ZESZYTY NAUKOWE POLITECHNIKI ŚLĄSKIEJ 2000
Seria: AUTOMATYKA z. 129 Nr kol. 1474

Zbigniew BUCHALSKI
Politechnika Wrocławska

SZEREGOWANIE ZADAŃ W SYSTEMACH WIELOMASZYNOWYCH Z CZASEM REALIZACJI ZALEŻNYM OD ILOŚCI ZASOBÓW

Streszczenie. Praca dotyczy zagadnienia czasowo-optymalnego przydziału n zadań niezależnych i zasobu nieodnawialnego do m różnych maszyn równoległych. Zakłada się, że występuje stałość przydziału zasobów w czasie wykonywania całego zbioru zadań. Dla zadanej funkcji czasu realizacji zadań sformułowano model matematyczny zagadnienia i podano algorytm heurystyczny. Przedstawiono wyniki eksperymentów obliczeniowych.

TASK SCHEDULING IN MULTIMACHINES SYSTEMS WITH EXECUTION TIME DEPENDS ON THE NUMBER OF RESOURCES

Summary. In the paper the problem of time-optimal allocation of n independent tasks and nonrenewable resources to m different parallel machines is considered. We assume, that is constancy of resources allocation in execution time all tasks set. For some tasks execution time function the mathematical model of this problem is formulated and an heuristic algorithm is presented. Some results of executed numerical experiments are presented.

1. Wprowadzenie

Od wielu lat prowadzi się intensywne badania problematyki czasowo-optymalnego szeregowania zadań i rozdziału zasobów [1, 5, 6, 7, 8, 9]. Prezentowana praca bazuje na wynikach tych badań i jest kontynuacją wcześniejszych prac autora [2, 3, 4].

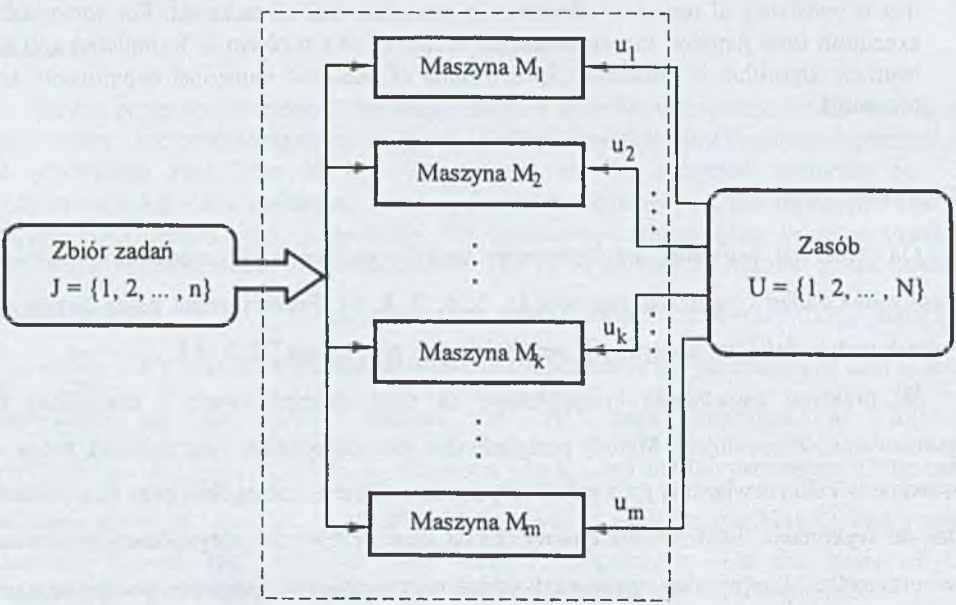
W praktyce zagadnienia kolejnościowe są dość skomplikowane i najczęściej są zagadnieniami *NP*-zupełnymi. Metody przeglądu (np. metoda podziału i ograniczeń), które są stosowane w celu rozwiązania tych zagadnień, są czasochłonne, szczególnie przy dużej liczbie zadań do wykonania. Fakt ten jest typowy dla tej klasy problemów optymalizacji dyskretnej i w przypadku, kiedy zależy nam na krótkim czasie obliczeń, jedynym podejściem jest zastosowanie algorytmów heurystycznych.

