ZESZYTY NAUKOWE POLITECHNIKI ŚLĄSKIEJ 1986
Seria: AUTOMATYKA z. 84 Nr kol. 894

Adam Janiak
Politechnika Wrocławska

PROBLEMY SZEREGOWANIA ZADAŃ NA JEDNEJ MASZYNIE Z ROZDZIAŁEM ZASOBÓW

Streszczenie. W pracy dokonano przeglądu klasycznych jednomaszynowych problemów szeregowania zadań przy kryteriach: maksymalny koszt wykonywania zadań, maksymalny czas zakończenia zadań, maksymalna nieterminowość wykonywania zadań, maksymalne opóźnienie wykonania zadań, ważona suma czasów zakończenia zadań, ważona suma opóźnień wykonania zadań. Problemy te uogólniono na przypadek, gdy czasy trwania zadań zależą liniowo od ilości lokalnie i globalnie ograniczonych zasobów. Dla większości z tych problemów znaleziono efektywne /wielomianowe/ algorytmy sterowania. Wykazano, że pewne problemy są NP-trudne.

1. Wstęp. Przegląd literatury.

W wielu gałęziach przemysłu, w szczególności w tych gałęziach, w których mamy do czynienia z dyskretnymi procesami przemysłowymi, np. w przemyśle metalurgicznym, maszynowym, samochodowym, pojawiają się problemy szeregowania operacji /zadań/ na maszynach i równoczesnego rozdziału ograniczonych zasobów różnego rodzaju. W wielu takich produkcyjnych procesach często występuje tylko jedna krytyczna maszyna /np. z dużym kosztem zakupu lub eksploatacji/, na której problem szeregowania zadań /operacji/ i rozdziału zasobów jest rzeczywiście istotnym problemem. Natomiast z punktu widzenia rozpatrywanych kryteriów optymalności, problemy szeregowania na pozostałych maszynach są praktycznie nieistotne i można w przybliżeniu rozpatrywać te maszyny jako maszyny o nieograniczonej przepustowości. Zatem, w wielu praktycznych przypadkach można ograniczyć się do problemu szeregowania operacji i rozdziału zasobów tylko na jednej krytycznej maszynie, a czasy obróbki rozpatrywanych elementów na wcześniejszych maszynach mogą być uwzględnione w tzw. najwcześniejszych momentach dostępności do obróbki danych elementów na maszynie krytycznej.

Niniejsza praca dotyczy właśnie wspomnianych powyżej jednomaszynowych problemów szeregowania. Przy czym zakłada się, że czasy trwania operacji zależą liniowo od lokalnie i globalnie ograniczonych, nieodnawialnych, podzielnych w sposób ciągły zasobów, np. energia, tlen, surowce, paliwo /pieniądze/. Z każdą operacją związany jest najwcześniejszy moment rozpoczęcia wykonywania tej operacji oraz żądany termin jej zakończenia. Uwzględniono także możliwość występowania pewnych technologicznych uwarunkowań kolejnościowych między operacjami. Rozpatrzono następujące kryteria

optymalności:
- maksymalny koszt wykonywania operacji,
- maksymalny czas zakończenia operacji,
- maksymalna nieterminowość wykonania operacji,
- maksymalne opóźnienie wykonania zadań,
- ważona suma czasów zakończenia operacji,
- ważona suma opóźnień wykonywania operacji.

W pracy starano się znaleźć jak najszerszą klasę jednomaszynowych problemów szeregowania zadań z równoczesnym rozdziałem zasobów, dla których istnieją wielomianowe algorytmy sterowania.

Rozpatrywanie jednomaszynowych problemów szeregowania zadań z rozdziałem zasobów, istotne jest także z teoretycznego punktu widzenia, ponieważ problemy te pojawiają się jako problemy wyznaczania dolnych ograniczeń w wielomaszynowych problemach szeregowania zadań i rozdziału zasobów.

