ZESZYTY NAUKOWE POLITECHNIKI ŚLĄSKIEJ 1996
Seria: AUTOMATYKA z. 117 Nr kol. 1337

Józef GRABOWSKI, Paweł WERWIŃSKI
Politechnika Wrocławska

ALGORYTMY OPTYMALNEGO WYBORU MASZYN W DYSKRETNYCH PROCESACH PRODUKCYJNYCH

Streszczenie. W pracy rozważa się zagadnienie, w którym realizacja określonego zadania produkcyjnego w dyskretnych procesach ma być dokonana w określonym terminie i należy dokonać wyboru zestawu maszyn, aby koszt zakupu (dzierżawy) oraz eksploatacji tych maszyn był minimalny. W pracy przedstawia się model matematyczny, własności oraz heurystyczne algorytmy jego rozwiązania.

ALGORITHMS OF OPTIMAL CHOOSING A SET OF MACHINES FOR JOB-SHOP PROBLEM WITH PARALLEL MACHINES

Summary. This paper is devoted to the job-shop problem with parallel machines in which there is the requirement that the completion time of processing of all operations is not greater than a due date. Then the problem arises to choice a set of machines that total costs of used machines is minimal. The heuristic algorithm of solving is presented.

1. Sformułowanie zagadnienia, model matematyczny

Dany jest zbiór zadań

$$J = \left\{J_1, \ldots, J_n\right\},$$

które mogą być wykonywane za pomocą zbioru maszyn

$$M = \left\{M_1, \ldots, M_m\right\}.$$

Zakładamy, że zbiór M jest dostatecznie duży, aby wszystkie zadania mogły zostać wykonane w wymaganym terminie. Każde zadanie $J_i \in J$ składa się z ciągu operacji

$$J_i = \left\langle o_{i1}, \ldots, o_{in_i}\right\rangle.$$

wykonywanych w porządku technologicznym

$$o_{ik-1} < o_{ik} < o_{ik+1}, \quad k = 2, \ldots, n_{i-1}, \; i = 1, 2, \ldots, n.$$

Z każdym zadaniem J_i związana jest marszruta technologiczna

$$v_i = \left\langle v_1^i, \ldots, v_{n_i}^i\right\rangle,$$

zawierająca indeksy stanowisk, na których wykonywane są operacje $o_{ij} \in J_i$. Należy rozumieć, że operacja o_{ij} jest wykonywana na stanowisku v_j^i.

Niech N będzie zbiorem wszystkich operacji.

Zbiór N możemy podzielić na q rozłącznych podzbiorów

$$N = \bigcup_{k=1}^{q} N^k, \; N^k \subset N$$

$$N^k = \left\{ o_{il} \in v \middle| v_l^i = k \right\}, \quad k = 1,\ldots,q$$

Do zbioru N^k należą operacje wykonywane na k-tym stanowisku produkcyjnym.

Podobnie zbiór M podzielić można na q rozłącznych podzbiorów

$$M = \bigcup_{k=1}^{q} M^k, \; M^k \cap M^l \neq \emptyset, \quad k \neq l, \; k,l = 1,\ldots,q$$

gdzie M^k będzie zbiorem maszyn tego samego typu, lecz różnej wydajności.

Zbiór zadań możemy przedstawić równoważnie za pomocą grafu

$$Z = \langle N, RT \rangle,$$

gdzie

N – zbiór operacji,

RT – marszruta technologiczna zadań,

$$RT = \bigcup_{l=1}^{n} \bigcup_{k=1}^{n_l - 1} \left\{ \langle o_{lk}, o_{lk+1} \rangle \right\}$$

W grafie Z wierzchołki są obciążone czasami wykonywania operacji.

Zadanie optymalizacji polegać będzie na dokonaniu takiego przydziału maszyn do poszczególnych operacji oraz ustaleniu wykonywania operacji na maszynach, aby minimalizować łączny koszt użytych maszyn, przy czym termin zakończenia wszystkich operacji nie może przekroczyć żądanego terminu C.

