ZESZYTY NAUKOWE POLITECHNIKI ŚLĄSKIEJ 1982

Seria: AUTOMATYKA z. 63 Nr kol. 735

Adam JANIAK

Politechnika Wrocławska

PROBLEM KOLEJNOŚCIOWY GNIAZDOWY Z ROZDZIAŁEM ZASOBÓW

Streszczenie. W pracy rozpatruje się problem kolejnościowy gniazdowy z rozdziałem ograniczonych podzielnych w sposób ciągły zasobów jednego rodzaju przy liniowych modelach operacji. Problem zamodelowano za pomocą grafów dysjunktywnych. Algorytm rozwiązania oparto na metodzie podziału i ograniczeń z mieszaną strategią podziału, wykorzystując opracowaną technikę segmentową.

1. WSTĘP

Dyskretny proces przemysłowy charakteryzuje się przepływem materiałów (w ciągu technologicznym) w postaci pojedynczych elementów lub ich partii. Elementy te są poddawane operacjom obróbki na kolejnych maszynach. Czasy trwania poszczególnych operacji na maszynach mogą być ustalone lub mogą zależeć od ilości zasobów operacjom tym przydzielanych. Globalna ilość zasobów zwykle jest ograniczona. Powstaje zatem problem określenia takiej kolejności wykonywania elementów na poszczególnych maszynach, z zachowaniem określonego porządku technologicznego wykonywania operacji i takiego rozdziału ograniczonych zasobów dla poszczególnych operacji, by uzyskać minimalny czas wykonania całego zadania produkcyjnego. W pracy ograniczymy się do problemu kolejnościowego gniazdowego ("job-shop") przy założeniu, że czasy trwania wszystkich operacji zależą od tego samego rodzaju ograniczonych zasobów.

Klasyczny problem "job-shop" był wielokrotnie rozpatrywany w literaturze, np. w [5]. Pierwsze próby rozwiązania problemu rozpatrywanego w pracy pojawiły się w [4], natomiast podejście, dające przybliżone rozwiązanie podobnego problemu, pojawiło się w [7].

2. MODEL MATEMATYCZNY PROBLEMU

Problem kolejnościowy gniazdowy z rozdziałem zasobów (oznaczany $n/m/G, Res \geqslant 0/C_{max}$) można sformułować następująco. Danych jest n prac $J_1, J_2, \ldots, J_n$, które muszą być wykonane na m maszynach $M_1, M_2, \ldots, M_m$. Praca J_i $(i=1,2,\ldots,n)$ składa się z sekwencji n_i operacji O_j; przy czym operacje ponumerowane są następująco: $j = N_{i-1}+1, \ldots, N_i$, gdzie $N_i = \sum_{l=1}^{i} n_l$. Żadna z maszyn

M_v ($v \in M = \{1,2,\dots,m\}$) nie może wykonywać jednocześnie więcej niż jedną operację, zbiór operacji, które mają być wykonywane na maszynie M_v, będzie oznaczony przez N^v. Operacja O_j ($j = 1,\dots,N_n = n'$) odpowiada wykonywaniu pracy $\mathcal{l}_j$ bez przerwań na maszynie μ_j w czasie p_j.

Zakładamy, że czas p_j wykonywania operacji O_j ($j = 1,2,\dots,N_n = n'$) zależy liniowo od ilości zasobów u_j przydzielonych tej operacji, tzn.: $p_j \triangleq p_j(u_j) \triangleq a_j u_j + b_j$, gdzie $a_j < 0$, $b_j > 0$ są znanymi parametrami. Ponadto założymy, że zbiór dopuszczalnych rozdziałów zasobów jest następujący:

$$U = \Big\{ \bar{u} \in R^{n'} : \bar{u} = [u_1,\dots,u_j,\dots,u_{n'}] \wedge \sum_{j=1}^{n'} u_j \leq \hat{U}$$
$$\wedge\ \alpha_j \leq u_j \leq \beta_j,\ j = 1,\dots,n' \Big\},$$

gdzie $\hat{U}$ jest dysponowaną do rozdziału ilością zasobów, a $0 \leq \alpha_j \leq \beta_j \leq \infty$ są znanymi parametrami.