Jednomaszynowe problemy szeregowania zadań przy założeniu, że czasy trwania operacji są z góry zadane, mają bogatą literaturę. Dla dość licznej klasy tych problemów znaleziono efektywne /wielomianowe/ algorytmy /definicja np. w [7]/. Czas obliczeń /lub inaczej liczba obliczeniowych kroków/ takich algorytmów jest ograniczona wielomianem $w(n)$, gdzie n jest - w tym wypadku - liczbą szeregowanych zadań. W dalszej części pracy będziemy mówili, że złożoność obliczeniowa takich algorytmów /problemów/ jest $O(w(n))$. Natomiast pewna część rozpatrywanych w literaturze problemów jednomaszynowych należy do klasy problemów NP-trudnych [7], dokładniej decyzyjna wersja tych optymalizacyjnych problemów jednomaszynowych jest NP-zupełna [7]. To znaczy, są one równoważne dużej liczbie znanych kombinatorycznych problemów [7] w tym sensie, że znalezienie wielomianowego algorytmu dla któregokolwiek problemu z tej klasy spowodowałoby znalezienie podobnych /wielomianowych/ algorytmów dla wszystkich problemów z tej klasy. Przy czym w klasie tej wyróżnia się podklasę problemów silnie NP-zupełnych /ang. unary NP-complete/, problemy te nawet w przypadku kodowania przy podstawie 1 pozostają problemami NP-zupełnymi.

Zwięzły przegląd literatury dotyczącej jednomaszynowych problemów szeregowania z ustalonymi czasami wykonywania operacji, przy wspomnianych powyżej kryteriach optymalności, można przedstawić w sposób następujący.

Dla problemu z kryterium będącym maksymalną nieterminowością wykonywania operacji, przy założeniu, że wszystkie terminy dostępności operacji są jednakowe, znaleziony został przez Jacksona /1955/ w [9] trywialny algorytm o złożoności $O(n \cdot \log n)$ /gdzie n jest ilością zadań/. Natomiast przy założeniu, że terminy dostępności zadań są różne, problem ten staje się silnie NP-trudny, co zostało wykazane w [19] /Lenstra, Rinnooy Kan, Brucker 1977/. Do rozwiązania tego problemu zaproponowano algorytmy oparte o metodę podziału i ograniczeń: Baker i Su /1974/ [1]; McMahon i

Florian /1975/ [20]; Lageweg, Lenstra i Rinnooy Kan /1976/ [13] oraz Carlier /1980/ [6]. Natomiast dla przypadku jednostkowych czasów wykonywania operacji Horn /1974/ w [5] zaproponował algorytm o złożoności $O(n \cdot \log n)$.

Problem z kryterium będącym maksymalnym kosztem wykonywania operacji /kosztem związanym z terminem zakończenia operacji/ został rozwiązany przez Lawlera /1973/ w [14] w wielomianowym czasie $O(n^2)$ /przy jednakowych terminach dostępności niepodzielnych operacji/. Algorytm Lawlera został uogólniony przez Bakera Lawlera, Lenstra, Rinnooy Kana /1983/ w [2] na przypadek z różnymi terminami dostępności podzielnych operacji $O(n^3)$.

Problem z kryterium będącym ważoną sumą czasów zakończenia operacji, przy braku relacji poprzedzania i przy założeniu równości wszystkich najwcześniejszych momentów dostępności operacji, został rozwiązany w wielomianowym czasie $O(n \cdot \log n)$ przez Smitha /1956/ [24]. Rezultat ten został uogólniony, na przypadek gdy jest zadana relacja poprzedzania wśród operacji przez Horna /1972/ [4], Sidneya /1975/ [23] - dla relacji w postaci drzewa oraz przez Lawlera /1978/ [16] - dla szeregowo równoległych relacji poprzedzania. Dla przypadku dowolnej relacji poprzedzania problem ten staje się NP-trudny nawet dla jednostkowych czasów trwania operacji albo jednostkowych wag /Lawler /1978/ [16], Lenstra i Rinnooy Kan /1978/ [18]. Natomiast dla jednostkowych wag, ale przy założeniu dowolnych najwcześniejszych momentów dostępności operacji, problem ten staje się silnie NP-trudny. Jednakże dla operacji podzielnych ostatni przypadek może być rozwiązany za pomocą prostego rozszerzenia algorytmu Smitha ; natomiast w przypadku różnych wag staje się on silnie NP-trudny /Labetoulle, Lawler, Lenstra, Rinnooy Kan /1979/ [12]/.

Jednomaszynowy problem szeregowania operacji przy kryterium ważona suma opóźnień wykonania operacji należy do klasy problemów silnie NP-trudnych /Lawler /1977/ [15], Lenstra, Rinnooy Kan, Brucker /1977/ [19]/. Dla przypadku z jednostkowymi wagami znaleziony został przez Lawlera /1977/ [15] pseudowielomianowy algorytm /definicja w [7]/ o złożoności $O(n^4 \cdot \sum_{j=1}^{n} p_j)$ /gdzie p_j jest czasem trwania j-tej operacji/. Jednakże złożoność obliczeniowa wspomnianego przypadku pozostaje nadal otwartym problemem /wobec znalezienia dla niego pseudowielomianowego algorytmu wiadomo tylko, że problem ten nie jest silnie NP-trudny/.