1.1. Model matematyczny

Oznaczenia:

h_w – koszt zastosowania maszyny $M_w \in M$,

t_p – termin rozpoczęcia wykonywania wszystkich operacji ($t_p = 0$),

t_c – termin zakończenia wykonywania wszystkich operacji,

t_j^1 – termin rozpoczęcia wykonywania operacji $o_j \in w$,

t_j^2 – termin zakończenia wykonywania operacji $o_j \in w$,

p_{jw} – czas wykonywania operacji o_j na maszynie $M_w \in M^k$,

C – żądany termin wykonania wszystkich zadań (operacji),

x_{jw} – zmienna decyzyjna

$$x_{jw} = \begin{cases} 1, & \text{jeżeli } o_j \text{ jest przydzielona do } M_w \\ 0, & \text{w przeciwnym przypadku} \end{cases}$$

$\overline{x}$ – wektor zmiennych decyzyjnych x_{jw},

$$N_p = \left\{ o_j \in w \middle| \forall o_i \in N \quad \left(\langle o_i, o_j \rangle \notin RT \right) \right\}$$

$$N_c = \left\{ o_j \in w \middle| \forall o_i \in w \quad \left(\langle o_i, o_j \rangle \notin RT \right) \right\}$$

Zadanie optymalizacji polega na znalezieniu zmiennych decyzyjnych t_p, t_c, t_j^1, t_j^2, x_{jw}, które minimalizują funkcję celu:

(1) $$f(\overline{x}) = \sum_{M_w \in M} h_w \cdot \min\left(1, \sum_{j \in N} x_{jw}\right)$$

przy ograniczeniach

(2) $t_j^2 - t_j^1 \geq \sum_{M_w \in M^k} p_{jw} \cdot x_{jw}$, $\quad o_j \in N,\ k = 1, \ldots, q$,

(3) $t_j^1 - t_i^2 \geq 0$, $\quad \langle o_i, o_j \rangle \in RT$,

(4) $t_j^1 - t_0 \geq 0$, $\quad o_j \in N_p$,

(5) $t_c - t_j^2 \geq 0$, $\quad o_j \in N_c$,

(6) $t_0,\ t_j^1,\ t_j^2 \geq 0$, $\quad o_j \in N$,

(7) $(x_{iw} = 1) \wedge (x_{jw} = 1) \Rightarrow \left(t_j^1 - t_i^2 \geq 0\right) \dot{\vee} \left(t_j^1 - t_j^2 \geq 0\right)$, $\quad M_w \in M^k,\ o_i,\ o_j \in N^k,\ k = 1, \ldots, q$,

(8) $\sum_{M_w \in M^k} x_{jw} = 1$, $\quad k = 1, 2, \ldots, q$,

(9) $x_{jw} \in \{0, 1\}$, $\quad o_j \in N,\ M_w \in M$,

(10) $t_c \leq C$.

Wybór zestawu maszyn realizowany jest zmiennymi decyzyjnymi x_{jw}. Jeżeli maszyna M_w została wybrana do wykonania przynajmniej jednej operacji $o_j \in N$, wówczas zmienna x_{jw} przyjmie wartość równą 1 i w funkcji celu przejawi się koszt h_w, bowiem $\min\left(1, \sum_{j \in N} x_{jw}\right) = 1$. Jeżeli jednak M_w nie została wybrana do zestawu maszyn, to $\min\left(1, \sum_{j \in N} x_{jw}\right) = 0$ i koszt h_w w funkcji celu nie pojawi się.

Warunek (2) mówi, że różnica pomiędzy terminem rozpoczęcia i zakończenia operacji nie może być mniejsza niż czas jej trwania przy określonej maszynie przydzielonej do wykonywania tej operacji (patrz ogr. (8)). Warunek (3) zapewnia zachowanie porządku technologicznego RT podczas wykonywania operacji. Warunek (4) oznacza, że termin t_c jest terminem rozpoczęcia wykonywania wszystkich zadań, natomiast (5) wyraża, iż termin t_0 jest terminem zakończenia wykonywania wszystkich operacji. Warunek (7) mówi, że jeżeli dla dwóch różnych operacji o_i i o_j ze zbioru N^k istnieje możliwość przydzielenia tej samej maszyny $M_w \in M^k$, wówczas realizacja jednej z tych operacji musi się zakończyć przed rozpoczęciem drugiej. Ograniczenie (8) prezentuje warunek, iż dla wykonywania dowolnej operacji $o_j \in w^k$ może być przydzielona tylko jedna maszyna ze zbioru M^k. Ograniczenie (10) oznacza, że wybór zestawu maszyn oraz kolejność wykonywania operacji zapewni wykonywanie wszystkich operacji w żądanym terminie C.

