ZESZYTY NAUKOWE POLITECHNIKI ŚLĄSKIEJ 2006
Seria: AUTOMATYKA z. 144 Nr kol. 1727

Mateusz GORCZYCA, Adam JANIAK
Politechnika Wrocławska

MINIMALIZACJA POZIOMU ZASOBU PRZY OGRANICZENIU NA DŁUGOŚĆ USZEREGOWANIA ZADAŃ O MODELACH DYNAMICZNYCH NA PROCESORACH RÓWNOLEGŁYCH

Streszczenie. Rozpatrywany jest problem minimalizacji poziomu zasobu koniecznego do wykonania zadań na równoległych procesorach przy ograniczeniu na długość uszeregowania. Zadania dane są modelami dynamicznymi, gdzie prędkość wykonywania zadania w danej chwili zależy od ilości przydzielonego mu zasobu. Zasób jest odnawialny i podzielny w sposób ciągły. Wykazano własności problemu i skonstruowano algorytm optymalnego rozdziału zasobu.

THE MINIMIZATION OF RESOURCE LEVEL ON PARALLEL PROCESSORS WITH CONSTRAINED MAKESPAN AND DYNAMIC MODEL OF TASKS

Summary. The problem of resource level minimization on a parallel processors with constrained makespan is considered. The dynamic model of task is given, where the speed of change of task processing state in every moment depends on amount of alloted resource. The resource is renewable and continuously divisible. Some properties of the problem are proved and algorithm of optimal resource allocation is constructed.

1. Wstęp

Klasyczny model zadania, gdzie czas jego wykonywania jest zadany z góry i stały, często okazywał się zbyt prosty do modelowania problemów praktycznych z zadowalającą dokładnością. Już w latach 60. zaproponowano model dynamiczny, gdzie wykonywanie zadania interpretuje się jako proces, w którym szybkość zmiany stanu zależy w każdej chwili od ilości pewnego zasobu przydzielonego do tego procesu. Zasób taki jest podzielny w sposób ciągły, a jego ilość zadana z góry i ograniczona.

Dotychczas badano wieloprocesorowe problemy szeregowania zadań o modelach dynamicznych dla kryterium długości uszeregowania [4, 1], średniego czasu przepływu [3] oraz maksymalnej nieterminowości [1]. Rozważano również problem z zadanym ograniczeniem na długość uszeregowania przy kryterium poziomu zasobu [2] - wykazano pewne własności tego problemu i zaproponowano algorytm heurystyczny. Problem ten sformułowano w rozdziale 2. W rozdziale 3 wykazano nowe własności problemu, które wykorzystano w rozdziale 4 do skonstruowania algorytmu optymalnego rozdziału zasobu.

2. Sformułowanie problemu

Dany jest zbiór niepodzielnych zadań $J = \{1, 2, \ldots, n\}$ do wykonania na zbiorze procesorów $M = \{1, 2, \ldots, m\}$. Przyjmujemy, że zadany jest najpóźniejszy możliwy termin zakończenia wszystkich zadań $\hat{C} > 0$. Do wykonania zadania oprócz procesora niezbędny jest także zasób podzielny w sposób ciągły. Szybkość wykonywania zadania $j \in J$ w chwili t zależy od ilości przydzielonego mu zasobu:

$$\dot{x}_j(t) \triangleq \frac{dx_j(t)}{dt} = f_j(u_j(t)),$$

gdzie:

- $x_j(t)$ - stan zadania j w chwili $t \in [0, \hat{C}]$,
- $f_j(\cdot)$ - funkcja rosnąca, ciągła, wklęsła i spełniająca warunek $f_j(0) = 0$,
- $u_j(t)$ - ilość zasobu przydzielonego do zadania j w chwili $t \in [0, \hat{C}]$.

Każde zadanie $j \in J$ ma zadany stan początkowy $x_j(0) = 0$ oraz stan końcowy $\hat{x}_j > 0$, który musi zostać osiągnięty, aby zadanie zostało wykonane. Oznaczmy przez $S_j \triangleq \max\{t : x_j(t) = 0\}$ i $C_j \triangleq \min\{t : x_j(t) = \hat{x}_j\}$, $j = 1, \ldots, n$, moment odpowiednio rozpoczęcia i zakończenia wykonywania zadania j.