Problemy szeregowania zadań na maszynach z równoczesnym rozdziałem zasobów były także rozpatrywane w literaturze, i tak: problem gniazdowy i taśmowy przy założeniu, że czasy trwania prac zależą liniowo od podzielnych w sposób ciągły, ograniczonych zasobów były rozpatrywane przez Grabowskiego i Janiaka np. /1980/ w [10,8]. Problem szeregowania podzielnych operacji przy założeniu różniczkowych modeli operacji dla przypadku iden-

tycznych równoległych maszyn był rozpatrywany przez Węglarza /1979/ w [30]. Problemy szeregowania zadań przy założeniu dyskretnych żądań zasobowych /tzn. z pewnego skończonego zbioru możliwych przydziałów zasobów/ były rozpatrywane przez Błażewicza i współautorów, np. w [3] oraz przy podejściu wielokryterialnym przez Słowińskiego, np. w [25].

Jednomaszynowy problem szeregowania operacji ze sterowalnymi czasami wykonywania operacji i liniową zależnością kosztu wykonania operacji od jej czasu zakończenia przy kryterium będącym sumą kosztu związanego z czasami zakończenia wykonywania operacji i kosztu wykonania wszystkich operacji był rozpatrywany przez Vicksona /1980/ w [28]. Natomiast problem ten w przypadku, gdy pierwszy składnik wspomnianego kryterium był kosztem proporcjonalnym do maksymalnego opóźnienia operacji, był rozpatrywany w [29] /Vickson 1980/. Następnie Nowicki i Zdrzałka /1985/ w [22] rozszerzyli badania z [28] na przypadek dwumaszynowy.

Jednomaszynowy problem szeregowania operacji ze sterowalnymi czasami wykonywania operacji przy dwukryterialnym podejściu i przy pewnych istotnych założeniach odnośnie własności funkcji kosztowych był rozpatrywany przez Van Wassenhove i Bakera /1982/ w [27]. Badania te zostały następnie uogólnione przez Tuzikova /1984/ w [26].

Obecnie przejdziemy do precyzyjnego sformułowania problemów rozpatrywanych w pracy.

Dany jest zbiór zadań $J_1,\ldots,J_j,\ldots,J_n$, które mają być wykonane przy użyciu jednej maszyny, stanowiącej gniazdo krytyczne. Zakłada się, że w danej chwili czasu maszyna może wykonywać tylko jedno zadanie i, że zadania są niepodzielne, tzn. jeśli maszyna zacznie realizować dane zadanie to musi je skończyć bez przerwań. Dalej będziemy zakładali, że modele zadań /operacji - każde zadanie składa się tylko z jednej operacji/ są następujące:

$$p_j \triangleq p_j(u_j) \triangleq b_j - a_j u_j , \qquad /1/$$

gdzie: p_j jest czasem wykonywania zadania J_j, u_j - ilością zasobów przydzielonych do wykonania zadania J_j, natomiast $b_j > 0$, $a_j > 0$ są znanymi parametrami. Ponadto zakłada się, że ilość zasobów u_j przydzielona do zadania J_j spełnia następujące ograniczenia:

$$\alpha_j \le u_j \le \beta_j , \quad j=1,2,\ldots,n, \qquad /2/$$

$$\sum_{j=1}^{n} u_j \le \hat{U} , \qquad /3/$$

gdzie: $\hat{U}$ jest globalną dysponowaną do rozdziału ilością zasobów, α_j, β_j są zadanymi technologicznymi ograniczeniami /parametrami/, zwykle zakłada się, że $\alpha_j = 0$ /$\beta_j \le b_j/a_j$/.

