

AKADEMIA EKONOMICZNA im. OSKARA LANGEGO
we WROCŁAWIU
INSTYTUT CYBERNETYKI EKONOMICZNEJ

Lech Warężak

OPTIMALIZACJA W SEKWENCYJNYCH PROCESACH PRODUKCYJNYCH

Promotor:
Prof. dr hab. Zdzisław Hellwig

WROCŁAW 1975

Spis treści

	str.
Wstęp	1
1. Matematyczny model zadań programowania sekwencyjnego	9
1.1. Uwagi ogólne	9
1.2. Zagadnienie przydziału. Selekcja danych	12
1.2.1. Zbiór Γ i jego własności	12
1.2.2. Twierdzenie o rozbięciu zbioru Γ	19
1.2.3. Algorytm rozbięcia	23
1.3. Podstawowe pojęcia niezbędne do sformułowania zadań programowania sekwencyjnego	26
1.3.1. Działanie operatora T_{p1} i złożenia T_{pD}	26
1.3.2. Złożenie $T_{M,D}$	31
1.3.3. Zbiór operatorów \mathcal{T}_{p1} i rodzina \mathcal{T}_{MK}	33
1.4. Zagadnienie sekwencyjne. Problem podstawowy . .	40
1.4.1. Dalsze elementy modelu. Przygotowanie danych	40
1.4.2. Sformułowanie zadania programowania sekwencyjnego i jego szczególne przypadki	49
1.4.3. O możliwości rozbięcia zadania wyjścio- wego na zadania mniejszych rozmiarów . .	52
1.4.4. Zachowanie się wartości funkcji kryte- rium gdy S i S'' nie są podzbiorami niezależnymi ze względu na warstwy . . .	62

	str.
2. Przykłady zadań programowania sekwencyjnego . . .	66
2.1. Uwagi wstępne	66
2.2. Ustalanie harmonogramów produkcji	67
2.3. Ustalanie harmonogramów zajęć	83
2.4. Planowanie świadczenia usług	87
3. Metody rozwiązywania zadań programowania sekwencyjnego	90
3.1. Wprowadzenie	90
3.2. Algorytm Johnsona	92
3.3. Modele całkowite-liczbowe	105
3.3.1. Model Bewmana	107
3.3.2. Model Jankowskiej-Zorychty I	116
3.3.3. Model Jankowskiej-Zorychty II	122
3.3.4. Model Manne'a	125
3.3.5. Model Wagnera	133
3.4. Metody podziału i ograniczeń	136
3.4.1. Idea metody podziału i ograniczeń	136
3.4.2. Metoda Lomnickiego	144
3.4.3. Metoda Ignalla i Schrage'a	145
3.4.4. Metoda Mc Mahona i Burtona	146
3.4.5. Metoda Browna i Lomnickiego	147
3.4.6. Metoda Czernewej	148
3.4.7. Modyfikacja metody podziału i ograniczeń	155
3.5. Metody eliminacyjne	157
3.5.1. Wprowadzenie	157
3.5.2. Metoda Shmitha-Dudka	159
3.5.3. Metoda Szwarea	160

III

	str.
3.5.4. Metoda Bagga i Chakravarti	161
3.5.5. Metoda Dudka i Teutona	161
3.5.6. Twierdzenie o metodach eliminacyjnych	162
3.6. Metody przybliżone	171
3.6.1. Uwagi na temat metod przybliżonych . .	171
3.6.2. Metoda Pietrowa i Sokoliejna	172
3.6.3. Metoda Pietrowa	177
3.6.4. Metoda Palmera	180
3.6.5. Metoda Campbella, Dudka i Smitha . .	181
3.6.6. Metoda Tiutiukina I.	184
3.6.7. Metoda Tiutiukina II	186
3.6.8. Metoda wzajemnego poprzedzania	188
3.6.9. Metoda uśredniania miar jednoczesności działania I	189
3.6.10. Metoda uśredniania miar jednoczesności działania II	193
3.6.11. Metoda dróg hamiltonowskich	197
3.6.12. Metoda Monte Carlo	200
Zakończenie	207
Dodatek ,	216
Bibliografia	236

WSTEP

Gospodarowanie jest tak stare, jak stary jest rodzaj ludzki, ale na różnych szczeblach rozwoju społeczeństwa gospodarowanie wymagało różnych zasobów wiedzy i umiejętności.

Na niskim szczeblu kultury materialnej cała wiedza o gospodarowaniu opierała się na tradycji wzbogacanej minimalnie bieżącym doświadczeniem. Prymitywny podział pracy, lokalny charakter produkcji i dystrybucji rzadko uzupełnianej przypadkową wymianą nadwyżek, prosta i przejrzysta organizacja pracy - wszystko to nie wymagało głębszych dociekań i rozważań mających na celu usprawnienie procesów produkcji, które i tak były proste i nieskomplikowane.

Doświadczenia starszych rodu i fatalistyczne poddawanie się żywiołom, to były dwa zasadnicze elementy wyznaczające gospodarczą działalność człowieka.

Jednak wraz z postępem technicznym, wraz z rosnącym zapotrzebowaniem na dobra produkcyjne powstaje konieczność poznawania mechanizmów i praw rządzących procesami gospodarczymi, jego obiektywną strukturą i dynamiką. Konieczność ta rodzi się między innymi i po to, by móc ustalać obiektywne możliwości działania a tym samym wyznaczać realistyczne a nie woluntarystyczne cele działalności. Równocześnie, z drugiej strony okazuje się, że na to by działać nie wystarczy drogą nawet najsubtelniejszych abstrakcji ustalić kierunki działania, trzeba też wiedzieć jak działać, gdyż

w praktyce życia gospodarczego do kierowania gospodarką planową - i to niezależnie od szczebla - nie wystarcza wiedza o kategoriach i prawach ekonomicznych, jak też nie wystarcza sama praktyka i doświadczenie. Wynika to z tego iż wobec bogatej struktury organizacyjnej, dużej liczby ograniczeń, często bliskich sprzeczności rosnącego zapotrzebowania na dobra produkcyjne decyzje podejmowane w oparciu o intuicję nie mogą być decyzjami optymalnymi.

Na każdym kroku człowiek w swojej działalności napotyka na konieczność wyboru optymalnego, w danych warunkach wariantu działania, częstokroć nie zdając sobie sprawy z istnienia formalnych metod optymalizacyjnych.

Można dyskutować, czy znajomość tych metod jest mu potrzebna, a po drugie, czy w każdej sytuacji aparat formalny da się zastosować. Znamiennym jest tu przykład cytowany przez prof. Z. Czerwińskiego: "kasjer w celu obliczenia należności za zakupiony towar wyznacza iloczyn skalarny dwóch wektorów: zakupu i cen, nic na ogół nie wiedząc o istnieniu takiego iloczynu".

Nie ulega jednak wątpliwości, że wszędzie gdzie decyzje dotyczą grup społecznych, działalności gospodarczej człowieka czy też systemu, tam muszą one być podejmowane w oparciu o obiektywne przesłanki. I coraz częściej człowiek w swojej działalności gospodarczej ucieka się do pomocy metod matematycznych. Metody matematyczne bowiem ułatwiają wybór właściwej drogi realizacji zamierzeń, a często mają przekonać o niemożliwości zrealizowania w danych warunkach tego, czy innego cząstkowego zadania. W najgorszym razie rozwiązanie

uzyskane drogą formalizacji zagadnienia służyć może do kontroli rozwiązania intuicyjnego lub stanowić konkurencyjny wariant do rozwiązań proponowanych przez praktyków.

Najszerze i najefektywniejsze zastosowanie metod matematycznych notuje się wszędzie tam, gdzie przedmiotem rozwiązań są problemy o charakterze ekonomiczno-organizacyjnym. Dotyczy to zwłaszcza zagadnień, gdzie idzie o wybór najlepszego z wariantów technologicznych, rodzajów kooperacji czy też rozwiązań transportowych.

Wobec ewidentnych korzyści płynących ze stosowania metod matematycznych, praktyka coraz śміielej formułuje zadania pod adresem nauki. Wiele rozwiązań znajduje zastosowanie, wiele jednak wobec dużego stopnia komplikacji obliczeń, a więc trudności czysto rachunkowych, pozostaje rozwiązaniami tylko teoretycznymi. Dopiero stosowanie elektronicznej techniki obliczeniowej umożliwia wprowadzanie nowych rozwiązań do praktyki.

Atakuje się coraz to nowe dziedziny, idea optymalizacji staje się obowiązującą doktryną w działalności gospodarczej człowieka. Dostrzega się możliwości optymalizacji nie tylko kosztów, czy też zysku, przedmiotem optymalizacji staje się czas. Już nikt nie wątpi w znaczenie minimalizacji czasu wykonania zadanych zadań. W literaturze¹ znaleźć można następujące argumenty: "Skrócenie cyklu produkcyjnego przyniosłoby przedsiębiorstwu następujące korzyści:

¹ patrz Organizacja i planowanie w przedsiębiorstwie przemysłowym, pod red. A. Grossmana.

- lepsze wykorzystanie wyposażenia technicznego,
- zwiększenie stopnia wykorzystania zdolności produkcyjnych,
- wzrost produkcji,
- przyspieszenie krążenia środków trwałych,
- obniżenie kosztów własnych".

Dodajmy od siebie, że skrócenie cyklu produkcyjnego poprzez zabiegi organizacyjne nie wymaga żadnych dodatkowych nakładów, a wdrożenie do praktyki rozwiązań nie pociąga za sobą konsekwencji w postaci okresów adaptacyjnych, które to okresy charakteryzują się doraźnym spadkiem efektów ekonomicznych przedsiębiorstwa.

W latach 50-tych sformułowane zostało zadanie z dziedziny organizacji produkcji¹.

Usiłuje się budować optymalne harmonogramy zajęć, wyznaczać plan wykonywania zadanych w danym okresie usług.

Zadania tego typu w pracy niniejszej nazwano zadaniami programowania sekwencyjnego, gdyż optymalizuje się tutaj pewien funkcjonal, którego wartość zależy w sposób istotny od kolejności wykonywanych czynności.

Po raz pierwszy zadanie tego typu zostało sformułowane przez Bellmana. Pierwsza próba jego rozwiązania pochodzi od Johnsona. W latach 60-tych notuje się bardzo ożywioną działalność mającą na celu wypracowanie metod jego rozwiązania. Próby te jednak kończą się raczej niepowodzeniem. Zdaniem autora, jedną z przyczyn tego niepowodzenia jest to, że proponowane rozwiązania mają czysto teoretyczny charakter.

¹ patrz S. Johnson ...Optimal Two-And Three-Stage Production Schedules with Setup Times Included.

Wynika to z tego, że zadania te są zadaniami krótkookresowymi. Dane w zadaniach zmieniają się w bardzo krótkich okresach czasu, co z kolei powoduje, że rozwiązanie optymalne dla zadania rozwiązywanego wczoraj już dla zadania, które trzeba rozwiązywać dzisiaj jest nieaktualne.

Jednak z chwilą coraz szerszej komputeryzacji, a więc łatwego dostępu do komputera, który gwarantuje szybkość wykonywanych obliczeń zasadnicza trudność, o której była mowa przestaje być ograniczeniem. Z dnia na dzień bowiem, mając wystarczające dane, można budować optymalne plany pod warunkiem, że dysponuje się odpowiednimi algorytmami.

Mając to na uwadze autor podjął próbę zbudowania jednolitej teorii dla szerokiej klasy zadań sekwencyjnych, a więc zadań w których należy wyznaczyć kolejność zadanych czynności. Dodajmy od razu, że zadania sekwencyjne stawiane w literaturze są szczególnymi przypadkami ogólnego zadania programowania sekwencyjnego zdefiniowanego w tej pracy.

Praca ta składa się z trzech zasadniczych rozdziałów oraz dodatku.

Rozdział pierwszy poświęcony jest formalnemu opisowi zadań programowania sekwencyjnego. Używając modnego dzisiaj terminu, w rozdziale tym zbudowano model matematyczny zagadnień sekwencyjnych.

W rozdziale tym podjęto również próbę sformułowania warunków jakie muszą być spełnione by bez szkody dla jakości rozwiązania, można było wyjściowe zadanie rozbijać na dwa, lub więcej zadania, których rozwiązanie dawałoby łącznie rozwiązanie zadania wyjściowego. Idzie bowiem o to, że w praktyce spotyka się zadania o dużych rozmiarach co poważ-

nie utrudnia ich rozwiązanie. Praktycznie wówczas nie ma możliwości wyznaczenia dla takiego zadania rozwiązania optymalnego, a rozwiązanie przybliżone może się znacznie różnić od optymalnego, podczas gdy dla zadań o małych rozmiarach rozwiązania takie można uzyskiwać.

Próba formalizacji i jej kierunek podjęte w tym rozdziale wynika z dwóch powodów.

Po pierwsze, łatwość programowania dla celów elektronicznej techniki obliczeniowej tak sformułowanych zadań.

Po drugie, całą bogatą strukturę zadania w zaproponowanym języku daje się opisać w miarę prosto i w sposób jednoznaczny. Dotyczy to zwłaszcza zadań, w których dane są do zrealizowania kilka technologii lub stanowiska są wieloczynnościowe. Wówczas do opisu tych zadań używać trzeba kilku macierzy¹ co z kolei rodzi określone trudności przy maszynowej realizacji algorytmu wyznaczającego rozwiązanie takiego zadania. Podejście proponowane przez autora w niniejszej pracy jest tego mankamentu pozbawione. Całe zadanie opisuje się jedną macierzą, rzecz jasna dzieje się to kosztem jej wymiaru. Dokładniej mówiąc liczba kolumn macierzy jest odpowiednio większa.

W rozdziale drugim podano kilka przykładów zadań programowania sekwencyjnego, które spotyka się w praktyce. Są to zadania z dziedziny organizacji produkcji, budowania harmonogramów oraz ustalania planów świadczenia usług takich jak remonty czy też naprawy. W rozdziale tym uzasadniono celowość wprowadzenia do modelu niektórych założeń oraz podano odpowiedniki z praktyki pojęć wprowadzonych w rozdziale pierwszym.

¹ patrz L. Wareżak "O pewnym uogólnieniu zagadnienia sekwencyjnego".

Rozdział trzeci to rozdział monograficzny poświęcony prezentacji znanych metod wyznaczania rozwiązania zadania sekwencyjnego. Metody te powstawały na bazie konkretnych zadań, które są szczególną klasą zadań programowania sekwencyjnego nazwaną w rozdziale pierwszym zadaniami typu E. Zadania te w przeważającej większości pochodzą z dziedziny organizacji produkcji.

Algorytmy te z konieczności nie mogły być przytaczane w ich oryginalnym brzmieniu lecz zostały "przetłumaczone" na język rozdziału pierwszego. Omawiane metody podzielono na cztery grupy:

- metody programowania całkowito-liczbowego,
- metody podziału i ograniczeń,
- metody eliminacyjne,
- metody przybliżone.

Wszystkie omawiane metody zostały ilustrowane prostymi przykładami. W rozdziale tym zawarto propozycje własne autora rozwiązywania tego typu zagadnień. Wymieńmy kilka:

- modyfikacja metody podziału i ograniczeń,
- metoda grafowa,
- procedura Monte Carlo.

W dodatku, na kilkunastu przykładach dokonuje się porównania metod oraz ilustruje się zbieżność procedury Monte Carlo. Dane do przykładów były generowane na maszynie cyfrowej z przedziału $[1,100]$ z rozkładem jednostajnym.

Wszystkie prezentowane tam wyniki wyznaczone były na maszynie cyfrowej ODRA 1204 w Ośrodku Obliczeniowym Instytutu Cybernetyki Ekonomicznej Akademii Ekonomicznej im. Oskara Langego we Wrocławiu.

Składam gorące podziękowanie prof.dr hab. Z.Hellwigowi
za zwrócenie mojej uwagi na podjętą w pracy tematykę, pomoc
i zachętę w trakcie jej pisania.

1. MATEMATYCZNY MODEL ZADAN PROGRAMOWANIA SEKWENCYJNEGO

1.1. Uwagi ogólne

W rozdziale tym zaprezentowany zostanie abstrakcyjny model zadań programowania sekwencyjnego. Rozdział ten składa się z trzech części, z których pierwsze dwie mają charakter pomocniczy, natomiast w części trzeciej formułuje się zasadnicze zagadnienie tej pracy.

Jest rzeczą oczywistą, że przystępując do rozwiązywania dowolnego zagadnienia praktycznego należy dokonać selekcji danych. W wielu bowiem przypadkach dysponuje się niekompletnym, bądź za obszernym zbiorem danych, co utrudnia lub wręcz uniemożliwia rozwiązanie zadania. W związku z tym autor uznał za celowe sprawom selekcji danych poświęcić nieco miejsca w niniejszej pracy. Zagadnienie to omówione zostało w pierwszej części niniejszego rozdziału.

Zagadnienie sekwencyjne rozpatrywane w tej pracy dotyczy szczególnej klasy najogólniej pojętych procesów produkcyjnych, tzn. takich, w których istnieje wzajemna odpowiedniość pomiędzy wyrobem a surowcem, z którego wyrób ten został otrzymany. Wobec tego znajduje tu zastosowanie tzw. "zadanie przydziału". Ponieważ w literaturze, przedmiotowi temu poświęcono wiele miejsca, znane i cytowane są algorytmy rozwiązania tej klasy zadań¹, praktycznie rozwiązane jest za-

¹ Galle "Teoria liniowych modeli ekonomicznych",
J.Kucharczyk, M.Sysło "Algorytmy optymalizacji w języku
Algol-60".

gadnienie wyznaczania przydziałów równoważnych¹, sprawy te potraktowano w pracy marginesowo. Więcej natomiast uwagi poświęcono problemowi redukcji rozmiarów zadania, co jest ważne ze względu na możliwość uzyskania rozwiązania.

W tym celu wprowadza się zbiór Γ , który jest podzbiorem produktu kartezjańskiego dwóch wyjściowych zbiorów S oraz W . Należy zaznaczyć, że zbiór Γ jest dany i poszukuje się takiego jego podzbioru, by zbiór następników był dokładnie zbiorem W , a równocześnie by zadany funkcjonal uzyskał wartość minimalną.

Okazuje się, że często zagadnienie wyjściowe rozbić można na dwa zadania mniejszych rozmiarów i to tak, by ich rozwiązanie łącznie dawało rozwiązanie zadania wyjściowego.

Po to by można było stwierdzić istnienie takiej możliwości wprowadza się pojęcie niezależności podzbiorów zbioru

Podaje się również twierdzenie, które ma charakter konstruktywny, tzn. wynika z niego sposób wyznaczania niezależnych podzbiorów. W części tej znajduje się odpowiedni algorytm wraz z dowodem jego poprawności.

W drugiej części przedstawiono pojęcia i terminy niezbędne do opisanie zasadniczego problemu pracy. W tym celu wprowadza się funkcję g_{pl} oraz operator T_{pl} . Funkcja g_{pl} nie ma odpowiednika w praktyce, służy jedynie do opisanie działania operatora T_{pl} , który daje się interpretować jako operacja na wyrobie p -tym. Wyrób lub surowiec w niniejszej pracy opisany jest jako wektor, natomiast proces powstawania wyrobu, traktuje się jako kolejne zmiany składowych wektora wyjściowego.

¹ M. Rymarczyk "O wyznaczaniu przydziałów równoważnych".

Tak więc operator T_{p1} jest odpowiednikiem operacji 1-tej na wyrobie p-tym. Ponieważ cały proces składa się z wielu takich operacji wykonywanych na różnych surowcach, wprowadza się przeto złożenie operatorów $T_M^{\wedge D}$, które można interpretować jako pewien podzbiór D zbioru wszystkich operacji K wykonywanych na podzbiorze detali M .

Ponieważ interesuje nas kolejność działania operatorów należących do złożenia, dlatego też wprowadza się funkcję sterującą α . Często okazuje się, że istnieje kilka operatorów działających na 1-tą składową wektora p-tego, a co za tym idzie, są one nierozróżnialne z punktu widzenia wyniku działania. W celu opisania takich sytuacji wprowadza się zbiór operatorów \mathcal{T}_{p1} . W praktyce oznacza to, że operację 1-tą na detalu p-tym wykonać można na jednym z kilku stanowisk. Odpowiednikiem stanowiska jest tutaj warstwa operatorów. Ponieważ w praktyce wymaga się by operacja wykonywana była na jednym tylko stanowisku, do modelu wprowadza się jeszcze funkcję sterującą β , której zadaniem jest przydzielenie odpowiedniej operacji jednemu tylko stanowisku.

W trzeciej części, po zdefiniowaniu niezbędnych pojęć formułuje się zagadnienie sekwencyjne. Uprzednio jednak wprowadza się funkcje θ oraz \varnothing . Funkcje te pozwalają dane potrzebne do zadania zapisać w możliwie prosty sposób. Ponadto, by można było zdefiniować kryterium optymalności wprowadza się funkcjonal H oraz podaje się sposób wyznaczania jego wartości na dowolnym podzbiorze zbioru operatorów sterowanych funkcjami α i β .

Następnie formułuje się zadanie programowania sekwencyjnego oraz niektóre jego szczególne przypadki.

W części tej podjęto również próbę rozbicia zadania pierwotnego na dwa zadania mniejszych rozmiarów, co poważnie ułatwia procedurę poszukiwania rozwiązania. W tym celu formułuje się twierdzenie o rozbiciu zbioru S na dwa podzbiory niezależne ze względu na warstwy operatorów. Ponieważ zagadnienie to sprowadza się do omówionego uprzednio rozbicia zbioru Γ , do efektywnego rozwiązania omawianego problemu wykorzystuje się algorytm o rozbiciu tego zbioru.

Na zakończenie rozważa się sytuację kiedy zbiór S nie rozbija się na dwa niezależne podzbiory.

1.2. Z a g a d n i e n i e p r z y d z i a ł u .

S e l e k c j a d a n y c h .

1.2.1. Zbiór Γ i jego własności

Niech K, N, M będą zbiorami liczb naturalnych odpowiednio od 1 do k , od 1 do n oraz od 1 do m , tzn.:

$$K = \{1, 2, \dots, k\}$$

$$N = \{1, 2, \dots, n\}$$

$$M = \{1, 2, \dots, m\}$$

Założmy, że:

$$n \geq m$$

Przyjmijmy, że mamy danych k skończonych zbiorów C_i ^{1/} ($i \in K$) oraz zbiór Γ , którego własności podamy dalej. Oznaczmy produkt kartezjański tych zbiorów symbolem C^k , tzn.

^{1/} elementy tych zbiorów nie muszą być liczbami.

$$C^k := C_1 \times C_2 \times \dots \times C_k.$$

Niech S oraz W będą danymi podzbiorami zbioru C^k :

$$S \subset C^k,$$

$$W \subset C^k.$$

Założmy, że liczności tych zbiorów są odpowiednio n oraz m :

$$\text{card}(S) = n,$$

$$\text{card}(W) = m.$$

Niech $R(S, N)$ będzie rodziną różnowartościowych odwzorowań zbioru S na zbiór N , zaś $R(W, M)$ rodziną różnowartościowych odwzorowań zbioru W na zbiór M .

Przyjmijmy, że $\psi \in R(S, N)$ a $\varphi \in R(W, M)$. Zapis s_p będzie wówczas oznaczał, że:

$$s_p \in S \wedge \psi(s) = p \wedge p \in N,$$

i analogicznie, w_q oznacza, że:

$$w_q \in W \wedge \varphi(w) = q \wedge q \in M.$$

Wymieńmy teraz własności zbioru :

$$1^0 \Gamma \subset S \times W,$$

$$2^0 \forall w \in W \exists s \in S: (s, w) \in \Gamma, \quad 1.1$$

$$3^0 i \neq j \wedge (s_i, w_i) \in \Gamma \wedge (s_i, w_j) \in \Gamma \Rightarrow$$

$$\Rightarrow \exists r \neq 1: (s_r, w_i) \in \Gamma \vee (s_r, w_j) \in \Gamma.$$

Elementy zbioru Γ oznaczać będziemy γ_{pq} , gdzie:

$$\gamma_{pq} := (s_p, w_q) \in \Gamma.$$

Oznaczmy przez W_p podzbiór zbioru W tych jego elementów, dla których prawdziwa jest relacja:

$$(s_p, w) \in \Gamma,$$

tzn.:

$$W_p := \{w \in W : (s_p, w) \in \Gamma\}.$$

I analogicznie, symbol S_q oznaczać będzie podzbiór zbioru S tych jego elementów, dla których prawdziwa jest relacja:

$$(s, w_q) \in \Gamma,$$

tzn.:

$$S_q := \{s \in S : (s, w_q) \in \Gamma\}.$$

Z definicji zbioru W_p wynika, że:

$$\bigcup_{p=1}^n W_p = W.$$

Γ_p oznaczać będzie podzbiór zbioru Γ , przy czym:

$$\Gamma_p := \{\gamma_{pq} \in \Gamma : w_q \in W_p\}$$

i podobnie Γ_q^{-1} oznaczać będzie podzbiór zbioru Γ :

$$\Gamma_q^{-1} := \{\gamma_{pq} \in \Gamma : s_p \in S_q\},$$

a zatem:

$$\bigcup_{p=1}^n \Gamma_p = \Gamma$$

i

$$\bigcup_{q=1}^m \Gamma_q^{-1} = \Gamma .$$

Niech teraz $R(S^*, W)$ oznacza rodzinę różnowartościowych odwzorowań zbioru S^* ($S^* \subset S$) na zbiór W .

Twierdzenie 1.1

Istnieje co najmniej jeden podzbiór Γ' zbioru Γ należący do $R(S^*, W)$.

Dowód:

Dla dowodu utworzymy zbiór Γ' i następnie pokażemy, że należy on do $R(S^*, W)$. W tym celu wybierzmy dowolny element zbioru W , niech elementem tym będzie w_q . Element ten wyznacza podzbiór S_q zbioru S . Możliwe są dwie sytuacje, rozpatrzmy je kolejno:

- 1^o do S_q należy takie s_p , które nie należy do żadnego innego S_j ($j \neq q$ oraz $j=1, 2, \dots, m$),
- 2^o każdy element zbioru S_q należy do jakiegoś podzbioru S_j ($j=1, 2, \dots, m$).

W przypadku pierwszym do zbioru Γ' zaliczymy parę (s_p, w_q) .

W przypadku drugim wybierzmy dowolny $s_p \in S_q$ i niech element ten ponadto należy do S_j , co oznacza, że:

$$(s_p, w_q) \in \Gamma \quad \text{oraz} \quad (s_p, w_j) \in \Gamma .$$

Wobec własności 3 zbioru w w zbiorze S istnieć musi takie s_r , że $(s_r, w_j) \in \Gamma$ lub $(s_r, w_q) \in \Gamma$. Przyjmijmy, że $(s_r, w_j) \in \Gamma$ wówczas do zbioru Γ' należeć będzie para (s_r, w_j) oraz para (s_p, w_q)

Jeśli zaś $(s_{r,w_j}) \in \Gamma$ i $(s_{r,w_q}) \in \Gamma$ to do Γ' może należeć (s_{r,w_j}) i (s_{p,w_q}) albo (s_{r,w_q}) i (s_{p,w_j}) .

Powtarzając to postępowanie dla każdego elementu zbioru W (co można zrobić, gdyż jest odwzorowaniem na) otrzymujemy, że należy do $R(S^*, W)$. Co kończy dowód.

Niech teraz $R_\Gamma(S^*, W)$ oznacza rodzinę odwzorowań różnowartościowych zbioru S^* na zbiór W generowanych przez Γ , tzn. takich, które są podzbiórami zbioru Γ , R^+ oznacza zbiór liczb rzeczywistych dodatnich.

Dany jest funkcjonal F :

$$F: \Gamma \rightarrow R^+$$

o własności:

$$F(\Gamma') = \sum_p \sum_q F(\gamma'_{pq}) \quad 1.2$$

gdzie:

$$\Gamma' \in R_\Gamma(S^*, W), \quad \gamma'_{pq} \in \Gamma'.$$

Jak już pokazano wyżej $R_\Gamma(S^*, W) \neq \emptyset$ powstaje zatem naturalne zadanie:

Wyznaczyć odwzorowanie należące $R_\Gamma(S^*, W)$ tak, by funkcjonal F przyjmował wartość najmniejszą z możliwych.

Jeśli Γ^* oznaczać będzie to odwzorowanie, to powyższe zadanie można sformułować:

Wyznaczyć

$$\Gamma^* \in R_\Gamma(S^*, W)$$

tak, by

$$F(\Gamma^*) \rightarrow \min.$$

Zadania tego typu rozwiązuje się metodami programowania matematycznego, a ściślej mówiąc korzysta się tutaj z programowania binarnego, gdzie zadania to ma następującą postać:

Wyznaczyć zmienne x_{pq} tak, by:

$$\sum_p \sum_q x_{pq} F(\gamma_{pq}) \rightarrow \min.$$

przy warunkach:

$$\sum_{p=1}^n x_{pq} = 1 \quad q=1,2,\dots,m$$

$$\sum_{q=1}^m x_{pq} \leq 1 \quad p=1,2,\dots,n$$

$$x_{pq} = 0 \vee 1.$$

Wówczas:

$$\gamma_{pq} \in \Gamma^* \Leftrightarrow x_{pq} = 1.$$

Przykład 1

$$S = \{s_1, s_2, s_3, s_4, s_5\}$$

$$W = \{w_1, w_2, w_3, w_4\}$$

$$\Gamma = \{(s_1, w_1), (s_1, w_2), (s_2, w_1), (s_2, w_4), (s_3, w_2), (s_3, w_3), (s_4, w_3), (s_5, w_4)\}$$

$$F(\gamma_{11}) = 4$$

$$F(\gamma_{12}) = 8$$

$$F(\gamma_{21}) = 3$$

$$F(\gamma_{24}) = 10$$

$$F(\gamma_{32}) = 9$$

$$F(\gamma_{33}) = 5$$

$$F(\gamma_{43}) = 2$$

$$F(\gamma_{54}) = 6$$

Wszystkie te informacje zapisać można w postaci tablicy

$[a_{ij}]$ o wymiarach 5×4 , gdzie:

$$a_{ij} = F(\gamma_{ij}) \quad \text{jeśli} \quad \gamma_{ij} \in \Gamma$$

$$a_{ij} = \infty \quad \text{jeśli} \quad \gamma_{ij} \notin \Gamma$$

	w_1	w_2	w_3	w_4
s_1	4	8	∞	∞
s_2	3	∞	∞	10
s_3	∞	9	5	∞
s_4	∞	∞	2	∞
s_5	∞	∞	∞	6

Mamy tutaj:

$$W_1 = \{w_1, w_2\}$$

$$S_1 = \{s_1, s_2\}$$

$$W_2 = \{w_1, w_4\}$$

$$S_2 = \{s_1, s_3\}$$

$$W_3 = \{w_2, w_3\}$$

$$S_3 = \{s_3, s_4\}$$

$$W_4 = \{w_3\}$$

$$S_4 = \{s_2, s_5\}$$

$$W_5 = \{w_4\}$$

Wymieńmy kilka odwzorowań należących do $R_\Gamma(S^*, W)$, oto one:

$$\{(s_1, w_1), (s_3, w_2), (s_4, w_3), (s_2, w_4)\}$$

$$\{(s_1, w_1), (s_3, w_2), (s_4, w_3), (s_5, w_4)\}$$

$$\{(s_2, w_1), (s_1, w_2), (s_3, w_3), (s_5, w_4)\}$$

itd...

Rozwiązaniem zadania jest:

$$\Gamma^* = \{(s_2, w_1), (s_1, w_2), (s_4, w_3), (s_5, w_4)\}$$

$$\text{i } F(\Gamma^*) = 19.$$

1.2.2. Twierdzenie o rozbiciu zbioru Γ .

Wprowadzając zadania tego typu nie są przedmiotem szczególnych zainteresowań niniejszej pracy, to jednak należy tu zwrócić uwagę na pewien fakt. Idzie mianowicie o to, że jeśli n i m są dostatecznie duże, a więc zadanie jest dużych rozmiarów to poszukiwanie jego rozwiązania nawet przy użyciu maszyny cyfrowej jest czasochłonne. Z drugiej zaś strony często zadanie daje się rozbić na dwa mniejsze i to takie, że ich rozwiązania łącznie, dają rozwiązanie zadania wyjściowego. Problem ten zostanie tu pokrótce omówiony. Niech Γ^1 i Γ^2 oznaczają podzbiory zbioru Γ .

Definicja 1.1

Dwa niepuste podzbiory Γ^1 i Γ^2 zbioru Γ są niezależne jeśli:

$$1^0 \quad \Gamma^1 \cup \Gamma^2 = \Gamma.$$

$$2^0 \quad \gamma_{pq} \in \Gamma^1 \wedge \gamma_{pr} \in \Gamma \wedge \gamma_{sq} \in \Gamma \Rightarrow \gamma_{pr} \in \Gamma^1 \wedge \gamma_{sq} \in \Gamma^1.$$

Lemat 1.1

Zbiór Γ daje się rozbić na dwa niezależne podzbiory jeżeli:

$$\exists L \subset \mathbb{N}: \left(\bigcup_{l \in L} W_l \right) \cap \left(\bigcup_{k \notin L} W_k \right) = \emptyset.$$

Dowód:

Pokażemy, że założenie lematu implikuje warunek 1^0 i 2^0 definicji niezależności.

1^0 Zbiór $\bigcup_{l \in L} W_l$ generuje podzbiór zbioru Γ , a mianowicie $\bigcup_{l \in L} \Gamma_l$. Oznaczmy go Γ^1 . Jeżeli teraz

$p \in \mathbb{N} \wedge p \notin L$ to $W_p \subset \bigcup_{k \notin L} W_k$, a więc Γ_p jest podzbiorem

zbioru $\bigcup_{k \notin L} \Gamma_k$ generowanego przez $\bigcup_{k \notin L} W_k$.

Oznaczmy go Γ^2 .

Tak więc każdy element zbioru należy do Γ^1 lub do Γ^2 , czyli:

$$\Gamma^1 \cup \Gamma^2 = \Gamma.$$

2^0 Załóżmy, że istnieje podzbiór L o którym mowa w lemacie.

Oznaczmy

$$W^1 = \bigcup_{l \in L} W_l$$

$$W^2 = \bigcup_{k \notin L} W_k.$$

Niech $p \in L$ oraz $\gamma_{pq} \in \Gamma^1$ i $\gamma_{pr} \in \Gamma$.

Na mocy definicji zbioru W_p mamy:

$$w_q \in W_p$$

oraz

$$w_r \in W_p.$$

Ponieważ $W_p \subset W^1$ czyli y_{pr} należy do Γ^1 generowanego przez W^1 . Niech teraz $y_{sq} \in \Gamma$, a więc $w_q \in W_s$, ale poprzednio $w_q \in W_p$ a więc W_s i W_p muszą być podzbiorami tego samego zbioru W^1 lub W^2 (w przeciwnym bowiem razie $W^1 \cap W^2 = w_q$ co przeczy założeniu lematu). Ponieważ $W_p \subset W^1$ to i $W_s \subset W^1$, czyli y_{sq} należy do Γ^1 . Co kończy dowód.

Niech:

$$\Gamma^* \in R_{\Gamma}(S^*, W)$$

i minimalizuje funkcjonal F na zbiorze Γ . Γ^1 i Γ^2 będą podzbiorami zbioru Γ . Γ^{1*} i Γ^{2*} będą odwzorowaniami różnowartościowymi minimalizującymi funkcjonal F odpowiednio na Γ^1 i Γ^2 .

Twierdzenie 1.2

Jeżeli zbiór Γ rozbija się na dwa niezależne podzbiory Γ^1 i Γ^2 to:

$$\Gamma^{1*} \cup \Gamma^{2*} = \Gamma^*$$

$$F(\Gamma^{1*}) + F(\Gamma^{2*}) = F(\Gamma^*).$$

Prawdziwość powyższego twierdzenia wynika bezpośrednio z addytywności funkcjonału F (patrz 1.2) i definicji niezależności zbiorów Γ^1 i Γ^2 .

Tak więc jeśli zbiór Γ rozbija się na dwa niezależne podzbiory to zamiast jednego zadania mamy dwa zadania, ale mniejszych rozmiarów.

Przykład 2

$$S = \{s_1, s_2, s_3, s_4, s_5, s_6, s_7\}$$

$$W = \{w_1, w_2, w_3, w_4, w_5, w_6\}$$

Zbiór oraz funkcja F zapiszemy w postaci tablicy

	w_1	w_2	w_3	w_4	w_5	w_6
s_1	7	∞	∞	∞	9	∞
s_2	∞	∞	4	∞	∞	∞
s_3	∞	10	∞	∞	6	∞
s_4	∞	∞	8	12	∞	∞
s_5	∞	∞	∞	8	∞	4
s_6	∞	∞	∞	11	∞	10
s_7	∞	∞	∞	∞	10	∞

Rozwiązaniem zadania jest:

$$\Gamma^* = \{(s_1, w_1), (s_3, w_2), (s_2, w_3), (s_6, w_4), (s_7, w_5), (s_5, w_6)\}$$

$$F(\Gamma^*) = 7+10+4+11+10+4 = 46$$

Zbiór Γ rozpada się na dwa niezależne podzbiory:

	w_1	w_2	w_5
s_1	7	∞	9
s_3	∞	10	6
s_7	∞	∞	10

	w_3	w_4	w_6
s_2	4	∞	∞
s_4	8	12	∞
s_5	∞	8	4
s_6	∞	11	10

Zbiór Γ^{1*} minimalizujący funkcjonal F na pierwszym z nich:

$$\Gamma^{1*} = \{(s_1, w_1), (s_3, w_2), (s_7, w_5)\}$$

i $F(\Gamma^{1*}) = 7 + 10 + 10 = 27$

Zbiór Γ^{2*} minimalizujący funkcjonal F na drugim:

$$\Gamma^{2*} = \{(s_2, w_3), (s_6, w_4), (s_5, w_6)\}$$

i $F(\Gamma^{2*}) = 4 + 11 + 4 = 19$

Widać, że:

$$\Gamma^* = \Gamma^1 \cup \Gamma^2$$

i $F(\Gamma^*) = F(\Gamma^{1*}) + F(\Gamma^{2*})$.

1.2.3. Algorytm rozbicia

Algorytm, który pozwala stwierdzić czy zbiór Γ rozbija się na niezależne podzbiory, a jeśli tak, to je wyznaczyć jest następujący:

Krok 1.

Wybieramy dowolny γ_{pq} .

$$\Gamma^1 := \{\gamma_{pq}\}$$

i przechodzimy do kroku 2.

Krok 2.

Jeśli w zbiorze $\Gamma - \Gamma^1$ istnieje Γ_p , to:

$$\Gamma^1 := \Gamma_p$$

i przechodzimy do kroku 3.

Jeśli nie to przechodzimy do kroku 3.

Krok 3.

Jeśli w zbiorze $\Gamma - \Gamma^1$ istnieją takie Γ_q^{-1} , że $(s, w_q) \in \Gamma^1$ to:

$$\Gamma^1 := \Gamma^1 \cup \left(\bigcup_{q \in Q} \Gamma_q^{-1} \right) \text{ gdzie: } \{Q = q \mid M: (s, w_q) \in \Gamma^1\}$$

i przechodzimy do kroku 4.

Jeśli nie to koniec postępowania.

Krok 4.

Jeśli w zbiorze $\Gamma - \Gamma^1$ istnieją Γ_d takie, że $(s_d, w) \in \Gamma^1$ to:

$$\Gamma^1 := \Gamma^1 \cup \left(\bigcup_{d \in D} \Gamma_d \right) \text{ gdzie } D = \{d \in N: (s_d, w) \in \Gamma^1\}$$

i przechodzimy do kroku 3.

Jeśli nie to koniec postępowania.

Algorytm ten jest skończony co wynika ze skończoności zbioru

Γ .

Założmy, że w wyniku tego postępowania otrzymaliśmy zbiór Γ^1 i $\Gamma - \Gamma^1 \neq \emptyset$. Ponieważ każdy γ_{pq} będzie należał do Γ^1 bądź do $\Gamma^2 = \Gamma - \Gamma^1$ więc otrzymane zbiory spełniają pierwszy warunek niezależności:

$$\Gamma^1 \cup \Gamma^2 = \Gamma$$

Niech $y_{pq} \in \Gamma^1$, wobec kroku 4 algorytmu do zbioru Γ^1 należeć też będzie każdy element $y_{pr} \in \Gamma^1$, a krok 3 algorytmu zabezpiecza zaliczenie do zbioru Γ^1 każdego elementu $y_{sq} \in \Gamma^1$. Otrzymane zbiory spełniają zatem i drugi warunek niezależności.

Jeśli zaś $\Gamma - \Gamma^1 = \emptyset$ to zbiór Γ nie daje się rozbić na dwa niezależne podzbiory.

Postępowanie całe można powtórzyć w stosunku do zbioru $\Gamma - \Gamma^1$, gdyż zdarzyć się może, że zbiór ten rozbija się znowu na dwa (lub więcej) niezależne podzbiory.

W wyniku rozważań przeprowadzonych w tej części uzyskaliśmy zbiór Γ^* , który jest odwzorowaniem wzajemnie jednoznacznym zbioru S^* na zbiór W . Odwzorowanie to wybiera ze zbioru S podzbiór S^* taki, że:

$$\forall s \in S^* \exists w \in W: (s, w) \in \Gamma^* .$$

Wobec tego, że w dalszym ciągu nie będzie nas interesować zbiór S a tylko jego podzbiór S^* na jego oznaczenie używać będziemy symbolu S , pamiętając, że:

$$S = \{s \in S: (s, w) \in \Gamma^*\} .$$

Γ^* jest odwzorowaniem różnowartościowym, a zatem dla każdego $s \in S$ wyznacza w sposób jednoznaczny $w \in W$. Można więc napisać, że:

$$\Gamma^*(s) = w .$$

Niech $R(S, M)$ będzie rodziną różnowartościowych odwzorowań zbioru S na zbiór M . Wybierzmy z rodziny tej takie odwzo-

rowanie Ψ by:

$$\forall s \in S \quad \Psi(s) = \varphi(\Gamma^*(s)),$$

gdzie

$$\varphi \in R(W, M).$$

1.3. Podstawowe pojęcia niezbędne do sformułowania zadania programowania sekwencyjnego

1.3.1. Działania operatora T_{p1} i złożenie T_{pD} .

Przypomnijmy, że zbiory S oraz W są podzbiórmi zbioru C^k , a więc ich elementy są k -wymiarowymi "wektorami".

Każdy z tych "wektorów" jest numerowany liczbami od 1 do m , czyli:

$$s_p = (s_{p1}, s_{p2}, \dots, s_{pk})$$

$$w_p = (w_{p1}, w_{p2}, \dots, w_{pk})$$

gdzie $p = 1, 2, \dots, m$.

Dla każdego $p \in M$ i każdego $l \in K$ skonstruujemy funkcję ξ_{p1} :

$$\xi_{p1} : K \rightarrow C_1$$

gdzie:

$$\xi_{p1}(i) := \begin{cases} s_{pi} & \text{dla } i \neq 1 \\ w_{p1} & \text{dla } i = 1 \end{cases}$$

$$p = 1, 2, \dots, m$$

$$l \neq 1, 2, \dots, k$$

$$i = 1, 2, \dots, k.$$

Przyjmijmy ponadto, a oznaczenie

$$g_{p1}(K) := (g_{p1}(i))^{1/} \quad 1.3$$

gdzie naturalnie $g_{p1}(i) \in C^k$.

Zdefiniujmy teraz założenie funkcji g_{p1} , będzie to definicja rekurencyjna. Niech D będzie niepustym podzbiorem zbioru K . Symbol g_{pD} oznaczał będzie złożenie funkcji g_{pr} takich, że $r \in D$. Jeżeli $t \notin D$ to złożenie funkcji g_{pt} z uprzednio już wyznaczonym złożeniem g_{pD} zapisywać będziemy:

$$g_{pt} \times g_{pD}$$

Zdefiniujmy teraz sposób uzyskiwania takiego złożenia.

$$g_{pt} \times g_{pD} := g_{pD'}$$

$$i \quad g_{pD'}(i) := \begin{cases} g_{pD}(i) & \text{dla } i \neq t \\ g_{pt}(i) & \text{dla } i = t \end{cases}$$

natomiast:

$$g_{pD}(i) := (g_{pr} \times g_{pD''})(i) = \begin{cases} g_{pD''}(i) & \text{dla } i \neq r \\ g_{pr}(i) & \text{dla } i = r \end{cases}$$

gdzie:

$$\begin{aligned} D &\subset K \\ D' &= D \cup \{t\} \\ D'' &= D - \{r\}. \end{aligned}$$

^{1/} symbol $(g_{p1}(i))$, oznacza ciąg elementów $g_{p1}(i)$, $i=1,2,\dots,k$

Załóżmy teraz, że dany jest operator T_{p1} , którego działanie na s_p jest zdefiniowane jak następuje:

$$T_{p1}(s_p) := g_{p1}(K).$$

Zdefiniujmy teraz złożenie operatorów T_{p1} . Symbol T_{pD} oznaczał będzie złożenie wszystkich operatorów T_{pr} , takich że $r \in D$. Reguła składania dowolnego operatora T_{p1} gdzie $1 \notin D$ z wcześniej już wyznaczonym złożeniem T_{pD} jest następująca:

$$(T_{p1} \times T_{pD})(s_p) := (T_{p1} g_{pD}(K)) := g_{pD'}(K).$$

gdzie:

$$D \subset K$$

$$D' = D \cup \{1\}.$$

Z definicji operatora T_{p1} oraz złożenia T_{pD} wynikają następujące wnioski:

Wniosek I

Wynik działania złożenia T_{pD} nie zależy od kolejności w jakiej składa się operatory należące do T_{pD} .

Wniosek II

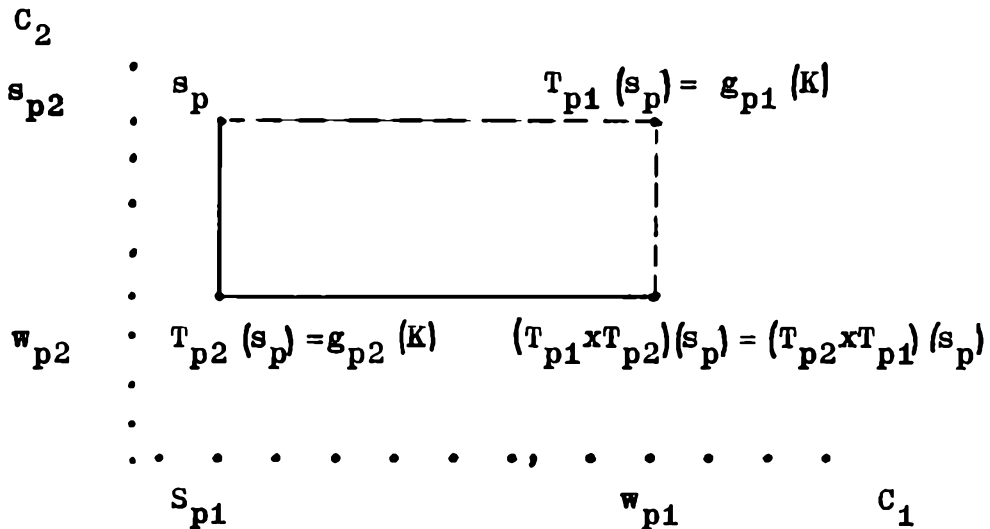
Jeśli $D = K$ to w wyniku działania złożenia T_{pK} uzyska się w_p , czyli:

$$T_{pK}(s_p) = \dots = w_p.$$

Wniosek III

Jeżeli $D' \neq D''$ to wynik działania złożenia $T_{pD'}$, będzie inny niż wynik działania złożenia $T_{pD''}$.

Wnioski te mają przejrzystą ilustrację graficzną:



Linia ciągłą zaznaczono złożenia $T_{p1} \times T_{p2}$, linią przerywaną $T_{p2} \times T_{p1}$.

Mamy tutaj $D = K = \{1, 2\}$. Budując złożenie $T_{p1} \times T_{p2}$ otrzymaliśmy ten sam punkt co przy złożeniu $T_{p2} \times T_{p1}$. Punkt ten ma współrzędne (w_{p1}, w_{p2}) a więc jest już elementem zbioru W . Ponadto $D' = \{1\}$, $D'' = \{2\}$ i widać, że $T_{p1}(s_p) = (w_{p1}, s_{p2})$, natomiast $T_{p2}(s_p) = (s_{p1}, w_{p2})$.

Ponieważ wynik działania złożenia T_{pD} nie zależy od kolejności w jakiej składa się operatory należące do tego złożenia, można zatem zamiast "złożenie T_{pD} " mówić "zbiór T_{pD} ". Tym niemniej dla wyspecyfikowania kolejności w jakiej składa się te operatory wprowadzimy pojęcie funkcji sterującej.

W tym celu wprowadzimy następujące oznaczenia:

$$b := \text{card}(T_{pD})$$

$$B := \{1, 2, \dots, b\}.$$

Wówczas funkcję sterującą oznaczymy α_{pD} , gdzie:

$$\alpha_{pD} \in R(B, T_{pD})$$

gdzie w szczególnym przypadku $D = K$.

