury i działania podsystemów modelowanego obiektu, co w ogólności nie dotyczy jednak systemu procesów produkcyjnych realizowanych w ESP.

3.4.2. Sieci kolorowane

Jakkolwiek sieci PrT okazały się w wielu przypadkach wygodnym narzędziem modelowania, to analiza ich zachowania, w szczególności w oparciu o metody niezmienników miejsc, okazała się bardzo trudna [62,46].

Powyższy fakt stanowi główną przyczynę rozwoju tzw. kolorowanych sieci Petriego [93], których geneza sięga sieci PrT, zasadnicze różnice występując jednak w spostrzeżeniu, że [95]:

- (1) Zbiór kolorów opisujących typ znaczników, jakie mogą występować w danym miejscu jest jawnie określony (w sieciach PrT odpalanie przejścia może generować nowe kolory, których może być nieskończona ilość).
- (2) Liczba znaczników dodawana lub usuwana z danego miejsca może być różna dla różnych kolorów odpalenia tego samego przejścia.
- (3) Zbiór operatorów i predykatów nie jest jawnie określony (można go jednak wygenerować, jeżeli zachodzi taka potrzeba, w oparciu o wybraną algebrę).

Poniższe definicje opisujące formalnie kolorowane sieci Petriego podajemy w oparciu o [5].

Definicja 3.12

Szóstkę uporządkowaną $CPN = (P, T, C, Pre, Post, M_0)$ nazywamy kolorowaną siecią Petriego (siecią CP) jeżeli:

- (i) $P = \{p\}$ i $T = \{t\}$ są skończonymi zbiorami odpowiednio miejsc i przejść,
- (ii) $C(p)$ i $C(t)$ są zbiorami kolorów, przyporządkowanych odpowiednio każdemu miejscu $p \in P$ i każdemu przejściu $t \in T$.

- (iii) Pre i Post są odpowiednio funkcją wejścia i funkcją wyjścia określonymi na $P \times T$ i takimi że:
- $$\text{Pre}(p,t), \text{Post}(p,t) : C(t) \rightarrow C(p) \cup \{0\}, \quad \forall (p,t) \in P \times T.$$
- (iv) M_0 , markowanie początkowe, jest funkcją określoną na P w taki sposób, że $M_0(p) : C(p) \rightarrow N, \quad \forall p \in P.$

Definicja 3.13

Dla sieci CP jak w Definicji 3.12 definiujemy:

- (i) Markowaniem sieci CP jest funkcja M określona na P taka, że $M(p) : C(p) \rightarrow N, \quad \forall p \in P,$
- (ii) Przejście $t \in T$ jest przygotowane dla koloru $c^* \in C(t)$ w markowaniu M jeżeli $(\forall p \in P)(M(p)(c) \geq 1 \ \& \ c = \text{Pre}(p,t)(c^*) \neq 0),$
- (iii) Przejście przygotowane dla c^* w M może być odpalone, co powoduje zmianę markowania z M na M' taką, że:

$$M'(p)(c) = \begin{cases} M(p)(c)-1, & \text{jeżeli } c = \text{Pre}(p,t)(c^*) \\ M(p)(c)+1, & \text{jeżeli } c = \text{Post}(p,t)(c^*) \\ M(p)(c), & \text{w pozostałych przypadkach.} \end{cases}$$

Uwaga

Powyższe definicje określają sieci CP w sposób uproszczony w stosunku do wprowadzonych pierwotnie w [93], zakładając, że funkcje $\text{Pre}(p,t)$ i $\text{Post}(p,t)$ odwzorowują pojedyncze kolory w pojedyncze kolory (w [93] przyjmuje się możliwość odwzorowań n -tek kolorów w n -tki kolorów).

Podobnie jak sieci PrT, sieci kolorowane znajdują zastosowanie w modelowaniu systemów o podobnej lub identycznej strukturze działania podsystemów. W przypadku ESP oznacza to, że mogłyby być przydatne w reprezentacji systemu procesów dotyczących przepływu różnych materiałów, ale wzdłuż tej samej

marszruty technologicznej. W pracy niniejszej będziemy rozważać sytuacje bardziej ogólne, co skłania nas do wyboru jako narzędzia modelowania innej klasy systemów sieciowych.

3.5. Czasowe systemy sieciowe

Dążąc do objęcia zakresem zastosowań modeli sieciowych nie tylko jakościową, ale także ilościową analizę działania reprezentowanych przez nie systemów, zaproponowana została [40, 145] koncepcja powiązania zmian stanu sieci z czasem. Związek ten może mieć charakter deterministyczny lub stochastyczny i wyrażany jest w modelu w postaci przypisania miejscom lub przejściom stałych (w przypadkach deterministycznych), np. [223], lub funkcji losowych (w przypadkach stochastycznych), np. [138], określających czas palenia przejść względnie czas pobytu znacznika w danym miejscu. Bazą dla modeli czasowych są zazwyczaj sieci P/T, rzadziej sieci wyższego rzędu, np. sieci kolorowane, w których opis czasowy dotyczy par (kolor, miejsce) [5].

Większość osiągnięć teoretycznych w dziedzinie systemów sieci Petriego dotyczy sieci tradycyjnych (nieczasowych), a rezultaty uzyskane dla sieci czasowych obowiązują jedynie w bardzo ograniczonych ich klasach. Niewątpliwie jednak, ze względu na możliwość ich bezpośredniego wykorzystania jako modeli symulacyjnych, stanowią one przydatny aparat dla ilościowej oceny działania systemów asynchronicznych.

3.6. Własności systemów sieciowych

Istnienie i pojawianie się wciąż nowych klas systemów sieci Petriego uwarunkowane jest z jednej strony potrzebami

opisu modelowanych obiektów, z drugiej zaś analizą lub syntezą modeli pod kątem posiadania przez nie określonych własności (konstrukcja warunków koniecznych lub wystarczających dla tego aby model spełniał daną własność jest na ogół możliwa dla pewnych klas systemów sieci Petriego).

Przedmiotem zainteresowań analityków są [129] zarówno własności zależne od markowania początkowego (behavioralne) jak i niezależne od M_0 (strukturalne). Do najczęściej stawianych tu problemów należą: czy dany system jest (względnie jak dokonać jego modyfikacji aby był) ograniczony, odwracalny, żywy, trwały, zachowawczy, zgodny oraz czy (względnie jak zapewnić aby) określone markowanie M' jest osiągalne z innego wyróżnionego markowania M .

Własności te dla systemów typu P/T definiowane są następująco (dla innych klas sieci Petriego są one określane identycznie lub analogicznie).

Definicja 3.14

Dla systemu sieci Petriego typu P/T w formie podstawowej lub rozszerzonej określa się:

- (1) markowanie M' jest osiągalne z markowania M , jeżeli $M' \in [M>$, tzn. istnieje skończona sekwencja paleń $\sigma = M_1 t_1 M_2 t_2, \dots, t_{n-1} M_n$ taka, że $M_1 = M$ i $M_n = M'$,
- (2) system jest k-ograniczony, jeżeli $(\forall M \in [M_0>)(\forall p \in P)(M(p) \leq K)$,
- (3) system jest trwały, jeżeli $(\forall M \in [M_0>)(\exists t \in T)(t \text{ jest przygotowane w } M)$,
- (4) system jest żywy, jeżeli $(\forall M \in [M_0>)(\forall t \in T)(\exists M' \in [M>)(t \text{ jest przygotowane w } M)$,

- (5) system jest zachowawczy, jeżeli istnieje funkcja
 $v : P \rightarrow \{0, 1, 2, \dots\}$ taka, że $(\forall M, M' \in [M_0])$
 $(\sum_{p \in P} v(p) \cdot M(p) = \sum_{p \in P} v(p) \cdot M'(p))$, wektor $[v(p)]_{n \times 1}$,
 $n = \|P\|$, jest nazywany P-niezmiennikiem danego systemu,
- (6) system jest zgodny, jeżeli istnieje funkcja
 $w : T \rightarrow \{0, 1, 2, \dots\}$ taka, że $(\forall p \in P) (\sum_{t \in p} w(t) \cdot W(t, p) =$
 $= \sum_{t \in p} w(t) W(p, t))$, wektor $[w(t)]_{m \times 1}$, $m = \|T\|$, jest na-
 zywany T - niezmiennikiem rozważanego systemu,
- (7) T - niezmiennik jest realizowany dla markowania $M \in [M_0]$,
 jeżeli istnieje sekwencja paleń $\sigma = M_1 t_1 M_2 t_2 \dots t_{s-1} M_s$ ta-
 ka, że $M_1 = M$ i każde przejście $t \in T$ występuje w σ dok-
 ładnie $w(t)$ razy; sekwencja taka prowadzi do markowania
 $M_s = M$.

Do podstawowych metod analizy systemów sieci Petriego pod względem posiadania przez nie określonych własności należą techniki oparte na analizie grafu osiągalności, równaniach macierzowych, redukcji i dekompozycji oraz analizie niezmienników.

Graf osiągalności złożony jest ze zbioru wierzchołków reprezentujących markowania $M \in [M_0]$ oraz łuków skierowanych od M do M' , jeżeli M' jest bezpośrednio osiągalne z M , z przypisanymi im przejściami, których zapalenie powoduje odpowiednią zmianę markowania. Analiza grafów osiągalności umożliwia potencjalnie badanie wszystkich behawioralnych własności systemów sieciowych. W praktyce możliwość tę ogranicza jednak znaczny, na ogół, rozmiar generowanych grafów.

Analiza sieci wykorzystująca równania macierzowe łączy się z konstrukcją tzw. macierzy zmian kompleksowych D [141] pozwa-

lających na powiązanie markowania M' z M poprzez $M' = M + \sigma_t D$, gdzie $\sigma_t = [t_1, t_2, \dots, t_n]$ jest wektorem paleń. Równania te stosowane są w niektórych przypadkach badań związanych z osiągalnością markowań i zachowawczością sieci.

Redukcja i dekompozycja systemu umożliwia często uproszczenie analizy jego żywości lub ograniczoności. Warunkiem koniecznym jest tu wyznaczenie takich transformacji, które zachowywałyby badaną własność w sieci zredukowanej lub każdym z podsystemów uzyskanych w drodze dekompozycji.

Zastosowanie niezmienników ogranicza się głównie do analizy własności strukturalnych takich jak ograniczoność, zachowawczość i zgodność. P- i T-niezmienniki mogą być wyznaczane w sposób automatyczny poprzez rozwiązywanie układu równań liniowych $x^T C = 0$ i $Cy = 0$, gdzie x, y wektory kolumnowe o liczbie elementów równych odpowiednio $\|P\|$ i $\|T\|$, C - macierz incydencji rozważanego systemu sieciowego.

Wyniki badań teoretycznych (np. [129, 141, 155, 185]) wskazują, że dla pewnych klas systemów sieci Petriego można już w chwili obecnej podać warunki konieczne lub wystarczające na to, aby spełniona była dla nich określona własność. Rezultaty takie odnośnie np. żywości grafów markowanych zostały przedstawione w [155]. Zagadnienie podania takich warunków w ogólnym przypadku np. systemów typu P/T jest problemem otwartym.

Należy zauważyć, że badane własności modeli sieciowych mają swoje odpowiedniki we własnościach modelowanych obiektów. Przykładowo, badaniu sterowalności systemu odpowiada problem osiągalności jego modelu sieciowego, badaniu k-ograniczoności - możliwość realizacji procesów przy zadanych ograniczeniach zasobowych. Oznacza to przydatność modeli sieciowych i metod ba-

dania ich własności do zadań analizy i syntezy obiektów rzeczywistych.

3.7. Zastosowania

Zalety aparatu sieci Petriego wiążą się z formułowaniem modeli grafowych (reprezentacji dającej możliwość odwoływania się do intuicji i doświadczenia projektanta) oraz modeli algebraicznych pozwalających na wyrażenie modeli sieciowych w językach programowania różnych poziomów. Formalizm ten umożliwia dynamiczny opis systemów, których dominującą cechą jest występowanie wzajemnie warunkujących się procesów, tzn. dynamiczny opis asynchronicznie zachodzących w systemie zdarzeń. Z każdym ze zdarzeń związany jest tu zbiór warunków wejściowych (koniecznych dla jego zajścia), których spełnienie umożliwia jego zajście i w konsekwencji spełnienie zbioru warunków wyjściowych stanowiących warunki wejściowe dla zachodzenia kolejnych zdarzeń.

Modele sieci Petriego odgrywają ważną rolę w zadaniach projektowania i eksploatacji systemów informatycznych, m.in. procesów obliczeniowych, metod alokacji zasobów, współpracy układów cyfrowych oraz struktury protokołów [7,32,39,46,61,94,155,185]. W okresie ostatnich dziesięciu lat obserwowany jest dynamiczny rozwój prac związanych z wykorzystaniem tego aparatu w zadaniach planowania i sterowania przepływem produkcji w ESP [5,10,11,13,43,52,59,124,125,133,157,177,221]. Ponadto systemy sieci Petriego znajdują zastosowanie także w modelowaniu procesów ciągłych np. występujących w przemyśle chemicznym [185]. W ogólności oznacza to, że mogą one być wykorzystywane

w modelowaniu systemów, gdzie występują zarówno ciągle, jak i dyskretne procesy przepływu strumieni materiałowych, energii i informacji.

Jak już wspomniano, podstawowym narzędziem modelowania przyjętym w rozważaniach niniejszej pracy są systemy typu miejsce/przejsięcie oraz ich rozszerzenia - systemy sieci Petriego ze wzbronieniami. Prowadzone badania mają na celu określenie warunków umożliwiających automatyczną syntezę modeli sieciowych złożonych procesów produkcyjnych, w sposób gwarantujący żywość i odwracalność generowanego systemu.

4. MODELOWANIE I SYNCHRONIZACJA PROCESÓW PRODUKCYJNYCH W ESP

4.1. Systemy współbieżnych złożonych procesów potokowych

Rozważając przepływ przedmiotów pracy w najbardziej, jak dotąd, powszechnej formie ESP - systemach obróbkowych, wystarczy zazwyczaj ograniczyć się do procesów liniowych, tj. takich, w których wytworzenie jednostki produktu łączy się z sekwencją operacji wykonywanych kolejno na pojedynczym elemencie na różnych stanowiskach systemu.

W systemach elastycznych o innych technikach wytwarzania (np. obróbki blach, odlewniczych, spawania, montażu) lub systemach tworzących wydziały umożliwiające pełną produkcję określonego asortymentu wyrobów, obejmującą różne fazy produkcji (np. produkcję przygotówek, obróbkę, montaż, wykańczanie, pakowanie) procesy przepływu materiałów posiadają często charakter bardziej złożony.

Przedmioty produkowane w fazie początkowej w sposób niezależny mogą zostać połączone (np. w wyniku montażu podzespołów,

pakowania kompletów, skompletowania partii wyrobów celem wspólnego transportu itp.) i poddawane dalszym operacjom jako zespół, względnie przeciwnie - przedmiot traktowany w pewnej fazie produkcji jako jednostkowy może ulec rozdziałowi (np. w wyniku cięcia blach, demontażu form odlewniczych, rozdziału wspólnie transportowanej partii materiałów itp.) i każda z części podlegać właściwym dla siebie dalszym operacjom. Procesy takie, jako uogólnienie procesów liniowych (prostych), będziemy dalej nazywać procesami złożonymi.

Wytworzenie serii jednakowych wyrobów wymaga wielokrotnego powtórzenia tego samego zbioru operacji na kolejno wprowadzanych do ESP przedmiotach pracy, przy czym rozpoczęcie kolejnych procesów jednostkowych (tj. związanych z wytworzeniem poszczególnych jednostek produktu) może, w miarę dostępności wolnych urządzeń, rozpoczynać się jeszcze przed ukończeniem wcześniejszych. W systemie generowany jest zatem potok jednakowych procesów jednostkowych lub inaczej - proces potokowy. Ponadto, występująca często w systemach elastycznych jednoczesna produkcja różnych wyrobów powoduje, że różne procesy potokowe będą tu przebiegać współbieżnie.

Przedmiotem naszych dalszych rozważań będzie system współbieżnie realizowanych względem siebie potoków jednostkowych procesów złożonych. Strukturę taką będziemy dalej nazywać systemem współbieżnych złożonych procesów potokowych (systemem WZPP, w skrócie).

4.2. Sformułowanie problemu

Będziemy traktować ESP jako system zasobów, w którym wyróżnimy zasoby zwrotne, tj. takie, które mogą być wielokrotnie uży-

wane przez ten sam lub różne procesy, oraz zasoby niezwrotne, których jednostki będą wytwarzane i zużywane w trakcie realizacji procesów. Do zasobów zwrotnych będziemy zaliczać dowolne grupy urządzeń (np. stacje obróbkowe, montażowe itp.) lub pojedyncze urządzenia (np. roboty przemysłowe, maszyny pomiarowe itp.) nazywając je ogólnie realizatorami, oraz wszelkiego typu bufory przystanowiskowe, magazyny międzyoperacyjne, stoły odkładnicze itp., nazywając je ogólnie magazynami i charakteryzując poprzez ich pojemność wyrażaną skończoną liczbę naturalną.

Zasoby niezwrotne będą reprezentować przedmioty pracy wyróżnione ze względu na ich rodzaj oraz fazę procesu produkcyjnego, z którą są związane. Oznacza to, że dwa elementy tego samego typu będziemy traktować jako jednostki tego samego przedmiotu wtedy i tylko wtedy, jeżeli są one na tym samym etapie wytwarzania. Poddanie danego elementu kolejnej operacji, powodując zmianę stopnia zaawansowania jego produkcji, będzie w kategoriach modelowych traktowane jako zużycie jednostki jednego przedmiotu celem wytworzenia jednostki innego. Interpretacja taka pozwala nam w szczególności wyrazić zmiany wynikające z realizacji operacji łączenia (np. montażu) lub rozdziału (np. demontażu, cięcia).

Zasoby niezwrotne będą dalej nazywane przedmiotami i charakteryzowane aktualną, zmieniającą się na skutek realizacji procesów, liczbą jednostek.

W rozważanym systemie realizowane są złożone procesy potokowe, z których każdy opisany jest grafem przepływu wskazującym na relację jaka zachodzi między operacjami i przedmiotami, oraz dwiema funkcjami określającymi wymagania procesu dotyczące zasobów stałych. Dane te będą ujmowane w specyfikacji procesu, którą określimy jak poniżej.

Definicja 4.1

Specyfikacją złożonego procesu potokowego π będziemy nazywać piątkę $S = (R, G, H_\pi, \beta_\pi, \gamma_\pi)$, gdzie:

(1) R jest skończonym zbiorem realizatorów r ;

(2) G jest skończonym zbiorem magazynów b ;

(3) $H_\pi = (T_\pi \cup E_\pi, \alpha_\pi)$, gdzie

T_π - skończony zbiór operacji t ,

E_π - skończony zbiór przedmiotów e ,

$\alpha_\pi \subseteq (T \times E) \cup (E \times T)$, jest grafem przepływu procesu spełniającym warunki:

(3.1) H_π jest skierowany, słabospójny i acykliczny,

(3.2) każdy element zbioru E_π posiada dokładnie jeden łuk wejściowy i jeden łuk wyjściowy;

(4) $\beta_\pi: T_\pi \rightarrow R$, $\gamma_\pi: E_\pi \rightarrow G$ są funkcjami wymagań zasobowych procesu π wskazującymi odpowiednio realizator $r = \beta_\pi(t)$ wyznaczony dla wykonania operacji $t \in T$ i magazyn $g = \gamma_\pi(e)$ wyznaczony dla przechowania przedmiotu $e \in E$.

Będziemy zakładać, że zarówno zbiory operacji jak i przedmiotów są jednoznacznie związane z określonym procesem, tj. zbiory operacji (przedmiotów) dwóch różnych procesów potokowych są rozłączne. Procesy współbieżne mogą natomiast korzystać z tych samych zasobów stałych ESP. Stąd, opisując system WZPP wygodnie jest połączyć programy poszczególnych procesów we wspólną strukturę, jak poniżej.

Definicja 4.2

Specyfikacją systemu WZPP będziemy nazywać piątkę

$S = (R, G, H, \beta, \gamma)$, $S = S(\pi)$, gdzie:

(1) π jest skończonym zbiorem procesów π o specyfikacji jak

w Definicji 4.1, którego każde dwa elementy π, π' spełniają implikację $(\pi \neq \pi') \Rightarrow (T_\pi \cap T_{\pi'} = \emptyset \ \& \ E_\pi \cap E_{\pi'} = \emptyset)$;

$$(2) \quad R = \bigcup_{\pi \in \Pi} \left\{ r \mid (\exists t \in T_\pi) (\beta_\pi(t) = r) \right\},$$

$$(3) \quad G = \bigcup_{\pi \in \Pi} \left\{ g \mid (\exists e \in E_\pi) (\gamma_\pi(e) = g) \right\},$$

$$(4) \quad H = (T \cup E, \alpha), \text{ gdzie } T = \bigcup_{\pi \in \Pi} T_\pi, \ E = \bigcup_{\pi \in \Pi} E_\pi,$$

$$\alpha = \bigcup_{\pi \in \Pi} \alpha_\pi;$$

$$(5) \quad \beta : T \longrightarrow R \text{ jest funkcją taką, że } (\forall \pi \in \Pi) (\beta|_{T_\pi} = \beta_\pi),$$

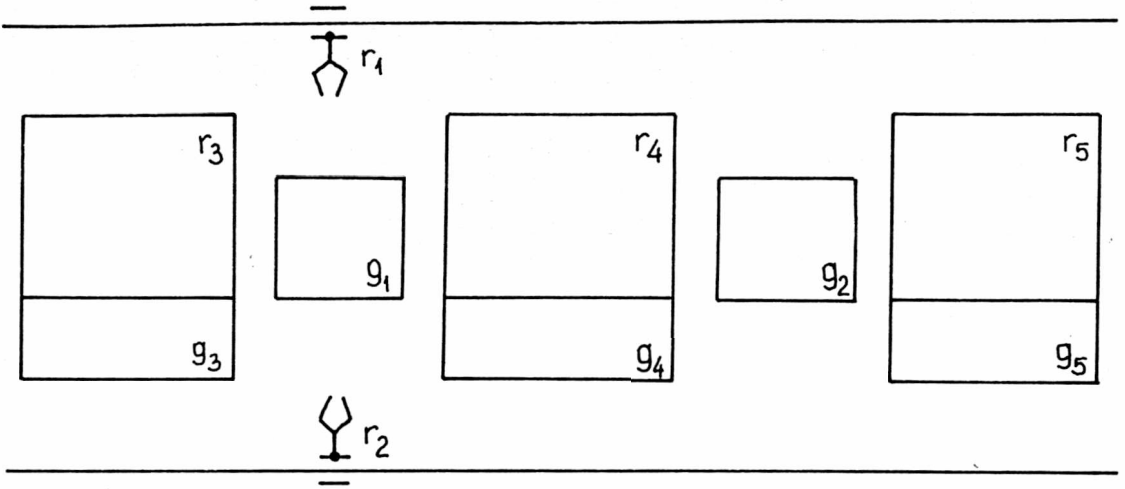
$$\gamma : E \longrightarrow G \text{ jest funkcją taką, że } (\forall \pi \in \Pi) (\gamma|_{E_\pi} = \gamma_\pi),$$

gdzie $\beta|_{T_\pi}$ ($\gamma|_{E_\pi}$) funkcja β (odpowiednio γ) obcięta do zbioru T_π (odpowiednio E_π).

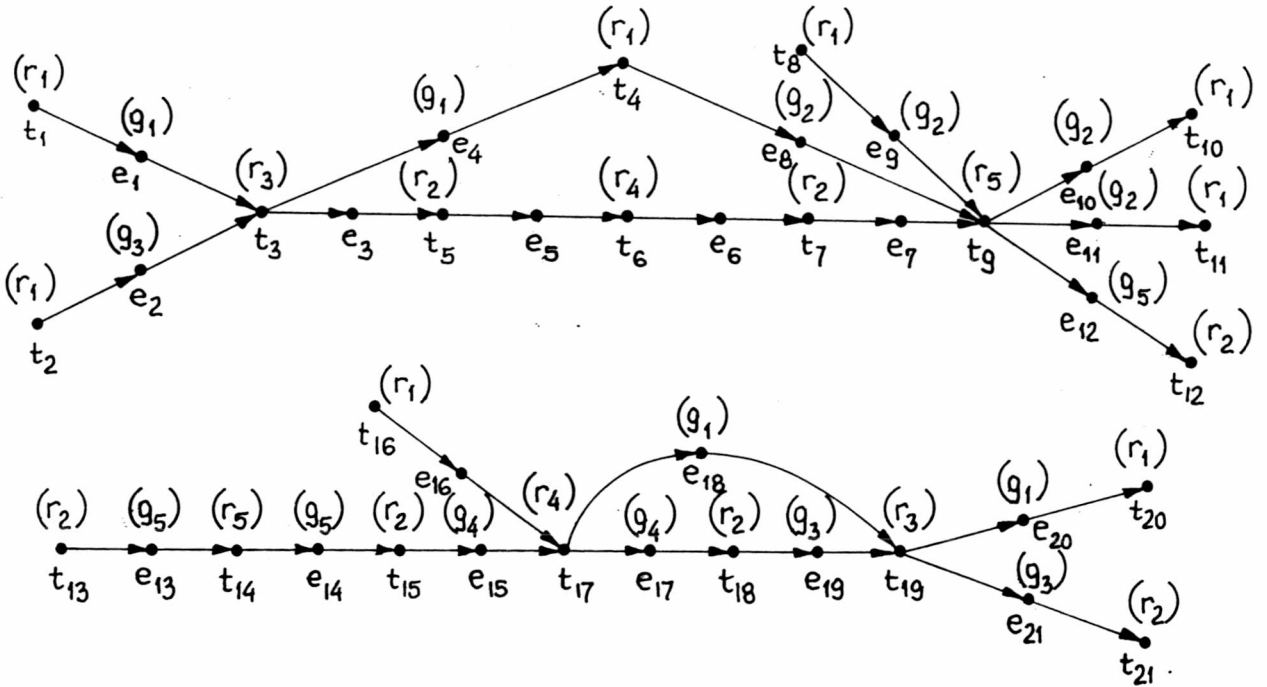
Warto zwrócić uwagę, że zgodnie z powyższymi definicjami rozważane zbiory procesów będą stanowić systemy otwarte - zarówno początkowe (tj. nie mające poprzedników), jak i końcowe (tj. nie mające następników) wierzchołki grafu przepływu H mogą należeć jedynie do zbioru operacji T ; będziemy je wiązać odpowiednio z wprowadzaniem i wyprowadzaniem przedmiotów do/z systemu.

Przykład 4.1

Rysunek 4.1 przedstawia schemat elastycznego systemu montażowego, w którym palety z elementem bazowym przekazywane są przy pomocy robota transportowego r_2 do odpowiednich stacji montażowych r_3, r_4, r_5 , wyposażonych w bufory przystanowiskowe g_3, g_4, g_5 . Palety z elementami wmontowywanymi w elementy bazowe przekazywane są przez robota transportowego r_1 do magazynów g_1 lub g_2 dostępnych bezpośrednio dla stacji r_3 i r_4 oraz (odpowiednio) r_4 i r_5 . Rysunek 4.2 przedstawia w formie graficznej przykład specyfikacji S obejmującej dwa procesy złożone realizowane współ-



Rys. 4.1. Elastyczny system montażowy; ilustracja do Przykładu 4.1.



Rys. 4.2. Przykładowa specyfikacja współbieżnych procesów złożonych występujących w systemie przedstawionym na Rys. 4.1.

bieżnie w powyższym systemie. Pokazany na rysunku graf jest grafem przepływu $H = (T \cup E, \alpha)$, natomiast symbole ujęte w nawiasy wyznaczają wymagania zasobowe procesów: $\beta(t)$, $t \in T$ i $\gamma(e)$, $e \in E$, zgodnie z typem wierzchołka.

Zakłada się, że realizacja procesów powoduje następujące zmiany stanu systemu zasobów:

- (1) W stanie początkowym realizatory i magazyny (wszystkie ich jednostki) są wolne, a liczba jednostek każdego z przedmiotów wynosi zero.
- (2) Stan zasobów nie ulega zmianie, gdy nie jest wykonywana żadna operacja.
- (3) Operacje wykonywane są asynchronicznie.
- (4) Wykonanie operacji $t \in T$ obejmuje:
 - zajęcie realizatora $\beta(t)$,
 - zwolnienie magazynów $g = \gamma(e)$, $e \in \{e | (e, t) \in \alpha\}$ (poprzez pobranie każdego z przedmiotów wejściowych operacji), w liczbie jednostek $\|\{e | (e, t) \in \alpha \ \& \ \gamma(e) = g\}\|$,
 - zużycie jednostki każdego z przedmiotów $e \in \{e | (e, t) \in \alpha\}$ i wytworzenie jednostki każdego z przedmiotów $e \in \{e | (t, e) \in \alpha\}$,
 - zajęcie magazynów $g = \gamma(e)$, $e \in \{e | (t, e) \in \alpha\}$ (poprzez złożenie każdego z przedmiotów wyjściowych operacji), w liczbie jednostek $\|\{e | (t, e) \in \alpha \ \& \ \gamma(e) = g\}\|$,
 - zwolnienie realizatora $\beta(t)$.

Poprawny przebieg procesów wymaga wprowadzenia zasad synchronizacji zapewniających, że nie zostaną przekroczone następujące ograniczenia:

- (1) (a) Operacja jest działaniem elementarnym, którego wykonywanie nie może zostać przerwane,
- (b) realizator może zostać zajęty tylko wtedy, jeżeli jest wolny,
- (c) jednostka magazynu może zostać zajęta tylko wtedy, gdy liczba zajętych jednostek tego magazynu jest mniejsza od jego pojemności,

(d) jednostka przedmiotu może zostać zużyta tylko wtedy, jeżeli liczba jednostek tego przedmiotu jest dodatnia.

(2) W systemie nie może wystąpić wzajemna blokada procesów.

Przykład 4.2

Celem ilustracji zjawiska blokady rozważmy system WZPP działający zgodnie ze specyfikacją jak na Rys. 4.2. Załóżmy, że system działa w oparciu o zasady synchronizacji uniemożliwiające wykonanie operacji zawsze i tylko wtedy jeżeli powodowałoby to przekroczenie ograniczeń (1). Możliwe jest zatem (przez wielokrotne powtórzenie operacji t_8) zajęcie wszystkich jednostek magazynu g_2 przedmiotami e_9 , co także oznacza, że liczba jednostek e_8 wynosi zero. Działanie zgodne z wymaganiami zasad synchronizacji nie pozwala aby operacja t_9 wystąpiła przed operacją t_8 (ponieważ wymagałoby to przekroczenia ograniczenia (1.c)) oraz aby operacja t_8 wystąpiła przed operacją t_9 (ponieważ wymagałoby to przekroczenia ograniczenia (1.d)). W sytuacji takiej żadna z operacji nigdy nie będzie mogła być wykonana i w konsekwencji mamy do czynienia z blokadą.

Problemem rozważanym w dalszej części tego rozdziału jest konstrukcja modelu dynamicznego ujmującego w możliwie pełnym zakresie asynchronizm systemu WZPP, a jednocześnie działającego w sposób poprawny. Innymi słowy, będziemy poszukiwać modelu, który pozwala wygenerować wyłącznie dopuszczalny i każdy z dopuszczalnych, ze względu na ograniczenia (1 - 2), współbieg procesów opisanych specyfikacją S i działających zgodnie z założeniami (1 - 4).

Jak wykażemy dalej, rozwiązanie takie jest możliwe, jednakże synchronizacja procesów wymaga wówczas podejmowania decyzji

w oparciu o algorytmy o nadmiernej, uniemożliwiającej wykorzystanie tych algorytmów, złożoności. W związku z powyższym postulowany wyżej problem zostanie nieco osłabiony i właściwym zadaniem tego rozdziału będzie konstrukcja modelu systemu WZPP działającego w sposób poprawny, zezwolimy jednak na wyłączenie, spośród osiągalnych w modelu, pewnych przebiegów dopuszczalnych, na korzyść obniżenia złożoności algorytmu synchronizacji.

Dla celów modelowania zostanie wykorzystany formalizm sieci Petriego, przy czym poszukiwany model zostanie zbudowany w dwóch etapach. W fazie pierwszej wyznaczymy model dynamiki osiągalnej ("osiągalnej" ze względów fizykalnych - ograniczenia (1)) systemu WZPP. W fazie drugiej dokonamy restrykcji dynamiki wcześniej otrzymanego modelu, celem niedopuszczenia do powstawania blokad (ograniczenie (2)), otrzymując w wyniku model dynamiki dopuszczalnej systemu WZPP.

4.3. Model dynamiki osiągalnej

Biorąc pod uwagę wcześniej omówione założenia dotyczące współdziałania rozważanych procesów, łatwo zauważyć, że ich osiągalna dynamika zależy od dwóch czynników - realizowanego zbioru procesów, opisanego programem S , i pojemności magazynów systemu, w którym procesy te są realizowane. Przedstawiony poniżej model ma charakter ogólny, tj. został zdefiniowany w oparciu o zmienne występujące w specyfikacji S i dla dowolnej funkcji pojemności magazynów $C : G \rightarrow \mathbb{N}^+$, wyznaczając w ten sposób klasę modeli, której elementy mogą być bezpośrednio wygenerowane dla dowolnych konkretnych wartości pary (S, C) .

Dla wyrażenia cech statycznych i dynamicznych modelu wykorzystano system P/T (patrz Definicje 3.6, 3.7), nieznacznie modyfikując jedynie formułę opisującą przygotowanie przejść.

Definicja 4.3

Modelem dynamiki osiągalnej systemu WZPP wyznaczonego parą (S, C) , gdzie S - specyfikacja systemu jak w Definicji 4.2, $C : G \rightarrow \{1, 2, 3, \dots\}$ - funkcja pojemności magazynów, będziemy nazywać system P/T, jak w Definicji 3.6, $MO_{SC} = (P, T, F, K, W, M_0)$ taki, że:

$$(1) \quad P = E \cup G \cup R,$$

$$(2) \quad T = T$$

$$(3) \quad F = FE \cup FG \cup FR, \quad \text{gdzie}$$

$$(3.1) \quad FE = \alpha,$$

$$(3.2) \quad FG = \left\{ (t, g) \mid (\exists e \in E)((t, e) \in \alpha \ \& \ \gamma(e) = g) \right\} \cup \\ \cup \left\{ (g, t) \mid (\exists e \in E)((e, t) \in \alpha \ \& \ \gamma(e) = g) \right\},$$

$$(3.3) \quad FR = \left\{ (r, t), (t, r) \mid t \in T \ \& \ \beta(t) = r \right\},$$

$$(4) \quad K(p) = \begin{cases} 1 & \text{jeżeli } p \in R \\ C(g) & \text{jeżeli } p \in G \\ C(g) & \text{jeżeli } p \in E \ \& \ \gamma(e) = g, \end{cases}$$

$$(5) \quad W(p, t) = \begin{cases} 1 & \text{jeżeli } p \in E \cup R \\ \|\{e \mid (e, t) \in \alpha \ \& \ \gamma(e) = g\}\| & \text{jeżeli } p = g, \end{cases}$$

$$W(t, p) = \begin{cases} 1 & \text{jeżeli } p \in E \cup R \\ \|\{e \mid (t, e) \in \alpha \ \& \ \gamma(e) = g\}\| & \text{jeżeli } p = g, \end{cases}$$

$$(6) \quad M_0(p) = \begin{cases} 1 & \text{jeżeli } p \in R \\ 0 & \text{jeżeli } p \in E \cup G. \end{cases}$$

Definicja 4.4

W systemie MO_{SC} , jak w Definicji 4.3, przejście $t \in T$ jest przygotowane w markowaniu M , jeżeli spełnione są warunki:

- (i) $(\forall p \in \cdot t)(M(p) \geq W(p, t)),$
- (ii) $(\forall p \in t \cdot)(M(p) \leq K(p) + W(p, t) - W(t, p)).$

Uwaga: Dla uproszczenia zapisu funkcja W występuje tu w postaci kanonicznej tzn. $W(p, t) = 0$ i $W(t, p) = 0$ dla wszystkich par takich, że odpowiednio $(p, t) \notin F$ i $(t, p) \notin F$.

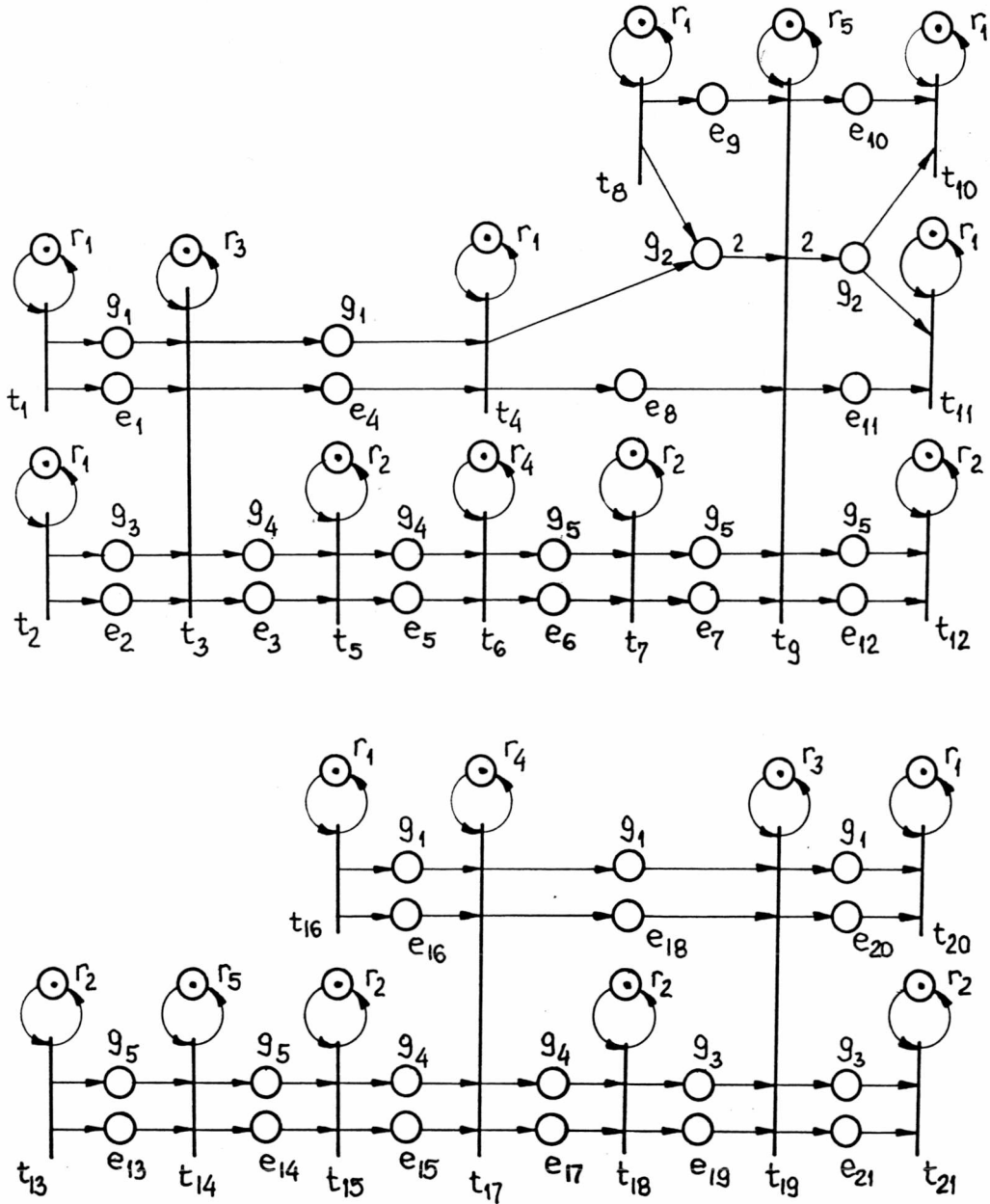
Pozostałe definicje dotyczące dynamiki MO_{SC} są zgodne z klasycznymi definicjami dla systemów P/T (patrz Definicja 3.7).

Przykład 4.3

Rysunek (4.3) przedstawia system P/T uzyskany zgodnie z Definicją 4.3 dla systemu WZPP o specyfikacji jak w Przykładzie 1, Rys. 4.2.