Należy znaleźć taką kolejność wykonywania operacji na poszczególnych maszynach i taki rozdział zasobów $\hat{U}$ pomiędzy wszystkie operacje, by czas trwania wszystkich prac J_i ($i = 1,2,\dots,n$) był minimalny.

Oczywiste jest, że przy arbitralnie ustalonym rozdziale zasobów $\hat{U}$, rozpatrywany problem sprowadza się do klasycznego problemu kolejnościowego gniazdowego ("job-shop") - $n/m/G/C_{max}$, rozpatrywanego np. w [5].

Jak łatwo zauważyć, rozpatrywany problem jest NP-zupełny.

Celem znalezienia optymalnego rozwiązania sformułowanego problemu, będziemy modelować go za pomocą grafu dysjunktywnego $D = \langle A, U \cup V \rangle$, gdzie:

- A jest zbiorem wierzchołków reprezentujących operacje, włączając fikcyjne operacje: początkową O_{z_0} i końcową O_{z_1}:

$$A = \{z_0, 1, 2, \dots, N_n, z_1\};$$

- U jest zbiorem skierowanych łuków reprezentujących porządek technologiczny wykonywania operacji, należących do poszczególnych prac:

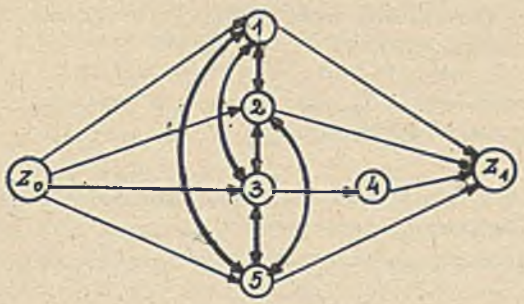
$$U = \left\{ \langle j, j+1 \rangle \Big| \mathcal{l}_j = \mathcal{l}_{j+1} \right\} \cup \left\{ \langle z_0, N_{i-1}+1 \rangle \langle N_i, z_1 \rangle / i = 1,\dots,n \right\};$$

- V jest zbiorem skierowanych łuków reprezentujących wszystkie możliwe kolejności realizacji operacji, wykonywanych na tej samej maszynie:

$$V = \left\{ \langle j_1, j_2 \rangle / \mu_{j_1} = \mu_{j_2},\ j_1 \neq j_2 \right\};$$

- każdy wierzchołek $j \in A$ jest obciążony wagą p_j, przy czym $p_{z_0} = p_{z_1} = 0$.

Multidigraf dysjunktywny dla przykładu $(4/2/G, Res \geq 0/C_{max})$ z rozdz. 5 jest przedstawiony na rys. 1.



Rys. 1. Multidigraf dysjunktywny $D = \langle A, U \cup V \rangle$ dla przykładu z rozdziału 5

Łuk dysjunktywny $\langle j_1, j_2 \rangle$ z każdej pary łuków dysjunktywnych $\{\langle j_1, j_2 \rangle, \langle j_2, j_1 \rangle\}$ nazywamy zgodnym, jeśli został on dołączony do zbioru reprezentantów łuków dysjunktywnych $S_r \subset V$, a łuk przeciwny $\langle j_2, j_1 \rangle$ został odrzucony. Wybór łuku $\langle j_1, j_2 \rangle$ oznacza, że na wspólnej maszynie wykonywana jest operacja j_1 przed j_2.

Dopuszczalne uszeregowanie operacji wykonywanych na wspólnej maszynie zdefiniujemy za pomocą podzbioru $S_r \subset V$, takiego, że:

- $\langle j_1, j_2 \rangle \in S_r \Longleftrightarrow \langle j_2, j_1 \rangle \in V \setminus S_r$

- digraf $D_r(S_r) = \langle A, U \cup S_r \rangle$ jest acykliczny.

Niech $R_s = \{S_1, S_2, \ldots, S_r, \ldots, S_s\}$ będzie rodziną wszystkich takich podzbiorów, a $R_D \triangleq \{D_r = \langle A, U \cup S_r \rangle\}$ - rodzinę wszystkich grafów odpowiadających tym podzbiorom. Zatem R_D jest rodziną digrafów odpowiadających dopuszczalnym strukturom rozwiązania problemu $n/m/G, Res \geq 0/C_{max}$, a problem ten jest równoważny zagadnieniu znalezienia drogi minimaksymalnej:

$$L^{*} \triangleq L^{*}(u^{*}) \triangleq \min_{D_r \in R_D, \bar{u}_r \in U} L_r(\bar{u}_r)$$

z optymalną reprezentacją łuków dysjunktywnych S_r^{*} oraz optymalnym rozdziałem zasobów $\bar{u}^{*} \in U(L_r(u_r)$ jest długością drogi krytycznej w grafie D_r przy rozdziale zasobów $\bar{u}_r \in U)$.