Niech funkcja $q_j(t)$ sygnalizuje wykonywanie zadania j w następujący sposób:

$$q_j(t) \triangleq \begin{cases} q_j(t) = 1 \text{ gdy } S_j < t \leqslant C_j, \\ q_j(t) = 0 \text{ w przeciwnym wypadku.} \end{cases}$$

Oznaczmy przez $Q(t) \triangleq \sum_{j=1}^{n} q_j(t)$ liczbę zadań wykonywanych w momencie t. *Rozdziałem zasobu* będziemy nazywać funkcję wektorową $\mathbf{u}(t) \triangleq [u_1(t), u_2(t), \ldots, u_n(t)]$ spełniającą warunki:

1. $0 \leqslant S_j < C_j \leqslant \hat{C}$ dla $j = 1, \ldots, n$,
2. $u_j(t) \geqslant 0$ dla $t \in [S_j, C_j]$, $j = 1, \ldots, n$,
3. $u_j(t) = 0$ dla $t \in [0, S_j) \cup (C_j, \hat{C}]$, $j = 1, \ldots, n$,
4. $u_j(t)$ jest przedziałami ciągłe dla $t \in [S_j, C_j]$, $j = 1, \ldots, n$,

5. $\int_{S_j}^{C_j} f_j(u_j(t))dt = \hat{x}_j,\ j = 1, \ldots, n,$

6. $Q(t) \leqslant m$ dla $t \in [0, \hat{C}]$.

Problem polega na znalezieniu takiego rozdziału zasobu $\mathbf{u}^*(t)$, który zminimalizuje kryterium poziom zasobu $R(\mathbf{u}(t)) \triangleq \max_{t \in [0,\hat{C}]} \sum_{j=1}^{n} u_j(t)$. Analizowany problem jest NP-trudny [2].

3. Własności problemu

Lemat 1. *Dla dowolnych* $0 \leqslant u_1 < u_2$, $x_1 \geqslant 0$, $x_2 > 0$, $T_1 > 0$, $T_2 > 0$ *oraz ciągłej, wklęsłej i rosnącej funkcji* $f(\cdot)$ *spełniającej warunek* $f(0) = 0$, *jeżeli zachodzi* $x_1 = f(u_1) \cdot T_1$ *oraz* $x_2 = f(u_2) \cdot T_2$, *to:*

(a) dla każdej liczby u_1' *takiej, że* $u_1 < u_1' \leqslant f^{-1}(K)$, *istnieje liczba* u_2' *taka, że* $f^{-1}(K) \leqslant u_2' < u_2$ *oraz*

$$x_1 + x_2 = f(u_1) \cdot T_1 + f(u_2) \cdot T_2 = f(u_1') \cdot T_1 + f(u_2') \cdot T_2, \quad (1)$$
$$u_1 \cdot T_1 + u_2 \cdot T_2 \geqslant u_1' \cdot T_1 + u_2' \cdot T_2, \quad (2)$$

gdzie $K = \frac{T_1 \cdot f(u_1) + T_2 \cdot f(u_2)}{T_1 + T_2}$;

(b) dla każdej liczby u_2' *takiej, że* $f^{-1}(K) \leqslant u_2' < u_2$, *istnieje liczba* u_1' *taka, że* $u_1 < u_1' \leqslant f^{-1}(K)$ *oraz spełnione są warunki (1) i (2).*

Dowód. Rozważmy najpierw punkt (a). W pierwszej kolejności wykażemy, że istnieje liczba u_2', dla której jest spełniony warunek (1). Weźmy dowolną liczbę u_1' taką, że $u_1 < u_1' \leqslant f^{-1}(K)$. Przekształcając równanie (1) otrzymujemy $f(u_2') = \frac{T_1}{T_2} \cdot (f(u_1) - f(u_1')) + f(u_2)$. Ponieważ $\frac{T_1}{T_2} > 0$ i $f(u_1) - f(u_1') < 0$ to $f(u_2') < f(u_2)$, a więc skoro funkcja $f(\cdot)$ jest rosnąca, to $u_2' < u_2$. Z drugiej strony, aby zachodziło $u_2' \geqslant f^{-1}(K)$, musi zachodzić $f(u_2') \geqslant K$. Korzystając z równania (1) otrzymujemy $\frac{T_1}{T_2} \cdot (f(u_1) - f(u_1')) + f(u_2) \geqslant K$, które po prostych przekształceniach sprowadza się do $u_1' \leqslant f^{-1}(K)$.