Celem niniejszej pracy jest znalezienie takiego uszeregowania n niezależnych niepodzielnych zadań na m maszynach równoległych oraz takiego przydziału zasobu nieodnawialnego do tych maszyn, aby minimalizować czas zakończenia wykonywania wszystkich zadań. W pracy przedstawiono model matematyczny zagadnienia, algorytm heurystyczny rozwiązujący problem oraz podano wyniki eksperymentów obliczeniowych przeprowadzonych za pomocą tego algorytmu.

2. Model matematyczny zagadnienia

Dany jest system wielomaszynowy, przedstawiony na rys.1, złożony z m maszyn równoległych tworzących zbiór $M = \{M_1, M_2, \ldots, M_k, \ldots, M_m\}$. Każda maszyna reprezentuje urządzenie realizujące określony zbiór zadań niezależnych niepodzielnych. Zbiór wszystkich zadań przeznaczonych do wykonania na m maszynach oznaczmy przez $J = \{1, 2, \ldots, n\}$. Każde zadanie może być wykonane na dowolnej maszynie i w trakcie realizacji nie może być przerywane. Liczba zadań do wykonania jest większa od liczby maszyn ($n > m$).

Zakładamy, że zasób jest dyskretny i dysponujemy N identycznymi jednostkami tego zasobu; $N \geq m$. Przez $u_k \in U$ oznaczmy tę ilość zasobu, która zostanie przydzielona maszynie M_k w trakcie wykonywania zadań uszeregowanych na tej maszynie, gdzie $U = \{1, 2, \ldots, N\}$.



Rys.1. System maszyn równoległych
Fig.1. Parallel machines system

Ograniczenia zasobowe są następujące:

$$\sum_{k=1}^{m} u_k \le N, \quad 1 \le k \le m, \quad u_k \in U.$$

Czas wykonania i-tego zadania na maszynie M_k, jeżeli przydzielono jej u_k jednostek zasobu nieodnawialnego, określony jest funkcją:

$$T_i(u_k, k) = a_{ik} + \frac{b_{ik}}{u_k}, \quad u_k \in U, \quad 1 \le k \le m, \quad i \in J \tag{1}$$

Parametry a_{ik}>0, b_{ik} >0 charakteryzują i-te zadanie i k-tą maszynę.

Problem polega na znalezieniu takiego uszeregowania zadań na maszynach i takiego przydziału zasobów nieodnawialnych do maszyn, aby minimalizować czas zakończenia wykonywania wszystkich zadań.

Oznaczając przez $J_k \subset J$ zbiór zadań uszeregowanych na maszynie M_k, należy rozwiązać następujący problem minimalizacji:

$$Q = \min_{\substack{J_1, J_2, \ldots, J_m \\ u_1, u_2, \ldots, u_m}} \max_{1 \le k \le m} \left\{ \sum_{i \in J_k} T_i(u_k, k) \right\}, \tag{2}$$

przy ograniczeniach:

(i) $J_s \cap J_t = \emptyset, \quad s, t = 1, 2, \ldots, m \quad s \ne t, \quad \bigcup_{k=1}^{m} J_k = J,$

(ii) $\sum_{k=1}^{m} u_k \le N, \quad u_k \in U, \quad k = 1, 2, \ldots, m$.

W celu rozwiązania powyższego problemu minimalizacyjnego dokonamy relaksacji warunku $u_k \in U$ zastępując go warunkiem $0 \le u_k \le N$, $k = 1, 2, \ldots, m$. Po rozwiązaniu odpowiedniego zadania minimalizacji dyskretno-ciągłej otrzymane wartości u_k zaokrąglamy do wartości całkowitych (patrz krok 11 algorytmu heurystycznego).