Odnośnie do rozpatrywanego przykładu, graf D_1 odpowiada wykonywaniu operacji na M_1 w kolejności $\langle O_1, O_2, O_3, O_5 \rangle$ i $\langle O_4 \rangle$ na maszynie M_2. Droga krytyczna $L_1(u_1^*)$ o długości równej 8 jest zaznaczona na rys. 2.

Jak łatwo zauważyć, w rozpatrywanym problemie dla danego $D_r \in R_D$ z optymalnym rozdziałem zasobów $\bar{u}_r^* \in U$, mamy zwykle nie jedną, ale wiele dróg krytycznych i oczywiste jest, że droga krytyczna w D_r byłaby najkrótsza, gdyby dopuszczalne rozwiązanie było takie, że wszystkie drogi w D_r stałyby się drogami krytycznymi.

3. WŁASNOŚCI ROZWIĄZANIA PROBLEMU

Celem przedstawienia własności optymalnego rozwiązania problemu wprowadzone zostaną następujące definicje:

Przez C_r^l będziemy oznaczać zbiór łuków l-tej drogi krytycznej w $D_r \in R_D$.

Segmentem P w grafie $D_r \in R_D$ nazywamy ciąg operacji $\langle j_1, j_2, \ldots, j_q \rangle$ zawierający największą liczbę operacji, takich, że istnieje w tym grafie droga:

$$B = \langle\langle j_1, j_2\rangle, \langle j_2, j_3\rangle, \ldots, \langle j_{q-1}, j_q\rangle\rangle,$$

taka, że przez każdy łuk tej drogi przechodzi taka sama ilość dróg krytycznych, tzn. zachodzi:

$$\langle j_i, j_{i+1}\rangle \in S_r \cap C_r^l, \quad i = 1,2,\ldots,g-1; \quad l = l_1, l_2, \ldots, l',$$

gdzie l' jest ilością dróg krytycznych przechodzących przez B.

Na przykład na rys. 2 w grafie D_1 sekwencja operacji $\langle 1,2,3\rangle$ jest segmentem, a sekwencja $\langle 2,3,5\rangle$ nie jest segmentem, ponieważ przez łuk $\langle 2,3\rangle$ przechodzą dwie drogi krytyczne, a przez łuk $\langle 3,5\rangle$ tylko jedna droga krytyczna.

Segmentem wewnętrznym segmentu $\langle j_1, j_2, \ldots, j_q\rangle$ będziemy nazywali następujący ciąg operacji $\langle j_2, j_3, \ldots, j_{q-1}\rangle$.

Wykorzystując przedstawione w [2] własności optymalnego rozdziału zasobów w dowolnym grafie $D_r \in R_D$, dla modeli operacji będących funkcjami wypukłymi oraz tw. 4.2 z [1] można wykazać następującą własność:

Tw. 1. Jeśli spełnione są założenia rozpatrywanego problemu oraz jeśli graf $D_s \in R_D$ otrzymano z grafu $D_r \in R_D$ przez przesunięcie operacji w wewnętrznych segmentach, to

$$L_s(\bar{u}_s^*) \geqslant L_r(\bar{u}_r^*).$$

Stąd wynika wniosek, że jeśli chcemy uzyskać skrócenie drogi krytycznej $L_r(\bar{u}_r^*)$ w $D_r \in R_D$, to należy przesuwać operacje na zewnątrz segmentów (tzn. np. bezpośrednio przed pierwszą lub bezpośrednio za ostatnią operację segmentu), ale - w ten sposób, by uzyskany graf $D_s \in R_D$, tzn., by nie zawierał konturów.