Zapis $T_{pD}(s_p) \alpha_{pD}$ czytamy: "złożenie zbior T_{pD} jest sterowane funkcją α_{pD} ", zaś sterowanie to jest następujące:

$$T_{pD}(s_p) \alpha_{pD} := (\alpha(b) \times \dots \times \alpha(2) \times \alpha(1)) (s_p) \quad 1/$$

Inaczej mówiąc funkcja sterująca α_{pD} ustala kolejność składania operatorów należących do T_{pD} . Najogólniej biorąc funkcji tych może być $b!$. Działanie złożenia oraz funkcji sterującej zilustrujemy na następującym przykładzie:

Przykład 3

Mamy następujące dane:

$$s_1 \in S \subset C^4, \quad w_1 \in W \subset C^4, \quad D = \{1, 3, 4\}$$

czyli $D \neq K$ bowiem $K = \{1, 2, 3, 4\}$. Wobec tego $B = \{1, 2, 3\}$ zaś $T_{1D} = \{T_{11}, T_{13}, T_{14}\}$. Dana jest również funkcja sterująca:

$$\alpha_{1D}(1) = T_{14}$$

$$\alpha_{1D}(2) = T_{11}$$

$$\alpha_{1D}(3) = T_{13}$$

Sterowanie i działanie złożenia T_{1D} jest następujące:

$$\begin{aligned} T_{1D}(s_1) \alpha_{1D} &= (T_{13} \times T_{11} \times T_{14})(s_1) = (T_{13} \times T_{11} \times T_{14})(s_{11}, s_{13}, s_{14}) = \\ &= (T_{13} \times T_{11})(s_{11}, s_{12}, s_{13}, w_{14}) = T_{13}(w_{11}, s_{12}, s_{13}, w_{14}) = \\ &= (w_{11}, s_{12}, w_{13}, w_{14}) \end{aligned}$$

^{1/}opuszczono tu indeksy przy funkcji α ponieważ są one identyczne jak indeksy zbioru sterowanego.

1.3.2. Złożenie $T_{M',D}$

Dotychczas mówiliśmy o działaniu operatora lub złożeniu operatorów na dowolny element zbioru S . Zdefiniujemy teraz działanie takich operatorów na dowolny podzbiór zbioru S .

Niech $r \neq p$, złożenie funkcji g_{rt} i funkcji g_{pl} zdefiniujemy następująco:

$$(g_{rt} \times g_{pl})(K) := g_{rt}(K), g_{pl}(K)$$

gdzie po prawej stronie równości występuje ciąg $(g_{rt}(i))$ oraz ciąg $(g_{pl}(i))$ (patrz 1.3).

Jeśli M' jest podzbiorem (niepustym) zbioru M , to $g_{M',D}$ oznaczać będzie złożenie funkcji g_{rt} takich, że $r \in M'$, $t \in D$ i złożenie to zdefiniujemy podobnie jak złożenie g_{pD} , rekurencyjnie:

$$(g_{pl} \times g_{M',D})(K) := \begin{cases} g_{M',D}(K), g_{pl}(K) & \text{gdy } p \notin M' \\ g_{M',D'}(K) & \text{gdy } p \in M' \end{cases}$$

gdzie:

$$M' \subset M$$

$$D \subset K$$

$$D' = D \cup \{1\}$$

zaś

$$g_{M',D}(K) = \{g_{rt}(K) \in C^K : g_{rt} \in g_{M',D}\}.$$

Oznaczmy teraz symbolem $T_{M',D}$ złożenie operatorów T_{rt} , złożenie to działa na podzbiór zbioru S tych jego elementów s_r dla których $r \in M'$, podzbiór ten oznaczmy $S_{M'}$. Wówczas

$$(T_{p1} \times T_{M',D}) (S_{M''}) := T_{p1} (g_{M',D} (K)) := (g_{p1} \times g_{M',D}) (K).$$

gdzie:

$$M'' = M' \quad \text{gdy } p \in M'$$

$$M'' = M' \cup \{p\} \quad \text{gdy } p \notin M'.$$

Tutaj podobnie jak dla złożenia T_{pD} wynik działania zbioru operatorów $T_{M',D}$ nie zależy od kolejności składania operatorów do niego należących. Wobec tego podobnie jak poprzednio wprowadzimy funkcję sterującą $\alpha_{M'D}$, która ustali kolejność składania operatorów.

Oznaczmy:

$$b := \text{card}(T_{M',D})$$

$$B := \{1, 2, \dots, b\}$$

wtedy

$$\alpha_{M',D} \in R(B, T_{M',D})$$

i sterowanie jest następujące:

$$T_{M',D} (S_{M'}) := (\alpha(b) \times \dots \times \alpha(2) \times \alpha(1)) (S_{M'})^{1/}$$

Przykład 4

$$S = \{s_1, s_2, s_3\} \subset C^3, \quad W = \{w_1, w_2, w_3\} \subset C^3.$$

$$\text{Mamy zbiór } T_{M',D} = \{T_{11}, T_{13}, T_{22}, T_{23}, T_{31}, T_{32}\}$$

i niech

$$\alpha_{(1)} = T_{11}$$

$$\alpha_{(2)} = T_{22}$$

^{1/} Tutaj podobnie jak poprzednio opuszczono indeksy przy funkcji α

$$\alpha(3) = T_{31}$$

$$\alpha(4) = T_{23}$$

$$\alpha(5) = T_{32}$$

$$\alpha(6) = T_{13}$$

Wówczas:

$$\begin{aligned} T_{M^D}(s_1, s_2, s_3)_{\alpha} &= (T_{13} x^T_{32} x^T_{23} x^T_{31} x^T_{22} x^T_{11}) (s_1, s_2, s_3) = \\ &= (T_{13} x^T_{32} x^T_{23} x^T_{31} x^T_{22} x^T_{11}) ((s_{11}, s_{12}, s_{13}), (s_{21}, s_{22}, s_{23}), (s_{31}, s_{32}, s_{33})) = \\ &= (T_{13} x^T_{32} x^T_{23} x^T_{31} x^T_{22}) ((w_{11}, s_{12}, s_{13}), (s_{21}, s_{22}, s_{23}), (s_{31}, s_{32}, s_{33})) = \\ &= (T_{13} x^T_{32} x^T_{23} x^T_{31}) ((w_{11}, s_{12}, s_{13}), (s_{21}, w_{22}, s_{23}), (s_{31}, s_{32}, s_{33})) = \\ &= (T_{13} x^T_{32} x^T_{23}) ((w_{11}, s_{12}, s_{13}), (s_{21}, w_{22}, s_{23}), (w_{31}, s_{32}, s_{33})) = \\ &= (T_{13} x^T_{32}) ((w_{11}, s_{12}, s_{13}), (s_{21}, w_{22}, w_{23}), (w_{31}, s_{32}, s_{33})) = \\ &= T_{13} ((w_{11}, s_{12}, s_{13}), (s_{21}, w_{22}, w_{23}), (w_{31}, w_{32}, s_{33})) = \\ &= (w_{11}, s_{12}, w_{13}), (s_{21}, w_{22}, w_{23}), (w_{31}, w_{32}, s_{33}). \end{aligned}$$

1.3.3. Zbiór operatorów T_{pl} i rodzina T_{MK} .

Dotychczas zakładaliśmy, że jest jeden tylko operator działający na l-tą składową elementu p-tego i oznaczaliśmy ten operator T_{pl} . Przyjmiemy teraz, że operatorów takich jest nie mniej niż jeden, przy czym są one nierozróżnialne z punktu widzenia wyniku ich działania. Dla rozróżnienia tych operatorów wprowadzimy jeszcze jeden indeks i operatory te oznaczać będziemy T_{plr} .

Niechaj dany będzie zbiór N liczb naturalnych $1, 2, \dots, n$ i $R(N)$ niech oznacza rodzinę wszystkich niepustych podzbiorów zbioru N .

Dana jest funkcja ω :

$$\omega: M \times K \rightarrow R(N).$$

przy czym:

$$\bigcup_{p, l} \omega(p, l) = N.$$

Oznaczmy:

$$N_{pl} := \omega(p, l)$$

$$n_{pl} := \text{card}(N_{pl}).$$

Wprowadźmy teraz zbiór operatorów \mathcal{T}_{pl} , który zdefiniujemy:

$$\mathcal{T}_{pl} := \{T_{plr} : \forall r \in N_{pl}, T_{plr}(s_p) = T_{pl}(s_p)\}.$$

Z powyższej definicji wynika, że:

$$\text{card}(\mathcal{T}_{pl}) = n_{pl}.$$

Ponieważ, zgodnie z definicją zbioru \mathcal{T}_{pl} operatory T_{plr} są nierozróżnialne ze względu na wynik działania, a interesować nas będzie działanie tylko jednego z nich, wprowadzimy przeto dla zbioru \mathcal{T}_{pl} funkcję sterującą β_{pl} :

$$\beta_{pl} \in R(\{1, 2, \dots, n_{pl}\}, N_{pl}).$$

Zapis $\mathcal{T}_{pl}^{\beta_{pl}}$ oznaczał będzie, że zbiór \mathcal{T}_{pl} jest sterowany funkcją β_{pl} , które to sterowanie polega na następującym:

$$\mathcal{T}_{pl}^{\beta_{pl}}(s_p) \equiv T_{p,1}, \quad \beta_{pl}^{(1)}(s_p).$$

Innymi słowy, funkcja sterująca β_{pl} wybiera ze zbioru β_{pl} jeden tylko operator.

Jeśli $\mathcal{T}_{M'D}$ oznaczać będzie rodzinę zbiorów operatorów \mathcal{T}_{pl} takich, że $\mathcal{T}_{pl} \in \mathcal{T}_{M'D}$, to zapis $\mathcal{T}_{M'D}^{\beta}(S_M)_{\alpha}$ należy rozumieć: "Rodzina operatorów $M'D$ jest sterowana funkcjami α i β . Przy czym każdy ze zbiorów \mathcal{T}_{pl} należący do rodziny jest sterowany funkcją β_{pl} natomiast funkcja α ustala kolejność składania zbiorów \mathcal{T}_{pl} należących do tej rodziny". Opuszczono tu indeksy funkcji sterujących, co nie powinno prowadzić do nieporozumień ponieważ indeksy te są takie jak indeksy zbioru sterowanego.

Gdy w szczególności:

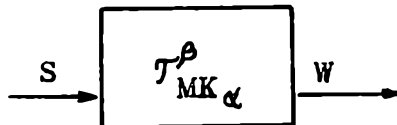
$$M' = M$$

$$D = K$$

to:

$$\forall \alpha \wedge \forall \beta \mathcal{T}_{MK}^{\beta}(S)_{\alpha} = W,$$

co schematycznie można przedstawić



Przykład 5

$$S = \{s_1, s_2, s_3\} \subset C^3, \quad W = \{w_1, w_2, w_3\} \subset C^3$$

$$N = \{1, 2, 3, 4\}$$

$$\alpha_{MK} = \mathcal{T}_{32} \times \mathcal{T}_{21} \times \mathcal{T}_{23} \times \mathcal{T}_{13} \times \mathcal{T}_{33} \times \mathcal{T}_{22} \times \mathcal{T}_{12} \times \mathcal{T}_{31} \times \mathcal{T}_{11}$$

$$\begin{array}{ll}
 \omega(1,1) = \{1,2\} & \beta_{11}(1) = 1 \\
 \omega(1,2) = \{2,3,4\} & \beta_{12}(1) = 4 \\
 \omega(1,3) = \{3,4\} & \beta_{13}(1) = 3 \\
 \omega(2,1) = \{1,3\} & \beta_{21}(1) = 3 \\
 \omega(2,2) = \{2\} & \beta_{22}(1) = 2 \\
 \omega(2,3) = \{3,4\} & \beta_{23}(1) = 4 \\
 \omega(3,1) = \{1,4\} & \beta_{31}(1) = 1 \\
 \omega(3,2) = \{1,2,3,4\} & \beta_{32}(1) = 2 \\
 \omega(3,3) = \{2,3\} & \beta_{33}(1) = 3 .
 \end{array}$$

Wówczas:

$$\begin{aligned}
 \mathcal{T}_{MK}^{\beta}(S)_{\alpha} &= (T_{322}x^T T_{213}x^T T_{234}x^T T_{133}x^T T_{333}x^T T_{222}x^T T_{124}x^T T_{311}x^T T_{111}) (S) = \\
 \dots\dots &= \{(w_{11}, w_{12}, w_{13}), (w_{21}, w_{22}, w_{23}), (w_{31}, w_{32}, w_{33})\} .
 \end{aligned}$$

Wprowadźmy teraz następujące pojęcia:

Definicja 1.2

Jeśli zbiór N jest zbiorem jednoelementowym to działanie operatorów nazywać będziemy działaniem szeregowym.

Definicja 1.3

Jeśli zbiór N jest zbiorem n -elementowym ($n > 1$) to działanie operatorów nazywać będziemy działaniem potencjalnie n -równoległym,

Definicja 1.4

Jeśli każdy ze zbiorów $\mathcal{T}_{p1} \in \mathcal{T}'_{MK}$ jest sterowany funkcją β_{p1} i $\text{card}(\cup_{p,1} \beta_{p1}^{(i)}) = n'$ ($1 < n' \leq n$) to działanie operatorów nazywać będziemy działaniem n' -równoległym.

Z powyższych definicji wynikają następujące wnioski:

Wniosek I

Jeżeli działanie operatorów jest działaniem szeregowym to:

$$\mathcal{T}_{MK} = T_{MK}.$$

Wniosek ten jest oczywisty jeśli się zważy, że zbiór N jest zbiorem jednoelementowym co oznacza, że każdy zbiór operatorów \mathcal{T}_{p1} jest również zbiorem jednoelementowym. Mamy zatem sytuację taką jak przy zbiorze T_{MK} .

Wniosek II

Jeżeli działanie operatorów jest działaniem potencjalnie n' -równoległym, to działanie to może być działaniem co najwyżej n' -równoległym.

Wniosek III

Jeżeli działanie operatorów jest działaniem n' -równoległym, to rodzinę \mathcal{T}_{MK} rozbić można na n' podzbiorów

$$\{T_{M_1 D_1 1}\} \cdot \{T_{M_2 D_2 2}\}, \dots, \{T_{M_{n'} D_{n'} n'}\}$$

gdzie:

$$\bigcup_{i=1}^{n'} M_i = M, \quad \bigcup_{i=1}^{n'} D_i = K.$$

Wniosek również oczywisty, gdyż:

$$T_{M_i D_i i} = \left\{ T_{pli} \in \mathcal{T}_{MK} : \beta_{pl}^{(1)} = i \right\}.$$

Definicja 1.5

Każdy podzbiór $T_{M_i D_i i}$ nazywać będziemy i -tą warstwą operatorów. Warstwę tę oznaczymy Z_i $i = 1, 2, \dots, n'$.

Analogicznie można zdefiniować potencjalną warstwę operatorów wychodząc od działania potencjalnie n -równoległego, a więc nie sterowanego funkcją β .

Różnica między warstwą operatorów a potencjalną warstwą operatorów polega na tym, że przy podziale na warstwy potencjalne operator T_{pl} może należeć do różnych warstw Z_i , gdzie $i \in N_{pl} = \omega(p, l)$ co nie może się zdarzyć przy warstwach operatorów.

Niech teraz Z_i będzie i -tą warstwą operatorów (potencjalną lub nie) i $s_p \in S$.

Definicja 1.6

Warstwa Z_i działa na s_p wtedy i tylko wtedy, gdy istnieje operator T_{pli} ($l=1, 2, \dots, k$), który działa na s_p .

Fakt, że warstwa Z_1 działa na s_p zapisujemy $Z_1(s_p)$.

Czyli:

$$Z_1(s_p) := \{ p_{li} \in Z_1 : T_{pli}(s_p) \}$$

Definicja 1,7

Elementy s_p oraz s_r zbioru S są zależne ze względu na warstwy operatorów (potencjalne lub nie), jeśli istnieje warstwa (odpowiednio potencjalna lub nie) Z_1 która działa na s_p i na s_r .

Jak wynika z definicji warstwy operatorów z faktu, że dwa elementy są zależne ze względu na potencjalne warstwy operatorów nie wynika, że będą one zależne ze względu na warstwy operatorów. Natomiast implikacja w drugą stronę jest prawdziwa, gdyż działanie potencjalne zawiera wszystkie możliwe sterowania funkcją β .

Definicja 1.8

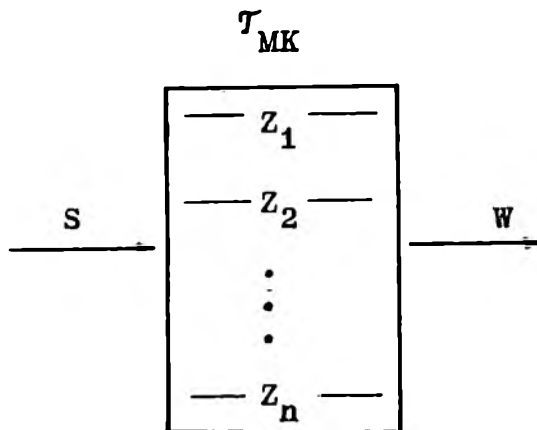
Zbiór S jest zbiorem elementów zależnych ze względu na warstwy (potencjalne lub nie) operatorów, jeśli nie istnieją dwa podzbiory S' i S'' zbioru S takie, że:

$$1^0 \quad S' \cup S'' = S$$

2⁰ żaden element zbioru S' nie jest zależny ze względu na warstwy (odpowiednio potencjalne lub nie) z żadnym elementem zbioru S'' .

Założmy, że dany jest zbiór S elementów niezależnych ze względu na warstwy operatorów^{1/}, a działanie zbioru operatorów \mathcal{T}_{MK} jest działaniem potencjalnie n -równoległym i jest sterowane funkcją α a Z_1, Z_2, \dots, Z_n są potencjalnymi warstwami operatorów.

Działanie takie można przedstawić na schemacie:



Jeśli zbiór \mathcal{T}_{MK} sterowany będzie jeszcze funkcją β , to schemat się nie zmieni, co najwyżej liczba warstw będzie mniejsza.

1.4. Zagadnienie sekwencyjne.

Problem podstawowy.

1.4.1. Dalsze elementy modelu. Przygotowanie danych.

W części tej chcemy zaproponować pewien sposób zapisywania danych, uwzględniając postulat by sposób ten był możliwie najprostszy.

^{1/} Problem czy zbiór S jest zbiorem elementów zależnych ze względu na warstwy operatorów, a jeśli nie, to jak wyznaczyć podzbiory S' i S'' , o których mowa w definicji, zostanie omówiony w dalszej części.

W tym celu wprowadzimy funkcję θ , którą zdefiniujemy jak następuje:

dla danej rodziny \mathcal{T}_{MK} i danego zbioru N

$$\theta : N \times (M \times K) \rightarrow \{0,1\}$$

gdzie:

$$\theta(r, (p, l)) := \begin{cases} 1 & \text{jeśli } r \in \omega(p, l) \\ 0 & \text{w przeciwnym przypadku} \end{cases}$$

Funkcję tę zapisać można w postaci macierzy. Dla przykładu 5 mamy:

	1,1	1,2	1,3	2,1	2,2	2,3	3,1	3,2	3,3
$\theta =$	1	0	0	1	0	0	1	1	0
	2	1	0	0	1	0	0	1	1
	3	0	1	1	0	1	0	1	1
	4	0	1	0	0	1	1	1	0

Mając funkcję θ można również zbudować funkcję, oznaczmy ją \mathcal{T} , gdzie:

$$\mathcal{T} : N \rightarrow M \times K$$

gdzie:

$$\mathcal{T}(i) := (p, l) \iff i \in \omega(p, l)$$

Zarówno jedna jak druga funkcja opisują działanie operatorów potencjalnie n -równoległe.

Jeśli dana jest funkcja sterująca α i β to można zbudować funkcję opisującą działanie operatorów. Funkcję tę oznaczmy

θ_{α}^{β} , daje się ona zapisać również w postaci macierzy zbudowanej według poniższych zasad:

1^o kolumnę j-tą znakujemy parą (p,l) jeśli $\alpha(j) = \mathcal{T}_{pl}$,

2^o w kolumnie oznaczonej parą (p,l) występuje jedna jedynka. Znajduje się ona w wierszu i-tym jeśli

$$\beta_{pl}(i) = 1.$$

Dla naszego przykładu mamy:

$$\theta_{\alpha}^{\beta} =$$

	1,1	3,1	1,2	2,2	3,3	1,3	2,3	2,1	3,2
1	1	1	0	0	0	0	0	0	0
2	0	0	0	1	0	0	0	0	1
3	0	0	0	0	1	1	0	1	0
4	0	0	1	0	0	0	1	0	0

Funkcja ta opisuje działanie n'-równoległe. W naszym przypadku $n'=n$ i działanie jest działaniem 4-równoległym, mamy bowiem cztery warstwy operatorów:

$$\{T_{111}, T_{311}\}, \{T_{222}, T_{322}\}, \{T_{333}, T_{133}, T_{213}\}, \{T_{124}, T_{234}\}.$$

Nie definiując, wyróżnimy dwa rodzaje działania operatorów

\mathcal{T}_{MK} . Rodzaj działania zależy od funkcji sterujących α i β .

Dwa operatory mogą działać kolejno lub współbieżnie.

Sformułujemy teraz założenia dotyczące rodzaju działania

dwóch operatorów należących do \mathcal{T}_{MK} :

- 1^o dwa operatory należące do jednej warstwy działają kolejno,
- 2^o każde dwa operatory, z których jeden należy do $\mathcal{J}_{p_1^1}$ a drugi do $\mathcal{J}_{p_1^2}$ działają kolejno niezależnie od tego czy należą do jednej czy do różnych warstw,
- 3^o wszystkie inne operatory mogą działać **współbieżnie**, przy czym jeśli operator należący do $\mathcal{J}_{p_1^1}$ może działać **współbieżnie** z operatorem należącym do $\mathcal{J}_{p_2^2}$ i operatorem należącym do $\mathcal{J}_{p_3^3}$ to działa on z tym, dla którego wartość funkcji α^{-1} jest mniejsza (α^{-1} jest funkcją odwrotną do α).

Wracając do naszego przykładu, operatory T_{111} i T_{311} działają kolejno ponieważ należą do jednej warstwy. Podobnie operatory T_{111} i T_{133} działają kolejno ponieważ działają one na ten sam element zbioru S . Operator T_{222} może działać **współbieżnie** z operatorem T_{111} lub operatorem T_{311} lub też z operatorem T_{124} . Ponieważ jednak:

$$\alpha^{-1}(T_{111}) < \alpha^{-1}(T_{311}) < \alpha^{-1}(T_{124})$$

to operator T_{222} działa **współbieżnie** z operatorem T_{111} . Założenia dotyczące rodzaju działania operatorów opisują strukturę działania operatorów należących do \mathcal{J}_{MK} . Struktura ta może być opisana formalnie za pomocą funkcji, którą nazwiemy \varnothing .

Niech A oznacza zbiór liczb naturalnych od 1 do a ($a \leq b$)^{1/}

^{1/} przypomnijmy, że $b = \text{card}(\mathcal{J}_{MK})$

Wówczas funkcja \varnothing jest zdefiniowana jak następuje:

$$\varnothing : N \times A \rightarrow M \times K$$

gdzie:

$$\varnothing(i_1, 1) := (p_1, l_1) \text{ takie, że } \alpha(1) = T_{p_1 l_1} \wedge \beta_{p_1 l_1}(1) = i_1$$

$$\varnothing(i, 1) := (p, l) \text{ takie, że } T_{pl} \text{ działa współbieżnie}$$

$$\text{z } T_{p_1 l_1} \wedge \beta_{pl}(1) = i,$$

$$\varnothing(i_1, j) := (p_2, l_2) \text{ takie, że } T_{p_2 l_2} \text{ działa kolejno}$$

z którymkolwiek z operatorów

$$T_{\varnothing(N, j-1)} \wedge \beta_{p_2 l_2}(1) = i_1,$$

$$\varnothing(i, j) := (p, l) \text{ takie, że } T_{pl} \text{ działa współbieżnie}$$

$$\text{z } T_{p_2 l_2} \wedge \beta_{pl}(1) = i.$$

Funkcję \varnothing zapisać można w postaci "macierzy", którą uzyskuje się z macierzy odpowiadającej funkcji θ_α^β biorąc kolejno od pisrewzej do ostatniej kolumny tej macierzy.

Dla naszego przykładu mamy:

	1	2	3	4	5
1	1,1	3,1			
2	2,2			3,2	
3			3,3	1,3	2,1
4		1,2	2,3		

Niechaj teraz dany będzie funkcjonał H :

$$H : \mathcal{T}_{M,D}^{\beta, \alpha} \longrightarrow \mathbb{R}^+$$

gdzie α oraz β są dowolnymi funkcjami sterującymi natomiast M' i D w szczególnym przypadku mogą być odpowiednio M oraz K . Zakłada się przy tym, że dla każdego p, l, r dana jest wartość funkcjonału równa $H(T_{plr})$.

Niech $H(\mathcal{T}_{M,D}^{\beta, \alpha})$ oznacza wartość funkcjonału H na zbiorze operatorów sterowanych funkcjami α i β . Jak już powiedzieliśmy funkcja β rozбивa zbiór $\mathcal{T}_{M,D}$ na warstwy operatorów. Niech Z_i oraz Z_j będą dwoma różnymi warstwami, zaś $u-c$ oraz u będą argumentami funkcji α i to takimi, że jeśli operator $\alpha(u-c)$ działa na p -ty element zbioru S to operator $\alpha(u)$ działa również na ten element i jest to operator najbliższy^{1/}.

Oznaczmy jeszcze:

$H(Z_i)$ - wartość funkcjonału H dla i -tej warstwy,

$Z_i | \alpha_{M,D}(g)$ - część warstwy Z_i do operatora $\alpha_{M,D}(g)$ włącznie.

Wówczas:

$$H(T_{pli} \times \mathcal{T}_{M,D}^{\beta, \alpha}) := H(T_{pli}) + \max \left[H(Z_i), H(Z_j | \alpha_{M,D}(u-c)) \right]$$

gdzie:

$$T_{pl} := \alpha_{M,D}(u)$$

$$i := \beta_{pl}(1)$$

^{1/} inaczej mówiąc żaden z operatorów $\alpha(u-g)$ ($0 < g < c$) nie działa na p -ty element zbioru S .

$$\alpha_{M,D}(u-c) = T_{p1_1}$$

$$j := \beta_{p1_1}(1).$$

Wtedy:

$$H(\mathcal{T}_{MK}^{\beta}) := \max_i H(Z_i). \quad 1.4$$

Ponieważ rodzina \mathcal{T}_{MK} jest określona przez funkcję θ , po wprowadzeniu zaś funkcjonału H należałoby podać drugą tablicę różniącą się od θ tym, że w miejscu jedynek miałyby odpowiednie wartości $H(T_{p1_i})$ obie te tablice połączymy i tak zbudowaną tablicę nazwiemy $H(\theta)$, gdzie:

$$H(\theta(r, (p, l))) = H(T_{p1_r}).$$

Analogicznie, jeśli dana jest funkcja sterująca α oraz β zbudować można tablicę $H(\theta^{\beta})$.

Wartość funkcjonału dla całego zbioru \mathcal{T}_{MK} zależy od struktury działania tych operatorów, a struktura ta jest określona funkcją \emptyset , dla wyznaczenia zatem wartości $H(\mathcal{T}_{MK}^{\beta})$ posługiwać się będziemy "macierzą" $H(\emptyset)$, gdzie:

$$H(\emptyset(i, j)) := H(Z_i | \alpha_{MK}(c))$$

przy czym:

$$\alpha_{MK}(c) := T_{p1} : \emptyset(i, j) = (p, l).$$

Przykład 6

$$S = \{s_1, s_2, s_3, s_4\} \quad W = \{w_1, w_2, w_3, w_4\}$$

$$S \subset C^5, \quad W \subset C^5$$

$$N = \{1, 2, 3\}$$

Dany jest funkcjonal H , który zapiszemy w postaci tablicy $H(\theta)$.

	11	12	13	14	15	21	22	23	24	25	31	32	33	34	35	41	42	43	44	45	
$H(\theta) =$	1	10		6	14	16	14	8				12	4	8	6	8	6	4		12	4
	2		4	8	6	12	6	4				14	10		4		4	10	8	6	10
	3	10					12	16	14	6	4		8	6	12	10	12	14		8	4

Rozpatrzmy teraz dwie funkcje $\theta_{\alpha_1}^{\beta_1}$ i $\theta_{\alpha_2}^{\beta_2}$:

	13	31	15	22	11	23	24	21	35	32	34	44	42	43	41	45	14	12	33	25
$H(\theta_{\alpha_1}^{\beta_1}) =$	1	6	12							4			4			4	14			
	2			12	4			6			4			8	4			4		
	3				10	14	6		10			8							6	4

	13	22	44	15	31	23	35	42	11	32	34	24	43	21	41	45	12	33	25	14
$H(\theta_{\alpha_2}^{\beta_2}) =$	1	6			16			8		4				14				8		
	2		4			14			10		4		8		4		4			6
	3			8			14			10		6				4				4

Struktura działań operatorów w obu przypadkach jest następująca:

$$\phi_1 =$$

	1	2	3	4	5	6	7	8	9	10	11	12	13
1	1,3	3,1					5,2	4,2			4,5	1,4	
2		1,5	2,2			2,1		3,4	4,3	4,1			1,2
3			1,1	2,3	2,4	3,5	4,4		3,3	2,5			

$$\phi_2 =$$

	1	2	3	4	5	6	7	8	9
1	1,3	1,5	3,5	3,2	2,1	3,3			
2	2,2	3,1	4,2		3,4	4,3	4,1	1,2	1,4
3	4,4	2,3	1,1	2,4				4,5	2,5

Wyznamy teraz $H(\phi_1)$ oraz $H(\phi_2)$:

$$H(\phi_1) =$$

	1	2	3	4	5	6	7	8	9	10	11	12	13
1	6	18					62	70			86	100	
2		18	22			54		66	78	82			104
3			28	42	48	58	66		72	76			

$$H(\phi_2) =$$

	1	2	3	4	5	6	7	8	9
1	6	22	30	34	52	60			
2	4	18	28		38	46	50	54	60
3	8	22	32	38				54	58

Tak wienc $H(J_{MK}^{\beta_1}) = 104$

$H(J_{MK}^{\beta_2}) = 60.$

1.4.2. Sformułowanie zadania programowania sekwencyjnego i jego szczególne przypadki

Jak widać z powyższego przykładu, wartość funkcjonału H na zbiorze \mathcal{T}_{MK} zależy od struktury działania operatorów należących do tego zbioru, a ta z kolei zależy od wartości funkcji sterujących α i β .

Wobec tego w naturalny sposób powstaje zadanie, które sformułujemy jak następuje:

Dany jest zbiór S oraz zbiór W , przy czym: $S \subset C^k$, $W \subset C^k$ $\text{card}(S) = \text{card}(W)$. Dana jest rodzina operatorów \mathcal{T}_{MK} oraz funkcjonał H .

Wyznaczyć funkcje sterujące α i β tak, by wartość funkcjonału H na zbiorze \mathcal{T}_{MK} sterowanym funkcjami α i β była minimalna.

Zadanie takie nazywać będziemy deterministycznym zadaniem sekwencyjnym. Funkcje α i β realizujące minimum funkcjonału H na zbiorze \mathcal{T}_{MK} nazwiemy optymalnymi.

Zadanie to jest zadaniem kombinatorycznym, zbiór rozwiązań spośród których należy wybrać rozwiązanie optymalne jest zbiorem skończonym. Wydawać się zatem może, że w celu znalezienia rozwiązania optymalnego należy sprawdzić wszystkie rozwiązania i wybrać te, które realizuje minimum. Jednak praktyczna przydatność tej metody jest znikoma, gdyż liczność zbioru rozwiązań, które trzeba byłoby przebadać jest bardzo duża i wynosi:

$$(m \cdot k)! \cdot \prod_{p,l} n_{pl}$$

gdzie $\prod_{p,1} n_{p1}$ jest iloczynem wszystkich n_{p1}
(przypomnijmy, że $n_{p1} = \text{card}(\mathcal{J}_{p1})$).

Dla naszego przykładu liczba możliwych rozwiązań jest równa $20! \cdot 2^7 \cdot 3^8$, jest to więc liczba olbrzymia mimo tego, iż samo zadanie jest niedużych rozmiarów. W świetle tego celowym wydaje się poszukiwanie metod wyznaczania optymalnego rozwiązania takiego zadania.

Zdaniem autora tak postawione zadanie jest najogólniejszą postacią zadania sekwencyjnego i nigdzie w dostępnej autorowi literaturze nie było formułowane. Znane są natomiast dwa inne zadania, które są szczególnymi przypadkami zbudowanego modelu. Wypiszmy jeszcze raz zadanie ogólne oraz jego szczególne przypadki.

Dla wszystkich zadań mamy następujące dane:

zbiory S oraz W , rodzina operatorów \mathcal{J}_{MK} ,
funkcjonał H .

Zadanie A (ogólne)

Wyznaczyć funkcje sterujące α i β tak, by wartość funkcjonału na zbiorze \mathcal{J}_{MK} sterowanym funkcjami α i β była minimalna.

Liczba wszystkich rozwiązań tego zadania jest równa:

$$(m \cdot k)! \cdot \prod_{p,1} n_{p1}$$

Zadanie B

Dla każdego $p \in M$ oraz każdego $l \in K$, $\text{card}(\mathcal{J}_{pl}) = 1$.
Wyznaczyć optymalną funkcję α .

W zadaniu tego typu mamy zamiast rodziny \mathcal{T}_{MK} zbiór operatorów T_{plr} a liczba rozwiązań jest tutaj równa:

$$(m \cdot k)!$$

Zadanie C

Dla każdego $p \in M$ dana jest funkcja sterująca α_{pK} .
Wyznaczyć optymalne funkcje sterujące α_{MK} i β ,
funkcja α_{MK} nie może naruszać wartości żadnej
z funkcji α_{pK} .

Wyjaśnijmy, co oznacza, że funkcja α_{MK} nie narusza wartości żadnej z funkcji α_{pK}

$$\forall p \in M \left[\alpha_{pK}^{-1}(\mathcal{T}_{pl_1}) > \alpha_{pK}^{-1}(\mathcal{T}_{pl_2}) \Rightarrow \alpha_{MK}^{-1}(\mathcal{T}_{pl_1}) > \alpha_{MK}^{-1}(\mathcal{T}_{pl_2}) \right]$$

Wszystkich rozwiązań zadania C jest:

$$\frac{(m \cdot k)!}{k!^m} \prod_{p,l} n_{pl}$$

Zadanie D

Dla każdego $p \in M$ dana jest funkcja sterująca α_{pK} ,
dla każdego $l \in K$ $\text{card}(\mathcal{T}_{pl}) = 1$.

Wyznaczyć optymalną funkcję sterującą α_{MK} i to taką,
by nie naruszała wartości żadnej funkcji α_{pK} .

Wszystkich rozwiązań jest w tym przypadku:

$$\frac{(m \cdot k)!}{k!^m} \cdot$$

Zadanie E

Dla każdego $p \in M$ dana jest funkcja sterująca α_{pK} przy czym:

$$\forall c \in B \wedge \forall l \in K \quad \alpha_{p_1 l}(c) = \alpha_{p_2 l}(c)$$

i ponadto:

$$\forall p_1, p_2 \in M \wedge \forall l \in K \quad \omega(p_1, l) = \omega(p_2, l)$$

i jest to zbiór jednoelementowy, gdzie:

$$B = \{1, 2, \dots, b\}, \quad b = \text{card} (\mathcal{T}_{pK}).$$

Wyznaczyć optymalną funkcję α i to taką, by nie naruszała wartości żadnej z funkcji α_{pK} .

Wszystkich rozwiązań jest w tym przypadku:

$$m!$$

W literaturze spotyka się zadanie typu D oraz E.

1.4.3. O możliwości rozbicia zadania wyjściowego na zadania mniejszych rozmiarów

Poprzednio zakładaliśmy, że zbiór S jest zbiorem elementów zależnych ze względu na warstwy operatorów.

Założmy teraz, że mamy daną funkcję ω , która jak już powiedziano rozbija zbiór (rodzinę) operatorów na warstwy, oznaczmy zbiór tych warstw przez Z .

Powstają teraz trzy pytania:

1^o Jak sprawdzić, czy zbiór S jest zbiorem elementów zależnych ze względu na warstwy?

2^o Jeśli zbiór S nie jest zbiorem elementów zależnych ze względu na warstwy, to jak wyznaczyć podzbiory, w których wszystkie elementy byłyby zależne?

3^o Co wynika z faktu, że zbiór S zawiera te podzbiory?

Na pytania te chcemy teraz dać odpowiedź. Przyjmijmy jeszcze następującą definicję:

Definicja 1.9

Dwa podzbiory S' i S'' zbioru S nazywać będziemy niezależnymi ze względu na warstwy operatorów jeżeli żaden element zbioru S' nie jest zależny ze względu na warstwy z żadnym elementem zbioru S'' ^{1/}.

Mamy zatem dane: zbiór S , rodzinę \mathcal{J}_{MK} , zbiór N oraz funkcję ω , a tym samym mamy również daną funkcję θ . Przypomnijmy jeszcze, że zapis $Z_i(s_p)$ oznacza, że i -ta warstwa (potencjalna lub nie) działa na p -ty element zbioru S . Niech:

$$R := \left\{ (i, p) \in N \times M : Z_i(s_p) \right\}.$$

Zbiór R tworzy się wykorzystując funkcję θ w sposób następujący:

$$(i, p) \in R \Leftrightarrow \exists l \in K : \theta(i, (p, l)) = 1.$$

Oznaczmy:

$$\Delta Z_i := \{ s_p \in S : (i, p) \in R \}$$

$$\Delta^{-1} s_p := \{ z_i \in Z : (i, p) \in R \}.$$

^{1/} patrz definicja 1.7

Jest oczywistym, że:

$$\begin{aligned} \bigcup_{i \in N} Z_i &= S \\ \bigcup_{p \in M} \Delta^{-1} s_p &= Z \end{aligned} \quad 1.5$$

Pierwsza z powyższych równości oznacza, że na każdy element zbioru S działa przynajmniej jedna warstwa, a jej prawdziwość wynika z istnienia dla każdego $p \in M$ operatora T_{p1} .

Natomiast druga równość oznacza, że każda warstwa operatorów działa przynajmniej na jeden element zbioru S . Prawdziwość tej równości wynika bezpośrednio z definicji funkcji ω .

Zbiór R można zawsze rozbić na dwa podzbiory R' i R'' takie, że:

$$\begin{aligned} R' \cup R'' &= R \\ R' \cap R'' &= \emptyset \end{aligned} \quad 1.6$$

gdzie:

$$\begin{aligned} R' &\subset N' \times M' \\ R'' &\subset N'' \times M'' \end{aligned}$$

gdzie wobec 1.5 i 1.6

$$\begin{aligned} N' \cup N'' &= N \\ M' \cup M'' &= M \end{aligned}$$

Twierdzenie 1.3

Zbiór S daje się rozbić na dwa podzbiory S' i S'' niezależne ze względu na warstwy wtedy i tylko wtedy, gdy zbiór R daje się rozbić na dwa niezależne w sensie definicji 1.1 podzbiory R' i R'' , przy czym:

$$S' = \{s_p \in S : p \in M'\}$$
$$S'' = \{s_p \in S : p \in M''\}.$$

Dowód.

Założmy, że zbiór R rozбивa się na dwa niezależne podzbiory, tzn.:

$$(i_1, p_1) \in R' \wedge (i_1, p_2) \in R \wedge (i_2, p_1) \in R \Rightarrow$$
$$\Rightarrow (i_1, p_2) \in R' \wedge (i_2, p_1) \in R' \quad 1.7$$

W ten sposób rozbitcie zbioru R generuje rozbitcie zbioru S na dwa podzbiory S' i S'' przy czym:

$$S' = \{s_p \in S : p \in M'\}$$
$$S'' = \{s_p \in S : p \in M''\},$$

pokażemy, że podzbiory te są niezależne ze względu na warstwy. Ponieważ $p_i \in M'$ to $s_{p_i} \in S'$. Z 1.7 wynika, że do S' należeć będzie każdy element zbioru S , na który działa warstwa Z_{i_1} . Jeżeli ponadto warstwa Z_{i_1} działa na s_{p_2} i warstwa Z_{i_2} działa na s_{p_2} to na mocy 1.7 do S' zostaną zaliczone wszystkie elementy zbioru S , na które działa warstwa Z_{i_1} . W ten sposób poza zbiorem S' pozostaną tylko te elementy zbioru S , które nie są zależne ze względu na warstwy z żadnym elementem zbioru S' . Co dowodzi niezależności zbiorów S' i S'' ze względu na warstwy.

Niech teraz S' i S'' będą dwoma niezależnymi ze względu na warstwy operatorów podzbiorymi zbioru S , przy czym $S' \cup S'' = S$. Rozbitcie to generuje rozbitcie R na dwa podzbiory R' i R'' .

Oznaczmy:

$$R' \subset N' \times M'$$

$$R'' \subset N'' \times M''$$

gdzie:

$$M' = \{ p \in M : s_p \in S' \}$$

$$M'' = \{ p \in M : s_p \in S'' \}$$

$$N' = \{ i \in N : (i, p) \in R \wedge s_p \in S' \}$$

$$N'' = \{ i \in N : (i, p) \in R \wedge s_p \in S'' \}$$

ponieważ S' i S'' są niezależne ze względu na warstwy to:

$$S' \cap S'' = \emptyset$$

a zatem:

$$M' \cap M'' = \emptyset$$

Należy pokazać, że: $N' \cap N'' = \emptyset$. Załóżmy, że tak nie jest, tzn. istnieje takie i , że $N' \cap N'' = i$, co znaczyłoby, iż warstwa Z_i działałaby na element należący do S' oraz na element należący do S'' , co przeczy założeniu o niezależności zbiorów S' i S'' ze względu na warstwy. Zatem $N' \cap N'' = \emptyset$.

Z lematu 1.1 wynika, że R' i R'' są niezależne w sensie definicji 1.1. Co kończy dowód.

Ponieważ rozbicie zbioru R na niezależne podzbiory implikuje rozbicie zbioru S na podzbiory niezależne i podzbiory R' i R'' jednoznacznie wyznaczają podzbiory S' i S'' , dla wyznaczenia (jeśli takie istnieją) tych podzbiorów można stosować algorytm, który opisany został na stronie 23. Algorytm ten rzecz jasna stosować trzeba dla zbioru R .

Niech α_1 będzie dowolną funkcją sterującą na rodzinie operatorów \mathcal{T}_{MK} działających na S . Niech S' i S'' będą podzbiorami zbioru S i to takimi, że:

$$S' \cup S'' = S.$$

Funkcja α_1 ustala pewien porządek składania operatorów działających na S' oraz pewien porządek składania operatorów działających na S'' . Załóżmy teraz, że mamy inną funkcję sterującą α_2 ale taką, że porządek składania operatorów działających na S' jest taki sam jak porządek ustalony przez α_1 i tak samo z kolejnością składania operatorów działających na S'' , tzn.:

$$\forall (p_1 \in M' \wedge p_2 \in M')$$

$$\alpha_1^{-1}(\mathcal{T}_{p_1 l_1}) > \alpha_1^{-1}(\mathcal{T}_{p_2 l_2}) \Rightarrow$$

$$\Rightarrow \alpha_2^{-1}(\mathcal{T}_{p_1 l_1}) > \alpha_2^{-1}(\mathcal{T}_{p_2 l_2})$$

$$\forall (p_3 \in M'' \wedge p_4 \in M'')$$

$$\alpha_1^{-1}(\mathcal{T}_{p_3 l_1}) > \alpha_1^{-1}(\mathcal{T}_{p_4 l_2}) \Rightarrow$$

$$\Rightarrow \alpha_2^{-1}(\mathcal{T}_{p_3 l_1}) > \alpha_2^{-1}(\mathcal{T}_{p_4 l_2})$$

gdzie $\mathcal{T}_{M',K}$ i $\mathcal{T}_{M'',K}$ to rodziny operatorów działających odpowiednio na S' i S'' natomiast l_1 i l_2 to dowolne ele-

menty zbioru K . Przy powyższych oznaczeniach udowodnimy następujące twierdzenie:

Twierdzenie 1.4

Jeżeli zbiory S' i S'' są niezależne ze względu na warstwy to dla dowolnej funkcji β

$$H(\mathcal{J}_{MK}^{\beta}_{\alpha_1}) = H(\mathcal{J}_{MK}^{\beta}_{\alpha_2})$$

Dowód.

Zauważmy, że rozbięcie zbioru S na dwa podzbiory S' i S'' implikuje rozbięcie rodziny \mathcal{J}_{MK} na dwie podrodziny $\mathcal{J}_{M'K}$ i $\mathcal{J}_{M''K}$ działające odpowiednio na S' i na S'' , gdzie:

$$\begin{aligned} M' &= \{p \in M : s_p \in S'\} \\ M'' &= \{p \in M : s_p \in S''\}. \end{aligned}$$

Jeśli zbiory S' i S'' są niezależne ze względu na warstwy operatorów, funkcja sterująca β dla zbioru $\mathcal{J}_{M'K}$ wybierac będzie elementy podzbioru N' zbioru N , zaś dla zbioru $\mathcal{J}_{M''K}$ podzbioru $N-N'$.

Na mocy założeń dotyczących działania operatorów żadna para operatorów, z których jeden należy do $\mathcal{J}_{M'K}$ a drugi do $\mathcal{J}_{M''K}$ nie działają kolejno (ponieważ nie należą do jednej warstwy). Z drugiej strony w zbiorze warstw Z istnieje podzbiór Z' taki, że do niego należą wszystkie operatory $\mathcal{J}_{M'K}$ i podzbiór Z'' do którego należą wszystkie operatory $\mathcal{J}_{M''K}$, przy czym:

$$Z' = \{ Z_i \in Z : i \in N' \}$$

$$Z'' = \{ Z_i \in Z : i \in N - N' \}.$$

Z założenia twierdzenia funkcja α_2 nie narusza kolejności ustalonej przez funkcję α_1 na zbiorze $\mathcal{T}_{M',K}$ i $\mathcal{T}_{M''K}$, a zatem struktura działania operatorów $\mathcal{T}_{M',K}$ i $\mathcal{T}_{M''K}$ dla funkcji α_2 jest taka sama jak dla funkcji α_1 , czyli:

$$\vartheta_1 = \vartheta_2$$

a stąd

$$H(\vartheta_1) = H(\vartheta_2). \text{ Co kończy dowód.}$$

Twierdzenie 1.5

Jeżeli podzbiory S' i S'' zbioru S są niezależne ze względu na warstwy to dla dowolnej funkcji α i β istnieją funkcje α', β' oraz α'', β'' i dla funkcji α', β' oraz α'', β'' istnieją funkcje α, β takie, że:

$$H(\mathcal{T}_{MK_\alpha}^\beta) = \max \left[H(\mathcal{T}_{M',K_{\alpha'}}^{\beta'}), H(\mathcal{T}_{M''K_{\alpha''}}^{\beta''}) \right]$$

Dowód.

Przy dowodzie powyższego twierdzenia skorzystamy z wzoru 2.4. Mamy:

$$H(\mathcal{T}_{MK_\alpha}^\beta) = \max_i H(Z_i) = \max \left[\max_i H(Z'_i), \max_j H(Z''_j) \right],$$

oraz dla strony prawej:

$$\max \left[H(\mathcal{T}_{M',K_{\alpha'}}^{\beta'}), H(\mathcal{T}_{M''K_{\alpha''}}^{\beta''}) \right] = \max \left[\max_i H(Z'_i), \max_j H(Z''_j) \right].$$

Rozbicie zbioru S na dwa podzbiory pociąga za sobą roz-
bicie rodziny \mathcal{T}_{MK} na dwie podrodziny $\mathcal{T}_{M',K}$ i $\mathcal{T}_{M'',K}$
oraz rozbicie zbioru warstw na dwa podzbiory Z' i Z'' .

Z niezależności zbiorów S' i S'' wynika, że żaden operator
należący do \mathcal{T}_{MK} nie należy do $\mathcal{T}_{M'',K}$ i żaden operator
działający w jednej z warstw należących do Z' nie działa
w żadnej z warstw należących do Z'' . Wystarczy zatem dobrać
funkcję α taką by nie naruszała kolejności ustalonej przez
 α' i kolejności ustalonej przez α'' oraz funkcję β taką, że:

$$\beta_{pl}(1) = \begin{cases} \beta'_{pl}(1) & \text{dla } p \in M' \\ \beta''_{pl}(1) & \text{dla } p \in M'' \end{cases}$$

Wykorzystując twierdzenie 2.4 otrzymujemy żadaną w tezie
równość, co kończy dowód.

Twierdzenie 1.6

Jeżeli S' i S'' są podzbiorem zbioru S niezależ-
nymi ze względu na warstwy, funkcje α' , β' oraz
 α'' , β'' są optymalne na podrodzinach $\mathcal{T}_{M',K}$ i
 $\mathcal{T}_{M'',K}$ to funkcje α i β dla których:

$$H(\mathcal{T}_{MK}^{\beta}) = \max \left[H(\mathcal{T}_{M',K}^{\beta'}), H(\mathcal{T}_{M'',K}^{\beta''}) \right] \quad 1.8$$

są optymalne dla rodziny \mathcal{T}_{MK} .

Dowód.