Porównując Rys. 4.2 i Rys. 4.3 łatwo zauważyć, że system MO_{SC} uzyskuje się z grafu przepływu procesu H poprzez uznanie zbioru węzłów T (reprezentujących operacje) za przejścia systemu, zaliczenie zbioru węzłów E (reprezentujących przedmioty) do zbioru miejsc systemu i uzupełnieniu go o miejsca reprezentujące realizatory R i magazyny G oraz powiązania przejść i miejsc relacją F . Relacja ta w przypadku par (e, t) i (t, e) odpowiada relacji α , a w pozostałych przypadkach, wraz z funkcją wagową łuków, wskazuje na sposób korzystania z zasobów stałych przez procesy. Wyrażając operację przy pomocy jednego tylko przejścia, którego zapalenie reprezentuje wykonanie wszystkich czynności wchodzących w skład operacji, zapewniamy jej nieprzerwaność (ograniczenie 1.a), a wyznaczone funkcją K pojemności miejsc i reguły określające dynamikę systemu wymuszają zachowanie pozostałych ograniczeń (1b - 1d). Opracowany model spełnia zatem wszystkie stawiane mu wymagania za wyjątkiem niedopusz-

czenia do powstawania blokad w systemie, co będzie przedmiotem dalszych rozważań.



Rys. 4.3. Przykładowy system MDSC. Model odpowiadający systemowi WZPP o specyfikacji S przedstawionym na Rys. 4.2 i dowolnej funkcji C.

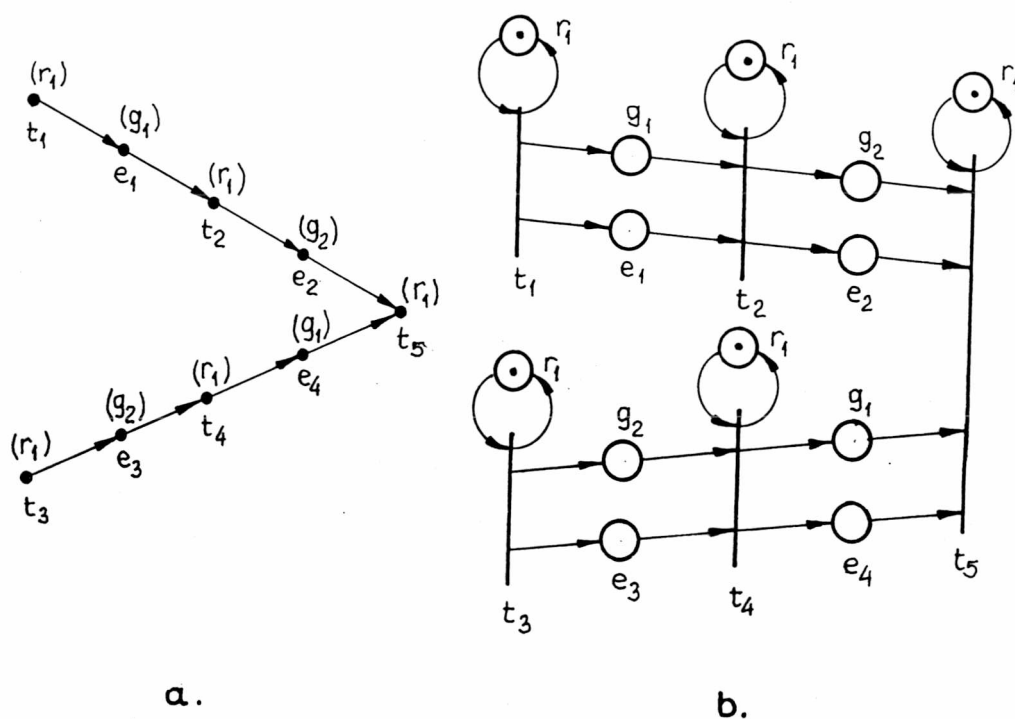
4.4. Problemy realizowalności i powrotu

Wyznaczając model MO_{SC} nie narzucaliśmy, jak dotąd, żadnych ograniczeń na pojemność magazynów C ze względu na specyfikację S realizowanych procesów. Rozważając dowolną parę (S, C) należy liczyć się z tym, co ilustruje poniższy przykład, że zbyt mała pojemność magazynów uniemożliwia poprawną realizację programu, tj. wystąpienie blokady jest nie do uniknięcia.

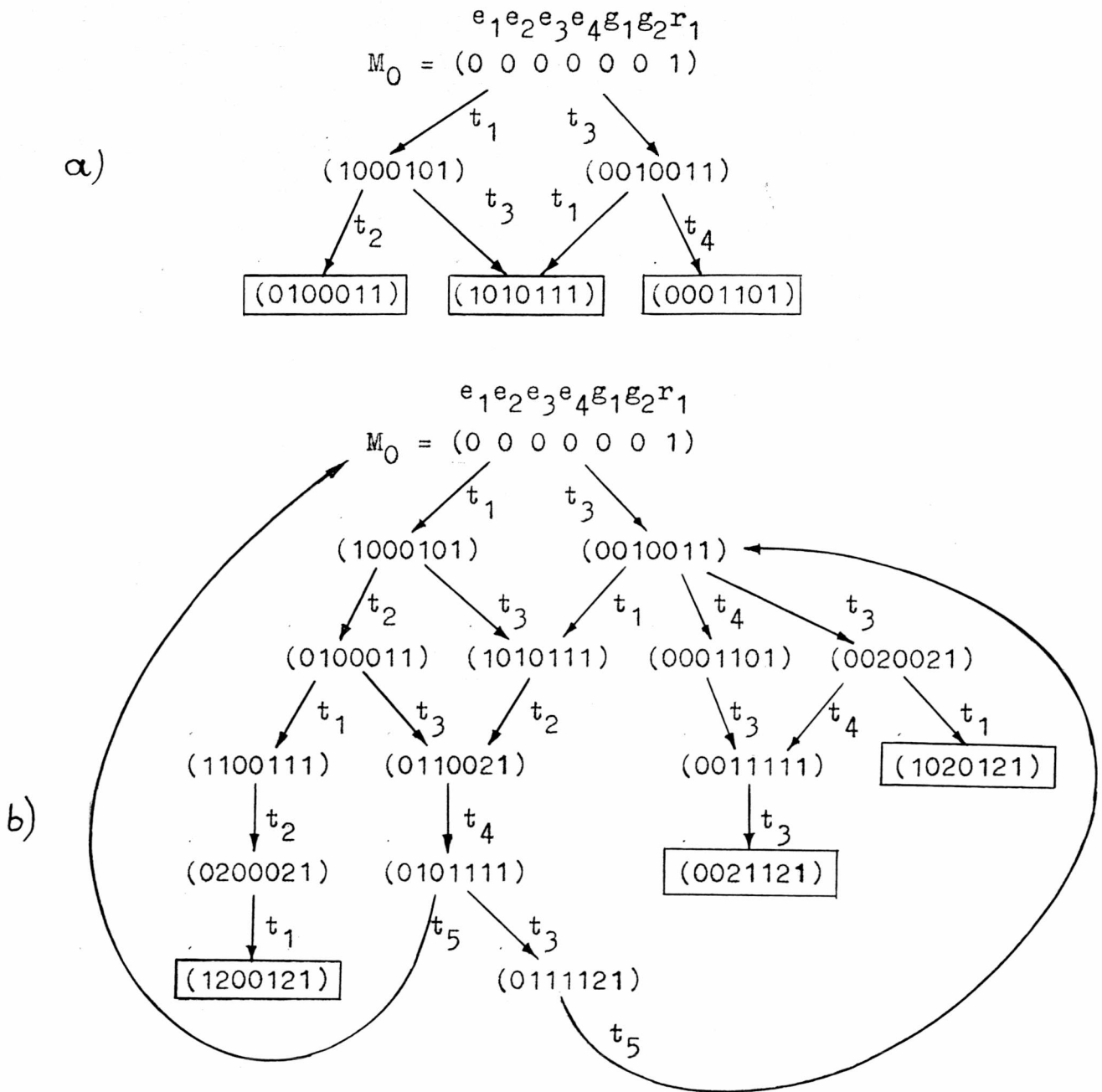
Przykład 4.4

Rysunki 4.4a i 4.4b przedstawiają specyfikację pewnego systemu WZPP i odpowiadający jej model MO_{SC} , natomiast Rys. 4.5a i 4.5b stanowią grafy markowań osiągalnych tegoż modelu przy założeniu: (a) $C(g_1) = C(g_2) = 1$ i (b) $C(g_1) = 1, C(g_2) = 2$.

Jak łatwo zauważyć, każda realizacja programu prowadzi w sytuacji (a) do blokady, podczas gdy w sytuacji (b) istnieje możliwość poprawnego wykonania wszystkich operacji przewidzianych specyfikacją i powrót do stanu początkowego.



Rys. 4.4. Ilustracja do Przykładu 4.4



Rys. 4.5. Ilustracja do przykładu 4.4.

W kategoriach przyjętego przez nas modelu stwierdzenie, czy program S jest realizowalny/nierealizowalny w ESP o pojemności magazynów C jest równoważny następującemu problemowi.

Definicja 4.5

Realizowalnością systemu M_{0SC} będziemy nazywać następujący problem:

Parametry: System $MO_{SC} = (P, T, F, K, W, M_0)$, jak w Def. 4.3, 4.4.

Pytanie: Czy MO_{SC} jest realizowalny, tzn. czy spełnia warunek $(\forall t \in T)(\exists M \in [M_0 >))(t \text{ jest przygotowane w } M)$?

Twierdzenie 4.1

Realizowalność systemu MO_{SC} jest problemem silnie NP-zupełnym.

Dowód

Oznaczmy problem realizowalności systemu MO_{SC} symbolem PR. Dane opisujące konkretny problem $I \in D_{PR}$ można opisać poprzez wyszczególnienie par $(t, p), (p, t)$ składających się na relację F, wag poszczególnych łuków $W(f)$, $f \in F$, oraz pojemności $K(p)$ i markowania początkowego $M_0(p)$ poszczególnych miejsc $p \in P$. Przyjmując, że każda para (t, p) (i analogicznie (p, t)) wraz z jej wagą zajmuje trzy słowa pamięci, a każda para $(K(p), M_0(p))$ dwa słowa, to liczba słów zużytych na opisanie problemu wynosi $n = 3 \|F\| + 2 \|P\|$. Jak łatwo sprawdzić warunek $\|P\| < \|F\| < 6 \|P\|$ jest spełniony dla każdego MO_{SC} . Stąd $5 \|P\| < n < 20 \|P\|$, a zatem liczba elementów zbioru P stanowi parametr charakterystyczny dla problemu PR, który możemy uznać za jego rozmiar $N(I) = \|P\|$.

Aby udowodnić, że PR jest silnie NP-zupełny wystarczy wykazać (patrz [18]), że:

A. PR należy do klasy NP,

B. istnieje wielomian q określony dla liczb całkowitych, dla którego PR_q jest NP-zupełny, gdzie PR_q jest podproblemem PR otrzymanym przez ograniczenie D_{PR} tylko do tych instancji, dla których maksymalna wartość występujących w tym problemie

liczb $\text{Max}(I)$ i rozmiar problemu $N(I)$ spełniają warunek $\text{Max}(I) \leq q(N(I))$.

ad. A

Biorąc pod uwagę strukturę MO_{SC} łatwo zauważyć, że warunek realizowalności (Def. 4.5) jest równoważny istnieniu permutacji elementów zbioru przejść $m = t_1 t_2 \dots t_n$ takiej, że istnieje sekwencja palen $\sigma = M_0 t_1 M_1 t_2 \dots M_{n-1} t_n M_0$, gdzie M_0 jest markowaniem początkowym MO_{SC} . Wygenerowanie permutacji przejść i sprawdzenie czy na jej podstawie można zbudować sekwencję σ może być zrealizowane przez DTM w czasie ograniczonym od góry przez wielomian pierwszego stopnia zależny od $k = \|T\| \cdot \|P\|$. Tym samym (biorąc pod uwagę, że $\|T\| < \|P\|$ dla każdego MO_{SC}) czas ten jest ograniczony wielomianem drugiego stopnia zależnym od $\|P\|$ tj. od rozmiaru problemu PR. Rozważany problem może być zatem rozwiązany przez NDTM w czasie wielomianowym i w konsekwencji $\text{PR} \in \text{NP}$.

ad. B

Parametrami liczbowymi problemu PR są $M_0(p)$, $K(p)$, $p \in P$, i $W(f)$, $f \in F$, przy czym dla każdego MO_{SC} spełniona jest zależność (patrz Def. 4.3) $\max_{p \in P} M_0(p) \leq \max_{p \in P} K(p)$.

Rozważmy podproblem PR_q , gdzie $q(N(I)) = N(I)$, tj. problem PR otrzymany przez ograniczenie D_{PR} tylko do tych instancji, dla których $\max_{p \in P} K(p) \leq \|P\|$ i $\max_{f \in F} W(f) \leq \|P\|$.

Wykażemy, że PR_q jest NP-zupełny transformując wielomianowo NP-zupełny problem spełnialności do problemu PR_q . Przyjmiemy za [120] definicję problemu spełnialności.

Problem spełnialności PS

Parametry: Rodzina zbiorów $D = \{D_1, D_2, \dots, D_n\}$, gdzie D_j jest podzbiorem literałów ze zbioru $X = \{x_1, x_2, \dots, x_m, \bar{x}_1, \bar{x}_2, \dots, \bar{x}_m\}$.

Pytanie: Czy koniunkcja K_D wyrażeń D'_j , gdzie D'_j jest alternatywą literałów należących do D_j , jest spełnialna, tzn. czy istnieje podzbiór $Y \subset X$ taki, że Y nie zawiera komplementarnej pary literałów $(x_i, \bar{x}_i)$ i $Y \cap D_j \neq \emptyset$ dla każdego $j \in \overline{1, n}$.

Mając dany konkretny problem spełnialności wyznaczmy odpowiedający mu MO_{SC} w następujący sposób.

(1) Wyznaczyć specyfikację $S = (R, G, H, \beta, \chi)$, gdzie

$$(1.1) R = \{r\},$$

$$(1.2) G = D \cup \{x_i \mid x_i \in DD \vee \bar{x}_i \in DD\}, \text{ gdzie}$$

$$DD = \bigcup_{j \in \overline{1, n}} D_j,$$

$$(1.3) H = (T \cup E, \alpha), \text{ gdzie}$$

(1.3.1) T jest sumą rozłącznych zbiorów

$$\{t_0\} \cup \{t_i^k \mid x_i \in DD \ \& \ k \in \overline{1, 3}\} \cup \\ \cup \{\bar{t}_i^k \mid \bar{x}_i \in DD \ \& \ k \in \overline{1, 3}\},$$

(1.3.2) E jest sumą rozłącznych zbiorów

$$\{e_{ij}^1 \mid x_i \in D_j\} \cup \{\bar{e}_{ij}^1 \mid \bar{x}_i \in D_j\} \cup \\ \cup \{e_i \mid x_i \in DD\} \cup \{\bar{e}_i \mid \bar{x}_i \in DD\} \cup \\ \cup \{e_{ij}^2 \mid x_i \in DD \ \& \ j \in \overline{1, n}\} \cup \\ \cup \{\bar{e}_{ij}^2 \mid \bar{x}_i \in DD \ \& \ j \in \overline{1, n}\},$$

$$(1.3.3) \alpha = (T \times E \cup E \times T) \cap \alpha^{\mathbb{K}}, \text{ gdzie}$$

$$\alpha^{\mathbb{K}} = \{(t_0, e_{ij}^1), (e_{ij}^1, t_i^1), (t_i^1, e_i),$$

$$\begin{aligned}
& (e_i, t_i^2), (t_i^2, e_{ij}^2), (e_{ij}^2, t_i^3), \\
& (\bar{t}_0, \bar{e}_{ij}^1), (\bar{e}_{ij}^1, \bar{t}_i^1), (\bar{t}_i^1, \bar{e}_i), \\
& (\bar{e}_i, \bar{t}_i^2), (\bar{t}_i^2, \bar{e}_{ij}^2), (\bar{e}_{ij}^2, \bar{t}_i^3) \mid \\
& i \in \overline{1, m} \ \& \ j \in \overline{1, n} \}
\end{aligned}$$

$$(1.4) \quad \beta(t) = r, \quad t \in T,$$

$$(1.5) \quad \gamma(e_{ij}^k) = \gamma(\bar{e}_{ij}^k) = D_j, \quad e_{ij}^k, \bar{e}_{ij}^k \in E,$$

$$\gamma(e_i) = \gamma(\bar{e}_i) = x_i, \quad e_i, \bar{e}_i \in E.$$

(2) Wyznaczyć funkcję $C : G \rightarrow N^+$, gdzie

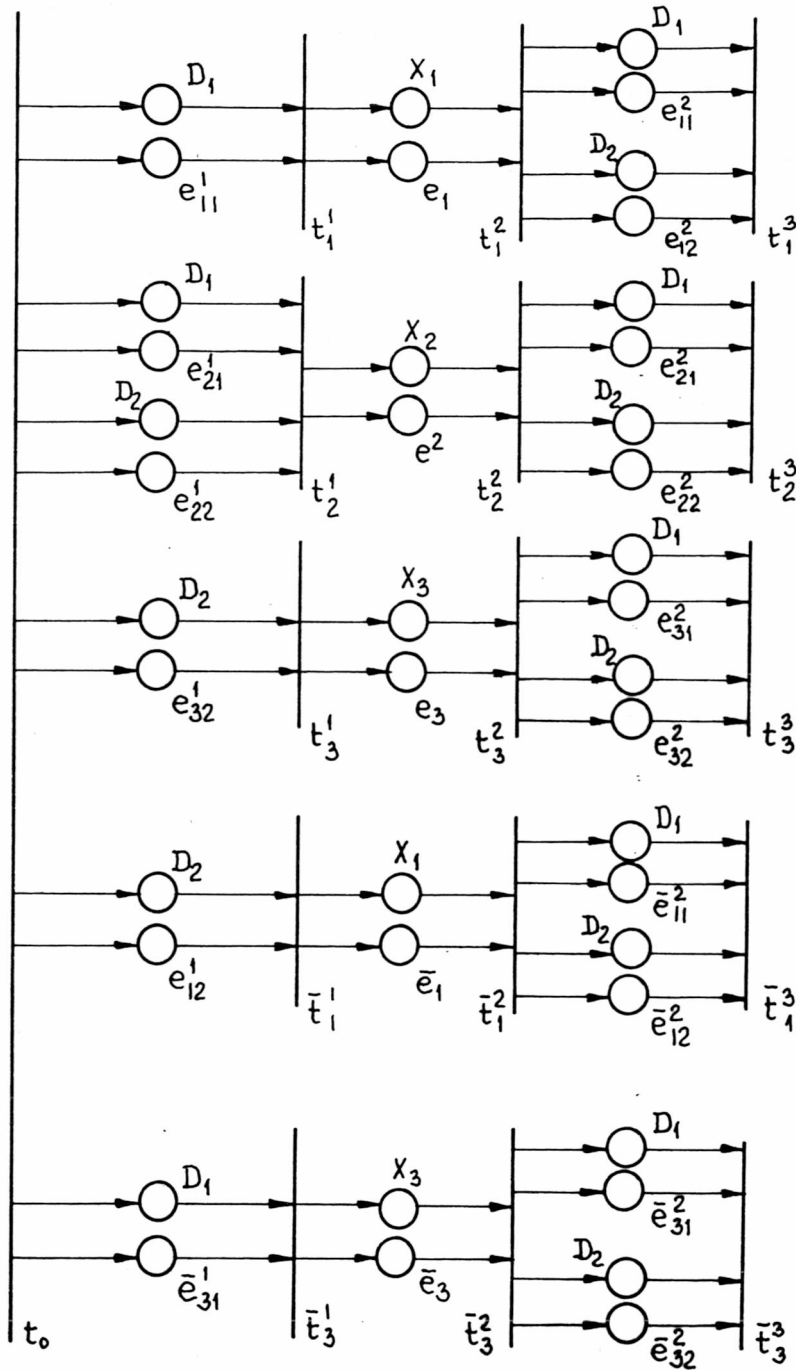
$$C(x_i) = 1, \quad x_i \in G,$$

$$C(D_j) = \|D_j\|, \quad D_j \in G.$$

(3) Wyznaczyć zgodnie z Definicją 4.3 system MO_{SC} o parametrach otrzymanych w p. 1-2.

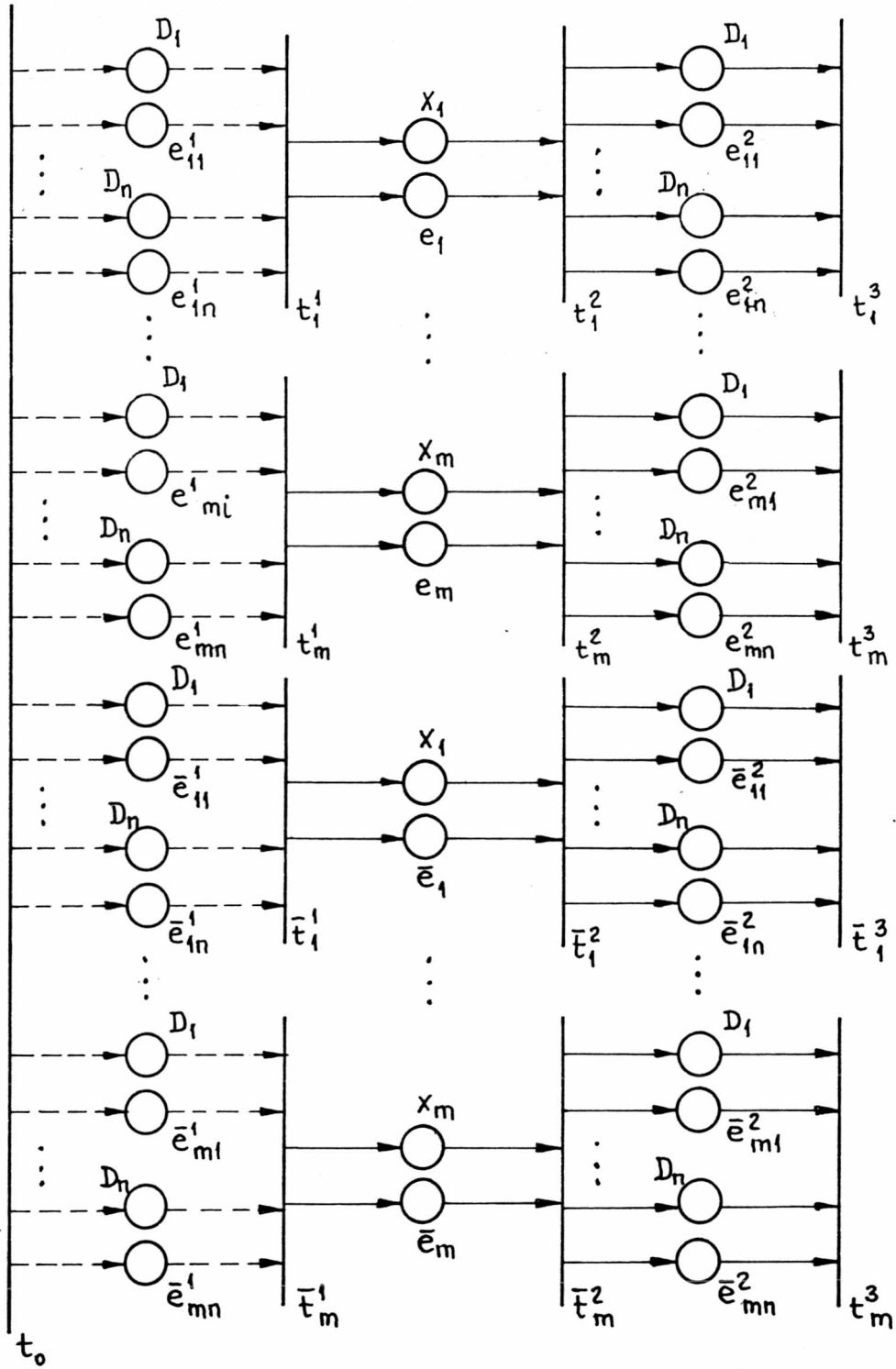
Celem graficznej ilustracji struktury modelu uzyskiwanego w wyniku opisanej transformacji przedstawiono poniżej dwa rysunki (Rys. 4.6 i 4.7), z których pierwszy stanowi konkretny przykład, natomiast drugi jest schematem ogólnym. W obu przypadkach model MO_{SC} jest pokazany w postaci uproszczonej, tzn. z pominięciem miejsc reprezentujących realizatory, które w sposób oczywisty nie mają wpływu na realizowalność modelu. W wersji ogólnej (Rys. 4.7) uwzględniono wszystkie przejścia $t_i^k, \bar{t}_i^k$, $i \in \overline{1, m}$, natomiast w przypadkach szczególnych będą występować tylko te przejścia, które odpowiadają elementom $x_i, \bar{x}_i$ należącym do co najmniej jednego ze zbiorów D_j . Ponadto, linią przerywaną oznaczono łuki $(t_0, e_{ij}^1), (e_{ij}^1, t_i^1), (t_0, \bar{e}_{ij}^1), (\bar{e}_{ij}^1, \bar{t}_i^1)$ dla podkreślenia ich opcjonalnej obecności w sieci (podobnie jak miejsc e_{ij}^1 i $\bar{e}_{ij}^1$). Wystąpienie konkretnego z tych elementów,

jak wskazuje p. 2.3 algorytmu, jest uzależnione od przynależności literału $x_i/\bar{x}_i$ do zbioru D_j .



$$C(x_1) = C(x_2) = C(x_3) = 1, \quad C(D_1) = C(D_2) = 3$$

Rys. 4.6. Uproszczony (z pominięciem miejsc reprezentujących realizatory) model MO_{SC} uzyskany w wyniku transformacji problemu spełnialności o parametrach: $D = \{D_1, D_2\}$, $X = \{x_1, x_2, x_3, \bar{x}_1, \bar{x}_2, \bar{x}_3\}$, $K_D = (x_1 \vee x_2 \vee \bar{x}_3) \& (\bar{x}_1 \vee x_3 \vee x_2)$.



Rys. 4.7. Uproszczony (z pominięciem miejsc reprezentujących realizatory) schemat modelu MO_{SC} uzyskanego w wyniku transformacji przedstawionej w dowodzie Twierdzenia 4.1.

Zauważmy, że w systemie MO_{SC} otrzymanym w wyniku przedstawionej transformacji obowiązuje:

$$\max_{f \in F} W(f) = 1 \ \& \ \|P\| > 1,$$

$$\max_{p \in P} K(p) = \max_{D_j \in D} \|D_j\| \ \& \ (\forall D_j \in D) (\|P\| > \|D_j\|).$$

$$\text{Stąd } \max_{f \in F} W(f) < \|P\| \ \text{ i } \max_{p \in P} K(p) < \|P\|,$$

a zatem dla każdej instancji $I \in D_{PS}$ transformacja ta prowadzi do instancji $I \in D_{PR_q}$.

Udowodnimy obecnie, że $PS \propto PR_q$.

Celem wykazania, że koniunkcja K_D jest spełnialna wtedy i tylko wtedy, gdy odpowiadający jej model MO_{SC} jest realizowalny zauważmy, że:

- (1) MO_{SC} jest realizowalny wtedy i tylko wtedy, jeżeli osiągalne jest markowanie $M \in [M_0]$ takie, że istnieje przejście $t_i^2 \in T$ lub $\bar{t}_i^2 \in T$, które jest przygotowane w M .
- (2) Zapalenie przejścia t_0 powoduje całkowite wypełnienie wszystkich magazynów D_j , natomiast przygotowanie przejścia t_i^2 lub $\bar{t}_i^2$ wymaga zwolnienia co najmniej jednego miejsca, w każdym magazynie D_j .
- (3) Zwolnienie miejsca w magazynie D_j wymaga zapalenia przejścia $\bar{t}_i^1$ takiego, że $\bar{x}_i \in D_j$, lub przejścia t_i^1 takiego, że $x_i \in D_j$.
- (4) Tylko jedno z przejść każdej komplementarnej pary $(t_i^1, \bar{t}_i^1)$ może zostać zapalone przed osiągnięciem markowania M .
- (5) Po zapaleniu przejścia t_0 wszystkie przejścia $t_i^1, \bar{t}_i^1$ są przygotowane. Zapalenie przejścia należącego do jednej z par komplementarnych nie wycofuje przygotowania przejść należących do innych.

Konfrontując powyższe spostrzeżenia dochodzimy do wniosku, że markowanie M jest osiągalne wtedy i tylko wtedy, jeżeli istnieje zbiór $Y \subset X$ taki, że $Y \cap D_j \neq \emptyset$ dla każdego $j \in \overline{1, n}$, który nie zawiera komplementarnej pary $(x_i, \bar{x}_i)$. A zatem $MO_{SC} = MO_{SC}(D)$ jest realizowalny wtedy i tylko wtedy, jeżeli koniunkcja K_D jest spełnialna.

Łatwo jest również wykazać, że czas konstrukcji danych konkretnego problemu $I \in D_{PR_q}$ jest ograniczony od góry przez wielomian zależny od m i n , tj. parametrów charakterystycznych dla problemu spełnialności PS. Zasadniczą częścią kroku pierwszego jest wyznaczenie grafu H (np. w postaci jego macierzy incydencji) co wymaga głównie sprawdzenia przynależności poszczególnych $x_i, \bar{x}_i$ do zbiorów D_j , gdzie $i \in \overline{1, m}$, $j \in \overline{1, n}$. W kroku drugim następuje przepisanie (lub policzenie w zależności od struktury danych problemu PS) wartości mocy zbiorów D_j , $j \in \overline{1, n}$. Wreszcie w kroku trzecim konstruowane są dane opisujące system MO_{SC} , co łączy się głównie z rozszerzeniem macierzy incydencji grafu $H = (T \cup E, \alpha)$ do macierzy incydencji sieci $(T \cup P, F)$, gdzie $P = E \cup R \cup G$, $F = \alpha \cup FG \cup FR$. Biorąc pod uwagę Def. 4.3 i strukturę danych (S, C) wyznaczonych w krokach 1-2, łatwo zauważyć, że konstrukcja ta polega na przepisaniu pewnych danych zawartych w opisie programu S , których liczba jest ograniczona wielomianem pierwszego stopnia zależnym od $k = m \cdot n$.

C.B.D.O.

Wniosek 4.1

Transformacja problemu spełnialności w problem realizowalności systemu MO_{SC} , przedstawiona w dowodzie Twierdzenia 4.1, prowadzi do systemu będącego modelem dynamiki osiągalnej poje-

dynczego procesu złożonego π opisanego specyfikacją $S_\pi = S$ oraz takiego, że $t \in T_{we}$ jest przygotowane w M_0 . Tym samym udowodni-
 liśmy, że podproblem problemu PR ograniczony do instancji speł-
 niających powyższe założenia jest silnie NP-zupełny. Podproblem
 ten będziemy dalej oznaczać symbolem PR^1 .

O ile Twierdzenie 4.1 wymagało stosunkowo złożonego dowo-
 du, o tyle już intuicyjnie prawdziwe jest twierdzenie następu-
 jące.

Twierdzenie 4.2

System MO_{SC} jest realizowalny jeżeli realizowalny jest
 system MO_{SC}' taki, że $(\forall g \in G)(C'(g) \leq C(g))$.

Dowód

Dowolna sekwencja paleń, która jest dopuszczalna w MO_{SC}
 jest również dopuszczalna w MO_{SC}' , a zatem zbiór markowań o-
 siągalnych w MO_{SC} zawiera zbiór markowań osiągalnych w MO_{SC}' .

C.B.D.O.

Realizowalność MO_{SC} jest własnością konieczną, jednakże
 nie wystarczającą dla bezblokowej pracy systemu. Niedopusz-
 czenie do występowania blokad w systemach realizowalnych wyma-
 ga ograniczenia ich dynamiki w sposób zapewniający osiągalność
 jedynie tych stanów, z których możliwy jest powrót do stanu
 początkowego.

Postępowanie możliwie mało restrykcyjne, tj. eliminacja
 wyłącznie tych działań $M[t \rangle M'$ systemu, które prowadzą do sta-
 nów niedopuszczalnych $M' \notin [M_0 \rangle$, łączy się z rozwiązywaniem w
 czasie bieżącym następującego problemu.

Definicja 4.6

Problemem powrotu w systemie MO_{SC} będziemy nazywać zagadnienie:

parametry: realizowalny system MO_{SC} , jak w Def. 4.3, 4.5; markowanie $M \in [M_0]$ takie, że $M_0 \in [M]$; przejście $t \in T$ przygotowane w M ,

pytanie: Czy markowanie początkowe M_0 jest osiągalne z markowania M' , gdzie $M[t] M'$?

Wykażemy, że problem powyższy jest NP-trudny.

Twierdzenie 4.3

Problem powrotu w systemie MO_{SC} jest silnie NP-trudny.

Dowód

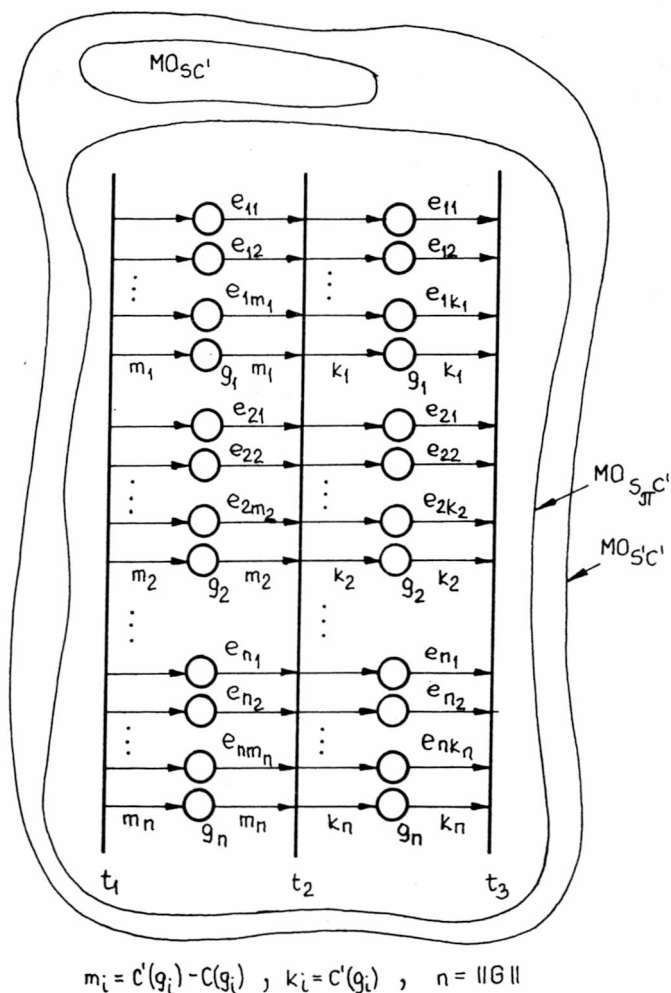
Twierdzimy, że rozważany problem jest NP-trudny, a nie NP-zupełny, ponieważ prawdopodobnie nie należy on do klasy NP. Dla słuszności twierdzenia wystarczy zatem wykazać, że istnieje problem NP-zupełny transformowalny wielomianowo do podproblemu problemu powrotu spełniającego ograniczenie $\text{Max}(I) \leq q(N(I))$ dla pewnego wielomianu q .

Podobnie jak w dowodzie Twierdzenia 4.1 przyjmujemy wartość $\|P\|$ jako rozmiar rozważanego problemu oraz zdefiniujemy wielomian q jako $q(\|P\|) = \|P\|$.

Oznaczmy problem powrotu symbolem PP. Wykażemy, że $PR_q^1 \propto PP_q$, gdzie PR_q^1 jest podproblemem problemu PR opisanym we Wniosku 4.1.

Mając dany konkretny MO_{SC} wyznaczmy konkretne dane problemu powrotu (MO_{SC}', M, t) w następujący sposób:

A. MO_{SC}' jest rozszerzeniem systemu MO_{SC} powstałym przez
(patrz Rys. 4.8):



Rys. 4.8. Ilustracja do dowodu Twierdzenia 4.3. $MO_{S\pi C'}$ przedstawiono w postaci uproszczonej, tzn. z pominięciem miejsc reprezentujących realizatory

(1) Rozszerzenie specyfikacji $S = (R, G, (T \cup E, \alpha), \beta, \gamma)$ do specyfikacji S' przez uwzględnienie dodatkowego procesu opisanego programem $S = (R_\pi, G_\pi, (T_\pi \cup E_\pi, \alpha_\pi), \beta_\pi, \gamma_\pi)$, gdzie:

$$(1.1) \quad R_\pi = \{r\}, \quad r \in R,$$

$$(1.2) \quad G_\pi = \{g_i \mid g_i \in G \ \& \ i \in \overline{1, \|G\|}\},$$

$$(1.3) \quad T = \{t_1, t_2, t_3\},$$

$$(1.4) \quad E = E_\pi^1 \cup E_\pi^2, \quad \text{gdzie}$$

$$E_\pi^1 = \{e_{ij} \mid i \in \overline{1, \|G\|} \ \& \ j \in \overline{1, C'(g_i) - C(g_i)}\},$$

$$E_{\pi}^2 = \{e_{ij} \mid i \in \overline{1, \|G\|} \text{ \& \& } j \in \overline{1, C'(g_i)}\},$$

$$(1.5) \alpha_{\pi} = \{(t_1, e), (e, t_2) \mid e \in E^1\} \cup \\ \cup \{(t_2, e), (e, t_3) \mid e \in E^2\},$$

$$(1.6) \beta_{\pi}(t) = r, \quad t \in T,$$

$$(1.7) \gamma_{\pi}(e_{ij}^k) = g_i, \quad e_{ij}^k \in E_{\pi},$$

$$(2) C'(g) = \max(C(g)+1, \|\{e \mid e \in E \text{ \& \& } \gamma(e) = g\}\|).$$

$$B. \quad M(p) = \begin{cases} 1 & \text{jeżeli } p \in R \\ 0 & \text{jeżeli } p \in E \cup E_{\pi}^2 \\ 1 & \text{jeżeli } p \in E_{\pi}^1 \\ C'(g_i) - C(g_i) & \text{jeżeli } p = g_i, i \in \overline{1, \|G\|}. \end{cases}$$

C. t jest dowolnym przejściem wejściowym MO_{SC} .

Zauważmy najpierw, że maksymalna wartość danych liczbowych wygenerowanego problemu PP wynosi $\max_{i \in \overline{1, \|G\|}} C'(g_i)$, podczas gdy jego rozmiar (moc zbioru P') jest nie mniejszy niż

$P + \max_{i \in \overline{1, \|G\|}} C'(g_i)$. Przedstawiona transformacja prowadzi zatem wyłącznie do instancji spełniających ograniczenie $\text{Max}(I) \leq \leq N(I)$.

Biorąc pod uwagę strukturę i wyznaczoną pojemność C' magazynów systemu $MDO_{S'C'}$, łatwo również zauważyć, że wygenerowane dane spełniają założenia problemu PP (tzn. $MO_{S'C'}$ jest realizowalny, $M \in [M_0]$, $M_0 \in [M]$, t jest przygotowane w M) ponieważ:

- (a) dowolne przejście $t \in T/t \in T_{\pi}$ może, wychodząc ze stanu początkowego, zostać przygotowane poprzez zgodne z porządkiem $\alpha^{\#}/\alpha_{\pi}^{\#}$ (gdzie $\alpha^{\#}/\alpha_{\pi}^{\#}$ - tranzytywne domknięcie α/α_{π}) zapalenie wszystkich przejść poprzedzających t , a zatem skonstruowany system $MO_{S'C'}$ jest realizowalny;

- (b) markowanie M jest osiągalne w wyniku $M_0 \xrightarrow{t_1} M$,
- (c) markowanie M_0 jest osiągalne w wyniku $M \xrightarrow{t_2, t_3} M_0$,
- (d) $t \in T_{we}$, a zatem jest przygotowane w markowaniu początkowym systemu MO_{SC} . Ponieważ $(\forall g \in G)(M(g) = C'(g) - C(g))$, liczba wolnych miejsc w magazynach jest taka sama, jak w przypadku markowania początkowego M_0 systemu MO_{SC} . Stąd t jest również przygotowane w M .