Niech U oznacza zbiór wszystkich dopuszczalnych /tzn. spełniających ogra-

niczenia /2/ i /3// rozdziałów zasobów $u = [u_1,\dots,u_j,\dots,u_n]$.
Dla każdego zadania J_j może być zadany najwcześniejszy możliwy czas dostępności tego zadania do wykonania r_j /np. będzie to suma czasów trwania tego zadania na maszynach poprzedzających rozpatrywaną maszynę krytyczną/ oraz może być zadany pożądany termin zakończenia tego zadania d_j.
Pomiędzy zadaniami mogą być zadane technologiczne uwarunkowania kolejnościowe. Zwykle są one reprezentowane za pomocą następujących zbiorów:

$$A_j \triangleq \{ i \mid J_j < J_i \}, \quad j=1,2,\dots,n, \qquad /4/$$

lub równoważnie za pomocą zbiorów

$$B_j \triangleq \{ i \mid J_i < J_j \}, \quad j=1,2,\dots,n, \qquad /5/$$

gdzie: $J_j < J_i$ oznacza, że wykonywanie zadania J_i nie może rozpocząć się zanim nie zostanie zakończone zadanie J_j.

Niech $\pi = [\pi(1),\dots, \pi(j),\dots, \pi(n)]$ będzie dowolną permutacją indeksów zadań, tzn. liczb $\{1,\dots,j,\dots,n\}$. Każda permutacja π określa kolejność wykonywania zadań $\langle J_{\pi(1)},\dots,J_{\pi(j)},\dots,J_{\pi(n)}\rangle$ na rozpatrywanej maszynie. Niech Π będzie zbiorem wszystkich permutacji spełniających zadane ograniczenia kolejnościowe /4/ /lub /5//.

Dla zadanej dopuszczalnej permutacji zadań $\pi \in \Pi$ oraz zadanego dopuszczalnego rozdziału zasobów $u \in U$ można wyznaczyć dla każdego zadania J_j:
- czas zakończenia wykonywania $C_j(\pi,u)$,
- nieterminowość wykonania $L_j(\pi,u) = C_j(\pi,u) - d_j$,
- opóźnienie $T_j(\pi,u) = \max(0, L_j(\pi,u))$.

Permutacja zadań $\pi \in \Pi$ wraz z rozdziałem zasobów $u \in U$ może być oceniana względem różnych kryteriów; w prezentowanej pracy będą minimalizowane następujące kryteria:
- maksymalny czas zakończenia wykonywania zadań:

$$C_{max} \triangleq C_{max}(\pi,u) \triangleq \max_{1\le j\le n} \{ C_j(\pi,u) \}$$ - rozdział 2.1;

- maksymalna nieterminowość wykonania zadań:

$$L_{max} \triangleq L_{max}(\pi,u) \triangleq \max_{1\le j\le n} \{ L_j(\pi,u) \}$$ - rozdział 2.2;

- maksymalne opóźnienie wykonywania zadań:

$$T_{max} \triangleq T_{max}(\pi,u) \triangleq \max_{1\le j\le n} \{ T_j(\pi,u) \}$$ - rozdział 2.3;

- maksymalny koszt wykonania zadań:

$$f_{max} \triangleq f_{max}(\pi,u) \triangleq \max_{1\le j\le n} \{ f_j(C_j(\pi,u)) \},$$

gdzie: $f_j(t)$ jest pewną monotonicznie niemalejącą funkcją względem czasu t - rozdział 2.4;

- ważona suma czasów zakończenia wykonywania zadań:

$\sum w_j C_j \triangleq \sum_{j=1}^{n} w_j C_j(\pi, u)$, gdzie $w_j > 0$ jest pewną wagą przypisaną do zadania J_j - rozdział 2.5;

- ważona suma opóźnień wykonywania zadań:

$\sum w_j T_j \triangleq \sum_{j=1}^{n} w_j T_j(\pi, u)$ - rozdział 2.6;

- całkowity koszt wykonania zadań:

$\sum f_j \triangleq \sum_{j=1}^{n} f_j\left(C_j(\pi, u)\right)$ - rozdział 2.6.

W dalszej części pracy będzie stosowany powszechnie używany schemat krótkiej notacji problemów szeregowania [17] - $\alpha|\beta|\gamma$, gdzie α charakteryzuje założenie dotyczące maszyn, dla jednomaszynowych problemów szeregowania przyjmuje się, że $\alpha = 1$;

$\beta \subset \{prec, r_j, p_j(u_j)\}$ - charakteryzuje założenia dotyczące zadań, przy czym:

prec /ang. precedence constraints/ - oznacza, że są zadane ograniczenia kolejnościowe pomiędzy zadaniami /brak "prec" oznacza brak tych ograniczeń/;

r_j - oznacza, że najwcześniejsze momenty dostępności zadań nie są jednakowe /brak "r_j" - oznacza, że wszystkie r_j są sobie równe/;

$p_j(u_j)$ - oznacza, że czasy wykonywania zadań liniowo zależą od ilości przydzielonych im zasobów /brak $p_j(u_j)$ - oznacza, że czasy wykonywania zadań są zadane /nie zależą od zasobów//;

natomiast γ precyzuje kryterium optymalności.