Dokonując wspomnianej relaksacji problem (2) można sformułować jako następujący problem optymalizacji dyskretno-ciągłej:

$$Q = \min_{\substack{J_1, J_2, \ldots, J_m \\ u_1, u_2, \ldots, u_m}} \max_{1 \le k \le m} \left\{ \sum_{i \in J_k} T_i^1(u_k, k) \right\} \tag{3}$$

przy następujących ograniczeniach:

(i) $J_s \cap J_t = \emptyset, \quad s, t = 1, 2, \ldots, m \quad s \ne t, \quad \bigcup_{k=1}^{m} J_k = J,$

(ii) $\sum_{k=1}^{m} u_k \le N, \quad u_k \ge 0, \quad k = 1,2,...,m$

gdzie:

$T_i^1 : [0,N] \times \{1,2,...,m\} \to R^+$ jest rozszerzeniem funkcji $T_i : \{1,2,...,N\} \times \{1,2,...,m\} \to R^+$ i określone jest przez funkcję:

$$T_i^1(u_k,k) = a_{ik} + \frac{b_{ik}}{u_k}, \quad u_k \in [0,N], \quad 1 \le k \le m, \quad i \in J. \tag{4}$$

Przez $u_k^*, J_k^*, k = 1,2,...m$ oznaczmy rozwiązania zadania (3). W dalszych rozważaniach pomocny będzie następujący lemat:

LEMAT 1

Jeżeli $u_k^*, J_k^*, k = 1,2,...m$ są rozwiązaniami optymalnymi zadania (3), to:

(i) $\sum_{k=1}^{m} u_k^* = N; \quad u_k^* > 0, \quad k : J_k^* \ne \emptyset, \quad k = 1,2,...m;$

$u_k^* = 0, \quad k : J_k^* = \emptyset, \quad k = 1,2,...m;$

(ii) $\sum_{i \in J_k^*} T_i^1(u_k^*,k) = const, \quad k : J_k^* \ne \emptyset, \quad k = 1,2,...,m.$

Warunek (i) w LEMACIE 1 mówi, że w przydziale czasowo-optymalnym zasobu i zadań do maszyn wykorzystuje się wszystkie jednostki zasobu, a warunek (ii), że czasy pracy tych maszyn, które wykonują jakieś zadania, są identyczne.

Zdefiniujmy funkcję $F(J_1, J_2, \dots, J_m)$ określoną dla m zbiorów $J_1, J_2, \dots, J_m$, dla których zachodzi ograniczenie 3(i). Wartość tej funkcji jest rozwiązaniem następującego układu równań:

$$\begin{cases} \sum_{i \in J_k} a_{ik} + \frac{\sum_{i \in J_k} b_{ik}}{u_k} = F(J_1, J_2, \dots, J_m), \; k : J_k \ne \emptyset, \; k = 1,2,...,m \\ \sum_{k : J_k \ne \emptyset} u_k = N; \quad u_k > 0 \quad k : J_k \ne \emptyset, \quad k = 1,2,...,m. \end{cases} \tag{5}$$

Wykorzystując LEMAT 1 oraz (5), zadanie (3) przyjmie więc ostateczną postać:

$$Q = \min_{J_1, J_2, \dots, J_m} F(J_1, J_2, \dots, J_m) \tag{6}$$

przy następujących ograniczeniach:

(i) $J_s \cap J_t = \emptyset, \quad s,t = 1,2,...,m, \quad s \ne t,$

(ii) $\bigcup_{k=1}^{m} J_k = J$.

Jeżeli $J_1^*, J_2^*, ..., J_m^*$ są rozwiązaniem zadania (6), to $u_k^*, J_k^*, k = 1,2,...,m$, gdzie:

$$u_k^* = \begin{cases} \dfrac{\sum\limits_{i \in J_k^*} b_{ik}}{F(J_1^*, J_2^*, ... J_m^*) - \sum\limits_{i \in J_k^*} a_{ik}}; & k: J_k^* \neq \emptyset, \quad 1 \leq k \leq m \\ 0 \quad ; & k: J_k^* = \emptyset, \quad 1 \leq k \leq m \end{cases} \tag{7}$$

jest rozwiązaniem zadania (3).