Wykażemy obecnie, że przedstawiona transformacja posiada własności transformacji wielomianowej $PR^1 \propto PP_q$. W tym celu zauważmy, że:

- (e) zapalenie przejścia t powoduje zmianę markowania $M \xrightarrow{t} M'$, $M' \neq M$;
- (f) powrót do M_0 z M' wymaga zapalenia wszystkich przejść $t \in T$ oraz przejść $t_2, t_3 \in T_{\pi}$;
- (g) t_2 jest przygotowane w M i nie jest przygotowane w żadnym innym markowaniu niż M ; t_3 jest przygotowane po zapaleniu przejścia t_2 .

A zatem, powrót do M_0 z M' może nastąpić wtedy i tylko wtedy, jeżeli najpierw zostaną zapalone wszystkie przejścia $t \in T$ (co pozwala ponownie osiągnąć markowanie M), a następnie t_2, t_3 . Ponieważ tak długo, dopóki nie zostanie zapalone przejście t_2 , przedmioty $e \in E_{\pi}^1$ zajmują $C'(g_i) - C(g_i)$ miejsc w każdym magazynie g_i , liczba dostępnych miejsc w magazynach $= C(g_i)$, tj. tak jak w przypadku ich realizacji w systemie MO_{SC} . Wykonanie wszystkich operacji $t \in T$, i w konsekwencji powrót do M_0 , może zatem nastąpić wtedy i tylko wtedy, jeżeli MO_{SC} jest realizowalny.

Zasadniczą częścią przedstawionej transformacji jest wyznaczenie pojemności magazynów $C'(g_i)$ i wygenerowanie zbioru E_{π} .

Biorąc pod uwagę odpowiednie wzory oraz fakt, że

$\max_{i \in 1, \|G\|} C(g_i) \leq \|P\|$ łatwo zauważyć, że obie te operacje mogą zostać wykonane w czasie ograniczonym od góry przez wielomian drugiego stopnia zależny od $\|P\|$ tj. od rozmiaru problemu PR^1 .

C.B.D.O.

Dążąc do minimalnie restrykcyjnego modelu WZPP, tj. znajdującego zastosowanie we wszystkich ESP, których pojemność magazynów jest wystarczająca dla realizacji procesów o specyfikacji S , oraz ujmującego wszystkie dopuszczalne przebiegi procesów, natknęliśmy się na problemy NP-zupełne. Ponieważ jednak, trudno oczekiwać aby w codziennej praktyce decyzje kwalifikujące wykonalność zaplanowanego dla danego ESP zadania i dopuszczalność wykonania określonej operacji w bieżącym stanie systemu, były podejmowane w oparciu o algorytmy o tak wysokiej złożoności, wydaje się celowe przyjęcie rozwiązania alternatywnego.

Rozwiązanie to, jak wspomniano wcześniej, polega na opracowaniu algorytmów podejmowania decyzji w sposób bardziej restrykcyjny (tj. takich, w których dopuszcza się uzyskanie odpowiedzi "nie" w przypadku pewnych konkretnych problemów realizowalności i powrotu, których właściwym rozwiązaniem jest odpowiedź "tak"), a w zamian za to obniżeniu ich złożoności obliczeniowej. Innymi słowy decyzje o tym czy system MO_{SC} jest realizowalny, bądź czy zapalenie pewnego przejścia t w pewnym markowaniu M jest bezpieczne, będą podejmowane w oparciu o warunki dostateczne na to, aby MO_{SC} był realizowalny oraz stan początkowy M_0 był osiągalny ze stanu M' , gdzie $M[t \rangle M'$.

Należy zauważyć, że ograniczenie możliwości palenia przejść w sposób bardziej restrykcyjny niż jest to konieczne dla uniknięcia blokady powoduje "wycięcie" z grafu markowań osiągalnych

również pewnych ścieżek poprawnych, tj. prowadzących od pewnego markowania osiągalnego do markowania początkowego. Warunek ograniczający przygotowanie przejść musi zatem zostać tak skonstruowany, aby nie istniała możliwość wystąpienia blokady wtórnej, tj. stanu, który nie jest blokadą w systemie MO_{SC} , jest nią natomiast w systemie rozszerzonym o dodatkowy warunek przygotowania przejść, ze względu na "wycięcie" wszystkich ścieżek, po których mógłby nastąpić powrót do M_0 . W systemie rozszerzonym nie będą występować ani blokady pierwotne ani wtórne jeżeli zapewnimy, że będzie on prawidłowy w sensie poniższej definicji.

Definicja 4.7

Dowolny (w tym również rozszerzony) system P/T będziemy nazywać:

realizowalnym jeżeli $(\forall t \in T)(\exists M \in [M_0 >))(t \text{ jest przygotowane w } M)$,

odwracalnym jeżeli $(\forall M \in [M_0 >))(M_0 \in [M >))$,

prawidłowym jeżeli jest realizowalny i odwracalny.

Zauważmy, że własność prawidłowości jest równoważna koniunkcji standardowych własności systemów P/T - żywości i odwracalności. Wprowadzona dla celów niniejszej pracy cecha realizowalności wynika z tego, że jest ona opisana warunkiem prostszym niż żywość systemu.

Proponowana tu koncepcja przeciwdziałania blokadom w systemach WZPP polega na:

- (1) wyznaczeniu dolnego ograniczenia pojemności magazynów (funkcji minimalnej pojemności magazynów $\xi(g)$) zależnego od struktury relacji F systemu MO_{SC} ,
- (2) rozszerzeniu relacji incydencji F systemu MO_{SC} o relację $D \subseteq P \times T$ zależną od struktury relacji F ,

- (3) rozszerzeniu formuły przygotowania przejść obowiązującej w systemie MO_{SC} o formułę, której wartość logiczna dla danego $t \in T$ zależy od markowania miejsc związanych z t rozszerzoną relacją incydencji.

W rozdziale następnym przedstawimy konstrukcję wymienionych wyżej elementów oraz wykazemy, że zbudowany w ten sposób system MD_{SC} , tzn. system MO_{SC} rozszerzony o relację D i dodatkową formułę przygotowania przejść oraz spełniający warunek $(\forall g \in G)(K(g) \supseteq \mathcal{E}(g))$, jest prawidłowy.

Metoda, która zostanie przedstawiona jest rozszerzeniem wyników uzyskanych w pracy [158] o elementy koncepcji zaproponowanej w [14]. Rozszerzenie to pozwoliło, w stosunku do wyników pracy [158], na zmniejszenie poziomu restrykcji zarówno dolnej pojemności magazynów (co zwiększa zakres określoności metody), jak i formuły ograniczającej przygotowanie przejść (co zwiększa elastyczność działania modelu).

4.5. Model dynamiki dopuszczalnej

Rozważania dalszej części tego rozdziału ograniczymy do podklasy systemów MO_{SC} wyznaczonej zbiorem par (S, C) takich, że funkcja C będzie ograniczona od dołu funkcją minimalnej pojemności magazynów $\mathcal{E}$, zależną od struktury specyfikacji. Celem prostego sformułowania powyższej zależności wprowadzimy wcześniej następujące definicje.

Definicja 4.8

Zbiorem węzłów w specyfikacji S będziemy nazywać podzbiór $T^V \subseteq T$ złożony ze wszystkich operacji $t \in T$, które posiadają

więcej niż jeden poprzednik w relacji α (przedmiot wejściowy) lub więcej niż jeden następnik w relacji α (przedmiot wyjściowy). Pozostałe operacje $t \in T \setminus T^V$ będą stanowić zbiór operacji prostych $T^R = T \setminus T^V$.

Definicja 4.9

Zbiorem magazynów węzłowych w specyfikacji S będziemy nazywać podzbiór $G^V \subseteq G$ złożony ze wszystkich magazynów $g \in G$ takich, że istnieje przedmiot $e \in \{e \mid \gamma(e) = g\}$ należący do zbioru przedmiotów wejściowych lub wyjściowych dowolnego węzła $t \in T^V$.

Zbiorem magazynów przejściowych w specyfikacji S będziemy nazywać podzbiór $G^R \subseteq G$ złożony ze wszystkich magazynów $g \in G$ takich, że istnieje przedmiot $e \in \{e \mid \gamma(e) = g\}$, który nie należy do zbioru przedmiotów wejściowych lub zbioru przedmiotów wyjściowych żadnego z węzłów.

Definicja 4.10

Magazyn $g \in G$ w specyfikacji S będziemy nazywać niekonfliktowym, jeżeli zbiór przedmiotów $E_g = \{e \mid \gamma(e) = g\}$ jest jednoelementowy lub może być uporządkowany w sposób $E_g = \{e_1, e_2, \dots, e_n\}$ taki, że dla każdego $i \in \overline{1, n-1}$ istnieje przejście $t \in T$ takie, że $e_i \cap e_{i+1} = \{t\}$. W przeciwnym przypadku magazyn $g \in G$ będziemy nazywać konfliktowym. Zbiory magazynów konfliktowych i niekonfliktowych będziemy oznaczać odpowiednio przez G^K i G^N .

Nazwy wprowadzone w powyższych definicjach będziemy również stosować do miejsc i przejść systemu MO_{SC} odpowiadających wymienionym magazynom i operacjom.

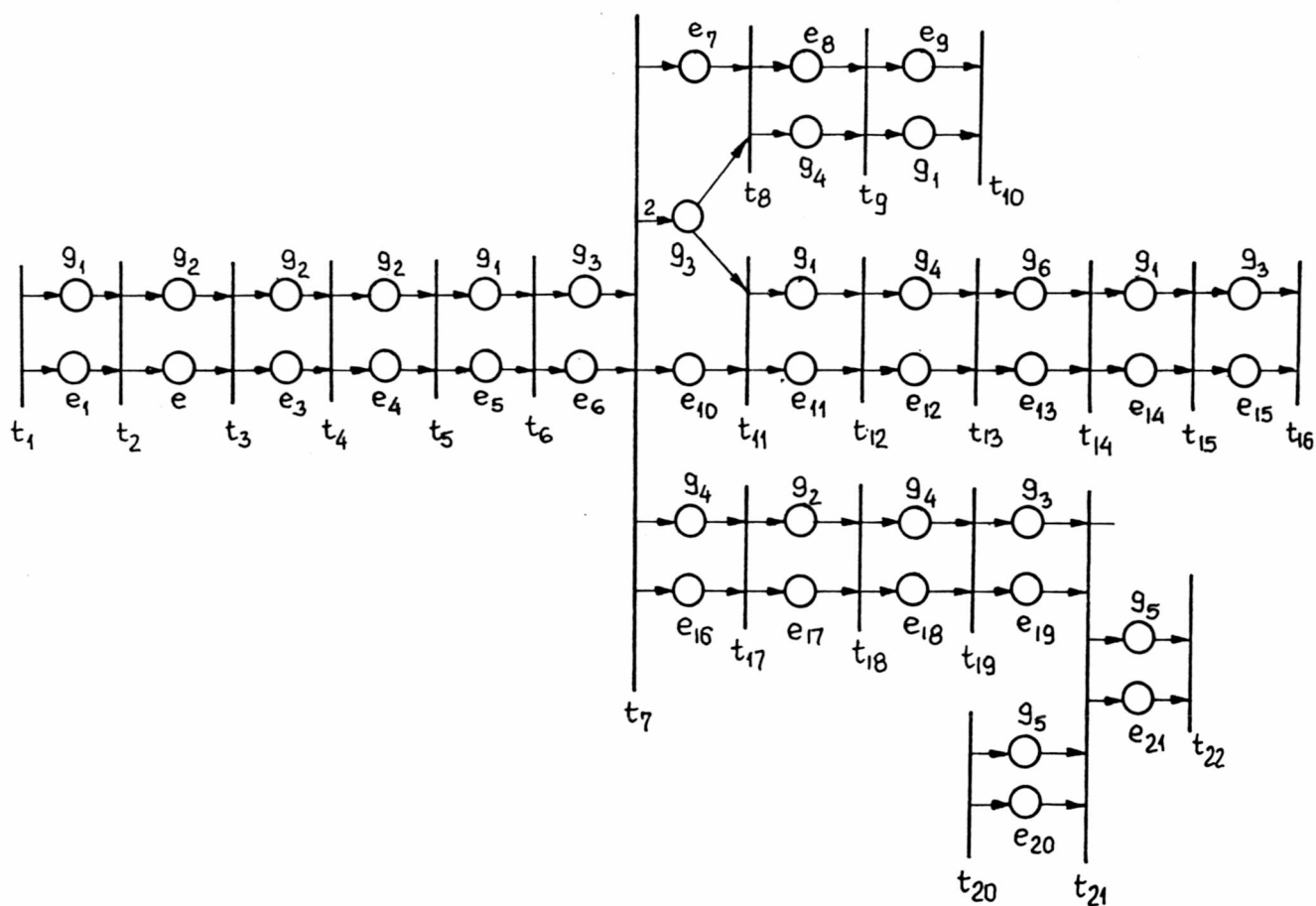
Przykład 4.5

Dla systemu MO_{SC} przedstawionego w postaci uproszczonej

(tzn. z pominięciem miejsc reprezentujących realizatory, które w przykładzie przedstawionym są nieistotne) na Rys. 4.9 wyznaczmy zbiory T^V , T^R , G^V , G^R , G^K , G^N .

$$\begin{aligned} T^V &= \{t_7, t_{21}\} & T^R &= \{t_1, t_2, \dots, t_6, t_8, t_9, \dots, t_{20}, t_{22}\}, \\ G^V &= \{g_3, g_4, g_5\}, & G^R &= \{g_1, g_2, g_3, g_4, g_6\}, \\ G^N &= \{g_2, g_5, g_6\}, & G^K &= \{g_1, g_3, g_4\}. \end{aligned}$$

Wykorzystując wprowadzone wyżej pojęcia zdefiniujemy funkcję.



Rys. 4.9. System MO_{SC} ilustrujący Przykład 4.3.

Definicja 4.11

Minimalną pojemnością magazynów systemu MO_{SC} będziemy nazywać funkcję $\xi: G \rightarrow \mathbb{N}^+$ taką, że $\xi(g) = \xi'(g) + \xi''(g)$, gdzie

$$\mathcal{E}'(g) = \begin{cases} 1 & \text{jeżeli } g \in G \setminus (G^V \cap G^K) \\ \sum_{t \in T^V} W(g, t) & \text{jeżeli } g \in (G^V \cap G^K) \setminus G^R \\ (\sum_{t \in T^V} W(t, g)) + 1 & \text{jeżeli } g \in G^V \cap G^K \cap G^R, \end{cases}$$

$$\mathcal{E}''(g) = \begin{cases} 0 & \text{jeżeli } g \in G \setminus (G^V \cap G^K) \\ \max_{t \in T^V} W(t, g) & \text{jeżeli } g \in G^V \cap G^K. \end{cases}$$

Przykład 4.6

Dla systemu MO_{SC} omawianego w Przykładzie 4.3 wyznaczmy funkcję minimalnej pojemności magazynów $\mathcal{E}$.

$$G^V \cap G^K = \{g_3, g_4\}, \quad G \setminus (G^V \cap G^K) = \{g_1, g_2, g_5, g_6\},$$

$$(G^V \cap G^K) \setminus G^R = \emptyset,$$

$$G^V \cap G^K \cap G^R = \{g_3, g_4\},$$

$$\sum_{t \in T^V} W(g_3, t) = W(g_3, t_7) + W(g_3, t_{21}) = 2,$$

$$\sum_{t \in T^V} W(g_4, t) = W(g_4, t_7) + W(g_4, t_{21}) = 0,$$

$$\max_{t \in T^V} W(t, g_3) = \max_{t \in T^V} (W(t_7, g_3), W(t_{21}, g_3)) = \max(2, 0) = 2,$$

$$\max_{t \in T^V} W(t, g_4) = \max_{t \in T^V} (W(t_7, g_4), W(t_{21}, g_4)) = \max(1, 0) = 1.$$

Stąd:

$$\mathcal{E}(g_1) = \mathcal{E}(g_2) = \mathcal{E}(g_5) = \mathcal{E}(g_6) = 1,$$

$$\mathcal{E}(g_3) = 2 + 1 + 2 = 5,$$

$$\mathcal{E}(g_4) = 0 + 1 + 1 = 2.$$

Twierdzenie poniższe formułuje warunek dostateczny realizowalności systemu MO_{SC} .

Twierdzenie 4.4

Dowolny system MO_{SC} taki, że $K(g) \geq \mathcal{E}(g)$ dla każdego $g \in G$, jest realizowalny.

Prawdziwość powyższego twierdzenia zostanie wykazana w dalszej części pracy, udowodnimy bowiem realizowalność systemów MD_{SC} (systemów MO_{SC} rozszerzonych o łuki wzbraniające), która to własność będzie bezpośrednio implikować realizowalność systemów MO_{SC} spełniających warunek Twierdzenia 4.4.

Przedstawiona dalej metoda unikania blokad wykorzystuje koncepcję polegającą na rozszerzeniu formuły przygotowania przejść o warunki uwzględniające nie tylko markowanie miejsc incydentnych do danego przejścia t , ale również stan miejsc należących do pewnego lokalnego sąsiedztwa t , którego dobór zależy będzie od rozkładów węzłów oraz magazynów niekonfliktowych w rozważanym systemie MO_{SC} .

Definicja 4.12

Lokalnym sąsiedztwem operacji $t \in T$ należącej do specyfikacji S będziemy nazywać rodzinę Z_t złożoną ze wszystkich zbiorów par

$$Z = \{(e_1, g_1), (e_2, g_2), \dots, (e_{\|Z\|}, g_{\|Z\|})\}, \quad e_i \in E, \quad g_i \in G,$$

takich, że:

- (1) $(t, e_1) \in \alpha$,
- (2) $(\exists t \in T^r)((e_i, t) \in \alpha \ \& \ (t, e_{i+1}) \in \alpha)$, dla każdego $i \in \overline{1, \|Z\|-1}$,
- (3) $g_i = \gamma(e_i)$, dla każdego $i \in \overline{1, \|Z\|}$,
- (4) $g_i \in G^K$, dla każdego $i \in \overline{1, \|Z\|-1}$,
- (5a) $e_{\|Z\|}$ jest przedmiotem wejściowym operacji węzłowej i $g_{\|Z\|} \in G^K$, lub
- (5b) $e_{\|Z\|}$ jest przedmiotem wyjściowym operacji prostej, która nie posiada następników w grafie H , i $\|Z\| > 1$, lub
- (5c) $g_{\|Z\|} \in G^N$ i $\|Z\| > 1$.

Zbiory Z należące do sąsiedztwa Z_t będziemy nazywać strefami wzbronień operacji t (lub w odniesieniu do systemu MO_{SC} strefami wzbronień przejścia t). W zależności od tego, który z warunków 5 jest spełniony, strefę Z będziemy zaliczać do klasy:

$$Z \in \begin{cases} Z^V, & \text{jeżeli spełniony jest warunek (5a)} \\ Z^R, & \text{w przeciwnym przypadku.} \end{cases}$$

Interpretując Definicję 4.12 w sposób nieformalny, sąsiedztwo lokalne przejścia t jest rodziną zbiorów, z których każdy jest związany z jedną z "gałęzi" sieci "wychodzących" z t i zawiera pary $(e, g) \in E \times G$ "leżące" pomiędzy przejściem t , a najbliższym miejscem $g \in G^N$ (jeżeli takie występuje w danej gałęzi) lub najbliższym węzłem, względnie przejściem wyjściowym (jeżeli gałąź zawierająca t jest wyjściową gałęzią sieci). Stąd sąsiedztwo przejść prostych jest rodziną co najwyżej jednoelementową (w szczególności może to być zbiór pusty, jeżeli $t \cdot \cap G^K = \emptyset$ lub przejście leżące "na prawo" od t jest przejściem wyjściowym sieci), natomiast sąsiedztwo węzłów zawiera nie więcej stref niż liczba miejsc $e \in E$ incydentnych wyjściowo do danego węzła.

Przykład 4.7

Wyznamy sąsiedztwo lokalne przejść należących do systemu MO_{SC} omawianego w Przykładach 4.5 i 4.6.

$$Z_{t_1} = \{Z'_{t_1}\}, \quad Z'_{t_1} = \{(e_1, g_1), (e_2, g_2)\}, \quad Z'_{t_1} \in Z^R,$$

$$Z_{t_2} = \emptyset, \quad Z_{t_3} = \emptyset, \quad Z_{t_4} = \emptyset,$$

$$Z_{t_5} = \{Z'_{t_5}\}, \quad Z'_{t_5} = \{(e_5, g_1), (e_6, g_3)\}, \quad Z'_{t_5} \in Z^V,$$

$$Z_{t_6} = \{Z'_{t_6}\}, \quad Z'_{t_6} = \{(e_6, g_3)\}, \quad Z'_{t_6} \in Z^V,$$

$$Z_{t_7} = \{Z'_{t_7}, Z''_{t_7}, Z'''_{t_7}\},$$

$$Z'_{t_7} = \{(e_7, g_3), (e_8, g_4), (e_9, g_1)\}, \quad Z'_{t_7} \in Z^R,$$

$$Z_{t_7}' = \{(e_{10}, g_3), (e_{11}, g_1), (e_{12}, g_4), (e_{13}, g_6)\}, \quad Z_{t_7}' \in Z^r,$$

$$Z_{t_7}'' = \{(e_{16}, g_4), (e_{17}, g_1), (e_{18}, g_4), (e_{19}, g_3)\}, \quad Z_{t_7}'' \in Z^v,$$

$$Z_{t_8} = \{Z_{t_8}'\}, \quad Z_{t_8}' = \{(e_8, g_4), (e_9, g_1)\}, \quad Z_{t_8}' \in Z^r,$$

$$Z_{t_9} = \emptyset, \quad Z_{t_{10}} = \emptyset,$$

$$Z_{t_{11}} = \{Z_{t_{11}}'\}, \quad Z_{t_{11}}' = \{(e_{11}, g_1), (e_{12}, g_4), (e_{13}, g_6)\}, \quad Z_{t_{11}}' \in Z^r,$$

$$Z_{t_{12}} = Z_{t_{12}}', \quad Z_{t_{12}}' = \{(e_{12}, g_4), (e_{13}, g_6)\}, \quad Z_{t_{12}}' \in Z^r,$$

$$Z_{t_{13}} = \emptyset,$$

$$Z_{t_{14}} = \{Z_{t_{14}}'\}, \quad Z_{t_{14}}' = \{(e_{14}, g_1), (e_{15}, g_3)\}, \quad Z_{t_{14}}' \in Z^r,$$

$$Z_{t_{15}} = \emptyset, \quad Z_{t_{16}} = \emptyset,$$

$$Z_{t_{17}} = \{Z_{t_{17}}'\}, \quad Z_{t_{17}}' = \{(e_{17}, g_1), (e_{18}, g_4), (e_{19}, g_3)\}, \quad Z_{t_{17}}' \in Z^v,$$

$$Z_{t_{18}} = \{Z_{t_{18}}'\}, \quad Z_{t_{18}}' = \{(e_{18}, g_4), (e_{19}, g_3)\}, \quad Z_{t_{18}}' \in Z^v,$$

$$Z_{t_{19}} = \{Z_{t_{19}}'\}, \quad Z_{t_{19}}' = \{(e_{19}, g_3)\}, \quad Z_{t_{19}}' \in Z^v,$$

$$Z_{t_{20}} = \{Z_{t_{20}}'\}, \quad Z_{t_{20}}' = \{(e_{20}, g_5)\}, \quad Z_{t_{20}}' \in Z^v,$$

$$Z_{t_{21}} = Z_{t_{22}} = \emptyset.$$

Wykorzystując wcześniej wprowadzone definicje przedstawimy obecnie poszukiwany model dynamiki dopuszczalnej systemu WZPP.

Definicja 4.13

Modelem dynamiki dopuszczalnej systemu WZPP będziemy nazywać system P/T z łukami wzbraniającymi $MD_{SC} = (MO_{SC}, D)$ określony dla dowolnego MO_{SC} takiego, że $(\forall g \in G) (K(g) \geq \xi(g))$, gdzie $\xi(g)$ jak w Definicji 4.11, $D = D^I \cup D^{II} \cup D^{III} \cup D^{IV}$, $D^I, D^{IV} \subseteq E \times T$, $D^{II}, D^{IV} \subseteq G \times T$, jest relacją wzbraniającą taką, że:

$$(1) \quad (e, t) \in D^I \Leftrightarrow (\exists Z \in Z_t \cap Z^r) (\exists i \in \overline{1, \|Z\| - 1}) (e_i = e) \vee \\ \vee (\exists Z \in Z_t \cap Z^v) (\exists i \in \overline{1, \|Z\|}) (e_i = e),$$

$$(2) \quad (g, t) \in D^{II} \Leftrightarrow (\exists Z \in \mathbf{Z}_t)(\exists i \in \overline{2, \|Z\|})(g_i = g \ \& \ g_i \neq g_1),$$

$$(3) \quad (g, t) \in D^{III} \Leftrightarrow (\exists Z \in \mathbf{Z}_t)(\exists i, j \in \overline{2, \|Z\|})(g = g_1 \ \& \ g = g_i \ \& \ g \neq g_j \ \& \ j < i),$$

$$(4) \quad (e, t) \in D^{IV} \Leftrightarrow (\exists Z \in \mathbf{Z}_t)(\exists i \in \overline{1, \|Z\|})(g_i \in G^V \cap G^K \ \& \ e \in E_{g_i}^V),$$

gdzie

$$E_g^V = \{e \mid e \in E \ \& \ \gamma(e) = g \ \& \ (\exists t \in T^V)(e, t) \in \alpha\}.$$

Wprowadzona wyżej relacja D pozwala przyporządkować każdemu z przejść $t \in T$ zbiór miejsc $\{p \mid (p, t) \in D\}$, które wraz z jego miejscami wejściowymi i wyjściowymi będą decydować o przygotowaniu t . W podejmowaniu tych decyzji zostanie wykorzystana funkcja nazwana stanem istotnym magazynów i zdefiniowana jak poniżej.

Definicja 4.14

W systemie MO_{SC}/MD_{SC} stanem istotnym magazynów będziemy nazywać funkcję

$$\omega(M, g) = \begin{cases} M(g) & \text{jeżeli } g \in G \setminus (G^V \cap G^K) \\ M(g) - \sum_{e \in E_g^V} a_M(e) & \text{jeżeli } g \in G^V \cap G^K, \end{cases}$$

gdzie

E_g^V jak w p. (4) Definicji 4.13,

$$a_M = \begin{cases} 0 & \text{jeżeli } M(e) = 0 \\ 1 & \text{jeżeli } M(e) > 0 \end{cases}$$

Następnym stanem istotnym magazynów będziemy nazywać funkcję częściową Ω określoną dla dowolnej trójki (t, M, g) takiej, że t jest przygotowane w M w sensie Definicji 4.4, przez

$$\Omega(t, M, g) = \omega(M', g), \quad \text{gdzie } M[t] > M'.$$

Definicja powyższa pozwoli nam zdefiniować formułę przygotowania przejść w systemie MD_{SC} .

Definicja 4.15

W systemie MD_{SC} przejście $t \in T$ jest przygotowane w markowaniu M , jeżeli t jest przygotowane w M w sensie Definicji 4.4 i spełniony jest co najmniej jeden z niżej podanych warunków:

- (1) $(\forall g \in t^{\bullet})(\Omega(t, M, g) \leq K(g) - \mathcal{E}(g))$,
- (2) $(\forall e \in \dot{t})(M(e) = 0) \& (\forall g \in \ddot{t})(\Omega(t, M, g) \leq K(g) - \mathcal{E}'(g)) \& (\forall g \in \ddot{t} \setminus \ddot{\ddot{t}})(\Omega(t, M, g) \leq K(g) - \mathcal{E}'(g) + 1)$,

gdzie

Ω jak w Definicji 4.14,

$\mathcal{E}, \mathcal{E}'$ jak w Definicji 4.11,

$\dot{t} = \{e | (e, t) \in D^I\}$, $\ddot{t} = \{g | (g, t) \in D^{II}\}$, $\ddot{\ddot{t}} = \{g | (g, t) \in D^{III}\}$.

Warunki (1 - 2) przedstawione w powyższej definicji, wykorzystujące skonstruowane wcześniej funkcje $\mathcal{E}, \mathcal{E}'$ i Ω , stanowiąc będą test dopuszczalności markowania M' osiągalnego w systemie MO_{SC} w wyniku operacji $M[t \rangle M'$. Test ten odrzuca wszystkie markowania M' takie, że $M_0 \notin [M' \rangle$ (tj. nie dopuszcza do przygotowania przejścia t w M), przy czym jak wspomniano wcześniej, koniecznym kompromisem na rzecz niskiej złożoności testu jest fakt, że może on odrzucić również pewną część dopuszczalnych markowań M' tj. takich, że $M_0 \in [M' \rangle$. Zauważmy jednak, że z warunków (1 - 2) wynika, że względna liczba stanów dopuszczalnych, które mogą być potencjalnie pominięte jest tym mniejsza im większa jest pojemność magazynów.

Przykład 4.8 przedstawiony poniżej, ilustruje sposób wyznaczania zbiorów E_g , $g \in G^V \cap G^K$, od markowania których zależy wartość funkcji ω , oraz relacji D dla systemu $MD_{SC} = (MO_{SC}, D)$,

gdzie MO_{SC} jest systemem omawianym w Przykładach 4.5, 4.6, 4.7. Kolejny przykład (Przykład 4.9) ilustruje w sposób całościowy transformację $MO_{SC} \rightarrow MD_{SC}$ w oparciu o system MO_{SC} skonstruowany w Przykładach 4.1, 4.3.

Przykład 4.8

Wyznamy zbiory E_g , $g \in G^V \cap G^K$ oraz relację D dla systemu $MD_{SC} = (MO_{SC}, D)$, gdzie MO_{SC} jest systemem omawianym w przykładach 4.5, 4.6, 4.7.

$$1. G^V \cap G^K = \{g_3, g_4\},$$

$$E_{g_3} = \{e_6, e_{19}\}, \quad E_{g_4} = \emptyset.$$

$$2. D^I = \{(e_1, t_1), (e_5, t_5), (e_6, t_5), (e_6, t_6), (e_7, t_7), (e_8, t_7), \\ (e_{10}, t_7), (e_{11}, t_7), (e_{12}, t_7), (e_{16}, t_7), (e_{17}, t_7), \\ (e_{18}, t_7), (e_{19}, t_7), (e_8, t_8), (e_{11}, t_{11}), (e_{12}, t_{11}), \\ (e_{12}, t_{12}), (e_{14}, t_{14}), (e_{17}, t_{17}), (e_{18}, t_{17}), (e_{19}, t_{17}), \\ (e_{18}, t_{18}), (e_{19}, t_{18}), (e_{19}, t_{19}), (e_{20}, t_{20})\},$$

$$D^{II} = \{(g_2, t_1), (g_3, t_5), (g_4, t_7), (g_1, t_7), (g_6, t_7), (g_3, t_7), \\ (g_1, t_8), (g_4, t_{11}), (g_6, t_{11}), (g_6, t_{12}), (g_3, t_{14}), \\ (g_4, t_{17}), (g_3, t_{17}), (g_3, t_{18})\},$$

$$D^{III} = \{(g_4, t_7)\},$$

$$D^{IV} = \{(e_6, t_5), (e_{19}, t_5), (e_6, t_7), (e_{19}, t_7), (e_6, t_{14}), (e_{19}, t_{14}), \\ (e_6, t_{17}), (e_{19}, t_{17}), (e_6, t_{18}), (e_{19}, t_{18})\}.$$

Przykład 4.9

Dla systemu MO_{SC} skonstruowanego w Przykładach 4.1, 4.3 i przedstawionego na Rys. 4.3 przeprowadzimy transformację

$$MO_{SC} \rightarrow MD_{SC}.$$

1. Zbiory T^V , T^R , G^V , G^R , G^K , G^N mają następującą postać:

$$T = \{t_1, t_2, \dots, t_{21}\}, \quad T^V = \{t_3, t_9, t_{17}, t_{19}\}, \quad T^R = T \setminus T^V, \\ G = G^V = G^K = \{g_1, g_2, g_3, g_4, g_5\}, \quad G^R = \{g_4, g_5\}, \quad G^N = \emptyset.$$

2. Funkcje $\mathcal{E}'$, $\mathcal{E}''$, $\mathcal{E}$ mają postać:

$$\begin{aligned} \mathcal{E}'(g_1) &= 3, & \mathcal{E}''(g_1) &= 1, & \mathcal{E}(g_1) &= 4 \\ \mathcal{E}'(g_2) &= 2, & \mathcal{E}''(g_2) &= 1, & \mathcal{E}(g_2) &= 3 \\ \mathcal{E}'(g_3) &= 2, & \mathcal{E}''(g_3) &= 1, & \mathcal{E}(g_3) &= 3 \\ \mathcal{E}'(g_4) &= 1 + 1 = 2, & \mathcal{E}''(g_4) &= 1, & \mathcal{E}(g_4) &= 3 \\ \mathcal{E}'(g_5) &= 1 + 1 = 2, & \mathcal{E}''(g_5) &= 1, & \mathcal{E}(g_5) &= 3 \end{aligned}$$

3. Sąsiedztwa lokalne przejść stanowią rodziny:

$$\begin{aligned} Z_1 &= \{\{(e_1, g_1)\}\}, & Z_2 &= \{\{(e_2, g_3)\}\}, \\ Z_3 &= \{\{(e_4, g_1), (e_8, g_2)\}, \{(e_3, g_4), (e_5, g_4), (e_6, g_5), (e_7, g_5)\}\}, \\ Z_4 &= \{\{(e_8, g_2)\}\}, & Z_5 &= \{\{(e_5, g_4), (e_6, g_5), (e_7, g_5)\}\}, \\ Z_6 &= \{\{(e_6, g_5), (e_7, g_5)\}\}, & Z_7 &= \{\{(e_7, g_5)\}\}, \\ Z_8 &= \{\{(e_9, g_2)\}\}, & Z_9 &= Z_{10} = Z_{11} = Z_{12} = \emptyset, \\ Z_{13} &= \{\{(e_{13}, g_5), (e_{14}, g_5), (e_{15}, g_4)\}\}, \\ Z_{14} &= \{\{(e_{14}, g_5), (e_{15}, g_4)\}\}, & Z_{15} &= \{\{(e_{15}, g_4)\}\}, \\ Z_{16} &= \{\{(e_{16}, g_1)\}\}, & Z_{17} &= \{\{(e_{18}, g_1)\}, \{(e_{17}, g_4), (e_{13}, g_3)\}\}, \\ Z_{18} &= \{\{(e_{19}, g_3)\}\}, & Z_{19} &= Z_{20} = Z_{21} = \emptyset, \\ Z^R &= \emptyset, & Z^V &= \bigcup_{i \in \overline{1, 21}} Z_i. \end{aligned}$$

4. Zbiory E_g , $g \in G^V \cap G^K$ składają się z:

$$\begin{aligned} E_{g_1} &= \{e_1, e_{16}, e_{18}\}, & E_{g_2} &= \{e_8, e_9\}, & E_{g_3} &= \{e_2, e_{19}\}, \\ E_{g_4} &= \{e_{15}\}, & E_{g_5} &= \{e_7\}. \end{aligned}$$

5. System MD_{SC} zawiera następujące łuki wzbraniające:

$$\begin{aligned} D^I &= \{(e_1, t_1), (e_2, t_2), (e_3, t_3), (e_4, t_3), (e_5, t_3), (e_6, t_3), \\ &\quad (e_7, t_3), (e_8, t_3), (e_8, t_4), (e_5, t_5), (e_6, t_5), (e_7, t_5), \\ &\quad (e_6, t_6), (e_7, t_6), (e_7, t_7), (e_9, t_8), (e_{13}, t_{13}), (e_{14}, t_{13}), \end{aligned}$$

$$(e_{15}, t_{13}), (e_{14}, t_{14}), (e_{15}, t_{14}), (e_{15}, t_{15}), (e_{16}, t_{16}), \\ (e_{16}, t_{17}), (e_{18}, t_{17}), (e_{19}, t_{17}), (e_{19}, t_{18})\},$$

$$D^{II} = \{(g_5, t_3), (g_2, t_3), (g_5, t_5), (g_4, t_{13}), (g_4, t_{14}), (g_3, t_{17})\},$$

$$D^{III} = \emptyset$$

$$D^{IV} = \{(e_7, t_3), (e_8, t_3), (e_9, t_3), (e_7, t_5), (e_{15}, t_{13}), (e_{15}, t_{14}), \\ (e_2, t_{17}), (e_{19}, t_{17})\}.$$

System MD_{SC} w postaci graficznej jest przedstawiony na Rys. 4.10.

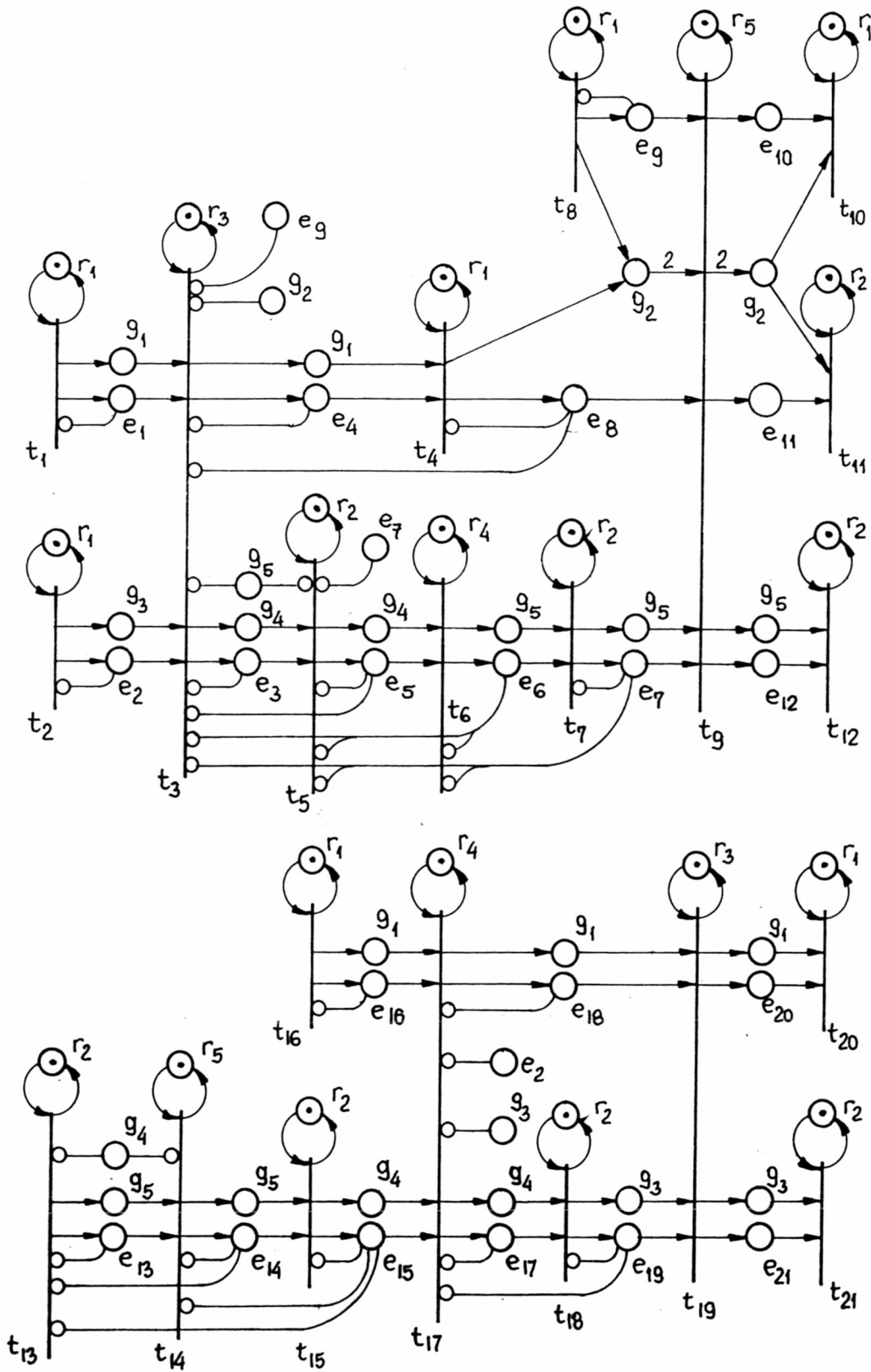
Rozszerzenie systemu MO_{SC} do MD_{SC} zapewnia poprawne współdziałanie rozważanych procesów złożonych. Przedstawiony dalej dowód prawidłowości systemu MD_{SC} skonstruowano w oparciu o następujący schemat. Rozważając dwa rozłączne i wyczerpujące zbiór $[M_0]$ przypadki markowań osiągalnych M wykazemy, że w każdym z nich istnieje niepusty zbiór przejść przygotowanych w M i dodatkowo, że jeżeli $M \neq M_0$ to w zbiorze tym istnieje przejście t , którego liczba wystąpień $k(t_\sigma, \sigma)$ w sekwencji palen $\sigma = M_0 t_1 M_1 \dots \dots t_n M_n$, $M_n = M$ spełnia warunek $k(t_\sigma, \sigma) < \max_{t \in T} k(t, \sigma)$, dla dowolnej σ prowadzącej z markowania początkowego do markowania M . Własność ta pozwoli nam udowodnić odwracalność systemu, a następnie jego realizowalność i w konsekwencji prawidłowość MD_{SC} .