2. Jednomaszynowe problemy szeregowania zadań i rozdziału zasobów

2.1. Problem $1|prec, r_j, p_j(u_j)|C_{max}$

Można łatwo wykazać następującą własność:

Własność 2.1.

Problem $1|prec, r_j, p_j(u_j)|C_{max}$ można rozwiązać w $O(n^2)$ krokach szeregując zadania zgodnie z regułą Jackson'a [9] - $\pi^* \in \Pi$ /tzn. zgodnie z rosnącymi r_j przy równoczesnym uwzględnieniu zadanych ograniczeń kolejnościowych/ oraz rozdzielając zasoby zgodnie z następującą procedurą:

Procedura 2.1.

Krok 1. Podstaw $u_j^{\pi} := 0$, $j=1,2,\ldots,n$, oraz $l := 1$. Znajdź momenty rozpoczęcia wykonywania zadań używając następujących rekurencyjnych wzorów:

$S_{\pi^*(1)} := r_{\pi^*(1)}$, $S_{\pi^*(j)} := \max\left[r_{\pi^*(j)}, S_{\pi^*(j-1)} + b_{\pi^*(j-1)}\right]$,

$j=2,3,\ldots,n$ i przejdź do Kroku 2;

Krok 2. Znajdź największy indeks zadania k, $1 \le k \le n$, który spełnia następującą równość: $r_{\pi^*(k)} = S_{\pi^*(k)}$. Następnie znajdź zbiór: $P := \{\pi^*(j) \mid k \le j \le n,\ u^*_{\pi^*(j)} < \beta_{\pi^*(j)}\}$. Jeśli zbiór P jest pusty lub $\hat{U} = 0$, wówczas Stop - u^*_j, $j=1,2,\ldots,n$, jest optymalnym rozdziałem zasobów, w przeciwnym wypadku przejdź do Kroku 3.

Krok 3. Znajdź indeks zadania t, dla którego $a_{\pi^*(t)} = \max_{\pi^*(j)\in P} a_{\pi^*(j)}$ i następnie podstaw $d := \min_{t<i\le n} \left(S_{\pi^*(i)} - r_{\pi^*(i)}\right)$,

$z := \min\{\beta_{\pi^*(t)}, \hat{U}, d/a_{\pi^*(t)}\}$. Następnie uaktualnij rozdział zasobów i momenty rozpoczęcia wykonywania zadań przez podstawienie: $u^*_{\pi^*(t)} := u^*_{\pi^*(t)} + z$, $\hat{U} := \hat{U} - z$,

$S_{\pi^*(i)} := S_{\pi^*(i)} - z \cdot a_{\pi^*(t)}$ dla $t < i \le n$.

Ostatecznie podstaw $l := k$ i przejdź do Kroku 2.

2.2. Problemy: $1|prec,p_j(u_j)|L_{max}$, $1|r_j,p_j(u_j)|L_{max}$

Analogicznie jak w klasycznym przypadku /tzn. bez rozdziału zasobów [17]/ problem $1|prec,p_j(u_j)|L_{max}$ można potraktować jako "odwrotną wersję" problemu $1|prec,r_j,p_j(u_j)|C_{max}$. Zatem problem $1|prec,p_j(u_j)|L_{max}$ można rozwiązać w $O(n^2)$ krokach przez uszeregowanie zadań zgodnie z rosnącymi żądanymi terminami zakończenia wykonywania zadań, przy równoczesnym uwzględnieniu zadanych ograniczeń kolejnościowych, oraz przez zastosowanie procedury generującej optymalny rozdział zasobów analogicznej do Procedury 2.1.

Następująca własność wynika z faktu, że klasyczny NP-trudny problem $1|r_j|L_{max}$ [17] jest szczególnym przypadkiem problemu $1|r_j,p_j(u_j)|L_{max}$:

Własność 2.2.

Problem $1|r_j,p_j(u_j)|L_{max}$ jest NP-trudny.