Pokażemy teraz, że mając liczbę u_2' spełniającą warunek (1), spełniony jest także warunek (2). W związku z tym, że $f(u_1) < f(u_1') \leqslant f(u_2') < f(u_2)$, można zapisać:

$$f(u_1') = (1 - \alpha) \cdot f(u_1) + \alpha \cdot f(u_2),\ \alpha \in (0, 1), \quad (3)$$
$$f(u_2') = \beta \cdot f(u_1) + (1 - \beta) \cdot f(u_2),\ \beta \in (0, 1). \quad (4)$$

Podstawiając równania (3) i (4) do równania (1), otrzymujemy po prostych przekształceniach $\frac{\alpha}{\beta} = \frac{T_2}{T_1}$. Z drugiej strony wykorzystując funkcję odwrotną $f^{-1}(\cdot)$ z równań (3) i (4) otrzymujemy:

$$\begin{aligned} u_1' &= f^{-1}\left((1-\alpha)\cdot f(u_1)+\alpha\cdot f(u_2)\right), \\ u_2' &= f^{-1}\left(\beta\cdot f(u_1)+(1-\beta)\cdot f(u_2)\right). \end{aligned}$$

Z wypukłości funkcji $f^{-1}(\cdot)$ mamy:

$$\begin{aligned} u_1' &\leqslant (1-\alpha)\cdot f^{-1}\left(f(u_1)\right)+\alpha\cdot f^{-1}\left(f(u_2)\right), \\ u_2' &\leqslant \beta\cdot f^{-1}\left(f(u_1)\right)+(1-\beta)\cdot f^{-1}\left(f(u_2)\right), \end{aligned}$$

czyli

$$u_1' \leqslant (1-\alpha)\cdot u_1+\alpha\cdot u_2, \tag{5}$$

$$u_2' \leqslant \beta\cdot u_1+(1-\beta)\cdot u_2. \tag{6}$$

Wykorzystując proporcję $\frac{\alpha}{\beta}=\frac{T_2}{T_1}$ oraz sumując nierówności (5) - (6) otrzymujemy nierówność (2). W analogiczny sposób można wykazać poprawność punktu (b) lematu.

□

Jeżeli w pewnym przedziale czasu $[t_1, t_2]$ przydział zasobu do zadań jest stały i przybiera wartości $u_j, j=1,\ldots,n$, to zmiana stanu $x_j(t_2)-x_j(t_1)=\int_{t_1}^{t_2} f_j(u_j(t))dt = f_j(u_j)\cdot\int_{t_1}^{t_2} dt = f_j(u_j)\cdot(t_2-t_1)$.

3.1. Przypadek $n \leqslant m$

Twierdzenie 1. *Każdemu rozdziałowi zasobu* $\mathbf{u}(t)$*, powodującemu w przedziale czasu o niezerowej długości pewną zmianę stanów zadań, odpowiada rozdział zasobu o niegorszej wartości funkcji kryterialnej, stały w tym przedziale i powodujący taką samą zmianę stanów zadań.*

Dowód. Niech $\mathbf{u}(t)$ będzie w rozpatrywanym przedziale funkcją schodkową. Weźmy dwa spośród schodków o czasie trwania odpowiednio T_1 i T_2 oraz o wartościach rozdziału zasobu odpowiednio u_{j1} i u_{j2}, $j=1,\ldots,n$. Zmiana stanów zadań w tych przedziałach to $x_{ji}=f_j(u_{ji})\cdot T_i$, $j=1,\ldots,n$, $i=1,2$, natomiast poziom zasobu $R_i=\sum_{j=1}^n u_{ji}$, $i=1,2$.

Zastąpmy rozdział zasobu o wartościach u_{j1} i u_{j2} rozdziałem o wartości $u_j, j=1,\ldots,n$, w przedziale o długości $T=T_1+T_2$, zmieniającym stan zadań o $x_j=x_{j1}+x_{j2}$ i o poziomie zasobu $R=\sum_{j=1}^n u_j$. Równanie $x_j=x_{j1}+x_{j2}$ można zapisać jako $f_j(u_j)\cdot T = f_j(u_{j1})\cdot T_1 + f_j(u_{j2})\cdot T_2$ i dalej $f_j(u_j)=\frac{f_j(u_{j1})\cdot T_1+f(u_{j2})\cdot T_2}{T}$, a następnie $u_j=f_j^{-1}(\frac{f_j(u_{j1})\cdot T_1+f(u_{j2})\cdot T_2}{T})$. Oznaczając $\alpha=\frac{T_1}{T}$, przy czym zachodzi $0<\frac{T_1}{T}=\frac{T_1}{T_1+T_2}<1$, otrzymujemy $1-\alpha=\frac{T_2}{T}$. Zatem $u_j=f_j^{-1}(\alpha\cdot f_j(u_{j1})+(1-\alpha)\cdot f_j(u_{j2}))$. Z wypukłości funkcji $f_j^{-1}(\cdot)$ mamy $u_j\leqslant\alpha\cdot f_j^{-1}(f_j(u_{j1}))+(1-\alpha)\cdot f_j^{-1}(f(u_{j2}))$, czyli $u_j\leqslant\alpha\cdot u_{j1}+(1-\alpha)\cdot u_{j2}$. Stąd $R\leqslant\alpha\cdot R_1+(1-\alpha)\cdot R_2$, z czego otrzymujemy $R\leqslant\max\{R_1,R_2\}$.