Dla uproszczenia analizy wprowadzimy następujące pojęcia i oznaczenia.

Definicja 4.16

Dla systemu MD_{SC} definiujemy:

1. zbiór przejść wejściowych $T_{we} = \{t \mid \bullet t \cap E = \emptyset\}$,
2. zbiór przejść wyjściowych $T_{wy} = \{t \mid t \bullet \cap E = \emptyset\}$,
3. funkcję osiągalności określoną dla każdego $M \in [M_0]$ przez:
$$\Sigma(M) = \{\sigma \mid \sigma \text{ jest sekwencją palen prowadzącą od } M_0 \text{ do } M\},$$



Rys. 4.10. System MD_{SC} uzyskany w wyniku rozszerzenia systemu MO_{SC} przedstawionego na Rys. 4.3 (patrz Przykład 4.9).

4. funkcję krotności $k(t, \sigma)$ określoną dla dowolnej pary (t, σ) ,
 $t \in T$, $\sigma \in \Sigma(M)$, $M \in [M_0\rangle$ przez liczbę wystąpień t w σ ,
5. zbiór przejść czołowych w markowaniu M określony dla każdego
 $M \in [M_0\rangle$ przez:

$$Y(M) = \left\{ t \mid (t \in T_{we} \vee (\forall e \in \bullet t)(M(e) > 0)) \ \& \right. \\
 \left. \& (t \in T_{wy} \vee (((t, e) \in \alpha^{\#}) \Rightarrow (M(e) = 0))) \right\},$$
6. zbiór magazynów znaczących, będący funkcją określoną dla
każdego $M \in [M_0\rangle$ przez:

$$X(M) = \{ g \mid g \in G^K \ \& \ \omega(M, g) > K(g) - \varepsilon(g) \}.$$

Poniżej wprowadzimy dwa lematy mówiące o pewnych cechach markowania osiągalnego w MD_{SC} , wynikających ze struktury systemu i reguł przepływu znaczników w sieci.

Lemat 4.1

Dana jest para przejść (t, t') systemu MD_{SC} taka, że $(t, t') \in \alpha^{\#}$. Dla pary tej oraz dla każdego $M \in [M_0\rangle$ i każdego $\sigma \in \Sigma(M)$ obowiązuje:

- (1) $k(t, \sigma) \geq k(t', \sigma)$,
- (2) $(k(t, \sigma) > k(t', \sigma)) \iff (\exists e \in E)((t, e) \in \alpha^{\#} \ \& \ (e, t') \in \alpha^{\#} \ \& \ M(e) > 0))$.

Dowód

Teza lematu wypływa bezpośrednio ze struktury i reguł działania MD_{SC} .

Lemat 4.2

W systemie MD_{SC} zbiór $Y(M)$ jest niepusty dla każdego $M \in [M_0\rangle$.

Dowód

Rozważmy dwa przypadki wyczerpujące zbiór $[M_0\rangle$.

A. $M = M_0$.

Jak łatwo zauważyć $Y(M_0) = T_{we}$, $T_{we} \neq \emptyset$ a zatem lemat jest prawdziwy.

B. $M \in [M_0] \setminus \{M_0\}$.

Oznaczmy przez $Y'(M)$ zbiór przejść $t \in T$ takich, że

$$(\exists e \in \bullet t)(M(e) > 0) \ \& \ ((\exists e \in E)((t, e) \in \alpha^{\#}) \Rightarrow (M(e) = 0)).$$

Jak łatwo zauważyć $Y'(M)$ jest niepusty dla każdego $M \in [M_0] \setminus \{M_0\}$.

Stąd, istnieje $t_{\min} \in Y'(M)$ takie, że $k(t_{\min}, \sigma) = \min_{t \in Y'(M)} k(t, \sigma) = k_{\sigma}$ dla dowolnej $\sigma \in \Sigma(M)$. Jeżeli $(\forall e \in \bullet t_{\min})(M(e) > 0)$, to $t_{\min} \in Y(M)$, co spełnia tezę lematu. W przeciwnym przypadku istnieje $e \in \bullet t_{\min}$ takie, że $M(e) = 0$, a zatem, biorąc pod uwagę strukturę MD_{SC} , istnieje $t_{\sigma} \in T$ takie, że:

$$(a) \ (t_{\sigma}, t_{\min}) \in \alpha^{\#},$$

$$(b) \ (\exists e \in E)((t_{\sigma}, e) \in \alpha^{\#} \ \& \ (e, t_{\min}) \in \alpha^{\#}) \Rightarrow (M(e) = 0),$$

$$(c) \ t_{\sigma} \in T_{we} \vee (\forall e \in \bullet t_{\sigma})(M(e) > 0).$$

Udowodnimy, że $t_{\sigma} \in Y(M)$. W tym celu zauważmy, że:

$$(1) \text{ Z lematu 4.1 oraz (a) i (b) wynika, że } k(t_{\sigma}, \sigma) = k(t_{\min}, \sigma) = k_{\sigma}.$$

(2) Jeżeli $t_{\sigma} \notin Y(M)$, to z definicji zbioru $Y(M)$ oraz struktury MD_{SC} wynika, że $(\exists e \in E)(\exists t' \in e \bullet)((t_{\sigma}, e) \in \alpha^{\#} \ \& \ M(e) > 0 \ \& \ t' \in Y'(M))$. Stąd, biorąc pod uwagę lemat 4.1, wynika, że $k(t_{\sigma}, \sigma) > k(t', \sigma)$.

(3) Ponieważ wniosek (2) jest sprzeczny z (1), stąd założenie (2) jest fałszywe i w konsekwencji $t_{\sigma} \in Y(M)$.

Ponieważ przypadki A, B wyczerpują zbiór $[M_0]$, lemat jest prawdziwy.

C.B.D.O.

Zauważmy, że z dowodu Lematu 4.2 wynika, że ponieważ $t_{\min} \in Y'(M)$, zatem istnieje $t \in T$ oraz $e \in E$ takie, że $e \in t^* \cap t_{\min}^*$ & $M(e) > 0$. Stąd na podstawie Lematu 4.1 i założeń dowodu zachodzi $k(t, \sigma) > k(t_{\min}, \sigma)$, a zatem także $k(t, \sigma) > k(t_\sigma, \sigma)$. Ponieważ udowodniono, że $t_{\min} \in Y(M)$ lub $t_\sigma \in Y(M)$, możemy sformułować:

Wniosek 4.1

W systemie MD_{SC} dla każdego $M \in [M_0\rangle$ i każdej $\sigma \in \Sigma(M)$ zachodzi:

$$M \neq M_0 \Rightarrow (\exists t_\sigma \in Y(M))(k(t_\sigma, \sigma) < \max_{t \in T} k(t, \sigma)).$$

Przedstawione wyżej lematy pozwolą nam udowodnić istnienie przejścia przygotowanego w pierwszym z rozważanych przez nas przypadków markowań osiągalnych w systemie MD_{SC} . Przypadek ten obejmuje dowolne markowanie takie, że $X(M) = \emptyset$, tzn. stan istotny $\omega(M, g)$ każdego z magazynów konfliktowych g jest nie większy niż różnica ich pojemności $K(g)$ i pojemności minimalnej $\xi(g)$.

Lemat 4.3

W systemie MD_{SC} dla każdego $M \in [M_0\rangle$ zachodzi:
 $X(M) = \emptyset \Rightarrow (\exists t \in T)(t \text{ jest przygotowane w } M).$

Dowód

Z Lematu 4.2 wynika, że $Y(M) \neq \emptyset$ dla dowolnego $M \in [M_0\rangle$. Rozważmy zatem dowolną parę (M, t) taką, że $X(M) = \emptyset$ & $t \in Y(M)$. Biorąc pod uwagę Definicje 4.15, 4.16 wystarczy dla dowodu lematu pokazać, że

- (a) $(\forall g \in t^*)(M(g) \leq K(g) - W(t, g) + W(g, t)),$
- (b) $(\forall g \in \ddot{t})(\Omega(t, M, g) \leq K(g) - \xi'(g)),$
- (c) $(\forall g \in \ddot{\ddot{t}})(\Omega(t, M, g) \leq K(g) - \xi'(g) + 1).$

Z definicji 4.8 - 4.16 oraz faktu, że $X(M) = \emptyset$ wynika, że:

1. Jeżeli $g \in G \setminus (G^V \cap G^K)$ to:

- $K(g) \geq \xi(g)$,
- $W(t, g) \leq 1$, $W(g, t) \leq 1$, $\xi(g) = \xi'(g) = 1$,
- $\omega(M, g) = M(g)$, $\Omega(t, M, g) = M(g) + W(t, g) - W(g, t)$,
- $g \in G^N \Rightarrow M(g) \leq K(g) \ \& \ g \notin \dot{t} \ \&$
 $\quad \& \ (g \in \ddot{t} \Rightarrow M(g) = 0 \vee W(t, g) = 0)$
 $\quad \& \ (g \in t^\bullet \Rightarrow M(g) = 0 \vee W(g, t) = 1)$,
- $g \in G^V \Rightarrow (\omega(M, g) \leq K(g) - \xi(g)) \ \&$
 $\quad (g \in \ddot{t} \Rightarrow W(t, g) = 0).$

Przekształcając (a,b,c) poprzez podstawienie powyższych warunków, bezpośrednio wykazuje się ich słusność

$\forall g \in G \setminus (G^V \cap G^K).$

2. Jeżeli $g \in (G^V \cap G^K)$, to:

- $\omega(M, g) \leq K(g) - \xi(g)$,
- $M(g) \leq (M, g) + \xi(g) - W(t, g)$,
- $\Omega(t, M, g) \leq (M, g) + W(t, g)$,
- $W(t, g) \leq \xi''(g)$,
- $\xi''(g) = \xi(g) - \xi'(g).$

Z warunków powyższych wynika, że:

$$M(g) \leq K(g) - W(t, g),$$

$$\Omega(t, M, g) \leq K(g) - \xi'(g),$$

co dowodzi słusności (a,b,c) dla każdego $g \in (G^V \cap G^K).$

Ponieważ przypadki 1, 2 wyczerpują zbiór G , lemat jest prawdziwy.

C.B.D.O.

Zauważmy, że dowodząc Lematu 4.3 rozpatrywaliśmy dowolne $t \in Y(M)$. Stąd, biorąc pod uwagę Wniosek 4.1 oraz fakt, że $Y(M_0) = T_{we}$, możemy sformułować:

Wniosek 4.2

W systemie MD_{SC} dla każdego $M \in [M_0\rangle$ i każdej $\sigma \in \Sigma(M)$ zachodzi:

$$M \neq M_0 \ \& \ X(M) = \emptyset \Rightarrow (\exists t_\sigma \in T)(t \text{ jest przygotowane w } M \\ \& \ k(t_\sigma, \sigma) < \max_{t \in T} k(t, \sigma)).$$

Wniosek 4.3

W systemie MD_{SC} dowolne $t \in T_{we}$ jest przygotowane w M_0 .

Kolejne dwa lematy, z których pierwszy jest lematem pomocniczym, a drugi właściwym, pozwolą nam wykazać istnienie przejścia przygotowanego także i w drugim z możliwych przypadków markowań osiągalnych. Przypadek ten obejmuje dowolne markowanie takie, że $X(M) \neq \emptyset$, tzn. istnieje magazyn konfliktowy g , którego stan istotny $\omega(M, g)$ jest większy od różnicy jego pojemności $K(g)$ i pojemności minimalnej $\xi(g)$.

Lemat 4.4

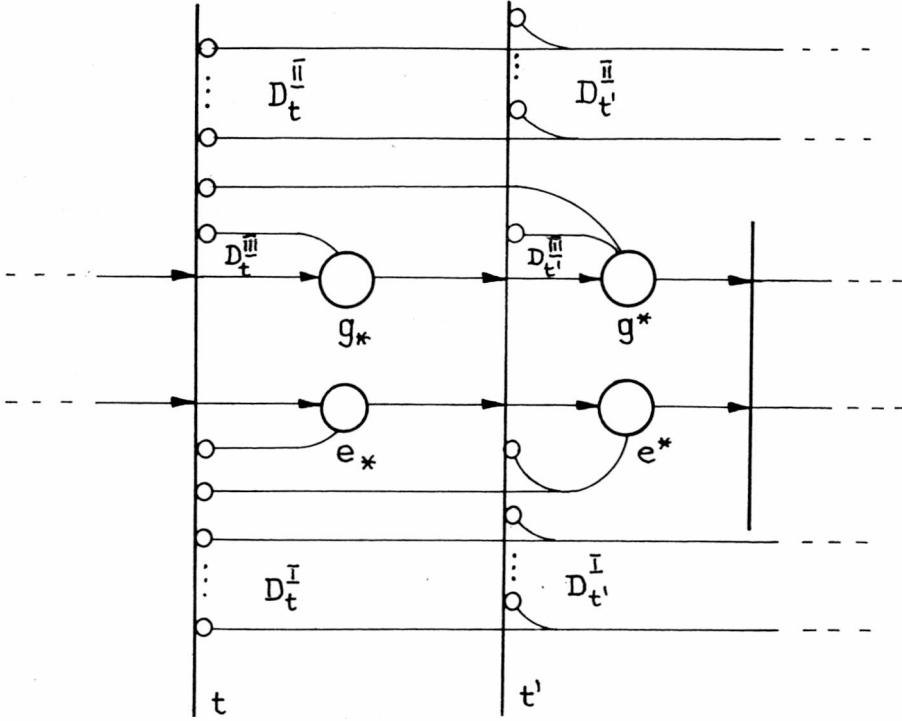
W systemie MD_{SC} dla każdego $e_{\mathbf{x}} \in E$, każdego $g_{\mathbf{x}} \in G$, każdej pary $t, t' \in T$ i każdego $M \in [M_0\rangle$, zachodzi: $(e_{\mathbf{x}} \in t \cdot \cap \cdot t') \ \& \ (g_{\mathbf{x}} \in t \cdot \cap \cdot t') \ \& \ (t' \notin T^v) \ \& \ (t \text{ jest przygotowane w } M) \ \& \ (g_{\mathbf{x}} \in X(M')) \Rightarrow (t' \text{ jest przygotowane w } M')$, gdzie $M[t \rangle M'$.

Dowód

Biorąc pod uwagę Definicję 4.15 oraz założenia Lematu wystarczy wykazać, że (patrz Rys. 4.11):

- (a) $M'(e_{\mathbf{x}}) > 0$,
- (b) $M'(g_{\mathbf{x}}) \leq K(g_{\mathbf{x}}) - W(t', g_{\mathbf{x}}) + W(g_{\mathbf{x}}, t')$,
- (c) $(\forall e \in i')(M'(e) = 0) \ \& \ (\forall g \in i')(\Omega(t', M', g) \leq K(g) - \xi'(g)) \ \& \ (\Omega(t', M', g_{\mathbf{x}}) \leq K(g_{\mathbf{x}}) - \xi'(g_{\mathbf{x}}) + 1)$.

Warunek (a) jest oczywisty w sytuacji $M[t \rangle M'$.



Rys. 4.11. Ilustracja do Lematu 4.4

Wykazując prawdziwość (b,c) zauważmy, że z definicji MD_{SC} i założeń lematu wynika:

$$(1) \quad g^{\mathfrak{K}} \in \ddot{t} \quad \text{lub} \quad g^{\mathfrak{K}} \in \ddot{\ddot{t}},$$

$$g^{\mathfrak{K}} \in \ddot{\ddot{t}} \Rightarrow g^{\mathfrak{K}} = g_{\mathfrak{K}} \Rightarrow W(g^{\mathfrak{K}}, t') = W(t', g^{\mathfrak{K}}),$$

$$g \in G^r \ \& \ (\forall M \in [M_0]) (\forall g \in G^r) (M(g) \leq \omega(M, g) + \varepsilon'(g) - 1),$$

$$(2) \quad (\forall e \in \dot{t}) (M(e) = 0),$$

$$(\forall g \in \ddot{t} \setminus \ddot{\ddot{t}}) (\Omega(t, M, g) \leq K(g) - \varepsilon'(g)),$$

$$(\ddot{\ddot{t}} = \emptyset) \vee (\ddot{\ddot{t}} = \{g_{\mathfrak{K}}\} \ \& \ \Omega(t, M, g_{\mathfrak{K}}) \leq K(g_{\mathfrak{K}}) - \varepsilon'(g_{\mathfrak{K}}) + 1),$$

$$(3) \quad \dot{t}' = \dot{t} \setminus \{e_{\mathfrak{K}}\},$$

$$\ddot{t}' = \begin{cases} \ddot{t} \setminus \{g^{\mathfrak{K}}\} & \text{jeżeli } \ddot{\ddot{t}} = \emptyset \text{ lub } g^{\mathfrak{K}} = g_{\mathfrak{K}} \\ \ddot{t} \setminus (\{g^{\mathfrak{K}}\} \cup \{g_{\mathfrak{K}}\}) & \text{jeżeli } \ddot{\ddot{t}} = \{g_{\mathfrak{K}}\} \ \& \ g_{\mathfrak{K}} \neq g^{\mathfrak{K}}, \end{cases}$$

$$\ddot{\ddot{t}}' = \emptyset \quad \text{lub} \quad \ddot{\ddot{t}}' = \{g^{\mathfrak{K}}\},$$

$$(4) \quad (\forall e \in \dot{t} \setminus \{e_{\mathfrak{K}}\}) (M'(e) = M(e)),$$

$$(\forall g \in \ddot{t} \setminus \{g^{\mathfrak{K}}\}) (\Omega(t', M', g) = \Omega(t, M, g)),$$

$$(g_{\mathfrak{K}} = g^{\mathfrak{K}}) \Rightarrow (\Omega(t', M', g_{\mathfrak{K}}) = \Omega(t, M, g_{\mathfrak{K}})),$$

$$(g_{\mathbf{x}} \neq g^{\mathbf{x}}) \Rightarrow (\Omega(t', M', g_{\mathbf{x}}) = \Omega(t, M, g_{\mathbf{x}}) - 1),$$

$$(g_{\mathbf{x}} \neq g^{\mathbf{x}}) \Rightarrow (\Omega(t', M', g^{\mathbf{x}}) = \Omega(t, M, g^{\mathbf{x}}) + 1).$$

Konfrontując wnioski 1 i 2 dochodzimy do warunku (b). Konfrontując wnioski 1, 2, 3 i 4 dochodzimy do warunku (c).

C.B.D.O.

Lemat 4.5

W systemie MD_{SC} dla każdego $M \in [M_0]$ zachodzi:

$$(X(M) \neq \emptyset) \Rightarrow (\exists t \in T)(t \text{ jest przygotowane w } M).$$

Dowód

Z definicji struktury i reguł działania MD_{SC} wynika, że zmiana wypełnienia magazynu $g \in G^K$ $\omega('M, g) \rightarrow (''M, g)$, gdzie $\omega('M, g) \leq K(g) - \xi(g)$, $\omega(''M, g) > K(g) - \xi(g)$, może nastąpić wyłącznie na skutek zapalenia przejścia $t \in \cdot g$ takiego, że $t \in \cdot e$ & $e \notin \cdot T^V$ i tylko wtedy jeżeli $'M(e) = 0$. Operacja $'M[t]''M$ powoduje wypełnienie e do $'M(e) = 1$ oraz włączenie g do zbioru X tj. $g \notin X('M)$ & $g \in X(''M)$. Zwolnienie e przez zapalenie przejścia $t \in e \cdot$ w pewnym z kolejnych stanów $'''M \in [''M]$ spowodowałoby (o ile nie zostanie wcześniej w analogiczny sposób wypełnione $e' \neq e$, $\chi(e') = g$) wyłączenie g ze zbioru X . A zatem w dowolnym markowaniu M , takim, że $X(M) \neq \emptyset$ istnieje czwórka (e, g, t, t') taka, że $e, g \in t \cdot \cap \cdot t'$ & $t' \notin T^V$ & $g \in X(M)$ & $M(e) = 1$. Zakładając, że M zostało osiągnięte w wyniku sekwencji paleń $\sigma \in \Sigma(M)$, możemy spośród wszystkich czwórek spełniających powyższe warunki w M wybrać taką, dla której wypełnienie e nastąpiło najpóźniej, tj. oznaczając $\sigma = M_0 t_1 M_1 \dots M_{l-1} t_l M_l \dots \dots t_n M_n$, $M_n = M$, wybierzemy $l \in \overline{1, n}$ takie, że:

$$(a) \quad t_l = t \text{ \& } t \in \cdot e_{\mathbf{x}} \text{ \& } t \in \cdot g_{\mathbf{x}} \text{ \& } e \notin \cdot T^V,$$

$$(b) \quad g_{\mathbf{x}} \in X(M_s), \quad s \in \overline{1, n},$$

$$(c) \quad M_{l-1}(e_{\mathfrak{x}}) = 0 \ \& \ M_s(e_{\mathfrak{x}}) = 1, \ s \in \overline{1, n},$$

$$(d) \quad (\forall g \in G^K)(\omega(M_l, g) \geq \omega(M_n, g)).$$

Ponieważ $t' \notin T^V$, $t' \in e_{\mathfrak{x}}^\bullet$, t jest przygotowane w M_{l-1} oraz $g_{\mathfrak{x}} \in X(M_l)$, zatem czwórka $(e_{\mathfrak{x}}, g_{\mathfrak{x}}, t, t')$ spełnia założenie lematu 4.4 i w konsekwencji t' jest przygotowane w M_l . Wykażemy, że t' jest również przygotowane w M_n . W tym celu wystarczy udowodnić, że:

$$(1) \quad M_n(e_{\mathfrak{x}}) = M_l(e_{\mathfrak{x}}),$$

$$(2) \quad (\forall e \in \dot{t}')(M_n(e) = 0),$$

$$(3) \quad (\forall g \in \ddot{t}' \cup \ddot{\ddot{t}}')(\Omega(t', M_n, g) \leq \Omega(t', M_l, g)),$$

$$(4) \quad M_n(g^{\mathfrak{x}}) \leq M_l(g^{\mathfrak{x}}), \quad g^{\mathfrak{x}} \in (t')^\bullet.$$

Warunek (1) dany jest w założeniu (c).

Warunek (2) wynika z konfrontacji faktów:

- $\dot{t}' = \emptyset$ lub $t' = \{e_1, e_2, \dots, e_m\}$,
- zmiana markowania e_1 wymaga zapalenia t' , zmiana markowania e_i wymaga aby $M(e_{i-1}) > 0$, $i \in \overline{2, m}$,
- $M_{l-1}[t > M_l \ \& \ \omega(M_l, g_{\mathfrak{x}}) > K(g_{\mathfrak{x}}) - \mathcal{E}(g_{\mathfrak{x}})]$, stąd $(\forall e \in \dot{t} \setminus \{e_{\mathfrak{x}}\})(M_l(e) = 0)$,
- $\dot{t}' = \dot{t} \setminus \{e_{\mathfrak{x}}\}$,
- $M_s(e_{\mathfrak{x}}) = 1$, $s \in \overline{1, n}$ stąd t' nie zostało zapalone w żadnym markowaniu M_s .

Warunek (3) wynika z konfrontacji faktów:

- $g \in \ddot{t} \Rightarrow g \in G^K \vee g \in G^N$,
- $g \in \ddot{\ddot{t}} \Rightarrow g \in G^K$,
- $g \in G^K \Rightarrow \omega(M_n, g) \leq \omega(M_l, g)$,
- $g \in G^N \ \& \ g \in \ddot{t} \Rightarrow M_n(g) = M_l(g) \Rightarrow \omega(M_n, g) = \omega(M_l, g)$.

Warunek (4) wynika z konfrontacji faktów:

- $g^{\mathfrak{x}} \in G^r \ \& \ (g \in G^N \vee g \in G^K)$,
- $g^{\mathfrak{x}} \in G^N \Rightarrow (M_n, g^{\mathfrak{x}}) = M_l(g^{\mathfrak{x}})$,

- $g^{\#} \in G^K \Rightarrow \omega(M_n, g^{\#}) \leq \omega(M_1, g^{\#})$,
- $(\forall M \in [M_0]) (\forall g \in G^T) (M(g) \leq \omega(M, g) + \varepsilon'(g) - 1)$.

Ponieważ warunki (1 - 4) są prawdziwe, zatem t' jest przygotowane w M_n .

C.B.D.O.

Zauważmy, że w dowodzie powyższym wykazaliśmy przygotowanie przejścia t' w markowaniu M takim, że $M(e_{\#}) = 1$, $e_{\#} \in t' \cap t$. Stąd na podstawie Lematu 4.1 zachodzi $k(t, \sigma) > k(t', \sigma)$ dla dowolnego $\sigma \in \Sigma(M)$. Możemy zatem sformułować:

Wniosek 4.4

W systemie MD_{SC} dla każdego $M \in [M_0]$ i każdej $\sigma \in \Sigma(M)$ zachodzi:

$$X(M) \neq \emptyset \Rightarrow (\exists t_{\sigma} \in T) (t_{\sigma} \text{ jest przygotowane w } M \ \& \ k(t_{\sigma}, \sigma) < \max_{t \in T} k(t, \sigma).$$

Konfrontując wnioski wynikające z poprzednich lematów możemy sformułować Lemat 4.6 znajdujący bezpośrednie zastosowanie w dowodzie twierdzenia o odwracalności systemu MD_{SC} .

Lemat 4.6

W systemie MD_{SC} dla każdego $M \in [M_0]$ i każdej $\sigma \in \Sigma(M)$ zachodzi:

$$M \neq M_0 \Rightarrow (\exists t_{\sigma} \in T) (t_{\sigma} \text{ jest przygotowane w } M \ \& \ k(t_{\sigma}, \sigma) < \max_{t \in T} k(t, \sigma).$$

Dowód

Konfrontując Wnioski 4.2 i 4.4, Lemat jest oczywisty.

Twierdzenie 4.5

System MD_{SC} jest odwracalny.

Dowód

Należy wykazać, że $(\forall M \in [M_0]) (M_0 \in [M])$. Lemat ten jest w sposób oczywisty spełniony dla przypadku $M = M_0$. Rozważmy zatem dowolne $M \in [M_0]$, $M \neq M_0$ oraz dowolną sekwencję $\sigma_1 \in \Sigma(M)$.

Na mocy Lematu 4.6 możemy skonstruować sekwencję $\sigma_2 = \sigma_1 t_1 M_1$ taką, że $k(t_1, \sigma_1) < \max_{t \in T} k(t, \sigma_1)$. A zatem $k(t_1, \sigma_2) \leq \max_{t \in T} k(t, \sigma_1)$ i w konsekwencji $\max_{t \in T} k(t, \sigma_2) = \max_{t \in T} k(t, \sigma_1)$. Konstruuując kolejne sekwencje $\sigma_3 = \sigma_1 t_1 M_1 t_2 M_2 = \sigma_2 t_2 M_2$ i dalej $\sigma_{i+1} = \sigma_i t_i M_i$ takie, że $k(t_i, \sigma_i) < \max_{t \in T} k(t, \sigma_i)$, otrzymujemy $\max_{t \in T} k(t, \sigma_i) = \max_{t \in T} k(t, \sigma_{i-1}) = \dots = \max_{t \in T} k(t, \sigma_1)$.

Ponieważ dla każdej pary $(i-1, i)$ zachodzi $(\exists t \in T) (k(t, \sigma_i) > k(t, \sigma_{i-1}))$, to biorąc pod uwagę skończoność zbioru T , wystąpi sekwencja $\sigma_{n+1} = \sigma_n t_n M_n$ prowadząca do markowania M_n , w którym żadne z przejść przygotowanych t' nie będzie spełniać warunku $k(t', \sigma_{n+1}) < \max_{t \in T} k(t, \sigma_{n+1})$.

Zauważmy, że $M_n \notin [M_0] \setminus \{M_0\}$ ponieważ przeczyłoby to Lematowi 4.6. A zatem $M_n = M_0$ i w konsekwencji $M_0 \in [M]$.

C.B.D.O.

Twierdzenie 4.6

System MD_{SC} jest realizowalny.

Dowód

Zauważmy, że:

1. $(\forall t \in T_{we}) (t \text{ jest przygotowane w } M_0)$,
2. z powyższego oraz Twierdzenia 4.5 wynika, że:

$$(\forall t \in T_{we}) (\exists \sigma_t \in \Sigma(M_0)) (\sigma_t = M_0 t \dots M_0),$$

3. ze struktury systemu wynika, że:

$$(\exists t \in T_{we}) (\exists t' \in T \setminus T_{we}) (t, t' \in \alpha^{\mathbb{K}}) \Rightarrow (k(t', \sigma_t) > 0),$$

$$(\forall t' \in T \setminus T_{we}) (\exists t \in T_{we}) ((t, t') \in \alpha^{\mathbb{K}}).$$

Konfrontując powyższe uwagi dochodzimy do wniosku:

$(\forall t' \in T)(\exists \sigma \in \Sigma(M_0))(k(t, \sigma) > 0)$, a zatem $(\forall t \in T)(\exists M \in [M_0])$
 (t jest przygotowane w M). C.B.D.O.

W oparciu o Twierdzenie 4.6 możemy bezpośrednio wykazać słuszność Twierdzenia 4.4, które pozostawiliśmy bez dowodu.

Dowód Twierdzenia 4.4

Zauważmy, że dla dowolnego systemu MO_{SC} takiego, że $(\forall g \in G)(K(g) \geq \mathcal{E}(g))$ istnieje system $MD_{SC} = (MO_{SC}, D)$. Ponieważ z Definicji 4.13, 4.15 wynika, że dowolna sekwencja paleń w MD_{SC} jest również sekwencją paleń dla MO_{SC} , stąd realizowalność systemu MD_{SC} implikuje realizowalność systemu MO_{SC} o w/w ograniczeniu. A zatem Twierdzenie 4.4 jest prawdziwe.

Twierdzenie 4.7

System MD_{SC} jest prawidłowy.

Dowód

Wobec Definicji 4.7 i Twierdzeń 4.5, 4.6, dowód jest natychmiastowy.

Udowodnione w powyższym rozdziale Lematy 4.1 - 4.5 pozwoliły nam wykazać słuszność Lematu 4.6, mówiącego, że przy dowolnej sekwencji paleń σ , prowadzącej do dowolnego markowania $M \neq M_0$ systemu MD_{SC} istnieje przejście przygotowane t , które w sekwencji σ występuje mniej razy niż inne. Fakt ten pozwolił nam udowodnić odwracalność MD_{SC} .

Własność powyższa w konfrontacji ze strukturą sieci oraz łatwym do sprawdzenia faktem, że każde przejście wejściowe systemu MD_{SC} jest przygotowane w stanie początkowym, doprowadziły nas do wniosku, że MD_{SC} jest realizowalny. Dowodząc odwracalności i realizowalności systemu MD_{SC} wykazaliśmy tym samym jego prawidłowość.

Zachowanie przez MD_{SC} prawidłowości działania oznacza, że każda jego realizacja jest dopuszczalna ze względu na kryterium bezblokadowości modelowanych procesów.

4.6. Złożoność obliczeniowa syntezy i analizy bieżącej

Rozważania poprzednich podrozdziałów ujmują w sposób definicyjno-opisowy metodę konstrukcji systemu MD_{SC} dla dowolnej pary danych (S, C) . Rozpisanie powyższej metody w postaci szczegółowego algorytmu stanowi prosty problem natury technicznej, którego schemat zostanie przedstawiony w rozdziale 5.1. W oparciu o sformułowane wcześniej definicje można natomiast oszacować złożoność poszczególnych etapów syntezy modelu.

Zakładając, że każda z danych opisujących parę (S, C) może być zapisana przy użyciu jednego słowa pamięci, to łatwo zauważyć, że liczba słów użytych na całość zapisu nie przekracza wartości $a \cdot k$ (gdzie "a" jest pewną stałą niezależną od (S, C) , a "k" jest liczbą wierzchołków grafu H). Podobnie, zakodowanie danych opisujących system MO_{SC} wymaga nie więcej słów niż $b \cdot k$ (gdzie "b" stała niezależna od (S, C)).

Proporcjonalności te pozwalają nam przyjąć jako argument funkcji złożoności obliczeniowej zmienną k , oraz przez analizę zależności ujętych definicjami, ocenić rząd złożoności poszczególnych etapów konstrukcji MD_{SC} (Tabela 4.1).

Sumaryczna złożoność obliczeniowa algorytmu syntezy MD_{SC} jest zatem rzędu $O(k^2)$.

Zadaniem analizy bieżącej modelu (dokonywanej w fazie symulacyjnej realizacji modelowanych procesów) jest wyznaczanie pod-

zbioru przejść przygotowanych w określonym stanie systemu. Biorąc pod uwagę Definicje 4.4 i 4.13, wymagany dla tego celu algorytm nie przekracza rzędu złożoności $O(k^2)$.

Tabela 4.1

Rząd złożoności obliczeniowej syntezy MD_{SC}

etap syntezy	nr definicji	rząd złożoności obliczeniowej
MO_{SC}	4.3	$O(k)$
$T^v, T^r, G^v, G^r, G^K, G^N$	4.8 - 4.10	$O(k)$
$\xi(g), g \in G$	4.11	$O(k)$
$Z_t, t \in T$	4.12	$O(k^2)$
D	4.13	$O(k)$

Niska złożoność obliczeniowa syntezy MD_{SC} stwarza możliwość komputerowej konstrukcji modelu w trybie interakcyjnym (co będzie istotne dla dalszej części tej pracy). Z kolei niska złożoność analizy bieżącej modelu sieciowego umożliwia wykorzystanie MD_{SC} dla celów sterowania operatywnego ESP tak w systemach rzeczywistych jak i symulacyjnych.

4.7. Model etapowy

Dla celów analizy jakościowej związanej z problemami dopuszczalności sterowania dokonaliśmy dyskretyzacji procesów wytwórczych do poziomu operacji. Jej wystąpienie uznaliśmy za zdarzenie prowadzące do elementarnej zmiany stanu procesów i rozdziału zasobów w systemie. Rozpatrując działanie systemu w czasie

musimy uznać, że zmiana taka dokonywana jest w sposób etapowy, gdzie wyróżnienie poszczególnych etapów jest zależne od projektu struktury wymiany informacji pomiędzy systemem sterowania a procesami.

Dla potrzeb niniejszej pracy założymy, że zmiany stanu systemu związane z realizacją pojedynczej operacji generowane są przez sekwencyjnie następujące po sobie trzy zdarzenia:

Zdarzenie A' - wzbudzenie procesu poprzez przydział zasobów dla realizacji kolejnej operacji (rozkaz inicjacji operacji przesyłany przez system sterowania do realizatora),

Zdarzenie A'' - zwrot magazynów wejściowych przez proces (informacja zwrotna od realizatora do systemu sterowania o zakończeniu pierwszego etapu operacji, tj. pobraniu przedmiotów z magazynów),

Zdarzenie A''' - zwrot realizatora przez proces i przejście do stanu oczekiwania (informacja zwrotna od realizatora do systemu sterowania o zakończeniu operacji i złożeniu przedmiotów w magazynach wyjściowych).

W modelu sieciowym wykonaniu operacji odpowiada usunięcie znaczników z miejsc wejściowych przejścia t reprezentującego daną operację oraz zajęcie jednostek (których jest skończona liczba) miejsc wyjściowych t przez pojawienie się w nich znaczników. Wyznaczając określone przejście do palenia dokonujemy niejako rezerwacji tych elementów, tak aby chwilowa obecność znaczników (zanim nie zostaną usunięte) lub chwilowo wolne jednostki miejsc wyjściowych (zanim nie zostaną one zajęte przez pojawiające się w nich znaczniki) nie mogła stać się przyczyną przygotowania innego przejścia.

Rozważania takie nie są konieczne, gdy rozpatrujemy działanie systemu jako sekwencję palenia przejść (jak np. w systemie MD_{SC}). Ponieważ zdarzenia A', A'', A''' związane z różnymi operacjami mogą się przeplatać, tzn. wykonanie pewnej operacji może rozpocząć się jeszcze przed zakończeniem innej, w kategoriach modelu sieciowego musimy założyć równoległe palenie pewnych przejść. W sytuacji takiej niezbędne jest zabezpieczenie się przed przydziałem elementów "wykorzystywanych" przez jedno palące się przejścia innym. Możliwość taką daje wprowadzenie drugiego typu znaczników, tzw. znaczników rezerwacji, których usunięcie (gdy są one na miejscu wejściowym) lub zastąpienie znacznikiem właściwym (gdy są one na miejscu wyjściowym) może nastąpić jedynie w wyniku zdarzeń związanych z paleniem przejścia, które spowodowało wystąpienie tej rezerwacji.

Zdarzenie A' - rozkaz wykonania operacji - będzie wówczas interpretowane jako zastąpienie znaczników miejsc wejściowych znacznikami rezerwacji oraz wystąpienie znaczników rezerwacji w miejscach wyjściowych (rezerwacja zasobów potrzebnych dla wykonania operacji - przedmiotów, realizatora i miejsc w magazynach wyjściowych). Zdarzeniu A'' będzie odpowiadać zniknięcie znaczników rezerwacji w miejscach wejściowych (pobranie przedmiotów operacji z magazynów wejściowych). Wreszcie zdarzenie A''' będzie związane z zastąpieniem znaczników rezerwacji w miejscach wyjściowych właściwymi znacznikami (złożenie wykonanych przedmiotów w rezerwowanych wcześniej jednostkach magazynów).

Przedstawiając wprowadzoną tu koncepcję w sposób formalny rozszerzymy pojęcie systemu P/T z łukami wzbraniającymi do systemu, który będziemy nazywać etapowym i którego strukturę i zasady działania określimy następująco.

Definicja 4.17

Systemem etapowym zbudowanym na systemie P/T z łukami wzbra-
niającymi $PH = ((P, T, F, K, W, M_0), D)$ będziemy nazywać parę $PE =$
 $= (PH', Ms_0)$, w której $PH' = ((P, T, F, K, W), D)$, Ms_0 jest stanem po-
czątkowym systemu PE wyznaczonym przez $Ms_0 = (MM_0, s_0)$, gdzie
 $MM_0 = (M_0, P \rightarrow \{0\})$ jest markowaniem początkowym miejsc i $s_0 :$
 $T \rightarrow \{0\}$ jest stanem początkowym przejść.