3. Algorytm heurystyczny i wyniki obliczeń

Kolejne kroki algorytmu heurystycznego są następujące:

Krok 1. Połóż $u_k = \dfrac{N}{m}$, $k = 1,2,...,m$.

Krok 2. Oblicz czasy wykonywania zadań na poszczególnych maszynach $T_i(u_k, k) = a_{ik} + \dfrac{b_{ik}}{u_k}$, $i = 1, 2, ..., n$, $k = 1, 2,...,m$.

Krok 3. Uszereguj malejąco zadania wg czasu ich trwania.

Krok 4. Przydziel początkowych m zadań uszeregowanych w **kroku 3** kolejnym m maszynom $M_1, M_2, ..., M_k, ..., M_m$.

Krok 5. Przydziel kolejne zadania uszeregowane w **kroku 3** do maszyn w odwrotnej kolejności, tzn. do maszyn $M_m, ..., M_k, ..., M_m, M_2, M_1$. Jeżeli lista zadań się nie wyczerpie, to wykonaj następny krok, w przeciwnym wypadku skocz do **kroku 7**.

Krok 6. Przydziel kolejne zadania uszeregowane w kroku 3 do maszyn w kolejności $M_1, M_2, ..., M_k, ..., M_m$. Jeżeli lista zadań się nie wyczerpie, skocz do **kroku 5**, w przeciwnym wypadku przejdź do **kroku** 7.

Krok 7. Oblicz czasy wykonywania zadań uszeregowanych na poszczególnych maszynach.

Krok 8. Usuń najkrótsze zadanie z maszyny o najdłuższym czasie wykonywania zadań i przydziel je do maszyny o najkrótszym czasie wykonywania zadań.

Krok 9. Oblicz maksymalny czas wykonywania zadań na maszynach po zamianie zadań wg **kroku 8**. Jeżeli czas ten ulegnie skróceniu, to wróć do **kroku 7**, w przeciwnym przypadku cofnij ostatnio wykonaną czynność w **kroku 8** i zakończ wykonywanie szeregowania zadań na maszynach.

Krok 10. Dla przydziału $J_1, J_2, ..., J_m$ wyznaczonego w **krokach 3÷9** przydziel zasoby $u_1, u_2, ..., u_m$ kolejnym maszynom $M_1, M_2, ..., M_m$ zgodnie z zależnością (7), w której wielkości $J_1^*, J_2^*, ..., J_m^*$ należy zastąpić przez $J_1, J_2, ..., J_m$; wielkość $F(J_1, J_2, ..., J_m)$ należy wyznaczyć z układu równań (5).

Krok 11. Wyznacz dyskretne ilości zasobów $\hat{u}_k$, $k = 1,2,...,m$ według zależności:

$$\hat{u}_{\alpha(k)} = \begin{cases} \lfloor u_{\alpha(k)} \rfloor + 1, & k = 1,2,\ldots,\Delta, \\ \lfloor u_{\alpha(k)} \rfloor, & k = \Delta+1, \Delta+2,\ldots,m, \end{cases}$$

gdzie $\Delta = N - \sum_{j=1}^{m} \lfloor u_j \rfloor$ oraz α jest permutacją elementów zbioru $\{1,2,\ldots,m\}$ taką, że $u_{\alpha(1)} - \lfloor u_{\alpha(1)} \rfloor \geq u_{\alpha(2)} - \lfloor u_{\alpha(2)} \rfloor \geq \ldots \geq u_{\alpha(m)} - \lfloor u_{\alpha(m)} \rfloor$.

Jeżeli istnieją takie maszyny, którym przydzielono zerowe ilości zasobów, to przydziel każdej z tych maszyn po jednej jednostce zasobu pobierając je z kolejnych maszyn poczynając od maszyny, której przydzielono największą ilość zasobów.

