ZESZYTY NAUKOWE POLITECHNIKI ŚLĄSKIEJ 1984
Seria: AUTOMATYKA z. 74 Nr kol. 810

Zbigniew Buchalski
Politechnika Wrocławska
Instytut Sterowania i Techniki Systemów

PEWNE ZAGADNIENIE PRZYDZIAŁU ZADAŃ I ZASOBU NIEODNAWIALNEGO DO DWÓCH RÓŻNYCH MASZYN W DYSKRETNYM SYSTEMIE PRODUKCYJNYM

Streszczenie. W pracy formułuje się zadanie czasowo-optymalnego przydziału zasobu nieodnawialnego podzielnego w sposób dyskretny i n zadań do dwóch różnych maszyn równoległych. Po rezygnacji z dyskretności tego zasobu wykazuje się, że takie zadanie jest NP-trudne i konstruuje się algorytm /dla zasobu podzielnego w sposób ciągły/ oparty na metodzie podziału i ograniczeń. Wyznaczone wartości zasobu zaokrągla się do najbliższych liczb naturalnych. W pracy podaje się wyniki eksperymentów obliczeniowych dla n=15, 30, 40,..., 100.

1. Wstęp

Od pewnego czasu prowadzi się intensywne badania problematyki czasowo-optymalnego sterowania rozdziałem zadań i zasobów w systemach typu kompleks operacji [1,2,5,7,8]. Przegląd rezultatów badań w tym zakresie oraz ich syntetyczna charakterystyka zawarta jest w pracach [2,8]. Niniejsza praca bazuje na wynikach badań tej problematyki i jest kontynuacją wcześniejszych prac autora [3,4].

Zakładamy, że dysponujemy jednym rodzajem zasobu nieodnawialnego, podzielnego w sposób dyskretny, który należy rozdzielić pomiędzy dwie maszyny. Czas wykonania zadania zależy od tego, na jakiej maszynie jest ono wykonywane i jaką ilość zasobu jej przydzielono.

W rozdziale drugim formułuje się rozpatrywane zadanie czasowo-optymalnego przydziału zasobu nieodnawialnego i n zadań do dwóch różnych maszyn i wykazuje się, że jest ono NP-trudne. Następnie w rozdziale trzecim podaje się pewne własności czasowo-optymalnego przydziału i konstruuje algorytm oparty na metodzie podziału i ograniczeń, wyznaczający rozwiązanie optymalne. W rozdziale czwartym przedstawia się wyniki eksperymentów obliczeniowych przeprowadzonych na tym algorytmie dla n = 15,30,40,...,100.

2. Sformułowanie problemu. Podstawowe własności

W pracy rozpatrujemy dyskretny system produkcyjny, o którym zakładamy, że:

/i/ posiada dwie różne maszyny M_1, M_2, na których należy wykonać n niezależnych zadań $J = \{1,2,\ldots,n\}$,

/ii/ zadanie może być wykonywane na dowolnej maszynie i w trakcie jego wykonywania nie może być przerywane,

/iii/ posiada N jednakowych jednostek zasobu nieodnawialnego,

/iv/ maszyna M_k w trakcie wykonywania zadań jej przydzielonych $I_k \subset J$ wykorzystuje część jednostek zasobu nieodnawialnego u_k i w każdej chwili może wykonywać tylko jedno zadanie; $k = 1,2$ $(u_1 + u_2 \leqslant N)$,

/v/ czas wykonania zadania i-tego na maszynie M_k, jeżeli przydzielono jej u_k jednostek zasobu nieodnawialnego, określony jest funkcją

$$T_i(u_k,k) = a_{ik} + \frac{b_{ik}}{u_k}, \quad u_k \in \{1,2,\dots,N\}, \quad k = 1,2, \quad i \in J, \tag{1}$$

gdzie $a_{ik} > 0$, $b_{ik} > 0$ - parametry określające zadanie i-te i maszynę M_k.