Celem znalezienia digrafu D_* i u^*, wykorzystując powyższy wniosek, stosuje się zmodyfikowaną metodę podziału i ograniczeń, analogicznie jak przy rozwiązywaniu ogólnego problemu sekwencyjnego w [1]. Zgodnie z tą metodą, rozpoczynając od dowolnego grafu początkowego $D_1 = \langle A, U \cup S_1 \rangle \in R_D$ (tzn. od węzła początkowego w drzewie rozwiązań) generujemy - przez przesunięcie pewnej operacji na zewnątrz pewnego segmentu - ciąg grafów bezkonturowych $D_r = \langle A, U \cup S_r \rangle \in R_D$.

Stosując tę metodę w każdym węźle drzewa rozwiązań, tzn. dla każdego digrafu D_r pewne łuki dysjunktywne $F_r \in S_r$ są ustalone i nie mogą one ulec zmianie w żadnym z następników D_s generowanym z D_r.

Rozpatrywany graf $D_r \in R_D$ można odrzucić i cofnąć się do jego bezpośredniego poprzednika, w przypadku gdy:

(i) dolne ograniczenie LB_r na wartość $L_s(\bar{u}_s^*)$ wszystkich możliwyc[h] następników D_s grafu D_r jest nie mniejsze niż aktualne gór[ne] ograniczenie,

(ii) żaden z segmentów nie zawiera więcej niż jedną operację,

(iii) żadna operacja nie może być przesunięta przed pierwszą lub za ostatnią operację w segmencie bez naruszenia ustalonej relacji poprzedzania, wyrażonej przez łuki należące do F_r.

Przez $E_k^b(E_k^a)$ w grafie D_r będzie oznaczany zbiór operacji - kandydatów, które mogą być - przy ustalonych łukach z F_r - przesunięte przed pierwszą (za ostatnią) operację w k-tym segmencie, $k = 1,2,\ldots,K_r$, gdzie K_r - jest liczbą segmentów w D_r. Kandydaci do przesunięcia mogą być oceniani, za pomocą wyrażeń zdefiniowanych w [2] dla modeli operacji będących funkcjami wypukłymi. Optymalnego rozdziału zasobów w dowolnym grafie $D_r \in R_D$ można dokonać za pomocą algorytmu [6] opartego o znajdowanie minimalnego przekroju w grafie.

4. DOLNE OGRANICZENIA

Dolne ograniczenie wartości $L_s(\bar{u}_s^*)$ dla następników grafu D_r można wyznaczyć jako długość drogi krytycznej przy optymalnym rozdziale zasobów $L(F_r, \bar{u}_{F_r}^*)$ w grafie $D(F_r) \triangleq \langle A, U \cup F_r \rangle$ lub przez relaksację możliwości wykonawczych wszystkich maszyn oprócz wybranej wtej. W tym celu dla każdej operacji $j \in N^V$ należy wyznaczyć:

- najwcześniejszy możliwy termin rozpoczęcia jej realizacji v_j, tzn. długość najdłuższej drogi w $D(F_r)$, wybranej z dróg zaczynających się w węźle z_0 i dochodzącej do węzła j; przy czym długość każdej z tych dróg liczona jest przy optymalnym rozdziale zasobów:

$$U - \sum_{i\in N\setminus N'} \varphi_i$$

wśród operacji leżących na rozpatrywanej drodze, a zbiór których będziemy oznaczać przez N'.

- końcówkę q_j, tzn. długość najdłuższej drogi w $D(F_r)$ wybranej z dróg zaczynających się w węźle j i kończących się w węźle z_1, przy czym długość każdej z nich liczona jest identycznie jak wielkości r_j.

Należy wyznaczyć optymalną kolejność operacji ze zbioru N^v na v-tej maszynie i optymalny rozdział zasobów $U - \sum_{i\in N\setminus N^v} \varphi_i$ wśród tych operacji z uwzględnieniem wartości $r_j \geqslant 0$ oraz $q_j \geqslant 0$.

Jest to problem kolejnościowy postaci:

$$N^v(1)r_j \geqslant 0,\ q_j \geqslant 0,\ Res \geqslant 0/C_{max}$$ i jest on NP-zupełny.