Na bazie powyższego algorytmu przeprowadzono eksperymenty obliczeniowe. Przyjęto $n = 20, 50, 70$, $m = 3, 5, 10$ oraz $N = 100$. Następnie dla każdej kombinacji liczby zadań n oraz liczby maszyn m wygenerowano 20 zestawów a_{ik}, b_{ik} ze zbioru {0.1, 0.2,..., 9.9. 10.0} przez generator o jednostajnym rozkładzie prawdopodobieństwa. Łącznie przebadano $3 \cdot 3 \cdot 20 = 180$ instancji. Wyniki przedstawione zostały w tabeli 1.

Tabela 1

Wyniki eksperymentów obliczeniowych algorytmu heurystycznego

Liczba zadań/liczba maszyn n/m	Średni czas obliczeń [sek]	Mediana czasów obliczeń [sek]
20/3	2,8	3
20/5	3,9	4
20/10	4,9	5
50/3	4,2	4
50/5	9,4	9
50/10	12,7	12
70/3	8,9	9
70/5	12,8	13
70/10	16,4	17

LITERATURA

1. Błażewicz J., Lenstra J.K., Rinnooy Kan A.H.G.: Scheduling subject to resource constraints: classification and complexity. Discrete Appl. Math., Mathemtisch Centrum. Amsterdam 1980.
2. Buchalski Z.: Zagadnienie czasowo-optymalnego szeregowania zadań i rozdziału zasobów w systemie wielomaszynowym. Prace Konferencji Naukowo-Technicznej "Problematyka budowy i eksploatacji maszyn i urządzeń w ujęciu systemowym", AGH Kraków 1986, str. 62-69.
3. Buchalski Z.: Algorytm dla problemu szeregowania zadań na maszynach dla pewnych funkcji czasu wykonywania zadań. Zeszyty Naukowe Politechniki Śląskiej, s.Automatyka, z. 94, Gliwice 1988, str. 61-68.

4. Buchalski Z.: Zagadnienie przydziału zadań i zasobów do maszyn równoległych dla pewnych funkcji czasu wykonywania zadań. Wydawnictwa AGH, Automatyka, Półrocznik tom 1, zeszyt 1, Kraków 1997, str. 61-70.
5. Ishii H., Martel C., Masuda T., Nishida T.: A generalized uniform processor system. Oper. Res. Vol. 33, nr 2, 1985, pp. 346-362
6. Janiak A. : Wybrane problemy i algorytmy szeregowania zadań i rozdziału zasobów. Akademicka Oficyna Wydawnicza PLJ, Warszawa 1999.
7. Janiak A., Kovalyov M.: Single machine scheduling subject to deadlines and resources dependent processing times. European Journal of Operational Research 1996, vol. 94, p. 284-291.
8. Węglarz J.: Project Scheduling with Continously-Divisible Doubly Constrained Resources, Mgt. Sci, vol. 27, nr 3. 1981.
9. Węglarz J.: Synthesis problems in allocating continuous, doubly constrained resources among dynamic activities, Operation Research 1990, pp. 715-730.

Recenzent: Dr inż. K.Pieńkosz

Abstract

In the paper the problem of time-optimal tasks scheduling and nonrenewable resources allocation on different, parallel machines in multimachines systems is considered. Results obtained here are for some processing time function $T_i(u_k, k) = a_{ik} + \frac{b_{ik}}{u_k}$, $u_k \in U$, k=1, 2, ... m, i = 1, 2, ... , n where $a_{ik} > 0, b_{ik} > 0$ are parameters characterized i-th task and k-th machine, u_k is number of resources allocated to M_k machine.

We assume, that all n tasks are independent and number of tasks is greater than number of machines. We also assume, that is constancy of resources allocation in execution time all tasks set.

This tasks scheduling and resources allocation problem in multimachines systems can be formulated as follows: given n independent tasks , to be run on the m machines at the same time , how should the N available resources be partitioned among m machines, that schedule length criterion is minimized. Because our problem belongs to the class of NP-hard problems we propose an heuristic algorithm. Some results of executed numerical experiments are presented.