Do rozwiązania problemu $1|r_j,p_j(u_j)|L_{max}$ można zastosować podobny algorytm do algorytmu, który zastosowano do rozwiązania ogólnego problemu gniazdowego z rozdziałem zasobów. Algorytm ten w pierwotnej wersji został zaprezentowany w [8], a następnie znacznie udoskonalony w [11]. Udoskonalony algorytm był w stanie rozwiązywać przykłady problemów gniazdowych z zasobami, konstruowanych na bazie znanego /nierozwiązanego do tej pory/ problemu o 10 zadaniach wykonywanych na 10 maszynach /100 operacji/ /sformułowanego w [21]/.

2.3. Problem $1|prec,p_j(u_j)|T_{max}$

Łatwo można wykazać następującą własność:

Własność 2.3.

Optymalne rozwiązanie /tzn. permutacja i rozdział zasobów/ problemu $1|prec,p_j(u_j)|L_{max}$ jest jednocześnie optymalnym rozwiązaniem problemu $1|prec,p_j(u_j)|T_{max}$.

Oba problemy, jednakże, nie są równoważne, tzn. odwrotna własność nie zawsze jest prawdziwa.

Analogicznie do Własności 2.2 może być wykazana następująca własność:

Własność 2.3'.

Problem $1|r_j,p_j(u_j)|T_{max}$ jest NP-trudny.

Problem $1|r_j,p_j(u_j)|T_{max}$ można rozwiązać bądź przez rozwiązanie odpowiadającego mu problemu $1|r_j,p_j(u_j)|L_{max}$, bądź przez skonstruowanie algorytmu w sposób podobny jak dla problemu $1|r_j,p_j(u_j)|L_{max}$.

2.4. Problem $1|prec,p_j(u_j)|f_{max}$

Można wykazać następującą własność:

Własność 2.4.

Jeśli dla wszystkich modeli zadań $a_j := a$, $j=1,2,\ldots,n$, wówczas następująca procedura generuje w $O(n^2)$ krokach rozwiązanie optymalne /tzn. $\pi^* \in \Pi$ oraz $u^* \in U$/:

Procedura 2.4:

<u>Krok 1</u>. Podstaw $S := \{1,2,\ldots,n\}$, $k := n$,

$C := \sum_{j=1}^{n} b_j - a \cdot \min\{\sum_{j=1}^{n} \beta_j, \hat{U}\}$ i przejdź do Kroku 2;

<u>Krok 2</u>. Znajdź zbiór $Q := \{h | h \in S, A_h \cap S = \emptyset\}$ oraz znajdź indeks $i \in S$, dla którego $f_i(C) = \min_{h \in Q} f_h(C)$, następnie podstaw: $\pi^*(k) := i$,

$S := S - \{i\}$, $u_i^* := \min\{\beta_i, \max\{0, \hat{U} - \sum_{j \in S} \beta_j\}\}$,

$\hat{U} := \hat{U} - u_i^*$, $C := C - (b_i - a \cdot u_i^*)$; $k := k - 1$

i przejdź do Kroku 3;

<u>Krok 3</u>. Jeśli $S \neq \emptyset$, wówczas przejdź do Kroku 2, w przeciwnym wypadku Stop - π^* i u^* są optymalnym rozwiązaniem.

Złożoność obliczeniowa powyższego problemu dla przypadku z różnymi współczynnikami a_j, $j=1,2,\ldots,n$ pozostaje otwartym problemem.

2.5. Problem $1|p_j(u_j)|\sum w_j C_j$

W problemie $1|p_j(u_j)|\sum w_j C_j$ będziemy mówili, że zadanie J_j "dominuje" zadanie J_i, oznaczając $J_i \lessdot J_j$, wówczas gdy $b_i \leq b_j \wedge a_i \geq a_j \wedge \beta_i \geq \beta_j \wedge w_i \geq w_j$.

Łatwo można zweryfikować następującą własność:

Własność 2.5.

Jeśli dla każdych dwóch zadań J_i oraz J_j, $i=1,2,\ldots,n$, $j=1,2,\ldots,n$, $i \neq j$, albo J_j dominuje J_i, tzn. $J_i \lessdot J_j$ albo odwrotnie: J_i dominuje J_j, tzn. $J_j \lessdot J_i$, to wówczas problem $1|p_j(u_j)|\sum w_j C_j$ można rozwiązać w $O(n \cdot \log n)$ krokach szeregując zadania zgodnie z relacją dominacji "$\lessdot$" /otrzymując permutację π^*/ i rozdzielając zasoby $u^* \in U$ zgodnie z następującą procedurą:

Procedura 2.5:

Krok 1. Od $i := 1$ do n wykonaj Krok 2.