Zatem musimy go sprowadzić dalej do postaci:

$$N^v(1)r_j \geqslant 0, q^*, Res \geqslant 0/C_{max} \tag{1}$$

lub do postaci:

$$N^v(1)r^*, q_j \geqslant 0, Res \geqslant 0/C_{max},$$

gdzie:

$$q^* = \min_{j\in N^v} q_j \quad a \quad r^* = \min_{j\in N^v} r_j.$$

Przy czym jeśli rozwiązanie problemu

$$N^v(1)r_j \geqslant 0, Res \geqslant 0/C_{max}$$ oznaczymy przez $L_v(r,\bar{u}_v^*)$,

a rozwiązania problemu

$$N^v(1)q_j \geqslant 0, Res \geqslant 0/C_{max}$$ - przez $L_v(q,\bar{u}_v^*)$,

to rozwiązanie problemu (1) będzie równe

$$L_v(r,\bar{u}_v^*,q^*) = L_v(r,\bar{u}_v^*) + q^*$$

a rozwiązanie problemu (2) będzie równe

$$L_v(r^*,\bar{u}_v^*,q) = L_v(q,\bar{u}_v^*) + r^*.$$

W konsekwencji dla każdego grafu $D_r \in R_D$ dolne ograniczenie ma postać:

$$LB_r = \max\left[L(F_r,\bar{u}_{F_r}^*),\ \max_{v\in M} L_v(r,\bar{u}_v^*,q^*),\ \max_{v\in M} L_v(r^*,\bar{u}_v^*,q)\right]. \tag{3}$$

Należy zaznaczyć, że można przerwać obliczenia dolnego ograniczenia, jeśli wartość któregokolwiek z członów prawej strony wyrażenia będzie nie mniejsza niż wartość aktualnego górnego ograniczenia L^*.

5. ALGORYTM ROZWIĄZANIA PROBLEMU, PRZYKŁAD OBLICZENIOWY

Krok 1 (testowanie). Obliczyć dolne ograniczenie LB_r dla następników grafu D_r. Jeśli $LB_r \geq L^*$ (gdzie L^* - najlepsze do tej pory znalezione rozwiązanie) to przejść do kroku 4. W przeciwnym przypadku przejść do kroku 2.

Krok 2 (rozwiązywanie). Dla digrafu D_{r_*} znaleźć optymalny rozdział $\bar{u}_r^*$ i wyznaczyć $L_r(\bar{u}_r^*)$. Jeśli $L_r(\bar{u}_r^*) < L^*$, to $L^* := L_r(\bar{u}_r^*)$. Określić zbiory E_k^a i E_k^b operacji do przesunięcia, jeśli zbiory te są puste, to przejść do kroku 4, inaczej przejść do kroku 3.