Definicja 4.18

Dla systemu PE jak w Definicji 4.17 definiujemy:

- (i) stan systemu jest określony przez $Ms = (MM, s)$, gdzie $MM =$
 $= (MM_1, MM_2)$ jest parą funkcji $MM_1 : P \rightarrow N$, $MM_2 : P \rightarrow N$,
ograniczonych warunkiem $(\forall p \in P)(MM_1(p) + MM_2(p) \leq K(p))$,
określających markowanie miejsc,
 $s : T \rightarrow \{0, 1, 2\}$ jest funkcją określającą stan przejść,
gdzie

$$s(t) = \begin{cases} 0, & \text{jeżeli } t \text{ nie jest zapalone,} \\ 1, & \text{jeżeli } t \text{ jest w 1-szym etapie palenia,} \\ 2, & \text{jeżeli } t \text{ jest w 2-gim etapie palenia;} \end{cases}$$
- (ii) przejście $t \in T$ jest przygotowane w stanie $Ms = (MM, s)$
jeżeli:
 - a) $s(t) = 0$,
 - b) $(\forall p \in {}^*t)(MM_1(p) \geq W(p, t))$,
 - c) t jest przygotowane w systemie PN_H w markowaniu $M = MM_1 + MM_2$;
przejście $t \in T$ może w stanie $Ms = (MM, s)$:
 - a) zostać zapalone, jeżeli jest przygotowane w Ms ,
 - b) zmienić etap palenia, jeżeli $s(t) = 1$,
 - c) zostać wygaszone, jeżeli $s(t) = 2$;
- (iii) zapalenie, zmian etapu palenia, wygaszenie przejścia $t \in T$
w stanie $Ms = (MM, s)$ powodują zmianę stanu systemu $Ms[t] > Ms'$,

$Ms' = (MM', s')$, określoną odpowiednio:

a) $MM1'(p) = MM1(p) - W(p, t)$, $p \in P$,

$MM2'(p) = MM2(p) + \max(W(p, t), W(t, p))$, $p \in P$,

$s'(t) = 1$,

b) $MM1'(p) = MM1(p)$, $p \in P$

$MM2'(p) = MM2(p) - \max(0, W(p, t) - W(t, p))$, $p \in P$,

$s'(t) = 2$,

c) $MM1'(p) = MM1(p) + W(t, p)$, $p \in P$,

$MM2'(p) = MM2(p) - W(t, p)$, $p \in P$,

$s'(t) = 0$.

(iv) zbiorem stanów osiągalnych systemu PE ze stanu Ms jest najmniejszy zbiór stanów $[Ms\rangle$ taki, że

a) $Ms \in [Ms\rangle$,

b) jeżeli $Ms' \in [Ms\rangle$ i $Ms'[t\rangle Ms''$ dla dowolnego $t \in T$, to $Ms'' \in [Ms\rangle$;

(v) sekwencję $\tilde{\sigma} = Ms_0 t_1 Ms_1 t_2 Ms_2 \dots t_n Ms_n$ taką, że $Ms_0 [t_1\rangle Ms_1$, $Ms_1 [t_2\rangle Ms_2, \dots, Ms_{n-1} [t_n\rangle Ms_n$ będziemy nazywać sekwencją zdarzeń w systemie PE; zbiór wszystkich sekwencji $\tilde{\sigma}$ takich, że $Ms_n = Ms$ będziemy oznaczać przez $\Sigma(Ms)$, zbiór wszystkich sekwencji zdarzeń w PE będziemy oznaczać przez $\Sigma(PE)$;

(vi) przez $T(Ms)$, $TT(Ms)$, $Ms \in [Ms_0\rangle$, będziemy oznaczać funkcje określone przez:

$$T(Ms) = \{t \mid t \in T \text{ \& } t \text{ jest przygotowane w } Ms\},$$

$$TT(Ms) = \{t \mid t \in T \text{ \& } (t \in T(Ms) \vee s(t) \in \{1, 2\})\},$$

które będziemy nazywać odpowiednio zbiorem przejść przygotowanych (przejść, które mogą zostać zapalone) w Ms , i zbiorem zdarzeń, które mogą wystąpić w Ms ;

(vii) system PE jest:

a) realizowalny, jeżeli $(\forall t \in T)(\exists Ms \in [Ms_0\rangle)(t \in T(Ms))$,

- b) odwracalny, jeżeli $(\forall Ms \in [Ms_0]) (Ms_0 \in [Ms])$,
 c) prawidłowy, jeżeli jest realizowalny i odwracalny.

Z punktu widzenia zastosowań systemu PE istotne jest, że zachowuje on własności posiadane przez system PH, na którym został on zbudowany.

Twierdzenie 4.8

Realizowalność/odwracalność/prawidłowość systemu PH implikuje odpowiednio realizowalność/odwracalność/prawidłowość systemu PE.

Dowód

Oznaczmy przez:

$Ms'[(t)]Ms''$ zmianę stanu PE spowodowaną sekwencją zdarzeń -
 - zapalenie przejścia $t \in T(Ms)$, zmiana etapu palenia t ,
 wygaszenie t ;

$\sigma_E = Ms_0(t_1)Ms_1(t_2)Ms_2 \dots (t_n)Ms_n$ - sekwencję taką, że

$Ms_0[(t_1)]Ms_1, Ms_1[(t_2)]Ms_2 \dots Ms_{n-1}[(t_n)]Ms_n$;

$\Sigma_E(Ms)$, $Ms \in [Ms_0]$ - zbiór wszystkich sekwencji σ_E prowadzących do Ms ;

$T(M)$ - zbiór przejść przygotowanych w PH w markowaniu $M \in [M_0]$.

Zauważmy, że z Definicji 4.17, 4.18 wynika:

$$(1) (\exists Ms \in [Ms_0]) (\Sigma_E(Ms) \neq \emptyset) \iff (Ms = (MM, s) \ \& \ MM2 = 0 \ \& \ s = 0)$$

$$(2) ((\exists M \in [M_0]) \ \& \ (\exists Ms \in [Ms_0])) (Ms = ((M, 0), 0) \implies \implies (\forall t \in T) ((M[t]M' \iff Ms[(t)]Ms') \ \& \ Ms' = ((M', 0), 0)),$$

$$(3) Ms_0 = ((M_0, 0), 0),$$

$$(4) (\forall Ms \in [Ms_0]) (\exists Ms' \in [Ms]) (Ms' = (MM', s') \ \& \ MM2' = 0 \ \& \ s' = 0).$$

Z wniosków powyższych wynika, że istnieje funkcja

$\psi: [M_0] \rightarrow [Ms_0]$ taka, że

- (i) $\psi(M) = ((M, 0), 0)$,
- (ii) $T(M) = T(\psi(M))$,
- (iii) $M \in [M_0] \ \& \ M' \in [M] \iff \psi(M) \in [Ms_0] \ \& \ \psi(M') \in [\psi(M)]$,
- (iv) $Ms \in [Ms_0] \implies (\exists M \in [M_0])(\psi(M) \in [Ms])$.

Biorąc pod uwagę, że $Ms_0 = \psi(M_0)$, z (i) - (iv) wynika, że:

- $(\forall t \in T)(\exists M \in [M_0])(t \in T(M)) \implies (\forall t \in T)(\exists Ms \in [Ms_0])(t \in T(Ms))$,
- $(\forall M \in [M_0])(M_0 \in [M]) \implies (\forall Ms \in [Ms_0])(Ms_0 \in [Ms])$.

C.B.D.O.

Uwaga

Dla celów niniejszej pracy wystarczające jest wykazanie prawdziwości implikacji ujętej Twierdzeniem 4.8. Nieco większym nakładem pracy można wykazać, że zachodzi także implikacja, "w odwrotną stronę", a w konsekwencji udowodnić równoważność systemów PH i PE pod względem wymienionych w Twierdzeniu 4.8 własności.

Zachowanie własności gwarantujących bezblokadowe działanie systemu PH po jego transformacji do PE pozwala nam przekształcić zbudowany wcześniej model dopuszczalnej dynamiki systemu WZPP MD_{SC} do jego odpowiednika ME_{SC} , będącego systemem etapowym zbudowanym na MD_{SC} . System ten umożliwi nam reprezentację nie tylko stanów pasywnych procesów (oczekiwania na wykonanie kolejnej operacji), ale również stanów aktywnych (wykonywania operacji).

5. MODEL SYMULACYJNY PROCESÓW PRODUKCYJNYCH W ESP

Przedstawione w poprzednim rozdziale wyniki teoretyczne mogą być wykorzystane zarówno dla celów automatycznej syntezy modeli sterowania przebiegiem rzeczywistych procesów produkcyjnych, jak i budowy systemów komputerowo wspomaganego modelowania i oceny efektywności funkcjonowania ESP. Zasadniczym elementem takiej implementacji jest opracowanie algorytmu transformacji danych opisujących parę (S, C) w dane opisujące strukturę modelu MD_{SC} (a zatem i ME_{SC}).

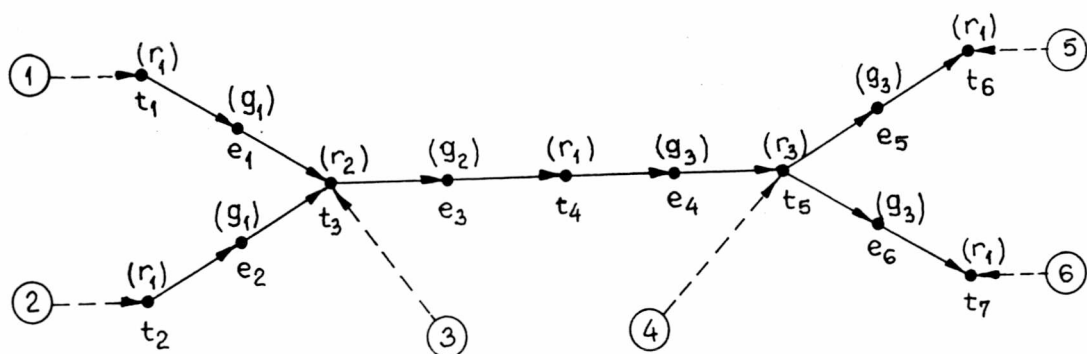
5.1. Algorytm syntezy modelu

Będziemy zakładać, że dane opisujące strukturę S (Def. 4.1, 4.2) deklarowane są w postaci tablicy list

$$(5.1) \quad LS = (LS_i \mid \pi_i \in \Pi)$$

odpowiadających poszczególnym procesom $\pi_i \in \Pi$ składającym się na system WZPP. Każda z list LS_i zawiera rekordy specyfikujące marszruty technologiczne procesów liniowych składających się na π_i oraz numery wierzchołków początkowych i końcowych poszczególnych marszrut, wyznaczając w ten sposób ich uporządkowanie.

Przykładowo:



Specyfikacja procesu w postaci graficznej

wierzchołek początkowy	marszruta procesu składowego	wierzchołek końcowy
1	$r_1 g_1 r_2$	3
2	$r_1 g_1 r_2$	3
3	$r_2 g_2 r_1 g_3 r_3$	4
4	$r_3 g_3 r_1$	5
4	$r_3 g_3 r_1$	6

Specyfikacja procesu w postaci listy rekordów (nr wierzchołka początkowego, marszruta, nr wierzchołka końcowego)

Listy te nie deklarują w sposób jawny elementów zbiorów E i T, pozwalają jednak na ich bezpośrednią generację.

Drugi z elementów opisu danych wejściowych - funkcja pojemności magazynów C zadana jest w postaci listy

$$(5.2) \quad CG = (C(g_i) | i \in \overline{1, \|G\|}).$$

Wygodnie jest założyć, że dane opisujące model MD_{SC} będą ustrukturalizowane w sposób ułatwiający ich analizę bieżącą tj. ustalanie które z przejść jest przygotowane w danym stanie systemu. Zgodnie z definicjami 4.4, 4.15 i 4.18 sprawdzenie przygotowania $t \in T$ wymaga znajomości markowania miejsc $p \in P$ incydentnych do t , tj. należących do zbiorów $\cdot t$, $t \cdot$, $\dot{t}$, $\ddot{t}$, $\ddot{\bar{t}}$ oraz do poszczególnych zbiorów E_g^V (Definicja 4.13) takich, że $g \in t \cdot \cup \dot{t} \cup \ddot{t}$. Dla każdego $t_i \in T$ utworzymy zatem listę rekordów L zawierających indeksy miejsc należące do w/w zbiorów. Lista ta będzie złożona z dwóch części $L_i = (L'_i, L''_i)$, gdzie:

$$(5.3) \quad L'_i = (L'_{i1}, L'_{i2}) \text{ zawiera indeksy "j" miejsc } p_j \text{ należących odpowiednio do } p_j \in \dot{t}_i \text{ i } p_j \in \cdot t_i,$$

$$(5.4) \quad L''_i = (L''_{i1}, L''_{i2}, L''_{i3}) \text{ zawiera indeksy "j" miejsc } p_j \text{ należących odpowiednio do } p_j \in \dot{t}_i, p_j \in \ddot{t}_i, p_j \in \ddot{\bar{t}}_i.$$

Tablicę złożoną z list L_i , $i \in \overline{1, \|T\|}$, (podobnie L'_i , L''_i) oznaczmy wspólnym symbolem L (odpowiednio L' , L'').

Miejsca należące do zbiorów E_g^V zostaną ujęte wspólną listą EG , gdzie:

$$(5.5) \quad EG = (EG_i | g_i \in G) \text{ zawiera rekordy } EG_i \text{ składające się z indeksów "j" miejsc } p_j \in E_{g_i}^V.$$

Ponadto dla sprawdzenia wartości logicznych formuł opisujących przygotowanie przejść konieczna jest znajomość funkcji pojemności miejsc K i wag łuków W (Def. 4.3) oraz minimalnej pojemności magazynów (Def. 4.11). Dane te mogą być przedstawione w postaci list:

$$(5.6) \quad KP = (K(p_i) | i \in \overline{1, \|P\|}),$$

$$(5.7) \quad EP = (\varepsilon(g_i) | g_i \in G),$$

$$(5.8) \quad WPT = (((W(t_i, p_j) | p_j \in t_i^*) | i \in \overline{1, \|T\|}), ((W(p_j, t_i) | p_j \in {}^*t_i) | i \in \overline{1, \|T\|}))).$$

Wymienione dane w pełni charakteryzują strukturę MD_{SC} , natomiast dla ich wyznaczenia potrzebne są pewne dane pośrednie opisujące zbiory T^V , T^R , G^V , G^R , G^N , G^K (Definicje 4.8 - 4.10) oraz strefy wzbronień Z_t poszczególnych miejsc $t \in T$ (Definicja 4.12). Dane te będą ujęte w postaci list TVR, GVR, GNK i tablicy SW, gdzie:

$$(5.9) \quad TVR = (x_i | i \in \overline{1, \|T\|}), \text{ gdzie } x_i = 0 \text{ jeżeli } t_i \in T^R, x_i = 1 \text{ jeżeli } t_i \in T^V,$$

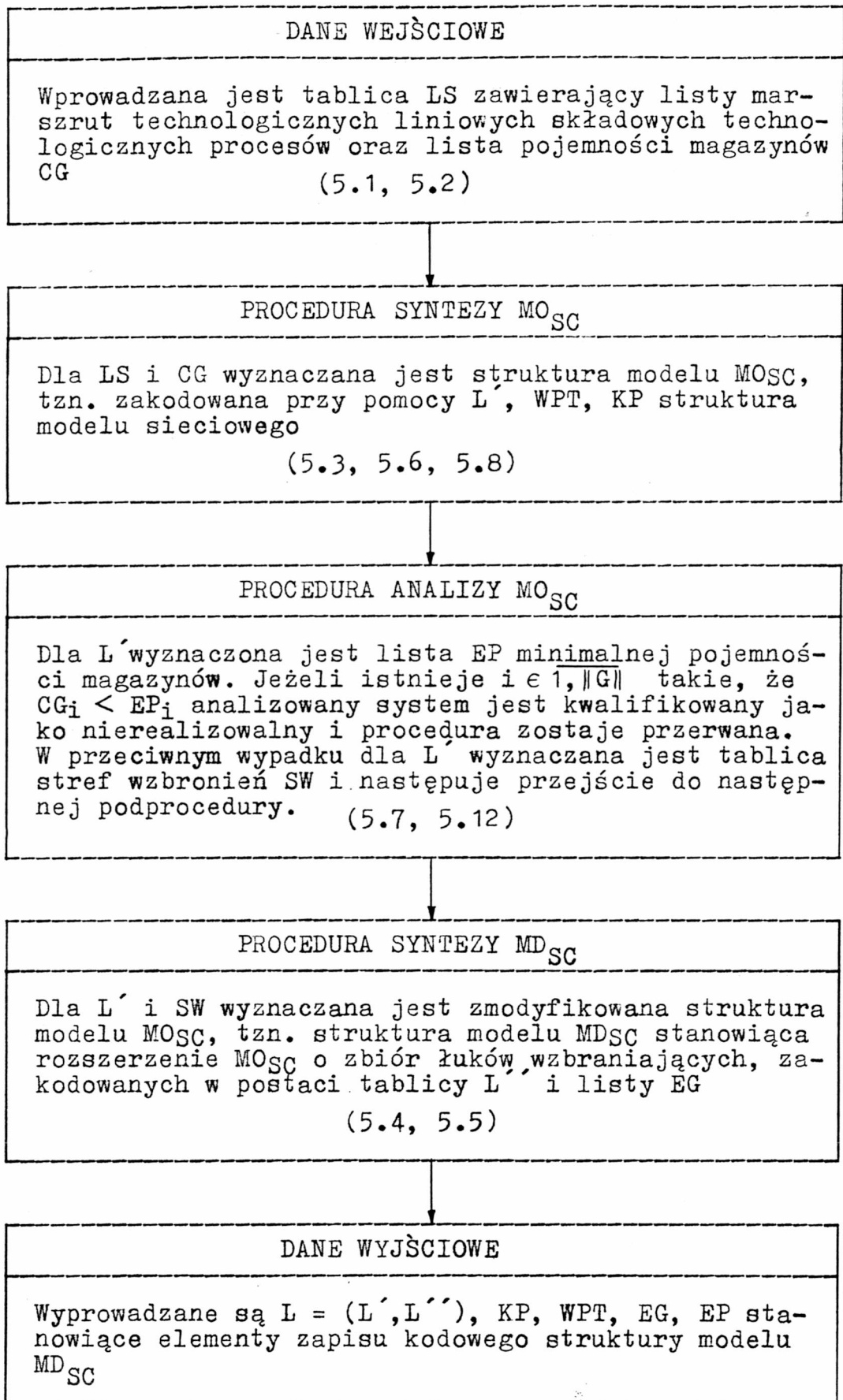
$$(5.10) \quad GVR = (x_i | i \in \overline{1, \|G\|}), \text{ gdzie } x_i = -1 \text{ jeżeli } g_i \in G \setminus G^V, x_i = 0 \text{ jeżeli } g_i \in G \setminus G^R, x_i = 1 \text{ jeżeli } g_i \in G^R \cap G^V,$$

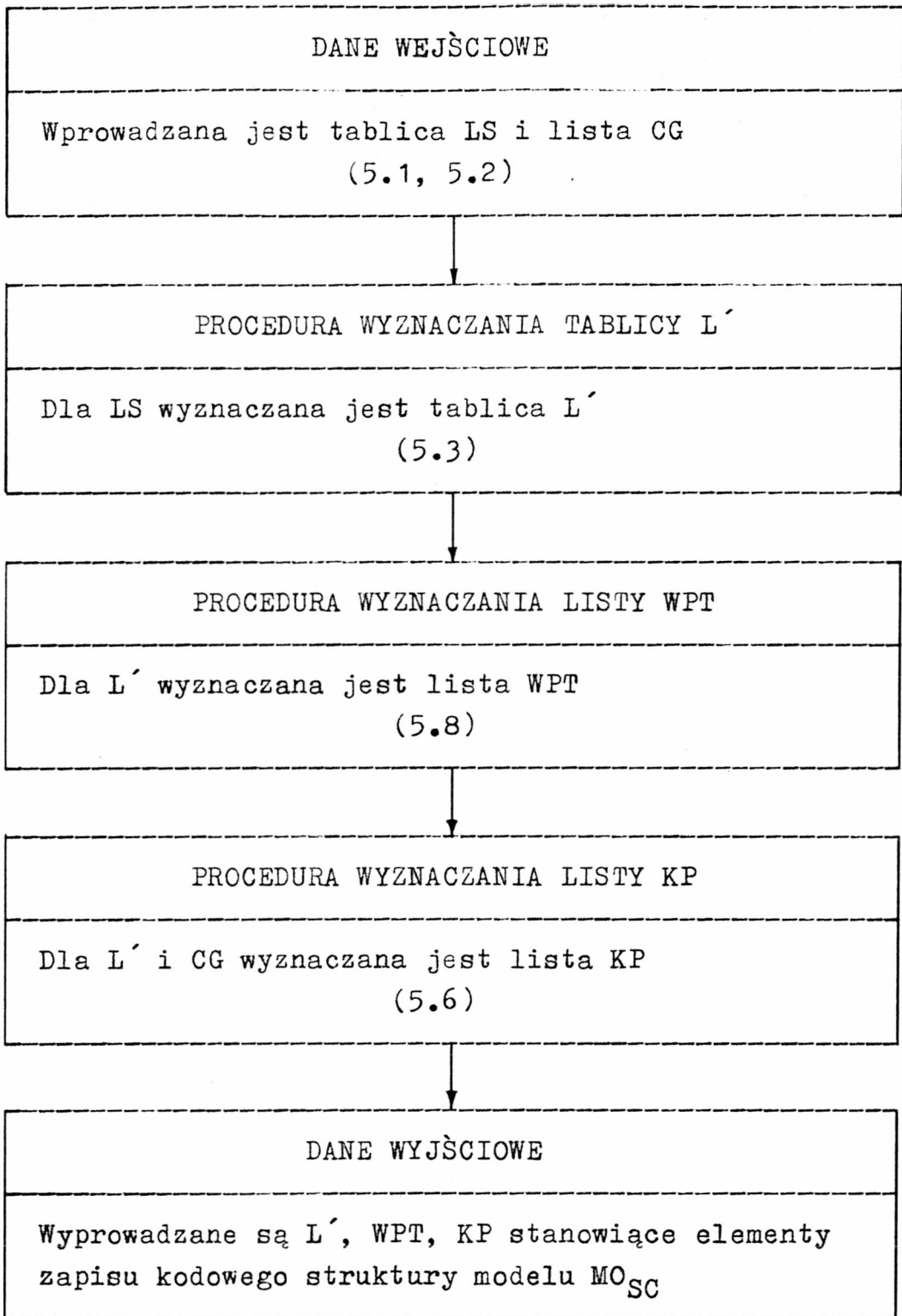
$$(5.11) \text{ GNK} = (\overline{x_i | i \in 1, \|G\|}), \text{ gdzie } x_i = 0 \\ \text{jeżeli } g_i \in G^N, x_i = 1 \text{ jeżeli } g_i \in G^K,$$

$$(5.12) \text{ SW} = (\overline{\text{SW}_i | i \in 1, \|T\|}), \\ \text{SW}_i = (\overline{((x_k^1, y_k^1) | k \in 1, \|Z_{t_i}^1\|) | l \in 1, \|Z_{t_i}^1\|}), \\ \text{gdzie } (x_k^1, y_k^1) \text{ indeksy miejsc składających się na k-tą} \\ \text{parę l-tej strefy przejścia } t_i.$$

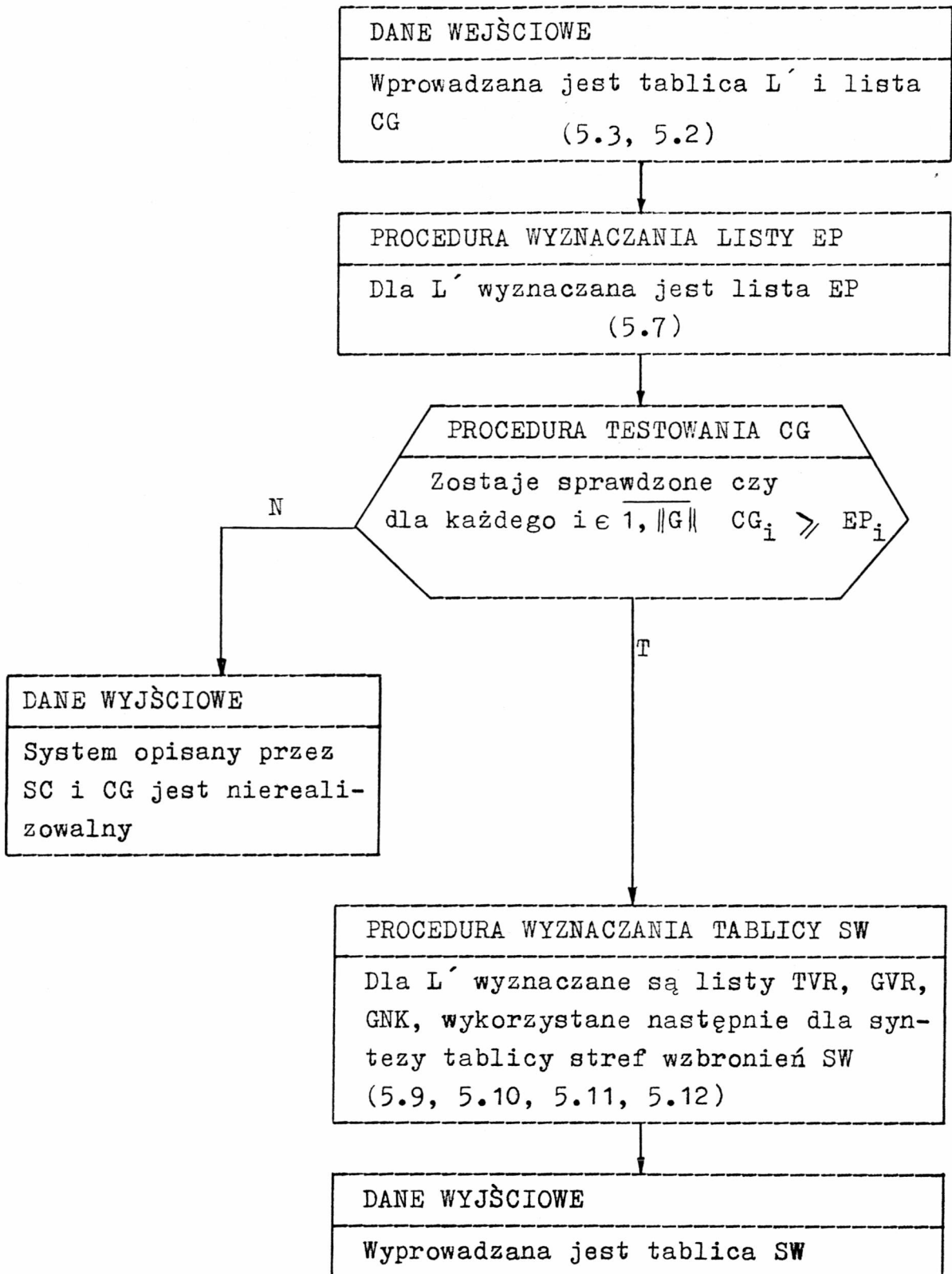
Algorytm transformacji danych opisujących parę (S, C) w dane opisujące strukturę systemu MD_{SC} przedstawia Rys. 5.1, a Rys. 5.2, 5.3, 5.4 jego poszczególne procedury. Obliczenia dokonywane w blokach poszczególnych procedur są bezpośrednią implementacją definicji (wyszczególnionych wcześniej przy określaniu struktur danych).

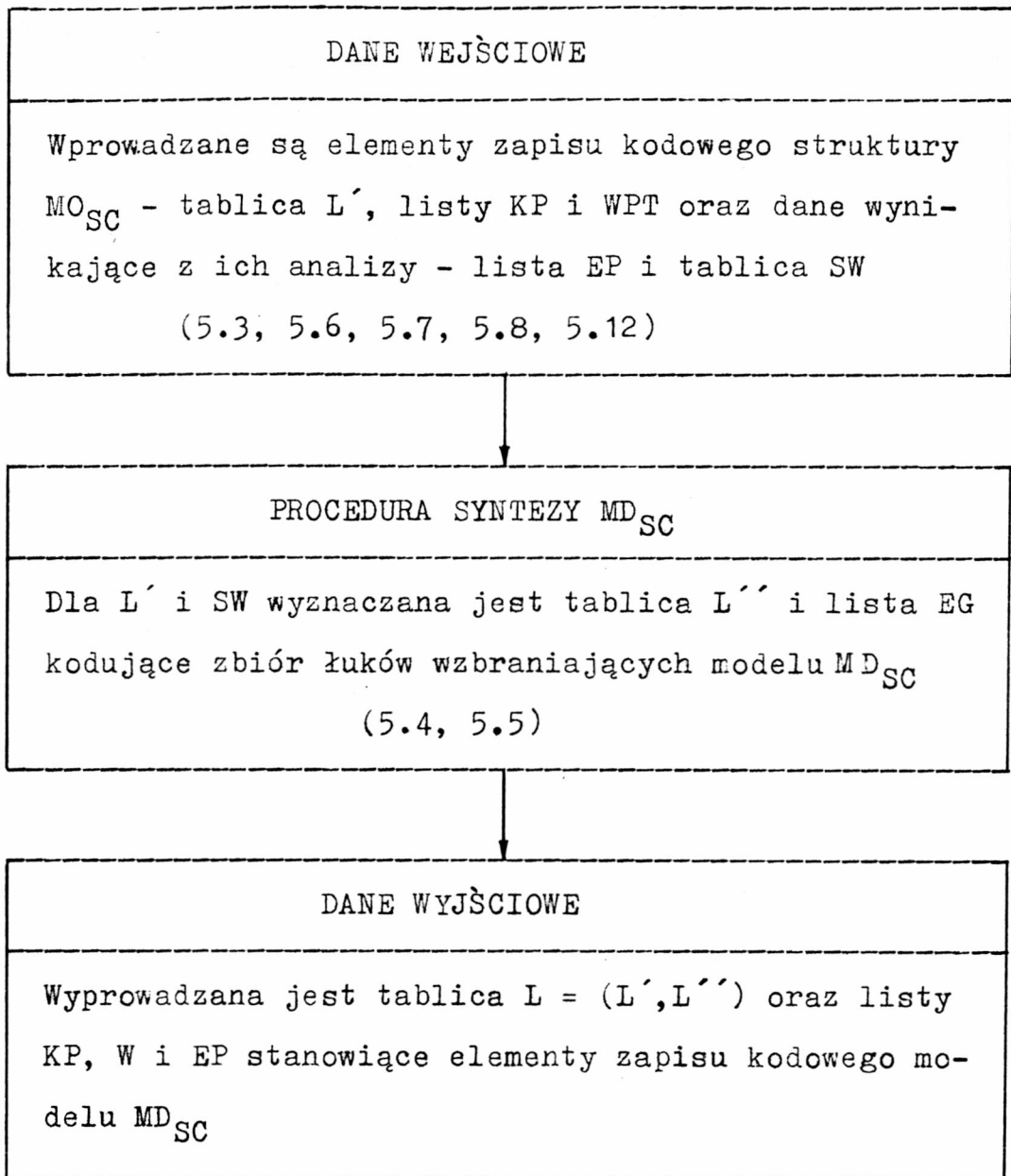
Przedstawione procedury wyznaczają przyjęty sposób automatycznej syntezy struktury systemu MD_{SC} (i jednocześnie ME_{SC}) stanowiącego model dopuszczalnych realizacji procesów należących do systemu WZPP. Model ten nie zawiera parametrów określających czas trwania zdarzeń lub odstępów czasowych między zdarzeniami. Oznacza to, że może być wykorzystany w konstrukcji systemu sterowania w czasie rzeczywistym (gdzie zdarzenia występują w czasie nieznanym z góry) jak i w systemie symulacyjnym (gdzie można założyć dowolne wartości parametrów czasowych).

Rys. 5.1. Schemat algorytmu syntezy modelu MD_{SC}



Rys. 5.2. Schemat procedury syntezy modelu MO_{SC}

Rys. 5.3. Schemat procedury analizy MO_{SC}



Rys. 5.4. Schemat procedury syntezy modelu MD_{SC}

5.2. Model czasowy

Przekształcenie systemu MD_{SC} do ME_{SC} pozwoliło nam wyróżnić oprócz pasywnych stanów procesów (oczekiwania na realizację operacji) także stany aktywne (wykonywania operacji). Zauważmy, że model ten nie definiuje momentów czasowych poszczególnych zdarzeń, powodujących zmiany stanu systemu, a narzuca jedynie ograniczenia na ich dopuszczalną kolejność. W systemach sterowania procesami rzeczywistymi momenty te odpowiadają chwilom występowania zdarzeń A' , A'' , A''' w systemie produkcyjnym. Wykorzystanie zbudowanego modelu dla celów symulacji komputerowej wymaga jego uzupełnienia o parametr czasu (pozwalający uwzględnić długość trwania poszczególnych operacji), co wiąże się z przekształceniem ME_{SC} do odpowiedniego modelu czasowego. Jego konstrukcję oprzemy na następujących założeniach.

Biorąc pod uwagę, że drugi z etapów palenia przejścia reprezentuje jedynie czynności właściwe dla danej operacji, możemy uznać odstęp czasowy pomiędzy zdarzeniami A'' i A''' za stały parametr danego przejścia. Etap pierwszy natomiast, rozpoczyna się od zdarzenia A' odpowiadającego podjęciu decyzji o numerze kolejno realizowanej operacji i przydziału niezbędnego dla jej realizacji zasobów. W zakres czynności tego etapu mogą zatem wchodzić również czynności przygotowawcze, jak np. dojazd robota, wymiana narzędzia, itp., zależne od aktualnego stanu ESP.

Odcinek czasu pomiędzy zapaleniem przejścia, a zmianą etapu jego palenia (zdarzeniami A' i A'') uznamy zatem za funkcję dwóch zmiennych - przejścia oraz sekwencji zdarzeń poprzedzającej stan, w którym przejście to zostaje zapalone. Ponadto, przyjmując, że czas przetwarzania danych bieżących (w systemie sterowania ESP) jest pomijalnie mały w stosunku do czasu realizacji

operacji technologicznych założymy, że zdarzenia nie będące ze sobą w konflikcie mogą zachodzić równocześnie. W oparciu o powyższe uwagi model czasowy systemu WZPP możemy zdefiniować następująco.

Definicja 5.1

Modelem czasowym systemu WZPP będziemy nazywać parę

$ME_{SC}\tau = (ME_{SC}, \tau)$, gdzie

(1) ME_{SC} jest systemem etapowym zbudowanym na MD_{SC} , jak w Def. 4.13, 4.17, 4.18,

(2) $\tau = (\tau_1, \tau_2)$ jest parą funkcji, gdzie:

(2.1) $\tau_1 : T \times \Sigma(ME_{SC}) \rightarrow R^+$ jest funkcją częściową określoną dla dowolnej pary $(t, \tilde{\sigma})$ takiej, że

$\tilde{\sigma} = Ms_0 t_1 Ms_1 t_2 Ms_2 \dots t_n Ms_n$ i t jest przygotowane w Ms_n , wyznaczającą czas pierwszego etapu palenia przejścia t w przypadku zapalenia go w stanie Ms_n osiągniętym w wyniku sekwencji zdarzeń $\tilde{\sigma}$,

(2.2) $\tau_2 : T \rightarrow R^+$ jest funkcją określającą czas drugiego etapu palenia przejść.

Definicja 5.2

Sekwencją czasową w systemie $ME_{SC}\tau = (ME_{SC}, \tau)$ będziemy nazywać sekwencję

$$\sigma u = Ms_0(t_1, u_1)Ms_1(t_2, u_2)Ms_2 \dots (t_n, u_n)Ms_n,$$

taką, że:

(1) $\tilde{\sigma} = Ms_0 t_1 Ms_1 t_2 Ms_2 \dots t_n Ms_n$ jest sekwencją zdarzeń w ME_{SC} ,

(2) $u_1 u_2 \dots u_n$ jest niemalejącym ciągiem liczb naturalnych określających momenty występowania kolejnych zdarzeń sekwencji $\tilde{\sigma}$,

(3) dla każdej pary $i, j \in \overline{1, n}$ takiej, że $(j > i) \& (t_i = t_j) \& (\forall k \in \overline{i+1, j-1}) (t_i \neq t_j)$ spełnione są implikacje:

$$(3.1) \quad s_{i-1}(t_i) = 0 \Rightarrow u_j = u_i + \tau_1(t_i, i-1),$$

$$(3.2) \quad s_{i-1}(t_i) = 1 \Rightarrow u_j = u_i + \tau_2(t_i).$$

Interpretując powyższą definicję w sposób nieformalny, przez sekwencję czasową będziemy rozumieć "historię" działania systemu ME_{SC} opisaną poprzez jego kolejne stany Ms_i generowane w wyniku wystąpienia zdarzeń t_i (zapalenia, zmiany etapu palenia lub wygaszenia przejścia t_i - w zależności od wartości funkcji $s_{i-1}(t_i)$) w momentach czasu u_i .

Zauważmy, że warunki (2) i (3) Definicji 5.2 ograniczają w stosunku do systemu ME_{SC} dopuszczalną kolejność zdarzeń, stanowiąc wymaganie aby jako pierwsze, spośród przejść palących się w Ms_i , zostało wygaszone lub zmieniło etap palenia to przejście t , dla którego moment zachodzenia tej zmiany jest najbliższy. Biorąc pod uwagę, że jedynym warunkiem na to, aby przejście mogło zmienić etap palenia lub zostać wygaszone jest, aby przejście to aktualnie paliło się (Def. 4.18), spełnienie warunków powyższej definicji jest możliwe dla dowolnych wartości funkcji τ .

5.3. Sterowanie działaniem systemu WZPP

Bezblokadowy charakter działania systemu ME_{SC} , a zatem i $ME_{SC}\tau$, pozwala na rozważanie dowolnie długich sekwencji czasowych. Z praktycznego punktu widzenia interesować nas będzie klasa sekwencji odpowiadających realizacji zadań produkcyjnych opisanych wektorem powtórzeń $d = d_1, d_2, \dots, d_v$, specyfikującym wielkości serii produkcyjnych poszczególnych współbieżnie wykonywanych typów wyrobów. Oznacza to, że każda operacja $t \in T_{\pi_i}$ procesu π_i należącego do zbioru procesów $\Pi = \{\pi_1, \pi_2, \dots, \pi_v\}$,

opisanych specyfikacją S , zostanie wykonana d_i razy. Ponieważ z wykonaniem każdej operacji $t \in T$ łączymy trzy zdarzenia, rozważana sekwencja czasowa σu będzie składać się z $3 \sum_{i \in \overline{1, v}} d_i$ zdarzeń.

Automatyczna generacja sekwencji czasowych, reprezentujących wykonanie określonego zadania, wymaga zadeklarowania algorytmicznej reguły pozwalającej na bieżący wybór inicjowanej operacji, tzn. przejścia, którego zapalenie będzie prowadzić do kolejnego stanu systemu (zdarzenia wynikające z realizacji operacji będą następować w odpowiednich odstępach czasowych od jej zainicjowania).

Będziemy zakładać, że wybór powyższy dokonywany jest poprzez poszukiwanie przejścia, którego priorytet oceniany przy zastosowaniu pewnego zbioru funkcji kryterialnych $X = \{x_1, x_2, \dots, x_m\}$ jest najwyższy (najniższy). Ich wartości mogą mieć charakter statyczny, tzn. zależeć od wartości parametrów S, c, τ_2, d (ewentualnie τ_1 , jeżeli funkcja τ_1 ma charakter niezmienny względem σ), lub dynamiczny, tzn. zależeć od pewnych danych związanych z aktualnym stanem procesów i ich wcześniejszym przebiegiem.

W ogólności funkcja kryterialna może być zatem zdefiniowana następująco.

Definicja 5.3

I-tą funkcję kryterialną, $x_i \in X$, w systemie $ME_{SC\tau}$ będziemy nazywać funkcję częściową $x_i : PR \times \Sigma(ME_{SC}) \times T \rightarrow R^+$ określoną dla trójek $(PR, \sigma u, t)$ takich, że PR jest wektorem wartości parametrów S, C, τ, d , $\sigma u = Ms_0(t_1, u_1)Ms_1(t_1, u_2)Ms_2 \dots (t_n, u_n)Ms_n$ jest sekwencją czasową, t jest przejściem przygotowanym w Ms_n .

Będziemy żądać aby w zbiorze X istniała co najmniej jedna funkcja x_i spełniająca, dla każdego $PR \in PR$ i każdego

$\sigma u \in \Sigma(ME_{SC\gamma})$, warunek jednoznaczności wyboru ($t \neq t'$) $\Rightarrow$
 $\Rightarrow x_i(PR, \sigma u, t) \neq x_i(PR, \sigma u, t')$. Ustalona funkcję $x^* \in X$
spełniającą powyższą własność będziemy nazywać funkcją bazową
w X .

Reguły wyboru określone dla zbioru X mogą różnić się wartościami parametrów opisanych poniższą specyfikacją.