Każdy z wektorów $\overline{x}$ generuje pewien zestaw MASZYN WYBRANYCH

$$ZW(\overline{x}) = \left\{ M_w \in M \middle| \sum_{j \in N} x_{jw} \geq 1 \right\}.$$

Funkcja celu przyjmie zatem postać

$$f(\overline{x}) = \sum_{M_w \in ZW(\overline{x})} h_w ,$$

oraz $ZW \subseteq M$.

Dla pewnego dopuszczalnego $\overline{x} \in X$ zadanie sprowadzi się do znalezienia takiej kolejności zadań na maszynach, aby $t_c \leq C$.

1.2. Graf dysjunktywny

Podobnie jak w pracach [1], [2], [3], [5] dla dopuszczalnego rozwiązania $\overline{x}_p \in X$ można skonstruować graf dysjunktywny:

$$\overline{G}_{x_p} = \left\langle A, U, V_{x_p} \right\rangle$$

gdzie:

A – zbiór wierzchołków

U – zbiór łuków technologicznych

V_{x_p} – zbiór łuków dysjunktywnych reprezentujących wszystkie uszeregowania operacji na przydzielonych do ich realizacji maszynach

$$V_{x_p} = \bigcup_{k=1}^{m} V_{x_p}^k$$

Dla danego $\overline{G}_{x_p}$ możemy stworzyć rodzinę reprezentacji R_{x_p} taką, że każdy element tej rodziny zawiera po jednym łuku z pary dysjunktywnej.

$$R_{x_p} = \left\{ S_1^{\overline{x}_p}, S_2^{\overline{x}_p}, \ldots, S_v^{\overline{x}_p} \right\}$$

Dla określonego wektora $\bar{x}_p \in X$ stworzyć możemy rodzinę grafów

$$R_{Gx_p} = \left\{ G_{vx_p} = \left\langle A, U \cup S_v^{x_p} \right\rangle \middle| S_v^{x_p} \in R_{x_p} \right\}.$$

Następnie z R_{Gx_p} możemy wybrać taką podrodzinę R'_{Gx_p} grafów, które nie zawierają konturów.

$$R'_{Gx_p} = \left\{ G_{vx_p} \in R_{Gx_p} \middle| G_{vx_p} \quad \textit{nie zawiera konturów} \right\}.$$

W dowolnym grafie $G_{vx_p} \in R'_{Gx_p}$ możemy wyznaczyć $C_v^{x_p}$ – ciąg wierzchołków drogi krytycznej, i jej długość $L_{vx_p} \in t_c$, oraz zbiór bloków L_B, będących zbiorami co najmniej dwóch operacji należących do drogi krytycznej i wykonywanych kolejno na tej samej maszynie.

$$L_B = \left\{ L_1^{x_p}, \ldots, L_b^{x_p} \right\}.$$

Spośród grafów $G_{dx_p} \in R'_{Gx_p}$ możemy wybrać te, dla których długość drogi krytycznej jest nie większa od C.

$$R''_{Gx_p} = \left\{ G_{dx_p} \in R'_{Gx_p} \middle| L_{dx_p} \le C \right\}.$$

Dla tak opisanego problemu zadanie optymalizacji sprowadzi się do znalezienia takiego wektora $\bar{x}_0 \in X$, który minimalizuje funkcję celu

$$f(\bar{x}_0) = \min_{\bar{x}_p \in X} \left\{ \sum_{M_w \in ZW(\bar{x}_p)} h_w \middle| R''_{Gx_p} \ne \emptyset \right\}$$

2. Schemat algorytmu

Dolne ograniczenie funkcji celu

Zbiór M jest podzielony na rozłączne podzbiory M^k, z których każdy zawiera maszyny *k*-tego typu.

Jeżeli zbiór operacji N^k nie jest pusty, to przynajmniej jedna maszyna typu *k* musi być wybrana.