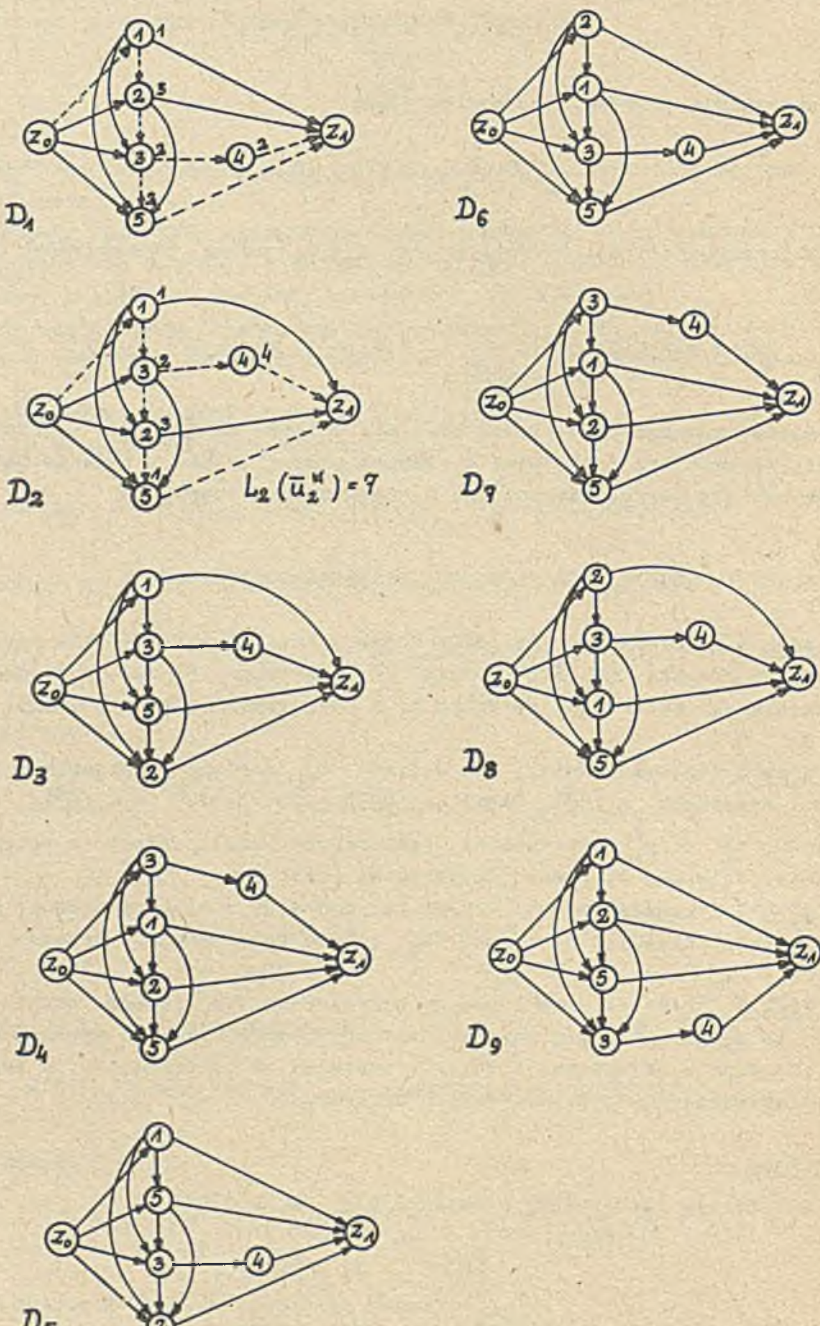
Krok 3 (rozgałęzienia). Wybrać [2] operację w bloku i dokonać jej przesunięcia, otrzymując nowy graf D_s i nowe rozwiązanie. Przejść do kroku 1.

Krok 4 (cofanie). Jeśli nie ma poprzednika, do którego można się cofnąć, to Stop, L^* jest rozwiązaniem optymalnym. Inaczej wykonać czynności związane z aktualizacją zbiorów operacji do przesunięcia w segmentach (analogicznie jak w [1]). Przejść do kroku 3.

Przykład:

Rozpatrzmy następujący problem 4/2/G, Res $\geq$ 0/C_{max}:

$$J_1 = \{O_1\},\quad J_3 = \{O_3,O_4\},$$
$$J_2 = \{O_2\},\quad J_4 = \{O_5\}.$$



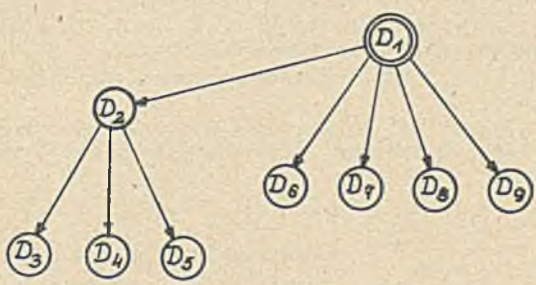
Rys. 2. Ciąg grafów D_r

przy następujących modelach operacji:

$$p_1 = 9-4\,u_1; \quad 0 \leqslant u_1 \leqslant 2, \quad U = 7$$
$$p_2 = 6-3\,u_2; \quad 0 \leqslant u_2 \leqslant 1$$
$$p_3 = 5-3\,u_3; \quad 0 \leqslant u_3 \leqslant 1$$
$$p_4 = 4-2\,u_4; \quad 0 \leqslant u_4 \leqslant 1{,}5$$
$$p_5 = 4 - u_5; \quad 0 \leqslant u_5 \leqslant 3$$

Multidigraf dysjunktywny dla tego przykładu przedstawiono na rys. 1.

Zgodnie z algorytmem startujemy np. z grafu $D_1 = \langle A, U \cup S_1 \rangle$ (rys. 2), graf ten jest korzeniem drzewa rozwiązań, przedstawionego na rys. 3.