Definicja 5.4

Specyfikacja reguły wyboru określonej dla zbioru funkcji kryterialnych X jest skończoną sekwencją
 $rx = (ext_1, j_1)(ext_2, j_2) \dots (ext_k, j_k)$, gdzie
 $j_i \in \overline{1, m}$ jest indeksem funkcji $x_{j_i} \in X$,
 ext_i jest symbolem należącym do zbioru $\{max, min\}$, $i \in \overline{1, k}$.

Specyfikacja powyższa określa regułę prowadzącą do wyboru $t \in T$ poprzez poszukiwanie podzbiorów przejść ekstremalizujących kolejne funkcje $x_{j_1}, x_{j_2}, \dots, x_{j_k}$. Pierwszy krok poszukiwań dotyczy podzbioru przejść przygotowanych w rozważanym stanie Ms . W krokach następnych zbiory przejść są zawężane do zbioru ekstremalizującego funkcję rozważaną w kroku poprzednim. Poszukiwania są przerywane w momencie uzyskania wyniku jednoznacznego (rozważana funkcja posiada jedno ekstremum). Jeżeli po przejściu całej sekwencji rx brak jest jednoznacznego wyniku, wówczas ostateczna decyzja podejmowana jest w oparciu o zadaną funkcję bazową x^* .

Należy zauważyć, że te same cele produkcyjne mogą być osiągnięte w różnych systemach ESP i w oparciu o różne sposoby ich realizacji. W kategoriach wykorzystywanego modelu możliwości te odzwierciedlane są przez istnienie różnych sekwencji czasowych σu .

Analiza możliwych alternatywnych realizacji procesów wymaga zdefiniowania pewnego zbioru miar umożliwiających wielokryterialną ilościową ocenę efektywności funkcjonowania ESP.

Definicja 5.6

Wskaźnikiem efektywności działania systemu $ME_{SC\tau}$ będziemy nazywać funkcję częściową $\varphi : PR \times \Sigma(ME_{SC\tau}) \rightarrow R^+ \cup \{0\}$ określoną dla par (PR, σ_u) takich, że PR jest wektorem wartości parametrów S, C, τ, d systemu $ME_{SC\tau}$, a σ_u jego sekwencją czasową.

Sekwencje czasowe $\sigma_u \in \Sigma(MO_{SC})$ stanowiące argumenty wyżej zdefiniowanej funkcji φ nie są zmiennymi niezależnymi systemu. Przeciwnie, są one funkcjami zależnymi tak od wartości parametrów PR , jak i reguły wyboru oraz dodatkowych czynników, które mogą tu być uwzględniane, np. momentów wprowadzania partii produkcyjnych do modelowanego ESP, okresowej niesprawności jego komponentów itp. Ponieważ zależność σ_u od w/w zmiennych nie wyraża się bezpośrednią formułą algebraiczną, analiza ich wpływu na funkcję efektywności φ wymaga rekurencyjnej generacji kolejnych elementów sekwencji czasowych. Oznacza to zatem konieczność budowy odpowiedniego systemu komputerowo wspomaganego modelowania i efektywności funkcjonowania ESP.

6. KOMPUTEROWY SYSTEM MODELOWANIA I OCENY EFEKTYWNOŚCI FUNKCJONOWANIA ESP

Adekwatność modelu ME_{SC} względem przyjętych w rozdziale 4.2 założeń dotyczących wymaganej synchronizacji procesów wytwórczych pozwala na jego bezpośrednią implementację w systemie sterowania operatywnego ESP. Co więcej, możliwość automatycznej syntezy modelu dla dowolnie zadanej pary (S,C) oraz poprawność realizacji procesów dla dowolnych sekwencji zdarzeń pozwalają na konstrukcję systemu sterowania o uniwersalnym wykorzystaniu, tj. zdolnego do sterowania przebiegiem dowolnych, w ramach klasy WZPP, procesów wytwórczych, w oparciu o dowolną regułę wyboru i w dowolnym systemie bądź podsystemie ESP, w którym procesy te mogą występować.

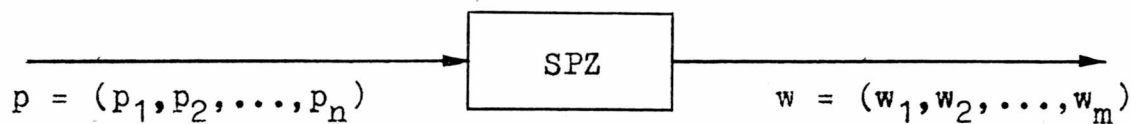
Opracowanie takiego systemu jest istotne z dwóch względów. Jego implementacja w warunkach rzeczywistych znosi konieczność strukturalnego przeprogramowywania systemu sterowania przy każdorazowej zmianie zadań produkcyjnych (zastępując powyższe czynności interakcyjną specyfikacją danych przez operatora), podczas gdy system taki wykorzystany w warunkach symulacyjnych umożliwia wiarygodną (opartą na tym samym modelu) i wielokrotną (dla różnych parametrów) predykcję przebiegu procesów produkcyjnych, a co za tym idzie eksperymentalną optymalizację efektywności ESP.

Przedmiotem dalszej części pracy jest koncepcja systemu SPZ (Symulatora Procesów Złożonych) [159], obrazująca drugie z w/w zastosowań modelu ME_{SC} . Należy jednak podkreślić, że ogólna ideologia konstrukcji systemu symulacyjnego może być bezpośrednio przeniesiona dla celów konstrukcji uniwersalnego systemu sterowania operatywnego nadzorującego procesy rzeczywiste,

z wymianą jedynie procedur specyficznych dla obu systemów, tj. procedur obliczania funkcji czasowych na procedurę wymiany informacji z otoczeniem rzeczywistym.

6.1. Charakterystyka użytkowa

Podstawową ideą konstrukcji pakietu SPZ jest stworzenie zautomatyzowanego narzędzia symulacyjnej predykcji działania ESP posiadającego charakter uniwersalny (tj. znajdującego zastosowanie w różnych systemach i realizujących różne zadania produkcyjne), a jednocześnie nie wymagającego od użytkownika znajomości modelu symulacyjnego jak i konieczności programowania proceduralnego. Konsekwencją powyższych założeń jest opracowanie systemu stanowiącego pewien obiekt (Rys. 6.1) transformujący określony strukturalnie (standardowej postaci) wektor parametrów, których wartości specyfikowane są przez użytkownika, stanowiący opis zadania produkcyjnego i warunków jego realizacji, w wektor wyników umożliwiając wielokryterialną ocenę ilościową funkcjonowania modelowanego obiektu.



Rys. 6.1. Ogólna koncepcja użytkowa pakietu SPZ

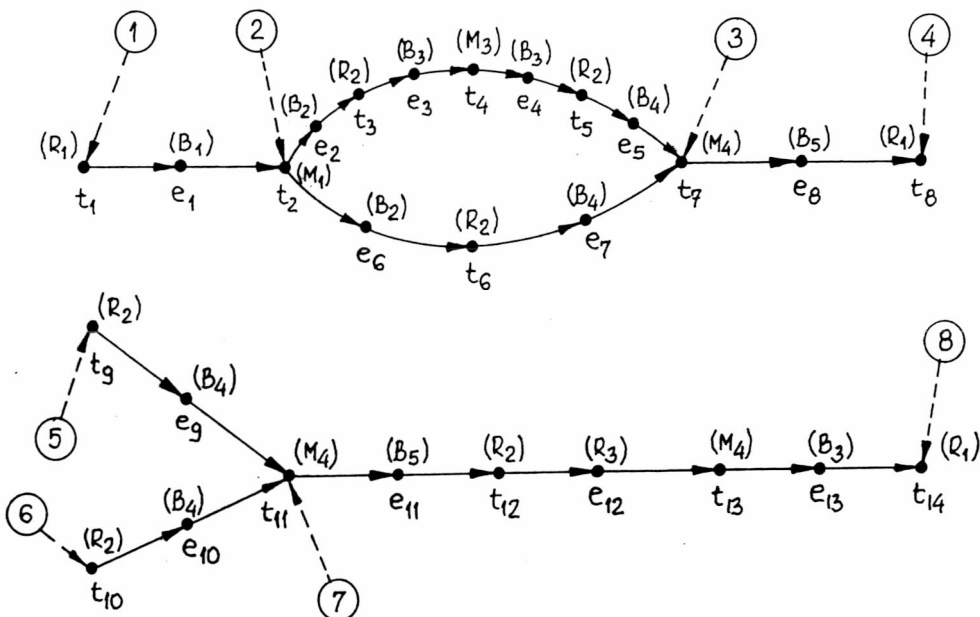
Opisując swój problem użytkownik określa następujące parametry:

- S - specyfikację procesów produkcyjnych, tzn. ich strukturę i wymagania zasobowe,
- d - wielkości serii i/lub partii wyrobów dla poszczególnych podzadania (procesów),

q - czasy realizacji operacji,
 l - odległości między urządzeniami technologicznymi systemu,
 v - prędkości ruchów robotów (urządzeń transportowych),
 C - pojemności magazynów,
 r - regułę wyboru.

Jak wspomniano w rozdziale 5.1 specyfikacja procesów produkcyjnych S zadawana jest w postaci zbioru tabel deklarujących marszruty technologiczne poszczególnych podprocesów liniowych składających się na proces złożony π , oddzielnie dla każdego π należącego do zbioru Π procesów współbieżnych. Porządek występowania procesów liniowych wynika z zadanych numerów ich wierzchołków początkowych i końcowych. Dodatkowo rozróżnia się tu realizatory wykonujące operacje technologiczne (oznaczane symbolami $M_1, M_2, \dots$) i operacje transportowe (oznaczane symbolami $R_1, R_2, \dots$).

Dla zachowania jednolitości zapisu magazyny oznaczane są również dużymi literami ($B_1, B_2, \dots$). Przykładowo:



proces pierwszy		
nr wierzchołka początkowego	marszruta procesu składowego	nr wierzchołka końcowego
1	$R_1 B_1 M_1$	2
2	$M_1 B_2 R_2 R_3 M_3 B_3 R_2 B_4 M_4$	3
2	$M_1 B_2 R_2 B_4 M_4$	3
3	$M_4 B_5 R_1$	4

proces drugi		
nr wierzchołka początkowego	marszruta procesu składowego	nr wierzchołka końcowego
5	$R_2 B_4 M_4$	7
6	$R_2 B_4 M_4$	7
7	$M_4 B_5 R_2 B_3 M_3 B_3 R_1$	8

Specyfikacja procesów w postaci tabel marszrut technologicznych procesów składowych

Rozróżnienie urządzeń technologicznych i transportowych wynika z innego sposobu naliczania w trakcie symulacji czasu przydziału zasobów procesom. Czas realizacji operacji technologicznych jest deklarowany explicite przez użytkownika, natomiast czasy operacji transportowych wyznacza system w oparciu o zadane odległości urządzeń technologicznych i magazynów oraz prędkości ruchu robotów (realizujących operacje transportowe), wliczając w to ruch jałowy - czas dojazdu, zależny od aktualnej pozycji robota.

W aktualnej wersji systemu SPZ użytkownik może skorzystać z jednej z siedmiu reguł wyboru (inaczej reguł priorytetowa-

nia) w jakie został wyposażony pakiet. Zestaw ten obejmuje:

- (1) FIFO (First In - First Out). Ze wszystkich przejść tworzona jest kolejka obsługi. W stanie początkowym kolejka ta zachowuje porządek α^* , natomiast po każdorazowym zapaleniu przejścia, jest ono usuwane i wstawiane na koniec kolejki. Do palenia jest zawsze wybierane pierwsze przygotowane przejście znajdujące się najbliżej początku kolejki.
- (2) LIFO (Least In - First Out). Do palenia wybierane jest przejście o najwyższym numerze spośród przejść przygotowanych, przy czym numeracja przejść zachowuje porządek α^* .
- (3) RRS (reguła równomiernego spływu). Każdemu z przejść przyporządkowana zostaje wartość będąca sumą czasów operacji następujących po operacji reprezentowanej przez to przejście w procesie liniowym, do którego operacja ta należy. Ze wszystkich przejść tworzona jest kolejka, która w stanie początkowym zachowuje niemalejący porządek powyższych wartości. Do palenia wybierane jest pierwsze przygotowane przejście z kolejki po czym, analogicznie jak w FIFO, jest usuwane na jej koniec.
- (4) RPK (reguła podobnego końca). Wybór przejścia do palenia dokonywany jest w dwóch etapach. W fazie pierwszej wyznaczany jest proces, który posiada przygotowane przejście i dla którego różnica pomiędzy zadaną liczbą powtórzeń (wielkością serii wyrobów), a liczbą rozpoczętych wykonań jest największa. W fazie drugiej (wybór przejścia spośród przejść wybranego procesu) realizowana jest reguła FIFO, w ograniczeniu do danego procesu.
- (5) RZM (reguła równomiernego zapełniania magazynów). Dla każdego z magazynów obliczana jest dynamicznie różnica między

jego pojemnością, a aktualnym wypełnieniem. Spośród przejść przygotowanych do palenia wybierane jest przejście, którego magazyn wejściowy cechuje najmniejsza wartość tejże różnicy. W przypadku niejednoznaczności wyboru decyduje reguła FIFO.

- (6) RNJ (reguła najkrótszego czasu). Do palenia wybierane jest przejście reprezentujące operację o najkrótszym czasie realizacji. W przypadku niejednoznaczności decyduje reguła FIFO.
- (7) RND (reguła najdłuższego czasu). Do palenia wybierane jest przejście reprezentujące operację o najdłuższym czasie realizacji. W przypadku niejednoznaczności decyduje reguła FIFO.

Reguły powyższe można przedstawić w ustrukturalizowanej formie.

W oparciu o wyspecyfikowane przez użytkownika parametry tworzony jest model symulacyjny (model czasowy systemu WZPP) procesów produkcyjnych, a następnie prowadzona jest symulacja ich przebiegu z bieżącą wizualizacją pracy realizatorów (ruchome wykresy Gantt'a).

Właściwym efektem symulacji jest ilościowa ocena predyko- wanego działania systemu produkcyjnego, która w aktualnej im- plementacji obejmuje wskaźniki efektywności wymienione w Tabe- li 6.1.

System umożliwia także ingerencję użytkownika w proces symulacji poprzez jego przerwanie, dokonanie zmian wartości po- czątkowo deklarowanych parametrów i ponowne wznowienie przebie- gu.

Wskaźniki efektywności funkcjonowania modelowanego przez SPZ systemu

Wskaźnik efektywności	Sposób naliczania
całkowity czas produkcji	$TP = \max_{i \in \overline{1,v}} (mm(i, d_i)) - \min_{i \in \overline{1,v}} (m(i, 1))$
czas realizacji procesu i-tego	$T_i = mm(i, d_i) - m(i, 1)$
średni czas cyklu	$T_c = \frac{TP}{d}$
średni czas cyklu dla procesu i-tego	$T_{c_i} = \frac{T_{p_i}}{d_i}$
średnia produkcja w toku	$P_t = \frac{1}{TP} \sum_{i \in \overline{1,v}} \sum_{j \in \overline{1, d_i}} (mm(i, j) - m(i, j))$
średnia produkcja w toku dla procesu i-tego	$P_{t_i} = \frac{1}{T_{p_i}} \sum_{j \in \overline{1, d_i}} (mm(i, j) - m(i, j))$
średnie wykorzystanie maszyny k-tej	$WM_k = \frac{1}{TP} \sum_{i \in \overline{1,v}} OM(i, k)$
średnie wykorzystanie maszyny k-tej w i-tym procesie	$WM_{ki} = \frac{OM(i, k)}{T_{p_i}}$
średnie wypełnienie magazynu k-tego	$PB_k = \frac{1}{TP} \sum_{i \in \overline{1, TP}} WB(i, k)$
średnie wykorzystanie magazynu k-tego	$WB_k = \frac{PB_k}{C(B_k)}$

Legenda:

1, 2, ..., v - nr-y procesów,

d_i - długość serii i-tego procesu,

$d = \sum_{i=1}^v d_i$,

$m(i, j)$ - moment rozpoczęcia i-tej realizacji (j-tego elementu) i-tego procesu,

$mm(i, j)$ - moment zakończenia j-tej realizacji i-tego procesu,

- OM(i,k) - obciążenie maszyny M_k w procesie i-tym (sumaryczny czas realizacji operacji procesu i-tego przez M_k),
- OR(i,k) - obciążenie robota R_k w procesie i-tym (bez czasów dojazdu),
- OD(i,k) - obciążenie jałowe robota R_k w procesie i-tym (sumaryczny czas dojazdu dla realizacji operacji procesu i-tego),
- WB(l,k) - liczba przedmiotów przechowywanych w magazynie B_k w odcinku czasu $l-1, l$,
- $C(B_k)$ - pojemność magazynu B_k .

6.2. Struktura i działanie

Konstrukcja pakietu SPZ obejmuje zespół modułów, na który składają się następujące elementy:

A. Bloki programowe:

- BKU - komunikacji z użytkownikiem,
- ED1-ED7 - edycji danych,
- BSM - syntezy modelu,
- BS - symulacji,
- BST - statystyk;

B. Biblioteka procedur obejmująca:

- PSM - procedury syntezy modelu,
- PAB - procedury analizy bieżącej modelu,
- PNS - procedury generacji następnego stanu,
- PT, PU, PUU - procedury obliczeniowe funkcji czasowych,
- PW - procedury wyboru,
- PNZ - procedurę najwcześniejszego zdarzenia,
- PGW - procedurę gromadzenia wyników,
- POS - procedury naliczania statystyk (wskaźników efektywności),
- PG - procedury graficzne,
- POP - procedurę obsługi przerwania;

C. dynamiczna baza danych globalnych (dostępnych dla wszystkich modułów i procedur programu) gromadzącą dane edytowane przez użytkownika, opisujące wygenerowany model oraz wyniki symulacji wymagane dla obliczania wskaźników efektywności, jak również służącą jako obszar komunikacji (przekazywania danych bieżących) między modułami względnie procedurami systemu.

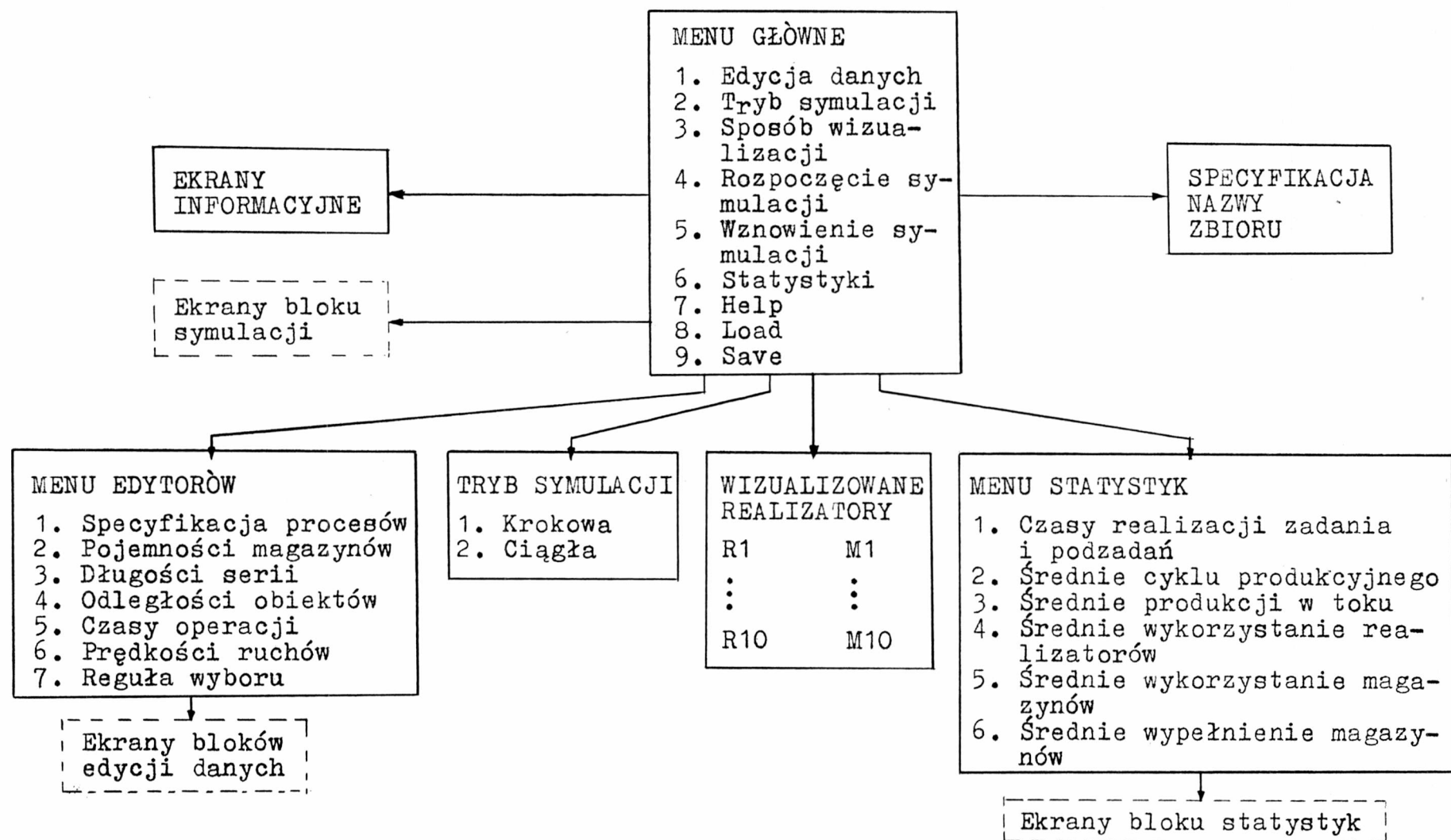
6.2.1. Blok komunikacji z użytkownikiem

Blok komunikacji z użytkownikiem BKU jest częścią programu inicjowaną automatycznie po włączeniu systemu. Jego zadaniem jest informacja o przeznaczeniu SZP oraz rodzaju działań jakie może podjąć użytkownik przy współpracy z systemem, a następnie pośredniczenie w ich wyborze. Zadanie to realizowane jest w oparciu o dwupoziomowe MENU (Rys. 6.2) obejmujące edycję danych, deklarację opcji symulacji, inicjację lub wznowienie procesu symulacji, odczyt statystyk oraz żądania HELP (wświetlenie ekranów informacyjnych) i LOAD/SAVE (wprowadzanie/wyprowadzanie danych do/z zbiorów dyskowych).

Zlecenia 2, 3, 7, 8, 9 (patrz Rys. 6.2) realizowane są bezpośrednio w bloku BKU, natomiast pozostałe powodują przejście do odpowiednich modułów wykonawczych.

6.2.2. Bloki edycji danych

SPZ wyposażony jest w 7 bloków edycji danych inicjowanych opcjonalnie zgodnie z wybraną przez użytkownika pozycją w MENU EDYTORÓW (Rys. 6.2). Każdy z bloków ED1 + ED7 pełni tę samą



Rys. 6.2. Struktura ekranów bloku komunikacji z użytkownikiem

funkcję różniąc się jedynie rodzajem wpisywanych danych. Ich zadaniem jest maksymalne uproszczenie specyfikacji wartości parametrów przez użytkownika, weryfikacja ich poprawności, oraz generacja odpowiednich tablic w bazie danych przechowujących zadeklarowane wartości.

Prostota edycji danych wynika z wyposażenia każdego z bloków w ekrany opisujące dany parametr, na których użytkownik posługując się kursorem wybiera właściwe pola i definiuje z klawiatury pożądaną ich zawartość.

W każdym z bloków dostępna jest także opcja HELP precyzująca przeznaczenie, dopuszczalny zakres zmienności i sposób specyfikacji określonego parametru. Weryfikacja wprowadzanych danych dotyczy ich przynależności do dopuszczalnych przedziałów liczbowych, a w przypadku parametru S także jego poprawności syntaktycznej. Wystąpienie błędu jest sygnalizowane w postaci odpowiedniego komunikatu, natomiast deklaracje prawidłowe wraz z pewnymi pośrednio wynikającymi z nich wielkościami (np. liczba użytych realizatorów, liczba operacji itp.) zostają zapamiętane w bazie danych.

Edytory służą zarówno do specyfikacji wstępnej jak i wprowadzania zmian w trakcie symulacji (po jej przerwaniu). Po zakończeniu edycji użytkownik ma możliwość powrotu do menu.

6.2.3. Blok syntezy modelu

Blok syntezy modelu BSM wywoływany jest z bloku BKU w wyniku wyboru przez użytkownika opcji, "rozpoczęcie symulacji", względnie "wznowienie symulacji" o ile po przerwaniu zostały zmienione parametry decydujące o modelu, tj. S lub C. Po zakoń-

czeniu właściwego dla bloku BSM zadania następuje powrót do bloku BKU, który z kolei inicjuje blok symulacji; synteza modelu jest zatem dla użytkownika działaniem przezroczystym.

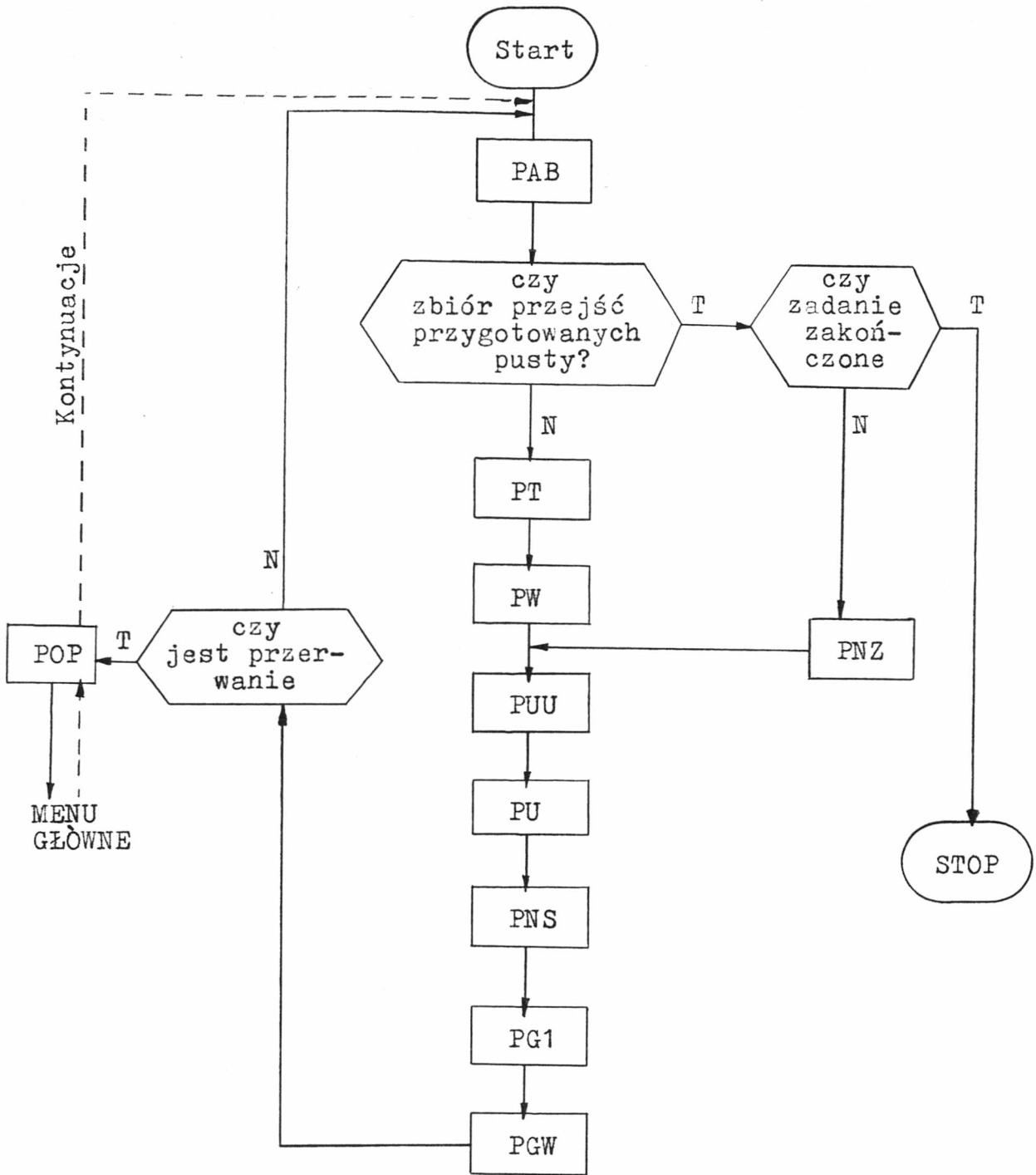
Zadaniem programu BSM jest sterowanie procesem syntezy modelu, tj. wywoływanie odpowiednich procedur bloku PSM, które realizują właściwe czynności obliczeniowe pobierając dane wejściowe a następnie składając wyniki swojego działania z/do wspólnej dla całego programu bazy danych.

6.2.4. Blok symulacji

W związku z przyjętym modelem procesów produkcyjnych, symulacja ich przebiegu sprowadza się do generacji sekwencji czasowej systemu $ME_{SC\tau}$ w oparciu o jego parametry strukturalne wyznaczane w bloku syntezy modelu oraz zadeklarowane bezpośrednio przez użytkownika parametry q , l , v (decydujące o wartościach funkcji τ) i regułę wyboru.

Pod względem treści programowej blok symulacji BS stanowi pętlę, z której wywoływane są procedury realizujące określone grupy czynności dla kolejnych stanów systemu. W każdym cyklu wyznaczane są (Rys. 6.3):

- zbiór przejść przygotowanych (procedura PAB),
- czasy palenia powyższych przejść (procedura PT),
- spośród przejść wybranych powyżej, przejście na które wskazuje ustalona reguła wyboru (procedura PW),
- jeżeli zbiór wyznaczony przez PAB jest pusty - spośród przejść palących się przejście najwcześniej kończące swój aktualny etap palenia (procedura PNZ),



Rys. 6.3. Schemat działania bloku symulacji

- moment zakończenia palenia wybranego przez PW lub PNZ przejścia (procedura PUU),
- aktualny czas działania systemu (procedura PU),
- następny stan systemu (procedura PNS).

Ponadto procedura PG1 generuje zmiany ekranu wizualizującego proces symulacji, oraz procedura PGW dokonuje wstępnego przetworzenia danych w oparciu o które naliczane są później wyniki statystyczne.

W przypadku chwilowego wstrzymania przez użytkownika procesu symulacji wywoływana jest procedura obsługi przerwania, z której możliwe jest wywołanie menu głównego i przejście do edytorów (celem zmiany wstępnie ustalonych parametrów) i dalsza kontynuacja procesu.

6.2.5. Blok statystyk

Blok ten składa się z dwóch funkcjonalnie różnych części - podbloku wywoływanego automatycznie przez system po zakończeniu procesu symulacji, zadaniem którego jest dokonanie w oparciu o procedury POS i zgromadzone w bazie danych wyniki końcowego naliczenia statystyk, oraz podbloków wywoływanych przez użytkownika z MENU STATYSTYK, powodujących wyświetlenie odpowiednich ekranów zawierających tabele uzyskanych danych statystycznych.

Dzięki swojej modularnej strukturze SPZ jest łatwo podatny na ewentualne zmiany i rozszerzenia dotyczące np. funkcji kryterialnych, struktury reguły wyboru, zależności czasowych operacji itp., a rozbudowany system komunikacji z użytkownikiem wsparty opcjami HELP czyni go prostym w użyciu nawet dla osób nie mających żadnej praktyki w pracy z komputerem.

6.3. Przykłady zastosowań

Rozważany jest elastyczny system montażowy, przedstawiony na Rys. 6.4, składający się z:

- stacji montażu komponentów M_1 ,
- stacji smarowania M_2 ,
- stacji montażu całości M_3 ,
- robota transportowego R ,
- stacji paletyzacji/depaletyzacji przedmiotów pracy D/P .

Stacje M_1 i M_2 wyposażone są w pojedyncze magazyny przystanowiskowe odpowiednio B_1 i B_2 , natomiast stacja M_3 w dwa takie magazyny - B_3 i B_4 .

Przedmiotem badań symulacyjnych jest współbieżna realizacja dwóch potokowych procesów produkcyjnych opisanych specyfikacją przedstawioną schematycznie na Rys. 6.5.

W przeprowadzonych eksperymentach przyjęto jednakowe i niezmiennie czasy realizacji operacji technologicznych oraz prędkość transportu elementów przez robota (odpowiednio $q = 10$ i $v = 5$, w jednostkach umownych), natomiast badaniu poddano zależność całkowitego czasu produkcji i średniego cyklu produkcji od reguły pierwszeństwa i pojemności magazynów, z uwzględnieniem różnych długości serii produkcyjnych.

Eksperyment 1

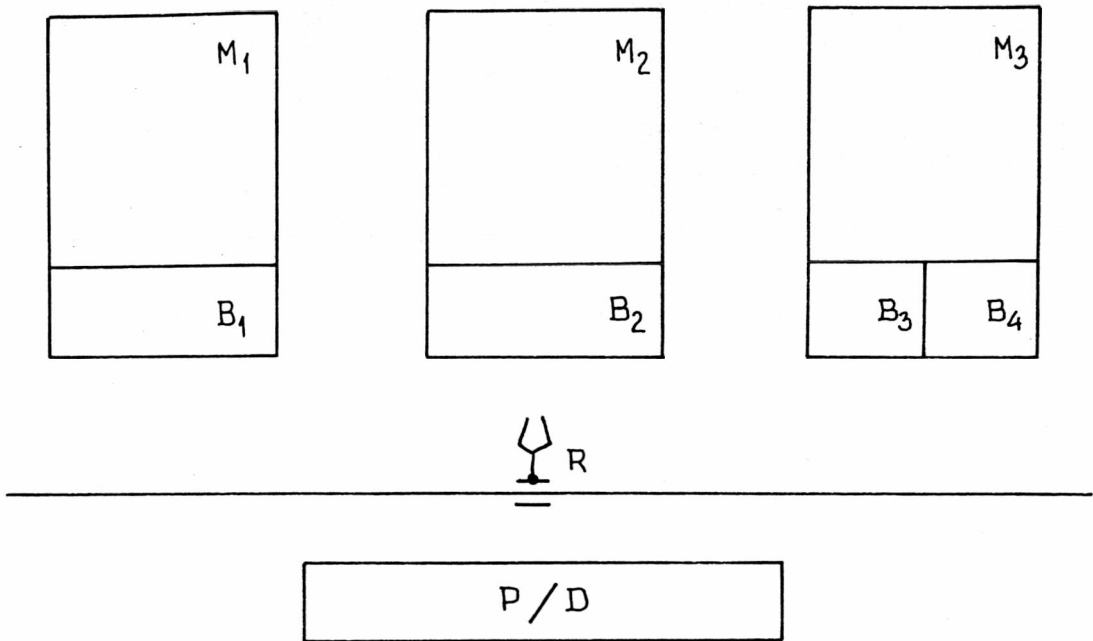
Badanie wpływu reguł wyboru na całkowity czas produkcji przy parametrach:

(A) Krótkie serie produkcyjne ($d_1 = d_2 = 5$) oraz:

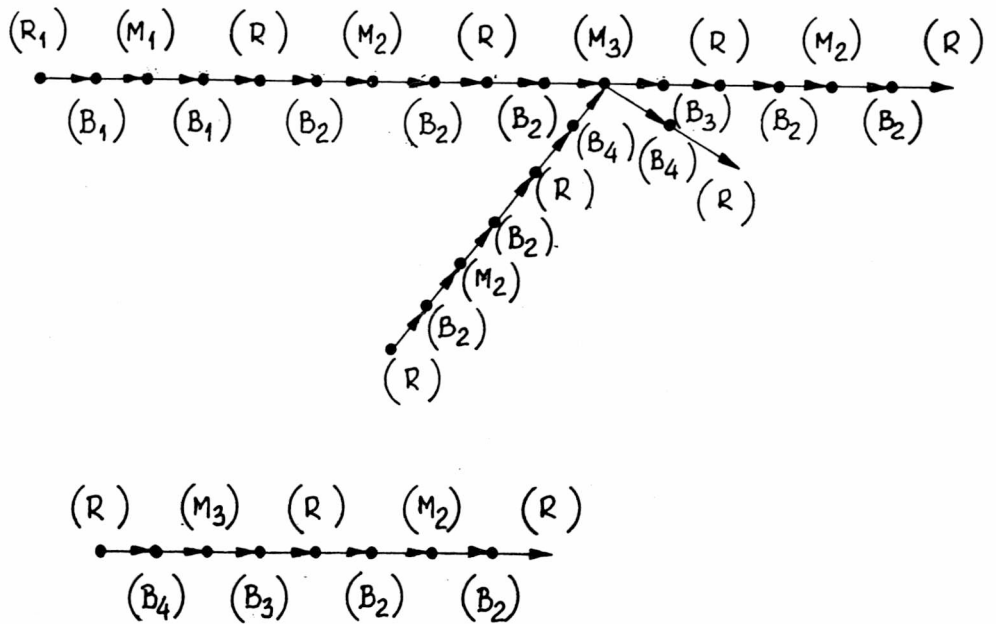
(a) małe pojemności magazynów:

$$C(B_1) = 1, \quad C(B_2) = 4, \quad C(B_3) = C(B_4) = 3,$$

(b) pojemności magazynów pięciokrotnie wyższe niż w (a);



Rys. 6.4. Elastyczny system montażowy będący obiektem badań symulacyjnych



Rys. 6.5. Specyfikacja symulowanych procesów produkcyjnych

- (B) dłuższe serie produkcyjne ($d_1 = d_2 = 30$), pojemności magazynów jak w przypadku (A).

Wyniki symulacji

(A)

całkowity czas pro- dukcji reguła	TP(a)	TP(b)	TP(b)/TP(a)
FIFO	327	280	0.86
LIFO	373	282	0.75
RPK	330	291	0.88
RZM	335	286	0.85
RRS	384	384	1.0

(B)

całkowity czas pro- dukcji reguła	TP(a)	TP(b)	TP(b)/TP(a)
FIFO	1902	1607	0.84
LIFO	2114	1832	0.87
RPK	1955	1716	0.88
RZM	1985	1652	0.83
RRS	2005	2205	1.0

Wnioski

- (1) We wszystkich rozważanych przypadkach najkrótszy czas produkcji osiągany jest przy regule FIFO. Należy zauważyć, że przy długości serii produkcyjnych $d = 5$ pięciokrotne zwiększenie pojemności magazynów pozwala na skrócenie produkcji o około 14%, a przy $d = 30$ o około 16%.

W obu powyższych przypadkach długości serii produkcyjnych, a zatem i wielkość produkcji w toku są mniejsze od sumarycznej pojemności magazynów systemu (nazywanej dalej pojemnością systemu).

- (2) Wpływ pozostałych reguł na czas produkcji w funkcji pojemności magazynów i długości serii produkcyjnych jest bardzo nieregularny. Przykładowo dla $d = 5$ pięciokrotny wzrost pojemności magazynów powoduje, że czas produkcji uzyskany przy regule LIFO jest mniejszy od czasu uzyskiwanego przy regułach RPK i RZM. Natomiast przy $d = 30$ czas produkcji przy regule LIFO jest dłuższy niż przy RPK i RZM.

Z kolei, przy początkowo zadanych pojemnościach magazynów w obu przypadkach, $d = 5$ i $d = 30$, czasy produkcji uzyskiwane przy zastosowaniu reguły RPK są krótsze niż dla RZM. Pięciokrotny wzrost pojemności magazynów powoduje, że reguła RZM pozwala osiągnąć krótsze czasy niż reguła RPK.

Przedstawione uwagi nasuwają wniosek, że zależność łącząca regułę i czas produkcji ma charakter nieregularny w sytuacjach, gdy sumaryczna długość serii nie przekracza sumarycznej pojemności magazynów (wielkość produkcji w toku jest mniejsza od pojemności systemu). W przypadku stałej pojemności magazynów i długościach serii przekraczających pojemność systemu badane zależności wykazują większą regularność tzn. czasy produkcji uzyskiwane przy różnych regułach priorytetowania rosną proporcjonalnie do długości serii produkcyjnych.

Eksperyment 2

Badanie wpływu reguł pierwszeństwa na średnie czasy cyklu produkcyjnego przy parametrach jak w eksperymencie 1.