Założmy, że α i β nie są optymalne. Niech zatem α^* , β^* będą optymalnymi funkcjami sterującymi dla rodziny \mathcal{T}_{MK} , co oznaczałoby, że:

$$H(\mathcal{T}_{MK}^{\beta}) > H(\mathcal{T}_{MK}^{\beta^*}) \quad 1.9$$

Korzystając z twierdzenia 1.5 można skonstruować funkcje $\alpha^{*'}, \beta^{*'}$ oraz $\alpha^{*''}, \beta^{*''}$ takie, że:

$$H(\mathcal{T}_{MK}^{\beta^*}) = \max \left[H(\mathcal{T}_{M'K}^{\beta^{*'}}), H(\mathcal{T}_{M''K}^{\beta^{*''}}) \right] \quad 1.10$$

Porównując 1.8 i 1.9 dostajemy, że:

$$\max \left[H(\mathcal{T}_{M'K}^{\beta'}), H(\mathcal{T}_{M''K}^{\beta''}) \right] > \max \left[H(\mathcal{T}_{M'K}^{\beta^{*'}}), H(\mathcal{T}_{M''K}^{\beta^{*''}}) \right]$$

co wobec niezależności zbiorów S' i S'' oraz twierdzenia 1.4 przeczy optymalności funkcji α', β' oraz α'', β'' i kończy dowód. Z udowodnionego twierdzenia wynika bardzo pożyteczny dla praktyki wniosek.

Wniosek.

Jeśli zbiór S rozpada się na dwa podzbiory niezależne ze względu na warstwy to zamiast poszukiwać optymalnych funkcji sterujących na rodzinie \mathcal{T}_{MK} można wyznaczać optymalne funkcje sterujące na podrodzinach odpowiadających zbiorom S' i S'' co pozwala na redukcję rozmiarów zadania.

1.4.4. Zachowanie się wartości funkcji kryterium gdy S' i S'' nie są podzbiorami niezależnymi ze względu na warstwy

W sytuacji gdy zbiór S rozбивa się na dwa podzbiory niezależne ze względu na warstwy można było zadanie rozbić na dwa, rozwiązać je i następnie łącząc te rozwiązania, otrzymać rozwiązanie zadania wyjściowego. Powstaje pytanie: czy postępowanie takie jest możliwe i równie efektywne jeśli pominie się założenie o niezależności?

Zanim odpowiadamy na to pytanie zwróćmy uwagę na następujący fakt: Całe zadanie opisane jest przy pomocy macierzy θ , a właściwie $H(\theta)$. Jeśli teraz S' i S'' są podzbiórami zbioru S to z macierzy θ wydzielić można dwie podmacierze θ_1 i θ_2 , które odpowiadają podrodzinie operatorów działających odpowiednio na S' i na S'' . Można to przedstawić schematycznie:

$$\theta = \begin{bmatrix} \theta_1 & \theta_2 \end{bmatrix}$$

Jeśli teraz S' i S'' są to podzbiory niezależne ze względu na warstwy to w zbiorze warstw istnieje podzbiór Z' , taki że żadna warstwa należąca do tego podzbioru nie działa na żaden element zbioru S'' . W takiej sytuacji, po przestawieniu wierszy macierzy θ otrzymamy macierz o następującej budowie:

$$\theta = \begin{bmatrix} \theta_1 & 0 \\ 0 & \theta_2 \end{bmatrix}$$

Niech teraz S_1 i S_2 są podzbiorami zbioru S i to takimi, że:

$$S_1 \cup S_2 = S$$

$$S_1 \cap S_2 = \emptyset.$$

Wówczas rodzina operatorów \mathcal{T}_{MK} rozbiła się na dwie podrodziny \mathcal{T}_{M_1K} i \mathcal{T}_{M_2K} działające odpowiednio na S_1 i S_2 .

Niech α^1 i α^2 będą funkcjami sterującymi odpowiednio na \mathcal{T}_{M_1K} i \mathcal{T}_{M_2K} .

Oznaczmy:

$$\text{card}(\mathcal{T}_{M_1K}) = m_1$$

$$\text{card}(\mathcal{T}_{M_2K}) = m_2$$

Niech α i β będą funkcjami sterującymi na rodzinie \mathcal{T}_{MK} takimi, że:

$$\alpha(c) = \begin{cases} \alpha^1(c) & \text{dla } c=1, 2, \dots, m_1 \\ \alpha^2(c) & \text{dla } c=m_1+1, \dots, m_1+m_2 \end{cases}$$

$$\beta_{pl}(1) = \begin{cases} \beta_{p1}^1(1) & \text{dla } p \in M_1 \\ \beta_{p1}^2(1) & \text{dla } p \in M_2 \end{cases}$$

Lemat 1.2

$$H(\mathcal{T}_{MK}^\beta_\alpha) \leq H(\mathcal{T}_{M_1K}^{\beta^1}_{\alpha^1}) + H(\mathcal{T}_{M_2K}^{\beta^2}_{\alpha^2})$$

Dowód.

Prawdziwość lematu jest oczywista jeśli tylko zbuduje się funkcję ϑ_1 odpowiadającą sterowaniu funkcjami α^1, β^1 na rodzinie \mathcal{T}_{M_1K} oraz funkcję ϑ_2 odpowiadającą sterowaniu funkcjami α^2, β^2 na rodzinie \mathcal{T}_{M_2K} . Przedstawmy to schematycznie:

$$\begin{array}{cc} \vartheta_1 & \vartheta_2 \\ \mathcal{T}_{M_1K} & \mathcal{T}_{M_2K} \end{array}$$

Żaden z operatorów należących do \mathcal{T}_{M_2K} nie może działać współbieżnie z żadnym z operatorów należących do \mathcal{T}_{M_1K} .

Jeśli natomiast wyznaczymy funkcję ϑ odpowiadającą sterowaniu funkcjami α, β na całej rodzinie \mathcal{T}_{MK} to może się zdarzyć, że któryś z operatorów z \mathcal{T}_{M_2K} działa współbieżnie z któryś z operatorów należących do \mathcal{T}_{M_1K} . Mówiąc inaczej tablica ϑ nie może mieć więcej kolumn niż tablica ϑ_1 i ϑ_2 razem. Oznacza to, że nierówność:

$$H(\mathcal{T}_{MK}^{\beta}) > H(\mathcal{T}_{M_1K}^{\beta^1}) + H(\mathcal{T}_{M_2K}^{\beta^2})$$

nie jest możliwa. Co kończy dowód lematu.

Twierdzenie 1.7

Jeśli $\alpha^1, \beta^1, \alpha^2, \beta^2$ są optymalnymi funkcjami sterującymi na podzbiorach \mathcal{T}_{M_1K} i \mathcal{T}_{M_2K} a α, β są optymalne dla całej rodziny \mathcal{T}_{MK} to:

$$H(\mathcal{T}_{MK}^{\beta}) \leq H(\mathcal{T}_{M_1K}^{\beta^1}) + H(\mathcal{T}_{M_2K}^{\beta^2})$$

Dowód.

Dowód poprowadzimy niewprost. Załóżmy w tym celu, że prawdziwa jest nierówność

$$H(\mathcal{J}_{MK}^{\beta}) > H(\mathcal{J}_{M_1 K}^{\beta^1}) + H(\mathcal{J}_{M_2 K}^{\beta^2})$$

Biorąc wówczas funkcje α^* , β^* takie jak w lemacie 1.2 otrzymalibyśmy:

$$H(\mathcal{J}_{MK}^{\beta}) > H(\mathcal{J}_{MK}^{\beta^*})$$

Przeżyłoby to optymalności funkcji α i β oraz kończy dowód twierdzenia.

Z udowodnionego powyżej twierdzenia wynika następujący wniosek.

Wniosek.

Jeżeli zbiór S nie rozbija się na dwa podzbiory niezależne ze względu na warstwy, to rozbijanie zadania programowania sekwencyjnego na dwa zadania mniejszych rozmiarów i łączenie rozwiązań tych zadań w jedno, prowadzi przeważnie do straty na wartości funkcjonału H .

2. PRZYKŁADY ZADAN PROGRAMOWANIA SEKWENCYJNEGO

2.1. Uwagi wstępne

W rozdziale poprzednim zostało sformułowane zadanie nazwane zadaniem programowania sekwencyjnego. Zadanie to opisane zostało w sposób formalny bez podawania interpretacji przyjętych założeń, ani też nic nie mówiło się o możliwościach zastosowania tego typu zadań. Wszystko to chcemy zaprezentować w tym rozdziale. Mówiąc inaczej w rozdziale drugim:

- podano przykłady zastosowań zadania programowania sekwencyjnego,
- uzasadnia się celowość przyjętych założeń przy konstruowaniu formalnego modelu zadania.

W rozdziale tym podajemy trzy przykłady zastosowań zadania programowania sekwencyjnego. Najpełniej a niekiedy z drobiazgową dokładnością omawia się zadanie z dziedziny organizacji produkcji. Wynika to z dwóch powodów. Po pierwsze, zadanie to zostało najwcześniej sformułowane spośród wszystkich zadań dających się zaliczyć do zadań programowania sekwencyjnego. Po drugie, w literaturze poświęconej zadaniom nazwanym w tej pracy zadaniami programowania sekwencyjnego zadanie to zajmuje najwięcej miejsca. Powtórzmy raz jeszcze, na przykładzie tego zadania pokazuje się szereg konsekwencji wynikających z przyjętych w modelu założeń oraz uzasadnia się celowość ich przyjęcia. Drugi przykład dotyczy zagadnienia budowania harmonogramów zajęć. Temat ten jest obszerny

sam w sobie i może być przedmiotem rozprawy o objętości nie mniejszej niż niniejsza, co zmusiło autora do przedstawienia tego zadania w bardzo dużym skrócie.

Formułując to zadanie w najprostszym przypadku okazuje się, że formalnie jest to zadanie programowania sekwencyjnego. Ostatnim z cytowanych zastosowań jest przykład planowania usług. Podobnie jak i poprzednio pokazuje się, że zadanie to jest rzeczywiście zadaniem programowania sekwencyjnego.

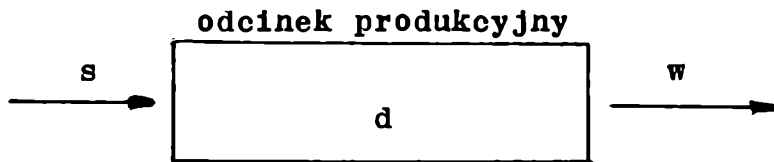
2.2. U s t a l a n i e h a r m o n o g r a m ó w p r o - d u k c j i

Z zadaniami tego typu spotyka się w przemyśle, gdzie idzie o to, by dla danego odcinka produkcyjnego ustalić taki plan pracy stanowisk wchodzących w skład tego odcinka, by przy istniejących warunkach żądane zadania produkcyjne wykonać w możliwie najkrótszym czasie. Przez warunki rozumie się tutaj urządzenia służące do obróbki, ludzi te urządzenia obsługujących oraz technologię.

Zadania rozumie się tutaj jako ilość i rodzaj produkowanych przedmiotów, które w dalszym ciągu nazywać będziemy detalami. Plan pracy (harmonogram) ma wskazywać dla każdego stanowiska, który z przedmiotów ma być na nim obrabiany, w jakiej kolejności, a w przypadku stanowisk wieloczynnościowych, którą z możliwych operacji stanowisko to ma wykonywać.

W dalszej części nazywać będziemy przedmiot, który jest poddawany obróbce detalem. Detal po opuszczeniu badanego odcinka produkcyjnego staje się wyrobem, zaś przed wejściem

na dany odcinek jest surowcem. Przedstawić to można graficznie jak na poniższym schemacie:



gdzie:

s - surowiec

d - detal

w - wyrób

Ponieważ interesować nas będą w dalszym ciągu takie procesy gdzie jest jednoznaczne przyporządkowanie "surowiec-wyrób"

tzn. nie ma montażu ani też podziału w trakcie procesu, będziemy używać zamiennie terminów: wyrób i detal.

W interesującym nas odcinku produkcyjnym odbywa się proces produkcyjny wyrobu. Proces ten rozumie się tutaj jako:

"zbiór operacji produkcyjnych realizowanych w uporządkowanej kolejności w celu wytworzenia określonego wyrobu przez przekształcenie (kolejne zmiany stanów) tworzywa"^{1/}.

Operacja ta: "część procesu produkcyjnego wyrobu wykonywana na określonym przedmiocie przez jednego pracownika lub grupę bez przerw i przebrojeń na jednym stanowisku roboczym"^{2/}. Natomiast stanowisko robocze rozumie się tutaj

jako: "elementarny składnik struktury produkcyjnej kojarzący w sobie trzy podstawowe czynniki procesu pracy: środki pracy, przedmiot pracy i samą pracę człowieka"^{3/}.

^{1/} S, Chajtman "Podstawy organizacji procesu produkcyjnego" s. 95

^{2/} tamże str. 96

^{3/} tamże str. 133-134

Tak więc dysponujemy dla danego odcinka produkcyjnego następującymi danymi:

Zbiór detali (wyróbów) - W

Zbiór stanowisk roboczych - N

Zbiór operacji - K .

Zakłada się, że:

$$\text{card}(N) \leq \text{card}(K)$$

co odpowiada sytuacji, że każde stanowisko może wykonać co najmniej jedną operację. Jeśli nierówność jest ostra to wśród stanowisk są stanowiska wieloczynnościowe, tzn. takie, które mogą wykonywać kilka operacji^{1/}.

Definicja 2.1

Każdy uporządkowany ciąg $k_{i_1}, k_{i_2}, \dots, k_{i_s}$ operacji ze zbioru K , taki że w wyniku działania kolejnych operacji na detal otrzyma się żądany wyrób w nazywamy technologią tego wyrobu.

Definicja 2.2

Każdy uporządkowany ciąg $n_{i_1}, n_{i_2}, \dots, n_{i_k}$ elementów zbioru N taki, że w wyniku wykonania operacji na kolejnych maszynach surowiec staje się wyrobem nazywamy marszrutą technologiczną tego wyrobu.

^{1/} Jest rzeczą oczywistą, że nierówność ta nie może być traktowana jako kryterium rozstrzygające czy stanowiska są wieloczynnościowe, czy nie. Sprawa ta została omówiona w rozdziale poprzednim, gdzie mówiło się o funkcji ω oraz π .

Definicja 2.3

Czynności stanowiska roboczego nazywać będziemy uporządkowaną parą (w, l) , gdzie $w \in W, l \in K$ taką, że stanowisko to może wykonać operację l na detalu w .

Definicja 2.4

Uporządkowany ciąg czynności stanowiska roboczego nazywać będziemy harmonogramem tego stanowiska.

W tej chwili nie interesuje nas to czy harmonogram jest dopuszczalny czy nie, gdyż można zbudować taki harmonogram, który ze względu na technologię nie może być realizowany. Harmonogram taki nazwiemy niedopuszczalnym nie definiując w tej chwili tego terminu. Posłużymy się natomiast następującym przykładem. Załóżmy, że pewne stanowisko jest dwuczynnościowe i może wykonać operacje l_1 i l_2 . Harmonogram tego stanowiska zawiera ciąg czynności taki, że $\dots(w, l_1), (w, l_2) \dots$ podczas gdy technologia wyrobu w wymaga by operacja l_2 była wykonywana przed operacją l_1 . Rzecz jasna, harmonogram powyższy nie może być realizowany.

Definicja 2.5

Zbiór harmonogramów wszystkich stanowisk roboczych danego odcinka produkcyjnego nazywać będziemy harmonogramem tego odcinka.

Zadanie wyznaczenia optymalnego harmonogramu można teraz sformułować następująco:

Dany jest zbiór wyrobów W , które należy wykonać dysponując przy tym zbiorem surowców S . Dany jest ponadto zbiór N stanowisk roboczych oraz zbiór K operacji, które należy wykonać na surowcach ze zbioru S by w ich wyniku otrzymać zbiór wyrobów W . Dla każdej operacji, każdego stanowiska i każdego wyrobu dany jest czas jej trwania.

Zakłada się, że:

- a. żaden detal nie jest obrabiany na dwóch stanowiskach równocześnie,
- b. żadne stanowisko nie wykonuje dwóch czynności jednocześnie.

Należy wyznaczyć taki harmonogram by przy tych założeniach wykonać wszystkie wyroby ze zbioru W w możliwie najkrótszym czasie.

Tak postawione zadanie jest najogólniejszym sformułowaniem zagadnienia ustalania harmonogramów dla procesów produkcyjnych tego typu. Można bowiem obok założeń a i b wymaganych od harmonogramu w każdej konkretnej sytuacji sformułować i inne.

Wymieńmy przykładowo kilka z nich:

A.

- c. dla każdego detalu dana jest dokładnie jedna technologia,
- d. dla każdego detalu dana jest co najmniej jedna technologia przy czym dokładnie jedna ma być realizowana,
- c. wszystkie detale mają jednakową technologię,

B.

- f. wszystkie stanowiska są jednoczynnościowe,
- g. co najmniej jedno stanowisko jest wieloczynnościowe,
- h. nie ma dwóch stanowisk jednego typu,
- i. co najmniej dwa stanowiska są jednego typu.

Założenia grupy A dotyczą technologii wyrobów, natomiast założenia grupy B dotyczą stanowisk roboczych.

Po tym wprowadzeniu pokażemy związek modelu formalnego prezentowanego w rozdziale poprzednim z tak rozumianym procesem produkcyjnym.

Jak już powiedziano wcześniej, interesować nas będą tylko takie procesy w których z wybranej sztuki surowca otrzymuje się określony wyrób i to wiadomo jaki. Gotowy wyrób opisać można przy pomocy wielu cech, które to cechy jednoznacznie określają wyrób. Cechami tymi mogą być własności techniczne jak: ciężar, twardość, przekrój, grubość itp., jak również własności rodzajowe jak: kolor, typ przekroju, typ uzębienia, typ i rodzaj połączeń jeśli wyrób ma na innym odcinku produkcyjnym być łączony z innym, itp.

A zatem każdy wyrób tego rodzaju formalnie daje się opisać przy pomocy wektora:

$$w = (w_1, w_2, \dots, w_k)$$

gdzie:

w - wyrób

$$w_i \in C_i \quad i=1, 2, \dots, k$$

k - ilość opisujących dany wyrób cech

C_i - zbiór możliwych wartości jakie może przyjmować cecha i .

Równocześnie surowiec można przedstawić również w postaci ta-

kiego wektora:

$$s = (s_1, s_2, \dots, s_k)$$

gdzie analogicznie:

s - surowiec

$$s_i \in C_i$$

Ponieważ S jest zbiorem surowców, W zbiorem wyrobów to oczywistym jest, że po to by można było wykonać wszystkie wyroby należące do zbioru W , surowców musi być co najmniej tyle ile jest wyrobów. Stąd też postulat 1.1, tzn.

$$\text{card}(S) \succcurlyeq \text{card}(W).$$

Jeśli teraz ponumerujemy elementy zbioru S (funkcja ψ) i elementy zbioru W (funkcja φ) będziemy mogli formalnie rozróżniać od siebie poszczególne surowce jak i poszczególne wyroby.

Jest rzeczą oczywistą, że dla ustalonego surowca można wskazać wyroby, które w trakcie procesu produkcyjnego na danym odcinku z tego surowca można uzyskać. I odwrotnie, dla ustalonego wyrobu można wskazać surowiec (surowce), z którego ten wyrób można otrzymać. Prowadzi to do wypisania zbioru par postaci (s_p, w_g) , która to para oznacza, że z p -tego surowca otrzymać można s -ty wyrób. Taki jest sens zbioru Γ (patrz 1.2.).

Postulat 2^o w tej definicji wymaga by dla każdego wyrobu, przedstawionego w planie istniał co najmniej jeden surowiec z którego ten wyrób można uzyskać. Postulat ten nazwiemy

postulatem wykonalności zadań. Postulat 3⁰, który nazwiemy postulatem niesprzeczności zadań wymaga aby: jeśli z jednego surowca s_1 uzyskać można dwa wyroby (różne) w_1 oraz w_j ze zbioru W (przypomnijmy, że zbiór W jest zbiorem wyrobów, które muszą być wykonane) to zbiór S zawierać musi taki surowiec s_r , z którego wykonać można bądź wyrób w_1 bądź wyrób w_j . W przeciwnym bowiem razie jednego z tych wyrobów nie można byłoby wykonać, a więc zadania byłyby sprzeczne z bazą surowcową.

Jest rzeczą naturalną, że jeśli surowców jest więcej niż wyrobów i z każdego surowca można wykonać co najmniej jeden wyrób to rozsądek wymaga by wybrać surowce tak, by ze względu na pewne kryterium (koszty surowca, koszty wytwarzania) wybór ten był najlepszy i równocześnie zabezpieczał wykonanie zadań. Stąd właśnie sformułowane zadanie wyboru podzbioru zbioru S . W konsekwencji otrzymuje się zbiór S^* taki, że:

- 1⁰ dla każdego surowca ze zbioru S^* istnieje jeden i tylko jeden wyrób,
- 2⁰ dla każdego wyrobu ze zbioru W istnieje jeden i tylko jeden surowiec ze zbioru S^* .

Wobec tych dwóch własności zbioru S^* jego elementy przenumerować można tak jak są ponumerowane elementy zbioru W , tzn. s_p oznacza surowiec, z którego otrzymuje się wyrób w_p . Przyjmuje się w modelu, iż jedna operacja wykonana na detalu spowoduje to, iż detal ten będzie miał już żadaną dla uzyskania z niego wyrobu cechę. Nie zmniejsza to ogólności, gdyż wymiar wektora opisującego wyrób można zwiększyć wprowadzając fikcyjną cechę w_i należącą do zbioru cech fikcyjnych C_i^1 .

Odpowiednikiem operacji jest w naszym modelu formalnym operator T_{p1} i odpowiada on operacji 1-tej wykonywanej na detalu p-tym. Zbiór wszystkich operacji, które należy wykonać na detalu p-tym w naszym modelu odpowiada zbiorowi operatorów T_{pD} , gdzie $D \subset K$. Jeśli dla wyrobu p-tego jest obojętne w jakiej kolejności należy wykonywać zadane operacje aby otrzymać ten wyrób to każde złożenie T_{pD} jest technologią wyrobu p, zaś wybór technologii do realizacji jest uwarunkowany wyborem funkcji sterującej \mathcal{L}_{pD} . Jeśli natomiast technologia wyrobu p-tego jest dana z góry narzucona to technologię tę wprowadza się do modelu poprzez funkcję sterującą \mathcal{L}_{pD} . Złożenie T_{MK} należy interpretować jak następuje: w badanym odcinku produkcyjnym jest jedno tylko stanowisko robocze, ale takie, że może wykonywać wszystkie operacje jakie są niezbędne by ze zbioru surowców uzyskać zbiór wyrobów W . Tutaj, jeśli jest obojętne w jakiej kolejności wykonuje się operacje dla poszczególnych wyrobów każde złożenie T_{MK} jest harmonogramem dla tak rozumianego "stanowiska". Jeśli natomiast dla wyrobu p-tego zadana jest z góry technologia, to jak już powiedziano wyżej technologia ta jest wprowadzona do modelu poprzez funkcję sterującą \mathcal{L}_{pD} , a postulat nienaruszalności wartości funkcji \mathcal{L}_{pD} przez funkcję \mathcal{L}_{MD} oznacza, iż harmonogram dany funkcją \mathcal{L}_{MD} musi być realizowalny.

Po to aby przejść teraz do sytuacji rzeczywistej tzn. do sytuacji, gdzie jest wiele stanowisk roboczych wprowadza się funkcję ω gdzie zbiór N jest zbiorem stanowisk roboczych. Każdemu detalowi i każdej operacji, a więc czynności można przyporządkować stanowisko stanowiska na którym czynność tę można wykonać. Funkcja π charakteryzuje natomiast zdol-

ności (możliwości) poszczególnych stanowisk.

W związku z tym, że stanowiska mogą być wieloczynnościowe zdarzyć się może, że czynność można wykonać na kilku stanowiskach. Wobec tego wprowadza się zbiór \mathcal{T}_{pl} operatorów postaci T_{plr} , co oznacza, że czynność (p,l) wykonana może być na każdym stanowisku r dla którego:

$$T_{plr} \in \mathcal{T}_{pl}.$$

Wybór jednego stanowiska do wykonania czynności odbywa się przy pomocy funkcji sterującej β .

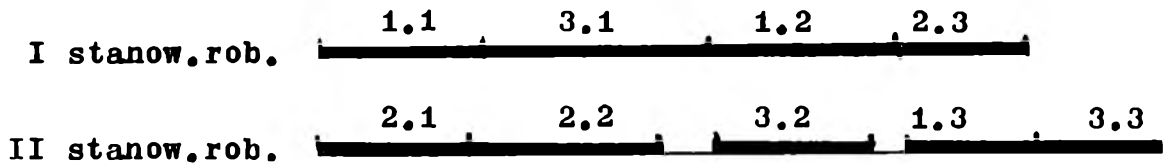
Wyjaśnijmy jeszcze sens pojęć: działanie potencjalnie n -równoległe i działanie n -równoległe. Rzecz polega na tym, że jeśli jest n stanowisk i żadne z nich nie jest zbędne w tym sensie, że może ono wykonać to najmniej jedną operację ze zbioru operacji potrzebnych do wykonania zadania, to istnieje potencjalna możliwość, że harmonogram dla całego odcinka produkcyjnego zawierał będzie zadania dla każdego z tych stanowisk. Innymi słowy harmonogram ten będzie sumą n harmonogramów dla stanowisk. Zależy to od funkcji sterującej β . Jeśli funkcja ta jest już ustalona i dziedziną jej jest pewien podzbiór N' zbioru N (w szczególności może to być zbiór N) to zadania zostały rozdzielone na n stanowisk i zadania te mogą być wykonywane równoległe: każde stanowisko wykonuje swoją czynność.

Pojęcie warstwy operatorów odpowiada stanowisku, a dokładnie mówiąc dla ustalonej funkcji sterującej α i β warstwa operatorów jest harmonogramem pracy stanowiska.

Funkcja θ jest opisem formalnym możliwości (zdolności) badanego odcinka produkcyjnego i jest równocześnie "sumą" funkcji ω .

Dla wyjaśnienia sensu dwu kolejnych terminów: współbieżnego oraz szeregowego działania operatorów jak i założeń, które zostały sformułowane w modelu formalnym a dotyczących działania operatorów posłużymy się następującym przykładem:

Dane są dwa stanowiska robocze, trzy wyroby dla wykonania których potrzebne są po trzy operacje. Jeden z możliwych harmonogramów przedstawimy na diagramie Gantta:



Pary liczb umieszczone na tym diagramie oznaczają czynności stanowisk. Ponieważ niektóre czynności są wykonywane jednocześnie np. (1,1) i (2,1) , a z czynnością (p,1) związany jest operator T_{p1} , stąd konieczność wprowadzenia pojęcia współbieżnego działania operatorów. Postulat 1^o na stronie 43 jest równoznaczny z założeniem, że żadne ze stanowisk nie wykonuje równocześnie dwóch różnych czynności.

Postulat 2^o jest równoznaczny z założeniem, że operacje na p-tym wyrobie wykonuje się w określonej kolejności (jeśli nie jest ona dana technologią to dana jest poprzez funkcję sterującą \mathcal{L}) i po drugie postulat ten odpowiada założeniu, że poszczególne operacje zostały rozdzielone w sensie definicji operacji technologicznej.

Natomiast postulat 3^o wymaga nieco szerszego omówienia.

Przy harmonogramach mówiliśmy o dopuszczalności harmonogramów (do sprawy tej jeszcze powrócimy), natomiast z punktu widzenia realizowalności, harmonogramów może być nieskończenie wiele. Wynika to z tego, iż nie powiedzieliśmy nic o tym kiedy przewidzianą harmonogramem czynność rozpoczynać. Niektórzy autorzy^{1/} rozpatrują harmonogramy, a wśród nich wydzielają klasę harmonogramów częściowo przesuwalnych w lewo, tzn. takich, dla których można przyspieszyć moment rozpoczęcia chociaż jednej operacji przewidzianej harmonogramem. Harmonogram, który nie jest częściowo przesuwalny w lewo nazywa się wówczas harmonogramem istotnym i tylko takie harmonogramy są przedmiotem badań.

Wydaje się, że proponowane w niniejszej pracy podejście jest prostsze. Otóż postulat 3^o żąda by czynność na stanowisku rozpoczynać najwcześniej jak jest to możliwe, tzn. wtedy, gdy stanowisko to jest już wolne i detal na którym ma być wykonywana ta czynność przeszedł już operację poprzedzającą.

Odpowiednikiem funkcji ϕ z modelu jest po prostu harmonogram badanego odcinka produkcyjnego, gdyż spełnia ona warunki jakie stawia się harmonogramom.

- 1^o dla każdego stanowiska wiadomo jaką czynnością rozpoczyna się jego praca,
- 2^o wiadomo jaka jest kolejna czynność tego stanowiska,
- 3^o odczytać można marszrutę technologiczną każdego detalu.

^{1/} Patrz Jankowska-Zorychta "Modele sekwencyjne i ich zastosowanie w planowaniu optymalnej organizacji w dyskretnych procesach produkcyjnych" str.37.

Dla przykładu 5 mamy następujące marszruty technologiczne dla:

detalu I - 1,4,3

detalu II - 2,4,3

detalu III - 1,3,2

Jeśli teraz interesuje nas optymalny harmonogram ze względu na czas wykonania wszystkich wyrobów ze zbioru W , musimy mieć dane czasy trwania każdej operacji na każdym stanowisku. Czasy te w modelu zadane są funkcjonalem H .

Sposób obliczania czasu wykonania całego zadania wynika z diagramu Gantta. Z tablicy $H(\phi)$ w połączeniu z funkcją ϕ odczytać można czasy zakończenia czynności na poszczególnych stanowiskach, licząc za zero czas rozpoczęcia najwcześniejszej dla badanego odcinka czynności.

Omówmy teraz kolejne zadania formalnie opisane w poprzednim rozdziale.

Zadanie A Jak się wydaje zadanie tego typu w przemyśle nie występuje, gdyż trudno zgodzić się z tym, iż dla wyrobu obojętna jest kolejność wykonywania operacji. Być może dla niektórych operacji kolejność jest obojętna, ale są i takie, dla których wymóg, by były wykonane w określonej kolejności jest istotny.

Zadanie B Zadanie to jest podobne do poprzedniego z tym tylko, że zakłada się iż żadne ze stanowisk nie jest wieloczynnościowe. Natomiast pozostaje w mocy założenie o tym, że kolejność wykonywania operacji dla poszczególnych detali jest obojętna.

Jak już powiedziano, zadania tego typu w praktyce przemysłowej spotyka się rzadko, tym niemniej ze względu na kolejne przykłady celowym jest ich formalne przedstawienie.

Zadanie C Jest to najbardziej typowe zadanie spotykane w praktyce przemysłowej. Dla każdego wyrobu dana jest technologia w postaci funkcji sterującej α_{pK} . Należy teraz tak rozplanować produkcję by była zachowana technologia i zadania zostały wykonane w możliwie najkrótszym czasie.

Sprowadza się to do znalezienia harmonogramu, a ponieważ buduje się go dla jednego uniwersalnego stanowiska stąd postulat by funkcja sterująca α_{MK} nie naruszała wartości żadnej z funkcji sterującej α_{pK} . Innymi słowy postulat ten zabezpiecza budowanie harmonogramów dopuszczalnych.

Zadanie D Zadanie tego typu otrzymuje się z zadania typu C przyjmując, że stanowiska są jednoczynnościowe. Sytuacja taka ma często miejsce w praktyce.

Zwróćmy uwagę na fakt, że zadanie typu D jest prostsze niż zadanie typu C, gdyż należy wyznaczyć tylko funkcję sterującą α natomiast funkcja β jest już z góry zadana.

Często w konkretnej sytuacji, gdy zadanie C jest dużych rozmiarów można je sprowadzić do zadania D w sposób następujący:

Dla każdej z czynności wybrać jedno tylko stanowisko, na którym czynność ta ma być wykonana i to tak, by żadne ze stanowisk po takim przyporządkowaniu nie było wieloczynnościowe. Wówczas otrzymujemy zadanie D.

Jest rzeczą oczywistą, że nie zawsze daje się to zrobić. Ponadto jeśli jest to możliwe i to na kilka sposobów, to trzeba zdecydować się które przyporządkowanie wybrać, a więc powstaje tu nowe zagadnienie wyboru kryterium orzekającego o wyższości jednego rozwiązania nad innym. Po trzecie wreszcie sposób taki jest wyraźnym uproszczeniem zadania wyjściowego.

Zadanie E Zadanie tego typu jest najczęściej spotykanym w literaturze. Zadanie to sprowadza się do tego, że:

- każdy detal ma identyczną technologię,
 - każdy detal ma identyczną marszrutę technologiczną,
 - harmonogram dla każdego stanowiska zawiera ciąg czynności, ciągi te dla różnych stanowisk różnią się tylko operacją.
- Oznacza to, że stanowiska obrabiają detale w tej samej kolejności i stanowisko jest jednoczynnościowe.

Zadanie to odbiega od typowych zadań z jakimi spotyka się w praktyce. Jednak ze względu na swoją prostotę (w porównaniu z innymi) stało się obiektem zainteresowania wielu matematyków i praktyków. "Może dlatego ta wielce uproszczona wersja problemu ogólnego stała się poligonem doświadczalnym, na którym badacze usiłują wypracować jakieś sensowne metody dla tej klasy zadań"^{1/}.

^{1/} Patrz Jankowska-Zorychta "Modele sekwencyjne i ich zastosowanie... str.42

Mimo wielu prób podejmowanych w celu rozwiązania tak uproszczonego zadania wyniki są dalekie od zadowalających. Do sprawy tej powrócimy przy okazji prezentowania znanych metod rozwiązania zadań sekwencyjnych.

Zwróćmy uwagę na jeszcze dwie rzeczy.

Jeśli stanowisko jest wieloczynnościowe, oczywiste jest, że przystosowanie go do wykonania innej operacji wymaga czasu. Tak więc dla stanowisk wieloczynnościowych powinna być jeszcze dana macierz czasów przebrojeń z operacji na operację. Jednak uwzględnienie tych danych nie komplikuje zadania, gdyż czasy te można doliczyć do czasów trwania operacji.

Jeśli w badanym odcinku produkcyjnym jest jedno tylko stanowisko robocze to z konieczności musi być ono wieloczynnościowe (zbiór N jest jednoelementowy) a więc operatory działają szeregowo w dalszym ciągu pozostaje problem wyznaczenia optymalnego harmonogramu, gdyż ze względu na czasy przebrojeń stanowiska, różne harmonogramy mogą dawać różne czasy wykonania zadania.

Na zakończenie zauważmy, że często mamy do czynienia z sytuacją gdy w badanym odcinku produkcyjnym jest grupa stanowisk jednorodnych. Wówczas można zamiast całej grupy stanowisk jednorodnych rozpatrywać jedno stanowisko, biorąc za czas trwania operacji na tym stanowisku liczbę:

$$\frac{H(T_{pl})}{r},$$

gdzie:

$H(T_{pl})$ - czas trwania czynności (p, l) na jednym ze stanowisk jednorodnych,

r - ilość stanowisk jednorodnych mogących wykonać czynność (p, l)

2.2. U s t a l a n i e h a r m o n o g r a m u z a j ę ć

Rozpatrzmy jednostkę dydaktyczną jaką może być np. szkoła, uczelnia, wydział, itp. Z interesującego nas punktu widzenia jednostkę tę charakteryzują następujące dane:

- ilość i rodzaj sal dydaktycznych wykładowe, seminaryjne, specjalistyczne ,
- ilość i specjalizacja wykładowców (przez wykładowcę rozumiemy tutaj osobę, która może w danej jednostce dydaktycznej prowadzić zajęcia), którymi dana jednostka dysponuje,
- ilość i rodzaj zajęć dydaktycznych prowadzonych w danej jednostce,
- ilość i rodzaj grup dydaktycznych, które dana jednostka ma obsłużyć.

Zwróćmy uwagę na następujący fakt: minimalizacja czasu wykonania zadania w wielu przypadkach jest równoważna z minimalizacją sumy czasów przerw międzyoperacyjnych. Z tego też względu zagadnienie układania harmonogramów ma dwa aspekty powodujące, że mogą tu powstać dwa zadania różne co do treści, a nierozróżnialne pod względem formalnym. Zadania te nazwiemy odpowiednio zadanie I i zadanie II.

Rzecz polega na tym, że grupy dydaktyczne mają ustalone kryterium oceny jakości harmonogramu. Może to być dla przykładu kryterium następujące: zajęcia dla danej grupy powinny trwać w sposób ciągły, tzn. by nie było okienek. Jeśli z jakichś powodów (przerwa na obiad, przerwa przed trudnymi zajęciami na powtórzenie) okienka takie są potrzebne, to można je w naszym zadaniu wprowadzić traktując jako zajęcia dydaktyczne.

W obu typach zadań sprawa ta nie odgrywa większej roli. Ważną natomiast rzeczą jest co innego, ograniczenia jednostki dydaktycznej mogą wynikać z braku sal bądź braku wykładowców. Wobec tego w pierwszym przypadku otrzymane harmonogramy będzie się porównywało ze względu na wykorzystanie sal, a drugim razem ze względu na wykorzystanie wykładowców. Innymi słowy w pierwszym przypadku harmonogram będzie tym lepszy im suma czasów kiedy sale są wolne będzie mniejsza, w drugim przypadku suma okienek wykładowców. W obu tych przypadkach liczy się okienka między zajęciami, a interes grupy dydaktycznej musi zostać podporządkowany racjom wyższego rzędu. Brak sal bądź wykładowców - zajęcia muszą odbywać się do późnego wieczora. Jeśli natomiast jest dostatecznie dużo sal i wykładowców o wyższości jednego harmonogramu nad innymi decyduje interes grupy, a więc np. minimum czasu przerw między zajęciami.

Wracając do naszego zadania rozpatrzemy dwa przypadki:

Zadanie I

Ograniczenia wynikają z braku sal.

Zadanie II

Ograniczenia wynikają z braku wykładowców.

Dla obu zadań wprowadzimy następujące oznaczenia, zgodne z formalnym opisem z poprzedniego rozdziału.

S - zbiór grup

K - zbiór przedmiotów, wzbogacony być może o postulowane przez przerwy. Zbiór ten jest sumą zbiorów K_i

($i=1,2,\dots,m$) tzn. przedmiotów dla poszczególnych grup.

Operator T_{p1} jest opisem formalnym zajęć dydaktycznych z przedmiotu l w grupie p .

Zadanie I

N - zbiór sal.

Ponieważ część sal jest uniwersalna, funkcja ω przyporządkowuje poszczególnym zajęciom sale w których zajęcia te mogą się odbywać. W poprzednim zadaniu z dziedziny organizacji produkcji odpowiada to sytuacji, że stanowiska są wieloczynnościowe. Jeśli sale są specjalistyczne, dostosowane do jednego typu zajęć wówczas funkcja ω dla odpowiedniego zajęcia wybierze właśnie tę salę. Funkcja ϕ jest po prostu harmonogramem zajęć uwzględniającym sale, grupę i przedmiot, z którego odczytać można dla każdej sali zajęcia kolejne oraz dla każdej grupy rozkład zajęć tzn. jakie kolejne zajęcia i w której sali się odbywają.

Zadanie II

N - zbiór wykładowców.

Funkcja ω wybierać teraz będzie dla każdego zajęcia podzbiór zbioru wykładowców mogących dane zajęcia poprowadzić. W tym przypadku funkcja ϕ będzie harmonogramem zajęć uwzględniającym wykładowców, grupę oraz przedmiot. W obu przypadkach funkcjonal H podaje czas trwania poszczególnych zajęć. Zadanie I jak i zadanie II sprowadzają się do zadań typu A lub B z poprzedniego rozdziału, gdyż na ogół kolejność zajęć dydaktycznych jest obojętna poza sporadycznymi wypadkami.

Harmonogramy podane w zadaniu I i II trudno uznać za harmonogramy dla jednostki dydaktycznej, gdyż ten powinien uwzględniać wszystkie elementy. W przypadku I brak jest wykładowców, w przypadku II brak jest sal. Ale jeśli się zważy na to, iż zadania te były formułowane przy założeniu "obfitości" jednych elementów w porównaniu z innymi, można otrzymany harmonogram uzupełnić o brakujące w nim elementy. Co oczywiście może powodować lokalne zmiany w harmonogramie.

Na zakończenie jeszcze jedna uwaga. Zajęcia mogą być różnego typu (ćwiczenia, wykłady) niektóre z nich muszą się odbywać w określonej porze (obiad). Wszystkie te warunki uwzględnić można poprzez dobór odpowiedniej funkcji sterującej α .

2.3. P l a n o w a n i e ś w i a d c z e n i a u s ł u g^{1/}

Przyjmijmy, że dana jest jednostka usługowa świadcząca usługi specjalistyczne na rzecz klientów. Załóżmy, że każdą usługę można rozbić na pojedyncze operacje. Wówczas z naszego punktu widzenia jednostka taka scharakteryzowana jest następującymi zbiorami:

S - zbiór klientów na rzecz których jednostka ma świadczyć swoje usługi,

N - zbiór wykonawców mogących te usługi wykonać,

K - zbiór operacji, które mogą być przez wykonawców wykonane, a które są niezbędne do wykonania usług na rzecz klientów.

^{1/} przykład zaczerpnięty z "Zastosowanie pewnej metody heurystycznej do rozwiązywania zagadnień z teorii przedsiębiorstw czasowych". A. Gospodarowicz.

Każdy klient ze zbioru S jest opisany przy pomocy wektora:

$$(s_{i1}, s_{i2}, \dots, s_{ik_i})$$

gdzie:

s_{i1} - jest 1-tą cechą dla i -tego klienta,

k_i - jest liczbą cech opisujących i -tego klienta.

Przy czym cechy te są niepożądane, a ich wystąpienie powoduje to, że klient trafia do jednostki usługowej. Usługa świadczona przez jednostkę na rzecz klienta formalnie polega na przekształceniu wektora s_i w nowy wektor:

$$w_i = (w_{i1}, w_{i2}, \dots, w_{ik_i})$$

gdzie:

w_{i1} - jest 1-tą cechą i jest to cecha pożądana.

Przy czym cechę w_{i1} uzyskuje się z cechy s_{i1} dzięki wykonaniu czynności $(i, 1)$.

Przyjmując dodatkowe założenia:

- każda z operacji jest wykonywana przez jednego tylko wykonawcę,
- nie ma przerw w trakcie wykonywania operacji, tzn. operacja trwa od momentu jej rozpoczęcia aż do momentu jej zakończenia

można sformułować następujące zadanie:

Ustalić taki plan wykonywania usług dla danego zbioru klientów by wykonać je w jak najkrótszym czasie.

Zadanie to jest szczególnym przypadkiem zadania sformułowanego w pierwszym rozdziale.

Zauważmy, że dla niektórych typów usług z góry dana jest kolejność wykonywania poszczególnych operacji, dla innych natomiast kolejność ta może być dowolna. Warunki opisujące kolejność operacji wprowadza się do modelu tak jak i w poprzednich przypadkach poprzez dobór odpowiedniej funkcji sterującej \mathcal{O} .

Operator T_{p1} jest tutaj formalnym odpowiednikiem l-tej operacji wykonywanej na rzecz p-tego klienta.

Funkcja \mathcal{O} wybiera ze zbioru wykonawców tych wykonawców, którzy mogą wykonać czynność (p, l) . Harmonogram świadczenia usług jest równoznaczny z funkcją \emptyset .

W przypadku planowania świadczenia usług spotkać można każdy typ zadania opisanego w rozdziale pierwszym.

Na zakończenie tego rozdziału zwróćmy uwagę na jeszcze jeden fakt. Chciałoby się aby harmonogram zawierał informacje typu: co?, gdzie?, kiedy?. Innymi słowy by w dowolnej chwili od momentu rozpoczęcia zadania można było z harmonogramu odczytać na jakim etapie znajduje się jego realizacja. Dla celów kontroli przebiegu realizacji zadania niezbędna jest możliwość porównywania planu realizacji i realizacji. Konieczną jest rzeczą zatem aby z harmonogramu odczytywać można było następujące informacje: co już powinno być zrobione?, co powinno się robić aktualnie? itp. Podany przez nas harmonogram (funkcja \emptyset) nie zawiera tych informacji co oczywiście jest jego wadą. Po to by mieć pełne informacje o planowanym przebiegu reali-

zacji zadania korzystać trzeba z funkcjonału H oraz funkcji \emptyset .

Te dwie tablice \emptyset i $H(\emptyset)$ dają pełny obraz przebiegu realizacji zadania.

3. METODY ROZWIĄZYWANIA ZADAŃ PROGRAMOWANIA SEKWENCYJNEGO

3.1. W p r o w a d z e n i e

W rozdziale tym chcemy przedstawić znane w literaturze metody rozwiązywania zadań sekwencyjnych. Zaznaczmy tu od razu, iż metody te były opracowane w przeważającej większości dla zadań z dziedziny organizacji produkcji, a wśród nich najczęściej formułowanym zadaniem było zadanie, które w modelu zaprezentowanym w rozdziale pierwszym zostało opisane jako zadanie typu E.

Ponadto w rozdziale tym chcemy zaproponować własne propozycje sposobów rozwiązywania tego typu zadań.

Jak już powiedziano powyżej najczęściej spotykanym w literaturze zadaniem jest zadanie, które w języku formalnym zaproponowanym w rozdziale pierwszym formułuje się jak następuje:

Dane:

zbiory M, N, K .

Zakłada się, że:

$$\forall p \in M \wedge \forall l \in K \quad \text{dany jest operator } T_{pl}^{1/}$$

$$\text{card}(N) = \text{card}(K)$$

$$\forall p_1 \in M \wedge \forall p_2 \in M \quad \omega(p_1, l) = \omega(p_2, l)$$

$$\forall p \in M \quad \text{card}(N_{pl}) = 1 \wedge N_{pl} = \{l\}$$

$$\forall l \in K \wedge \forall p_1, p_2 \in M \quad \alpha_{p_1 K}^{-1}(T_{p_1 l}) = \alpha_{p_2 K}^{-1}(T_{p_2 l})$$

^{1/} założenie to nie jest najistotniejsze, gdyż jeśli dla jakiegoś p nie istnieje operator T_{pl} to operator taki można wprowadzić przyjmując, że $M(T_{pl}) = 0$.

Wyznaczyć funkcję sterującą α taką, by:

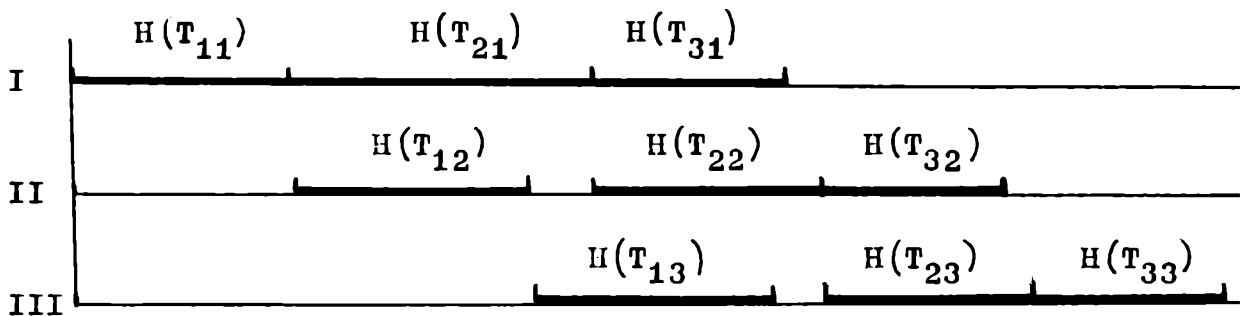
$$H(T_{MK}^{\alpha}) \longrightarrow \min_{\alpha}$$

Aby lepiej zilustrować sposób wyznaczania wartości funkcjonalu H na zbiorze (lub podzbiorze) operatorów T_{MK} posłużymy się diagramem Gantta. Dla przykładu weźmy trzy warstwy operatorów oraz funkcję α następującą:

$$\alpha(1) = T_{1K}$$

$$\alpha(2) = T_{2K}$$

$$\alpha(3) = T_{3K}^{1/}$$



Mamy tutaj:

$$H(T_{MK}) = \max_i H(Z_i) = H(Z_3)$$

$$H(T_{33} \times T_{MD}) = H(T_{33}) + \max[H(Z_3), H(Z_2|T_{32})] = H(T_{33}) + H(Z_3).$$

Zauważmy jeszcze, że:

^{1/} Zapis ten oznacza, że:

$$\alpha_{1K}(1) = T_{11}, \quad \alpha_{1K}(2) = T_{12}, \quad \alpha_{1K}(3) = T_{13}$$

$$\alpha_{2K}(1) = T_{21}, \quad \alpha_{2K}(2) = T_{22}, \quad \alpha_{2K}(3) = T_{23}$$

$$\alpha_{3K}(1) = T_{31}, \quad \alpha_{3K}(2) = T_{32}, \quad \alpha_{3K}(3) = T_{33} .$$

$$H(Z_1) = \sum_{p=1}^m H(T_{p1})$$

$$H(T_{1K}) = \sum_{l=1}^n H(T_{1l})$$

$$H(T_{M,D}) = H(T_{p1} \times T_{M,D}) = H(T_{p1}) + \max \left[H(Z_1), H(Z_{1-1} | T_{p,1-1}) \right].$$

Z powyższych wzorów korzystać będziemy przy prezentowaniu znanych algorytmów.