Rozważany przez nas problem polega na znalezieniu czasowo-optymalnego przydziału zadań I_k i zasobu nieodnawialnego u_k do maszyn M_k, $k = 1,2$, przy spełnieniu powyższych założeń.

Sformułowany powyżej problem można przedstawić jako następujące zadanie minimalizacji dyskretnej:

$$\min_{\substack{I_1,I_2\\u_1,u_2}} \max\left\{\sum_{i \in I_1} T_i(u_1,1), \sum_{i \in I_2} T_i(u_2,2)\right\} \tag{2}$$

przy ograniczeniach

$$\begin{aligned} &/i/ \quad I_1 \cup I_2 = J, \quad I_1 \cap I_2 = \emptyset, \\ &/ii/ \quad u_1 + u_2 \leqslant N, \\ &/iii/ \quad u_1,u_2 \text{ - całkowite dodatnie.} \end{aligned} \tag{3}$$

Przedstawione zadanie jest dość skomplikowane ze względu na ograniczenie /iii/. Dlatego też w dalszym ciągu pracy przyjmuje się założenie upraszczające polegające na rezygnacji z dyskretności zasobu nieodnawialnego. Przy tym założeniu wyznacza się przydział czasowo-optymalny zadań i zasobu nieodnawialnego do maszyn a następnie zaokrągla się otrzymane liczby zasobu nieodnawialnego do najbliższych liczb naturalnych. Postępowanie to jest uzasadnione tym, że gdy liczba N jednostek zasobu nieodnawialnego jest stosunkowo duża , to względna odległość między wartością kryterium dla tak otrzymanego przydziału, a wartością kryterium bez tego założenia jest stosunkowo mała. Oszacowanie tej odległości przedstawione jest w rozdziale 4. Uwzględniając powyższe uwagi w dalszym ciągu będziemy rozwiązywać następujące zadanie:

$$\min_{\substack{I_1,I_2\\u_1,u_2}} \max\left\{\sum_{i \in I_1} \tilde{T}_i(u_1,1), \sum_{i \in I_2} \tilde{T}_i(u_2,2)\right\} \tag{4}$$

przy ograniczeniach

$$\begin{aligned}&/i/\ I_1 \cup I_2 = J,\ I_1 \cap I_2 = \emptyset,\\&/ii/\ u_1 + u_2 \leqslant N,\ u_1 \geqslant 0,\ u_2 \geqslant 0,\end{aligned} \tag{5}$$

gdzie $\tilde{T}_i : [0,N] \times \{1,2\} \longrightarrow R^+$ jest rozszerzeniem funkcji $T_i : \{1,2,\ldots,N\} \times \{1,2\} \longrightarrow R^+$ określonym identycznym wzorem jak T_i, $i \in J$.

Uwzględniając fakt, że funkcje $\tilde{T}_i$ są malejące, zadanie (4),(5) można zapisać w równoważnej postaci:

$$\min_{I \subset J} \left[\min_{u \in [0,N]} \max\left\{ \sum_{i \in I} \tilde{T}_i(u,1),\ \sum_{i \in \bar{I}} \tilde{T}_i(N-u,2) \right\} \right], \tag{6}$$

gdzie $\bar{I} = J \setminus I$.

Z postaci powyższego zadania wynika, że jeżeli jego rozwiązaniem jest u^*, I^* oraz $I^* \neq \emptyset$ i $I^* \neq J$, to zachodzi

$$\sum_{i \in I^*} \tilde{T}_i(u^*,1) = \sum_{i \in \bar{I}^*} \tilde{T}_i(N-u^*,2) \triangleq Q(I^*), \tag{7}$$

czyli czas pracy obu maszyn jest identyczny.