Krok 2. Niech $u^*_{\pi^*(i)} := \min\{\beta_{\pi^*(i)}, \hat{U}\}$, $\hat{U} := \hat{U} - u^*_{\pi^*(i)}$.

Złożoność obliczeniowa rozpatrywanego problemu dla przypadku, gdy nie zachodzi relacja dominacji dla każdych dwóch zadań, pozostaje otwartą kwestią.

2.6. Problemy $1|p_j(u_j)|\sum w_j T_j$, $1|p_j(u_j)|\sum f_j$, $1|r_j,p_j(u_j)|\sum C_j$, $1|r_j,p_j(u_j)|\sum T_j$, $1|prec,p_j(u_j)|\sum C_j$ oraz $1|prec,p_j(u_j)|\sum T_j$.

Podobnie jak Własność 2.2 może być wykazana następująca własność:

Własność 2.6.

Następujące problemy są NP-trudne:

$1|p_j(u_j)|\sum w_j T_j$,

$1|p_j(u_j)|\sum f_j$,

$1|r_j,p_j(u_j)|\sum C_j$,

$1|r_j,p_j(u_j)|\sum T_j$,

$1|prec,p_j(u_j)|\sum C_j$,

$1|prec,p_j(u_j)|\sum T_j$.

Rozwiązanie powyższych problemów wymaga opracowania oddzielnych algorytmów optymalnych lub efektywnych /z dobrymi oszacowaniami/ algorytmów przybliżonych, które w wielu przypadkach mogą być z powodzeniem użyte, zamiast często dość czasochłonnych algorytmów dokładnych.

3. Zakończenie

Podobne rozważania warto byłoby przeprowadzić dla innych kryteriów optymalności oraz dla problemów dwumaszynowych, w szczególności dla tych, dla których w klasycznej wersji /bez zasobów/ istnieją wielomianowe /efektywne/ algorytmy.

LITERATURA

[1] Baker K.R., Su Z.S.: Sequencing with due-dates and early start times to minimize maximum tardiness. Naval Res. Logist. Quart. 21, 1974, str. 171-176.

[2] Baker K.R., Lawler E.L., Lenstra J.K., Rinnooy Kan A.H.G.: Preemptive scheduling of a single machine to minimize maximum cost subject to release dates and precedence constraints.Oper.Res.31,1983, s.381-386.

[3] Błażewicz J., Lenstra J.K., Rinnooy Kan A.H.G.: Scheduling subject to resource constraints: Classification and Complexity, Discrete Appl. Marth. 5, 1983, str. 11-24.

[4] Horn W.A.: Single-machine job sequencing with treelike precedence ordering and linear delay penalties. SIAM. J.Appl.Math. 23, 1972, str. 189-202.

[5] Horn W.A.: Some simple scheduling algorithms, Naval Res. Logist. Quart. 21, 1974, str. 177-185.

[6] Carlier J., The one-machine sequencing problem. Europ. J. Oper. Res. 11, 1980, str. 42-47.

[7] Garey M.R., Johnson D.S.: Computers and Intractability: a Guide to the Theory of NP-Completeness, Freeman, San Francisco 1979.

[8] Grabowski J., Janiak A., Job-shop scheduling with resource-time models of operations, Europ. J. Oper. Res. 1986 /w druku/.

[9] Jackson J.R.: Scheduling a production line to minimize maximum tardiness. Research Report 43, Management Science Research Project, University of California, Los Angeles, 1955.

[10] Janiak A., Grabowski J.: Optimizacija probliemy posliedovatielnosti opieracij s raspriedielienijem riesursov v diskrietnych proizvodstviennych sistiemach, Prace IBS PAN, 59, 1980, str. 129-138.

[11] Janiak A.: General sequencing problem with transferable resources, Raport ICT Politechniki Wrocławskiej, 12, 1986.

[12] Labetoulle J., Lawler E.L., Lenstra J.K., Rinnooy Kan A.H.G.: Preemptive scheduling of uniform machines subject to release dates. Report BW 99, Mathematisch Centrum, Amsterdam 1979.

[13] Lageweg B.J., Lenstra J.K., Rinnooy Kan A.H.G.: Minimizing maximum lateness on one machine: computational experience and some applications. Statist. Neerlandica, 30, 1976, str. 25-41.

[14] Lawler E.L.: Optimal sequencing of a single machine subject to precedence constraints, Management Sci., 19, 1973, str. 544-546.