3.2. A l g o r y t m J o h n s o n a

Jedynym jak dotąd efektywnym algorytmem pozwalającym wyznaczyć optymalną funkcję sterującą α jest algorytm opracowany w 1954 przez Johnsona. Jednak jego przydatność jest niewielka, gdyż opracowany został dla dwóch warstw operatorów, a w szczególnych przypadkach można stosować go dla trzech warstw operatorów.

Przedstawimy pokrótce ten algorytm oraz pochodzący od jego autora dowód efektywności^{1/}.

Niech $\text{card}(N) = 2$ zaś zbiór M niech będzie zbiorem m -elementowym. Niech ponadto funkcja sterująca α_{pK} gdzie $K = \{1, 2\}$ będzie jak następuje:

$$\forall p \in M \quad \alpha_{pK}(1) = T_{p1}$$

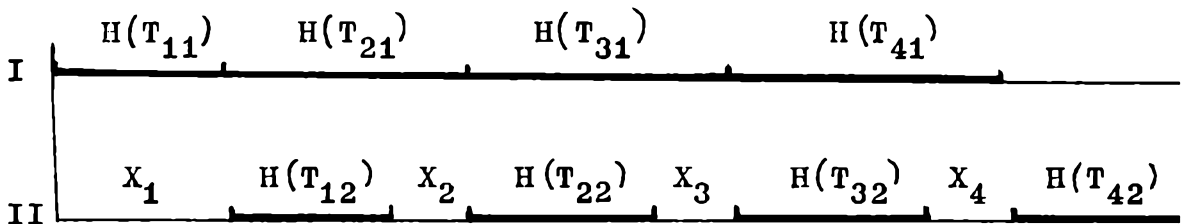
$$\alpha_{pK}(2) = T_{p2}$$

^{1/} nieco inny dowód znaleźć można w pracy: Korbut A.A.; Finkelsztejn J.J. "Programowanie dyskretne", str.202 i dalej

Założmy, że mamy dowolną funkcję sterującą \mathcal{L}_{MK} , bez straty ogólności przyjmijmy, że:

$$\forall p \in M \quad \mathcal{L}_{MK}(p) = T_{pK}$$

Graficznie wartości funkcjonału M na zbiorze operatorów T_{MK} przedstawić można przy pomocy znanego już diagramu Gantta:



Ponieważ operatory T_{p1} oraz T_{p2} muszą działać szeregowo a z ich działaniem związany jest czas, X_p oznacza czas oczekiwania operatora T_{p2} na rozpoczęcie działania.

Zauważmy z diagramu, że:

$$H(T_{MK}) = \sum_{p=1}^m H(T_{p2}) + \sum_{p=1}^m X_p.$$

Zatem, wyznaczenie optymalnej funkcji sterującej \mathcal{L} sprowadza się do minimalizacji składnika:

$$\sum X_p$$

gdyż drugi składnik jest stały.

Mamy:

$$X_1 = H(T_{11}),$$

$$X_2 = \max \left[H(T_{11}) + H(T_{21}) - H(T_{12}) - X_1, 0 \right],$$

$$X_1 + X_2 = \max \left[H(T_{11}) + H(T_{21}) - H(T_{12}), H(T_{11}) \right],$$

$$X_3 = \max \left[\sum_{p=1}^3 H(T_{p1}) - \sum_{p=1}^2 H(T_{p2}) - \sum_{p=1}^2 X_p, 0 \right],$$

$$\begin{aligned} \sum_{p=1}^3 X_p &= \max \left[\sum_{p=1}^3 H(T_{p1}) - \sum_{p=1}^2 H(T_{p2}), \sum_{p=1}^2 X_p \right] = \\ &= \max \left[\sum_{p=1}^3 H(T_{p1}) - \sum_{p=1}^2 H(T_{p2}), \sum_{p=1}^2 H(T_{p1}) - H(T_{p1}), H(T_{11}) \right]. \end{aligned}$$

i ogólnie:

$$\sum_{p=1}^m X_p = \max_{1 \leq u \leq m} B_u$$

gdzie:

$$B_u = \sum_{p=1}^u H(T_{p1}) - \sum_{p=1}^{u-1} H(T_{p2}).$$

Oznaczmy teraz:

$$F(\alpha) = \max_{1 \leq u \leq m} B_u$$

Poszukujemy takiej funkcji sterującej, α_0 , dla której:

$$F(\alpha_0) \leq F(\alpha) \text{ dla każdej } \alpha \in R(M, T_{MK}).$$

Rozważmy dwie funkcje α oraz α_1 .

Niech funkcje te będą następujące:

$$\alpha(i) = \alpha_1(i) \text{ dla } i < j \text{ oraz } i > j+1$$

$$\alpha(j) = \alpha_1(j+1)$$

$$\alpha(j+1) = \alpha_1(j)$$

gdzie:

$$i = 1, 2, \dots, m$$

$$j = 1, 2, \dots, m-1$$

Innymi słowy sterowanie funkcją α_1 jest takie same jak sterowanie funkcją α dla wszystkich $i \in M$ z wyjątkiem dwóch dowolnych (byle sąsiednich) operatorów, które są przedstawione. Wartości B_u oraz B'_u dla funkcji α i α_1 będą wobec tego takie same dla każdego u z wyjątkiem być może $u = j$ oraz $u' = j+1$.

Mamy więc:

$$F(\alpha) = F(\alpha_1)$$

Pod warunkiem, że:

$$\max(B_j, B_{j+1}) = \max(B'_j, B'_{j+1}).$$

Jeśli zaś:

$$\max(B_j, B_{j+1}) \neq \max(B'_j, B'_{j+1})$$

to jedna z dwóch funkcji sterujących α i α_1 jest lepsza w sensie minimalizacji $H(T_{MK})$.

Funkcja α dla której:

$$\alpha(j) = T_{j1}$$

$$i \quad \alpha(j+1) = T_{j+1,1}$$

jest lepsza od funkcji α_1 , dla której:

$$\alpha_1(j) = T_{j+1,1}$$

$$i \quad \alpha_1(j+1) = T_{j1}$$

jeśli:

$$\max(B_j, B_{j+1}) < \max(B'_j, B'_{j+1}) \quad 3.1$$

ale:

$$\max(B_j, B_{j+1}) = \max \left[\sum_{p=1}^j H(T_{p1}) - \sum_{p=1}^{j-1} H(T_{p2}), \sum_{p=1}^{j+1} H(T_{p1}) - \sum_{p=1}^j H(T_{p2}) \right]$$

i

$$\max(B'_j, B'_{j+1}) = \max \left[\sum_{p=1}^{j+1} H(T_{p1}) + H(T_{j+1,1}) - \sum_{p=1}^{j+1} H(T_{p2}) \right].$$

$$\sum_{p=1}^{j+1} H(T_{p1}) - \sum_{p=1}^{j-1} H(T_{p2}) - H(T_{j+1,2})$$

Jeśli powyższe podstawimy do nierówności 3.1 i obustronnie dodamy wyrażenie:

$$\sum_{p=1}^{j-1} H(T_{p2}) - \sum_{p=1}^{j+1} H(T_{p1})$$

to po wykonaniu odpowiednich działań otrzymujemy następującą nierówność:

$$\max \left[-H(T_{j+1,1}), -H(T_{j2}) \right] < \max \left[-H(T_{j1}), -H(T_{j+1,2}) \right],$$

lub równoważną jej:

$$\min \left[H(T_{j1}), H(T_{j+1,2}) \right] < \min \left[H(T_{j+1,1}), H(T_{j2}) \right].$$

Rozpatrzmy pewien porządek działania operatorów:

$$T_{1K}, \dots, T_{1K}, T_{sK}, \dots, T_{mK}$$

ustalony przez funkcję sterującą α_0 .

Nie zmienimy wartości funkcji \mathcal{L}_0 dla argumentu 1 oraz s jeśli:

$$\min \left[H(T_{11}), H(T_{s2}) \right] \leq \min \left[H(T_{s1}), H(T_{12}) \right]. \quad 3.2$$

Wzór 3.2 jest spełniony, gdy:

$$H(T_{11}) \leq H(T_{s1}), H(T_{s2}), H(T_{12}).$$

można go wówczas zapisać:

$$\min \left[H(T_{11}), H(T_{12}) \right] \leq \min \left[H(T_{s1}), H(T_{s2}) \right].$$

Jeśli więc funkcjonal H zadany na zbiorze operatorów ma własność, że:

$$\min_{l,i} H(T_{li}) = H(T_{s1})$$

$$i = 1, 2$$

$$l = 1, 2, \dots, m$$

to optymalna funkcja sterująca \mathcal{L}_0 powinna mieć następującą własność:

$$\mathcal{L}_0(1) = T_{sK}.$$

Jeśli zaś:

$$\min_{l,i} H(T_{li}) = H(T_{s1}) = H(T_{j1})$$

to funkcja \mathcal{L}_0 nie jest wyznaczona jednoznacznie, dokładniej mówiąc mamy wówczas dwie funkcje (obie optymalne), takie, że:

$$\mathcal{L}_0(1) = T_{sK}$$

$$\mathcal{L}_1(1) = T_{jK}.$$

Wzór 3.2 jest również spełniony gdy:

$$H(T_{s2}) \leq H(T_{s1}), H(T_{11}), H(T_{12}).$$

Wówczas można go zapisać w innej nieco postaci:

$$\min [H(T_{s1}), H(T_{s2})] \leq \min [H(T_{11}), H(T_{12})].$$

Jeśli więc

$$\min_{l,i} H(T_{li}) = H(T_{s2})$$

$$l = 1, 2$$

$$i = 1, 2, \dots, m$$

to optymalna funkcja sterująca α_0 powinna mieć następującą własność:

$$\alpha_0(m) = T_{sK}.$$

Ponadto jeśli:

$$\min_{l,i} H(T_{li}) = H(T_{s2}) = H(T_{j2})$$

to są dwie funkcje α_0 i α_1 , obie optymalne, takie że:

$$\alpha_0(m) = T_{sK}$$

$$\alpha_1(m) = T_{jK}.$$

Tak więc algorytm Johnsona pozwala w m krokach wyznaczyć optymalną funkcję sterującą α_0 .

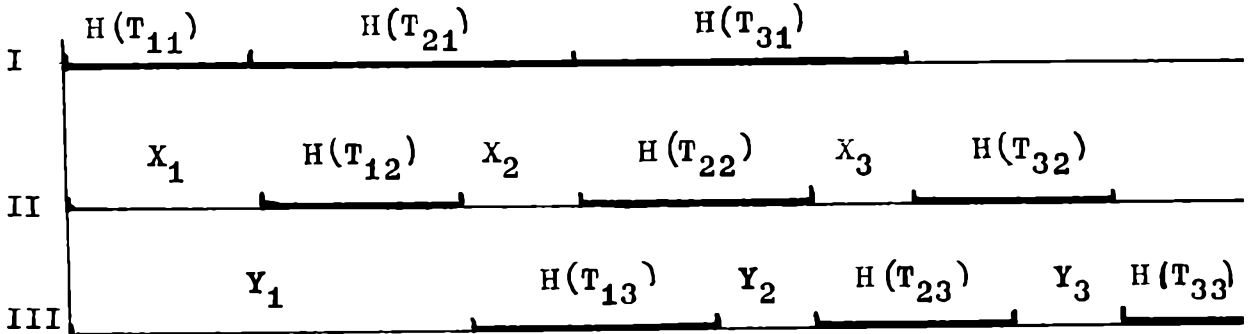
Johnson podał również algorytm wyznaczania optymalnej funkcji sterującej w przypadku 3-równoległego działania operatorów. Algorytm ów stosuje się w jednym z dwóch przypadków:

$$\min_i H(T_{i1}) \geq \max_i H(T_{i2})$$

$$\text{lub } \min_i H(T_{i3}) \geq \max_i H(T_{i2})$$

Narysujmy diagram Gantta dla trzech warstw operatorów i dla dowolnej funkcji sterującej α . Przyjmijmy, że:

$$\forall_{i \in M} \alpha(i) = T_{iK}$$



gdzie:

X_i oraz Y_i oznaczają to samo co poprzednio tylko odpowiednio dla warstwy drugiej i trzeciej.

Z diagramu widać, że:

$$Y_1 = X_1 + H(T_{12}),$$

$$Y_2 = \max \left[H(T_{12}) + H(T_{22}) + X_1 + X_2 - H(T_{13}) - Y_1, 0 \right],$$

i ogólnie:

$$Y_m = \max \left[\sum_{p=1}^m H T_{p2} + \sum_{p=1}^m X_p - \sum_{p=1}^{m-1} H T_{p3} - \sum_{p=1}^{m-1} Y_p, 0 \right]$$

oraz:

$$\sum_{p=1}^m Y_p = \max \left[\sum_{p=1}^m H T_{p2} - \sum_{p=1}^{m-1} H T_{p3} + \sum_{p=1}^m X_p, \sum_{p=1}^{m-1} Y_p \right] =$$

$$= \max \left[\sum_{p=1}^m H(T_{p2}) - \sum_{p=1}^{m-1} H(T_{p3}) + \sum_{p=1}^m X_p, \sum_{p=1}^{m-1} H(T_{p2}) - \right. \\ \left. - \sum_{p=1}^{m-2} H(T_{p3}) + \sum_{p=1}^{m-1} X_p, \dots, \sum_{p=1}^2 H(T_{p2}) - H(T_{13}) + \sum_{p=1}^2 X_p, H(T_{12}) + X_1 \right].$$

Niech:

$$A_v = \sum_{p=1}^v H(T_{p2}) - \sum_{p=1}^{v-1} H(T_{p3}) \quad 3.3$$

$$v = 1, 2, \dots, m$$

$$B_u = \sum_{p=1}^u H(T_{p1}) - \sum_{p=1}^{u-1} H(T_{p2})$$

$$u = 1, 2, \dots, m$$

Wówczas dostajemy:

$$\sum_{p=1}^m Y_p = \max_{1 \leq u \leq v \leq m} (A_v + \max B_u) = \max (A_v + B_u).$$

Rozpatrzmy znowu dwie funkcje sterujące α_1 i α_2 takie, że:

$$\alpha_1(i) = \alpha_2(i) \quad \forall i : 1 \leq i < j \quad j+1 < i \leq m$$

$$\alpha_1(j) = \alpha_2(j+1)$$

$$\alpha_1(j+1) = \alpha_2(j).$$

Otrzymamy w ten sposób dwa założenia operatorów:

$$T_{1K}, T_{2K}, \dots, T_{jK}, T_{j+1,K}, \dots, T_{mK}$$

oraz

$$T_{1K}, T_{2K}, \dots, T_{j+1,K}, T_{jK}, \dots, T_{mK}.$$

Dla obu funkcji α_1 oraz α_2 musi zachodzić następująca relacja:

$$\max_{1 \leq u \leq v < j} (A_v + B_u) = \max_{1 \leq u \leq v < j} (A'_v + B'_u).$$

Być może relacja ta nie zachodzi natomiast dla $u = j$ i $u = j+1$. Porównajmy zatem:

$$\max_{1 \leq u \leq j+1} (A_{j+1} + B_u, A_j + B_u)$$

dla funkcji α_1 , oraz:

$$\max_{1 \leq u \leq j+1} (A'_{j+1} + B'_u, A'_j + B'_u)$$

dla funkcji α_2 .

Jeśli teraz:

$$\min_p H(T_{p1}) \geq \max_p H(T_{p2})$$

to:

$$\max_{u \leq v} B_u = B_v.$$

Zatem funkcja sterująca α_1 jest lepsza od funkcji α_2 , jeśli:

$$\max(A_{j+1} + B_{j+1}, A_j + B_j) < \max(A'_{j+1} + B'_{j+1}, A'_j + B'_j) \quad 3.4$$

Napiszmy jeszcze wartości wyrażeń A'_{j+1} , B'_{j+1} , A'_j , B'_j dla funkcji α_2 .

Korzystając z 3.3 oraz z tego, że:

$$\alpha_2(j) = T_{j+1,K}$$

$$\alpha_2(j+1) = T_{jK}$$

mamy:

$$A'_{j+1} = \sum_{p=1}^{j+1} H(T_{p2}) - \sum_{p=1}^{j-1} H(T_{p3}) - H(T_{j+1,3}),$$

$$B'_{j+1} = \sum_{p=1}^{j+1} H(T_{p1}) - \sum_{p=1}^{j-1} H(T_{p2}) - H(T_{j+1,2}),$$

$$A'_j = \sum_{p=1}^{j-1} H(T_{p2}) + H(T_{j+1,2}) - \sum_{p=1}^{j-1} H(T_{p3}).$$

$$B'_j = \sum_{p=1}^{j-1} H(T_{p1}) + H(T_{j+1,1}) - \sum_{p=1}^{j-1} H(T_{p2}),$$

Jeśli teraz do obu stron nierówności 3.4 dodamy wyrażenie:

$$\sum_{p=1}^{j-1} H(T_{p2}) - \sum_{p=1}^{j+1} H(T_{p1}) - \sum_{p=1}^{j+1} H(T_{p2}) + \sum_{p=1}^{j-1} H(T_{p3})$$

to po wykonaniu działań dostajemy:

$$\max \left[-H(T_{j2}) - H(T_{j3}), -H(T_{j+1,2}) - H(T_{j+1,1}) \right] <$$

$$\max \left[-H(T_{j+1,2}) - H(T_{j+1,3}), -H(T_{j2}) - H(T_{j1}) \right],$$

lub

$$\min \left[H(T_{j1}) + H(T_{j2}), H(T_{j+1,2}) + H(T_{j+1,3}) \right] <$$

$$\min \left[H(T_{j+1,1}) + H(T_{j+1,2}), H(T_{j2}) + H(T_{j3}) \right].$$

A po podstawieniu

$$H^{\dagger}(T_{j1}) = H(T_{j1}) + H(T_{j2})$$

$$H^{\dagger}(T_{j2}) = H(T_{j2}) + H(T_{j3})$$

$$H^{\dagger}(T_{j+1,1}) = H(T_{j+1,1}) + H(T_{j+1,2})$$

$$H(T_{j+1,2}) = H(T_{j+1,2}) + H(T_{j+1,3})$$

$$\min [H(T_{j1}), H(T_{j+1,2})] < \min [H(T_{j2}), H(T_{j+1,1})].$$

Uzyskaliśmy taki sam wynik jak poprzednio kiedy rozpatrywano przypadek dwóch warstw operatorów.

Z przeprowadzonego rozumowania widać, że zadanie dla trzech warstw operatorów w przypadku, gdy:

$$\min_i H(T_{i1}) \geq \max_i H(T_{i2})$$

sprowadza się do rozpatrzonego już zadania dla dwóch warstw operatorów. Dowód dla przypadku:

$$\min_i H(T_{i3}) \geq \max_i H(T_{i2})$$

przebiega podobnie.

Na zakończenie przytoczmy uwagę Johnsona. W niektórych przypadkach poszukiwanie optymalnej funkcji sterującej dla trzech warstw operatorów można zastąpić dwoma zadaniami poszukiwania optymalnej funkcji dla dwóch warstw operatorów: osobno dla pierwszej i drugiej warstwy oraz drugiej i trzeciej warstwy. Jeśli obie funkcje będą jednakowe to jest ona optymalną dla całego zadania.

Mimo niewątpliwej prostoty algorytmu przytoczonego powyżej jego przydatność praktyczna jest niewielka.

3.3. M o d e l e c a ł k o w i t o - l i c z b o w e

Zadania programowania sekwencyjnego są bardzo często formułowane jako zadania programowania dyskretnego, a ściślej mówiąc, jako zadania programowania całkowito-liczbowego. Trzeba powiedzieć, że proponowane modele są interesujące z formalnego punktu widzenia, przy ich budowie uciekano się do wielu pomysłowych zabiegów, jednak ich praktyczna przydatność jest niewielka. Wynika to po pierwsze z faktu, iż są to zadania o dużych rozmiarach, a po drugie - co zresztą ma związek z pierwszym - metody rozwiązywania zadań o zmiennych całkowitych mimo postępów badań w tej dziedzinie są ciągle mało efektywne.

W literaturze^{1/} znaleźć można takie zdanie: "Przy rozwiązywaniu zadań PLC metodami płaszczyzn odcinających notuje się zarówno sukcesy jak i porażki ...najbardziej charakterystycznymi zadaniami, dla których miało miejsce niepowodzenie są: ...zadania teorii harmonogramów" i dalej "obecnie brakuje wyczerpujących uzasadnień tak sukcesów jak i porażek... wszakże dla zadania teorii harmonogramów wydaje się słuszne następujące rozumowanie. Formułowanie tych zadań w języku PLC jest nienaturalne. Dla stosunkowo niewielkiego zadania w naturalnym sformułowaniu w modelu PLC występuje ogromna liczba ograniczeń i zmiennych".

^{1/} patrz Korbut A.A. i Finkelsztejn J.J. "Programowanie dyskretne" str. 161 i dalej.

Dla potwierdzenia tej tezy prezentując modele programowania całkowito-liczbowego podamy bardzo proste przykłady.

Przed opisem znanych modeli przypomnijmy założenia dotyczące działania operatorów:

- operatory należące do jednej warstwy działają kolejno,
- każde dwa operatory $T_{p_1 l_1}$ i $T_{p_1 l_2}$ działają kolejno,
- jeśli operator $T_{p_1 l_1}$ może działać współbieżnie z dwoma $T_{p_2 l_2}$ i $T_{p_3 l_3}$, to działa on z tym dla którego wartość funkcji α_{MK}^{-1} jest mniejsza.

W większości modeli, które zostaną tu zaprezentowane dokonuje się następującego zabiegu:

Wybiera się liczbę T , o której wiadomo, że:

$$T > H(T_{MK}^{\beta}).$$

Liczbę taką nietrudno wyznaczyć biorąc dla przykładu:

$$T = \sum_p \sum_l H(T_{pl}).$$

Przyjmuje się przy tym, że każda z liczb $H(T_{pl})$ jest całkowitą. Jak wynika z przykładów przedstawionych w rozdziale poprzednim nie jest to zbyt silne ograniczenie. Wystarczy bowiem zauważyć, że jeśli tylko żadna z liczb nie jest niewymierna to poprzez zmianę jednostki uzyskać można ich całkowitoliczbowość, a niewymierność liczb $H(T_{pl})$ jest zapewniona w sposób naturalny z warunków zadania. Następnie wyznacza się największy wspólny dzielnik liczb $H(T_{pl})$, oznaczmy go r^* .

Liczba T na osi wartości funkcjonału $H^{1/}$ wyznacza pewien punkt. Odcinek $[0, T]$ dzieli się na t jednakowych odcinków o długości r^* każdy. Kolejne odcinki otrzymane z tego podziału ponumerujemy liczbami od 0 do $t-1$. Następnie przyjmuje się za jednostkę r^* i w tych jednostkach wyraża się liczby $H(T_{p1})$.

Jako pierwszy przedstawimy model Bormana.

3.3.1. Model Bowmana

Model ten skonstruowany został przy następujących założeniach:

$$- \forall s \in N \quad \Pi(s) = \{M \times 1\}$$

lub inaczej:

$$- \forall l \in K \quad \bigcup_M \omega(m, l) = \{s\} = \bigcap_M \omega(m, l).$$

- dla każdego $p \in M$ dana jest funkcja sterująca α_{pK} .

Jest to więc zadanie typu D , sformułowane w rozdziale pierwszym. Wprowadzimy teraz zmienną x_{spr} zdefiniowaną jak następuje:

$$x_{spr} = \begin{cases} 1 & \text{jeśli operator } T_{p1} \text{ działa w warstwie} \\ & \text{s-tej i w r-tej jednostce.} \\ 0 & \text{w przeciwnym przypadku.} \end{cases}$$

Zwróćmy od razu uwagę na to, iż wobec pierwszego założenia zmienna x_{spr} może być równa 1 tylko wówczas, gdy:

^{1/} patrz diagram Gantta.

$$\omega(p, l) = s$$

lub inaczej

$$T_{pl} \in Z_s$$

Innymi słowy wprowadzimy tylko takie zmienne x_s , które spełniają powyższy warunek.

Z założenia tego wynika jeszcze i to, że działanie zbioru operatorów jest działaniem k -równoległym.

Po to by prościej przedstawić warunki ograniczające modelu przyjmijmy następującą umowę:

$$T_{pl} \in Z_s$$

$$T_{p, l-1} \in Z_{s-1}$$

$$\alpha_{pK}^{-1}(T_{p, l-1}) = \alpha_{pK}^{-1}(T_{pl}) - 1.$$

Umowa powyższa znaczy tyle, że warstwy operatorów dla każdego p są ponumerowane zgodnie z wartościami funkcji α_{pK}^{-1} .

Warunki modelu przyjmą teraz następującą postać:

B.1

$$H(T_{p, l-1}) x_{spr} \leq \sum_{d=0}^{r-1} x_{s-1, p, d}$$

$$p = 1, 2, \dots, m$$

$$s = 2, 3, \dots, k$$

$$r = H(T_{p, l-1}), H(T_{p, l-1})+1, \dots, T - H(T_{p, l-1}).$$

B.2

$$\sum_{r=0}^{t-1} x_{spr} = H(T_{p1})$$

$$p = 1, 2, \dots, m$$

$$s = 1, 2, \dots, k.$$

B.3

$$H(T_{p1})x_{spr} - H(T_{p1})x_{s,p,r+1} + \sum_{d=r+2}^{t-1} x_{spd} \leq H(T_{p1}).$$

$$p = 1, 2, \dots, m$$

$$s = 1, 2, \dots, k$$

$$r = 0, 1, \dots, T-H(T_{p1}).$$

B.4

$$\sum_{p=1}^m x_{spr} \leq 1.$$

$$s = 1, 2, \dots, k$$

$$r = 0, 1, \dots, t-1.$$

Przy tak sformułowanych warunkach należy wyznaczyć minimalną wartość T . Omówmy teraz znaczenie poszczególnych ograniczeń powyższego modelu. Warunek B.1 zabezpiecza nienaruszalność wartości żadnej z funkcji α_{pK} . Wystarczy bowiem zauważyć, że jeśli $x_{spr} = 1$, to znaczy operator T_{p1} w s -tej warstwie działa w r -tej jednostce to w warstwie poprzedniej musiał działać operator $T_{p,1-1}$ i działanie to miało trwać przez $H(T_{p,1-1})$ jednostek.

Warunek B.2 jest równoznaczny z żądaniem by dla każdego operatora T_{p1} w rozwiązaniu przeznaczyć $H(T_{p1})$ jednostek.

Warunek B.3 wynika z założenia, że zbiór operatorów jest dobrze określony, tzn. jeśli działanie operatora rozpoczęło się w jednostce r -tej to musi być zakończone po upływie $H(T_{p1})$ jednostek.

Ostatni z warunków wynika z założenia o szeregowym działaniu operatorów w warstwie. W danej warstwie i danej jednostce nie mogą działać dwa operatory.

Jak wygląda zastosowanie tego modelu w zadaniu praktycznym zobaczymy na następującym przykładzie.

Przykład

Dane:

zbiór operatorów: $\{T_{11}, T_{12}, T_{21}, T_{22}\}$

		1,1	1,2	2,1	2,2
H(0)=	1	4		5	
	2		3		6

$$\mathcal{L}_{1K}(1) = T_{11}$$

$$\mathcal{L}_{2K}(1) = T_{22}$$

$$\mathcal{L}_{1K}(2) = T_{12}$$

$$\mathcal{L}_{2K}(2) = T_{21}$$

czyli:

$$T_{11} \in Z_1 \quad \text{i} \quad T_{21} \in Z_1$$

$$T_{12} \in Z_2 \quad \text{i} \quad T_{22} \in Z_2$$

Przyjmijmy $T = 13$

Zmienne x_{11r} dotyczyć będą warstwy pierwszej i operatora T_{11} ,

Zmienne x_{12r} dotyczyć będą warstwy pierwszej, operatora T_{21} ,

Zmienne x_{21r} dotyczyć będą warstwy drugiej, operatora T_{12} ,

Zmienne x_{22r} dotyczyć będą warstwy drugiej, operatora T_{22}

zaś $r = 0, 1, \dots, 12$.

Zadanie będzie teraz miało następującą postać:

Wyznaczyć zmienne x_{spr} , które spełniają następujące warunki:

B.1

$$4x_{214} \leq x_{110} + x_{111} + x_{112} + x_{113}$$

$$4x_{215} \leq x_{110} + x_{111} + \dots + x_{114}$$

$$4x_{216} \leq x_{110} + x_{111} + \dots + x_{115}$$

$$4x_{217} \leq x_{110} + x_{111} + \dots + x_{116}$$

$$4x_{218} \leq x_{110} + x_{111} + \dots + x_{117}$$

$$4x_{219} \leq x_{110} + x_{111} + \dots + x_{118}$$

$$6x_{126} \leq x_{220} + x_{221} + x_{222} + x_{223} + x_{224} + x_{225}$$

$$6x_{127} \leq x_{220} + x_{221} + \dots + x_{225} + x_{226}$$

B.2

$$x_{110} + x_{111} + \dots + x_{11,12} = 4$$

$$x_{120} + x_{121} + \dots + x_{12,12} = 5$$

$$x_{210} + x_{211} + \dots + x_{21,12} = 3$$

$$x_{220} + x_{221} + \dots + x_{22,12} = 6$$

B.3

$$\begin{aligned} 4x_{110} - 4x_{111} + x_{112} + x_{113} + \cdot \cdot \cdot + x_{11,12} &\leq 4 \\ 4x_{111} - 4x_{112} + x_{113} + x_{114} + \cdot \cdot \cdot + x_{11,12} &\leq 4 \\ 4x_{112} - 4x_{113} + x_{114} + x_{115} + \cdot \cdot \cdot + x_{11,12} &\leq 4 \\ 4x_{113} - 4x_{114} + x_{115} + x_{116} + \cdot \cdot \cdot + x_{11,12} &\leq 4 \\ 4x_{114} - 4x_{115} + x_{116} + x_{117} + \cdot \cdot \cdot + x_{11,12} &\leq 4 \\ 4x_{115} - 4x_{116} + x_{117} + x_{118} + \cdot \cdot \cdot + x_{11,12} &\leq 4 \\ 4x_{116} - 4x_{117} + x_{118} + x_{119} + \cdot \cdot \cdot + x_{11,12} &\leq 4 \\ 4x_{117} - 4x_{118} + x_{119} + x_{11,10} + \cdot \cdot \cdot + x_{11,12} &\leq 4 \\ 4x_{118} - 4x_{119} + x_{11,10} + x_{11,11} + x_{11,12} &\leq 4 \\ 4x_{119} - 4x_{11,10} + x_{11,11} + x_{11,12} &\leq 4 \\ \\ 5x_{120} - 5x_{121} + x_{122} + x_{123} + \cdot \cdot \cdot + x_{12,12} &\leq 5 \\ 5x_{121} - 5x_{122} + x_{123} + x_{124} + \cdot \cdot \cdot + x_{12,12} &\leq 5 \\ 5x_{122} - 5x_{123} + x_{124} + x_{125} + \cdot \cdot \cdot + x_{12,12} &\leq 5 \\ 5x_{123} - 5x_{124} + x_{125} + x_{126} + \cdot \cdot \cdot + x_{12,12} &\leq 5 \\ 5x_{124} - 5x_{125} + x_{126} + x_{127} + \cdot \cdot \cdot + x_{12,12} &\leq 5 \\ 5x_{125} - 5x_{126} + x_{127} + x_{128} + \cdot \cdot \cdot + x_{12,12} &\leq 5 \\ 5x_{126} - 5x_{127} + x_{128} + x_{129} + \cdot \cdot \cdot + x_{12,12} &\leq 5 \\ 5x_{127} - 5x_{128} + x_{129} + x_{12,10} + \cdot \cdot \cdot + x_{12,12} &\leq 5 \\ 5x_{128} - 5x_{129} + x_{12,10} + x_{12,11} + x_{12,12} &\leq 5 \end{aligned}$$

$$\begin{aligned}
 3x_{210} + 3x_{211} + x_{212} + x_{213} + \dots + x_{21,12} &\leq 3 \\
 3x_{211} - 3x_{212} + x_{213} + x_{214} + \dots + x_{21,12} &\leq 3 \\
 3x_{212} - 3x_{213} + x_{214} + x_{215} + \dots + x_{21,12} &\leq 3 \\
 3x_{213} - 3x_{214} + x_{215} + x_{216} + \dots + x_{21,12} &\leq 3 \\
 3x_{214} - 3x_{215} + x_{216} + x_{217} + \dots + x_{21,12} &\leq 3 \\
 3x_{215} - 3x_{216} + x_{217} + x_{218} + \dots + x_{21,12} &\leq 3 \\
 3x_{216} - 3x_{217} + x_{218} + x_{219} + \dots + x_{21,12} &\leq 3 \\
 3x_{217} - 3x_{218} + x_{219} + x_{21,10} + \dots + x_{21,12} &\leq 3 \\
 3x_{218} - 3x_{219} + x_{21,10} + x_{21,11} + x_{21,12} &\leq 3 \\
 3x_{219} - 3x_{21,10} + x_{21,11} + x_{21,12} &\leq 3 \\
 3x_{21,10} - 3x_{21,11} + x_{21,12} &\leq 3 \\
 \\
 6x_{220} - 6x_{221} + x_{222} + x_{223} + \dots + x_{22,12} &\leq 6 \\
 6x_{221} - 6x_{222} + x_{223} + x_{224} + \dots + x_{22,12} &\leq 6 \\
 6x_{222} - 6x_{223} + x_{224} + x_{225} + \dots + x_{22,12} &\leq 6 \\
 6x_{223} - 6x_{224} + x_{225} + x_{226} + \dots + x_{22,12} &\leq 6 \\
 6x_{224} - 6x_{225} + x_{226} + x_{227} + \dots + x_{22,12} &\leq 6 \\
 6x_{225} - 6x_{226} + x_{227} + x_{228} + \dots + x_{22,12} &\leq 6 \\
 6x_{226} - 6x_{227} + x_{228} + x_{229} + \dots + x_{22,12} &\leq 6 \\
 6x_{227} - 6x_{228} + x_{229} + x_{22,10} + x_{22,11} + x_{22,12} &\leq 6
 \end{aligned}$$

B.4

$$\begin{array}{ll} x_{110} + x_{120} \leq 1 & x_{210} + x_{220} \leq 1 \\ x_{111} + x_{121} \leq 1 & x_{211} + x_{221} \leq 1 \\ x_{112} + x_{122} \leq 1 & x_{212} + x_{222} \leq 1 \\ x_{113} + x_{123} \leq 1 & x_{213} + x_{223} \leq 1 \\ x_{114} + x_{124} \leq 1 & x_{214} + x_{224} \leq 1 \\ x_{115} + x_{125} \leq 1 & x_{215} + x_{225} \leq 1 \\ x_{116} + x_{126} \leq 1 & x_{216} + x_{226} \leq 1 \\ x_{117} + x_{127} \leq 1 & x_{217} + x_{227} \leq 1 \\ x_{118} + x_{128} \leq 1 & x_{218} + x_{228} \leq 1 \\ x_{119} + x_{129} \leq 1 & x_{219} + x_{229} \leq 1 \\ x_{11,10} + x_{12,10} \leq 1 & x_{21,10} + x_{22,10} \leq 1 \\ x_{11,11} + x_{12,11} \leq 1 & x_{21,11} + x_{22,11} \leq 1 \\ x_{11,12} + x_{12,12} \leq 1 & x_{21,12} + x_{22,12} \leq 1 \end{array}$$

i minimalizowały wartość T . Co jest równoznaczne z minimalizacją t .

Rozpatrzmy dwa rozwiązania przedstawione w poniższej tabeli:

r	s,p	I				II			
		1,1	1,2	2,1	2,2	1,1	1,2	2,1	2,2
0		1	0	0	1	0	0	0	1
1		1	0	0	1	0	0	0	1
2		1	0	0	1	1	0	0	1
3		1	0	0	1	1	0	0	1
4		0	0	0	1	1	0	0	1
5		0	0	0	1	1	0	0	1
6		0	1	1	0	0	0	0	0
7		0	1	1	0	0	0	0	0
8		0	1	1	0	0	1	0	0
9		0	1	0	0	0	1	0	0
10		0	1	0	0	0	1	1	0
11		0	0	0	0	0	1	1	0
12		0	0	0	0	0	1	1	0
13		0	0	0	0	0	0	0	0

Łatwo sprawdzić, że w powyższej tabeli przedstawione są rozwiązania dopuszczalne zadania tzn., takie, które spełniają układ równań i nierówności B.1 - B.4 . Jednak dla pierwszego wartość T wynosi 11, dla drugiego zaś 13, co oznacza, że pierwsze rozwiązanie jest lepsze.

3.3.2. Model Jankowskiej-Zorychty I

Model ten zbudowany został przy następujących założeniach:

- dla każdego $p \in M$ dana jest funkcja sterująca α_{pK} ,
- $\exists s \in N \wedge D \subset K$; $\bar{U}(s) = \{M \times D\}$,
- $\exists p \in M, l \in K$; $\omega(p, l) = N_{pl} \wedge \text{card}(N_{pl}) > 1$.

Ostatnie założenie można sformułować nieco inaczej:

$$\exists p \in M \wedge l \in K ; \text{card}(\mathcal{T}_{pl}) > 1.$$

Ponadto przyjmuje się, że:

$$H(\mathcal{T}_{pls_1}) = H(\mathcal{T}_{pls_2}) \quad \text{dla każdego } s_1, s_2 \in \omega(p, l).$$

Gdyby nie to ostatnie założenie, zadanie to byłoby zadaniem typu C, sformułowanym w pierwszym rozdziale. Jeśli się zaś uchyli to założenie, to proponowany model musi ulec pewnej modyfikacji.

Podobnie jak i w modelu poprzednim przyjmijmy następującą umowę:

$$\alpha_{pK}^{-1}(\mathcal{T}_{p, l-1}) = \alpha_{pK}^{-1}(\mathcal{T}_{pl})^{-1}.$$

Warunki zadania wówczas przyjmują następującą postać:

J.1

$$H(\mathcal{T}_{p, l-1, c})^{x_{\text{spr}}} \leq \sum_{c \in \omega(p, l-1)} \sum_{d=0}^{r-1} x_{\text{cpd}},$$

gdzie:

$$s \in \omega(p, l)$$

$$p = 1, 2, \dots, m$$

$$l = 1, 2, \dots, k_p$$

$$r = H(T_{pls}), H(T_{pls}) + 1, \dots, T - H(T_{pls}).$$

J.2

$$\sum_s \sum_{r=0}^{t-1} x_{spr} = H(T_{pls})$$

$$p = 1, 2, \dots, m$$

$$s \in N_{pl}.$$

J.3

$$H(T_{pls}) x_{s,p,r+1} - x_{spr} \leq \sum_{d=1}^{H(T_{pls})} x_{sp,r+d}.$$

$$r = 0, 1, \dots, T - H(T_{pls})$$

$$s \in N_{pl}$$

$$p = 1, 2, \dots, m.$$

J.4

$$\sum_p x_{spr} \leq 1$$

$$s \in N_{pl},$$

$$r = 0, 1, \dots, t-1.$$

Znaczenie poszczególnych ograniczeń jest takie samo jak w modelu Bowmana.

Dla demonstracji tego modelu posłużymy się przykładem z poprzedniego zadania uzupełniając dane tak, by spełniały założenia wymienione na wstępie.

Przykład:

Dane:

rodzina operatorów $\{T_{11}, T_{12}, T_{21}, T_{22}\}$

$$H(\theta) =$$

	1,1	1,2	2,1	2,2
1	4		5	6
2	4	3		6

$$\alpha_{1K}(1) = T_{11}$$

$$\alpha_{2K}(1) = T_{21}$$

$$\alpha_{1K}(2) = T_{12}$$

$$\alpha_{2K}(2) = T_{22}$$

Mamy zatem:

$$T_{11} = \{T_{111}, T_{112}\}$$

$$T_{12} = \{T_{122}\}$$

$$T_{21} = \{T_{211}\}$$

$$T_{22} = \{T_{221}, T_{222}\}.$$

Przyjmijmy jak poprzednio $T = 13$. W powyższym przykładzie mamy potencjalną możliwość zaliczenia do warstwy pierwszej operatory:

$T_{111}, T_{211}, T_{221}$, do drugiej zaś operatory: $T_{112}, T_{122}, T_{222}$.

Zmienne x_{11r} dotyczyć będą warstwy pierwszej i operatora T_{111} ,

zmienne x_{12r} dotyczyć będą warstwy pierwszej i operatora T_{211} ,

zmienne x_{13r} dotyczyć będą warstwy pierwszej i operatora T_{221} ,

zmienne x_{21r} dotyczyć będą warstwy drugiej i operatora T_{112} ,

zmienne x_{22r} dotyczyć będą warstwy drugiej i operatora T_{122} ,

zmienne x_{23r} dotyczyć będą warstwy drugiej i operatora T_{222} .

Zadanie teraz jest następujące:

Wyznaczyć zmienne x_{spr} ; $s=1,2$, $p=1,2,3$, $r=0,1,2,\dots,13$,
które minimalizowałyby T i spełniały następujące warunki:

J.1

$$4x_{224} \leq x_{110} + x_{111} + \dots + x_{113} + x_{210} + \dots + x_{213}$$

$$4x_{225} \leq x_{110} + \dots + x_{114} + x_{210} + \dots + x_{214}$$

⋮

$$4x_{229} \leq x_{110} + \dots + x_{118} + x_{210} + \dots + x_{218}$$

$$5x_{135} \leq x_{120} + \dots + x_{124}$$

$$5x_{136} \leq x_{120} + \dots + x_{125}$$

⋮

$$5x_{138} \leq x_{120} + \dots + x_{127}$$

$$5x_{235} \leq x_{120} + \dots + x_{124}$$

$$5x_{236} \leq x_{120} + \dots + x_{125}$$

⋮

$$5x_{238} \leq x_{120} + \dots + x_{127}$$

J.2

$$x_{110} + x_{111} + \dots + x_{11,13} + x_{210} + x_{211} + \dots + x_{21,13} = 4$$

$$x_{220} + x_{221} + \dots + x_{22,13} = 3$$

$$x_{120} + x_{121} + \dots + x_{12,13} = 5$$

$$x_{130} + x_{131} + \dots + x_{13,13} + x_{230} + x_{231} + \dots + x_{23,13} = 6$$

J.3

$$4 (x_{111} - x_{110}) \leq x_{111} + x_{112} + x_{113} + x_{114}$$

$$4 (x_{112} - x_{111}) \leq x_{112} + x_{113} + x_{114} + x_{115}$$

⋮

$$4 (x_{11,10} - x_{119}) \leq x_{11,10} + x_{11,11} + x_{11,12} + x_{11,13}$$

$$5 (x_{121} - x_{120}) \leq x_{121} + x_{122} + x_{123} + x_{124} + x_{125}$$

⋮

$$5 (x_{129} - x_{128}) \leq x_{129} + x_{12,10} + x_{12,11} + x_{12,12} + x_{12,13}$$

$$6 (x_{131} - x_{130}) \leq x_{131} + x_{132} + x_{133} + x_{134} + x_{135} + x_{136}$$

⋮

$$6 (x_{138} - x_{137}) \leq x_{138} + x_{139} + x_{13,10} + x_{13,11} + x_{13,12} + x_{13,13}$$

$$4 (x_{211} - x_{210}) \leq x_{211} + x_{212} + x_{213} + x_{214}$$

⋮

$$4 (x_{21,10} - x_{219}) \leq x_{21,10} + x_{21,11} + x_{21,12} + x_{21,13}$$

$$3 (x_{221} - x_{220}) \leq x_{221} + x_{222} + x_{223}$$

⋮

$$3 (x_{21,11} - x_{21,10}) \leq x_{21,11} + x_{21,12} + x_{21,13}$$

$$6 (x_{231} - x_{230}) \leq x_{231} + x_{232} + x_{233} + x_{234} + x_{235} + x_{236}$$

⋮

$$6 (x_{238} - x_{237}) \leq x_{238} + x_{239} + x_{23,10} + x_{23,11} + x_{23,12} + x_{23,13}$$

J.4

$$x_{110} + x_{120} + x_{130} \leq 1$$

$$x_{111} + x_{121} + x_{131} \leq 1$$

⋮

$$x_{11,13} + x_{12,13} + x_{13,13} \leq 1$$

$$x_{210} + x_{220} + x_{230} \leq 1$$

⋮

$$x_{21,13} + x_{22,13} + x_{23,13} \leq 1$$

Rozpatrzmy następujący układ liczb:

s,p r	1,1	1,2	1,3	2,1	2,2	2,3
0	0	1	0	1	0	0
1	0	1	0	1	0	0
2	0	1	0	1	0	0
3	0	1	0	1	0	0
4	0	1	0	0	1	0
5	0	0	1	0	1	0
6	0	0	1	0	1	0
7	0	0	1	0	0	0
8	0	0	1	0	0	0
9	0	0	1	0	0	0
10	0	0	1	0	0	0
11	0	0	0	0	0	0
12	0	0	0	0	0	0
13	0	0	0	0	0	0

Łatwo sprawdzić, że układ ten jest dopuszczalnym rozwiązaniem naszego zadania.

3.3.3. Model Jankowskiej-Zorychty II

Drugi model zaproponowany przez autorkę, zbudowany jest przy tych samych założeniach dla jakich zbudowano model Bowmana. Warunki tego modelu mają następującą postać:

JII.1

$$H(T_{p, l-1}) x_{spr} \leq \sum_{d=0}^{r-1} x_{s-1, p, d}$$

$$p = 1, 2, \dots, m$$

$$s = 1, 2, \dots, k$$

$$r = H(T_{p, l-1}), H(T_{p, l-1})+1, \dots, T-H(T_{p, l-1}) .$$

JII.2

$$\sum_{r=0}^{t-1} x_{spr} = H(T_{p1})$$

$$p = 1, 2, \dots, m$$

$$s = 1, 2, \dots, k .$$

JII.3

$$H(T_{p1})(x_{s, p, r+1} - x_{spr}) \leq \sum_{d=1}^{H(T_{p1})} x_{s, p, r+d}$$

$$p = 1, 2, \dots, m$$

$$s = 1, 2, \dots, k$$

$$r = 0, 1, 2, \dots, T-H(T_{p1}) .$$

JII.4

$$\sum_{p=1}^m x_{spr} \leq 1$$

$$s = 1, 2, \dots, k$$

$$r = 0, 1, \dots, t-1 .$$

Jak widać warunki JII.1, JII.2, JII.4 są identyczne jak warunki B.1, B.2, B.4 w modelu Bowmana. Warunek JII.3 jest natomiast szczególnym przypadkiem warunku J.3, który to przypadek otrzymuje się uwzględniając założenia modelu JII.

Jak funkcjonuje ten model pokażemy na przykładzie zadania rozpatrywanego przy modelu Bowmana, przy czym wypiszemy tylko warunki JII.3 gdyż pozostałe są identyczne z warunkami B.1, B.2, B.4.

JII.3

$$4(x_{111} - x_{110}) \leq x_{111} + x_{112} + x_{113} + x_{114}$$

$$4(x_{112} - x_{111}) \leq x_{112} + x_{113} + x_{114} + x_{115}$$

$$4(x_{113} - x_{112}) \leq x_{113} + x_{114} + x_{115} + x_{116}$$

$$4(x_{114} - x_{113}) \leq x_{114} + x_{115} + x_{116} + x_{117}$$

$$4(x_{115} - x_{114}) \leq x_{115} + x_{116} + x_{117} + x_{118}$$

$$4(x_{116} - x_{115}) \leq x_{116} + x_{117} + x_{118} + x_{119}$$

$$4(x_{117} - x_{116}) \leq x_{117} + x_{118} + x_{119} + x_{11,10}$$

$$4(x_{118} - x_{117}) \leq x_{118} + x_{119} + x_{11,10} + x_{11,11}$$

$$4(x_{119} - x_{118}) \leq x_{119} + x_{11,10} + x_{11,11} + x_{11,12}$$

$$4(x_{11,10} - x_{119}) \leq x_{11,10} + x_{11,11} + x_{11,12} + x_{11,13}$$

$$\begin{aligned} 5(x_{121} - x_{120}) &\leq x_{121} + x_{122} + x_{123} + x_{124} + x_{125} \\ 5(x_{122} - x_{121}) &\leq x_{122} + x_{123} + x_{124} + x_{125} + x_{126} \\ 5(x_{123} - x_{122}) &\leq x_{123} + x_{124} + x_{125} + x_{126} + x_{127} \\ 5(x_{124} - x_{123}) &\leq x_{124} + x_{125} + x_{126} + x_{127} + x_{128} \\ 5(x_{125} - x_{124}) &\leq x_{125} + x_{126} + x_{127} + x_{128} + x_{129} \\ 5(x_{126} - x_{125}) &\leq x_{126} + x_{127} + x_{128} + x_{129} + x_{12,10} \\ 5(x_{127} - x_{126}) &\leq x_{127} + x_{128} + x_{129} + x_{12,10} + x_{12,11} \\ 5(x_{128} - x_{127}) &\leq x_{128} + x_{129} + x_{12,10} + x_{12,11} + x_{12,12} \\ 5(x_{239} - x_{128}) &\leq x_{129} + x_{12,10} + x_{12,11} + x_{12,12} + x_{12,13} \end{aligned}$$

$$\begin{aligned} 3(x_{211} - x_{210}) &\leq x_{211} + x_{212} + x_{213} \\ 3(x_{212} - x_{211}) &\leq x_{212} + x_{213} + x_{214} \\ 3(x_{213} - x_{212}) &\leq x_{213} + x_{214} + x_{215} \\ 3(x_{214} - x_{213}) &\leq x_{214} + x_{215} + x_{216} \\ 3(x_{215} - x_{214}) &\leq x_{215} + x_{216} + x_{217} \\ 3(x_{216} - x_{215}) &\leq x_{216} + x_{217} + x_{218} \\ 3(x_{217} - x_{216}) &\leq x_{217} + x_{218} + x_{219} \\ 3(x_{218} - x_{217}) &\leq x_{218} + x_{219} + x_{21,10} \\ 3(x_{219} - x_{218}) &\leq x_{219} + x_{21,10} + x_{21,11} \\ 3(x_{21,10} - x_{219}) &\leq x_{21,10} + x_{21,11} + x_{21,12} \\ 3(x_{21,11} - x_{21,10}) &\leq x_{21,11} + x_{21,12} + x_{21,13} \end{aligned}$$

$$6(x_{221} - x_{220}) \leq x_{221} + x_{222} + x_{223} + x_{224} + x_{225} + x_{226}$$

$$6(x_{222} - x_{221}) \leq x_{222} + x_{223} + x_{224} + x_{225} + x_{226} + x_{227}$$

$$6(x_{223} - x_{222}) \leq x_{223} + x_{224} + x_{225} + x_{226} + x_{227} + x_{228}$$

$$6(x_{224} - x_{223}) \leq x_{224} + x_{225} + x_{226} + x_{227} + x_{228} + x_{229}$$

$$6(x_{225} - x_{224}) \leq x_{225} + x_{226} + x_{227} + x_{228} + x_{229} + x_{22,10}$$

$$6(x_{226} - x_{225}) \leq x_{226} + x_{227} + x_{228} + x_{229} + x_{22,10} + x_{22,11}$$

$$6(x_{227} - x_{226}) \leq x_{227} + x_{228} + x_{229} + x_{22,10} + x_{22,11} + x_{22,12}$$

$$6(x_{228} - x_{227}) \leq x_{228} + x_{229} + x_{22,10} + x_{22,11} + x_{22,12} + x_{22,13}$$

Rozwiązania dopuszczalne podane przy modelu Bowmana spełniają również powyższe warunki.