Zatem spośród maszyn ze zbioru M wybieramy maszyny o minimalnym koszcie z każdego typu maszyn *k* (tzn. $M_{wk} \in M^k$ oraz $h_{wk} = \min_{M_w \in M^k} h_w$). Otrzymamy zbiór ZM, który zawiera po jednej "najtańszej" maszynie każdego typu:

$$ZM = \left\{ M_{w1}, M_{w2}, \ldots, M_{wq} \right\}.$$

Wartość dolnego ograniczenia funkcji celu będzie wynosić:

$$h^0 = \sum_{M_{wk} \in ZM} h_{wk}\,.$$

Generowanie przydziałów operacji do maszyn oraz określanie kolejności ich wykonywania odbywać się będzie na zbiorze dyspozycyjnym maszyn $ZD \subseteq M$.

Rozpoczynamy dla $ZD_1 := ZM$.

Ogólnie, na zbiorze ZD_i określamy wektor przydziału maszyn $\overline{x}_i \in X$, optymalną kolejność, a także spośród ZD_i wyznaczamy zbiór ZW (maszyn wybranych).

Jeżeli długość drogi krytycznej $L_{0x_i} \leq C$, to wartość $f(\overline{x}_i)$ będzie rozwiązaniem problemu i algorytm kończy przebieg.

Jeżeli $L_{0x_i} > C$, to należy wyznaczyć inny zbiór ZD_{i+1}, taki że $L_{0x_{i+1}} \leq C$.

Algorytm 1

Zbiór początkowy niech będzie równy zbiorowi maszyn o minimalnym koszcie ZM.

$$ZD_1 = ZM.$$

Krok 1.

Za pomocą dowolnego algorytmu (np. przedstawionego w [4]) w zbiorze ZD_p wyznacz

– wektor $\overline{x}_p \in X$, $ZW(\overline{x}_p)$ oraz kolejność operacji na maszynach,

– drogę minimaksymalną w grafie $\overline{G}_{x_p}$.

Oblicz $f(\overline{x}_p)$. Jeżeli $L_{0x_p} \leq C$, to STOP, a wartość $f(\overline{x}_p)$ przyjmij za optymalną.

Jeżeli $L_{0x_p} > C$, to przejdź do *Krok* 2.

Krok 2.

Jeżeli $ZD_p = M$, to zagadnienie nie posiada rozwiązania. W przeciwnym przypadku spośród bloków $B_1, \ldots, B_b$ na drodze krytycznej $C_0^{x_p}$ wybierz B_k o najdłuższej drodze

$$L(B_k) = \max_{1 \leq s \leq b} L(B_s).$$

Jeżeli $B_k \subseteq N^k$, to wybierz maszynę M_p ze zbioru $M^k - ZD_p$ o minimalnym koszcie h_p i dołącz ją do zbioru maszyn dysponowanych ZD_{p+1}

$$h_p = \min_{M_w \in M^k - ZD_p} h_w$$

$$ZD_{p+1} = ZD_p \cup \{M_p\}$$

$$ZD_p = ZD_{p+s}$$

Przejdź do *Krok* 1.

Algorytm 2 (Algorytm "górskiego turysty pieszego")

Algorytm jest naturalnym rozwinięciem algorytmu 1 o fazę zejścia w dół z kosztami użytych maszyn.

FAZA I – wejście na "górę kosztów".

W fazie I wykorzystujemy algorytm 1. Faza I będzie trwać tak dopóty, dopóki nie uzyskamy takiego ZD_p, że spełniony będzie warunek (10), tj. $L_{0x_i} \le C$.

FAZA II – zejście w "dolinę kosztów".

Krok 1.

Jeżeli $ZD_p = ZM$, to STOP.

W przeciwnym przypadku wyznacz sumę T_k czasów operacji wykonywanych na każdym stanowisku i takich, które należą do ścieżki krytycznej. Spośród wszystkich stanowisk wyznacz stanowisko j takie, że

$$T_j = \min_{k=1,\dots,q} T_k .$$

Spośród maszyn ze zbioru ZD_p usuń tę maszynę typu j, która jest najdroższa, pod warunkiem że w zbiorze ZD_p znajdują się co najmniej 2 maszyny typu j. W przeciwnym przypadku znajdź inne stanowisko l, np. kolejne o najkrótszym czasie T_l.