Wyniki

(A)

średni czas cyklu w procesie 1-szym reguła	$Tc_1^{(a)}$	$Tc_1^{(b)}$	$Tc^{(b)}/Tc_1^{(a)}$
FIFO	73	95	1.3
LIFO	96	213	2.2
RPK	71.2	92	1.3
RZM	72.2	138	1.9
RRS	96.4	96	1

średni czas cyklu w procesie 2-gim reguła	$Tc_2^{(a)}$	$Tc_2^{(b)}$	$Tc_2^{(b)}/Tc_2^{(a)}$
FIFO	36.2	56.6	1.56
LIFO	35	98.6	2.8
RPK	35	46	1.3
RZM	36	51.2	1.4
RRS	40.2	40.2	1

średni czas cyklu reguła	$Tc^{(a)}$	$Tc^{(b)}$	$Tc^{(b)}/Tc^{(a)}$
FIFO	54.6	75	1.37
LIFO	65.5	156	2.4
RPK	53.1	70	1.3
RZM	54.1	95	1.76
RRS	68.3	68	1

(B)

średni czas cyklu w pro- cesie 1-szym reguła	$Tc_1^{(a)}$	$Tc_1^{(b)}$	$Tc^{(b)}/Tc^{(a)}$
LIFO	73.8	102	1.4
FIFO	96.6	578	6
RPK	71.9	95	1.3
RZM	72.9	650	8.9
RRS	97.2	97	1

średni czas cyklu w procesie 2-gim reguła	$Tc_2^{(a)}$	$Tc_2^{(b)}$	$Tc_2^{(b)}/Tc_2^{(a)}$
LIFO	36	59	1.6
FIFO	37.8	173	4.6
RPK	35	44	1.26
RZM	36	83	2.3
RRS	39.5	40	1

średni czas cyklu reguła	$Tc^{(a)}$	$Tc^{(b)}$	$Tc^{(b)}/Tc^{(a)}$
LIFO	54.9	81	1.47
FIFO	67.2	376	5.6
RPK	53.4	70	1.3
RZM	54.4	367	6.74
RRS	68.4	68	1

- (1) Z przedstawionych rezultatów wynika, że żadna z badanych reguł nie minimalizuje średniego czasu cyklu produkcji T_c we wszystkich czterech analizowanych przypadkach. Dla małych pojemności magazynów najkrótszy cykl produkcji pozwala uzyskać reguła RPK. Natomiast przy pięciokrotnym zwiększeniu pojemności magazynów lepsze rezultaty daje RRS. Zysk ten, w porównaniu z regułą RPK, jest jednak na tyle nieduży (około 2%), że za "najlepszą" we wszystkich przypadkach można uznać regułę RPK. Dla długości serii $d = 5$ i $d = 30$, pięciokrotne zwiększenie pojemności magazynów, przy zastosowaniu reguły RPK, powoduje zwiększenie średniego czasu cyklu produkcji o około 30%.
- (2) Spośród pozostałych badanych reguł wyboru małe czasy cyklu pozwala osiągnąć również reguła FIFO. Dla małych pojemności magazynów wzrost czasu cyklu w stosunku do "optymalnej" reguły RPK wynosi około 1%, dla pięciokrotnie wyższych (w stosunku do "optymalnej" w tym przypadku RRS) około 10% i 12%, odpowiednio dla $d = 5$ i $d = 30$.
- (3) Szczególną nieregularność wpływu na średni czas cyklu wykazuje reguła LIFO, która przy małej pojemności magazynów pozwala uzyskać mniejszą wartość wskaźnika T_c niż reguła RRS ("optymalna" dla dużych pojemności magazynów). Wykorzystanie reguły LIFO przy pięciokrotnym zwiększeniu pojemności magazynów powoduje wzrost czasu cyklu o 240% i 560%, odpowiednio dla $d = 5$ i $d = 30$. Podobny charakter wykazuje reguła RZM zwiększając T_c odpowiednio o 176% i 674%. Sytuacja taka wynika z zasady "tłoczenia" realizowanej przez powyższe reguły, co zwiększa wartość produkcji w toku (a

wraz z nią średni czas cyklu) w przypadku dużych pojemności magazynów i dłuższych serii produkcyjnych.

- (4) Analizując oddzielnie średni czas cyklu Tc_1 dla wyrobów pierwszego procesu i Tc_2 dla wyrobów drugiego procesu można zauważyć, że w prawie wszystkich przypadkach "najlepszą" regułą jest RPK. Jedynie dla dużych pojemności magazynów, w wyniku zastosowania reguły RRS, można uzyskać skrócenie Tc_2 o około 9% i 11% (odpowiednio dla $d = 5$ i $d = 30$). Zysk ten jest jednak opłacony zwiększeniem w nieco mniejszej proporcji czasu Tc_1 .

Przeprowadzone eksperymenty 1 i 2 wskazują w szczególności, że reguła dająca najkrótsze czasy produkcji (FIFO) jest różna od reguły prowadzącej w większości przypadków do najkrótszych cykli produkcyjnych (reguły RPK). Biorąc jednakże pod uwagę, że wzrost Tc , wynikający z zastosowania FIFO, w stosunku do Tc uzyskiwanego dla RPK, nie jest duży, można uznać, że (z dokładnością do badanych przypadków) zadowalające działanie rozważanych procesów pod względem obu analizowanych kryteriów jakości daje reguła FIFO.

Eksperyment 3

Badanie wrażliwości systemu (pod względem całkowitego czasu produkcji i średnich czasów cykli) na zmiany pojemności najczęściej wykorzystywanego magazynu (patrz Rys. 6.5), tj. B_2 , w przypadkach:

- (A1) reguła pierwszeństwa FIFO, małe pojemności pozostałych magazynów ($C(B_1) = 1$, $C(B_3) = C(B_4) = 3$) oraz trzech różnych długości serii produkcyjnych:

$$- d_1 = d_2 = 5,$$

$$- d_1 = d_2 = 10,$$

$$- d_1 = d_2 = 30;$$

(A2) jak w przypadku (A1), ale przy dużych pojemnościach pozostałych magazynów ($C(B_1) = C(B_3) = C(B_4) = 40$);

(B1) jak w przypadku (A1), ale dla reguły pierwszeństwa LIFO;

(B2) jak w przypadku (A2), ale dla reguły pierwszeństwa LIFO.

Otrzymane wyniki przedstawiają poniższe tabele i wykresy (Rys. 6.6 - Rys. 6.11).

(A1)

$C(B_2)$	TP	Tc_1	Tc_2	Tc
$d_1 = d_2 = 5$				
4	327	73	36	55
6	340	105	50	78
10	229	160	46	103
20	329	160	46	103
40	329	160	46	103
$d_1 = d_2 = 10$				
4	642	74	36	55
6	670	109	52	81
10	692	269	44	156
20	663	278	42	159
40	663	278	40	159
$d_1 = d_2 = 30$				
4	1902	74	36	55
6	1990	112	53	83
10	2013	383	41	212
20	1966	682	39	360
40	1985	762	41	402

(A2)

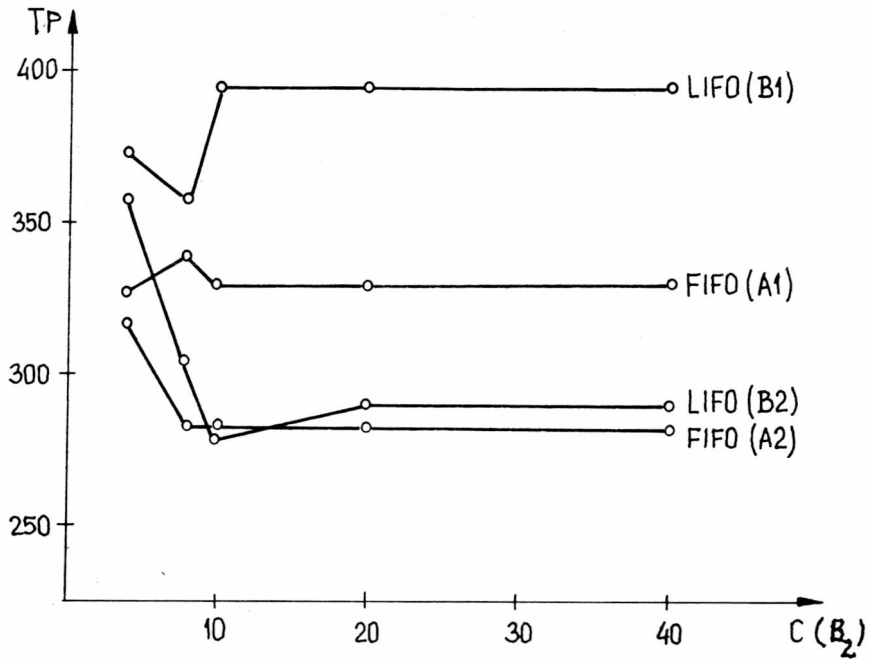
$C(B_2)$	TP	Tc_1	Tc_2	Tc
$d_1 = d_2 = 5$				
4	317	134	79	106
6	282	95	76	76
10	282	95	76	76
20	282	95	76	76
40	282	95	76	76
$d_1 = d_2 = 10$				
4	515	197	99	148
6	547	99	58	79
10	547	99	58	79
20	547	99	58	79
40	547	99	58	79
$d_1 = d_2 = 30$				
4	1824	410	108	259
6	1607	102	59	81
10	1607	102	59	81
20	1607	102	59	81
40	1607	102	59	81

(B1)

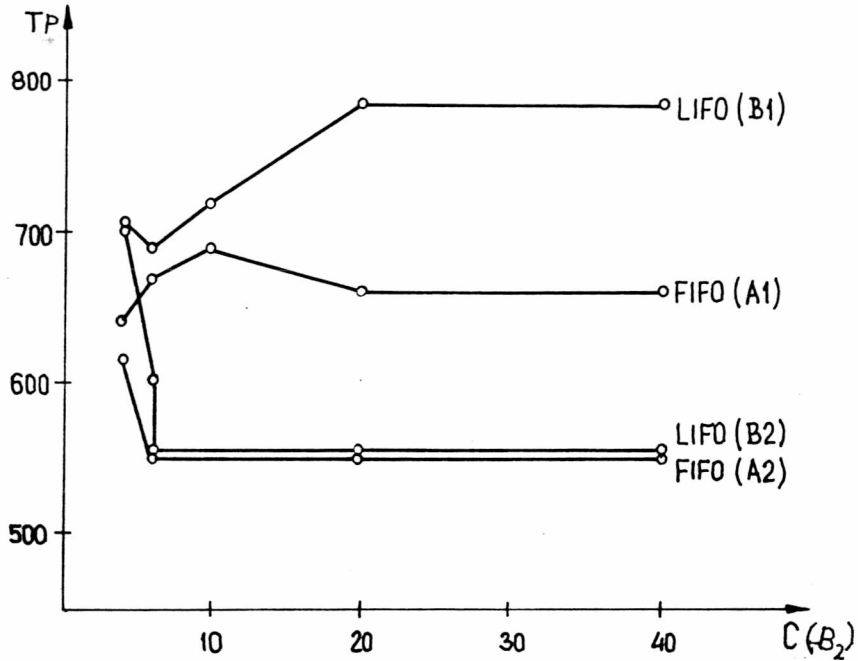
$C(B_2)$	TP	Tc_1	Tc_2	Tc
$d_1 = d_2 = 5$				
4	373	96	35	66
6	358	174	41	107
10	396	264	47	155
20	396	264	47	155
40	396	264	47	155
$d_1 = d_2 = 10$				
4	704	96	37	67
6	687	167	41	104
10	717	363	44	204
20	784	482	46	264
40	784	482	46	264
$d_1 = d_2 = 30$				
4	2114	97	38	67
6	1982	170	41	105
10	2009	400	43	222
20	2119	900	45	473
40	2379	1362	46	704

(B2)

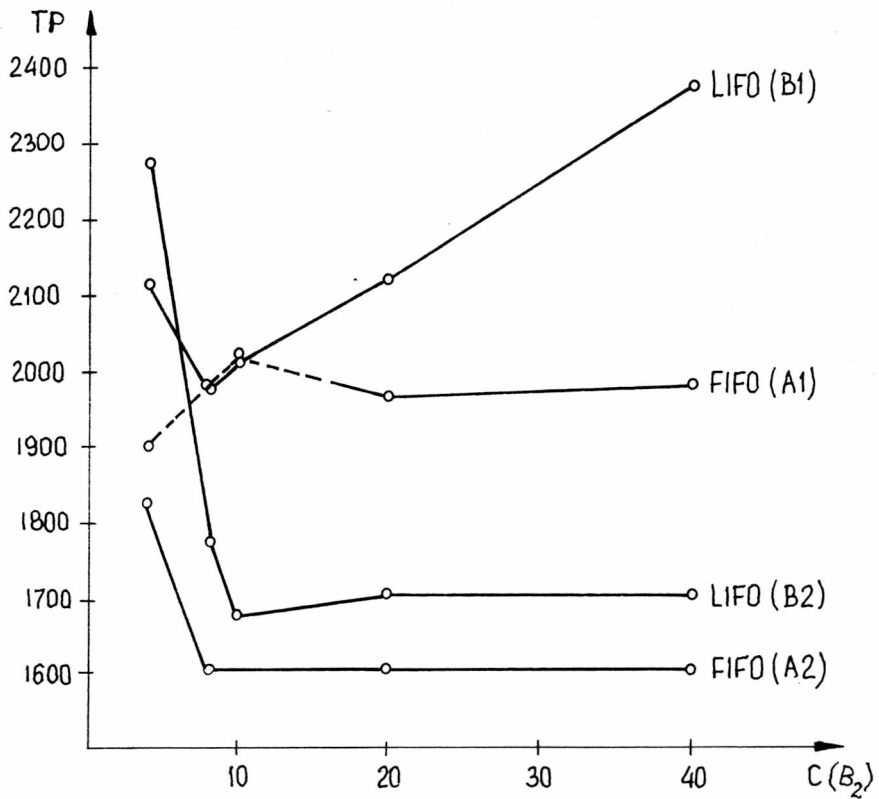
$C(B_2)$	TP	Tc_1	Tc_2	Tc
$d_1 = d_2 = 5$				
4	358	215	96	156
6	302	199	98	149
10	278	205	97	151
20	290	227	108	167
40	290	227	108	167
$d_1 = d_2 = 10$				
4	704	392	198	295
6	600	377	188	282
10	552	331	105	218
20	553	376	114	245
40	553	376	114	245
$d_1 = d_2 = 30$				
4	2275	1168	593	881
6	1778	989	586	787
10	1677	970	350	660
20	1706	1028	292	660
40	1703	1117	333	725



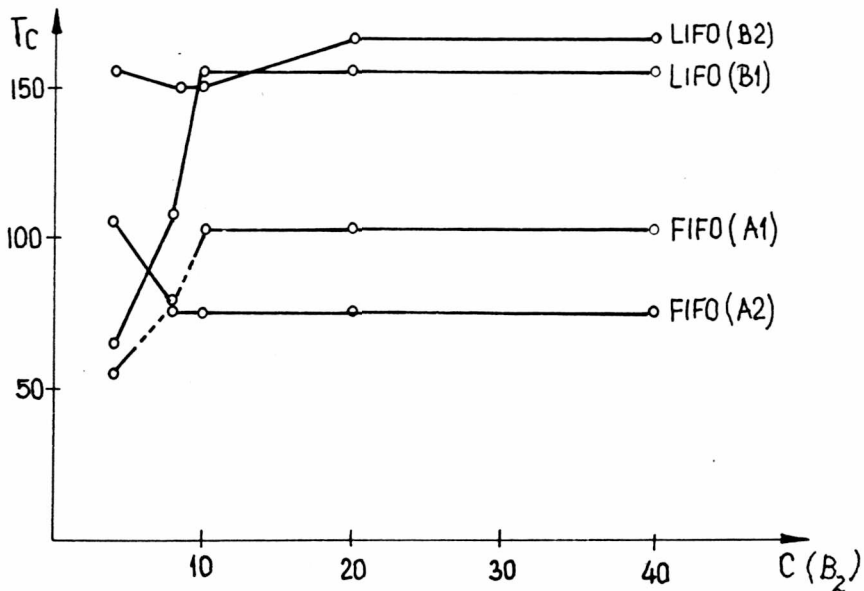
Rys. 6.6. Całkowity czas produkcji w funkcji pojemności magazynu B_2 dla serii produkcyjnych $d_1 = d_2 = 5$



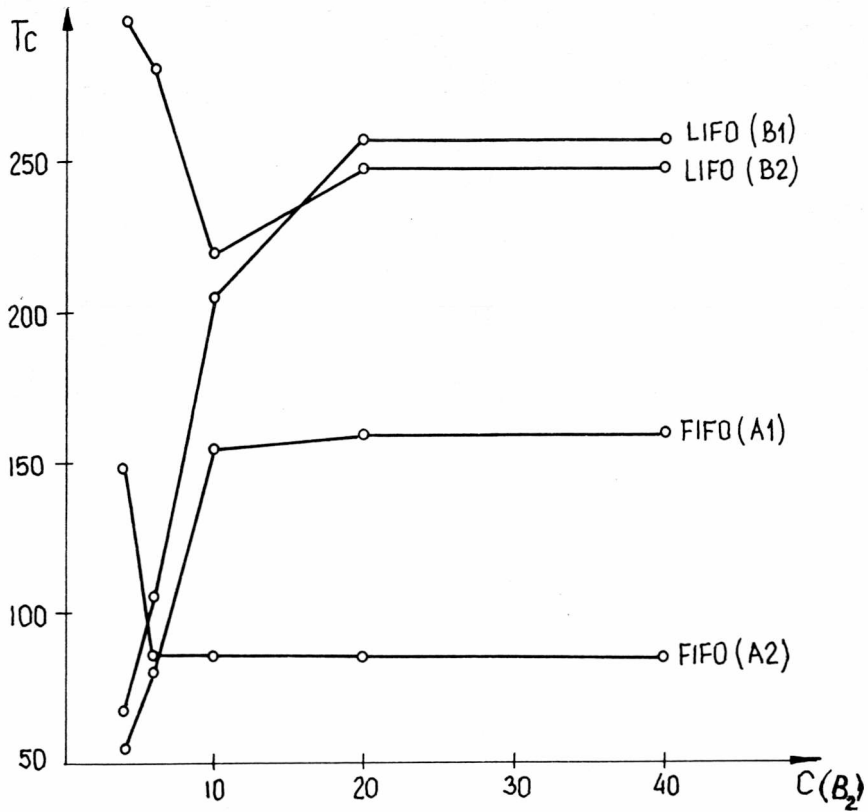
Rys. 6.7. Całkowity czas produkcji w funkcji pojemności magazynu B_2 dla serii produkcyjnych $d_1 = d_2 = 10$



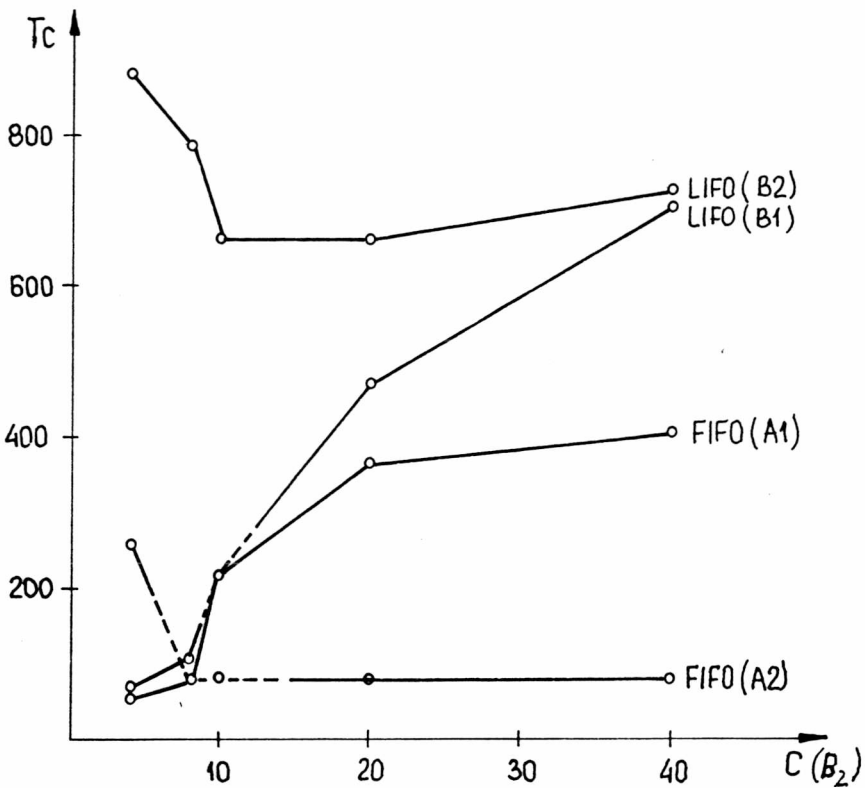
Rys. 6.8. Całkowity czas produkcji w funkcji pojemności magazynu B_2 dla serii produkcyjnych $d_1 = d_2 = 30$



Rys. 6.9. Średni czas cyklu w funkcji pojemności magazynu B_2 dla serii produkcyjnych $d_1 = d_2 = 5$



Rys. 6.10. Średni czas cyklu w funkcji pojemności magazynu B_2 dla serii produkcyjnych $d_1 = d_2 = 10$



Rys. 6.11. Średni czas cyklu w funkcji pojemności magazynu B_2 dla serii produkcyjnych $d_1 = d_2 = 30$

Wnioski

Podobnie jak w eksperymentach 1 i 2 obserwowana jest znaczna nieregularność wpływu reguły wyboru na wskaźniki jakości pracy systemu w przypadku, gdy pojemność systemu jest większa od sumarycznej długości serii. W sytuacjach, gdy sumaryczna długość serii produkcyjnych przekracza pojemność systemu można natomiast zauważyć, że:

- (1) Występuje regularna zależność typu: "reguła wyboru - ekstremalizowany przez nią wskaźnik jakości pracy systemu". Regularność ta oznacza, że wartości wskaźników ekstremalizowanych przez różne reguły są stałe (patrz Rys. 6.6 - 6.10).
- (2) Wzrost długości serii produkcyjnych powoduje zwiększenie całkowitego czasu produkcji, którego najniższe wartości uzyskiwane są przy zastosowaniu reguły FIFO. Zwiększenie pojemności wszystkich magazynów wpływa w sposób istotny na skrócenie całkowitego czasu produkcji. Dla serii produkcyjnych $d = 10$ obniżka ta wynosi 17%, a dla $d = 30$, 19%.
- (3) Wzrost długości serii produkcyjnych pociąga wzrost średniego czasu cyklu produkcyjnego T_c . Najniższą wartość uzyskuje się dla reguły FIFO. Zwiększenie pojemności wszystkich magazynów wpływa w sposób istotny na skrócenie średniego czasu cyklu. Dla serii produkcyjnych $d = 10$ i $d = 30$ obniżka ta wynosi odpowiednio 50% i 80%.
- (4) Analizując oddzielnie poszczególne procesy obserwuje się różny wpływ wzrostu pojemności systemu na średnie czasy cyklu T_{c1} i T_{c2} . Dla obu reguł FIFO i LIFO wzrost pojemności magazynów powoduje obniżenie T_{c1} , natomiast wzrost T_{c2} .

Przykładowo w przypadku $d = 10$ i dla reguły FIFO obniżka Tc_1 i wzrost Tc_2 następuje odpowiednio o 64% i 53%.

Przedstawione eksperymenty ukazują jedynie fragment analizy, jakiej można poddać działanie rozważanych procesów przy pomocy systemu SPZ. Nie wyczerpują także wszystkich możliwych zadań, jakie można rozwiązywać z wykorzystaniem tego systemu. Przykładowo zadania te mogą dotyczyć wyboru marszrut produkcyjnych procesów, rozmieszczenia komponentów ESP, badania wrażliwości wskaźników oceny na zmiany czasów wykonywania operacji, poszukiwania reguły optymalizującej inne wskaźniki oceny jak np. wielkość produkcji w toku, itp.

Opracowany system SPZ stanowi użyteczne narzędzie zarówno na etapie projektowania ESP jak i jego eksploatacji w ograniczeniu do klasy współbieżnych złożonych procesów potokowych; klasa ta obejmuje znaczną liczbę systemów produkcyjnych i procesów w nich realizowanych.

Jego dalsze rozszerzenia będą obejmować uwzględnienie stochastycznego charakteru zmiennych wejściowych, rozszerzenia zbioru reguł wyboru, wprowadzenia bloku planowania eksperymentów, itp.

7. UWAGI KOŃCOWE

Przedstawiona praca wprowadza ogólny sieciowy model dynamiczny szerokiej klasy systemów typu ESP. Model ten umożliwia budowę poprawnych jakościowo (nie prowadzących do powstawania blokad w sterowanych procesach) konkretnych modeli sterowania współbieżnego w czasie rzeczywistym. Uzyskane rezultaty stanowią przyczynek do badań z zakresu automatyzacji programowania systemów sterowania logicznego, znajdując zastosowanie w zadaniach automatycznej syntezy programów sterowania ESP, budowie systemów sterowania adaptacyjnego w czasie rzeczywistym oraz tworzeniu systemów komputerowo wspomaganego modelowania i oceny efektywności funkcjonowania ESP.

Jako aparatus konstrukcji modeli dynamicznych został zastosowany formalizm systemów sieci Petriego typu P/T (miejsce/ /przejście), w oparciu o który zdefiniowano trzy transformacje pozwalające kolejno na:

- syntezę, w oparciu o specyfikację procesów i zadeklarowane parametry ESP, w którym procesy te mają być realizowane, standardowego systemu P/T stanowiącego model synchronizacji procesów współbieżnych, zapewniającego poprawność ich przebiegu, przy założeniu bezblokowalności procesów,

- transformację wyżej wymienionego modelu do systemu P/T z łukami wzbraniającymi i oryginalną regułą przygotowania przejść gwarantującego pożądaną (nie dopuszczającą do powstawania blokad) synchronizację procesów,

- transformację wyżej wymienionego modelu do tzw. modelu etapowego, powstałego w wyniku "rozbitcia" zdarzenia reprezentującego wykonanie operacji na trzy zdarzenia składowe, który (po uzupełnieniu go dowolną regułą sterowania) stanowi poprawny model

sterowania procesami współbieżnymi w czasie rzeczywistym; dla zadanych czasów realizacji poszczególnych działań składowych model ten umożliwia symulacyjną generację przebiegu działania ESP w czasie.

Istotnym elementem przedstawianego podejścia do syntezy sieciowych modeli procedur sterowania współbieżnego jest sformułowanie i wykazanie warunków wystarczających dla unikania blokad w rozważanej klasie procesów (systemach WZPP). Gwarantuje to poprawność opracowanej metody syntezy i w konsekwencji poprawność będących jej efektem procedur sterowania współbieżnego w czasie rzeczywistym.

Opracowana metoda umożliwia budowę algorytmów syntezy modeli sieciowych (o złożoności obliczeniowej ograniczonej wielomianem drugiego stopnia względem rozmiaru specyfikacji procesów S) oraz procedur wykorzystujących powyższy model dla podejmowania decyzji sterujących w czasie rzeczywistym (również o złożoności kwadratowej względem S). Oznacza to praktyczną (przemysłową) przydatność uzyskanych wyników w suboptymalnym rozwiązywaniu problemów które, jak wykazano, mają NP-trudny charakter.

Przedstawiony w pracy system SPZ stanowi komputerową implementację uzyskanych wyników teoretycznych. Przykładowo opisane eksperymenty symulacyjne wskazują na jego przydatność w rozwiązywaniu zadań projektowania i eksploatacji rozważanej klasy systemów produkcyjnych.

Podsumowując, do najważniejszych osiągnięć pracy można zaliczyć:

(1) Przeprowadzenie analizy metod wykorzystywanych w zadaniach projektowania struktur wyposażenia technologicznego ESP oraz planowania i sterowania procesami produkcyjnymi; w oparciu

o wyniki powyższej analizy zaproponowano nową metodę kompleksowego modelowania i eksperymentalnej (symulacyjnej) optymalizacji procesów produkcyjnych, pozwalającą podwyższyć efektywność podejmowania decyzji poprzez jednoczesne uwzględnienie wszystkich czynników określających aktualnie rozpatrywany wariant systemu produkcyjnego.

(2) Opracowanie formalnego aparatu matematycznego dla modelowania dyskretnych procesów produkcyjnych poprzez modyfikację sieci Petriego, co pozwala rozwiązywać zadania syntezy algorytmów sterowania w oparciu o wspólną bazę zunifikowanych modeli dynamicznych reprezentujących wszystkie możliwe realizacje procesów współbieżnych.

(3) Wykazanie NP-trudności problemów decyzyjnych związanych z optymalnym rozstrzygnięciem konfliktów zasobowych w rozważanej klasie procesów współbieżnych, co uzasadnia potrzebę rozwiązań o charakterze suboptymalnym (opartych o warunki wystarczające dla unikania blokad systemu).

(4) Sformułowanie i wykazanie słuszności warunków wystarczających dla uniknięcia blokady w procesach konkurujących o dostęp do wspólnych zasobów systemu w szerszej niż rozpatrywana dotąd klasie procesów współbieżnych (współbieżnych złożonych procesów potokowych); implementacja powyższych wyników w procedurach sterowania pozwala na bardziej efektywne wykorzystanie zasobów w porównaniu ze znanymi metodami synchronizacji procesów współbieżnych.

(5) Opracowanie metody automatycznej syntezy procedur podejmowania decyzji w zadaniach sterowania operatywnego ESP, bazującej na opracowanym dynamicznym modelu sieciowym dyskretnych procesów produkcyjnych, znajdującej praktyczne zastosowanie w zadaniach planowania i sterowania ESP.

(b) Opracowanie pakietu programów użytkowych SPZ stanowiącego system symulacyjny służący do komputerowo wspomaganego modelowania i oceny efektywności funkcjonowania ESP.

Zakres zastosowań otrzymanych w pracy wyników obejmuje znaczną część potrzeb związanych z zagadnieniami automatyzacji programowania, sterowania i zarządzania obiektami klasy ESP.

Wysokie nakłady inwestycyjne związane z powstaniem ESP powodują, że jednym z głównych kryteriów ich konstrukcji i sposobu eksploatacji jest zapewnienie wysokiej efektywności działania systemu ocenianej przy pomocy kryteriów stanowiących, w ogólności, funkcjonały funkcji przebiegu procesów produkcyjnych w czasie. Ze względu na nieustalony charakter tych przebiegów (współbieżna realizacja krótkich serii produkcyjnych), wiarygodna ocena efektywności systemu wymaga możliwie wiarygodnej predykcji ich funkcji czasowych. To, z kolei, ze względu na wysoki stopień integracji elementów decydujący o ich współzależnym działaniu, wymaga dynamicznego modelu ESP pozwalającego na wiarygodne, tj. w oparciu o mechanizm sterowania nadrzędnego w czasie rzeczywistym, wyznaczanie powyższych przebiegów.

Jak wykazała analiza, dotychczas stosowane modele nie umożliwiają tego typu badań ze względu na selektywny charakter reprezentacji ESP pomijający istotne czynniki wpływu na działanie systemu. Dotyczy to przede wszystkim mechanizmu sterowania, pomijanego w modelach optymalizacyjnych i opartych na sieciach kolejek lub ujmowanego w wąskim zakresie konkretnych ESP w przypadku typowych modeli symulacyjnych.

W odróżnieniu od istniejących modeli, model proponowany w pracy ujmuje zmienność struktur i parametrów systemów i procesów produkcyjnych oraz umożliwia generację ich różnych, w uzależnieniu

niu od przyjętej reguły sterowania, przebiegów czasowych. Dzięki udowodnionej analitycznie poprawności zastosowanego mechanizmu sterowania, model powyższy cechuje jakościowa poprawność działania w dowolnym układzie wartości jego zmiennych.

Istotnym kryterium syntezy mechanizmu sterowania jest postulat bezblokowej realizacji procesów produkcyjnych. Spełnienie tego postulatu umożliwia eliminację przestojów systemu, obniżających jego efektywność, oraz interwencji operatora w rozwiązywanie blokad. Przeciwdziałanie blokadom w drodze zapobiegania ich występowaniu łączy się z potencjalnie znacznie wcześniejszą, niż wymagają tego procesy, rezerwacją dla nich zasobów systemu a przez to uniemożliwieniem w tym czasie dostępu do nich innym procesom i w konsekwencji spadkiem ich wykorzystania. Dlatego też w pracy zdecydowano się na metodę unikania blokad, polegającą na wprowadzeniu mechanizmu ograniczającego w każdym ze stanów systemu zbiór dopuszczalnych decyzji przydziału zasobów do tych, które zapewniają bezblokowy przebieg procesów.

Jak wykazano, uzyskanie rozwiązania optymalnego wymagałoby rozwiązywania w czasie rzeczywistym problemów o charakterze NP-trudnym. Opracowany algorytm sterowania, oparty na generowanych modelach sieciowych polega na sprawdzaniu w każdym ze stanów systemu pewnych warunków wystarczających dla uniknięcia blokady, nie zaś dostatecznych i koniecznych. Umożliwia to obniżenie złożoności obliczeń kosztem eliminacji wraz z przebiegami blokadowymi również pewnej części realizacji poprawnych.

Otrzymane wyniki teoretyczne mogą zostać bezpośrednio zastosowane do automatycznej syntezy modeli sterowania w rzeczywistych elastycznych systemach produkcyjnych, jak również, co zostało przedstawione w pracy, w konstrukcji komputerowego systemu mode-

lowania i oceny efektywności funkcjonowania ESP. Opracowana metoda pozwala na generację poprawnych jakościowo modeli symulacyjnych tworzonych w oparciu o dane opisujące strukturę i parametry systemu i procesów produkcyjnych oraz regułę sterowania ich przebiegiem. Dane te deklarowane są w sposób interakcyjny przez użytkownika systemu symulacyjnego. Możliwe jest zatem prowadzenie różnorodnych eksperymentów w zakresie doboru projektu ESP, planu produkcji i sposobu jego realizacji pod kątem określonych potrzeb. Analiza efektywności działania systemu elastycznego może być prowadzona przez osoby (np. inżynierów produkcji) nie posiadające wiedzy teoretycznej analityka systemów i programisty.

Przedstawione podejście umożliwia także budowę trenerów wykorzystywanych w procesie szkolenia przyszłych projektantów i operatorów ESP, automatyzację programowania sterowników logicznych czasu rzeczywistego (program sterowania opracowany w systemie interakcyjnym z komputerem może być przesłany bezpośrednio do pamięci EPROM sterownika przemysłowego) oraz konstrukcję systemów sterowania adaptacyjnego (w przypadku uszkodzenia komponentów systemu lub wprowadzania dodatkowych, nieplanowanych wcześniej zadań istnieje możliwość automatycznej korekty realizowanego programu sterowania nadrzędnego).

Dalszy kierunek badań stanowiących kontynuację niniejszej pracy powinien dotyczyć zarówno aspektów teoretycznych jak i elementów aplikacyjnych.

Do ważniejszych problemów przyszłych badań teoretycznych należy zaliczyć rozszerzenie klasy modelowanych procesów w sposób pozwalający na uwzględnienie sytuacji, w których:

- wykonanie jednej operacji wymaga wielu realizatorów,

- operacja może być wykonana przez jeden z alternatywnych realizatorów,

- przebieg marszruty technologicznej jest ustalany w sposób dynamiczny.

Istotnym zagadnieniem dalszych badań teoretycznych powinno być także poszukiwanie "lepszyc" metod synchronizacji, pozwalających na mniej restrykcyjne ograniczanie zbioru dopuszczalnych realizacji modelowanych procesów.

Ważniejsze zagadnienia o charakterze aplikacyjnym, będące rozszerzeniem wyników uzyskanych w niniejszej pracy, powinny dotyczyć opracowania dalszych wersji systemu SPZ, umożliwiających m.in.:

- rozszerzenie zbioru parametrów określających strukturę i działanie modelowanych obiektów pozwalających uwzględnić np. terminy wprowadzania serii produkcyjnych, niesprawność komponentów systemu, szerszy zakres reguł wyboru, itp.,

- rozważanie stochastycznego charakteru parametrów systemu, np. czasów realizacji operacji, długości serii produkcyjnych, reguł wyboru itp.,

- automatyczną realizację serii zaplanowanych eksperymentów i wnioskowanie o ich wynikach poprzez rozszerzenie SPZ o elementy sztucznej inteligencji.

LITERATURA

1. Abdul-Hussin M.H., The Modelling and Analysis of Flexible Manufacturing Systems Using Petri Nets, Ph.D. Thesis, Politechnika Wrockawska, 1989.
2. Afentakis P., A Model for Layout Design in FMS, Flexible Manufacturing Systems: Methods and Studies, Kusiak A. (ed.), Elsviere Sc. Publ. 1986, pp. 127-140.
3. Afentakis P., Layout Design for Flexible Manufacturing Systems: Models and Strategies, Proc. 2nd ORSA/TIMS Conf. on Flexible Manufacturing Systems, Ann Arbor, USA, 1986, pp. 221-232.
4. Ahmadi J.H., A Decision Support System for Production Planning in Discrete Electronic Parts Manufacturing, Proc. 2nd ORSA/TIMS Conf. on Flexible Manufacturing Systems, Ann Arbor, USA, 1986, pp. 345-358.
5. Alla H., Ladet P., Coloured Petri Nets: A Tool for Modelling, Validation and Simulation of FMS, Flexible Manufacturing Systems: Methods and Studies, Kusiak A. (ed.), Elsvier Sc. Publ., 1986, pp. 271-281.
6. Akella A., Bevans P.J., Choong Y., Simulation of a Flexible Manufacturing System, Report LIDS-P-1435, 1985.
7. Ammons J.C., Govindaraj T., Mitchel C.M., Decision Models for Aiding FMS Scheduling and Control, IEEE Trans. on Systems Man and Cybernetics, Vol. 18 (1988), No. 5, pp. 744-756.
8. Ancelin B., Erschler J., Monteil D., A Simulation Tool for Manufacturing Systems Design and Control Aid, Report L.A.A.S. No. 86 224, 1986.

9. Andreassen M.M., Ahm T., Flexible Assembly Systems, IFS Publications, Springer-Verlag Berlin, Heidelberg, New York, Tokyo, 1988.
10. Archetti F., Performance Evaluation of Flexible Manufacturing Systems Using Petri Net, Proc. Int. Conf. on Applicability of Petri Nets to Operations Research, Milano, Italy, 1986, pp. 128-143.
11. Archetti F., Sciomachen A., Representation, Analysis and Simulation of Manufacturing Systems by Petri Net Based Models, Discrete Event Systems: Models and Applications, Varaiya P., Kurzhanski A.B. (eds.), Springer Verlag 1988.
12. Azadivar F., Optimum Number of Buffer Spaces in Flexible Manufacturing System, Proc. 2nd Int. Conf. on Flexible Manufacturing Systems, Ann Arbor, USA, 1986, pp. 181-190.
13. Banaszak Z., Automatic design of adaptive control algorithms, Automatic Control, Proc. of 10-th Triennial World Congress of IFAC, Munich, July 1987, Pergamon Press 1988, pp. 275-280.
14. Banaszak Z., Roszkowska E., Deadlock avoidance in pipeline concurrent processes, Podstawy Sterowania, Vol. 18, (1988) pp. 3-17.
15. Beck C.L., Krogh B.H., Models for Simulation and Discrete Control of Manufacturing Systems, Proc. IEEE Int. Conf. on Robotics and Automation, San Francisco, USA, 1986, pp. 322-328.
16. Bessant J., Flexible Manufacturing Systems: An Overview, Microelectronics Monitor, 1984/XII.
17. Biggs J.R., Priority Rules for Shop Floor Control in Material Requirements Planning System under Various Levels of Capacity, Int. J. Prod. Res., Vol. 23 (1985), No. 1, pp. 33-46.