Po prostych przekształceniach z (7) dostaniemy

$$\begin{aligned} Q(I^*) = \frac{1}{2}\Bigg[&\sum_{i \in I^*} (a_{i1} + c_{i1}) + \sum_{i \in \bar{I}^*} (a_{i2} + c_{i2}) + \\ &+ \sqrt{\left[\sum_{i \in I^*} (a_{i1} + c_{i1}) + \sum_{i \in \bar{I}^*} (a_{i2} + c_{i2})\right]^2 - 4\left(\sum_{i \in I^*} a_{i1} \sum_{i \in \bar{I}^*} a_{i2} + \sum_{i \in I^*} a_{i1} \sum_{i \in \bar{I}^*} c_{i2} + \sum_{i \in \bar{I}^*} a_{i2} \sum_{i \in I^*} c_{i1}\right)}\ \Bigg] \end{aligned} \tag{8}$$

$$\text{zaś } u^* = \frac{N \cdot \sum_{i \in I^*} c_{i1}}{Q(I^*) - \sum_{i \in I^*} a_{i1}}, \tag{9}$$

gdzie $c_{ik} \triangleq \dfrac{b_{ik}}{N}$.

Po przeanalizowaniu przypadków $I^* = \emptyset$, $I^* = J$ ostatecznie dostaniemy, że w celu rozwiązania zadania (4),(5) należy rozwiązać następujące zadanie

$$\min_{I \subset J} Q(I). \tag{10}$$

Następnie, jeżeli I^* jest rozwiązaniem zadania (10), to jako rozwiązanie zadania (4) $u_1^*, u_2^*, I_1^*, I_2^*$ przyjąć $I_1^* = I^*$, $I_2^* = \bar{I}^*$, $u_1^* = u^*$, $u_2^* = N - u^*$, gdzie u^* jest wyznaczone wg wzoru (9).

Lemat 1

Zadanie (10) jest NP-trudne.

Dowód lematu wynika z faktu, że przyjmując $a_{ik} = \alpha_i$, $c_{ik} = \alpha_i$, $k = 1,2$, $i \in I$ zadanie (10) jest równoważne problemowi podziału polegającemu na rozwiązaniu zadania

$$\min_{I \subset J} \max \left\{ \sum_{i \in I} \alpha_i, \sum_{i \in \overline{I}} \alpha_i \right\}$$

i z tego, że problem podziału jest problemem NP-zupełnym [6].

Z powyższego lematu wynika, że skonstruowanie algorytmu wielomianowego dla zadania (10) jest mało prawdopodobne. Dlatego też w dalszej części pracy do rozwiązania tego zadania zastosujemy metodę podziału i ograniczeń.

3. Algorytm metody podziału i ograniczeń

W celu określenia algorytmu opartego na metodzie podziału i ograniczeń trzeba podać m. in. funkcję dolnych ograniczeń, funkcję górnych ograniczeń, podział węzła w drzewie rozwiązań i strategię przeglądania drzewa rozwiązań. Przed wprowadzeniem tych pojęć zdefiniujemy łamaną $\mathcal{L}(I', I'')$ dla $I' \subset I'' \subset J$, na podstawie której będzie określona funkcja dolnych ograniczeń. W tym celu dla dowolnych zbiorów $I' \subset I'' \subset J$ zdefiniujemy ciąg j_i, $i = 1,2,\dots,l$, który spełnia $\{j_1, j_2, \dots, j_l\} = I'' \setminus I'$ oraz

$$\frac{c_{j_1}}{a_{j_1}} \geqslant \frac{c_{j_2}}{a_{j_2}} \geqslant \dots \geqslant \frac{c_{j_l}}{a_{j_l}}, \tag{11}$$

gdzie $l = \overline{\overline{I'' \setminus I'}}$, $a_i \triangleq \min\{a_{i1}, a_{i2}\}$,

$c_i \triangleq \min\{c_{i1}, c_{i2}\}$, $i \in J$.