Łącząc w ten sposób kolejne schodki funkcji $\mathbf{u}(t)$ otrzymamy stały w czasie rozdział zasobu, przy czym poziom zasobu będzie mniejszy niż poziom zasobu w funkcji schodkowej. Z twierdzenia o całkowaniu Lebesgue'a wiemy, że każdą funkcję ciągłą możemy przybliżyć funkcją schodkową z dowolnie małym błędem. □

Z twierdzenia 1 wynika, że przydział zasobu $\mathbf{u}^*(t) = \mathbf{u}^* = const.$ taki, że $u_i^* = f_i^{-1}\left(\frac{\hat{x}_i}{\hat{C}}\right)$, $i = 1, \ldots, n$, daje optymalną wartość funkcji kryterialnej $R(\mathbf{u}^*) = \sum_{j=1}^{n} u_j^*$.

3.2. Przypadek $n > m$

Niech π będzie dla rozdziału zasobu $\mathbf{u}(t)$ taką permutacją numerów zadań, że $C_{\pi(1)} \leqslant C_{\pi(2)} \leqslant \ldots \leqslant C_{\pi(n)}$. Załóżmy dla uproszczenia, że π jest naturalną permutacją zbioru J, tj. $\pi(j) = j$, $j = 1, \ldots, n$.

Własność 1. *Istnieje optymalny rozdział zasobu, spełniający dla pewnej permutacji π warunki:*

1. *Rozdział zasobu jest stały w przedziałach* $[0, C_1]$ *i* $(C_{j-1}, C_j]$, $j = 2, \ldots, n$.
2. *Zachodzi* $C_n = \hat{C}$.

Dowód. Pokażemy, że dla każdego rozdziału zasobu $\mathbf{u}(t)$ istnieje rozdział zasobu $\mathbf{u}'(t)$ spełniający warunki *1-2* o niegorszej wartości funkcji kryterialnej. Istnienie rozdziału zasobu $\mathbf{u}'(t)$ spełniającego warunek *1* wynika wprost z twierdzenia 1.

Załóżmy dalej dla uproszczenia, że rozdział zasobu $\mathbf{u}(t)$ spełnia warunek *1*. Jeżeli dla rozdziału $\mathbf{u}(t)$ zachodzi $C_n < \hat{C}$, to zgodnie z twierdzeniem 1 możemy rozdział zasobu o zerowej wartości w przedziale $(C_n, \hat{C}]$ oraz niezerowej w przedziale $(C_k, C_n]$, gdzie $k = \max\{j : C_j < C_n\}$, zastąpić rozdziałem zasobu stałym w przedziale $(C_k, \hat{C}]$ o niegorszej wartości funkcji kryterialnej, spełniając tym samym warunek *2*. □

4. Algorytm optymalnego rozdziału zasobu

W dalszej części artykułu dla każdego rozdziału zasobu $\mathbf{u}(t)$, spełniającego warunki *1-2*, będziemy nazywać *przedziałami* te spośród przedziałów $[0, C_1]$, $(C_j, C_{j+1}]$, $j = 1, \ldots, n-2$ oraz $(C_{n-1}, \hat{C}]$, które mają niezerową długość. Zatem każdy rozdział zasobu $\mathbf{u}(t)$ spełniający warunki *1-2* składa się z $p \leqslant n$ przedziałów. Oznaczmy przez $W_i = \{j : q_j(t) = 1 \text{ w } i\text{-tym przedziale}\}$, $i = 1, \ldots, p$, zbiór zadań wykonywanych w i-tym przedziale. Dla takiego rozdziału zasobu istnieje ciąg $\mathbf{w} \triangleq W_1, \ldots, W_p$, zwany dalej *uszeregowaniem*, spełniający warunki: *I.* $|W_i| \leqslant m$, $i = 1, \ldots, p$; *II.* $\bigcup_{i=1}^{p} W_i = J$; *III.* Jeżeli zadanie pojawia się w więcej niż jednym zbiorze uszeregowania $\mathbf{w}$, to są to zbiory o kolejnych numerach. Warunek *I* zapewnia spełnienie ograniczenia na liczbę procesorów, warunek

II gwarantuje, że wszystkie zadania będą wykonane, a warunek *III* zapewnia nieprzerywalność zadań.