3.3.4. Model Manne'a

Model ten zbudowany został przy tych samych założeniach jak i model Bowmana czy też Jankowskiej-Zorychty II, wobec czego założeń tych nie będziemy przytaczać.

Autor wprowadza zmienną x_{sp} , którą definiuje jako numer jednostki, w której operator T_{p1} rozpoczyna działanie w s -tej warstwie.

x_{sp} zatem mogą przyjmować wartości $0, 1, \dots, t-1$.

Warunki modelu przyjmują następującą postać:

M.1

$$x_{sp} - x_{s-1,p} \geq H(T_{p,1-1})$$

$$p = 1, 2, \dots, m$$

$$s = 2, 3, \dots, k.$$

M.2

$$x_{sp} - x_{sr} \geq H(T_{r1}) \quad \text{lub} \quad x_{sr} - x_{sp} \geq H(T_{p1})$$

$$s = 1, 2, \dots, k$$

$$r = 1, 2, \dots, m$$

$$p = 1, 2, \dots, m .$$

Ponieważ dla zadania programowania liniowego nie można wprowadzić warunku alternatywnego typu M.2, autor wprowadza zmienną y_{spr} równą jeden lub zero.

Warunek M.2 można zapisać przy użyciu tych zmiennych w postaci dwóch następujących:

$$M.2' \quad (T + H(T_{p1}))y_{spr} + (x_{sp} - x_{sr}) \geq H(T_{p1})$$

$$M.2'' \quad (T + H(T_{r1})) (1 - y_{spr}) + (x_{sr} - x_{sp}) \geq H(T_{r1})$$

dla obu warunków:

$$s = 1, 2, \dots, k$$

$$p = 1, 2, \dots, m$$

$$r = 1, 2, \dots, m .$$

M.3

$$x_{sp} + H(T_{p1}) \leq T'$$

$$p = 1, 2, \dots, m$$

$$s = 1, 2, \dots, k$$

$$T' \leq T .$$

Omówmy teraz znaczenie poszczególnych ograniczeń w powyższym modelu. Warunek M.1 zabezpiecza by rozwiązanie nie naruszało wartości funkcji sterującej \mathcal{L}_{pK} , gdyż przypomnijmy, że:

$$T_{p,l-1} \in Z_{s-1}$$

$$T_{p1} \in Z_s$$

$$\alpha_{pK}^{-1}(T_{p,l-1}) = \alpha_{pK}^{-1}(T_{p1}) - 1.$$

Warunek M.2 żąda aby operatory należące do jednej warstwy działały szeregowo, przy czym momenty rozpoczęcia działania operatorów T_{p1} i T_{r1} były odległe od siebie co najmniej o wartość $H(T_{p1})$ lub $H(T_{r1})$ w zależności od tego, który z nich ma działać wcześniej.

Omówmy teraz warunki M.2' i M.2'', oraz sens zmiennej y_{spr} .

Zwróćmy uwagę na oczywistą nierówność:

$$|x_{sp} - x_{sr}| < T$$

wynikającą z określenia T . Jeśli $H(T_{p1}) \neq 0$ i $H(T_{r1}) \neq 0$ to:

$$x_{sp} - x_{sr} \neq 0$$

gdyż gdyby było inaczej to zmienna y_{spr} w M.2' musiałaby przyjąć wartość jeden, zaś w M.2'' wartość zero. Co zresztą byłoby sprzeczne z założeniem o szeregowym działaniu operatorów w warstwie.

$x_{sp} - x_{sr}$ mogłoby być równe zero tylko wówczas gdy jedna z liczb: $H(T_{p1})$ lub $H(T_{r1})$ byłaby równa zero.

Jeżeli:

$$x_{sp} - x_{sr} > 0$$

to y_{spr} w M.2' może przyjąć wartość zero lub jeden, zaś w M.2'' tylko wartość zero. Tak więc jedyną możliwą wartością

jaką może przyjąć zmienna y_{spr} w tym przypadku byłoby zero. Jeżeli natomiast:

$$x_{sp} - x_{sr} < 0$$

to z tych samych powodów zmienna y_{spr} może przyjąć wartość jeden. Z powyższych rozważań wynika, że:

$$y_{spr} = \begin{cases} 1 & \text{jeśli w } s\text{-tej warstwie operator } T_{pl} \\ & \text{poprzedza operator } T_{r1}. \\ 0 & \text{w przeciwnym przypadku.} \end{cases}$$

Warunek M.3 jest jednocześnie kryterium optymalizacji, gdyż żąda by $H(T_{MK_{\alpha}}^{\beta}) < T'$ i T' należy zminimalizować.

Zobaczmy jak funkcjonuje model Manne'a w konkretnej sytuacji na przykładzie zadania rozpatrywanego przy modelu Bowmana. Przypomnijmy ten przykład.

Przykład

Dany jest zbiór operatorów $\{T_{11}, T_{12}, T_{21}, T_{22}\}$

$H(\theta) =$

	1,1	1,2	2,1	2,2
1	4		5	
2		3		6

$$\alpha_{1K}(1) = T_{11}$$

$$\alpha_{2K}(1) = T_{22}$$

$$\alpha_{1K}(2) = T_{12}$$

$$\alpha_{2K}(2) = T_{21}$$

Przyjmijmy $T = 13$.

Zmienna x_{11} dotyczy pierwszej warstwy i operatora T_{11} .

Zmienna x_{12} dotyczy pierwszej warstwy i operatora T_{21} .

Zmienna x_{21} dotyczy drugiej warstwy i operatora T_{12} .

Zmienna x_{22} dotyczy drugiej warstwy i operatora T_{22} .

Dla naszego zadania warunki modelu są następujące:

M.1

$$x_{21} - x_{11} \geq 4$$

$$x_{12} - x_{22} \geq 6$$

M.2'

$$(13 + 5) y_{112} + (x_{11} - x_{12}) \geq 5$$

$$(13 + 4) y_{121} + (x_{12} - x_{11}) \geq 4$$

$$(13 + 3) y_{221} + (x_{22} - x_{21}) \geq 3$$

$$(13 + 6) y_{212} + (x_{21} - x_{22}) \geq 6$$

M.2''

$$(13 + 4) (1 - y_{112}) + (x_{12} - x_{11}) \geq 4$$

$$(13 + 5) (1 - y_{121}) + (x_{11} - x_{12}) \geq 5$$

$$(13 + 6) (1 - y_{221}) + (x_{21} - x_{22}) \geq 6$$

$$(13 + 3) (1 - y_{212}) + (x_{22} - x_{21}) \geq 3$$

M.3

$$x_{11} + 4 \leq T' \leq 13$$

$$x_{12} + 5 \leq T' \leq 13$$

$$x_{21} + 3 \leq T' \leq 13$$

$$x_{22} + 6 \leq T' \leq 13$$

Zobaczmy teraz dwa rozwiązania, które są zresztą identyczne z rozwiązaniami dopuszczalnymi modelu Bowmana:

I.

$$\begin{array}{ll} x_{11} = 0 & y_{121} = 0 \\ x_{12} = 6 & y_{112} = 1 \\ x_{21} = 6 & y_{221} = 1 \\ x_{22} = 0 & y_{212} = 0 \end{array}$$

II.

$$\begin{array}{ll} x_{11} = 2 & y_{121} = 0 \\ x_{12} = 8 & y_{112} = 1 \\ x_{21} = 10 & y_{221} = 1 \\ x_{22} = 0 & y_{212} = 0. \end{array}$$

Jak łatwo sprawdzić obydwa rozwiązania są dopuszczalnymi sprawdzają ograniczenia M.1 - M.3 , przy czym pierwsze z nich daje mniejszą wartość T^* . Dla pierwszego bowiem mamy $T^* = 11$ dla drugiego zaś $T^* = 13$.

Porównując prezentowane modele na przykładzie bardzo prostego zadania można stwierdzić, że model Manne'a jest naj-
ekonomiczniejszy w sensie rozmiarów otrzymanego zadania. Są
taki można znaleźć w literaturze^{1/}.

Jak już powiedziano poprzednio model Manne'a opisuje zada-
nie typu D sformułowane w rozdziale pierwszym. Jednak jak
wynika z rozważań przeprowadzonych w rozdziale drugim w kon-
kretnych sytuacjach spotyka się zadania, które mogą być od-
mianami zadań wyszczególnionych w rozdziale pierwszym.

^{1/}patrz Korbut A.A., Finkelsztein J.J. "Programowanie
dyskretne", str.66

Model Manne'a po pewnych modyfikacjach dokonanych przez autora modelu obejmuje wiele z tych zadań. Omówimy je kolejno. Przypuśćmy, że dla pewnego podzbioru T_{pK} zbioru operatorów T_{MK} dane są dwie funkcje sterujące α_{pK}^1 i α_{pK}^2 ,

przy czym:

$$\alpha_{pK}^1(i) = \alpha_{pK}^2(i)$$

dla

$$i = 1, 2, \dots, c-1, c+1, \dots, k$$

$$\alpha_{pK}^1(c) = T_{pl_1}$$

$$\alpha_{pK}^2(c) = T_{pl_2}^{1/}$$

Wówczas warunek M.1 zamienia się na dwa warunki:

M.1a

$$x_{sp} - x_{s_1p} \geq H(T_{pl_1})$$

$$x_{sp} - x_{s_2p} \geq H(T_{pl_2})$$

gdzie:

$$\alpha_{pK}^{1-1}(T_{pl}) = \alpha_{pK}^{1-1}(T_{pl_1}) - 1$$

1/ Odpowiada to sytuacji gdy detal p może być przed operacją 1 poddany operacji l_1 lub l_2 .

$$\mathcal{L}_{pK}^{2^{-1}}(T_{p1}) = \mathcal{L}_{pK}^{2^{-1}}(T_{p1_2}) - 1 ,$$

$$T_{p1_1} \in Z_{s_1}$$

$$T_{p1_2} \in Z_{s_2}$$

$$T_{p1} \in Z_s .$$

Inny wariant tego typu warunków uzyskać można, jeśli założyć się, że działanie operatora T_{p1} powinno rozpocząć się po upływie d jednostek od momentu zakończenia działania operatora $T_{p,i-1}$. Wówczas warunek M.1 przyjmie postać:

M.1b

$$x_{sp} - x_{s-1,p} = H(T_{p,l-1}) + d .$$

Można również nałożyć warunek na to, by działanie operatora T_{p1} zakończyło się nie później niż w d -tej jednostce.

Warunek M.1 przybierze wówczas postać:

M.1c

$$x_{sp} + H(T_{p1}) \leq d .$$

Do modelu można również wprowadzić założenie żądające by w danej warstwie w każdej jednostce działał jakiś operator^{1/}. Warunek ten wprowadza się do modelu w postaci równości w M.2 dla tych warstw dla których założenie to ma być spełnione.

Można również w modelu uwzględnić założenie, że dla określonego p , w każdej jednostce z przedziału $[0, \sum_1 H(T_{p1}) - 1]$

^{1/} w zadaniach z organizacji produkcji odpowiada to założeniu, że niektóre stanowiska muszą pracować bez przerw.

działa operator ze zbioru T_{pK} . Warunek ten wprowadza się do modelu w postaci równości w M.1 dla tych p , dla których to ma być.

3.3.5. Model Wagnera

Model ten został skonstruowany przy założeniach takich samych dla jakich opracowano model Bowmana, przyjmując przy tym dodatkowo, że:

$$\forall p_1 \in M, p_2 \in M \quad \mathcal{L}_{p_1 K}(i) = T_{p_1 l} \wedge \mathcal{L}_{p_2 K}(i) = T_{p_2 l} \cdot$$

Mówiąc inaczej, dla każdego p funkcje sterujące \mathcal{L}_{pK} są identyczne w tym sensie, że operator T_{pl} działa jako i -ty z kolei. Wobec tego można przenieńrować elementy zbioru K tak, aby:

$$\mathcal{L}_{pK}(i) = T_{pi}$$

a wobec tego, że:

$$\omega(p, l) = \{s\}$$

$$\pi(s) = (p, l)$$

po takim przenieńrowaniu uzyska się to, że:

$$T_{pl} \in Z_1 \cdot$$

Zauważmy, że przy tych założeniach zadanie sprowadza się do wyznaczenia funkcji sterującej \mathcal{L}_{M1} , tzn. funkcji ustalającej kolejność działania operatorów T_{M1} , czyli w warstwie pierwszej, (lub dowolnej innej) gdyż w pozostałych warstwach kolejność ta jest taka sama. Wprowadźmy wobec tego oznaczenia:

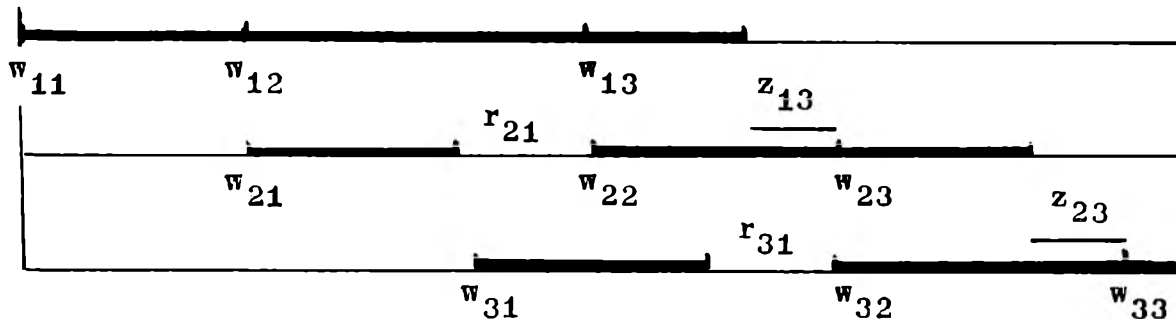
$$x_{pi} = \begin{cases} 1 & \text{jeśli operator } T_{pi} \text{ działa w warstwie} \\ & \text{jako } i\text{-ty z kolei,} \\ 0 & \text{w przeciwnym przypadku.} \end{cases}$$

w_{si} - jednostka w której i -ty z kolei operator rozpoczyna działanie w s -tej warstwie,

r_{si} - ilość jednostek jaka dzieli w warstwie s -tej moment rozpoczęcia działania operatora $i+1$ od momentu zakończenia działania operatora i -tego,

z_{si} - ilość jednostek jaka dzieli moment zakończenia działania i -tego operatora w warstwie s -tej od momentu rozpoczęcia działania tego operatora w warstwie $s+1$ -szej.

Znaczenie poszczególnych symboli można zobaczyć na poniższym diagramie:



Ograniczenia modelu są następujące:

W.1

$$w_{si} = w_{s,i-1} + \sum_{p=1}^m H(T_{ps}) x_{p,i-1} + r_{s,i-1}$$

$$s = 1, 2, \dots, k$$

$$i = 2, 3, \dots, m.$$

W.2

$$w_{si} = w_{s-1,i} + \sum_{p=1}^m H(T_{p,s-1}) x_{pi} + z_{s-1,i}$$

$$s = 2, 3, \dots, k$$

$$i = 1, 2, \dots, m$$

W.3

$$\sum_{p=1}^m x_{pi} = 1$$

$$i = 1, 2, \dots, m$$

W.4

$$\sum_{i=1}^m x_{pi} = 1$$

$$p = 1, 2, \dots, m$$

Warunki W.1 oraz W.2 są równoważne warunkom następującym:

W.5

$$\sum_{p=1}^m H(T_{p,s-1}) x_{pi} + z_{s-1,i} + r_{s-1,i-1} - z_{s-1,i-1} - r_{s,i-1} -$$

$$- \sum_{p=1}^m H(T_{ps}) x_{p,i-1} = 0$$

$$s = 2, 3, \dots, k$$

$$i = 2, 3, \dots, m$$

Przy tych warunkach należy zminimalizować:

$$\sum_{p=1}^m H(T_{pK}) + \sum_{i=1}^{m-1} r_{ki} + \sum_{p=1}^m \sum_{l=1}^{k-1} x_{pl} H(T_{pl}) .$$

Podobnie jak i w modelu Manne'a tak i tutaj można wprowadzić dodatkowe założenia, wynikające z konkretnej sytuacji.

Dla przykładu można dołączyć założenie o tym, by w danej warstwie np. s_1 w każdej jednostce działał pewien operator. Założenie to wprowadza się do modelu przyjmując $r_{s_1} = 0$ dla każdego i oraz dla tych s_1 , dla których to założenie ma być spełnione.

Można również wprowadzić założenie, że operatory T_{pK} muszą działać w sposób ciągły. Założenie to wprowadza się do modelu przyjmując w odpowiednich równaniach M.2 $z_{s_1} = 0$.

Wymieniliśmy tu przykładowo dwie modyfikacje modelu Wagnera, ale jak nietrudno się zorientować, modyfikacji tych może być więcej; tak samo zresztą jak i pozostałych modeli. Modyfikacje te zależą będą od konkretnego zadania, które trzeba rozwiązać.

3.4. Metody podziału i ograniczeń

3.4.1. Idea metody podziału i ograniczeń

Metoda podziału i ograniczeń (ang. Branch and Bound) jest jedną z najogólniejszych metod rozwiązywania zadań programowania matematycznego, nie też dziwnego, że znalazła zastosowanie w rozwiązywaniu zadań sekwencyjnych. Właściwie jest to pewien schemat postępowania, którego konkretne postacie umożliwiają rozwiązywanie różnego typu zadań.

Poszukiwanie optymalnego rozwiązania prowadzone według tego schematu polega na tym, że zbiór rozwiązań dopuszczalnych za-

dania dzieli się na coraz mniejsze podzbiory, przy czym istnieje sposób pozwalający na eliminowanie z dalszych rozważań podzbiorów, które są mniej perspektywiczne z punktu widzenia istnienia ekstremum. W praktyce zabezpiecza to całkowicie przed kombinatorycznym przeglądem wszystkich rozwiązań dopuszczalnych.

A oto idea metody podziału i ograniczeń.

Zadanie, które należy rozwiązać jest następujące:

Wyznaczyć minimum funkcji $f(X)$ przy warunku:

$$X \in D,$$

gdzie D jest ustalonym zbiorem.

Lemat 3.1

$$A \subset D, B \subset D \wedge B \subset A \Rightarrow a = \min_{X \in A} f(X) \leq \min_{X \in B} f(X) = b.$$

Dowód

Założmy, że jest inaczej tzn. $a > b$. Wówczas punkt $X_0 \in B$ taki, że $b = f(X_0)$ byłby, wobec zawierania się zbioru B w zbiorze A punktem minimalizującym f na zbiorze A co jest sprzeczne z założeniem.

Definicja 3.1

Liczba $G(A)$ nazywa się wartością graniczną funkcji f na zbiorze $A \subset D$ (krótko wartością graniczną), jeśli:

$$G(A) \leq f(X) \text{ dla } X \in A.$$

Procedura znajdowania rozwiązania optymalnego za pomocą metody

podziału i ograniczeń jest następująca:

1^o Wyznaczamy $G(D)$. Jeśli znajdziemy takie $X^0 \in D$, że:

$$f(X^0) = G(D)$$

to X^0 jest rozwiązaniem optymalnym, ponieważ z definicji:

$$G(D) \leq f(X) \quad \text{dla każdego } X \in D.$$

Jeśli nie udało się znaleźć takiego X^0 , to:

2^o a/ dzielimy zbiór D na r_1 takich podzbiorów, że:

$$D = \bigcup_{i=1}^{r_1} D_i^1.$$

Podział ten nazywamy podziałem pierwszego rzędu.

Obliczamy $G(D_i^1)$ dla każdego i , przy czym jeśli dla pewnej j ($1 \leq j \leq r_1$) D_j^1 jest zbiorem pustym, to przyjmujemy:

$$G(D_j^1) = \infty,$$

b/ wyznaczymy:

$$\min_i G(D_i^1) = G(D_s^1).$$

Jeśli istnieje taki punkt $X^1 \in D$, że:

$$f(X^1) = G(D_s^1)$$

to X^1 jest rozwiązaniem optymalnym. Wtedy bowiem:

$$f(X^1) \leq G(D_i^1) \leq f(X)$$

dla $X \in D_i^1 \quad i = 1, 2, \dots, r_1.$

A więc:

$$f(x^1) \leq f(x) \quad \text{dla } x \in D.$$

Jeśli nie znajdzie się punkt x^1 o żądanych własnościach, to:

3° Dzielimy zbiór D_s^1 na podzbiory:

$$D_s^1 = \bigcup_{i=1}^n D_{s,i}^1$$

i zmieniamy indeksy zbiorów w sposób następujący:

$$\begin{aligned} D_1^1 &= D_1^2, \quad D_2^1 = D_2^2, \quad \dots, \quad D_{s,1}^1 = D_k^2, \quad D_{s,2}^1 = \\ &= D_{k+1}^2, \quad \dots, \quad D_{s,n}^1 = D_{k+n}^2, \quad \dots, \quad D_{r_1}^1 = D_{r_2}^2. \end{aligned}$$

Otrzymaliśmy podział rzędu drugiego:

$$D = \bigcup_{i=1}^{r_2} D_i^2.$$

Dalsze postępowanie polega na znalezieniu zbioru D_j^2 , dla którego wartość graniczna $G(D_j^2)$ jest najmniejsza, a więc obliczamy:

$$\min_i G(D_i^2) \quad i = 1, 2, \dots, r_2$$

i przechodzimy do punktu 2b.

Wobec lematu, dzieląc zbiór D^\wedge na podzbiory $D_1^\wedge, \dots, D_p^\wedge$ w taki sposób, że:

$$D^\wedge = \bigcup_{i=1}^p D_i^\wedge$$

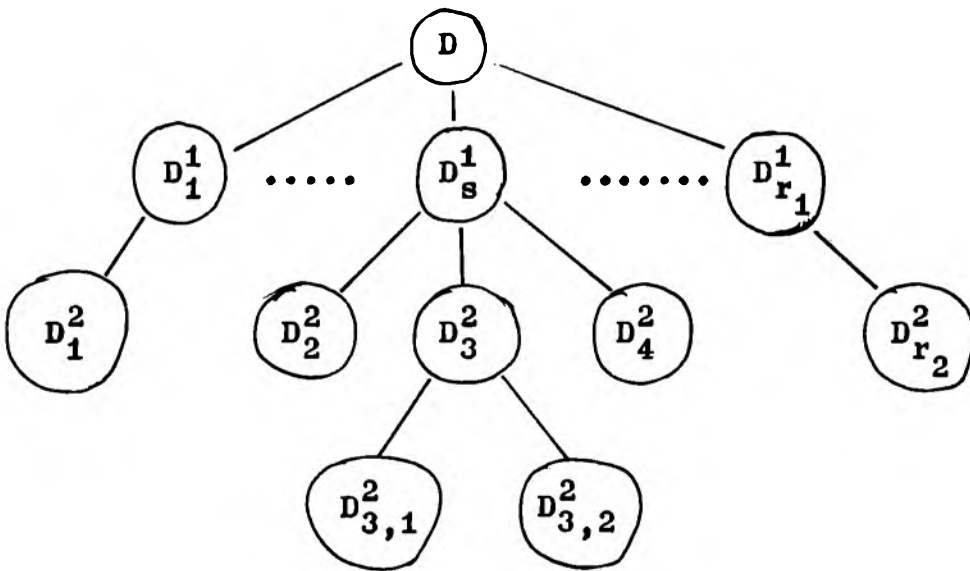
otrzymamy:

$$G(D_1^*) \geq G(D^*).$$

Stąd wniosek, w procesie podziału zbiorów wartości graniczne będą rosły a ściślej mówiąc nie malały. Szybszą zbieżność algorytmu można zapewnić, jeśli wartość graniczną tak się zdefiniuje, że:

$$G(\hat{D}_1) > G(D^*).$$

Poniższy graf prezentuje kilka kroków procedury podziału zbioru D .



Jak wynika z opisu, efektywność metody zależy od sposobu podziału zbioru D na podzbiory oraz od wartości granicznej G .

Należy zwrócić uwagę na pewną trudność na jaką można napotkać przy próbach rozwiązywania zadań programowania sekwencyjnego metodą podziału i ograniczeń. Trudność ta polega na tym, iż z góry nie wie się przy którym podziale napotka się taką wartość:

$$G(D_s^j) = f(x^j)$$

tn. rozwiązanie zadania. Ma to niebagatelne znaczenie przy maszynowej realizacji tego algorytmu. Wchodzi tu bowiem w rachubę problem organizacji pamięci, a ponadto przy zadaniach o dużych rozmiarach może zdarzyć się (i zdarza się to prawie zawsze), że pamięć nie pomieści wszystkich zbiorów, które należy zapamiętać, co przekreśla praktycznie wartość metody. Tym niemniej warto zapoznać się z propozycjami rozwiązania zadania programowania sekwencyjnego metodą podziału i ograniczeń tym bardziej, że można uniknąć wyżej wymienionych trudności poprzez niewielką modyfikację metody.

Propozycje różnych autorów wykorzystania idei metody podziału i ograniczeń dotyczą zadania spełniającego następujące założenia:

- dla każdego $p \in M$ dana jest funkcja sterująca α_{pK} ,
- dla każdego $p \in M$ i każdego $l \in K$

$$\omega(p, l) = \{s\}$$

$$\forall s \in N \quad \pi(s) = \{M \times l\},$$

- dla każdego $p_1 \in M$ i $p_2 \in M$ funkcje α_{p_1K} i α_{p_2K} są identyczne w tym sensie, że:

$$\alpha_{p_1K}(i) = T_{p_1}i$$

$$\alpha_{p_2K}(i) = T_{p_2}i.$$

Wobec tego można zapisać, że:

$$\omega(p, l) = l$$

oraz

$$\pi(s) = \{M \times s\}.$$

Zadanie przy tych założeniach sprowadza się do wyznaczenia funkcji sterującej α_{M1} , a więc tylko w jednej warstwie w pozostałych bowiem warstwach kolejność działania operatorów jest taka sama. Innymi słowy zadanie polega na wyznaczeniu takiej permutacji operatorów T_{M1} by:

$$H(T_{MK\alpha}) \rightarrow \min_{\alpha}$$

Omówimy teraz zasadę podziału na podzbiory w przypadku naszego zadania. W tym celu oznaczymy:

$P(i|k)$ - zbiór wszystkich permutacji operatorów T_{M1} , w których k pierwsze elementy operatory są ustalone pozostałe zaś są dowolne. Przy czym na miejscu k -tym znajduje się operator T_{i1} .

Zatem $P(i|m)$ jest ustaloną permutacją, a tym samym wyznacza jedną z funkcji α_{M1} .

$P(0|0)$ jest zbiorem wszystkich permutacji operatorów T_{M1} , a tym samym zbiorem wszystkich możliwych funkcji sterujących α_{M1} .

Podziałowi pierwszego rzędu podlega $P(0|0)$. W wyniku tego podziału uzyskuje się m następujących podzbiorów:

$$P(i_1|1) \quad \text{gdzie} \quad i_1 = 1, 2, \dots, m.$$

Następny podział daje $m-1$ podzbiorów:

$$P(i_2|2) \quad \text{gdzie} \quad i_2 \neq i_1$$

Podział następny daje $m-2$ podzbiory:

$$P(i_3|3) \quad \text{gdzie} \quad i_3 \neq i_1, i_2$$

itd.

W wyniku ostatniego podziału dostaje się zbiór jednoelementowy, postaci

$$P(i_m|m) \quad \text{gdzie} \quad i_m \neq i_1, i_2, \dots, i_{m-1}.$$

Powstaje teraz problem eliminacji z dalszego postępowania tych podzbiorów, dla których dalszy podział nie ma sensu, tzn. o których wiadomo, że nie zawierają optymalnej permutacji. Jako kryterium eliminujące można przyjąć

$$A = H(T_{MK_\alpha})$$

gdzie α jest dowolną funkcją sterującą.

Przyjmijmy, że:

$$B = \min_{\alpha} H(T_{MK_\alpha}).$$

Łatwo zauważyć, że dla ustalonego k :

$$A \geq B \geq \min_j G(D_j^k),$$

a więc z dalszego postępowania wypadać będą te podzbiory D_j^k

dla których:

$$A < G(D_j^k).$$

Z przeprowadzonego rozumowania wynika następująca uwaga.

Uwaga

Efektywność postępowania będzie tym większa (odrzucać będzie tym więcej podzbiorów) im różnica $A - \min_j G(D_j^k)$ będzie mniejsza.

Do uwagi tej powrócimy w dalszej części.

Ostatnia sprawa to wybór funkcji granicznej. Wszystkie proponowane metody, które tu zaprezentujemy różnią się właśnie sposobem wyznaczania granicznej wartości funkcjonału. Przed opisem tych metod wprowadzimy następujące oznaczenia:

I - zbiór indeksów $1, 2, \dots, m$,

I_k - k -elementowy podzbiór zbioru I ,

$P(i_1, i_2, \dots, i_k)$ - k -elementowa permutacja operatorów T_{M1} ,

$H(Z_s(P(i_1, i_2, \dots, i_k)))$ - wartość funkcjonału H liczona do warstwy s -tej i operatorów sterowanych zgodnie z permutacją i_1, i_2, \dots, i_k .

3.4.2. Metoda Lomnickiego

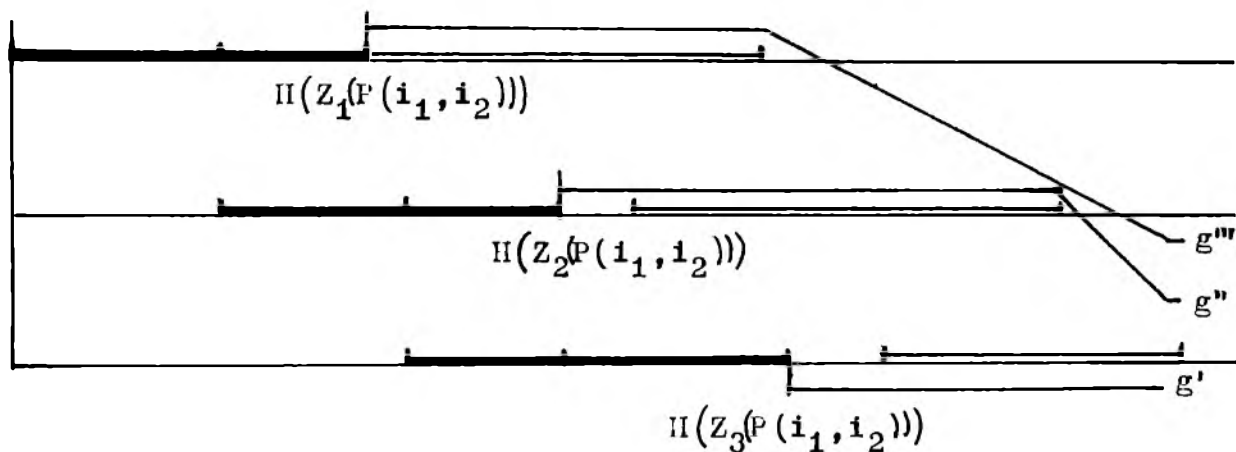
Autor stosuje metodę podziału i ograniczeń dla przypadku trzech warstw operatorów.

Proponuje przyjęcie następującej wartości granicznej funkcjonału H :

$$G(P(i_k | k)) = \max \begin{cases} H(Z_3(P(i_1, i_2, \dots, i_k))) + \sum_{p \in I - I_k} H(T_{p3}) & g^{\wedge} \\ H(Z_2(P(i_1, \dots, i_k))) + \sum_{p \in I - I_k} H(T_{p2}) + \min_{p \in I - I_k} H(T_{p3}) & g^{\wedge\wedge} \\ \min_{p \in I - I_k} [H(T_{p2}) + H(T_{p3})] + \sum_{p=1}^m H(T_{p1}) & g^{\wedge\wedge\wedge} \end{cases}$$

Sposób wyznaczenia wartości $H(Z_s(P(i_1, i_2, \dots, i_k)))$ podano w rozdziale pierwszym.

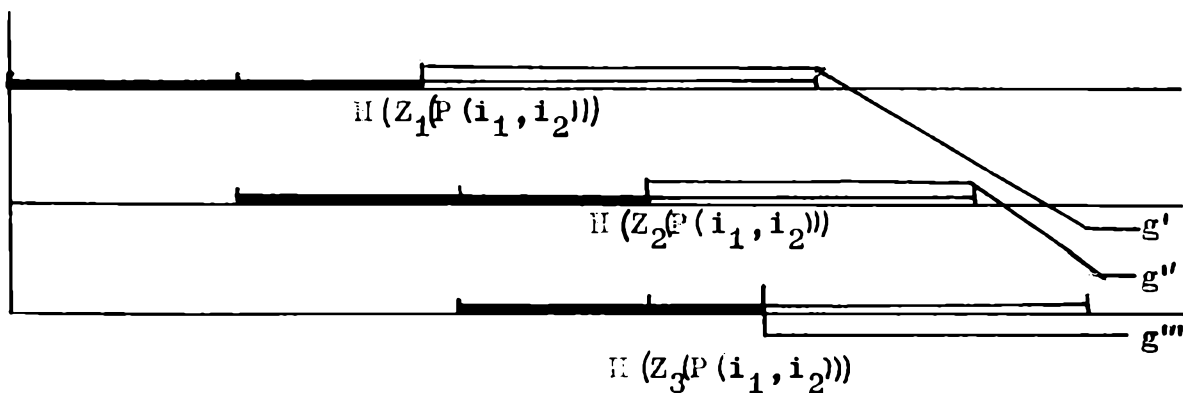
Tutaj natomiast, przy pomocy diagramu Gantta zilustrujemy funkcje g^{\cdot} , $g^{\cdot\cdot}$, $g^{\cdot\cdot\cdot}$.



3.4.3. Metoda Ignalla i Schrage'a

Metoda ta podobnie jak i poprzednia dotyczy trzech warstw operatorów. Autorzy proponują następujący sposób wyznaczania wartości granicznej funkcjonału:

$$G(P(i_k | k)) = \max \begin{cases} H(Z_1(P(i_1, \dots, i_k))) + \sum_{p \in I - I_k} H(T_{p1}) + \min_{p \in I - I_k} [H(T_{p2}) + H(T_{p3})] & g^{\cdot} \\ H(Z_2(P(i_1, \dots, i_k))) + \sum_{p \in I - I_k} H(T_{p2}) + \min_{p \in I - I_k} H(T_{p3}) & g^{\cdot\cdot} \\ H(Z_3(P(i_1, \dots, i_k))) + \sum_{p \in I - I_k} H(T_{p3}) & g^{\cdot\cdot\cdot} \end{cases}$$



Jak widać wartości graniczne dla obu metod są takie same.

3.4.4. Metoda McMahona i Burтона

Również i ta metoda została opracowana dla trzech warstw operatorów. Jako wartość graniczną autorzy proponują przyjąć:

$$G(P(i_k | k)) = \max[G_1(P(i_k | k)), G_2(P(i_k | k))]$$

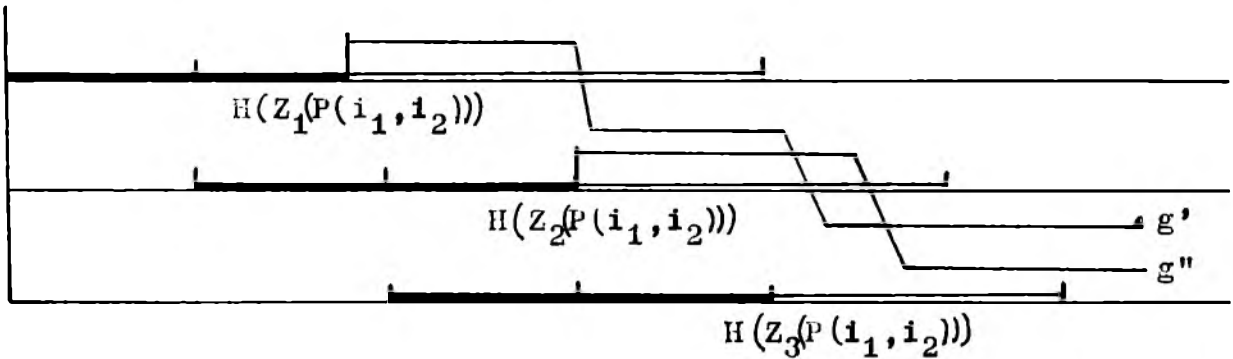
gdzie:

$$G_1(P(i_k | k)) = \max \begin{cases} H(Z_1(P(i_1, \dots, i_k))) + \sum_{p \in I - I_k} H(T_{p1}) + \min_{p \in I - I_k} [H(T_{p2}) + H(T_{p3})] \\ H(Z_2(P(i_1, \dots, i_k))) + \sum_{p \in I - I_k} H(T_{p2}) + \min_{p \in I - I_k} H(T_{p3}) \\ H(Z_3(P(i_1, \dots, i_k))) + \sum_{p \in I - I_k} H(T_{p3}) \end{cases}$$

$$G_2(P(i_k | k)) = \max \begin{cases} H(Z_1(P(i_1, \dots, i_k))) + \max_{p \in I - I_k} \left[\sum_{k \in I - I_k}^3 H(T_{pk}) + \sum_{k \in I - I_k} \min [H(T_{k1}), H(T_{k2})] \right] \\ H(Z_2(P(i_1, \dots, i_k))) + \max_{p \in I - I_k} \left[\sum_{l=2}^3 H(T_{pl}) + \sum_{\substack{k \in I - I_k \\ k \neq p}} \min [H(T_{k2}), H(T_{k3})] \right] \end{cases}$$

g'
 g''

Wartość $G_1(P(i_k|k))$ jest taka sama jak w poprzednich metodach natomiast $G_2(P(i_k|k))$ przedstawimy na diagramie:



Wprowadzie tak przyjęta wartość graniczna wymaga wykonania większej ilości obliczeń; jednak pozwala - jak twierdzą autorzy - skuteczniej eliminować te podzbiory, których już nie trzeba dzielić.

3.4.5. Metoda Browna i Lomnickiego

Metoda Lomnickiego dla trzech warstw operatorów została przez Browna i Lomnickiego uogólniona na przypadek n warstw. Wartość graniczną funkcjonału wyznacza się w sposób następujący:

$$G(P(i_k|k)) = \max (g^1, g^2, \dots, g^n)$$

gdzie:

$$g^j = H(Z_j(P(i_1, \dots, i_k))) + \sum_{p \in I - I_k} H(T_{pj}) + \min_{p \in I - I_k} \sum_{l=j+1}^n H(T_{pl})$$

$$g^n = H(Z_n(P(i_1, \dots, i_k))) + \sum_{p \in I - I_k} H(T_{pn}).$$

Podobne zagadnienie rozwiązywał Tiutiukin, który dla wyznaczenia wartości granicznej proponował:

$$G(P(i_k | k)) = H(Z_n(P(i_1, \dots, i_k))) + \sum_{p \in I - I_k} H(T_{pn})$$

Jak widać wartość ta pokrywa się z wartością funkcji g^n u Browna i Lomnickiego.

3.4.6. Metoda Czernowej

W celu prezentacji metody zaproponowanej przez Czernową wprowadźmy następujące oznaczenia:

$$L_j^1 = \sum_{l=1}^j H(T_{i_l 1})$$

$$j = 1, 2, \dots, n$$

$$L_1^l = \sum_{k=1}^l H(T_{i_k 1})$$

$$l = 1, 2, \dots, m.$$

$$L_j^l = \max(L_{j-1}^l, L_j^{l-1}) + H(T_{i_l j}).^{1/}$$

$$l = 2, 3, \dots, m$$

$$j = 2, 3, \dots, n$$

^{1/} Wielkości L_j^l można interpretować jako moment zakończenia działania operatora $\alpha_{M1}(l)$ w warstwie j -tej.

$$h_j^1 = L_j^1 - L_{j-1}^1$$

$$j = 2, 3, \dots, n$$

$$l = 2, 3, \dots, m$$

Oznaczmy:

$$R_j^1 = \max \left[R_{j-1}^1 + H(T_{i_1, j-1}) - h_j^{l-1}, 0 \right]$$

przyjmując przy tym, że:

$$R_1^1 = 0.$$

Czernowa proponuje przyjąć jako wartość graniczną funkcjo-
nału:

$$G(P(i_1 | l)) = \max \left[V(P(i_1 | l)), W(P(i_1 | l)) \right],$$

gdzie:

$$V(P(i_1 | l)) = \sum_{i=1}^{w-1} H(T_{i_1 j}) + \sum_{i=1}^m H(T_{i w}) + \min_{i \in I - I_1} \sum_{j=w+1}^n H(T_{i j}) + \sum_{k=2}^l R_w^k,$$

$$W(P(i_1 | l)) = \sum_{j=1}^n H(T_{i_1 j}) + \sum_{\substack{i=1 \\ i \neq i_1}}^m H(T_{i n}) + \sum_{k=2}^l R_n^k, \quad 3.5$$

jeśli:

$$\max \left\{ \begin{array}{l} \max_i \sum_{j=1}^n H T_{i j} = \sum_{j=1}^n H T_{q j} = b \\ \max_j \sum_{i=1}^m H T_{i j} = \sum_{i=1}^m H T_{i w} = c \end{array} \right\} = c.$$

Jeśli natomiast:

$$\max \left\{ \begin{array}{l} \max_i \sum_{j=1}^n H(T_{ij}) = \sum_{j=1}^n H(T_{qj}) = b \\ \max_j \sum_{i=1}^m H(T_{ij}) = \sum_{i=1}^m H(T_{iw}) = c \end{array} \right\} = b$$

to:

$$V(P(i_1 | l)) = \sum_{j=1}^n H(T_{i_1 j}) + \sum_{\substack{i=1 \\ i \neq i_1}}^m H(T_{in}) + \sum_{k=1}^l R_n^k$$

$$W(P(i_1 | l)) = \sum_{k=1}^l H(T_{i_k 1}) + \sum_{j=1}^n H(T_{qj}) + \sum_{\substack{i \in I - I_1 \\ i \neq q}} H(T_{in}) -$$

$$- \max_{\substack{i \in I - I_1 \\ i \neq q}} [H(T_{in}) - H(T_{i1}), 0] \quad 3.6$$

dla $q \notin I_1$

Jeśli q jest równe któremuś i_s ($s = 1, 2, \dots, l$) to:

$$W(P(i_1 | l)) = \sum_{k=1}^{s-1} H(T_{i_k 1}) + \sum_{j=1}^n H(T_{qj}) + \sum_{k=s+1}^l H(T_{i_k n}) +$$

$$+ \sum_{i \in I - I_1} H(T_{in}) + \sum_{k=s+1}^l R_n^k$$

W metodzie tej sposób wyznaczania wartości granicznej jest nieco kłopotliwy, w zamian jednak zyskuje się na jakości oszacowania, co pozwala wyeliminować więcej podzbiorów.

Prześledźmy jak działa ta metoda na poniższym przykładzie, ograniczając się jedynie do podziałów pierwszego, drugiego i trzeciego rzędu.

Przykład

$H(T_{ij}) =$

$j \backslash i$	1	2	3	4	\sum_j
1	9	8	12	6	35
2	5	15	10	4	34
3	4	18	11	3	36
4	8	6	5	7	26
5	6	13	9	5	23
6	7	12	13	4	37
\sum_i	36	72	60	29	

Zwróćmy uwagę na zapis zadania. Ponieważ zgodnie z założeniem:

$$\omega(p,1) = \{1\},$$

$H(\theta)$ można zapisać w postaci macierzy $H(T_{ij})$, gdzie $i = 1, 2, \dots, m$, $j = 1, 2, \dots, n$.

Ponieważ:

$$\max_i \sum_{j=1}^4 H(T_{ij}) = \sum_{j=1}^4 H(T_{6j}) = 37 = b$$

$$\max_j \sum_{i=1}^6 H(T_{ij}) = \sum_{i=1}^6 H(T_{i2}) = 72 = c$$

a więc; $\max(b,c) = c$, stosować będziemy dla wyznaczenia wartości granicznej funkcjonału 3,5, przyjmując przy tym $w=2$.

Dokonujemy teraz podziału zbioru $P(0|0)$ na podzbiory

$$P(0|0)$$

$$P(1|1), P(2|1), P(3|1), P(4|1), P(5|1), P(6|1)$$

i dla każdego podzbioru wyznaczamy wartość graniczną:

$V(P(1 1)) = 9+72+12+0=93$	$W(P(1 1)) = 35+23+0=58$	$G(P(1 1)) = 93$
$V(P(2 1)) = 5+72+12+0=89$	$W(P(2 1)) = 34+25+0=59$	$G(P(2 1)) = 89$
$V(P(3 1)) = 4+72+12+0=88$	$W(P(3 1)) = 36+26+0=62$	$G(P(3 1)) = 88$
$V(P(4 1)) = 8+72+14+0=94$	$W(P(4 1)) = 26+22+0=48$	$G(P(4 1)) = 94$
$V(P(5 1)) = 6+72+12+0=90$	$W(P(5 1)) = 23+24+0=47$	$G(P(5 1)) = 90$
$V(P(6 1)) = 7+72+12+0=91$	$W(P(6 1)) = 37+25+0=62$	$G(P(6 1)) = 91$

Wynika z tego, że najbardziej perspektywiczny z punktu widzenia istnienia rozwiązania optymalnego jest zbiór $P(3|1)$.

Dokonujemy zatem podziału drugiego rzędu i otrzymujemy:

$$P(1|2), P(2|2), P(4|2), P(5|2), P(6|2)$$

Wyznaczamy teraz wartości graniczne dla każdego wyżej wypisanego podzbioru.

Niezbędne do tego dane będziemy zapisywać w postaci macierzy:

$l = 2$				
j	1	2	3	4
$H(T_{1_{1-2}, j})$				
$H(T_{1_{1-1}, j}) = H(T_{3j})$	4	18	11	3
L_j^{1-2}				
$L_j^{1-1} = L_j^1$	4	22	33	36
$L_j^1 - L_{j-1}^1 = h_j^1$	4	18	11	3
$R_j^{11} = 1$	0	0	0	0
$R_j^{11=2}$	0	0	4	11
$R_j^{11=4}$	0	0	0	2
$R_j^{11=5}$	0	0	2	8
$R_j^{11=6}$	0	0	1	11

A wobee tego:

$$\begin{array}{lll}
 V(P(31|2)) = 4+72+12+0=88 & W(P(31|2)) = 36+26+9=71 & G(P(31|2)) = 88 \\
 V(P(32|2)) = 4+72+12+0=88 & W(P(32|2)) = 36+26+11=73 & G(P(32|2)) = 88 \\
 V(P(34|2)) = 4+72+14+0=90 & W(P(34|2)) = 36+26+2=64 & G(P(34|2)) = 90 \\
 V(P(35|2)) = 4+72+12+0=88 & W(P(35|2)) = 36+26+8=70 & G(P(35|2)) = 88 \\
 V(P(36|2)) = 4+72+12+0=88 & W(P(36|2)) = 36+26+11=73 & G(P(36|2)) = 88
 \end{array}$$

$P(31 2)$	$P(32 2)$	$P(34 2)$	$P(35 2)$	$P(36 2)$
$G=88$	$G=88$	$G=90$	$G=88$	$G=88$

Ponieważ minimalna wartość graniczna funkcjonału jest dla kilku podzbiorów jednakowa do podziału wybierzemy jeden z tych podzbiorów przy czym wyboru dokonujemy w sposób arbitralny. Niech to będzie dla przykładu $P(32|2)$.