Określmy teraz $2(l+1)$ punktów $GP(k)$, $DP(k)$, $k = 0,1,2,\dots,l$

$$\begin{aligned} GP(k) &\triangleq \Big(x(I') + \sum_{i=1}^{k} a_{j_i}, y(I') + \sum_{i=1}^{k} c_{j_i}\Big), \\ DP(k) &\triangleq \Big(x(I') + \sum_{i=1}^{k} a_{j_{l-i+1}}, y(I') + \sum_{i=1}^{k} c_{j_{l-i+1}}\Big), \end{aligned} \tag{12}$$

gdzie

$$\begin{aligned} x(I') &= \sum_{i \in I'} a_i - \tfrac{1}{2}\Big(\sum_{i \in J \setminus I''} a_{i2} - \sum_{i \in I'} a_{i1} + \sum_{i \in I'' \setminus I'} a_i\Big), \\ y(I') &= \sum_{i \in I'} c_i - \tfrac{1}{2}\Big(\sum_{i \in J \setminus I''} c_{i2} - \sum_{i \in I'} c_{i1} + \sum_{i \in I'' \setminus I'} c_i\Big). \end{aligned} \tag{13}$$

Zachodzi $GP(0) = DP(0)$ oraz $GP(l) = DP(l)$. Przez $G(k)$ oznaczmy odcinek łączący punkty $GP(k-1)$, $GP(k)$ a przez $D(k)$ odcinek łączący punkty $DP(k-1)$,

$GP(k-1)$; $k = 1,2,\ldots,l$. Ostatecznie wspomniana wyżej łamana jest zdefiniowana następująco

$$Ł(I',I'') \triangleq \left(\bigcup_{k=1}^{l} G(k)\right) \cup \left(\bigcup_{k=1}^{l} D(k)\right) \tag{14}$$

Niech

$$\varphi(A) \triangleq \frac{1}{2}\left[\sum_{i\in I'}(a_{i1}+c_{i1}) + \sum_{i\in J\setminus I''}(a_{i2}+c_{i2}) + \sum_{i\in I''\setminus I'}(a_i+c_i) + \right.$$

$$\left. + \sqrt{\left(\sum_{i\in I'}c_{i1} + \sum_{i\in J\setminus I''}c_{i2} + \sum_{i\in I''\setminus I'}c_i\right)^2 + 4\min_{(x,y)\in Ł(\emptyset, I''\setminus I')}(x^2+2xy)}\,\right],$$

$$A \subset 2^J, \tag{15}$$

gdzie $A = \{A_1, A_2, \ldots, A_{\bar{\bar{A}}}\}$, $I' = \bigcap_{i=1}^{\bar{\bar{A}}} A_i$, $I'' = \bigcup_{i=1}^{\bar{\bar{A}}} A_i$.

Lemat 2

Dla funkcji φ zdefiniowanej w (15) zachodzi

/i/ $\varphi(A) \leqslant \min_{I\in A} Q(I)$, $A \subset 2^J$,

/ii/ $\varphi(B) \geqslant \varphi(A)$, $B \subset A \subset 2^J$,

/iii/ $\varphi(I) = Q(I)$.

Dowód lematu można przeprowadzić korzystając z definicji wprowadzonych pojęć i prostych, ale dość uciążliwych, przekształceń algebraicznych. Z Lematu 2 wynika, że wprowadzona funkcja φ może być użyta jako funkcja dolnych ograniczeń w naszym algorytmie.

Przejdziemy teraz do omówienia drzewa rozwiązań i sposobu podziału węzłów w drzewie rozwiązań. Węzeł i-ty scharakteryzowany jest przez zbiór I_i. Węzłowi temu odpowiada rodzina $R(I_i)$ podzbiorów zbioru operacji J;

$$R(I_i) = \left\{I \subset J : I_i \subset I \subset I_i \cup \{j \in J : j > \max I_i\}\right\}.$$