Oznaczmy przez x_{ji} oraz u_{ji}, $j = 1, \ldots, n$, $i = 1, \ldots, p$ odpowiednio zmianę stanu oraz ilość zasobu przydzieloną do zadania j-tego w i-tym przedziale, natomiast przez T_i, $i = 1, \ldots, p$ czas trwania i-tego przedziału. Optymalne zmiany stanów zadań x_{ji} oraz długości trwania przedziałów T_i dla dowolnego uszeregowania **w** są otrzymywane przez rozwiązanie następującego problemu programowania wypukłego:

$$\min \max_{i=1,\ldots,p} \sum_{j=1}^{n} f_j^{-1}\left(\frac{x_{ji}}{T_i}\right) \tag{7}$$

$$\text{p.o.} \quad \sum_{i=1}^{p} T_i = \hat{C}, \tag{8}$$

$$\sum_{k=1}^{p} x_{jk} = \hat{x}_j,\ j = 1, \ldots, n, \tag{9}$$

$$x_{jk} \geqslant 0,\ j \in W_k,\ k = 1, \ldots, p, \tag{10}$$

$$x_{jk} = 0,\ j \notin W_k,\ k = 1, \ldots, p, \tag{11}$$

$$T_i > 0,\ k = 1, \ldots, p. \tag{12}$$

Ograniczenie (8) wynika z warunku *2* własności 1, ograniczenie (9) zapewnia, że każde z zadań zostanie w całości wykonane, natomiast ograniczenia (10)-(12) wynikają ze sformułowania problemu i definicji uszeregownia **w**. Dla danego uszeregowania **w** optymalny rozdział zasobu u_{ji}, $j = 1, \ldots, n$, $i = 1, \ldots, p$ jest wyliczany przez rozwiązanie problemu (7)-(12) i przy wykorzystaniu wzoru $u_{ji} = f_j^{-1}\left(\frac{x_{ji}}{T_i}\right)$.

5. Podsumowanie

W artykule zbadano własności problemu dla istotnej z praktycznego punktu widzenia klasy wklęsłych funkcji prędkość/zasób. Skonstruowano algorytm optymalnego rozdziału zasobu dla przypadku liczby zadań nie większej od liczby procesorów. Algorytm ten wykorzystano następnie przy konstruowaniu algorytmu optymalnego rozdziału zasobu dla dowolnej liczby zadań. Do wyznaczenia optymalnego rozdziału zasobu konieczne jest rozwiązanie problemu optymalizacji wypukłej. Autorzy są w trakcie badania własności problemu pozwalających ograniczyć zarówno przestrzeń rozdziałów zasobu, jak i liczbę uszeregowań.

LITERATURA

1. Janiak A.: Wybrane problemy i algorytmy szeregowania zadań i rozdziału zasobów. Akademicka Oficyna Wydawnicza PLJ, Warszawa 1999.

2. Janiak A., Słoniński P.: Minimizing the resource Level in a Scheduling Problem with Dynamic Models of Jobs on Parallel Machines. Automatyka 5, 2001, p. 283–288.
3. Józefowska J., Węglarz J.: Discrete-continuous scheduling problems - Mean completion time results. European Journal of Operational Research, Vol. 94, No. 2, 1996, p. 302–309.
4. Józefowska J., Węglarz J.: On a methodology for discrete-continuous scheduling. European Journal of Operational Research, Vol. 107, No. 2, 1998, p. 338–353.

Recenzent: Prof. dr hab. inż. Tadeusz Czachórski

Abstract

The resource level minimization problem on a parallel processors is considered. Maximal makespan is given in advance. The model of tasks is dynamic one. For each task a concave function is given, that relates the speed of change of task processing state at a time to the amount of continuously divisible resource allotted to this task. The resource is renewable. For each task there is also given its beginning and its final state, which must be achieved in order to complete this task. The objective is to find resource allocation over time and task schedule, that minimize the resource level.

In this paper, some properties of the problem are proved, which allows us to construct algorithm, that finds optimal resource allocation for a given task schedule by solving a convex optimization problem.