W wyniku podziału otrzymujemy podzbiory:

$P(321 3)$	$P(324 3)$	$P(325 3)$	$P(326 3)$
------------	------------	------------	------------

Wyznaczamy graniczną wartość, postępowanie jest takie same jak poprzednio:

$l = 3$				
j	1	2	3	4
$H(T_{1_{1-2},j}) = H(T_{3j})$	4	18	11	3
$H(T_{1_{1-1},j}) = H(T_{2j})$	5	15	10	4
L_j^{1-2}	4	22	33	36
$L_j^{1-1} = L_j^1$	9	37	47	51
$L_j^1 - L_{j-1}^1 = h_j^1$	9	28	10	4
$1_{1=1}$ $R_j^{1=1}$	0	0	0	0
$1_{1=4}$ $R_j^{1=4}$	0	0	0	1
$1_{1=5}$ $R_j^{1=5}$	0	0	3	8
$1_{1=6}$ $R_j^{1=6}$	0	0	2	11

Czyli:

$$\begin{aligned} V(P(321|3)) &= 4+72+12+0=88 & W(P(321|3)) &= 36+26+8+9=79 & G(P(321|3)) &= 88 \\ V(P(324|3)) &= 4+72+14+0=90 & W(P(324|3)) &= 36+26+1+2+65 & G(P(423|3)) &= 90 \\ V(P(325|3)) &= 4+72+12+0=88 & W(P(325|3)) &= 36+26+8+8=78 & G(P(325|3)) &= 88 \\ V(P(326|3)) &= 4+72+12+0=88 & W(P(326|3)) &= 36+26+11+11=84 & G(P(326|3)) &= 88 \end{aligned}$$

Postępowanie to trzeba kontynuować tak długo dopóki nie uzyska się zbioru jednoelementowego, dla którego wartość graniczna będzie mniejsza od wartości granicznych wszystkich podzbiorów otrzymanych w podziałach stopni niższych. Wynika z tego, że na każdym etapie należy pamiętać wartości graniczne funkcjonału dla podzbiorów uzyskanych w etapach poprzednich.

3.4.7. Modyfikacja metody podziału i ograniczeń

Proponowaną modyfikację tego postępowania można opisać w kilku następujących punktach:

- 1^o Wybrać dowolną funkcję sterującą \mathcal{C}_{M1} , wyznaczyć wartość funkcjonału H dla tej funkcji. Oznaczmy tę wartość A .
- 2^o Dokonać podziału pierwszego rzędu. Dla uzyskanych podzbiorów wyznaczyć dowolną metodą wartość graniczną funkcjonału. Usunąć te podzbiory, dla których:
$$G > A.$$
- 3^o Z pozostałych wybrać w sposób arbitralny dowolny podzbiór i dokonać jego podziału. Dla każdego z tych podzbiorów wyznaczyć wartość graniczną funkcjonału

i usunąć te podzbiory, dla których:

$$G > A.$$

Postępowanie kontynuować tak długo dopóki nie uzyska się przy którymś podziale tego, że uzyskany podzbiór będzie jednoelementowy, i wówczas mamy rozwiązanie nie gorsze od wybranego \mathcal{C}_{M1} , bądź to, że wszystkie podzbiory uzyskane z podziału zostaną usunięte ponieważ wartość graniczna dla nich będzie większa od A . W tym przypadku trzeba cofnąć się do jednego z podzbiorów uzyskanych z podziału rzędu o jeden niższego, dla takiego, że $G < A$.

Może również zdarzyć się, że wartość graniczna funkcjonału dla zbioru jednoelementowego będzie większa od wartości granicznej podzbiorów uzyskanych z podziału rzędu niższego a mniejsza od A . Znaczy to, że została znaleziona funkcja sterująca \mathcal{C}_{M1} lepsza od przyjętej na początku. Wówczas dokonujemy podziału jak poprzednio dowolnego podzbioru przyjmując za A wartość funkcjonału H dla funkcji \mathcal{C}_{M1} . Jak widać postępowanie takie musi w końcu dać rozwiązanie optymalne.

Postępowanie to ma niewątpliwie dwie zalety:

- 1^o Nie ma potrzeby pamiętania wartości granicznych dla wszystkich uzyskanych w procesie podziału podzbiorów. Ułatwia to znacznie pracę, gdyż nie ma obawy, że zbiór ten przerośnie pojemność pamięci maszyny.
- 2^o Proces obliczenia można w dowolnej chwili przerwać jeśli uzna się, że uzyskane rozwiązanie jest zadowalające, gdyż pracując zmodyfikowaną metodą zawsze dysponuje się pewnym rozwiązaniem.

Wadą proponowanej metody jest to, że obliczenia mogą trwać zbyt długo, gdyż w klasycznej metodzie podziału i ograniczeń penetracji dokonuje się jak gdyby szerokim frontem, natomiast przy tej modyfikacji penetruje się w głąb. Jednak przy zadaniach dużych rozmiarów jest to jedyna do przyjęcia metoda postępowania.

3.5. M e t o d y e l i m i n a c y j n e

3.5.1. Wprowadzenie

Kolejną grupą metod rozwiązywania zadań programowania sekwencyjnego to metody eliminacyjne. Metody te zaproponowano do zadań typu E z rozdziału pierwszego, a więc do zadań, dla których stosowano metody podziału i ograniczeń; wobec czego nie będziemy tutaj raz jeszcze przytaczali założeń tych zadań.

Istota każdej z metod eliminacyjnych polega na tym, że w oparciu o pewną regułę eliminuje się z rozważań niektóre funkcje sterujące \mathcal{C}_{M_1} jako te, które nie mogą być optymalne. Dla uniknięcia skomplikowanych zapisów, omawiając kolejne metody eliminacyjne korzystać będziemy z następujących oznaczeń:

R - złożenie co najwyżej $m-1$ operatorów ze zbioru T_{M_1}
gdzie l jest ustalone^{1/}, w szczególności R może być złożeniem pustym.

^{1/} przy założeniach naszego zadania wystarczy rozpatrywać sterowanie w jednej tylko warstwie, gdyż w pozostałych warstwach kolejność działania operatorów będzie taka sama.

$H(R, j)$ - wartość funkcjonału H dla złożenia R liczona do j -tej warstwy włącznie.

Ponadto przyjmujemy:

$$\begin{aligned} H(\emptyset, j) &= 0 && \text{dla } j=1, 2, \dots, n \\ H(R, 0) &= 0 && \text{dla każdego } R, \end{aligned} \quad 3.7$$

R_a - złożenie operatorów, w którym jako ostatni działa operator T_a , pozostałe są identyczne ze złożeniem R .

Jak pokazano w rozdziale pierwszym

$$\begin{aligned} H(R_a, j) &= \max[H(R_a, j-1), H(R, j)] + H(T_{a_j}) \\ j &= 1, 2, \dots, n \end{aligned} \quad 3.8$$

W szczególności:

$$\begin{aligned} H(a, j) &= H(a, j-1) + H(T_{a_j}) \\ H(R_a, 1) &= H(R, 1) + H(T_{a_1}) \\ H(a, 1) &= H(T_{a_1}) \end{aligned} \quad 3.9$$

Wszystkie metody eliminacyjne są zbudowane według reguły:

Jeśli pewien warunek jest spełniony, to wyeliminuj te funkcje sterujące α_{M_1} , które dają złożenie R_b .

Wymieniony warunek nosi nazwę kryterium eliminacyjnego i dla różnych metod jest ono inne.

Przed przystąpieniem do omawiania poszczególnych metod eliminacyjnych wprowadźmy jeszcze następujące oznaczenia:

$$\Delta_j = H(R_{ab}, j) - H(R_b, j) \quad 3.10$$

3.5.2. Metoda Smitha-Dudka

Kryterium eliminacyjne zaproponowane przez Smitha i Dudka jest następujące:

$$a. \Delta_{j-1} \leq H(T_{aj})$$

A.

$$b. H(R_{a,j-1}) \leq H(R_{b,j-1})$$

$$\text{dla } j=2,3,\dots,n$$

Zastanówmy się obecnie nad tym, jaki warunek musi być spełniony, aby - uwzględniając to kryterium - można było wyeliminować chociaż jedną funkcję \mathcal{L} .

W tym celu przyjmijmy:

$$R = \emptyset.$$

Wówczas korzystając z 3.8, 3.7 oraz A.a. mamy:

$$\begin{aligned} H(T_{aj}) &\geq H(ab, j-1) - H(b, j-1) = \max[H(ab, j-2), H(a, j-1)] + \\ &+ H(T_{b, j-1}) - H(b, j-1) \geq H(ab, j-2) + H(T_{b, j-1}) - H(b, j-2) - \\ &- H(T_{b, j-1}) = H(ab, j-2) - H(b, j-2) = \max[H(ab, j-3), H(a, j-2)] + \\ &+ H(T_{b, j-2}) - H(b, j-2) \geq H(ab, j-3) - H(b, j-3) \geq \dots \geq H(ab, 1) - \\ &- H(b, 1) = H(a, 1) + H(T_{b1}) - H(b, 1) = H(T_{a1}) + H(T_{b1}) - H(T_{b1}) = \\ &= H(T_{a1}). \end{aligned}$$

Czyli ostatecznie:

$$H(T_{aj}) \geq H(T_{a1})$$

3.11

Tak więc warunkiem koniecznym dla wyeliminowania chociaż jednej funkcji \mathcal{C} , jest istnienie takiego podzbioru T_{aK} zbioru operatorów T_{MK} , dla którego zachodzi 3.11.

Wobec warunku A.b:

$$H(R_{a,j-1}) \leq H(R_{b,j-1})$$

dla $j=2$ otrzymujemy:

$$H(R_{a,1}) \leq H(R_{b,1})$$

czyli:

$$H(R,1) + H(T_{a1}) \leq H(R,1) + H(T_{b1})$$

stąd:

$$H(T_{a1}) \leq H(T_{b1}) \quad 3.12$$

Oznacza to, że można próbować eliminować tylko te operatory, dla których zachodzi 3.12.

3.5.3. Metoda Szwarca

Kryterium eliminacyjne zaproponowane przez Szwarca ma następującą postać:

$$\begin{array}{l} \text{a. } \Delta_1 \leq \Delta_2 \leq \dots \leq \Delta_n \\ \text{B. } \text{b. } \Delta_j \leq H(T_{aj}) \quad \text{dla } j=2,3,\dots,n \end{array}$$

Warunek konieczny dla wyeliminowania chociaż jednej funkcji sterującej jest taki sam jak i u Smitha i Dudka, gdyż:

$$\text{B implikuje } \Delta_1 \leq H(T_{aj})$$

ale

$$\Delta_1 = H(ab,1) - H(b,1) = H(T_{a1}) + H(T_{b1}) - H(T_{b1}) = H(T_{a1}) .$$

3.5.4. Metoda Bagga i Chakravarti

Bagga i Chakravarti podają inne kryterium eliminacyjne, a mianowicie:

$$c. \Delta_j \leq H(T_{aj}) \quad \text{dla } j=2,3,\dots,n$$

Warunek konieczny dla wyeliminowania chociaż jednej funkcji sterującej, jest tutaj taki sam jak w obydwu poprzednich metodach, gdyż przyjmując $R=\emptyset$, mamy:

$$\begin{aligned} H(T_{aj}) &\geq H(ab,j) - H(b,j) = \max[H(ab,j-1), H(a,j)] + H(T_{bj}) - \\ &- H(b,j) \geq H(ab,j-1) - H(b,j-1) \geq H(ab,j-2) - H(b,j-1) \geq \dots \\ &\geq H(ab,1) - H(b,1) = H(T_{a1}) . \end{aligned}$$

Czyli:

$$H(T_{aj}) \geq H(T_{a1}) .$$

3.5.5. Metoda Dudka i Teutona

Ostatnią z omawianych tu metod eliminacyjnych jest metoda zaproponowana przez Dudka i Teutona, w której kryterium eliminacyjne ma postać:

$$D. H(Rab,j) \leq H(Rba,j) \quad \text{dla } j=2,3,\dots,n.$$

Warto dodać, że dwie ostatnie metody nie gwarantują, iż w zredukowanym zbiorze możliwych funkcji sterujących znajdzie się funkcja optymalna. Innymi słowy dwa ostatnie kryteria eliminacyjne, eliminować mogą funkcje optymalne na co w literaturze można znaleźć przykłady^{1/}.

W kryteriach tych szczególną rolę odgrywa podzbiór operatorów T_{aK} . Jest to mianowicie podzbiór operatorów potencjalnie usuwających inne operatory z danego miejsca w założeniu, gdyż korzystając z kryteriów eliminacyjnych można powiedzieć, iż wszystkie sterowanie R_b mogą być wyeliminowane jeśli odpowiednie kryterium jest spełnione przy zastąpieniu operatora T_b operatorem T_a w uporządkowanym zbiorze R_b .

3.5.6. Twierdzenia o metodach eliminacyjnych

Wypiszmy raz jeszcze poszczególne kryteria eliminacyjne:

- A.
- a. $\Delta_{j-1} \leq H(T_{aj})$
 - b. $H(R_{a,j-1}) \leq H(R_{b,j-1})$
- B.
- a. $\Delta_1 \leq \Delta_2 \leq \dots \leq \Delta_n$
 - b. $\Delta_j \leq H(T_{aj})$

^{1/}dla kryterium D patrz Karuch "Counterexample to a Proposed Algorithm for Optimal Sequencing of Jobs
dla kryterium C patrz Szwaro "A Counterexample to a Solution Method of the $m \times n$ Job Sequencing Problem.

$$C. \quad \Delta_j \leq H(T_{aj})$$

$$D. \quad H(Rab, j) \leq H(Rba, j)$$

Twierdzenie

$$A \Rightarrow B \Rightarrow C \Rightarrow D.$$

Dowód

Pokażemy, że $A \Rightarrow B$.

Zauważmy, że:

$$I. \quad H(Rab, j-1) - H(Rb, j-1) \leq H(T_{aj})$$

$$II. \quad H(Ra, j) - H(R, j) \geq H(T_{aj})$$

$$III. \quad H(Rab, j-1) > H(Rb, j-1)$$

$$IV. \quad H(Ra, j-1) \leq H(Rb, j-1)$$

Nierówności I oraz IV, to po prostu kryterium eliminacyjne A. natomiast nierówności II oraz III są oczywiste ze względu na sposób liczenia $H(R, j)$.

Tworzymy różnicę:

$$\begin{aligned} \Delta_j - \Delta_{j-1} &= H(Rab, j) - H(Rb, j) - (H(Rab, j-1) - H(Rb, j-1)) = \\ &= \max[H(Rab, j-1), H(Ra, j)] + H(T_{bj}) - \max[H(Rb, j-1), H(R, j)] - \\ &- H(T_{bj}) - (H(Rab, j-1) - H(Rb, j-1)) = \max[H(Rab, j-1), H(Ra, j)] - \\ &- \max[H(Rb, j-1), H(R, j)] - H(Rab, j-1) + H(Rb, j-1) . \end{aligned}$$

Zająć tu mogą cztery przypadki:

1^o

$$H(Rab, j-1) \geq H(Ra, j) \quad i$$

$$H(Rb, j-1) \geq H(R, j)$$

wówczas otrzymujemy:

$$H(Rab, j-1) - H(Rb, j-1) - H(Rab, j-1) + H(Rb, j-1) \geq 0$$

$$2^0 \quad H(Rab, j-1) \leq H(Ra, j) \quad \text{i} \quad H(Rb, j-1) \geq H(R, j)$$

wówczas:

$$H(Ra, j) - H(Rb, j-1) - H(Rab, j-1) + H(Rb, j-1) \geq 0$$

$$3^0 \quad H(Rab, j-1) \leq H(Ra, j) \quad \text{i} \quad H(Rb, j-1) \leq H(R, j)$$

wówczas:

$$H(Ra, j) - H(R, j) - (H(Rab, j-1) - H(Rb, j-1)) \geq 0$$

na mocy nierówności I i II.

$$4^0 \quad H(Rab, j-1) > H(Ra, j) \quad \text{i} \quad H(Rb, j-1) < H(R, j)$$

Dodając te dwie nierówności stronami dostajemy:

$$H(Rab, j-1) - H(Rb, j-1) > H(Ra, j) - H(R, j)$$

Nierówność powyższa na mocy I oraz II nie jest możliwa.

W każdym przypadku otrzymujemy, że:

$$\Delta_j \geq \Delta_{j-1}$$

Pokażemy teraz, że:

$$A \Rightarrow B, b$$

$$\begin{aligned} \Delta_j &= H(Rab, j) - H(Rb, j) = \max[H(Rab, j-1), H(Ra, j)] + \\ &+ H(T_{bj}) - \max[H(Rb, j-1), H(R, j)] - H(T_{bj}) = \max[H(Rab, j-1), \\ &H(Ra, j)] - \max[H(Rb, j-1), H(R, j)]. \end{aligned}$$

Czyli:

$$\Delta_j = \max[H(Rab, j-1), H(Ra, j)] - \max[H(Rb, j-1), H(R, j)]$$

3.13

Rozpatrzmy cztery przypadki:

1^o

$$H(Rab, j-1) \geq H(Ra, j) \quad \text{i} \quad H(Rb, j-1) \geq H(R, j)$$

podstawiając to do 3.13 mamy:

$$\Delta_j = H(Rab, j-1) - H(Rb, j-1) \leq H(T_{aj}).$$

Ostatnia nierówność wynika z nierówności I.

2^o

$$H(Rab, j-1) \geq H(Ra, j) \quad \text{i} \quad H(Rb, j-1) \leq H(R, j)$$

po podstawieniu do 3.13 otrzymujemy:

$$\Delta_j = H(Rab, j-1) - H(R, j) \leq H(Rab, j-1) - H(Rb, j-1) \leq H(T_{aj}).$$

3^o

$$H(Rab, j-1) \leq H(Ra, j) \quad \text{i} \quad H(Rb, j-1) \geq H(R, j)$$

Po podstawieniu do 3.13 dostajemy:

$$\begin{aligned} \Delta_j &= H(Ra, j) - H(Rb, j-1) = \max[H(Ra, j-1), H(R, j)] - \\ &- H(Rb, j-1) + H(T_{aj}) \leq H(T_{aj}). \end{aligned}$$

Ostatnia nierówność wynika z nierówności IV i drugiej z nierówności w 3^o.

$$4^o \quad H(R_{ab,j-1}) \leq H(R_{a,j}) \quad i \quad H(R_{b,j-1}) \leq H(R,j)$$

Druga z tych nierówności oraz nierówność IV dają:

$$H(R,j) \geq H(R_{a,j-1})$$

po podstawieniu do 3.13 dostajemy:

$$\begin{aligned} \Delta_j = H(R_{a,j}) - H(R,j) &= \max[H(R_{a,j-1}), H(R,j)] - H(R,j) + \\ &+ H(T_{aj}) \leq H(T_{aj}). \end{aligned}$$

W każdym zatem przypadku:

$$\Delta_j \leq H(T_{aj}).$$

Tp, że $B \Rightarrow C$ otrzymuje się natychmiast porównując obydwa kryteria. Należy jeszcze pokazać, że $C \Rightarrow D$, ale wiadomo z C, że:

$$H(R_{ab,j}) - H(R_{b,j}) \leq H(T_{aj})$$

czyli:

$$\begin{aligned} H(R_{ab,j}) &\leq H(R_{b,j}) + H(T_{aj}) \leq \max[H(R_{ab,j-1}), H(R_{b,j})] + \\ &+ H(T_{aj}) = H(R_{ba,j}). \end{aligned}$$

Co kończy dowód.

Wniosek.

Najwięcej funkcji sterujących można wyeliminować przy użyciu kryteriów D i potem kolejno C, B oraz A.

Wobec tego jednak, że dwie z tych metod: Dudka i Teutona oraz Bagga i Chakravarti mogą wyeliminować optymalną funkcję sterującą, celowym jest korzystanie z jednej z dwóch pierwszych metod.

Na zakończenie rozważań dotyczących metod eliminacyjnych zwróćmy uwagę na następujący fakt: na problem eliminacji można spojrzeć inaczej. Wystarczy bowiem liczyć wartość funkcjonału H "od końca", tzn. przyjmując za punkt wyjścia ostatni działający w kolejności operator i spróbować go wyeliminować z tej pozycji. Aby w tak postawionym zadaniu można było wyeliminować chociaż jedną funkcję sterującą, konieczny jest następujący warunek:

$$H(T_{an}) \leq H(T_{aj}) \quad j=1, 2, \dots, n-1$$

Prześledźmy jak działają metody eliminacyjne na następującym przykładzie:

$H(T_{ij}) =$

j \ i	1	2	3
1	2	5	6
2	4	5	2
3	6	3	4
4	3	7	5

Zastosujmy kryterium B. Zgodnie z nim można próbować wyeliminować wszystkie funkcje α , dla których $\alpha(1) = T_{1K}$ gdyż:

$$H(T_{11}) \leq H(T_{1j})$$

$$H(T_{11}) \leq H(T_{11}) \quad 1/$$

Przyjmujemy zatem $R = \emptyset$, $a=1$, wobec czego b może przyjmować 2,3 lub 4. Niech $b=2$. Dla wyznaczenia $\Delta_1, \Delta_2, \Delta_3$ trzeba wyliczyć

$$H(12,1) \quad , \quad H(12,2) \quad , \quad H(12,3)$$

oraz

$$H(2,1) \quad , \quad H(2,2) \quad , \quad H(2,3).$$

Na podstawie macierzy $H(T_{1j})$ wzorów 3.7 oraz 3.8 możemy zbudować poniższą tablicę:

j	1	2	3
$H(1,j)$	2	7	13
$H(12,j)$	6	12	15
$H(2,j)$	4	9	11

A zatem:

$$\Delta_1 = H(12,1) - H(2,1) = 6 - 4 = 2$$

$$\Delta_2 = H(12,2) - H(2,2) = 12 - 9 = 3$$

$$\Delta_3 = H(12,3) - H(2,3) = 15 - 11 = 4$$

1/ patrz 3.11 oraz 3.12

Czyli:

$$\Delta_1 \leq \Delta_2 \leq \Delta_3$$

Ponieważ równocześnie

$$\Delta_2 \leq H(T_{12})$$

oraz

$$\Delta_3 \leq H(T_{13})$$

kryterium B jest spełnione co oznacza, że funkcja α dla której $\alpha(1) = T_{2K}$ nie może być optymalną.

Przyjmijmy teraz $b=3$. Analogicznie jak poprzednio mamy:

j	1	2	3
H(1,j)	2	7	13
H(13,j)	8	11	17
H(3,j)	6	9	13

$$\Delta_1 = 2$$

$$\Delta_2 = 2$$

$$\Delta_3 = 4$$

a ponieważ

$$\Delta_2 \leq H(T_{12})$$

i

$$\Delta_3 \leq H(T_{13})$$

wnioskujemy w oparciu o kryterium eliminacyjne, że funkcja

α dla której $\alpha(1) = T_{3K}$ nie może być optymalną.

Przyjmijmy teraz $b=4$.

j	1	2	3
H(1,j)	2	7	13
H(14,j)	5	14	19
H(4,j)	3	10	15

$$\Delta_1 = 2$$

$$\Delta_2 = 4$$

$$\Delta_3 = 4$$

czyli

$$\Delta_1 \leq \Delta_2 \leq \Delta_3$$

i równocześnie

$$\Delta_2 \leq H(T_{12})$$

$$\Delta_3 \leq H(T_{13}).$$

Stąd wnioskujemy, że optymalna funkcja sterująca musi być taka, że:

$$\alpha(1) = T_{1K}.$$

Przystępując do eliminacji funkcji α ze względu na jej wartość dla argumentu 2 musimy przyjąć $R = 1$, $a = 4$ natomiast b może być równe 2 albo 3.

Wyznaczamy wartość:

$$H(142,1), \quad H(142,2), \quad H(142,3)$$

$$H(12,1), \quad H(12,2), \quad H(12,3)$$

Wartości te zapiszemy w tablicy:

j	1	2	3
H(1,j)	2	7	13
H(14,j)	5	14	19
H(142,j)	9	19	21
H(12,j)	6	12	15

Mamy wówczas:

$$\Delta_1 = 3$$

$$\Delta_2 = 7$$

$$\Delta_3 = 6$$

Jak z tego wyniku nie jest spełniony warunek B.a, zatem z drugiej pozycji nie można wyeliminować operatora T_{2K} . Postępując analogicznie można przekonać się, iż z drugiej pozycji nie można wyeliminować żadnego operatora, co oznacza, iż rozwiązania optymalnego należy poszukiwać wśród funkcji sterujących takich, dla których $\alpha(1) = T_{1K}$.

3.6. M e t o d y p r z y b l i ż o n e

3.6.1. Uwagi na temat metod przybliżonych

Jak wynika z dotychczasowego przeglądu, żadna z proponowanych metod nie może zostać uznana za metodę uniwersalną służącą do rozwiązywania zadań programowania sekwencyjnego. Wynika to z dwóch powodów:

- 1^o metody te odnoszą się do szczególnej klasy tych zadań,
- 2^o nawet dla tej szczególnej klasy zadań jeśli zadanie to będzie dużych rozmiarów, metody te zawodzą ze względu na ograniczoną pojemność pamięci komputera.

Wobec takiego stanu rzeczy powstała konieczność skonstruowania takich metod, które pozwalałyby wyznaczać rozwiązania zadań nawet kosztem ich dokładności. Mówiąc inaczej zrezygnowano z rozwiązania optymalnego uzyskując w zamian za to rozwiązanie "dobre" tzn. lepsze od przyjętego w sposób arbitralny. Metody te noszą nazwę "metod przybliżonych", gdyż prawie nigdy nie pozwalają wyznaczać rozwiązania optymalnego, a ponadto nie można oszacować jak dalece różni się uzyskane rozwiązanie od optymalnego. Tym niemniej ze względu na ich prostotę oraz szybkość otrzymywania rozwiązań są one jak do tej pory jedynymi efektywnymi stosowanymi do rozwiązywania zadań programowania sekwencyjnego. Wszystkie prezentowane tutaj metody przybliżone dotyczą zadań typu E opisanego w rozdziale pierwszym.

3.6.2. Metoda Pietrowa i Sokolicyna

Autorzy określają analityczny warunek, którego maksymalizacja ma być równoważna z minimalizacją funkcjonału H .

Warunek ten ma postać:

$$\sum_{i=1}^{m-1} (m-1) \left[\sum_{j=2}^n H(T_{ij}) - \sum_{j=1}^{n-1} H(T_{i+1,j}) \right]$$

Należy stwierdzić, że teza o równoważności minimalizacji funkcjonału H i maksymalizacji powyższej wartości jest nieprawdziwa. Tym niemniej autorzy w oparciu o tę tezę proponują wybierać funkcje sterujące zgodnie z poniższymi kryteriami:

I.

Wyznaczyć:

$$\sum_{j=2}^n H(T_{1j})$$

dla każdego $i=1,2,\dots,m$

Na podstawie tych sum wyznaczyć funkcję sterującą α , taką, że:

$$\alpha_{M1}(k) = T_{1_k K}, \quad k=1,2,\dots,m$$

przy czym operatory $T_{1_k K}$ muszą spełniać:

$$\sum_{j=2}^n H(T_{1_1 j}) \geq \sum_{j=2}^n H(T_{1_2 j}) \geq \dots \geq \sum_{j=2}^n H(T_{1_m j})$$

II.

Wyznaczyć:

$$\sum_{j=1}^{n-1} H(T_{ij})$$

dla każdego $i=1,2,\dots,m$

Na podstawie tych sum wybiera się funkcję α , taką, że:

$$\alpha_{M1}(k) = T_{1_k K} \quad k = 1,2,\dots,m$$

przy czym operatory $T_{i_k K}$ muszą spełniać:

$$\sum_{j=1}^{n-1} H(T_{i_1 j}) \leq \sum_{j=1}^{n-1} H(T_{i_2 j}) \leq \dots \leq \sum_{j=1}^{n-1} H(T_{i_m j})$$

III.

Wyznaczyć:

$$H(T_{i_n}) - H(T_{i_1})$$

dla każdego $i=1, 2, \dots, m$

Funkcja sterująca jest teraz taka, że:

$$\alpha_{M1}(k) = T_{i_k K} \quad k=1, 2, \dots, m$$

gdzie $T_{i_k K}$ spełniają:

$$H(T_{i_1 n}) - H(T_{i_1 1}) \geq H(T_{i_2 n}) - H(T_{i_2 1}) \geq \dots \geq H(T_{i_m n}) - H(T_{i_m 1})$$

IV.

Kryterium to stosuje się wówczas, gdy któraś z różnic wyznaczana dla kryterium trzeciego jest ujemna. Funkcja jest wtedy następująca:

$$\alpha_{M1}(k) = T_{i_k K} \quad k=1, 2, \dots, m$$

zaś $T_{i_k K}$ spełniają warunek:

$$|H(T_{i_1 n}) - H(T_{i_1 1})| \geq |H(T_{i_2 n}) - H(T_{i_2 1})| \geq \dots \geq |H(T_{i_m n}) - H(T_{i_m 1})|$$

W ten sposób mamy wyznaczonych co najmniej trzy funkcje sterujące spośród których wybiera się najlepszą tzn. taką,

która daje najmniejszą wartość funkcjonału H .

Zobaczmy to na przykładzie:

$H(T_{ij}) =$

$i \backslash j$	1	2	3	4
1	7	9	12	6
2	3	7	5	9
3	8	4	7	8
4	12	3	11	10
5	6	9	9	8

Wyznaczamy: $\sum_2^4 H(T_{ij})$

$$\sum H(T_{1j}) = 27$$

$$\sum H(T_{2j}) = 21$$

$$\sum H(T_{3j}) = 19$$

$$\sum H(T_{4j}) = 24$$

$$\sum H(T_{5j}) = 26$$

Na podstawie pierwszego kryterium mamy:

$$\alpha(1) = T_{1K}$$

$$\alpha(2) = T_{5K}$$

$$\alpha(3) = T_{4K}$$

$$\alpha(4) = T_{2K}$$

$$\alpha(5) = T_{3K}$$

Wartość funkcjonału H dla złożenia sterowanego tą funkcją wynosi 75.

Wyznaczamy teraz $\sum_1^3 H(T_{1j})$

$$\sum H(T_{1j}) = 28$$

$$\sum H(T_{2j}) = 15$$

$$\sum H(T_{3j}) = 19$$

$$\sum H(T_{4j}) = 26$$

$$\sum H(T_{5j}) = 24$$

Na podstawie drugiego kryterium mamy:

$$\alpha(1) = T_{2K}$$

$$\alpha(2) = T_{3K}$$

$$\alpha(3) = T_{5K}$$

$$\alpha(4) = T_{4K}$$

$$\alpha(5) = T_{1K}$$

Wartość funkcjonału dla złożenia sterowanego tą funkcją jest równa 64.

Wyznaczamy teraz: $H(T_{14}) - H(T_{11})$

$$H(T_{14}) - H(T_{11}) = -1$$

$$H(T_{24}) - H(T_{21}) = 6$$

$$H(T_{34}) - H(T_{31}) = 0$$

$$H(T_{44}) - H(T_{41}) = -2$$

$$H(T_{54}) - H(T_{51}) = 2.$$

Na podstawie kryterium trzeciego mamy:

$$\alpha(1) = T_{2K}$$

$$\alpha(2) = T_{5K}$$

$$\alpha(3) = T_{3K}$$

$$\alpha(4) = T_{1K}$$

$$\alpha(5) = T_{4K}$$

Wartość funkcjonału H jest tutaj równa 68.

Ponieważ dwie z powyższych różnic są ujemne, zastosujemy jeszcze kryterium czwarte.

Otrzymujemy wówczas dwie funkcje sterujące:

$$\alpha(1) = T_{2K} \quad \text{lub} \quad \alpha(1) = T_{2K}$$

$$\alpha(2) = T_{5K} \quad \alpha(2) = T_{4K}$$

$$\alpha(3) = T_{4K} \quad \alpha(3) = T_{5K}$$

$$\alpha(4) = T_{3K} \quad \alpha(4) = T_{3K}$$

$$\alpha(5) = T_{1K} \quad \alpha(5) = T_{1K}$$

Dla pierwszej z nich wartość funkcjonału wynosi 64, dla drugiej 66. Tak więc najlepszą funkcję sterującą uzyskaliśmy z drugiego i czwartego kryterium i jedną z nich przyjmujemy za rozwiązanie naszego zadania.

3.6.3. Metoda Pietrowa

Metoda Pietrowa w odróżnieniu od poprzedniej, daje tylko dwa kryteria wyboru funkcji sterującej.

Dla danego zadania znajdujemy:

$$b_{i1} = \sum_{j=1}^{\frac{n}{2}} H(T_{ij})$$

dla $i=1,2,\dots,m$

$$b_{i2} = \sum_{j=\frac{n}{2}+1}^n H(T_{ij})$$

Jeśli zaś n jest liczbą nieparzystą, to:

$$b_{i1} = \sum_{j=1}^{\frac{n+1}{2}} H(T_{ij})$$

dla $i=1,2,\dots,m$

$$b_{i2} = \sum_{j=\frac{n+1}{2}+1}^n H(T_{ij})$$

Następnie wyznacza się m różnic:

$$b_{i2} - b_{i1} .$$

W oparciu o te różnice Pietrow proponuje dwa kryteria wyboru funkcji.

I.

Wybrać taką funkcję α , by:

$$\alpha(k) = T_{i_k k} \quad k = 1, 2, \dots, m$$

przy czym $T_{i_k k}$ muszą spełniać:

$$b_{i_1 2} - b_{i_1 1} \geq b_{i_2 2} - b_{i_2 1} \geq \dots \geq b_{i_m 2} - b_{i_m 1}$$

II.

Wybrać taką funkcję sterującą, by:

$$\alpha(k) = T_{i_k K} \quad k=1,2,\dots,m$$

przy czym $T_{i_k K}$ spełniają warunek:

$$b_{i_1 1} \leq b_{i_2 1} \leq \dots \leq b_{i_l 1}$$

gdzie: $\{1,2,\dots,l\} = L$

$$i \quad L = \{k \in M; \quad b_{i_k 2} - b_{i_k 1} \geq 0\}.$$

Natomiast dla $k=l+1, l+2, \dots, m$

$T_{i_k K}$ spełniają warunek:

$$b_{i_{l+1} 2} \geq b_{i_{l+2} 2} \geq \dots \geq b_{i_m 2}.$$

Zobaczmy jak działa ta metoda na poprzednim przykładzie:

Wyznaczamy:

$b_{11} = 16$	$b_{12} = 18$	$b_{12} - b_{11} = 2$
$b_{21} = 10$	$b_{22} = 14$	$b_{22} - b_{21} = 4$
$b_{31} = 12$	$b_{32} = 15$	$b_{32} - b_{31} = 3$
$b_{41} = 15$	$b_{42} = 21$	$b_{42} - b_{41} = 6$
$b_{51} = 15$	$b_{52} = 17$	$b_{52} - b_{51} = 2$

Na podstawie pierwszego kryterium dostajemy dwie funkcje:

$$\alpha(1) = T_{4K} \quad \text{i drugą} \quad \alpha(1) = T_{4K}$$

$$\alpha(2) = T_{2K} \quad \alpha(2) = T_{2K}$$

$$\alpha(3) = T_{3K} \quad \alpha(3) = T_{3K}$$

$$\alpha(4) = T_{1K}$$

$$\alpha(4) = T_{5K}$$

$$\alpha(5) = T_{5K}$$

$$\alpha(5) = T_{1K}$$

Dla pierwszej z tych funkcji wartość funkcjonału H wynosi 68, dla drugiej zaś wartość ta wynosi 67.

Drugie kryterium daje:

$$\alpha(1) = T_{2K}$$

lub

$$\alpha(1) = T_{2K}$$

$$\alpha(2) = T_{3K}$$

$$\alpha(2) = T_{3K}$$

$$\alpha(3) = T_{4K}$$

$$\alpha(3) = T_{5K}$$

$$\alpha(4) = T_{5K}$$

$$\alpha(4) = T_{4K}$$

$$\alpha(5) = T_{1K}$$

$$\alpha(5) = T_{1K}$$

Pierwsza z tych funkcji daje wartość funkcjonału H równą 65, druga natomiast 64. Tak więc druga z tych funkcji jest rozwiązaniem naszego zadania.

3.6.4. Metoda Palmera

Inną metodą wyznaczenia funkcji sterującej "w przybliżeniu optymalnej" jest metoda zaproponowana przez Palmera.

Zgodnie z tą metodą, dla każdego podzbioru T_{iK} operatorów wyznacza się tzw. "wskaźnik nachylenia" według wzoru:

$$s_i = - \frac{n-1}{2} H(T_{i1}) - \frac{n-3}{2} H(T_{i2}) \dots + \frac{n-3}{2} H(T_{i,n-1}) + \\ + \frac{n-1}{2} H(T_{in}).$$

$$i = 1, 2, \dots, m$$

Następnie wyznacza się funkcję sterującą α taką, że:

$$\alpha(k) = T_{i_k K}$$

gdzie $T_{i_k K}$ spełniają warunek

$$s_{i_1} \succcurlyeq s_{i_2} \succcurlyeq \dots \succcurlyeq s_{i_n}.$$

Dla naszego przykładu mamy:

$$s_1 = -10,5 - 4,5 + 6 + 9 = 0$$

$$s_2 = - 4,5 - 3,5 + 2,5 + 13,5 = 8$$

$$s_3 = -12, - 4 + 3,5 + 12 = -0,5$$

$$s_4 = -18 - 1,5 + 5,5 + 15 = 1$$

$$s_5 = -9 - 4,5 + 4,5 + 12 = 3$$

Czyli funkcja sterująca musi być następująca:

$$\alpha(1) = T_{2K}$$

$$\alpha(2) = T_{5K}$$

$$\alpha(3) = T_{4K}$$

$$\alpha(4) = T_{1K}$$

$$\alpha(5) = T_{3K}$$

i wartość funkcjonału H jest wówczas równa 66.

3.6.5. Metoda Campbella, Dudka i Smitha

Autorzy tej metody proponują do wyznaczania funkcji "dobrej" konstruowanie $n-1$ zadań pomocniczych. Konstrukcja r -tego zadania pomocniczego jest następująca:

$$M_1 \begin{matrix} r \\ i \end{matrix} = \sum_{j=1}^r H(T_{ij})$$

$$M_2 \begin{matrix} r \\ i \end{matrix} = \sum_{j=n+1-r}^n H(T_{ij})$$

gdzie $r = 1, 2, \dots, n-1$.

Jest to redukcja zadania o n warstwach operatorów do zadania o dwóch warstwach. Dla każdego z tych zadań wyznacza się rozwiązania korzystając z algorytmu Johnsona przytoczonego na początku tego rozdziału. Spośród uzyskanych w ten sposób $n-1$ funkcji sterujących wybiera się najlepszą, która jest rozwiązaniem wyjściowego zadania. Oczywiście rozwiązaniem to, jest rozwiązaniem przybliżonym.

Zastosujmy tę metodę do naszego przykładu.

$r=1$

$$M_1 \begin{matrix} 1 \\ 1 \end{matrix} = 7$$

$$M_2 \begin{matrix} 1 \\ 1 \end{matrix} = 6$$

$$M_1 \begin{matrix} 1 \\ 2 \end{matrix} = 3$$

$$M_2 \begin{matrix} 1 \\ 2 \end{matrix} = 9$$

$$M_1 \begin{matrix} 1 \\ 3 \end{matrix} = 8$$

$$M_2 \begin{matrix} 1 \\ 3 \end{matrix} = 8$$

$$M_1 \begin{matrix} 1 \\ 4 \end{matrix} = 12$$

$$M_2 \begin{matrix} 1 \\ 4 \end{matrix} = 10$$

$$M_1 \begin{matrix} 1 \\ 5 \end{matrix} = 6$$

$$M_2 \begin{matrix} 1 \\ 5 \end{matrix} = 8$$

$r=2$

$$M_1 \begin{matrix} 2 \\ 1 \end{matrix} = 16$$

$$M_2 \begin{matrix} 2 \\ 1 \end{matrix} = 18$$

$$M_1 \begin{matrix} 2 \\ 2 \end{matrix} = 10$$

$$M_2 \begin{matrix} 2 \\ 2 \end{matrix} = 14$$

$$M_1 \quad \begin{matrix} 2 \\ 3 \end{matrix} = 12$$

$$M_2 \quad \begin{matrix} 2 \\ 3 \end{matrix} = 15$$

$$M_1 \quad \begin{matrix} 2 \\ 4 \end{matrix} = 15$$

$$M_2 \quad \begin{matrix} 2 \\ 4 \end{matrix} = 21$$

$$M_1 \quad \begin{matrix} 2 \\ 5 \end{matrix} = 15$$

$$M_2 \quad \begin{matrix} 2 \\ 5 \end{matrix} = 17$$

r=3

$$M_1 \quad \begin{matrix} 3 \\ 1 \end{matrix} = 28$$

$$M_2 \quad \begin{matrix} 3 \\ 1 \end{matrix} = 27$$

$$M_1 \quad \begin{matrix} 3 \\ 2 \end{matrix} = 15$$

$$M_2 \quad \begin{matrix} 3 \\ 2 \end{matrix} = 21$$

$$M_1 \quad \begin{matrix} 3 \\ 3 \end{matrix} = 19$$

$$M_2 \quad \begin{matrix} 3 \\ 3 \end{matrix} = 19$$

$$M_1 \quad \begin{matrix} 3 \\ 4 \end{matrix} = 26$$

$$M_2 \quad \begin{matrix} 3 \\ 4 \end{matrix} = 24$$

$$M_1 \quad \begin{matrix} 3 \\ 5 \end{matrix} = 24$$

$$M_2 \quad \begin{matrix} 3 \\ 5 \end{matrix} = 26.$$

Pierwsze zadanie pomocnicze daje dwie funkcje sterujace:

$$\alpha(1) = T_{2K} \quad \text{oraz} \quad \alpha(1) = T_{2K}$$

$$\alpha(2) = T_{5K} \quad \alpha(2) = T_{5K}$$

$$\alpha(3) = T_{3K} \quad \alpha(3) = T_{4K}$$

$$\alpha(4) = T_{4K} \quad \alpha(4) = T_{3K}$$

$$\alpha(5) = T_{1K} \quad \alpha(5) = T_{1K}$$

Wartosci funkcjonalu H sa tutaj odpowiednio rowne 64 i 64.

Drugie zadanie pomocnicze daje:

$$\alpha(1) = T_{2K} \quad \text{oraz} \quad \alpha(1) = T_{2K}$$

$$\alpha(2) = T_{3K} \quad \alpha(2) = T_{3K}$$

$$\alpha(3) = T_{4K} \quad \alpha(3) = T_{5K}$$

$$\alpha(4) = T_{5K} \quad \alpha(4) = T_{4K}$$

$$\alpha(5) = T_{1K}$$

$$\alpha(5) = T_{1K}$$

Odpowiednie wartości funkcjonału wynoszą 65 i 64.

Trzecie zadanie pomocnicze daje:

$$\alpha(1) = T_{2K}$$

oraz

$$\alpha(1) = T_{2K}$$

$$\alpha(2) = T_{3K}$$

$$\alpha(2) = T_{5K}$$

$$\alpha(3) = T_{5K}$$

$$\alpha(3) = T_{1K}$$

$$\alpha(4) = T_{1K}$$

$$\alpha(4) = T_{4K}$$

$$\alpha(5) = T_{4K}$$

$$\alpha(5) = T_{3K}$$

Wartości funkcjonału są tutaj kolejno 68 i 69.

Tak więc rozwiązaniem zadania może być jedna z trzech funkcji sterujących dająca wartość funkcjonału H równą 64.

3.6.6. Metoda Tiutiukina

Tiutiukin proponuje dla rozwiązania zadania wyznaczenie optymalnej funkcji sterującej stosować następujące postępowanie. Zadanie o n warstwach operatorów sprowadzić do zadania o dwóch warstwach w sposób następujący:

$$M_{i1} = \sum_{j=1}^{n-1} H(T_{ij})$$

$$M_{i2} = \sum_{j=2}^n H(T_{ij}) \quad i = 1, 2, \dots, m$$

Jest to $n-1$ zadanie pomocnicze z poprzedniej metody. Dla tych dwóch warstw po zastosowaniu algorytmu Johnsona uzysku-

je się optymalną funkcję sterującą, która jest równocześnie przybliżonym rozwiązaniem zadania wyjściowego. Autor podaje również warunek dostateczny na to by uzyskane tą drogą rozwiązanie było optymalne.

Wprowadźmy za autorem zbiór U , który jest zbiorem wektorów:

$$u = (u_0, u_1, \dots, u_n)$$

o składowych będących liczbami naturalnymi i spełniającymi następujący warunek:

$$1 = u_0 \leq u_1 \leq \dots \leq u_n = m.$$

Niech teraz α będzie funkcją sterującą wyznaczoną za pomocą omawianej metody. Wartość funkcjonału H dla złożenia sterowanego tą funkcją wyznaczyć można ze wzoru:

$$H(T_{MK\alpha}) = \max_{u \in U} \sum_{j=1}^n \sum_{k=u_{j-1}}^{u_j} H(T_{i_{kj}})$$

Wspomniany warunek jest następujący:

Funkcja α jest optymalna, jeśli:

$$H(T_{MK\alpha}) = \sum_{j=1}^n \sum_{k=u_{j-1}}^{u_j} H(T_{i_{kj}}),$$

gdzie:

$$u_1 = u_2 = \dots = u_{n-1}.$$

3.6.7. Metoda Tiutiukina II

Druga z metod zaproponowana przez Tiutiukina jest skonstruowana następująco.

Ze zbioru U wybieramy w sposób arbitralny wektor:

$$\bar{u} = \bar{u}_0, \bar{u}_1, \dots, \bar{u}_n$$

Następnie stosujemy opisane poniżej postępowanie iteracyjne. Cechą charakterystyczną tego postępowania jest to, że r -ty ($r=1, 2, \dots, m$) krok iteracyjny kończy się uzyskaniem wartości funkcji α dla argumentów od 1 do r . Przy czym funkcji tych jest $\binom{m}{r}$. Innymi słowy mamy ustalonych r pierwszych operatorów w złożeniu $W_r = (T_{i_r K} \times T_{i_{r-1} K} \times \dots \times T_{i_2 K} \times T_{i_1 K})$ i złożzeń tych jest $\binom{m}{r}$. Każde dwa z tych złożzeń różnią się między sobą co do "składu" operatorów. Tzn. jeśli uzyskaliśmy dwa złożenia W_r^{\wedge} i $W_r^{\wedge\wedge}$ to istnieje co najmniej jeden taki operator T_{iK} , który występuje w złożeniu W_r^{\wedge} a nie występuje w złożeniu $W_r^{\wedge\wedge}$.

Postępowanie iteracyjne można opisać następująco:

Krok 1

Tworzymy złożenia jednoelementowe $W_1 = T_{i_1 K}$ przyjmując za i_1 kolejno $1, 2, \dots, m$.

Krok r $r=2, 3, \dots, m$

Danych jest $\binom{m}{r-1}$ złożzeń postaci:

$$W_{r-1} = (T_{i_{r-1} K} \times T_{i_{r-2} K} \times \dots \times T_{i_2 K} \times T_{i_1 K})$$

przy czym każde dwa z nich różnią się co do "składu".

1.

Dla każdego ze złożzeń W_{r-1} tworzymy m_{r+1} złożzeń postaci:

$$W_r = (T_{i_r K} \times T_{i_{r-1} K} \times \dots \times T_{i_1 K})$$

przyjmując za $T_{i_r K}$ kolejno wszystkie operatory nie należące do W_{r-1} .

2.

Uzyskane złożzenia dzielimy na $\binom{m}{r}$ grup takich, że w każdej grupie wszystkie złożzenia zawierają te same operatory, różnią się tylko kolejnością.

3.

W każdej z grup wybieramy jedno złożenie, a mianowicie to, które minimalizuje wyrażenie:

$$\sum_{j=1}^n \sum_{k=\bar{u}_{j-1}}^{\bar{u}_j} H(T_{i_{kj}})$$

Pozostałe złożenia eliminujemy.

Tiutiukin dowodzi, że na to, aby złożenie:

$$W_m = (T_{i_m K} \times \dots \times T_{i_2 K} \times T_{i_1 K})$$

było optymalne (funkcja sterująca α dająca złożenie W_m była optymalna) wystarcza, że:

$$H(T_{MK}^{\alpha}) = \sum_{j=1}^n \sum_{k=\bar{u}_{j-1}}^{\bar{u}_j} H(T_{i_{kj}})$$

gdzie wektor \bar{u} jest wybranym na początku wektorem ze zbioru

ru U. Wydaje się jednak, że metoda ta jak na metodę przybliżoną jest zbyt skomplikowana.

3.6.8. Metoda wzajemnego poprzedzania

Inną metodę rozwiązywania zadania zaproponował W. Sikora. Istota tej metody polega na tym, że wybiera się dwa dowolne operatory $T_{i_1 K}$ oraz $T_{i_2 K}$. Następnie określa się który z nich powinien w złożeniu będącym rozwiązaniem zadania działać wcześniej.