Z węzłem tym związane jest więc następujące zadanie

$$\min_{I\in R(I_i)} Q(I). \tag{16}$$

Węzeł i-ty dzielimy na l_i = moc zbioru $\{j \in J : j > \max I_i\}$ węzłów o numerach $w_i+1, w_i+2, \ldots, w_i+l_i$, gdzie w_i jest liczbą już istniejących węzłów drzewa. Otrzymane węzły są scharakteryzowane zbiorami $I_{w_i+j} = I_i \cup \{n-l_i+j\}$, $j = 1,2,\ldots,l_i$. Łatwo można pokazać, że

$$R(I_{w_i+r}) \cap R(I_{w_i+s}) = \emptyset, \quad 1 \leqslant r < s \leqslant l_i \tag{17}$$

$$\bigcup_{j=1}^{l} R(I_{w_i+j}) = R(I) \setminus \{I_i\} \tag{18}$$

Warunek (17) oznacza, że rodziny podzbiorów odpowiadające węzłom otrzymanym z podziału węzła i-tego są rozłączne, natomiast warunek (18) mówi, że suma rodzin podzbiorów odpowiadających węzłom otrzymanym z podziału węzła i-tego jest równa rodzinie podzbiorów odpowiadającej węzłowi i-temu z dokładnością do zbioru I_i.

Jako węzeł początkowy w drzewie rozwiązań (korzeń drzewa rozwiązań) przyjmujemy węzeł scharakteryzowany przez zbiór $I_1 = \emptyset$; stąd rodzina podzbiorów odpowiadających temu węzłowi jest równa 2^J. Do podziału będziemy wybierać węzeł o najmniejszej wartości funkcji dolnych ograniczeń.

Na koniec sformułujemy funkcję górnych ograniczeń $REC(I_i)$ dla węzła I_i. Przyjmujemy, że $REC(I_i)$ jest równe minimalnej wartości $Q(I)$ dla $I = I_i$, $I_i \cup \{j_1\}, \ldots, I_i \cup \{j_1, j_2, \ldots, j_l\}$, $I_i \cup \{j_l\}, \ldots, I_i \cup \{j_l, j_{l-1}, \ldots, j_2\}$, gdzie ciąg $j_1, j_2, \ldots, j_l$ jest określony zgodnie z (11) dla $I' = I_i$ oraz $I'' = I_i \cup \{j \in J : j > \max I_i\}$. Ponadto przez $I^*(I_i)$ oznaczmy ten ze zbiorów I, dla którego $Q(I)$ osiąga wartość minimalną.

Wykorzystując wprowadzone powyżej pojęcia sformułujemy teraz algorytm oparty na metodzie podziału i ograniczeń, rozwiązujący zadanie (10):

<u>krok 0</u>

Przyjmujemy, że $I_1 = \emptyset$, wyliczamy $\varphi(R(I_1))$, $REC(I_1)$, $I^*(I_1)$ i kładziemy $\varphi_1 = \varphi(R(I_1))$, $REC = REC(I_1)$, $I^* = I^*(I_1)$. Jeżeli $\varphi_1 \geqslant REC$ to STOP. Następnie kładziemy $i = 1$, $ZA = \{1\}$, $w = 1$ i przechodzimy do kroku 1.

<u>krok 1</u>

Dzielimy i-ty węzeł na l_i węzłów o numerach $w+1, w+2, \ldots, w+l_i$ wg opisanego sposobu, kładziemy $j=1$ i przechodzimy do kroku 2.

<u>krok 2</u>

Wyliczamy $\varphi(R(I_{w+j}))$, $REC(I_{w+j})$, $I^*(I_{w+j})$ i kładziemy $\varphi_{w+j} = \varphi(R(I_{w+j}))$. Jeżeli $REC(I_{w+j}) < REC$, to $REC = REC(I_{w+j})$, $I^* = I^*(I_{w+j})$. Przyjmujemy, że $j = j+1$. Jeżeli $j \leqslant l_i$, to przechodzimy do początku kroku 2. Po wykonaniu powyższych czynności kładziemy $ZA = (ZA \setminus \{i\}) \cup \{w+1, \ldots, w+l_i\}$, $k=1$, przyjmujemy, że $ZA = \{j_1, j_2, \ldots, j_s\}$, $j_1 < j_2 < \ldots < j_s$, $s = \overline{\overline{ZA}}$ i przechodzimy do kroku 3.