Przyjmij:

$$j := l.$$

Krok 2.

Wyznacz w zbiorze ZD_p:

– wektor $\overline{x}_p \in X$, kolejność wykonywania operacji na maszynach, zbiór ZW,

– drogę krytyczną C_o^{x} w $\overline{G}_x$.

Jeżeli $L_{0\overline{x}_p} \le C$, to *Krok* 1.

Jeżeli $L_{0\overline{x}_p} > C$, to wróć do FAZA I

Algorytm kończy bieg, gdy $ZD_p = ZM$ lub po określonej liczbie kroków, ze względu na możliwość pojawienia się cykli.

3. Uogólnienie

Postawiony na wstępie problem da się uogólnić o dodatkowe elementy, które mogą pojawić się w zastosowaniach praktycznych. Pierwszym z nich jest dopuszczenie przekroczenia terminu C, jednak pojawią się wtedy dodatkowe koszty (kary) związane z przekroczeniem terminu.

$$T = \max(0, t_c - C) .$$

Koszty te będą miały postać:

$$KW(\bar{t}) = q\left[\max(0, t_c - C)\right] = q(T) .$$

Funkcja celu przyjmie postać:

$$f(\bar{x},\bar{t}) = \sum_{M_w \in M} h_w \cdot \min\left(1, \sum_{j \in N} x_{jw}\right) + q(T) .$$

Drugim uogólnieniem będzie dołączenie kosztów związanych z łącznym czasem trwania realizacji operacji przydzielonych maszynie M_w:

$$T_N = \sum_{j \in N} p_{jw} \cdot x_{jw} ,$$

Koszty te będą miały postać:

$$KE_w(\bar{x}) = q_w\left(\sum_{j \in N} p_{jw} \cdot x_{jw}\right) = q_w(T_w) .$$

Funkcja celu przyjmie teraz postać:

$$f(\bar{x},\bar{t}) = \sum_{M_w \in M}\left[h_w \cdot \min\left(1, \sum_{j \in N} x_{jw}\right) + q_w(T_w)\right] .$$

LITERATURA

1. Balas E.: Machine-Sequencing via Disjunctive Graphs-An Implicit Enumeration Algorithm. Oper. Res., vol. 17, (1969), s. 941-957.
2. Bouma R.W.: Job-Shop Scheduling: a comparison of three schemes in Branch-and-Bound Approach. Master's Thesis, Erasmus University, Rotterdam 1982.
3. Grabowski J.: Uogólnienie zagadnienia optymalizacji kolejności operacji w dyskretnych systemach produkcyjnych. Monografie ICT PWr., Wrocław (1979).
4. Grabowski J., Nowicki E., Smutnicki C., Zdrzałka S.: Teoria i algorytmy rozwiązywania zadań optymalizacji dyskretnej dla zagadnień kolejnościowych. Raport z serii SPRAWOZDANIA ICT, nr 8/85 (1985).
5. Grabowski J., Nowicki E.: Zagadnienia kolejnościowe gniazdowe z maszynami równoległymi. Materiały Konferencji Naukowo-Technicznej "Problematyka Eksploatacji Maszyn i Urządzeń w Ujęciu Systemowym", Kraków (1986), s. 70-75.
6. Rinnooy Kan A.H.G.: Machine Scheduling Problem Classification, Complexity and Computation. Niihoff, The Haque (1976).

Recenzent: Prof. dr hab. inż. Ryszard Gessing

Wpłynęło do Redakcji do 30.06.1996 r.

Abstract

This paper deals with the job-shop problem with parallel machines. The problem can be formulated as follows. There is the set of jobs $J_1,...,J_n$ which should be carried out by using the set of various type machines $M_1,...,M_m$ and q is the number of these types. Each job consist of a sequence of operations $J_i = \left\langle O_{i1}, O_{i2}, ..., O_{in_i} \right\rangle$. We shall assume that processing times of operations are fixed and different for different machines. The problem arises to determine the allocation of machines to the individual operation and to determine a sequence of operations on machines, taking into account technological requirements, that the total costs of allocated machines are minimal, subject to the completion time of processing of all operations is not greater then a given due date C. The mathematical model and some properties are presented. The heuristic algorithm for solving this problem is given.