Autor metody proponuje, by:

$$\alpha^{-1} (T_{i_1 K}) < \alpha^{-1} (T_{i_2 K}) \Leftrightarrow H(i_1 i_2, n) < H(i_2 i_1, n) \quad 1/$$

Po ustaleniu wzajemnego poprzedzania w złożeniu dla tych dwóch operatorów wybiera się w sposób arbitralny z pozostałych operator $T_{i_3 K}$. Ponieważ operator $T_{i_1 K}$ ma w złożeniu działać wcześniej niż operator $T_{i_2 K}$ więc dołączenie operatora $T_{i_3 K}$ może dawać tylko poniższe złożenia:

$$(T_{i_3 K} \times T_{i_2 K} \times T_{i_1 K}), (T_{i_2 K} \times T_{i_3 K} \times T_{i_1 K}), (T_{i_2 K} \times T_{i_1 K} \times T_{i_3 K}).$$

Dla każdego z tych złożzeń wyznacza się wartość funkcjonału H do n-tej warstwy włącznie. To złożenie, które daje najmniejszą wartość H określa wzajemne poprzedzenie tych

1/ patrz oznaczenia w "metody eliminacyjne".

trzech operatorów w końcowym złożeniu. Następnie dołącza się nowy operator i w analogiczny sposób określa się wzajemne poprzedzanie dla tych operatorów. Po $n-1$ krokach otrzymuje się rozwiązanie zadania.

3.6.9. Metoda uśredniania miar jednoczesności działania I

Metoda ta zaproponowana przez Jankowską-Zorychtę za podstawę konstrukcji funkcji sterującej przyjmuje macierz, którą za autorką nazwiemy macierzą jednoczesności działania operatorów. Macierz tę oznaczmy A .

Dla zbudowania tej macierzy należy dla każdego podzbioru operatorów T_{iK} ($i=1,2,\dots,m$) wyznaczyć następujące wielkości:

$$c_{ik} = \sum_{j=1}^{k-1} H(T_{ij})$$

$$d_{ik} = \sum_{k=j+1}^n H(T_{ij})$$

$$k=1,2,\dots,n$$

przyjmujemy przy tym:

$$\sum_{j=r}^s = 0 \quad \text{dla } s < r.$$

Następnie wyznaczamy macierz $A = [a_{ip}]$, gdzie:

$$a_{lp} = \begin{cases} \min_k (d_{lk} + c_{pk}), & \text{dla } l \neq p \\ 0 & \text{dla } l=p, p, l=1,2,\dots,m \end{cases}$$

Obliczamy następnie:

$$\lambda_j = \frac{1}{m-j} \sum_{i=j+1}^m a_{ji} \quad j=1,2,\dots,m-1$$

$$\mu_j = \frac{1}{m-j} \sum_{i=j+1}^m a_{ij} \quad j=1,2,\dots,m-1$$

Wielkości λ_j oraz μ_j są średnimi arytmetycznymi odpowiednio elementów wiersza lub kolumny licząc od głównej przekątnej do końca. Wielkości te są podstawą konstrukcji funkcji sterującej. Zasadę tej konstrukcji przedstawimy w postaci algorytmu:

Krok 1.

$$l:=0, \quad k:=0$$

przejdź do kroku drugiego.

Krok 2.

Dla $j=1,2,\dots,m-1$ wykonaj:

$$\text{jeśli } \lambda_j \geq \mu_j, \text{ to } l:=l+1 \text{ i } \alpha(l) = T_{jK}$$

$$\text{jeśli } \lambda_j < \mu_j, \text{ to } \alpha(m-k) = T_{jK}, \quad k:=k+1.$$

Przejdź do kroku trzeciego.

Krok 3.

$$\alpha(m-k) = T_{mK}$$

Autorka tej metody pisze: "W opisanym algorytmie istotne jest badanie warunku $\lambda_j < \mu_j$. Ze względu na uśredniane współczynniki nierówności tej nie należy traktować zbyt rygorystycznie. Oznacza to, że jeśli λ_j różni się względnie niewiele od μ_j to można przyjąć w warunku tym równość" ^{1/}. Konsekwencją tego jest fakt, iż w wyniku postępowania zaproponowanego w tej metodzie uzyskuje się więcej niż jedną funkcję sterującą, gdyż jeśli różnica między λ_j a μ_j jest niewielka to można przyjąć raz $\lambda_j < \mu_j$ a drugi raz $\lambda_j \geq \mu_j$.

Zastosujemy opisaną metodę do naszego przykładu. Wyznaczymy c_{ik} oraz d_{ik} a wyniki zanotujemy w macierzy:

k	1		2		3		4	
1	c_{i1}	d_{i1}	c_{i2}	d_{i2}	c_{i3}	d_{i3}	c_{i4}	d_{i4}
1	0	27	7	18	16	6	28	0
2	0	21	3	14	10	9	15	0
3	0	19	8	15	12	8	19	0
4	0	24	12	21	15	10	26	0
5	0	26	6	17	15	8	24	0

Wobec tego macierz A jest następująca:

1/ patrz Jankowska-Zorychta "Modele sekwencyjne i ich zastosowanie..." str.82.

	1	2	3	4	5	λ_j
1	0	15	18	21	21	18,7
2	11	0	19	21	20	20
3	19	15	0	19	19	19
4	24	15	19	0	24	24
5	24	15	19	23	0	-
μ_j	19,2	15	19	23	-	-

Widać, że: $\lambda_1 < \mu_1$, $\lambda_2 > \mu_2$, $\lambda_3 \gg \mu_3$, $\lambda_4 > \mu_4$
 przy czym różnica między λ_1 a μ_1 jest względnie nie-
 wielka możemy zatem otrzymać dwa złożenia:

I.

$$(T_{1K} \times T_{5K} \times T_{4K} \times T_{3K} \times T_{2K})$$

i II.

$$(T_{5K} \times T_{4K} \times T_{3K} \times T_{2K} \times T_{1K})$$

Pierwsze z nich daje wartość funkcjonau równą 65, natomiast drugie 69.

Zwróćmy uwagę na fakt, iż w tej metodzie na ostateczny wynik ma istotny wpływ kolejność operatorów, dla jakiej zbudowano macierz A. Podobnie rzecz wygląda w metodzie wzajemnego poprzedzania, gdzie w zależności od wyboru operatorów można uzyskiwać różne funkcje sterujące. Do zagadnienia tego jeszcze powrócimy.

3.6.10. Metoda uśredniania miar jednoczesności działania II

Metoda ta również zaproponowana przez Jankowską-Zorychtę podobnie jak i poprzednia za podstawę konstrukcji funkcji sterującej przyjmuje macierz A . Jest to postępowanie iteracyjne, którego cechą charakterystyczną jest to, że w każdym kroku iteracyjnym wybiera się jeden operator, który wstawia się do złożenia w odpowiednim miejscu, a przechodząc do kroku następnego rozpatruje się zbiór operatorów bez badanego w kroku poprzednim. Tym samym dziedzina funkcji sterującej po każdym kroku ulega zmianie a ściślej mówiąc redukcji.

Oznaczmy symbolem M_r dziedzinę funkcji \mathcal{C} po r -tym kroku iteracyjnym ($r=1,2,\dots,m-1$) oraz $m_r = \max M_r$, $\bar{m}_r = \min M_r$. Postępowanie to opiszemy w kolejnych krokach:

Krok 1.

Wyznaczamy:

$$\lambda_i^{(1)} = \sum_{j=1}^m a_{ij} \quad i=1,2,\dots,m$$

$$\mu_j^{(1)} = \sum_{i=1}^m a_{ij} \quad j=1,2,\dots,m$$

Niech

$$\lambda_{k_1}^{(1)} = \min_i \lambda_i^{(1)}$$

$$\mu_{l_1}^{(1)} = \min_j \mu_j^{(1)}$$

oznaczmy:

$$p_1 = \begin{cases} k_1, & \text{jeśli } \lambda_{k_1}^{(1)} \leq \mu_{l_1}^{(1)} \\ l_1, & \text{jeśli } \lambda_{k_1}^{(1)} > \mu_{l_1}^{(1)}. \end{cases}$$

Jeśli teraz:

$$p_1 = k_1$$

to

$$T_{p_1 K} = \alpha(m)$$

Jeśli zaś:

$$p_1 = l_1$$

to

$$T_{p_1 K} = \alpha(1)$$

i

$$M_1 = M - \{p_1\}.$$

Krok r.

Wyznaczamy:

$$\lambda_i^{(r)} = \lambda_j^{(r-1)} - a_{i, p_{r-1} j} \quad i \in M_r$$

$$\mu_j^{(r)} = \mu_j^{(r-1)} - a_{p_{r-1} j} \quad j \in M_r$$

Wyznaczamy

$$\lambda_{k_r}^{(r)} = \min_i \lambda_i^{(r)}$$

$$\mu_{l_r}^{(r)} = \min_j \mu_j^{(r)}$$

Oznaczamy:

$$p_r = \begin{cases} k_r, & \text{jeśli } \lambda_{k_r}^{(r)} \leq \mu_{l_r}^{(r)} \\ l_r, & \text{jeśli } \lambda_{k_r}^{(r)} > \mu_{l_r}^{(r)} \end{cases}$$

Jeśli:

$$p_r = k_r$$

to:

$$T_{p_r K} = \alpha(m_r)$$

Jeśli natomiast:

$$p_r = l_r$$

to:

$$T_{p_r K} = \alpha(\bar{m}_r).$$

Krok m.

Operator, który pozostał, umieszczamy na jedynym wolnym miejscu w złożeniu.

Jak z tego widać budowanie złożenia, które ma być rozwiązaniem zadania odbywa się z dwóch stron. W zależności od warunków operator $T_{p_r K}$ zostaje wstawiony w złożeniu na pierwsze wolne miejsce bądź z lewej, bądź prawej strony.

Postępowanie to prześledźmy na przykładzie, którym posługujemy się cały czas. Cały proces zapiszemy w postaci macierzy:

							r					
		1	2	3	4	5	1	2	3	4		
i	j											
	1	0	15	18	21	21	75	60	<u>42</u>			
	2	11	0	19	21	20	71					
	3	19	15	0	19	19	72	57				
	4	24	15	19	0	24	82	67	48	24		
5	24	15	19	23	0	81	66	47	<u>23</u>			
r	1	78	<u>60</u>	75	84	84	$\lambda_i^{(r)}$					
	2	67		<u>56</u>	63	64						
	3	48			44	45						
	4				<u>23</u>	24						

W tablicy tej podkreślono $\min(\lambda_i^{(r)}, \mu_j^{(r)})$

Sukcesywne powstawanie złożenia ilustruje poniższa tabelka:

M	1	2	3	4	5	p_r
1					T_{2K}	2
2				T_{3K}	T_{2K}	3
3	T_{1K}			T_{3K}	T_{2K}	1
4	T_{1K}	T_{4K}		T_{3K}	T_{2K}	4
5	T_{1K}	T_{4K}	T_{5K}	T_{3K}	T_{2K}	

Wartość funkcjonału H dla tego złożenia jest równa 64.

3.6.11. Metoda dróg hamiltonowskich

Obecnie chcemy zaproponować pewną procedurę wyznaczania funkcji sterującej, której idea oparta jest na wyznaczaniu drogi hamiltonowskiej w grafie. Przypomnijmy następującą definicję.

Dany jest graf $G = (X, U)$, gdzie X jest zbiorem wierzchołków tego grafu, U zbiorem jego łuków. Załóżmy przy tym, że:

$$\text{card}(X) = r.$$

Definicja.

Drogą hamiltonowską grafu G nazywamy taki ciąg jego różnych wierzchołków:

$$x_{i_1}, x_{i_2}, \dots, x_{i_r}$$

że:

$$(x_{i_s}, x_{i_{s+1}}) \in U \quad \text{dla każdego } s=1, 2, \dots, r-1.$$

Niech symbol $H(i_s, i_l; n)$ oznacza wartość funkcjonału H dla złożenia $(T_{i_l K} \times T_{i_s K})$ liczoną do n -tej warstwy włącznie.

Graf odpowiadający zadaniu budujemy jak następuje:

Każdy z wierzchołków grafu utożsamiamy z jednym z podzbiorów $T_{i K}$ zbioru operatorów $i=1, 2, \dots, m$.

Dla znalezienia zbioru łuków tego grafu wyznaczamy:

$$H(i_s, i_l; n) \quad s=1, 2, \dots, m \quad l=1, 2, \dots, m \quad l \neq s.$$

i

$$(i_s, i_l) \in U \iff H(i_s, i_l; n) \leq M(i_l, i_s; n)$$

$$(i_l, i_s) \in U \iff H(i_s, i_l; n) > H(i_l, i_s; n).$$

Jak widać z konstrukcji otrzymany graf jest grafem pełnym tzn. dla każdego s oraz l zachodzi:

$$(i_s, i_l) \in U \vee (i_l, i_s) \in U.$$

Niech

$$i_1, i_2, \dots, i_n$$

będzie drogą hamiltonowską w tym grafie. Ponieważ graf jest pełny, droga taka zawsze istnieje^{1/}.

$$H(i_1, i_2; n) < H(i_2, i_1; n)$$

$$H(i_2, i_3; n) < H(i_3, i_2; n)$$

⋮

$$H(i_{m-1}, i_m; n) < H(i_m, i_{m-1}; n)$$

złożenie:

$$T_{i_m K} \times \dots \times T_{i_2 K} \times T_{i_1 K}$$

powinno dawać wartość funkcjonału H bliską optymalnej.

Tak też jest w istocie. Okazuje się jednak, że rozwiązanie to można poprawić zmieniając w grafie G orientację jednego łuku. Zmiana ta polegać będzie na budowie nowego grafu

$G_1 = (X, U_1)$, który różni się od grafu G tym, że:

istnieje dokładnie jedna para (s, l) taka, że:

$$(i_s, i_l) \in U \Rightarrow (i_l, i_s) \in U_1.$$

^{1/} patrz Berge "Teoria grafów i jej primienienija" str. 123

Ponieważ nie można przewidzieć dla którego łuku należy zmienić orientację, procedurę budowania nowego grafu należy powtarzać aż do chwili wyczerpania wszystkich możliwości, tzn. po przebadaniu $\binom{m}{2}$ grafów. Metoda ta daje bardzo dobre wyniki^{1/} jednak wadą jej jest zbyt mała szybkość. Szybkość tej metody limitowana jest algorytmem wyznaczania dróg hamiltonowskich w grafie.

Zobaczmy na naszym przykładzie jak działa ta metoda.

Wyznaczamy:

$$H(1,2;4) = 43$$

$$H(2,1;4) = 37$$

$$H(1,3;4) = 43$$

$$H(3,1;4) = 42$$

$$H(1,4;4) = 49$$

$$H(4,1;4) = 46$$

$$H(1,5;4) = 45$$

$$H(5,1;4) = 42$$

$$H(2,3;4) = 32$$

$$H(3,2;4) = 36$$

$$H(2,4;4) = 39$$

$$H(4,2;4) = 45$$

$$H(2,5;4) = 36$$

$$H(5,2;4) = 41$$

$$H(3,4;4) = 44$$

$$H(4,3;4) = 44$$

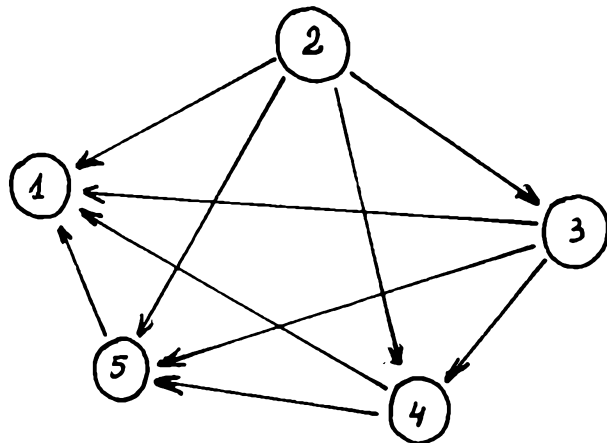
$$H(3,5;4) = 40$$

$$H(5,3;4) = 40$$

$$H(4,5;4) = 44$$

$$H(5,4;4) = 45$$

Graf odpowiadający naszemu zadaniu jest następujący:



^{1/} patrz dodatek.

Droga hamiltonowska w tym grafie:

2,3,4,5,1

Odpowiadające jej złożenie: $(T_{1K}xT_{5K}xT_{4K}xT_{3K}xT_{2K})$ daje wartość funkcjonału H równą 65.

Optymalne rozwiązanie uzyskuje się zmieniając łuk (3,5) na łuk (5,3) i wówczas złożenie: $(T_{1K}xT_{4K}xT_{3K}xT_{5K}xT_{2K})$ jest optymalnym o wartości funkcjonału H równej 64.

3.6.12. Metoda Monte Carlo

Chcemy teraz do rozwiązania zadań programowania sekwencyjnego zaproponować metodę bazującą na technice Monte Carlo. W literaturze spotkać można propozycje stosowania metod Monte Carlo do zadań tego typu, jednak propozycje te są mało konstruktywne. Procedurę Monte Carlo stosuje się bowiem do szacowania pewnych parametrów zmiennej losowej jaką jest tutaj wartość funkcjonału $H^{1/}$, bądź też wychodząc od jednego dowolnego rozwiązania przeszukuje się zbiór rozwiązań najbliższych mu, definiując oczywiście, odległość między rozwiązaniami^{2/}.

Nie będziemy tu przedstawiali szczegółowo samej techniki Monte Carlo bowiem literatura na ten temat jest obszerna, zacytujmy jedynie "Metoda ta polega na zastąpieniu rozwiązań analitycznych czy obliczeń numerycznych obserwacją odpowiedniego modelu statystycznego"^{3/}.

^{1/} patrz "Kalendarnoje planowanie".

^{2/} patrz A.Cichocki "Optymalizacja w deterministycznych modelach produkcyjnych".

^{3/} patrz Buslenko i inni "Metoda Monte Carlo" z przedmowy wydawcy wydania polskiego.

Zastosowanie metody Monte Carlo do zadań programowania sekwencyjnego polega na tym, że z populacji, którą jest zbiór rozwiązań dopuszczalnych w sposób losowy generuje się pewien jego podzbiór i najlepsze rozwiązanie z tego podzbioru przyjmuje się za rozwiązanie zadania.

Jest rzeczą oczywistą, że schemat losowania musi gwarantować to, że każde rozwiązanie ze zbioru rozwiązań dopuszczalnych ma jednakowe prawdopodobieństwo trafienia do wylosowanego podzbioru. Zabezpieczają to generatory liczb losowych z rozkładem jednostajnym. Jednym z takich generatorów jest generator biblioteczny "unif". Wprowadzając generatory te działają na zbiorach ciągłych, a w zadaniach sekwencyjnych mają do czynienia ze zbiorami dyskretnymi, to jednak niewielka modyfikacja tych generatorów pozwala pracować na zbiorach dyskretnych, a więc takich jak trzeba.

Warto tu zwrócić uwagę na jeszcze jeden fakt, ponieważ zbiór rozwiązań dopuszczalnych jest zbiorem skończonym należałoby zatem stosować schemat losowania bez zwracania, ale wobec dużej liczności tego zbioru można zupełnie bezpiecznie stosować prostszy schemat losowania ze zwracaniem.

Jak wynika z rozważań rozdziału drugiego wartość funkcjonału H zależy od funkcji ϕ , a ta zaś z kolei od funkcji sterujących α i β , najważniejszą zatem sprawą jest wybór tychże funkcji.

wprowadźmy jeszcze wygodną terminologię. Zbiór rozwiązań dopuszczalnych nazwiemy populacją, jedno rozwiązanie dopuszczalne nazywać będziemy elementem populacji, r elementów populacji nazywać będziemy próbką r -elementową.

Proponowana procedura stosowania techniki Monte Carlo do naszych zadań składa się z następujących etapów:

- losuje się próbkę r -elementową,
- wyznacza się element, dla którego wartość funkcjonału H jest najmniejsza w tej próbce,
- element ten przyjmuje się za rozwiązanie zadania,
- losuje się następną próbkę i wyznacza się element minimalizujący wartość funkcjonału H w tej próbce,
- jeśli wartość funkcjonału H jest mniejsza niż wartość funkcjonału dla poprzednio otrzymanego rozwiązania to ten nowy element przyjmuje się za rozwiązanie zadania,
- jeśli natomiast element ten daje wartość funkcjonału nie mniejszą od poprzednio wylosowanego rozwiązania to notujemy niepowodzenie,
- eksperyment przerywamy gdy uzyska się kolejno c niepowodzeń.

Należy zaznaczyć, że r oraz c są z góry, arbitralnie dobranymi liczbami.

Jak z tego opisu widać całe zagadnienie sprowadza się do wylosowania elementu populacji. Ponieważ dla każdego typu zadania odbywa się to nieco inaczej, omówimy teraz kolejno sposób generowania rozwiązania dopuszczalnego dla każdego z tych typów, rozpoczynając opis od zadania najprostszego, a więc typu E.

Symulacja zadania typu E.

Przypomnijmy, że w zadaniach tego typu rozwiązanie sprowadza się do wyznaczenia funkcji sterującej α dla dowolnej warstwy, bowiem w pozostałych warstwach operatorów sterowanie

jest analogiczne, ponadto operatorów w każdej warstwie jest tyle, jaka jest liczność zbioru S . Ponieważ jest ich m , rozwiązaniem zatem jest każda permutacja zbioru m elementowego, a więc rozwiązań dopuszczalnych jest $m!$.

Pierwszy sposób generowania rozwiązania dopuszczalnego polega na tym, że numeruje się kolejne permutacje liczbami od 1 do $m!^{1/}$. Następnie przy użyciu odpowiedniego generatora generuje się jedną z liczb naturalnych z przedziału $[1, m!]$. Liczbie tej odpowiada pewna permutacja, która jest elementem populacji.

Jednak przy dużych m ($m > 10$)^{2/} powstaje pewna trudność. Trudność ta bierze się z tego, że w komórce pamięci maszyny cyfrowej może zmieścić się liczba skończona i jeśli jest ona dostatecznie duża maszyna liczby takiej do pamięci przyjąć nie może. Trudność tę ominąć można w sposób następujący:

- za wejściową przyjmuje się dowolną np. naturalną permutację liczb od 1 do m ,
- z przedziału $[1, m]$ generuje się dwie liczby naturalne,
- w permutacji wejściowej liczby te zmienia się miejscami, otrzymuje się w ten sposób nową permutację, która jest teraz wejściową dla dalszego postępowania,
- generowanie i zamianę powtarza się kilkakrotnie (w eksperymentach prowadzonych przez autora powtarzano do $2m$ razy). W wyniku tego postępowania uzyskuje się element populacji, który jest równocześnie permutacją wejściową dla wyznaczenia następnego.

W poniższej tabeli zestawiono najlepszą wartość funkcjonału H uzyskaną metodami przybliżonymi (pominięto tu

^{1/} numerowanie takie jest możliwe, patrz W. Ostasiewicz "Numerowanie permutacji".

^{2/} dla EMC ODRA 1204.

metodę dróg hamiltonowskich), optymalną wartość funkcjonau H oraz wartość uzyskaną metodą Monte Carlo. Przykłady oraz dokładniejsze zestawienia zebrane są w dodatku.

Lp.	Metody przybliżone	Optymalna wartość	Metoda Monte Carlo
1.	833	817	859
2	964	945	976
3	1044	969	1006
4	1062	1040	1055
5	999	978	1012
6	888	870	893
7	909	903	935
8	967	944	979
9	934	919	967
10	894	885	904

Ponieważ dla zadań typu E metoda ta daje dobre rezultaty jest przy tym szybka, wydaje się, że dla pozostałych typów zadań wyniki uzyskane tą metodą powinny być równie dobre, tym bardziej, że właściwie nie ma innych metod na wyznaczanie rozwiązań takich zadań.

Symulacja zadania typu D.

Przypomnijmy, że w zadaniach tego typu dana jest dla każdego p funkcja sterująca α_{pK} oraz $\omega(p,l)$ dla każdego p oraz każdego l jest zbiorem jednoelementowym. Poszukiwana funkcja sterująca nie może naruszać wartości żadnej z funkcji α_{pK} .

Uzyskuje się to w następujący sposób:

- ze zbioru $\{\alpha_{pK}(1)\}_{p=1,2,\dots,m}$ losuje się jeden z operatorów. Wylosowany operator, niech nim będzie $T_{p_1 l_1}$ przyjmuje się za $\alpha_{MK}(1)$,
- ze zbioru $\{\alpha_{pK}(1)\}_{p \neq p_1}$ uzupełnionego o operator $\alpha_{p_1 K}(2)$ losuje się operator, który przyjmuje się jako $\alpha_{MK}(2)$, jeżeli będzie to operator $T_{p_1 l_2}$ to następnego losowania dokonuje się ze zbioru $\{\alpha_{pK}(1)\}_{p \neq p_1}$ uzupełnionego o operator $\alpha_{p_1 K}(3)$. Jeśli natomiast wylosowano operator $T_{p_2 l_1}$ to w jego miejsce do zbioru, z którego losujemy wprowadza się operator $\alpha_{p_2 K}(2)$. Powtarzając tę procedurę (m.k) razy otrzymuje się element populacji. Jak widać sposób ten zabezpiecza nienaruszalność poszczególnych funkcji sterujących przez otrzymaną funkcję α_{MK} .

Symulacja zadań typu C.

Zadanie tego typu jest bardzo podobne do poprzedniego z tą tylko różnicą, że:

$$\omega(p, l) = N_{pl} \cdot$$

Generowanie wartości funkcji sterującej α_{MK} jest tu identyczne jak poprzednio, jednak po każdym takim losowaniu należy jeszcze wylosować ze zbioru N_{pl} warstwę, w której operator ma działać, gdyż w pierwszym losowaniu otrzymuje się rodzinę operatorów \mathcal{T}_{pl} .

Symulacja zadania typu B.

W zadaniach tego typu nie ma ograniczenia o nienaruszalności wartości funkcji sterujących α_{pK} przez funkcję α_{MK} , a ponadto nie ma również potrzeby losowania warstwy operatorów z założenia bowiem $\text{card}(\mathcal{T}_{p1}) = 1$ dla każdego p oraz każdego 1 . Rozwiązaniem dopuszczalnym jest zatem dowolna permutacja operatorów T_{p1} przy czym permutacja ta jest $(m.k)$ elementowa. W zadaniach tego typu stosuje się zatem schemat postępowania opisany przy zadaniu typu E.

Symulacja zadania typu A.

Losowanie tutaj jest takie same jak poprzednio z tym tylko, że losuje się dowolną permutację już nie operatorów T_{p1} lecz rodzin operatorów \mathcal{T}_{p1} . Po wylosowaniu odpowiedniej rodziny należy tak jak w zadaniu typu C jeszcze każdorazowo ze zbioru N_{p1} wylosować odpowiednią warstwę, lub inaczej mówiąc operator z tej rodziny.

ZAKOŃCZENIE.

W prezentowanej pracy podjęto próbę stworzenia jednolitej teorii dla zadań z teorii organizacji procesów, dla których charakterystyczne jest to, iż składają się one z szeregu czynności czy zabiegów, z których każdy zmienia cechę podawanego tym zabiegom obiektu, przy czym uzyskana cecha jest taka jaką powinien mieć produkt finalny. Produktem tym może być np. wyrób w niektórych gałęziach przemysłu, może nim być grupa dydaktyczna w szkole lub uczelni, lub też przedmiot powszechnego użytku poddawany zabiegom renowacyjnym. Już po zakończeniu tej pracy autor spotkał się z jeszcze jednym przykładem zadania sekwencyjnego. Zadanie to dotyczyło ustalania kolejności wykonywania odpowiednich prefabrykatów w budownictwie. Prefabrykaty te są różnych typów i odlewa się je w formach. Dla każdego typu prefabrykatu formę należy odpowiednio uzbroić, przezbrojenia te skracają żywotność formy oraz są czasochłonne. Należy ustalić kolejność odlewania prefabrykatów by uzyskać żadaną ich strukturę, a ponadto by ilość przezbrojeń była minimalna. Takie samo zadanie powstaje gdy przy odlewaniu korzysta się nie z pojedynczej formy a zespołu form^{1/}.

W pracy tej zebrano również wszystkie interesujące propozycje rozwiązania zadań tego typu. Przegląd ten można by uzupełnić jeszcze, jednak zdaniem autora inne metody są bądź

^{1/} patrz "Optymalizacja planowania operatywnego w produkcji prefabrykatów w budownictwie" praca doktorska B.Tilgner. Poznań 1975

mniej interesujące lub mało efektywne. Do nich zaliczyć można propozycję Baran-Jarosz i Grabowskiego wykorzystania do tego rodzaju zadań programowania całkowito-liczbowego lub też propozycję Cichockiego bazującą na metodzie Monte Carlo. Ponadto w cytowanej już kilkakrotnie Jankowskiej-Zorychty znaleźć można model zadania, w którym czas wykonania operacji jest stosunkowo krótki i wykonuje się dużą ilość detali co poważnie zwiększa ilość ograniczeń w prezentowanych już modelach całkowito-liczbowych. W modelu tym zamiast czasu trwania operacji przyjmuje się wydajność stanowiska w jednostce czasu.

Jeszcze raz należy podkreślić, że przeważająca większość propozycji dotyczy szczególnego przypadku zadania programowania sekwencyjnego oznaczonego w tej pracy jako zadanie E. Porównując rozwiązania uzyskane różnymi metodami przedstawione w dodatku widać, że najsensowniejszą jest metoda Monte Carlo. Wynika to z dwóch powodów; szybkością przewyższa metody dające rozwiązanie dokładne a więc metody programowania całkowito-liczbowego oraz podziału i ograniczeń, natomiast szybkie metody przybliżone wyraźnie przewyższa jakością rozwiązania.

Warto zwrócić uwagę na to, że dwie z metod przybliżonych zawierają a w sobie element wyboru losowego. Pierwsza z nich to metoda uśredniania miar jednoczesności działania I. Ostateczna funkcja sterująca α , lub inaczej kolejność wykonywania operacji jaką uzyskuje się tą metodą zależy od tego w jakiej kolejności bierze się wiersze macierzy wyjściowej do tworzenia macierzy A.

Druga metoda, metoda wzajemnego poprzedzania również zawiera element losowy. Na ostateczne rozwiązanie uzyskane tą metodą wpływa wybór detalu do każdorazowego porównania. Metoda ta zdaniem autora wyraźnie przewyższa poprzednią. Wydaje się, że powtarzając kilkakrotnie całe postępowanie wybierając za każdym razem inne detale, powinno się uzyskać poprawę rozwiązania.

Wszystko to dotyczy zadań typu E. Ponieważ przy innych typach zadań wzrasta ich stopień komplikacji, z drugiej strony wobec braku sensownych metod rozwiązywania tego typu problemów wydaje się, że metoda Monte Carlo powinna tu dawać również niezłe rozwiązania.

Autor zdaje sobie sprawę z tego, że szereg zadań spotykanych w praktyce nie daje się zaszeregować do żadnego z typów wyszczególnionych w tej pracy. Tym niemniej jednak po niewielkich modyfikacjach można to uzyskać.

Warto jeszcze zwrócić uwagę na szereg zagadnień nie poruszanych w tej pracy ze względu na brak zadowalających wyników a rozwiązanie których miałyby poważne znaczenie w praktyce.

Do otwartych zagadnień należy zaliczyć:

- szacowanie minimalnej wartości funkcjonału H ,
- stabilizacja rozwiązania,
- wyznaczanie modułów z minimalną stratą na jakości rozwiązania.

Omówmy kolejno te zagadnienia.

Rozwiązując zadanie programowania sekwencyjnego wyznacza się najpierw funkcje sterujące i dla nich wartość funkcjonału H . Praktycznie nigdy się nie wie /jeśli nie korzysta się z metod dających rozwiązanie dokładne/ o ile wyznaczona wartość

funkcjonału H różni się od optymalnej. Gdyby natomiast znać optymalną wartość funkcjonału H /lub jej oszacowanie/ to możnaby porównując uzyskany metodą przybliżoną wynik z wartością optymalną uznać rozwiązanie za dostatecznie dobre lub za mało zadowalające. Znajomość optymalnej wartości funkcjonału H znakomicie uefektywniłaby zmodyfikowaną metodę podziału i ograniczeń. Ponadto znajomość ta pozwoliłaby wyznaczać rozwiązanie metodą Monte Carlo z dowolną dokładnością.

Drugim otwartym zagadnieniem o kapitalnym znaczeniu dla praktyki jest zagadnienie stabilizacji rozwiązania.

Założmy, że dla danego zadania wyznaczone mamy optymalne funkcje sterujące α i β oraz odpowiadającą im wartość funkcjonału H . Powstają teraz następujące dwa pytania:

- 1^o Jak mogą zmieniać się liczby $H(T_{p1})$ by otrzymane uprzednio funkcje sterujące były nadal optymalne?
- 2^o Czy można dla nowego zadania, różniącego się od rozwiązanego kilkoma tylko wartościami $H(T_{p1})$ wskazać bez wykonywania obliczeń optymalne funkcje α i β ?

Pozytywna i konstruktywna odpowiedź na te pytania miałyby niemałe znaczenie praktyczne. Opłacałoby się bowiem stosowanie dokładnych metod mimo ich czasochłonności, gdyż uzyskane rozwiązanie można byłoby wówczas przenosić na inne zadania. Ponadto w praktyce wartości $H(T_{p1})$ jakimi dysponuje się przy rozwiązywaniu zadań sekwencyjnych bardzo często znacznie odbiegają od rzeczywistych, co może przekreślić celowość podejmowania prób poszukiwania rozwiązania. Dla ilustracji

wagi tego problemu przytoczmy następujący przypadek. Na zlecenie przemysłu autor wyznaczył metodą Monte Carlo przybliżone rozwiązanie. Już w trakcie analizy danych widać było, że jedno ze stanowisk jest wąskim gardłem i limituje zdolności badanego wydziału, co znalazło zresztą potwierdzenie w uzyskanym rozwiązaniu. W trakcie konsultacji z przedstawicielami zakładu powstała paradoksalna sytuacja, okazało się bowiem, że ilość produkowanych wyrobów jest 75% wyższa od ilości wynikającej z rozwiązania. Wyjaśnienie powstałego paradoksu było bardzo proste, okazało się, że dostarczone dane odbiegały znacznie od rzeczywistości. Różnica była tak wielka, że po pierwszej weryfikacji danych stanowisko, które było wąskim gardłem przestało nim być, co od razu znalazło odbicie w rozwiązaniu, które i tak było nadal zledeniodawcy do niczego nieprzydatne, gdyż znalazło się tam inne wąskie gardło, którego w rzeczywistości nie było. Ponowna weryfikacja danych trwała długo i szła niezwykle opornie. Ma to zresztą swoje uzasadnienie psychologiczne. Znana jest niechęć do ujawniania rezerw i do pracy zgodnie z harmonogramami optymalnymi, gdyż te są z reguły napięte i małe zakłócenia powodują niewykonanie planów co ma przykre skutki moralne jak i materialne. Rozwiązanie zagadnienia stabilizacji rozwiązania uwolniłoby /przynajmniej częściowo/ od konieczności zbierania dokładnych danych.

Trzecim otwartym zagadnieniem jest zagadnienie wyznaczania modułów z minimalną stratą na jakości rozwiązania. Rozwiązanie tego problemu pozwoliłoby, zdaniem autora, na szersze niż dotychczas stosowanie programowania sekwencyjnego do zagadnień praktycznych.

Rzecz polega na tym, że zadania polegają na budowaniu optymalnych harmonogramów dla długich odcinków czasu /miesiąc, kwartał, półrocze a nawet rok/ co powoduje, że staje się ono zadaniem o olbrzymiej częstokroć liczbie danych. Poszukiwanie rozwiązania dla takiego zadania jest utrudnione chociażby z powodu ograniczonej pojemności komputera. Jedynym rozsądnym wyjściem jest w takim przypadku rozbicie zadania wyjściowego na zadania mniejsze, które tu nazwiemy modułami. Jednak jak wynika z ostatniego wniosku rozdziału pierwszego optymalizacja w tych modułach i łączenie rozwiązań z modułów w jedno rozwiązanie zadania pierwotnego powodować musi straty na jakości rozwiązania. W tym stanie rzeczy celowym jest poszukiwanie takich modułów by straty te były możliwie najmniejsze.

Zasygnalizowaliśmy tu kilka zagadnień wiążących się bezpośrednio z programowaniem sekwencyjnym, nad którymi warto pracować, gdyż ich rozwiązanie powinno spowodować szersze niż dotychczas stosowanie metod matematycznych w praktyce.

Dwa spośród nich, a to: zagadnienie stabilizacji rozwiązania i zagadnienie wyznaczania modułów, powinny już w niedalekiej przyszłości doczekać się rozwiązania, tym bardziej, że autor ma w tym zakresie pewne propozycje, dla których w tej chwili brak jest weryfikacji na maszynie.

Natomiast zagadnienie szacowania minimalnej wartości funkcjonału H wydaje się zagadnieniem ciągle otwartym i jak narazie bez widoków na rozwiązanie, podobnie jak zasadniczy problem tej pracy: wyznaczanie optymalnej funkcji α i β . W tym stanie rzeczy pozostaje tylko korzystać z metod przybliżonych, tym bardziej, że dzisiaj nikt już nie żąda roz-

wiązania dokładnego, jeśli koszt jego uzyskania jest niewspółmiernie wysoki.

Zwróćmy uwagę na następujący fakt, który autorowi wskazał promotor, prof.dr hab.Z.Hellwig.

Zagadnienie rozważane w tej pracy dotyczy sytuacji kiedy żądane i sprecyzowane zadania wykonuje się tylko raz. Taki proces nazwać można przedsięwzięciem. Zagadnienie radykalnie się zmienia jeśli następuje wielokrotne powielanie tych zadań. Jak wynika z twierdzenia 1.7 powielanie harmonogramu optymalnego przeważnie jest gorsze /ze względu na wartość funkcjonalu H / od optymalnego harmonogramu wyznaczonego dla tych zwielokrotnionych zadań. Załóżmy teraz, że powtórzeń takich musi być dużo i proces nasz przestaje być przedsięwzięciem, a staje się produkcją masową. Co się wówczas dzieje? Z uwagi na niejednakowe czasy trwania operacji na poszczególnych stanowiskach, na niektórych ze stanowisk powstają czasy oczekiwania na kolejne operacje^{1/}. W przypadku przedsięwzięcia nic groźnego się wówczas nie dzieje. Natomiast jeśli będzie się powielalo harmonogram wyznaczony dla przedsięwzięć to straty wynikające z tych martwych czasów będą rosły. Od pewnego momentu /od pewnej wielkości produkcji/ warto ponieść dodatkowe koszty w celu zmniejszenia strat z tytułu przerw międzyoperacyjnych. Te dodatkowe koszty mogą być przeznaczone np. na zainstalowanie dodatkowych maszyn i urządzeń w celu wyrównania czasów trwania operacji, a właściwie wyrównania przepustowości poszczególnych stanowisk.

^{1/} patrz diagram Gantt.

Z rozważań przeprowadzonych w tej pracy oraz z przykładów w niej zawartych wynikają następujące wnioski:

1. Zadanie, które zostało sformułowane obejmuje szeroką klasę zagadnień spotykanych w praktyce.
2. Szczególny przypadek tego zadania sformułowany przed 20 laty ma bardzo bogatą literaturę.
3. Świadczy to o dużym zainteresowaniu tymi zagadnieniami zarówno z praktycznego jak i teoretycznego punktu widzenia.
4. Mimo tak dużego zainteresowania, praktycznie nie ma metod wyznaczania rozwiązania tego typu zadania.
5. Do wyznaczania rozwiązania tej klasy zadań pozostają jedynie metody przybliżone.
6. Najbardziej efektywne spośród metod przybliżonych to: metoda wzajemnego poprzedzania, metoda Monte Carlo i metoda dróg hamiltonowskich.
7. Metodę dróg hamiltonowskich można uefektywnić przez przyspieszenie algorytmu wyznaczania dróg hamiltonowskich.
8. Pełne zastosowanie procedury Monte Carlo do metody wzajemnego poprzedzania wpłynęło na jakość uzyskiwanego rozwiązania.
9. Dla zadań pozostałych typów jako metoda ich rozwiązania pozostaje procedura Monte Carlo.
10. Zagadnienie wyznaczenia optymalnych funkcji sterujących jest w dalszym ciągu zagadnieniem otwartym.

11. Każda nowa metoda rozwiązania tego zadania powinna spełniać dwa postulaty:
 - być szybka
 - dawać rozwiązania lepsze niż metody dotychczasowe.
12. Sformułowane zadanie sekwencyjne zrodziło szereg nowych zagadnień, rozwiązanie których jest interesujące, z teoretycznego punktu widzenia, jak również miałyby niemałe znaczenie praktyczne.
13. Zadanie sformułowane w tej pracy dotyczy przedsięwzięcia, a więc sytuacji kiedy uzyskany harmonogram realizuje się jeden tylko raz.
14. Przy wielokrotnym powtarzaniu zadań produkcyjnych zadanie programowania sekwencyjnego bardzo często staje się zadaniem z dziedziny organizacji produkcji masowej.

DODATEK

W dodatku chcemy zaprezentować przykłady zadań programowania sekwencyjnego oraz porównać wyniki uzyskane różnymi metodami, które zostały opisane w niniejszej pracy. Podkreślamy raz jeszcze, że dane do zadań wygenerowano na maszynie cyfrowej i nie są one dobierane w jakiś szczególny sposób. Obliczenia wykonano w Ośrodku Obliczeniowym Instytutu Cybernetyki Ekonomicznej, wszystkie programy za pomocą których uzyskano demonstrowane wyniki są w posiadaniu autora. Pierwsze 10 zadań to zadania typu E, które można przedstawić w postaci macierzy 10×8 . Gdyby powołać się na przykład z dziedziny organizacji produkcji odpowiadałyby one zadaniom, w których należy wykonać 10 wyrobów dysponując 8 stanowiskami. Zadania te ponumerowano liczbami od 1 do 10. Po każdym zadaniu, na wykresie demonstruje się zbieżność procedury Monte Carlo. Na osi poziomej odłożono liczbę pobieranych próbek, na osi pionowej wartości funkcjonału H. W przykładach tych przyjmowano, że próbka liczy 20 elementów a eksperyment przerywano po 100 niepowodzeniach.

1.

$H(T_{ij}) =$

$i \backslash j$	1	2	3	4	5	6	7	8
1	59	4	17	6	45	48	35	24
2	27	37	10	79	24	65	2	75
3	39	16	11	62	19	70	12	34
4	48	79	31	63	11	74	72	96
5	89	37	94	89	17	11	45	3
6	83	71	33	42	86	35	57	44
7	35	35	88	22	3	25	91	55
8	33	54	10	26	87	77	90	17
9	51	25	59	47	90	15	56	61
10	48	18	6	75	2	79	52	62

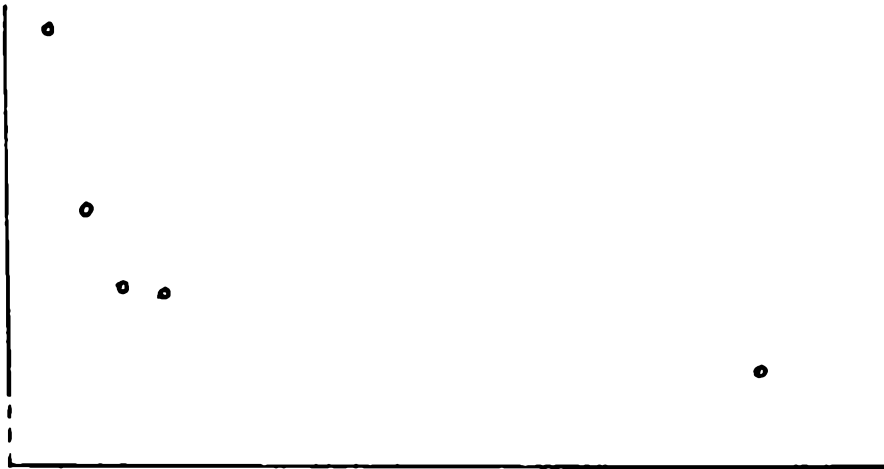
2.