<u>krok 3</u>

Jeżeli $\varphi_{j_k} \geqslant REC$, to $ZA = ZA \setminus \{j_k\}$. Przyjmujemy $k = k+1$. Jeżeli $k \leqslant s$, to przechodzimy do początku kroku 3. Po wykonaniu powyższych czynności sprawdzamy, czy $ZA = \emptyset$. Jeżeli tak, to STOP. Następnie przechodzimy do kroku 4.

<u>krok 4</u>

Wyznaczamy $j^* \in ZA$ takie, że $\varphi_{j^*} = \min\limits_{j \in ZA} \varphi_j$ i kładziemy $w = w+l_i$, $i=j^*$. Przechodzimy do kroku 1.

Gdy algorytm zakończy swoje działanie, I^* jest rozwiązaniem zadania (10), a REC odpowiadającą mu minimalną wartością kryterium. Zbieżność algorytmu wynika bezpośrednio z Lematu 2, własności (18) i uwzględnienia zbioru I_1 w definicji funkcji górnych ograniczeń.

4. Przykłady testujące i wyniki obliczeń algorytmu

Przykłady testujące otrzymano przez losowe generowanie danych. Dla określonej liczby n zadań wygenerowano 15 zestawów $a_{i1}, a_{i2}, c_{i1}, c_{i2}$, $i \in J$. Dla n = 15,30,40,...,100 liczby $a_{i1}, a_{i2}, c_{i1}, c_{i2}$ zostały wylosowane ze zbioru $\{0, 0.1, \ldots, 9.9, 10\}$ przez generator o jednostajnym rozkładzie prawdopodobieństwa. Wyniki przedstawiono w TAB. 1.

Tablica 1

Wyniki badań numerycznych algorytmu dla dwóch różnych procesorów

n	S	M	L_1	L_2	BS
	sek	sek		sek	%
15	8,2	8	11	0,15	0,02
30	35,0	33	9	0,20	0,03
40	97,3	98	11	0,68	0,03
50	122,4	120	8	1,51	0,02
60	196,1	206	12	2,32	0,01
70	312,6	308	10	3,41	0,02
80	414,7	422	9	5,37	0,02
90	601,3	604	12	7,08	0,03
100	847,2	844	10	9,25	0,02

W kolumnie S znajduje się średni czas obliczeń w sek na maszynie cyfrowej ODRA 1325 dla 15 zestawów danych przy odpowiednim n, a w kolumnie M - mediana tych czasów. Kolumna L_1 podaje liczbę zestawów danych, dla których znaleziono optymalne rozwiązanie w pierwszym wierzchołku drzewa, a kolumna L_2 - średni czas obliczeń w sek związany z pierwszym wierzchołkiem drzewa.

W przypadku, kiedy głównie zależy nam na krótkim czasie obliczeń możemy ograniczyć się tylko do pierwszego wierzchołka drzewa. Otrzymujemy wtedy rozwiązanie suboptymalne. Z analizy TAB. 1 wynika, że czas obliczeń jest stosunkowo krótki (rzędu kilku sek) oraz istnieje duże prawdopodobieństwo (można przyjąć, że $L_1/15$ jest pewnym oszacowaniem tego prawdopodobieństwa), że znajdziemy rozwiązanie optymalne w pierwszym wierzchołku drzewa rozwiązań.

Przejdziemy teraz do omówienia kolumny BS, tzn. do oszacowania błędu popełnionego przez uciąglenie zasobu nieodnawialnego. Miarą tego błędu jest wielkość $\frac{\tilde{Q} - Q^*}{Q^*}$, gdzie Q^* jest minimalną wartością kryterium zadania (2),(3), a $\tilde{Q}$ wartością kryterium określonego przez (2) dla u_k równych najbliższej liczbie całkowitej liczby u_k^* i dla $I_k = I_k^*$, $k = 1,2$, gdzie u_k^*, I_k^*, $k = 1,2$ jest otrzymanym rozwiązaniem zadania (4),(5). Ponieważ zachodzi $Q^* \geqslant Q(I^*)$ to