[15] Lawler E.L.: A "pseudopolynomial" algorithm for sequencing jobs to minimize total tardiness. Ann. Discrete Math. 1, 1977, str. 331-342.

[16] Lawler E.L.: Sequencing jobs to minimize total weighted completion time subject to precedence constraints. Ann. Discrete Math. 2, 1978, str. 75-90.

[17] Lawler E.L., Lenstra J.K., Rinnooy Kan A.H.G.: Recent developments in deterministic sequencing and scheduling: a survey. w: Dempster M.A.H., Lenstra J.K., Rinnooy Kan A.H.G. /eds./: Deterministic and Stochastic Scheduling, Reidel, Dordrecht, 1982, str. 35-73.

[18] Lenstra J.K., Rinnooy Kan, A.H.G.: Complexity of scheduling under precedence constraints. Oper. Res. 26, 1978, str. 22-35.

[19] Lenstra J.K., Rinnooy Kan A.H.G., Brucker P.: Complexity of machine scheduling problems. Ann.Disc.Math. 1, 1977, str. 343-362.

[20] Mc Mahon G.B., Florian M.: On scheduling with ready times and due dates to minimize maximum lateness. Oper.Res. 23, 1975, str. 475-482.

[21] Muth J.F., Thompson G.L.: Industrial Scheduling, Englewood Cliffs, New Jersey, Prentice-Hall, 1963.

[22] Nowicki E., Zdrzałka S.: Dwumaszynowy problem przepływowy ze zmiennymi czasami wykonywania zadań, Zeszyty Naukowe A.G.H. w Krakowie, Automatyka 39, 1985, str. 161-169.

[23] Sidney J.B.: Decomposition algorithms for single-machine sequencing with precedence relations and deferral costs. Oper. Res. 23, 1975, str. 283-298.

[24] Smith W.E.: Various optimizers for single-stage production. Naval Res. Logist. Quart., 3, 1956, str. 59-66.

[25] Słowiński R.: Multiobjective network scheduling with efficient use of renewable and nonrenewable resources, Europ. J. Oper. Res. 7, 1981, str. 265-273.

[26] Tyzikov A.W.: O dwuchkritierialnoj zadačie tieorii raspisanij s uciotom izmienienija dlitielnostiej obsłużivanija, Żurnał wyczislitielnoj matiematiki i matiematiczieskoj fiziki, 24, 1984, str. 1585-1590.

[27] Van Wassenhove L.N., Baker K.R.: A bicriterion approach to time/cost trade-offs in sequencing, Europ.J.Oper.Res. 1, 1982, str. 48-54.

[28] Vickson R.G.: Choosing the job sequence and processing time to minimize total processing plus flow cost on a single machine. Oper.Res., 28, 1980, str. 1155-1167.

[29] Vickson R.G.: Two single machine sequencing problems involving controllable job processing time. AIII Transaction, 12, 1980, str. 258-262.

[30] Węglarz J.: Project scheduling with discrete and continuous resources, IEEE Trans. SMC-9, 1979, str. 644-650.

Recenzent: Prof.dr inż.Henryk Kowalowski

Wpłynęło do Redakcji do 1986.04.30

ОДНОПРИБОРНЫЕ ПРОБЛЕМЫ РАСПИСАНИЙ С РАСПРЕДЕЛЕНИЕМ РЕСУРСОВ

Р е з ю м е

В работе дан обзор классических одноприборных проблем расписаний со следующими критериями оценки: максимальный штраф , максимальная длительность прохождения , максимальное временное смещение , максимальное запаздывание, суммарная длительность прохождения , суммарное запаздывание. Эти проблемы обобщены для случая, когда длительности выполнения операций линейно зависят от локально и совокупно ограниченных ресурсов. Для большинства этих проблем найдены полиномиальные алгоритмы . Некоторые из этих проблем P – трудные.

SINGLE MACHINE SCHEDULING PROBLEMS WITH RESOURCE ALLOCATION

S u m m a r y

In the paper the actual state of the classical single machine scheduling problems under criteria:the maximum cost,the maximum completion time ,the maximum lateness,the maximum tardiness ,the weighted sum of completion times and the weighted sum of tardinesses, is presented.These problems are generalized on the case when job processing times depend linearly on amounts of locally and globally constrained resources.For the most of the above problems polynomial-time algorithms are found.It is shown that certain problems are NP-hard.