18. Błażewicz J., Cellary W., Słowiński R., Węglarz J., *Badania Operacyjne dla Informatyków*, WNT Warszawa, 1983.
19. Bonlioli M., Garetti M., Pozzetti A., *Production Scheduling and Operational Control of Flexible Manufacturing Systems*, Robotica Vol. 3 (1985), Part 4, pp. 233-244.
20. Bourey J.P., Gentina J.C., *Computer Aided Design for Structuration and Representation of Control of Flexible Manufacturing Systems*, Proc. Eighth European Workshop on Application and Theory of Petri Nets, Zaragoza, Spain 24-26 June 1987, pp. 117-135.
21. Branam J.W., *MRP and Schedule Execution in the Job Shop of the Future*, Proc. 4-th Euro. Conf. Automated Manufacturing, Birmingham, V.K., 1987, pp. 413-424.
22. Browne J., Dubois D., Rathmill K., Sethi P.S., Stecke K.E., *Classification of Flexible Manufacturing Systems*, The FMS Magazine, Vol. 2 (1984), No. 2, pp. 114-117.
23. Bruno G. and Biglia P., *Performance Evaluation and Validation of Tool Handling in Flexible Manufacturing Systems Using Petri Nets*, Intern. Workshop on Timed Petri Nets, Torino, Italy, July 1-3, 1985, pp. 64-71.
24. Buckley J., Chan A., Graefe U., Neelamkavil J., Serrerev M., Thomson V., *An Integrated Production Planning and Scheduling System for Manufacturing Plants*, Robotics and CIM, Vol. 4 (1988), No. 3/4, pp. 517-523.
25. Buzacott, J.A., *Optimal Operating Rules for Automated Manufacturing Systems*, IEEE Transactions on Automatic Control, Vol. AC-27, No. 1, (1982), pp. 80-86.
26. Buzacott J.A., *Modelling Flexible Manufacturing Systems*, Operational Research '84, J.P. Brans (ed.), Washington, 1984, pp. 546-560.

27. Buzacott, J.A., Modelling Manufacturing Systems, IFAC 9th Triennial World Congress, Budapest, Hungary, 1984, pp. 2499-2504.
28. Buzacott J.A., Yao D.D., On Queueing Network Models of Flexible Manufacturing Systems, Queueing Systems 1(1986), pp. 5-27.
29. Buzacott J.A., Yao D.D., Flexible Manufacturing Systems: A Review of Models, Working Paper No. 82-007, Department of Industrial Engineering, University of Toronto, Canada, 1982.
30. Buzacott J.A., Yao D.D., Flexible Manufacturing Systems: A Review of Analytical Models, Management Science, Vol. 32, No. 7, July 1986, pp. 890-905.
31. Buzacott J.A., Shanthikimar J.G., Models for Understanding Flexible Manufacturing Systems, AIIE Transactions, December 1980, pp. 339-350.
32. Carlier J., Chretienne Ph., Girault C., Modelling Scheduling Problems with Timed Petri Nets, 4-th European Workshop on Applications and Theory of Petri Nets, Toulouse, France, 1983, pp. 148-167.
33. Carrie A.S., Persopoulos A.C., Operation Sequencing in a FMS, Robotica, Vol. 3 (1985), Part 4, pp. 259-265.
34. Carrie A.S., The Role of Simulation in FMS, Flexible Manufacturing Systems: Methods and Studies, Kusiak A. (ed.), Elsviere Sc. Publ. 1986, pp. 191-208.
35. Carter N., The Application of a Flexible Tooling System in a Flexible Manufacturing System, Robotica, Vol. 3 (1985), Part 4, pp. 221-228.
36. Cassandras C.G., Autonomous Material Handling in Computer Integrated Manufacturing, Modelling and Design of Flexible

- Manufacturing Systems, Kusiak A. (ed.), Elsevier Sc. Publ. 1986, pp. 81-98.
37. Chakravarty A., Shtub A., Selecting parts and loading Flexible Manufacturing Systems, Proc. 1st ORSA/TIMS Conf. on FMS, Ann Arbor 1984, preprint.
 38. Chakravarty A.K., Shtub A., Integration of Assembly Robots in Flexible Assembly System, Flexible Manufacturing Systems: Methods and Studies, Kusiak A. (ed.), Elsevier Sc. Publ. 1986, pp. 71-88.
 39. Cheng J., Ushijima K., Analyzing Deadlocks and Livelocks in Concurrent Ada Programs Using Extended Petri Nets, Memoirs of the Faculty of Engineering Kyushu University, Vol. 48 (1988), No. 2, pp. 153-165.
 40. Chrétienne Ph., Analyse des Regimes Transitoire et Asymptotique d'un Graphe d'Evenements Temporels, Recherche, Vol. 4 (1985), No. 1, pp. 127-142.
 41. Chryssolouris G., Wright K., Pierce J., Cobb W., Manufacturing Systems Operation: Dispatching Rules Versus Intelligent Control Robotics and CIM, Vol. 4 (1988), No. 3/4, pp. 531-544.
 42. Coffman E.G., Jr., Elphick M.J., Shoshani A., System deadlocks, Computing Surveys, Vol. 3 (1971), pp. 67-78.
 43. Colom J.M., Martinez J., Silva M., Packages for Validating Discrete Production Systems Modelled with Petri Nets, Proc. IMACS-IFAC Symposium on Modelling and Simulation for Control of Lumped and Distributed Parameter Systems, Vilneuve D'ASCQ France, June 3-6, 1986, pp. 457-462.
 44. Commoner F., Holt A.W., Even S., Pnueli A., Marked Directed Graphs. Journ. Computer and System Science 5, (Oct. 1971), pp. 511-523.

45. Control and Programming in Advanced Manufacturing, Rathmill K. (ed.), IFS (Publ.) Ltd, UK, Springer Verlag 1988.
46. Corbeel D., Vercauter C., Gentina J.C., Adaptive Petri Nets for Real time Applications. Digital Technics in Simulation, Proc. of the IMACS European, 1984, pp. 515-520.
47. Cser, I., Toth T., Computer-Aided Technological Process Planning: Methods, Systems, and Applications Experiences, Modelling and Design of Flexible Manufacturing Systems, A. Kusiak (ed.), Elsevier Sc. Publ. 1986, pp. 249-285.
48. Czajkiewicz Z.J., Justification of the Robots Application, Flexible Manufacturing Systems: Methods and Studies, Kusiak A. (ed.), Elsviere Sc. Publ. 1986, pp. 367-376.
49. Dallery Y., Frein Y., An Efficient Method to Determine the Optimal Configuration of a Flexible Manufacturing System, Proc. 2nd ORSA/TIMS Conf. on Flexible Manufacturing Systems, Ann Arbor, USA, 1986, pp. 269-282.
50. De Luca A., Software Tools for FMS Preliminary Design: an Industrial Case Study, Robotica Vol. 3 (1985), Part 4, pp. 273-278.
51. Dopp K., Ein Satz über Free-Choice-Petrinetze, Elektron. Informationsverarb. u. Kybern. Vol. 19 (1983), pp. 107-113.
52. Dubois D., Stecke K.E., Using Petri Nets to Represent Production Processes, Proc. IEEE Conference on Decision and Control, December 1983, pp. 1062-1067.
53. Dunkler O., Mitchell C.M., Govindaraj T., Ammons J.C., The Effectiveness of Supervisory Control Strategies in Scheduling Flexible Manufacturing Systems, IEEE Trans. on Systems Man and Cybernetics, Vol. 18 (1988), No. 2, pp. 223-237.

54. Dupont-Galteruand C., A Real Time Control System for an Unaligned Flexible Manufacturing System, Proc. of the 4-th IFAC/IFIP Symposium on Information Control Problems in Manufacturing Technology, pp. 169-172.
55. Dutta A., Decision Support Requirements for Production Planning in FMS Environments, Modelling and Design of Flexible Manufacturing Systems, Kusiak A. (ed.), Elsevier Sc. Publ. 1986, pp. 331-346.
56. Erschler J., Roubellat F., Thuriot C., Steady State Scheduling of a Flexible Manufacturing System with Periodic Releasing and Flow Time Constraint, Annals of Operations Res., Vol. 3 (1985), pp. 333-353.
57. Erschler J., Lasserre J.B., Roubellat F., A New Tool for Measuring Decision Flexibility in Production Planning and Scheduling, LAAS Rapport No 2800, Toulouse, France. 1983,
58. Erschler J., Roubellat F., An Approach to Solve Workshop Real Time Scheduling Problems, Advanced Information Technologies for Industrial Material Flow Systems, Nof S.Y, Moodie C.L. (eds.), Springer-Verlag 1989.
59. Favrel J., Lee K.H., Modelling, Analyzing, Scheduling and Control of Flexible Manufacturing Systems by Petri Nets, Modelling Production Management Systems, P. Falster and R.B. Mazumder (eds.), Elsevier Sc. Publ. 1985, pp. 223-243.
60. Gaimon Ch., An impulsive control approach to deriving the optimal dynamic mix of manual and automatic output, European J. Op. Res., Vol. 24 (1986), pp. 360-368.
61. Genrich H.J., Lautenbach K., The Analysis of Distributed Systems by Means of Predicate/Transition-Nets, Proc. Conf. Semantics of Concurrent Computation, 1979, Kahn G. (ed.) LINC'S Vol. 70, (1979), pp. 123-146.

62. Genrich H.J., Lautenbach K., System Modelling with High-Level Petri Nets, Theoretical Computer Science 13(1981), pp. 109-136.
63. Genrich H.J., Predicate/Transition Nets, Petri Nets: Central Models and their Properties, Advances in Petri Nets 1986, Part I, LINCOS Vol. 245, 1987, pp. 207-247.
64. Germain R., Descotes-Genon B., A Tool for Production Management: Computer Aided Simulations Modern Production Management Systems, Kusiak A. (ed.) Elsevier Sc. Publ. 1987, pp. 161-187.
65. Gershwin B.S., Akella R., Choong F., Short-term Production Scheduling of an automated Manufacturing Facility, IBM J. Res. Develop., Vol. 29 (1985), No. 4, pp. 392-400.
66. Gibkije proizvodstwiennoje Sistemy Japonii, tłum. z j. japońskiego na rosyjski Siemionowa A.L., Moskwa, Mašinostrojenie, 1987.
67. Gibkije Sboročnyje Sistemy, Heginbotham W.B. (ed.), tłum. z ang. na rosyjski Mironowa D.F., Moskwa, Mašinostrojenije 1988.
68. Gold E.M., Deadlock Prediction: Easy and Difficult Cases, SIAM in Computer, Vol. 7 (1978), No. 3, pp. 320-336.
69. Greene T.J., Sadowski R.P., A Mixed Integer Program for Loading and Scheduling Multiple Flexible Manufacturing Cells, European J. Op. Res., Vol. 24 (1986), pp. 379-386.
70. Griesse W., Liveness in NSC - Petri nets, Pape V. (ed.), Hanser 1979, pp. 255-264.
71. Grieve R.J., Smith G.V., Machine and Component Specification for Flexible Manufacturing Systems for Metal Cutting Processes, Flexible Manufacturing Systems: Methods and Studies, Kusiak A. (ed.), Elsevier Sc. Publ. 1986, pp. 59-70.

72. Grum J., Logar B., Hlebanja G., Peklenik J., Design of the Database for CAD Bases on Group Technology, Robotics and CIM, Vol. 4 (1988), No. 1/2, pp. 49-62.
73. Gunn E.A., Kusiak A., The Procurement Problem: An Integer Programming Problem Well Suited To a Solution Using Duality, Naval Res. Log. Quart., Vol. 33 (1986), pp. 613-620.
74. Habermann A.N., Prevention of System Deadlocks, Comm. ACM, Vol. 12 (1969), No. 7, pp. 74-84.
75. Hack M., Decision problems for Petri nets and vector addition systems, MAC Techn. Mem. 59, MIT 1975.
76. Hack M., Petri net languages, MIT Computation Structure Group, MEMO 124, 1975.
77. Haines C.L., An Algorithm for Carrier Routing in a Flexible Material-Handling System, IBM J. Res. Develop., Vol. 29, No. 4, July 1985, pp. 256-362.
78. Hall D.N., Stecke K.E., Design Problems of Flexible Assembly Systems, Proc. 2-nd ORSA/TIMS Conf. on Flexible Manufacturing Systems, Ann Arbor, USA, 1986, pp. 145-156.
79. Hammer H., Flexible Manufacturing Cells and Systems with Computer Intelligence, Robotics and CIM, Vol. 3 (1987), No. 1, pp. 39-54.
80. Hansen P.B., Podstawy Systemów Operacyjnych, WNT Warszawa, 1979.
81. Hara S., Azuma K., Cell Production System for Assembly, Robotics and CIM, Vol. 4 (1988), pp. 379-385.
82. Havender J.V., Avoiding deadlock in multitasking systems, IBM Systems J., Vol. 7 (1968), pp. 74-84.
83. Hildebrandt R.R., Scheduling Flexible Machining Systems when Machines are Prone, to Failure, Ph.D. Thesis, MIT, Cambridge, 1980.

84. Horvath M., Manufacturing Engineering: The Birth and Growth of a New Science, Robotics and CIM, Vol. 4 (1988), pp. 285-292.
85. Hura G.S., Petri Net as a Modelling Tool, Microelectron, Reliab., Vol. 22 (1982), No. 3, pp. 433-439.
86. Hutchinson G., The Design of an Automated Material Handling System for a Job Shop, Computers in Industry, Vol. 4, 1983.
87. Hwang S., A Constraint-Directed Method to solve the Part Selection Problem in Flexible Manufacturing Systems Planning Stage, Proc. 2nd ORSA/TIMS Conf. on Flexible Manufacturing Systems, Ann Arbor, USA, 1986, pp. 297-310.
88. Iszkowski W., Maniecki M., Programowanie współbieżne, WNT, Warszawa 1982.
89. Ito Y., Shinno H., Saito H., A Proposal for CAD/CAM Interface with Expert Systems, Robotics and CIM, Vol. 4 (1988), pp. 491-497.
90. Jackson J.R., Job-shop Like Queueing Systems, Management Science, Vol. 10 (1963), pp. 131-142.
91. Jain S., Foley W.J., Basis for Development of a Generic FMS Simulator, Proc. 2nd Int. Conf. on Flexible Manufacturing Systems, Ann Arbor, USA, 1986, pp. 393-404.
92. Janson P.A., Operating Systems: Structures and Mechanisms, Academic Press, Inc., New York, 1985.
93. Jensen K., Coloured Petri Nets and the Invariant - Method, Theoretical Comp. Sci., Vol. 14 (1981) North Holland Publ., pp. 317-336.
94. Jensen K., High-Level Petri Nets, Informatik-Fachberichte Vol. 66, Applications and Theory of Petri Nets, Pagnoni A., Rozenberg G. (eds.), Springer-Verlag, 1983, pp. 166-180.

95. Jensen K., Coloured Petri Nets, Petri Nets: Central Models and their Properties, Advance in Petri Nets, 1986, Part I, ONCS Vol. 254, 1987, pp. 249-299.
96. Kalkunte M.V., Sarin S.C., Wilhelm W.E., Flexible Manufacturing Systems: A Review of Modeling Approaches for Design, Justification and Operation, Flexible Manufacturing Systems: Methods and Studies, Kusiak A. (ed.), Elsevier Sc. Publ. 1986, pp. 3-28.
97. Keramidis S., Ein Kalkül zur Konstruktion deadlockfreier Systeme, Pape V. (ed.), Hanser 1979, pp. 265-268.
98. Kim S.R., Multiobjective Decision Making Model for a Flexible Manufacturing System, Flexible Manufacturing Systems, Kusiak A. (ed.), Elseviere Sc. Publ. 1986, pp. 165-172.
99. Kinemia G., Hierarchical Control of Production in Flexible Manufacturing Systems, Ph.D. Dissertation, Report. No. LIDS-TH-1215, Lab. for Infor: and Deci. Sys., Mass. Inst. of Tech., Cambridge, MA, 1982.
100. Kimemia J., Gershwin S.B., An Algorithm for the Computer Control of a Flexible Manufacturing System, IIE Transactions, Vol. 15 (1983), No. 4, pp. 353-362.
101. Kimemia J., Gershwin S.B., Flow Optimization in Flexible Manufacturing Systems, Int. J. Prod. Res., Vol. 23 (1985), No. 1, pp. 81-96.
102. Kochan D. (ed.), CAM Developments in Computer-Integrated Manufacturing, Springer-Verlage, Berlin/Heidelberg/New York/Tokyo, IFIP State of the Art. Reports, 1986.
103. Kodate H., Fujii K., Yamanoi K., Representation of FMS with Petri Net Graph and Its Application to Simulation of System Operation, Robotics and CIM, Vol. 3 (1987), No. 3, pp. 275-283.

104. Krause F.L., Armbrust P., Bienert M., Methodbases and Product Models as Bases for Integrated Design and Manufacturing, Robotics and CIM, Vol. 4 (1988), pp. 33-40.
105. Krogh B.H., Beck C.L., Synthesis of Place/Transition Nets for Simulation and Control of Manufacturing Systems, 4-th IFAC/IFORS Symposium on Large Scale Systems: Theory and Applications, August 26-29, 1986, Zurich, Switzerland, pp. 661-666.
106. Krogh B.H. Ekberg G., Automatic Programming of Controllers for Discrete Manufacturing Processes, Proc. IFAC 10-th World Congress, Munich, 1987, pp. 283-288.
107. Krogh B.H., Sreenivas R.S., Essentially Decision Free Petri Nets for Real-Time Resource Allocation, Proc. IEEE Conf. on Robotics and Automation, Raleigh, USA, 1987, pp. 587-594.
108. Krogh B.H., Banaszak Z., Deadlock Avoidance in Pipeline Concurrent Processes, 22-nd Annual Conf. on Information Sciences and Systems, Princeton University, USA, 1988.
109. Kumar R.K., Kusiak A., Vanelli A., Grouping of Parts and Components in Flexible Manufacturing Systems, Euro. J. Op. Res., Vol. 24 (1986), No. 3, pp. 387-397.
110. Kusiak A., Vanelli A., Kumar R.K., Grouping Problem in Scheduling Flexible Manufacturing Systems, Robotica, Vol. 3 (1985), Part 4, pp. 245-252.
111. Kusiak A., The Part Families Problem in Flexible Manufacturing Systems, Annals of Operations Research 3 (1985), pp. 279-300.
112. Kusiak A., Planning of Flexible Manufacturing System, Robotica, Vol. 3 (1985), Part 4, pp. 229-232.
113. Kusiak A., Flexible Manufacturing Systems: a Structural Approach, Int. J. Prod. Res, Vol. 23 (1985), pp. 1057-1073.

114. Kusiak A., Part and Tools Handling Systems, Modelling and Design of Flexible Manufacturing Systems, Kusiak A. (ed.), Elsevier Sc. Publ. 1986, pp. 99-109.
115. Kusiak A., Scheduling Flexible Machining and Assembly Systems, Proc. 2nd ORSA/TIMS Conf. On Flexible Manufacturing Systems, Ann Arbor, USA, 1986, pp. 521-532.
116. Kusiak A., Application of Operational Research Models and Techniques in Flexible Manufacturing Systems, Euro. J. Op. Res., Vol. 24 (1986), No. 3, pp. 336-345.
117. Kusiak A., Design and Management of Computer Integrated Manufacturing Systems, Fall Industrial Engineering Conference, Boston, Mass., December 7-10, 1986, pp. 353-361.
118. Kusiak A., Heragu S.S., The Facility Layout Problem, Euro. J. Oper. Res. Vol. 29 (1987), pp. 229-251.
119. Kusiak A., Finke G., Hierarchical Approach to the Process Planning Problem, Discrete Applied Mathematics, Vol. 18 (1987), pp. 175-184.
120. Lenstra J.K., Sequencing by Enumerative Methods, Mathematisch Centrum, Amsterdam 1977.
121. Lin L.S., Chung S., A Systematic FMS Model for Real-Time On Line Control and Question - Answerer Simulation Using Artificial Intelligence, Proc. 2nd ORSA/TIMS Conf. on Flexible Manufacturing Systems, Ann Arbor, USA, 1986, pp. 567-580.
122. Maley J.G., Managing the Flow of Intelligent Parts, Robotics and CIM, Vol. 4 (1988), No. 3/4, pp. 525-530.
123. Martinez J., Silva M., A Package for Computer Design of Concurrent Logic Control Systems, Proc. Third IFAC/IFIP Symposium, Madrid, Spain, 5-8 October 1982, pp. 243-248.
124. Martinez J., Alla H., Silva M., Petri Nets for the Specification of FMSs, Modelling and Design of Flexible

- Manufacturing Systems, Kusiak A. (ed.), Elsevier Sc. Publ. 1986, pp. 398-406.
125. Martinez J., Silva M., A Language for the Description of Concurrent Systems Modelled by Colored Petri Nets: Application to the Control of Flexible Manufacturing Systems, Languages for Automation, Shi-Kuo Chang (ed.), Plenum Press, 1986, pp. 369-388.
 126. Merchant M.E., The Precepts and Sciences of Manufacturing, Robotics and CIM, Vol. 4 (1988), pp. 1-6.
 127. Micheletti G.F., Cantino V., Fenoglio F., An Application of Simulation Theory in FMS, Robotica Vol. 3 (1985), Part 4, pp. 253-258.
 128. Micheletti G.F., Application of New Technologies for Fully Integrated Robotized Automobile Engine Production, Robotics and CIM, Vol. 4 (1988), No. 1/2, pp. 141-148.
 129. Murata T., Petri Nets: Properties, Analysis and Applications, Proc. of IEEE 77 (1989), pp. 541-580.
 130. Nills R.I., A Review of Simulation Languages for Manufacturing System Applications, Control and Programming in Advanced Manufacturing, Rathmil K. (ed.), IFS (Publications) Ltd, Springer Verlag 1988, pp. 217-240.
 131. Morton T.E., Smunt T.L., A Planning and Scheduling System for Flexible Manufacturing, Flexible Manufacturing Systems: Methods and Studies, Kusiak A. (ed.), Elsviere Sc. Publ. 1986, pp. 151-164.
 132. Müller K., Constructable Petri Nets, Elektron., Inf verarb. Kybern. EIK, Vol. 21 (1985), pp. 171-199.
 133. Narahari Y., Viswanadham N., A Petri Net Approach to the Modelling and Analysis of Flexible Manufacturing Systems, Annals of Operations Res., Vol. 3 (1985), pp. 449-472.

134. Neelamkavil J., Thomson V., Rao D., Use of Dispatching Rules in Manufacturing Applications, 5-th Canadian CAD/CAM and Robotics Conf., Toronto, Canada, June 1986.
135. Nelson C.A., A Scoring Model for Flexible Manufacturing Systems Project Selection, Euro. J. Oper. Res., Vol. 24 (1986), No. 3, pp. 346-359.
136. Newman W.E., Model to Evaluate the Benefits of FMS Pallet Flexibility, Proc. 2nd ORSA/TIMS Conf. on Flexible Manufacturing Systems, Ann Arbor, USA, 1986, pp. 209-220.
137. Novels M.D., Mackwell G.B., Graphical Simulation of Material Handling Systems, Proc. 4-th Euro. Conf. Automatek Manufacturing, Birmingham, V.K., 1987, pp. 507-527.
138. Pagnoni A., Stochastic Nets and Performance Evaluation, Petri Nets: Central Models and Their Properties, Advances in Petri Nets, Brauer W. (ed.), Bad Honnef 1986, pp. 460-478.
139. Panwalkar S.S., Iskander W., A Survey of Scheduling Rules, Operations Res., Vol. 25 (1977), No. 1, pp. 45-61.
140. Pavlov A.A., Banaszak Z., Grisha S.N., Misyura E.B., Computer-aided control of processes in group technology systems, Wyd. Politechnika Wrocławska, Wrocław 1990.
141. Peterson J.L., Petri Net Theory and the Modeling of Systems, Prentice-Hall. Inc., Englewood Clifss, NJ, 1981.
142. Produktionsforum '88, Die CIM-fähige Fabrik, Bullinger H.J. (ed.), Springer-Verlag, 1988.
143. Rahman M., Narayanan V., An Expert System for Process Planning, Robotics and CIM, Vol. 3 (1987), No. 3, pp. 365-372.
144. Rajagopalan S., Formulation and Heuristic Solutions for Parts. Grouping and Tool Loading in Flexible Manufacturing Systems, Proc. 2nd ORSA/TIMS Conf. on Flexible Manufacuturing Systems, Ann Arbor, USA, 1986, pp. 311-320.

145. Ramamoorthy C.V., Ho C.S., Performance Evaluation of Asynchronous Concurrency Systems Using Petri nets, IEEE Trans. Software Engi., SE-6 (1980), pp. 440-449.
146. Raman R., Talbot B., Rachamadugn R., Simultaneous Scheduling of Machines and Material Handling Devices in Automated Manufacturing, Proc. 2nd ORDA/TIMS Conf. on Flexible Manufacturing Systems, Ann Arbor, USA, 1986, pp. 455-465.
147. Ranky P., The Design and Operation of Flexible Manufacturing Systems, IFS Publ. Ltd., North-Holland Publ. Co., 1983.
148. Ranky P.G. FMS in CIM, Robotica, Vol. 3 (1985), Part 4, pp. 205-214.
149. Ranky P.G., Computer Integrated Manufacturing, Prentice/Hall International, U.K., Ltd, 1986.
150. Ranky P.G., A Generic Tool Management System Architecture for Flexible Manufacturing Systems (FMS), Robotica, Vol. 6 (1988), Part 3, pp. 221-235.
151. Ranky P.G., Flexible Manufacturing Systems in Computer Integrated Manufacturing, Robotica (1985), Vol. 3, pp. 205-214.
152. Rayson P.T., A Review of Expert System Principles and Their Role in Manufacturing Systems, Robotica Vol. 3 (1985), Part 4, pp. 279-287.
153. Recent Trends in Flexible Manufacturing, Economic Comission for Europe, United Nations Publications, New York 1986.
154. Reiser M., Lavenberg S.S., Mean value analysis of closed multichain queueing networks, J. ACM, Vol. 27 (1980), pp. 313-322.
155. Reisig W., Petri Nets, Berlin, Springer-Verlag, 1982.

156. Reisig W., Place/Transition Systems, Petri Nets: Center Models and their Properties, Advances in Petri Nets, 1986, Part I, LNCS Vol. 254, (1987), pp. 117-141.
157. Roszkowska E., Banaszak Z., Computer-aided control and management system for FMSs: a Petri net approach, Systems Prospects, Proc. of the UKSS Conference on Systems Prospects, Hull. G.B., July 1988, Plenum Press, New York, 1989, pp. 297-301.
158. Roszkowska E., Deadlock Avoidance in Concurrent Compound Pipeline Processes, Podstawy Sterowania, w druku.
159. Roszkowska E., Nowak P., System Symulacji Złożonych Procesów Produkcyjnych, Raport 47/90 ICT Politechnika Wrocławska.
160. Sackett P.J., Flexible Handling and Fixturing in Manufacture, Proc. 4-th Euro. Conf. Automated Manufacturing, Birmingham, U.K., 1987, pp. 131-140.
161. Sahraoui A., Courvoisier M., Valette R., Multilevel Control of Flexible Manufacturing Systems: An Integrated Approach, Report, LAAS No. 86101, 1986.
162. Sahraoui A., Atabakhche H., Courvoisier M., Valette R., Joining Petri Nets and Knowledge Based Systems for Monitoring Purposes, Proc. IEEE Int. Conf. on Robotic and Automation Raleigh, USA, 1987, pp. 1160-1165.
163. Santarek K., Strzelczak St., Elastyczne systemy produkcyjne, WNT Warszawa 1989.
164. Sarin S.C., Chen C.S., A Mathematical Model for Manufacturing System Selection, Flexible Manufacturing Systems: Methods and Studies, Kusiak A. (ed.), Elsevier Sc. Publ. 1986, pp. 99-112.

165. Sauve B., Collinot A., An Expert System for Scheduling in a Flexible Manufacturing System, Robotics and CIM, Vol. 3 (1987), No. 2, pp. 229-233.
166. Sawik T., Modelling and Scheduling a Flexible Manufacturing System, Joint Int. Conf. EURO IX - TIMS XXVIII, Paris, France, 1988, p. 206 (abstract).
167. Seliger G., Viehweger B., Wieneke B., Descriptive Methods for Computer-Integrated Manufacturing and Assembly, Robotics, and CIM, Vol. 3 (1987), pp. 15-21.
168. Seliger G., Viehweger B., Wieneke B., Decision Support in Design and Optimization of Flexible Automated Manufacturing and Assembly, Robotics and CIM, Vol. 3 (1987), pp. 221-227.
169. Seliger G., Viehweger B., Kommana S.R., Integrated Planning of Manufacturing Systems, Robotics and CIM, Vol 4 (1988), No. 3/4, pp. 593-600.
170. Semakula M.S., The Role of Process Planning in the Integration of CAD/CAM Systems, Proc. 4-th Euro. Conf. Automated Manufacturing, Birmingham, V.K., 1987, pp. 269-276.
171. Serzeniewski M., Design and Application of Programmable Flexible Transportation Systems, Proc. 4-th Euro. Conf. Automated Manufacturing, Birmingham, V.K., 1987, pp. 151-162.
172. Shanker K., Tzen Y.J., A Loading and Dispatching Problem in a Random Flexible Manufacturing System, Int. J. Prod. Res., vol. 23 (1985), pp. 579-595.
173. Shanthikumar J.G., Sargent G.R., A Hybrid Simulation/Analytic Model of a Computerized Manufacturing System, Operational Research '81, J.P. Brans (ed.), IFORS, Hamburg, 1981, pp. 901-915.

174. Shaw A.C., Projektowanie logiczne systemów operacyjnych, WNT Warszawa, 1980.
175. Shaw M.J.P., Whinston A.B., Applications of Artificial Intelligence to Planning and Scheduling in Flexible Manufacturing, Flexible Manufacturing Systems: Methods and Studies, A. Kusiak (ed.), Elsevier Sc. Publ. 1986, pp. 223-242.
176. Shen S., Chang Y., An AI Approach to Schedule Generation in a Flexible Manufacturing System, Proc. 2nd ORSA/TIMS Conf. on Flexible Manufacturing Systems, Ann Arbor, USA, 1986, pp. 581-591.
177. Silva M., Martinez J., A Software Environment for Designing with Colored Petri Nets and Their Implementation. Application to Flexible Manufacturing Systems, Proc. Int. Conf. on Application of Petri Nets to Operation Research, University of Milano, Milan, Italy, May 1986, pp. 95-127.
178. Sluga A., Butala P., Lavrac N., Gams M., An Attempt to Implement Expert System Techniques in CAPP, Robotics and CIM, Vol. 4 (1988), No. 1/2, pp. 77-82.
179. Smith L.M., Ramesh R., Dudek R.A., Blair E.L., Characteristics of V.S. Flexible Manufacturing Systems - A Survey, Proc. 2nd Int. Conf. on Flexible Manufacturing Systems, Ann Arbor, USA, 1986, pp. 477-486.
180. Souza K.D., Modelling Concurrent Manufacturing Systems Using Petri Nets, 5-th Int. Conf. on CAD/CAM, Norfolk USA, 2-5 December 1990, preprint.
181. Spooner P.D., A Simulation Based Control System for Batch Production, Proc. 4-th Euro. Conf. Automated Manufacturing, Birmingham, V.K., 1987, pp. 335-344.

182. Spur G., Advanced Manufacturing Systems, Robotics and CIM, Vol. 4 (1988), No. 1/2, pp. 7-12.
183. Sriskandarajah C., Ladet P., Germain R., Scheduling Methods for a Manufacturing System, Flexible Manufacturing Systems: Methods and Studies, Kusiak A. (ed.), Elsevier Sc. Publ. 1986, pp. 173-190.
184. Sriskandarajah C., Ladet P., Sethi S.P., A Scheduling Method for a Class of Flexible Manufacturing Systems, Modern Prod. Manage. Systems, Kusiak A. (ed.), Elsevier Sc. Publ., 1987, pp. 83-91.
185. Starke P.H., Sieci Petri, PWN Warszawa, 1987.
186. Stecke K.E., Formulation and Solution of Nonlinear Integer Production Planning Problem for Flexible Manufacturing Systems, Management Science, Vol. 29 (1983), pp. 273-288.
187. Stecke K.E., Solberg J.J., Loading and Control Policies for a Flexible Manufacturing System, Int. J. Prod. Res., 1981, Vol. 19, No. 5, pp. 481-490.
188. Stecke K.E., Design, Planning, Scheduling and Control Problems of Flexible Manufacturing Systems, Annals of Operation Research, Vol. 3 (1985), pp. 3-12.
189. Stecke K.E., Kim I., A Flexible Approach to Implementing the Short-Term FMS Planning Function, Proc. 2nd ORSA/TIMS Conf. on Flexible Manufacturing Systems, Ann Arbor, USA, 1986, pp. 283-296.
190. Stecke K.E., A Hierarchical Approach to Solving Machine Grouping and Loading Problems of Flexible Manufacturing Systems, Euro. J. Oper. Res., Vol. 24 (1986), No. 3, pp. 369-378.

191. Steffen M.S., Green T.J., An Application of Hierarchical Planning and Constraint-Directed Search to Scheduling Parallel Processors, Proc. Int. Conf. on Robotic and Automation, New York, 1986, pp. 910-917.
192. Stephan C. Graves: A Review of Production Scheduling, Operations Res., Vol. 29, No. 4, July-August 1981, pp. 899-908.
193. Subramanyam S., Askin R.G., An Expert Systems Approach to Scheduling in Flexible Manufacturing Systems, Flexible Manufacturing Systems: Methods and Studies, Kusiak A. (ed.), Elseviere Sc. Publ. 1986, pp. 243-256.
194. Suh N.P., A Perspective on Manufacturing, Robotics and CIM, Vol. 4 (1988), No. 3/4, pp. 297-307.
195. Takahashi K., Hasegawa K., A Deadlock Avoidance Method for Job-shop Manufacturing System, Int. Conf. on System Science - Abstract of Papers, Wrocław, Poland, 1989, p. 191.
196. Taylor I., Overall Control Within a Flexible Manufacturing System and the Development of a Cell Controller, Robotica, Vol. 3 (1985), Part 4, pp. 215-220.
197. Thesen A., Lei L., An Expert System for Scheduling Robots in a Flexible Electroplating System with Dynamically Changing Workloads, Proc. 2nd ORSA/TIMS Conf. on Flexible Manufacturing Systems, Ann Arbor, USA, 1986, pp. 555-566.
198. Thiagarajan P.S., Voss K., A fresh look at free choice nets, Inf. and Contr., Vol. 61 (1984), pp. 85-113.
199. Thiagarajan P.S., Elementary Net Systems, Petri Nets: Central Models and their Properties, Advances in Petri Nets LNCS, Vol. 254 (1987), pp.26-59.
200. Thuriot E., Valette R., Courvoisier M., Implementation of a Centralized Synchronization Concept for Production

Systems, Real-Time Systems Symposium, Arlington (USA), 6-8 December, 1983, preprint Rapport LAAS No. 2945, Toulouse, France.

201. Tierney K., Bowden R., Browne J., A Prototype Expert System for Technological Planning in Robot Based Flexible Assembly System, Proc. 2nd ORSA/TIMS Conf. on Flexible Manufacturing Systems, Ann Arbor, USA, 1986, pp. 169-180.
202. Towards the factory of the future, Bullinger H.J., Warnecke H.J. (eds), Springer Verlag 1985.
203. Tsatsoulis C., Kashyap R.L., A Case-Based System for Process Planning, Robotics and CIM, Vol.4 (1988) No.3/4, pp.557-570.
204. Valette R., Courvoisier M., Bigou J.M., Albukerque J., A Petri Net Based Programmable Logic Controller, First. International Conference on Computer Applications in Production and Engineering, Amsterdam, 25-28 April, 1983, preprint Rapport LAAS No. 2801, Toulouse, France.
205. Valette R., Courvoisier M., Demmon H., Bigou J.M., Desclaux C., Putting Petri Nets to Work for Controlling Flexible Manufacturing Systems, Rapport LAAS No. 85151, 1985.
206. Valette R., Atabakhche H., Courvoisier M., Lemaire T., Petri Nets and Production Rules for Decision Levels in FMS Control, LAAS Rapport No. 88038, 1988, Toulouse, France.
207. Valette R., Coordination Problems in FMS Control Systems, LAAS Rapport No. 88049, 1988, Toulouse, France.
208. Valk R., On the Computational Power of Extended Petri-Nets, Lecture Notes in Computer Sc., No.64, Springer Berlin, 1978, pp. 526-535.
209. Van Looveren A.J., Gelders L.F., Van Wassenhove L.N., A Review of FMS Planning Models, Modelling and Design of Flexible Manufacturing Systems, Kusiak A. (ed.), Elsevier Sc. Publ. 1986, pp. 3-32.

210. Villa A., Rossetto S., Towards a Hierarchical Structure for Production Planning and Control in Flexible Manufacturing Systems, Modelling and Design of Flexible Manufacturing Systems, Kusiak A. (ed.), Elsevier Sc. Publ. 1986, pp. 209-228.
211. Vinod B., Sabbagh M., Optimal Performance Analysis of Manufacturing Systems Subject to Tool Availability, Euro. J. Res., Vol. 24 (1986), No. 3, pp. 398-409.
212. Wang H., An Experimental Analysis of the Flexible Manufacturing System, Flexible Manufacturing Systems: Methods and Studies, Kusiak A. (ed.), Elseviere Sc. Publ. 1986, pp. 319-340.
213. Weil R., Integrating Dimensioning and Tolerancing in Computer-Aided Process Planning, Robotics and CIM, Vol. 4 (1988), No. 1/2, pp. 41-48.
214. Weston R.H., Sumpter C.M., Gascoigne J.D., Distributed Manufacturing Systems, Robotica, Vol. 4 (1986), pp. 15-26.
215. White C., Production Scheduling and Materials Requirement Planning for Flexible Manufacturing, Modelling and Design of Flexible Manufacturing Systems, Kusiak A. (ed.), Elsevier Sc. Publ. 1986, pp. 229-248.
216. Wilhelm W.E., Hyun-Myung Shin, Effectiveness of Alternate Operations in a Flexible Manufacturing System, Int. J. Res., Vol. 23 (1985), No. 1, pp. 65-79.
217. Yao D.D., Buzacott J.A., Modelling the Performance of Flexible Manufacturing Systems, Int. J. Prod. Res., 1985, Vol. 23, No. 5, pp. 945-959.
218. Yao D.D., Buzacott J.A., Models of Flexible Manufacturing Systems with Limited Local Buffers, Int. J. Prod. Res., 1986, Vol. 24, No. 1, pp. 107-118.

- 219. Yao D.D., Buzacott J.A., Modelling a Class of State-Dependent Routing in Flexible Manufacturing Systems, Ann. Oper. Res., Vol. 3 (1985), pp. 153-167.
- 220. Yao D.D., An Optimal Storage Model for a Flexible Manufacturing System, Flexible Manufacturing Systems: Methods and Studies Kusiak A. (ed.), Elsevier Sc. Publ. 1986, pp. 113-126.
- 221. Yao D.D., Some Resource Allocation Problems in Multi-Cell Systems, Proc. 2nd ORSA/TIMS Conf. on Flexible Manufacturing Systems, Ann Arbor, USA, 1986, pp. 245-256.
- 222. Zhou M.Ch., Lesare F., Rudolph D., Control of a Flexible Manufacturing Systems Using Petri Nets, Proc. 11-th IFAC World Congress on Automatic Control in the Service of Mankind, Aug. 13-17 1990, Estonia USSR, pp. 43-48.
- 223. Zörntlein G., Flexible Fertigungs-Systeme, Carl Hanser Verlag, 1988.
- 224. Zuberek W.M., Timed Petri Nets Preliminary Performance Evaluation, proc. of the 7-th Annual Symposium on Comp. Architecture, La Baule France, 1980, pp. 88-96.



mgr inż. Elżbieta Roszkowska
Instytut Cybernetyki Technicznej
Politechniki Wrocławskiej
ul. Janiszewskiego 11/17
50-372 Wrocław

Niniejszy raport otrzymują:

- | | |
|---|----------|
| 1. OINT | - 1 egz. |
| 2. Biblioteka Główna PW. | - 1 egz. |
| 3. Archiwum ICT | - 1 egz. |
| 4. Promotor | - 1 egz. |
| 5. Recenzenci | - 3 egz. |
| 6. Z-ca Dyrektora Instytutu ds. Nauki i Kształcenia
Kadry Naukowej | - 1 egz. |
| 7. Kierownik Zakładu Podstaw Cybernetyki i Robotyki | - 1 egz. |
| 8. Autor | - 2 egz. |

Razem: - 11 egz.

Raport wpłynął do redakcji I-6
w lutym 1991 r.