$$\frac{\tilde{Q} - Q^*}{Q^*} \cdot 100\% \leqslant \frac{\tilde{Q} - Q(I^*)}{Q(I^*)} \cdot 100\% \triangleq B$$

i wielkość B, którą możemy wyznaczyć, jest górnym oszacowaniem rozpatrywanego błędu. W kolumnie BS zamieszczono wartość średnią wielkości B dla 15- zestawów danych przy odpowiednim n. Z analizy tej kolumny wynika, że przyjęta metoda postępowania polegająca na uciągleniu zasobu nieodnawialnego prowadzi do bardzo małych błędów względnych.

Reasumując, wyniki eksperymentów obliczeniowych upoważniają do stwierdzenia, że przedstawiona metoda rozwiązywania zadania czasowo-optymalnego przydziału zasobow i zadań do dwóch różnych maszyn może być z powodzeniem zastosowana do problemów, w których liczba zadań jest rzędu 100. W związku z tym można ją wykorzystać w praktycznie występujących dyskretnych systemach produkcyjnych tego typu.

LITERATURA

[1] Błażewicz J., Cellary W., Słowiński R., Węglarz J.: Algorytmy sterowania rozdziałem zadań i zasobów w kompleksie operacji, Wydawnictwo Polit. Poznańskiej, Poznań 1979.

[2] Bubnicki Z.: Optymalizacja kompleksów operacji w sterowaniu dyskretnymi procesami produkcyjnymi, Prace VII KKA, tom III, Rzeszów 1977.

[3] Buchalski Z.: Czasowo-optymalny rozdział zasobów w dwuprocesorowym systemie komputerowym, Prace Krajowego Sympozjum Cybernetyki, Warszawa 1983 /w druku/.

[4] Buchalski Z.: Zastosowanie teorii kompleksów operacji do czasowo-optymalnego rozdziału pamięci operacyjnej w systemach komputerowych, Prace III krajowej konferencji pn. "Zastosowanie komputerów w przemyśle", Szczecin 1983.

[5] Burkov V.N.: Optimalnoje upravlenije kompleksami opieracji, Prace IV Kongresu IFAC, z. 35, Warszawa 1969.

[6] Coffman E.G.Jr.: Teoria szeregowania zadań, WNT, Warszawa 1980.

[7] Czechowicz K.: Algorytmy czasowo-optymalnego sterowania kompleksem operacji niezależnych, Praca doktorska, Komunikat Nr 112, ICT Polit. Wrocławskiej, Wrocław 1974.

[8] Węglarz J.: Sterowanie w systemach typu kompleks operacji, PWN, Warszawa-Poznań 1981.

Recenzent: Prof.dr hab.inż.Jan Węglarz

Wpłynęło do Redakcji do 30.03.1984r.

НЕКОТОРАЯ ПРОБЛЕМА РАСПРЕДЕЛЕНИЯ ЗАДАЧ И НЕВОССТАНАВЛИВАЕМОГО РЕСУРСА ДЛЯ ДВУХ РАЗНЫХ МАШИН В ДИСКРЕТНОЙ ПРОИЗВОДСТВЕННОЙ СИСТЕМЕ

Р е з ю м е

В работе формулируется задача временно-оптимального распределения невосстанавливаемого ресурса и заданий для двух разных машин. Показано, что проблема является NP - трудной и конструируется алгоритм решающий эту задачу. Алгоритм основан на методе ветвей и границ. Даны результаты вычислительных экспериментов приведённых для этого алгоритма.

SOME PROBLEMS OF TASKS AND NONRENEWABLE RESOURCES ALLOCATION IN TWO MACHINES DISCRETE PRODUCTION SYSTEM

S u m m a r y

In the paper the problem of time optimal allocation of n tasks and nonrenewable, discretely divisible resources to two machines is considered. It is shown that the problem is of NP-type. An algorithm to solve the problem on on the basis of branch and bound method is designed. The results of the numerical experiments for n = 15,30,40,...,100 are enclosed.