$H(T_{ij}) =$

$i \backslash j$	1	2	3	4	5	6	7	8
1	86	52	22	39	63	91	62	96
2	3	34	89	70	22	48	14	30
3	97	91	90	28	66	84	90	45
4	80	24	34	74	9	50	68	65
5	94	58	90	15	50	67	32	7
6	18	37	84	5	71	31	76	88
7	63	31	4	46	40	7	6	18
8	79	28	95	85	8	33	34	7
9	46	40	63	18	12	18	82	62
10	78	100	74	87	72	46	84	84

1001

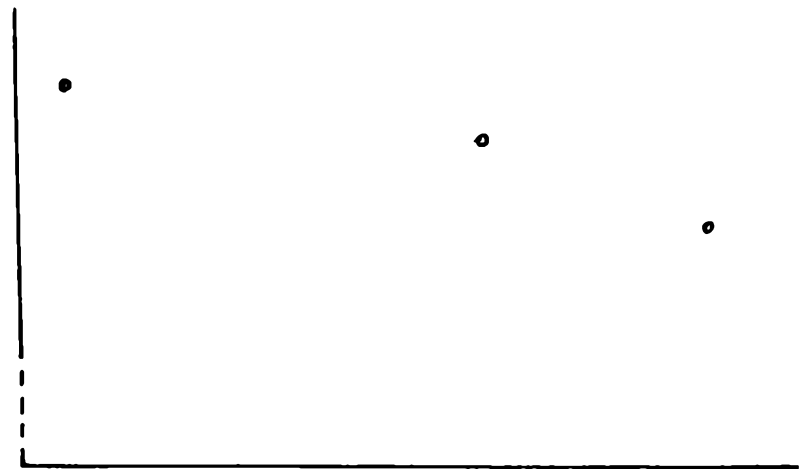
859



20

991

976



70

3

$H(T_{ij}) =$

$j \backslash i$	1	2	3	4	5	6	7	8
1	56	75	70	41	91	53	14	97
2	31	24	94	1	94	19	88	14
3	30	2	71	96	80	92	40	1
4	79	22	91	11	6	59	51	30
5	89	68	52	26	61	28	62	18
6	56	97	60	19	70	29	70	27
7	60	38	33	64	94	15	36	69
8	64	91	95	35	27	58	78	2
9	17	29	83	98	95	56	72	29
10	52	96	41	19	63	25	55	2

1074

1006

2

4

$H(T_{ij}) =$

$j \backslash i$	1	2	3	4	5	6	7	8
1	100	68	1	12	26	51	92	83
2	77	38	98	19	18	41	97	65
3	81	97	60	31	62	35	36	32
4	82	93	78	81	81	80	20	28
5	82	96	6	49	83	95	69	91
6	19	100	56	60	60	71	11	60
7	66	19	98	88	89	72	85	69
8	28	90	64	52	30	32	38	48
9	46	71	43	20	30	58	28	26
10	96	41	85	6	18	29	22	27

1204

1055

- 218 -

147

5

$$H(T_{ij}) =$$

i \ j	1	2	3	4	5	6	7	8
1	74	28	32	65	41	56	57	32
2	56	93	17	75	43	59	64	78
3	99	32	4	66	96	12	75	57
4	32	40	73	43	35	60	34	35
5	17	36	59	55	76	19	94	74
6	51	59	59	3	21	76	21	33
7	65	70	28	30	57	91	85	69
8	25	52	82	29	53	97	70	33
9	32	12	94	38	65	96	66	75
10	19	75	98	43	31	65	76	43

1084

1042

21

6

$$H(T_{ij}) =$$

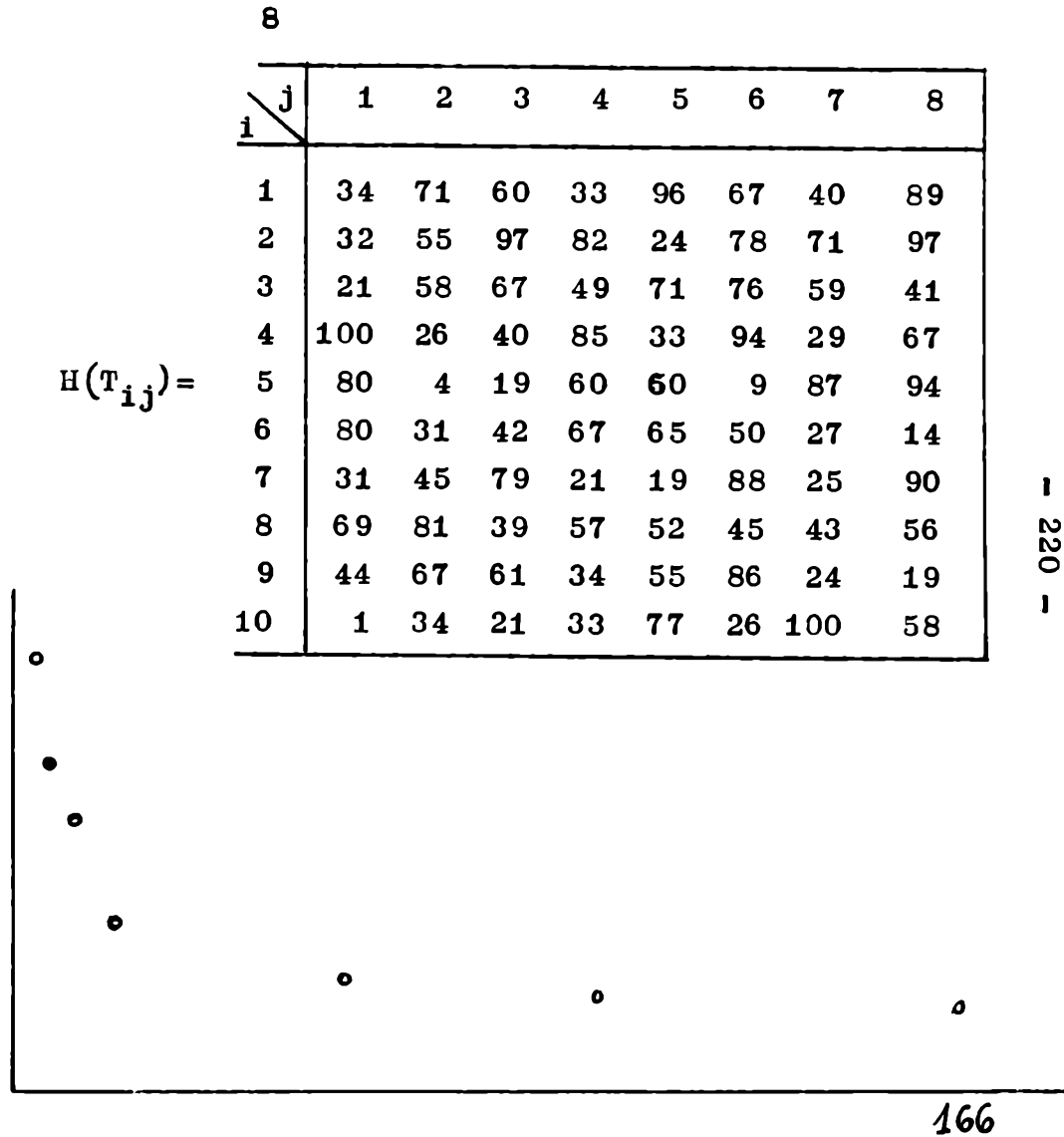
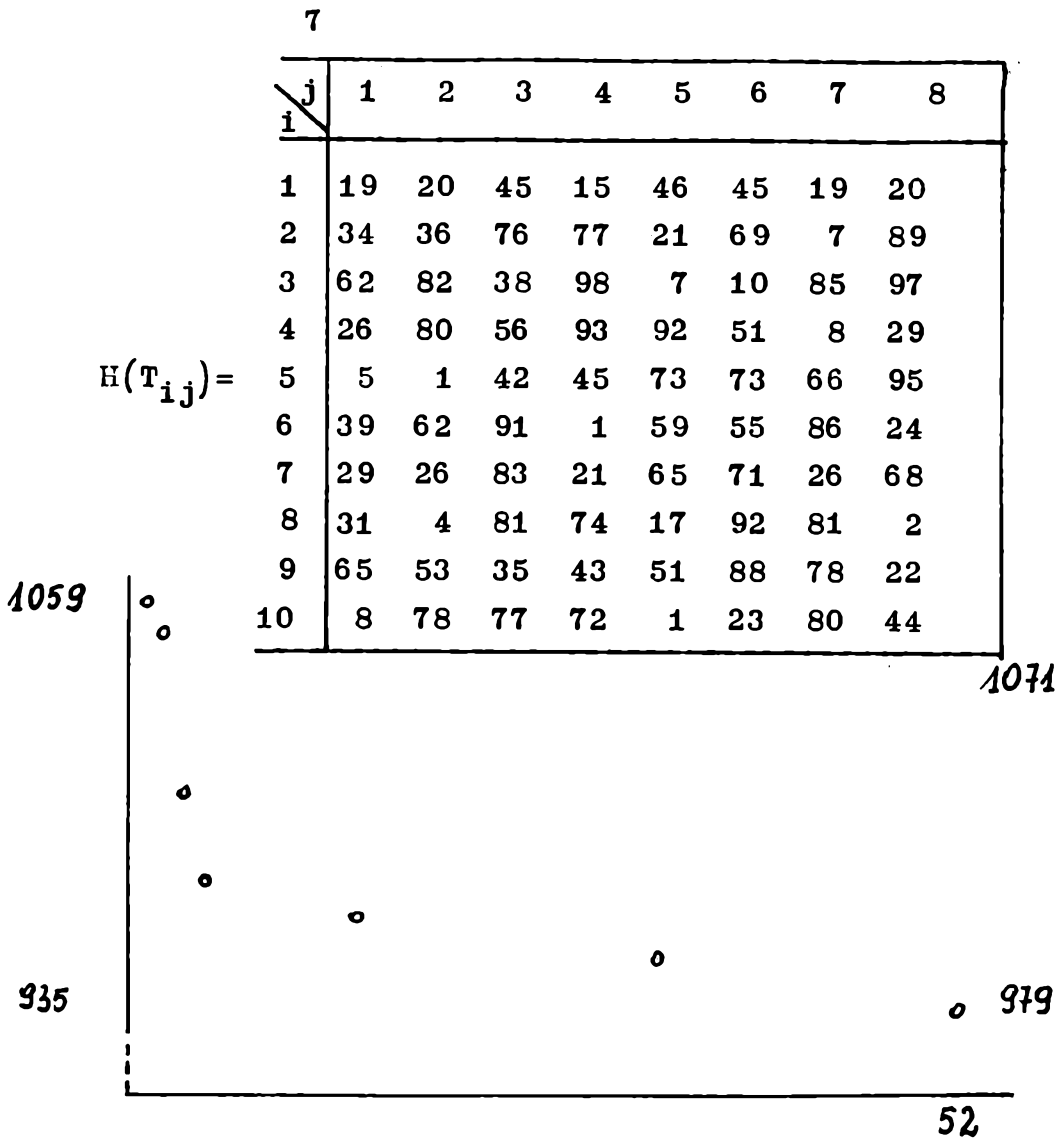
i \ j	1	2	3	4	5	6	7	8
1	14	19	29	17	68	72	91	77
2	77	58	85	89	79	25	41	88
3	57	21	99	72	99	4	32	92
4	98	15	28	94	4	89	14	5
5	30	20	38	80	2	81	95	87
6	67	86	5	51	39	5	43	44
7	8	34	15	26	91	6	86	33
8	36	60	62	20	72	53	11	53
9	18	29	49	45	29	34	64	53
10	6	80	92	11	43	62	50	28

977

893

- 219 -

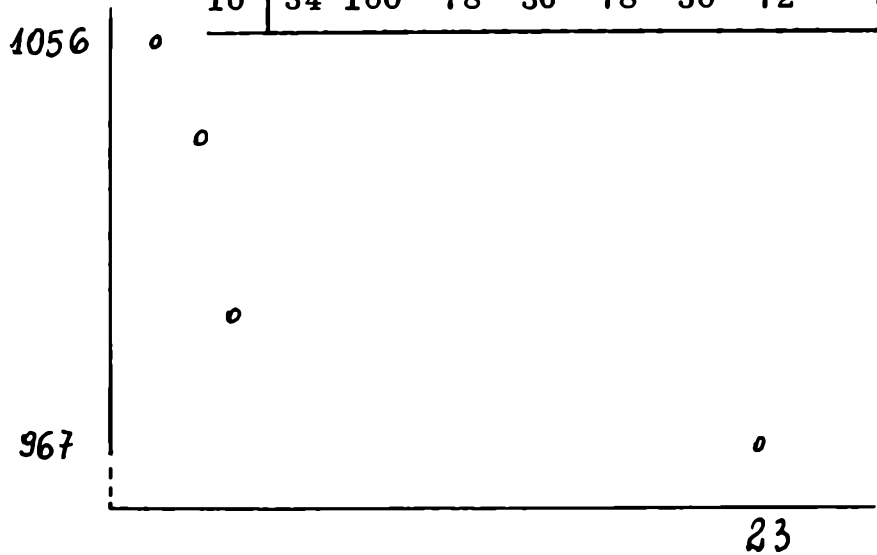
72



9

$H(T_{ij}) =$

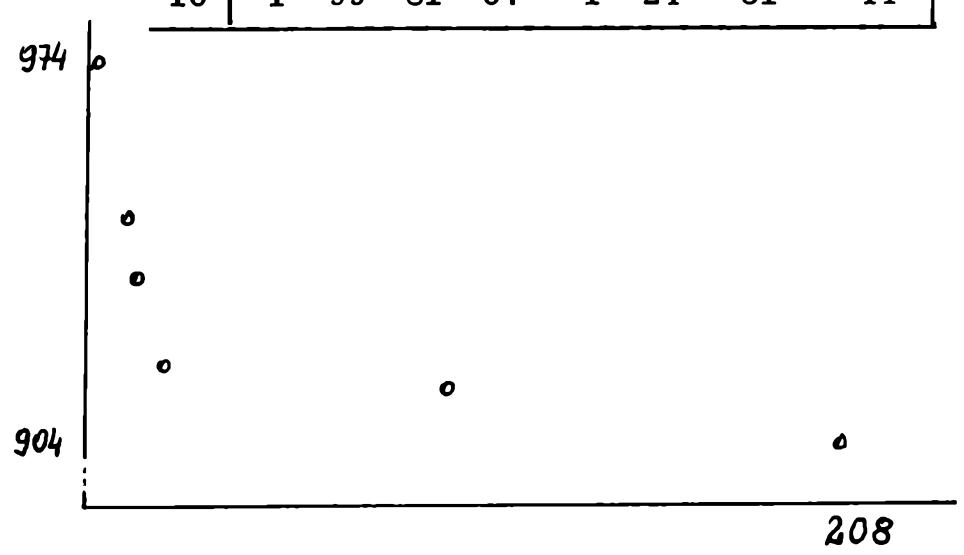
$\begin{matrix} j \\ i \end{matrix}$	1	2	3	4	5	6	7	8
1	86	69	30	98	28	5	10	55
2	47	50	95	38	38	23	61	11
3	28	20	1	99	81	67	1	24
4	51	44	70	2	84	49	18	78
5	73	9	83	19	58	54	1	55
6	83	23	80	67	98	39	85	31
7	8	16	59	8	83	33	57	82
8	6	41	80	54	79	35	54	79
9	72	70	60	61	33	20	11	91
10	34	100	78	36	78	30	72	83



10

$H(T_{ij}) =$

$\begin{matrix} j \\ i \end{matrix}$	1	2	3	4	5	6	7	8
1	39	79	58	95	81	44	41	9
2	16	89	96	61	98	17	18	83
3	50	70	44	74	48	71	6	18
4	45	16	36	44	53	12	19	48
5	67	70	93	53	45	43	8	28
6	39	20	21	53	67	63	57	35
7	46	1	8	72	68	69	86	69
8	30	98	28	5	10	55	47	50
9	95	38	38	23	61	11	28	20
10	1	99	81	67	1	24	51	44



1	2	3	4	5	6	7	8	9	10	11
1	901	924	875	900	908	883	833	843	859	817
2	1033	1009	964	964	1061	982	985	963	976	945
3	1050	1142	1111	1044	1205	1075	1054	1004	1006	969
4	1112	1141	1212	1112	1224	1147	1062	1080	1055	1040
5	1052	1038	1024	1052	1068	1047	999	1028	1012	978
6	941	909	918	926	963	952	888	888	893	870
7	1043	945	965	953	1112	909	922	903	935	903
8	998	1072	1006	967	1084	978	986	958	979	944
9	966	1001	1004	977	988	971	934	930	967	919
10	977	924	1003	951	981	969	894	903	904	885

W tablicy podano wyniki uzyskane różnymi metodami dla zadań od 1 do 10. W kolumnie pierwszej podano numer zadania w kolejnych kolumnach podano wyniki uzyskane następującymi metodami:

- 2 - Pietrowa,
- 3 - Pietrowa i Sokolicyna,
- 4 - Palmera,
- 5 - Campbella, Dudka i Smitha,
- 6 - uśredniania miar jecnoczesności działania I
- 7 - uśredniania miar jednoczesności działania II
- 8 - wzajemnego poprzedzania,
- 9 - Dróg hamiltonowskich
- 10 - Monte Carlo

W ostatniej kolumnie podano wartość funkcjonału H dla optymalnej funkcji sterującej wyznaczone zmodyfikowaną metodą podziału i ograniczeń.

Następne przykłady dotyczą kolejno zadań typu D, C, B, A. Zadania te zapisano w postaci macierzy $H(\theta)$. W zadaniach typu D oraz C, gdzie dane są funkcje \mathcal{L}_{pK} , funkcje te również zapisano w postaci macierzy. Rozwiązanie podano w postaci tablic θ oraz $H(\theta)$. Rozwiązanie generowano przy pomocy procedury Monte Carlo w związku z czym za każdym razem podano wartość funkcjonału H dla pierwszej wylosowanej próbki oraz liczbę próbek, które wylosowano poczynając od pierwszej a kończąc na dającej wartość funkcjonału przyjętą za rozwiązanie. Tutaj podobnie jak w przypadku zadania typu E przyjęto liczbę próbek 20 elementów i eksperyment przerywano jeśli 100 kolejnych próbek nie poprawiło wartości funkcjonału.

Zadanie typu D

	1,1	1,2	1,3	1,4	2,1	2,2	2,3	2,4	3,1	3,2	3,3	3,4	4,1	4,2	4,3	4,4	5,1	5,2	5,3	5,4	6,1	6,2	6,3	6,4
1	0	0	0	0	0	0	0	87	0	0	0	0	88	0	0	0	0	0	0	0	0	0	0	0
2	48	0	12	66	0	74	0	0	0	0	0	0	0	0	0	0	0	0	0	50	0	0	0	0
3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	76	0	39	0	0	28	0	0
4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	21	71
5	0	76	0	0	10	0	0	0	46	0	0	0	0	0	0	0	0	95	0	0	83	0	0	0
6	0	0	0	0	0	0	30	0	0	42	0	58	0	16	0	25	0	0	0	0	0	0	0	0
7	0	0	0	0	0	0	0	0	0	0	34	0	0	0	7	0	0	0	0	0	0	0	0	0

$H(\theta) =$

$\alpha_{pk}(o) =$

p \ c	c			
	1	2	3	4
1	T_{13}	T_{12}	T_{13}	T_{11}
2	T_{21}	T_{24}	T_{22}	T_{23}
3	T_{31}	T_{32}	T_{33}	T_{34}
4	T_{43}	T_{44}	T_{41}	T_{42}
5	T_{52}	T_{51}	T_{53}	T_{54}
6	T_{64}	T_{61}	T_{63}	T_{62}

$\varnothing =$

	1	2	3	4	5	6	7
1		2,4	4,1				
2	1,3		2,2	1,4	1,1	5,4	
3			5,1	5,3			6,2
4	6,4					6,3	
5	2,1	5,2	1,2	3,1	6,1		
6		4,4		4,2	3,2	2,3	3,4
7	4,3					3,3	

$H(\phi) =$

	1	2	3	4	5	6	7
1		97	185				
2	12		171	247	295	345	
3			181	220			359
4	71					331	
5	10	105	181	227	310		
6		32		201	269	299	361
7	7					303	

391 po 94 próbkach uzyskano 361.

$H(\beta) =$

	β_1	β_2	β_3	β_4	β_5	β_1	β_2	β_3	β_4	β_5	β_1	β_2	β_3	β_4	β_5	β_1	β_2	β_3	β_4	β_5	β_1	β_2	β_3	β_4	β_5
1	0	0	0	0	0	0	0	46	0	0	0	0	16	0	0	0	0	0	0	83	0	0	0	0	0
2	76	0	66	10	0	30	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	21	0	0	0
3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	95	0	50	0	0	0	71	96	61	
4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	99	0	0	0	28	0	0	0	0	
5	0	12	0	0	74	0	0	0	42	34	0	88	0	7	0	76	0	0	0	0	0	0	0	0	
6	0	0	0	0	0	0	87	0	0	0	58	0	0	0	25	0	0	0	0	0	0	0	0	0	

$\alpha_{pk(o)} =$

$p \backslash o$	1	2	3	4	5
1	T_{11}	T_{14}	T_{15}	T_{12}	T_{13}
2	T_{23}	T_{21}	T_{25}	T_{24}	T_{22}
3	T_{34}	T_{32}	T_{33}	T_{35}	T_{31}
4	T_{41}	T_{43}	T_{44}	T_{42}	T_{45}
5	T_{53}	T_{55}	T_{54}	T_{51}	T_{52}

$\phi =$

	1	2	3	4	5	6	7	8
1	2,2			3,3	4,5			
2	1,1	2,1	1,4				1,3	5,2
3	5,3	5,5	4,4	4,2	5,4			
4		4,3				5,1		
5	4,1	3,4	3,2	2,5	1,5	1,2	2,4	
6					3,5	3,1		2,2

$H(\beta) =$

	1	2	3	4	5	6	7	8
1	46			187	360			
2	76	106	116				357	422
3	71	132	182	277	373			
4		115					401	
5	76	83	171	205	279	291	333	
6					212	270		420

456 po 4 p66bkaoh uzyskano 422

$H(\theta) =$

	1,1	1,2	1,3	1,4	2,1	2,2	2,3	2,4	3,1	3,2	3,3	3,4	4,1	4,2	4,3	4,4	5,1	5,2	5,3	5,4	6,1	6,2	6,3	6,4	7,1	7,2	7,3	7,4	
1	0	0	30	87	0	0	0	58	0	0	0	0	78	0	0	0	0	0	0	71	0	0	0	0	0	0	0	100	0
2	10	0	0	0	0	42	0	0	0	0	0	0	0	0	0	0	83	0	21	0	0	61	0	0	0	0	0	0	0
3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	98	0	4	23	44	0	0	0	0
4	0	74	0	0	46	0	0	0	88	18	0	25	0	95	0	50	0	28	0	0	0	0	0	0	0	23	0	1	
5	0	0	0	0	0	34	0	0	0	0	7	0	0	0	39	0	0	0	0	0	0	0	0	0	0	0	0	0	

$\alpha_{pk}(0) =$

$\frac{0}{D}$	1	2	3	4
1	T_{13}	T_{11}	T_{12}	T_{14}
2	T_{24}	T_{22}	T_{23}	T_{21}
3	T_{33}	T_{32}	T_{31}	T_{34}
4	T_{42}	T_{43}	T_{44}	T_{41}
5	T_{54}	T_{53}	T_{51}	T_{52}
6	T_{61}	T_{64}	T_{63}	T_{62}
7	T_{73}	T_{74}	T_{71}	T_{72}

$\vartheta =$

	1	2	3	4	5	6	7	8	9	10
1	1,3	5,4	2,4	7,3	1,4		1,1			
2		1,1	5,3	2,2	6,2	5,1				
3	6,1	6,4	6,3			7,1				
4	4,2	3,2	3,1	1,2	7,4	4,4	2,1	5,2	7,2	3,4
5	3,3	4,3			2,3					

$H(\theta) =$

	1	2	3	4	5	6	7	8	9	10
1	30	101	159	259	360		436			
2		40	122	201	262	345				
3	96	119	123			318				
4	95	111	199	273	274	324	370	398	421	446
5	7	134			235					

pierwsza próbka dała 446

Zadanie typu C

	1 ₁	1 ₂	1 ₃	1 ₄	2 ₁	2 ₂	2 ₃	2 ₄	3 ₁	3 ₂	3 ₃	3 ₄	4 ₁	4 ₂	4 ₃	4 ₄	5 ₁	5 ₂	5 ₃	5 ₄	6 ₁	6 ₂	6 ₃	6 ₄
1	0	87	0	0	24	0	94	0	0	63	41	49	0	49	64	28	0	76	0	0	3	0	0	28
2	0	0	0	0	99	0	40	0	0	98	63	65	65	96	44	70	59	0	0	0	62	0	0	0
3	0	93	34	94	87	0	0	0	29	0	34	46	0	42	40	24	22	23	29	0	0	14	86	0
4	71	0	31	55	0	73	89	0	22	41	7	0	0	20	66	0	0	54	0	49	10	0	99	5
5	0	5	0	0	0	0	74	22	54	72	95	13	0	76	39	95	0	15	0	40	44	0	25	0
6	83	3	0	0	64	22	79	0	97	44	47	85	0	0	69	94	96	83	67	0	55	59	0	46
7	0	57	0	0	39	0	76	0	63	5	91	98	0	70	30	92	0	0	75	0	0	19	18	15

H(φ) =

$\mathcal{L}_{PK}(0) =$

p \ o	1	2	3	4
1	T ₁₂	T ₁₁	T ₁₄	T ₁₄
2	T ₂₃	T ₂₄	T ₂₂	T ₂₁
3	T ₃₂	T ₃₃	T ₃₁	T ₃₄
4	T ₄₄	T ₄₁	T ₄₂	T ₄₃
5	T ₅₄	T ₅₃	T ₅₂	T ₅₁
6	T ₆₁	T ₆₂	T ₆₃	T ₆₄

φ =

	1	2	3	4	5	6	7
1	6,1		3,3			4,3	
2		4,1					
3	4,4				5,1		3,4
4				1,4	1,3	3,1	
5	5,4	2,3	6,3	2,4			
6	1,2	3,2	1,1	5,2	2,2	2,1	
7		6,2	5,3	6,4	4,2		

H(φ) =

	1	2	3	4	5	6	7
1	3		88			288	
2		89					
3	24				235		284
4				185	216	238	
5	40	114	139	161			
6	3	47	130	213	235	299	
7		22	115	154	224		

357 po 71 próbkach uzyskano 299,

	1,1	1,2	1,3	1,4	1,5	2,1	2,2	2,3	2,4	2,5	3,1	3,2	3,3	3,4	3,5	4,1	4,2	4,3	4,4	4,5	5,1	5,2	5,3	5,4	5,5
1	37	79	0	0	92	0	42	0	9	0	21	78	0	37	72	83	93	34	0	0	40	0	79	0	22
2	0	89	39	0	3	0	17	62	0	79	16	0	0	56	13	71	3	94	0	0	0	76	0	97	63
3	0	0	31	0	72	34	55	0	53	35	15	24	0	68	41	5	0	31	39	64	0	0	0	0	0
4	0	21	0	89	0	0	15	0	53	77	49	78	0	86	51	87	0	0	99	0	73	0	89	29	0
5	92	33	0	0	7	15	25	0	69	10	56	70	0	0	6	0	0	0	0	0	87	0	94	22	54
6	0	34	0	81	89	0	74	0	13	90	70	82	91	89	88	52	57	55	24	0	22	74	0	0	0

$\alpha_{pK}(c) =$

p \ o	1	2	3	4	5
1	T ₁₄	T ₁₁	T ₁₂	T ₁₃	T ₁₅
2	T ₂₅	T ₂₄	T ₂₁	T ₂₃	T ₂₂
3	T ₃₂	T ₃₁	T ₃₄	T ₃₃	T ₃₅
4	T ₄₃	T ₄₄	T ₄₂	T ₄₅	T ₄₁
5	T ₅₄	T ₅₃	T ₅₅	T ₅₂	T ₅₁

$\vartheta =$

	1	2	3	4	5	6	7	8	9	10
1		2,4	1,1	3,4	2,2					
2				2,3		3,5	1,3	1,5	5,2	
3	2,5	4,3			4,5	4,1				
4	1,4	3,1	5,4	5,3	1,2					
5			2,1		5,5					
6	3,2		4,4	4,2	3,3					5,1

$H(\vartheta) =$

	1	2	3	4	5	6	7	8	9	10
1		44	126	175	217					
2			121		279	318	321	397		
3	35	66			211	216				
4	89	138	167	256	277					
5			59		310					
6	62		90	147	266				419	

427 po 8 próbkach uzyskano 419.

$H(\vartheta) =$

	1,1	1,2	1,3	1,4	2,1	2,2	2,3	2,4	3,1	3,2	3,3	3,4	4,1	4,2	4,3	4,4	5,1	5,2	5,3	5,4	6,1	6,2	6,3	6,4	7,1	7,2	7,3	7,4
1	92	89	34	0	0	0	15	0	15	0	10	35	0	56	91	37	0	41	6	0	87	0	0	0	99	22	0	89
2	0	21	0	31	81	7	89	25	55	69	79	0	0	0	70	68	51	88	52	0	0	3	31	0	64	0	74	79
3	0	33	0	0	92	0	17	0	0	13	53	90	21	62	78	0	0	13	0	0	83	0	0	39	0	40	0	0
4	37	0	0	0	39	0	72	0	42	62	77	70	0	16	24	86	0	72	0	0	0	93	34	94	0	87	0	76
5	0	79	0	0	89	3	34	0	74	9	53	49	0	15	78	56	89	0	0	71	5	0	57	55	24	73	0	64

$\alpha_{PK}(c) =$

ϑ	1	2	3	4
1	T_{12}	T_{14}	T_{13}	T_{11}
2	T_{24}	T_{22}	T_{23}	T_{21}
3	T_{31}	T_{33}	T_{32}	T_{34}
4	T_{42}	T_{43}	T_{41}	T_{44}
5	T_{53}	T_{52}	T_{54}	T_{51}
6	T_{62}	T_{63}	T_{61}	T_{64}
7	T_{73}	T_{71}	T_{72}	T_{74}

$\vartheta =$

	1	2	3	4	5	6	7	8
1	5,3	5,2	7,1		1,3		3,4	
2	7,3	6,2	2,4	1,4	2,2	5,1		
3	3		4,1	7,2		2,3	2,1	6,4
4	3,1	4,3	6,3		4,4	1,1	7,4	
5	4,2	1,2	5,4	3,3	3,2	6,1		

$H(\vartheta) =$

	1	2	3	4	5	6	7	8
1	6	47	173		207		262	
2	74	77	102	133	140	216		
3			87	213		230	322	361
4	42	66	111		197	244	320	
5	15	94	165	218	227	232		

391 po 47 próbkach uzyskano 361.

Zadanie typu B

	1,1	1,2	1,3	1,4	2,1	2,2	2,3	2,4	3,1	3,2	3,3	3,4	4,1	4,2	4,3	4,4	5,1	5,2	5,3	5,4	6,1	6,2	6,3	6,4
1	0	0	0	0	0	0	0	87	0	0	0	0	83	0	0	0	0	0	0	0	0	0	0	0
2	48	0	12	66	0	74	0	0	0	0	0	0	0	0	0	0	0	0	0	0	50	0	0	0
3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	76	0	39	0	0	28	0	0
4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	21	71
5	0	76	0	0	10	0	0	0	46	0	0	0	0	0	0	0	0	95	0	0	83	0	0	0
6	0	0	0	0	0	0	30	0	0	42	0	58	0	16	0	25	0	0	0	0	0	0	0	0
7	0	0	0	0	0	0	0	0	0	0	34	0	0	0	7	0	0	0	0	0	0	0	0	0

$H(\vartheta) =$

	1	2	3	4	5	6
1	2,4		4,1			
2	1,1	1,3	5,4	1,4	2,2	
3	5,3			6,2	5,1	
4		6,4	6,3			
5	6,1	3,1	2,1	5,2	1,2	
6	3,4	4,4		4,2	3,2	2,3
7	4,3		3,3			

$\vartheta =$

344 po 32 próbkach uzyskano 310

	1	2	3	4	5	6
1	87		175			
2	48	60	110	176	250	
3	39			203	310	
4		154	175			
5	83	129	139	234	310	
6	58	83		191	233	280
7	7		163			

$H(\vartheta) =$

	11	12	13	14	15	21	22	23	24	25	31	32	33	34	35	41	42	43	44	45	51	52	53	54	55
H(\varnothing)= 1	0	0	0	0	0	0	0	46	0	0	0	0	16	0	0	0	0	0	0	83	0	0	0	0	0
2	76	0	66	10	0	30	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	21	0	0	0
3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	95	0	50	0	0	0	71	96	61
4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	39	0	0	28	0	0	0	0	0
5	0	12	0	0	74	0	0	0	42	34	0	88	0	7	0	76	0	0	0	0	0	0	0	0	0
6	0	0	0	0	0	0	87	0	0	0	58	0	0	0	25	0	0	0	0	0	0	0	0	0	0

	1	2	3	4	5	6	7
$\varnothing =$ 1	4,5			3,3	2,3		
2		1,1	1,3	5,2	1,4	2,1	
3	5,3	5,5	4,2		5,3	4,4	
4		4,3				5,1	
5	1,2	2,4	3,2	1,5	4,1	3,4	2,5
6			2,2		3,5		3,1

	1	2	3	4	5	6	7
H(\varnothing)= 1	83			158	204		
2		88	154	175	238	268	
3	71	132	227		323	373	
4		122				351	
5	12	54	142	228	304	311	345
6			141		183		369

403 po 129 próbkach uzyskano Σu

	1,1	1,2	1,3	1,4	2,1	2,2	2,3	2,4	3,1	3,2	3,3	3,4	4,1	4,2	4,3	4,4	5,1	5,2	5,3	5,4	6,1	6,2	6,3	6,4	7,1	7,2	7,3	7,4
1	0	0	30	87	0	0	0	58	0	0	0	0	76	0	0	0	0	0	0	0	71	0	0	0	0	0	100	0
2	10	0	0	0	0	42	0	0	0	0	0	0	0	0	0	0	83	0	21	0	0	61	0	0	0	0	0	0
3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	98	0	4	23	41	0	0	0
4	0	74	0	0	46	0	0	0	86	16	0	25	0	95	0	50	0	28	0	0	0	0	0	0	0	23	0	1
5	0	0	0	0	0	0	34	0	0	0	7	0	0	0	39	0	0	0	0	0	0	0	0	0	0	0	0	

	1	2	3	4	5	6	7	8	9	10
1	7,3	1,3	5,4	2,4	4,1	1,4		2,2		
2	5,1	6,2	1,1		5,3					
3	6,4		6,1	6,3					7,1	
4	1,2	4,4	4,2	5,2	7,4	3,2	2,1	7,2	3,1	3,4
5	2,3	3,3		4,3						

	1	2	3	4	5	6	7	8	9	10
1	100	130	201	259	335	422				
2	63	144	154		268			352		
3	23		240	244					377	
4	74	124	219	247	248	284	310	333	421	446
5	34	41		258						

470 po 38 próbkach uzyskano 446.

Zadanie typu A

$H(\varnothing) =$	1,1	1,2	1,3	1,4	2,1	2,2	2,3	2,4	3,1	3,2	3,3	3,4	4,1	4,2	4,3	4,4	5,1	5,2	5,3	5,4	6,1	6,2	6,3	6,4
1	0	87	0	0	24	0	94	0	0	63	41	49	0	49	64	28	0	76	0	0	3	0	0	26
2	0	0	0	0	99	0	40	0	0	98	63	65	65	96	44	70	59	0	0	0	62	0	0	0
3	0	93	34	94	87	0	0	0	29	0	34	46	0	42	40	24	22	23	29	0	0	14	86	0
4	71	0	31	55	0	73	89	0	22	41	7	0	0	20	66	0	0	54	0	49	10	0	99	5
5	0	5	0	0	0	0	74	22	54	72	75	13	0	76	39	95	0	15	0	40	44	0	25	0
6	83	3	0	0	64	22	79	0	97	44	47	85	0	0	69	94	96	83	67	0	55	59	0	46
7	0	57	0	0	39	0	76	0	63	5	91	98	0	70	30	92	0	0	75	0	0	19	18	15

$\varnothing =$

	1	2	3	4	5	6	7
1	3,4	3,2		4,2			
2	2,3		4,1		2,1		
3	1,4			5,3			
4		2,2	5,4	1,1	1,3	6,1	
5			3,1	2,4	5,2	1,2	
6	5,1	6,2			4,3		
7	6,4	4,4		3,3			6,3

$H(\varnothing) =$

	1	2	3	4	5	6	7
1	49	112		221			
2	40		172		287		
3	94			191			
4		113	162	233	264	274	
5			166	188	206	269	
6	96	155			290		
7	15	107		257			292

	1,1	1,2	1,3	1,4	1,5	2,1	2,2	2,3	2,4	2,5	3,1	3,2	3,3	3,4	3,5	4,1	4,2	4,3	4,4	4,5	5,1	5,2	5,3	5,4	5,5
1	37	79	0	0	92	0	42	0	9	0	21	78	0	37	72	83	93	34	0	0	40	0	79	0	22
2	0	89	39	0	3	0	17	62	0	79	16	0	0	56	13	71	3	94	0	0	0	76	0	97	63
3	0	0	31	0	72	34	55	0	53	35	15	24	0	68	41	5	0	31	39	64	0	0	0	0	0
4	0	21	0	89	0	0	15	0	53	77	49	78	0	86	51	87	0	0	99	0	73	0	89	29	0
5	92	33	0	0	7	15	25	0	69	10	56	70	0	0	6	0	0	0	0	0	87	0	94	22	54
6	0	34	0	81	89	0	74	0	13	90	70	62	91	89	88	52	57	55	24	0	22	74	0	0	0

	1	2	3	4	5	6
1		1,1	2,2	2,4	3,4	
2	5,2		3,5	1,5	2,3	1,3
3	4,1	2,5		4,5	4,3	
4	1,4	5,4	5,3		1,2	3,1
5	2,1			5,5		
6	3,2	3,3	4,2		5,1	4,4

	1	2	3	4	5	6
1		126	168	177	214	
2	76		166	169	239	278
3	5	50		274	305	
4	89	118	207		228	277
5	15			261		
6	62	153	210		283	329

433 po 10 próbkach uzyskano 329.

	1,1	1,2	1,3	1,4	2,1	2,2	2,3	2,4	3,1	3,2	3,3	3,4	4,1	4,2	4,3	4,4	5,1	5,2	5,3	5,4	6,1	6,2	6,3	6,4	7,1	7,2	7,2	7,4
1	92	89	34	0	0	0	15	0	15	0	10	35	0	56	91	38	0	41	8	0	87	0	0	0	99	22	0	89
20	0	21	0	31	81	7	89	25	55	69	79	0	0	0	70	68	51	88	52	0	0	3	31	0	64	0	74	79
3	0	33	0	0	92	0	17	0	0	13	53	90	21	62	78	0	0	13	0	0	83	0	0	39	0	40	0	0
4	37	0	0	0	39	0	72	0	42	62	77	70	0	16	24	86	0	72	0	0	0	93	34	94	0	87	0	76
5	0	79	0	0	89	3	34	0	74	9	53	49	0	15	78	56	89	0	0	71	5	0	57	55	24	73	0	94

$\bar{g} =$

	1	2	3	4	5	6	7	8	9
1	7,2	3,1	4,4			1,3	5,3	5,2	1,1
2	6,2	6,3	7,3	5,1	1,2	2,4	1,4		
3	3,2	4,1		4,2					
4	2,1				7,4	4,3	3,3		
5		2,2	2,3	7,1	5,4	3,4	6,1	6,4	

$H(\bar{g}) =$

	1	2	3	4	5	6	7	8	9
1	22	37	74			214	236	277	369
2	3	34	108	159	180	205	245		
3	13	34		138					
4	39				208	232	356		
5		42	76	132	230	279	284	339	

416 po 93 próbkach uzyskano 369.

Bibliografia

1. Ackoff R. Sasieni M., Fundamentals of Operations Research, J.Willy New York, 1968.
2. Ackers S.B., Friedman J., A Non- Numerical Approach to Production Scheduling Problem /w:/ "Operations Research" 1955 nr.3.
3. Badania operacyjne w nowoczesnym zarządzaniu, Praca zbiorowa pod redakcją T.Kasprzaka, PWE Warszawa 1974.
4. Bagga P.C., Chakravarti N.K., Optimal m-Stage Production Schedule /w:/ "Journal of the Canadian Operations Research Society" 1968 nr.2.
5. Balas, E., Machine Sequencing via Disjunctive Graphs. An Implicit Enumeration Algorithm/w:/ "Operations Research", 1969 nr.6.
6. Balut S.J., Scheduling to Minimize the Number of Late Jobs when Set-up and Processing Times are Uncertain /w:/ "Management Science" 1973 nr.11.
7. Bank B., Seiffarte E., Ein "Branch and Bound" Algorithmus eines Reihenfolgeproblems.
8. Baran-Jarosz B., Grabowski W., Wyznaczanie kolejności detali przechodzących w procesie produkcji ciąg operacji /w:/ "Przegląd Statystyczny" 1971 nr.3-4.
9. Barankin E., Precedence Matrices /w:/ "Logistics Review and Military Logistics Journal" 1966 nr.6.
10. Barankin E., The Scheduling Problem as on Algebraic Generalization of ordinary Linear Programming, Maszynopis powielony.

11. Bellman R., *Mathematic Aspects of Scheduling Theory*. "Rand Raport P-651" 1952.
12. Berge C., *Teoria grafów i jej primienienija*. Izdatielstwo Inostrannoj literatury. Moskwa 1962. Tłumaczenie z francuskiego.
13. Biełkina M.W., Gronowa W.J., *Optimizacja posledowatielnosti wypołnienia opieracji*. /w:/ "Automatika i Telemechanika" 1965 nr.11.
14. Blau R.A., *N-Job, One Machine Sequencing Problems Under Uncertainty*, /w:/ "Management Science" 1973 nr.1.
15. Bowman E.H., *Production Scheduling by the Transportation Method of Linear Programming* /w:/ "Operations Research" 1956 nr.4.
16. Bowman E.E., *The Schedule Sequencing Problem* /w:/ "Operations Research" 1959 nr.7.
17. Brown A.P.G., Lomnicki B.A., *Some Applications of the "Branch-and-Bound" Algorithm to the Machine Scheduling Problem* /w:/ "Operations Research" 1966 nr.2.
18. Buslenko N.P. i inni, *Metoda Monte Carlo*, PWN, Warszawa, 1967. Tłumaczenie z rosyjskiego.
19. Campbell H.G., Dudek R.P., Smith M.L., *A Heuristic Algorithm for the n Job m Machine Sequencing Problem*, /w:/ "Management Science" 1970 nr.1.
20. Carlson R.C., Nemhauser G.L., *Scheduling to Minimize Interaction Cost* /w:/ "Operations Research" 1966 nr.1.
21. Chajtman S., *Podstawy organizacji procesu produkcyjnego*, PWE, Warszawa 1971.
22. Cichocki A., *Optymalizacja w deterministycznych modelach sekwencyjnych* /w:/ "Archiwum Automatyki i Telemechaniki" 1972 nr.3.

23. Czernowa G.W., Zadacza opredelenja posledowatielnosti obrabotki detalej imiejuszczich odinakowyje technologiczeskije marszruty /w:/ "Primiemienije Matematiki w Ekonomikie" 1967 nr.4.
24. Dantzig G.B., A Machine-Job Scheduling Model /w:/ "Management Science" 1960 nr.2.
25. Dudek R.A., Teuton O.F., Development of m-Stage-Decision Rule for Scheduling n Jobs Trough m Machines /w:/ "Operations Research" 1964 nr.3.
26. Elementy programowania nieliniowego i dynamicznego. Skrypt pod redakcją W.Bukietyńskiego, WSE Wrocław, 1974.
27. Elementy rachunku ekonomicznego, praca zbiorowa pod redakcją Z.Hellwiga, Warszawa 1972.
28. Galle D., Teoria liniowych modeli ekonomicznych, PWN, Warszawa 1969, tłumaczenie z angielskiego.
29. Giffler B. Thompson G.L., Algorithm for Solving Production Scheduling Problem /w:/ "Operations Research" 1960 nr.8.
30. Gleichgewicht B., Elementy algebry abstrakcyjnej, PZWS Warszawa 1970.
31. Golenko W.J., Tarnowski J.J., Optimalizacja kalendarnych planow metodami naprawленного poiska /w:/ "Kibiernietika" 1970 nr.6.
32. Gospodarowicz A., Zagadnienie układania harmonogramu zajęć przy pomocy komputera /w:/ Prace Naukowe WSE Wrocław 1974 nr.54.
33. Gospodarowicz A., Zastosowanie pewnej metody heurystycznej do rozwiązywania zagadnień z teorii przedsięwzięć czasowych /w:/ Prace Naukowe AE Wrocław 1975 nr.62.
34. Grabowski J., Sysło M., On Some Machine Sequencing Problems I /w:/ "Zastosowania Matematyki" 1973 nr.4.

35. Greenberg H.H., A Branch-Bound Solution to the General Scheduling Problem /w:/ "Operations Research" 1968 nr.2.
36. Hardgrave W.M., Nemhauser G.L., A Geometric Model and a Graphical Algorithm for a Sequencing Problem /w:/ "Operations Research" 1963 nr.6.
37. Heller J., Some Numerical Experiments for an MxJ Flow Shop and Its Decision. Theoretical Aspects /w:/ "Operations Research" 1960 nr.8.
38. Heller J. Logemann G., An Algorithm for the Construction and Evaluation of Possible Schedules /w:/ "Management Science" 1962 nr.8.
39. Hellwig Z., Szware W., O tak zwanym "problemie komiwojaze-
ra" /w:/ "Przegląd Statystyczny" 1960 nr.2.
40. Ignall E., Schrage L., Application of the Branch-and-Bound Technique to Some Flow Shop Scheduling Problems /w:/ "Operations Research" 1965 nr.3.
41. Irikow W.A., Nickotoryje zadacza uporjadoczenja /w:/ "Technicheskaja Kibiernietika" 1970 nr.4.
42. Jaeschke G., Vicinal Sequencing Problems /w:/ "Operations Research" 1972 nr.5.
43. Jankowska-Zorychta Z., Modele sekwencyjne i ich zastosowanie w planowaniu optymalnej organizacji w dyskretnych procesach produkcyjnych, PWN, Warszawa 1973.
44. Jankowska-Zorychta Z., Opis matematyczny pewnego zadania z dziedziny organizacji produkcji /w:/ "Przegląd Statystyczny" 1971 nr.2.
45. Jermakow S.M., Metod Monte-Karlo i smieżnyje woprosy, Nauka, Moskwa 1971.
46. Johnson S.M., Discussion: Sequencing n Jobs on Two Machines with Arbitrary Time Lags /w:/ "Management Science" 1959 nr.3.

47. Johnson S.M., Optimal Two-and. Three-Stage Production Schedules with Setup Times Included /w:/ Naval Research Logistics Quarterly" 1954 nr 1.
48. Kalendarnoje Płanirowanie, Progress, Moskwa 1966. Tłumaczenie z angielskiego.
49. Karuch W., Counterexample to o Proposed Algorithm for Optimal Sequencing of Jobs /w:/ "Operations Research" 1965 nr.2.
50. Kaufman A., Faure R., Badania operacyjne na co dzień, PWN Warszawa 1968, tłumaczenie z francuskiego.
51. Korbut A.A., Finkelsztejn J.J., Programowanie dyskretne, PWN, Warszawa 1974; tłumaczenie z rosyjskiego.
52. Krone M.J., Steiglitz K., Hauristic Programming Solution of a Flow shop-Scheduling Problem /w:/ "Operations Research 1974 nr.3.
53. Kucharczyk J., Sysło M., Algorytmy optymalizacji w języku ALGOL-60, PWN Warszawa 1975.
54. Kulawczuk T., Wydajność pracy w przemyśle socjalistycznym. Statystyczno-ekonometryczne badanie prawidłowości, PWN, Warszawa 1972.
55. Kuzin B.J., Optmalnoje kalendarnoje płanirowanije na potocznych liniach i przedmiotnych uczestkach, Izdatielstwo Leningradzkiego Uniwersiteta 1969.
56. Land A.H., Doig A.G., An Automatic Method of Solving Discrete Programming Problems /w:/ "Econometrica" 1960 nr.3.
57. Lawler E.L., Wood D.E., Branch-and-Bound Methods A Survey /w:/ "Operations Research" 1966 nr.4.
58. Little J.D.C., Murty K.G., Sweeney D.W., Karel C., An Algorithm for the Traweling Salesman. Problem /w:/ "Operations Research" 1963 nr.6.

59. Lomnicki B.A., A "Branch-and-Bound" Algorithm for the Three machine Scheduling Problem /w:/ "Operational Research Quarterly" 1965 nr.1.
60. Manne A.S., On the Job-Shop Scheduling Problem /w:/ "Operations Research" 1960 nr.2.
61. Malcew A.J., Algebraiczeskije systemy. "Nauka", Moskwa 1970.
62. Mc Mohon G.B., Burton P.G., Flow-Shop Scheduling with the Branch-and-Bound Method /w:/ "Operations Research" 1967 nr.3.
63. Meller P., A Review of Job-Shop Scheduling /w:/ "Operational Research Quarterly", 1966 nr.2.
64. Mitten L.G., Sequencing n Jobs on Two Machines with Arbitrary Time Lags /w:/ "Management Science" 1959 nr.5.
65. Miszułowicz B., Zdolność produkcyjna przedsiębiorstwa i zjednoczenia przemysłowego. PWE, Warszawa 1967.
66. Organizacja i planowanie w przedsiębiorstwie przemysłowym. Praca zbiorowa pod redakcją A.Grosmana, PWN, Warszawa, Wrocław 1972 wyd.VI.
67. Ostasiewicz W., Numerowanie permutacji /w:/ Prace Naukowe WSE Wrocław 1972, nr.33.
68. Palmer D.S., Sequencing Jobs Trough a Multi-Stage Process in the Minimum Total Time. A Quick Method of Obtaining a Near Optimum /w:/ "Operational Research Quarterly" 1965 nr.1.
69. Panwalkar S.S., Job Shop Sequencing Problem on Two Machines with Time Lag Constraints./w:/ "Management Science" 1973 nr.3.

70. Pawlak Z., Matematyczne aspekty procesu produkcyjnego. PWE, Warszawa 1969.
71. Pietrow D.A., Płanirowanije potocznogruppowogo proizwodstwa. Moskwa-Leningrad 1966.
72. Pietrow D.A., Sokolicyn S.A., Postrojenie optimalnogo kalendarnogo płana obrabotki dietalej na gruppowych potocznych liniach uproszcziennym matematичесkim metodom /w:/ "Matematiko-ekonomiczeskije problemy". Leningrad 1963.
73. Prabhakar, A Production Scheduling Problem with Sequencing Considerations. /w:/ Management Science 1974 nr.1.
74. Radzikowski W., Zastosowanie programowania liniowego do planowania produkcji wyrobów złożonych. /w:/ "Organizacja-Samorząd-Zarządzanie" 1962 nr.2.
75. Rymarczyk M., O wyznaczaniu przydziałów równoważonych. /w:/ Zeszyty Naukowe AE".
76. Siemionow A.J., Portugal W.M., Zadaczi teorii raspisanji w kalendarnom płanirowanji miełkosierinjogo proizwodstwa. Moskwa 1972.
77. Sikora W., Metody optymalizacyjne kolejności obróbki detali. /w:/ Zeszyty Naukowe WSE Poznań 1971, nr.38.
78. Smith R.D., Dudek R.A., A General Algorithm for Solution of the n-Job m Machine Sequencing Problem of the Flow Shop. /w:/ "Operations Research" 1967 nr.1.
79. Solich R., Metoda wyznaczania kolejności produkcji różnych kompletów detali. Prace CO PAN 1971, nr.47.
80. Starostecki A., Rytmiczność produkcji jednostkowej i małoseryjnej. PWE, Warszawa 1973.

81. Stolarek W., Organizacja cyklu produkcyjnego. WNT, Warszawa 1965.
82. Sysło M., On Some Machine Sequencing Problems III /w:/ "Zastosowania Matematyki" 1974 nr.1.
83. Szkurba W.W., O zadaczach uporządkowania. /w:/ "Kibernetika" 1967 nr.2.
84. Szkurba W. i inni, Zadacze kalendarowego planowania i metody ich rozwiązania. Kijów 1966.
85. Szwarc W., A Counterexample to a Solution Method of the $m \times n$ Job Sequencing-Problem /w:/ Carnegie Mellon University SUPA February 1971.
86. Szwarc W., Elimination Methods in the $m \times n$ Sequencing Problem, Carnegie Mellon University, SUPA February 1971.
87. Szwarc W., Solution of the Akers-Friedman Scheduling Problem. /w:/ "Operations Research" 1960 nr.6.
88. Tilgner B., Optymalizacja planowania operatywnego w produkcji prefabrykatów w budownictwie. AE Poznań, praca doktorska 1975.
89. Titow W.W., Zadacze kalendarowego planowania i jej przybliżone rozwiązanie. /w:/ "Optimalnoje planowanie" Nowosybirsk 1968.
90. Tiutiukin W.K., Nachożdenie optimalnych planow w mnogoperacionnych processach obrabotki izdielej metodom "wiewiej u granic". /w:/ "Primenienie matematiki w ekonomikie" 1970 nr.6.
91. Tiutiukin W.K., Ob optimizaczi kalendarnoj dlitelnosti obrabotki izdielej pri odinakowej posliedowatielnosti zapuska ich na wsiech stankach. /w:/ "Optimalnoje planowanie" 1970.

92. Wagner H.M., An Integer Programming Model for Machine Scheduling. /w:/ "Naval Research Logistics Quarterly" 1959 nr.6.
93. Wagner K.M., Osnovy issledowanija opieraczi, Izdatielstwo "Mir", Moskwa 1972, tłumaczenie z angielskiego.
94. Warężak L., O pewnym uogólnieniu zagadnienia sekwencyjnego. /w:/ Prace Naukowe AE 1975 nr.62.
95. Warężak L., O ustalaniu optymalnych ciągów operacji, Referat na IX Konferencji Ekonometryków i Statystyków Wyższych Szkół Ekonomicznych Katowic, Krakowa i Wrocławia. Karpacz 1973.
96. Warężak L., Przegląd wybranych metod konstruowania optymalnych ciągów operacji. /w:/ Prace Naukowe WSE 1974.nr.44.
97. Wybrane metody rozwiązywania problemów uszeregowania prac na maszynie. Prace CO PAN, 1972 nr.69.
98. Zieliński R., Metody Monte Carlo, WNT, Warszawa 1970.
99. Zykov A.A., Teoria koniecznych grafów, Izdatielstwo "Nauka" Nowosibirsk 1969.