Rys. 3. Drzewo rozwiązań

Na rys. 2 przedstawiono struktury grafów $D_1, D_2, \dots, D_9$, wygenerowanych za pomocą algorytmu, przerywaną linią zaznaczono drogi krytyczne; cyfry obok węzłów oznaczają czasy trwania operacji.

Optymalne rozwiązanie przykładu otrzymano dla reprezentacji łuków S_2, tzn. dla grafu D_2 z rozdziałem zasobów $u_1^* = \beta_1$, $u_2^* = \beta_2$, $u_3^* = \beta_3$, $\bar{u}_4^* = \alpha_4$, $u_5 = \beta_5$, $L_*(*) = L_2(\bar{u}_2^*) = 7$.

6. UWAGI KOŃCOWE

Rozpatrywany w pracy problem można łatwo uogólnić na przypadek, gdy operacje wykonywane na pewnej maszynie zależą od pewnego rodzaju zasobów, na innych - od innych rodzajów zasobów, a z kolei na pewnych maszynach czasy te są arbitralnie ustalone.

Warto także zauważyć, że zgodnie z algorytmem w każdej chwili czasu pamiętamy najlepsze znalezione do tej pory rozwiązanie i możemy go wykorzystać jako rozwiązanie przybliżone, co jest niezwykle istotne z praktycznego punktu widzenia. Jest to szczególnie istotne w komputerowych systemach pracujących w czasie rzeczywistym, gdzie przedstawiony algorytm może być wykorzystany do rozdziału pamięci operacyjnej i szeregowania programów na procesorach [3].

LITERATURA

[1] Grabowski J.: Uogólnione zagadnienia optymalizacji kolejności operacji w dyskretnych systemach produkcyjnych. Prace Naukowe ICT Politechniki Wrocławskiej, 1979, Monografie 50/9, rozdz. 4.

[2] Janiak A., Grabowski J.: Problem sekwencyjny z rozdziałem zasobów w dyskretnych procesach produkcyjnych. Zeszyty Naukowe AGH No 146, Kraków 1981.

[3] Janiak A., Grabowski J.: Optymalne szeregowanie programów z rozdziałem pamięci w systemach wieloprocesorowych, Prace II K.K. "Zastosowanie komputerów w przemyśle", Szczecin 1981.

[4] Janiak A., Grabowski J.: Optymalizacja sekwencji operacji z rozdziałem zasobów w dyskretnych systemach produkcyjnych. Zeszyty Naukowe Politechniki Śląskiej z. 54, Gliwice 1980.

[5] Lageweg B.J., Lenstra J.K., Rinnoy Kan A.H.G.: Job-shop scheduling by implicit enumeration, Management Sci, vol. 24 p.p. 441-450.

[6] Романовский Я.В.: Аогоритмы решения оптимальных задач. Москва, 1977, стр. 185-189.

[7] Węglarz J.: Project scheduling with discrete and continuous resources, IEEE Trans. on Systems, Man, and Cybernetics, vol. SMC-9, No 10, 1979.

Recenzent: Doc. dr hab. inż. Konrad WALA

Wpłynęło do Redakcji 15.05.1982 r.

ПРОБЛЕМА ПОСЛЕДОВАТЕЛЬНОСТИ ОПЕРАЦИЙ С РАСПРЕДЕЛЕНИЕМ РЕСУРСОВ

Р е з ю м е

В работе формулируется общая проблема последовательности операций с распределением ограниченных непрерывно делимых ресурсов. Здесь возникает проблема определения такого порядка исполнения задач на отдельных агрегатах, при сохранении заданного технологического режима и такого распределения ограниченных ресурсов между операциями, чтобы получить минимальное время выполнения всего производственного процесса (всех операций). Представлено алгоритм решения этой проблемы на базе метода ветвей и границ.

JOB-SHOP PROBLEM WITH THE ALLOCATION OF RESOURCE

S u m m a r y

This paper is devoted to the general job-shop problem with the resource allocation, denoted by n/m/G, Res$\geq$0/C_{max}. We seek to find a processing order on each machine and allocation of resource to operations such that the maximum completion time is minimized. This problem is NP-complete. It is modelled by means of disjunctive multigraph. A branch and bound technique with mixed strategy of branching is applied to solve our problem. A practical example is presented.