ZESZYTY NAUKOWE POLITECHNIKI ŚLĄSKIEJ 1988
Seria:AUTOMATYKA z.94 Nr kol.970

Stanisław Zdrzałka
Politechnika Wrocławska

PROBLEMY SZEREGOWANIA ZADAŃ Z PRZEZBROJENIAMI MASZYN

Streszczenie. W pracy rozpatrujemy klasę problemów szeregowania, w której zbiór zadań rozbity jest na grupy (podzbiory) wymagające innego uzbrojenia maszyn, przy czym z każdym przezbrojeniem związany jest pewien czas lub koszt. Dokonuje się przeglądu problemów z tej klasy, ich własności, istniejących algorytmów oraz wyników dotyczących złożoności obliczeniowej.

1. Wstęp

Celem pracy jest przedstawienie aktualnego stanu badań nad zagadnieniami szeregowania zadań z przezbrojeniami maszyn. Problemy z tej klasy charakteryzują się tym, że zbiór zadań do wykonania rozbity jest na grupy (podzbiory), z których każda wymaga innej organizacji pracy gniazda produkcyjnego, innego zestawu narzędzi itp., innymi słowy, innego uzbrojenia maszyn. Z każdym przezbrojeniem maszyn związany jest pewien czas lub koszt wpływający na wartość przyjętego wskaźnika jakości działania systemu produkcyjnego.

Wiele zagadnień z przezbrojeniami maszyn badanych było w kontekście planowania i sterowania produkcją. Przezbrojenia maszyn były stosunkowo rzadko uwzględniane w modelach procesów kolejnościowych, typowych dla teorii szeregowania - głównie ze względu na trudności charakterystyczne dla zagadnień optymalizacji kombinatorycznej. Niemniej jednak, od czasu pojawienia się w 1964 roku pierwszej, znanej nam, publikacji ujmującej przezbrojenia maszyn jako istotny element procesu kolejnościowego (Gilmore, Gomory [10]), pojawiła się już spora porcja literatury na ten temat. Szczególny wzrost zainteresowania obserwuje się ostatnio, w związku z zastosowaniami w elastycznych systemach produkcyjnych, w zautomatyzowanych, komputerowo sterowanych liniach produkcyjnych oraz w zagadnieniach związanych ze sterowaniem pracą robotów.

Dotychczasowe badania zagadnień szeregowania z przezbrojeniami maszyn nie były jednak systematyczne, tak jak w przypadku najprostszych modeli teorii szeregowania. W większości robione były one dla zauważonych, specyficznych problemów, niejako na marginesie głównego nurtu teorii szeregowania. Ponieważ brakuje w literaturze prac przeglądowych poświęconych tej tematyce, w niniejszej pracy podejmujemy próbę usystematyzowania dotychczasowych sformułowań zagadnień szeregowania zadań z przezbrojeniami maszyn i otrzymanych wyników, przyjmując jako bazę aparat pojęciowy stosowany w klasycznych zagadnieniach (bez przezbrojeń) oraz przyjęte dla tych zagadnień sposoby klasyfikacji problemów. W szczególności zwracamy uwagę na złożoność obliczeniową poszczególnych zagadnień, algorytmy i własności przydatne dla konstrukcji algorytmów, dokładnych bądź aproksymacyjnych.

2. Notacja i terminologia

Ponieważ rozważane w pracy zagadnienia są w większości uogólnieniami klasycznych zagadnień teorii szeregowania, wykorzystywać będziemy tutaj terminy i notację powszechnie stosowane w tej teorii (patrz np. praca przeglądowa Grahama et al. [12]), uzupełniając je jedynie o elementy związane ze specyfiką przezbrojeń.

Dany jest zbiór zadań $J=\{J_1,\ldots,J_n\}$ oraz zbiór maszyn $M=\{M_1,\ldots,M_m\}$; tam gdzie nie będzie to prowadziło do nieporozumień, zbiory te będziemy utożsamiać ze zbiorami indeksów, odpowiednio, zadań i maszyn. W danej chwili każde zadanie może być wykonywane przez co najwyżej jedną maszynę i każda maszyna może wykonywać co najwyżej jedno zadanie.

Opis zadań: Jeżeli zadanie J_j wykonywane jest przez więcej niż jedną maszynę, to opisywane jest za pomocą: zbioru operacji $\{O_{1j},\ldots,O_{mj}\}$ (gdzie operacja jest częścią zadania wykonywaną na jednej z maszyn), liniowego porządku na tym zbiorze, określającego kolejność wykonywania operacji, i funkcji μ przyporządkowującej O_{ij} indeks maszyny $\mu_{ij}=\mu(O_{ij})$. Ponadto, na opis zadania składają się: p_j (p_{ij}) – czas wykonywania zadania J_j (czas wykonywania zadania J_j na M_i lub czasy wykonywania wszystkich operacji O_{ij} zadania J_j); $p_j,p_{ij}\geq 0$; r_j – najwcześniejszy możliwy termin rozpoczęcia wykonywania J_j; $r_j\geq 0$; d_j – pożądany termin zakończenia wykonywania J_j; $d_j\geq 0$ (przez $\hat{d}_j$ oznaczamy linię krytyczną zadania J_j).

Zbiór zadań J jest rozbity na B wzajemnie rozłącznych podzbiorów $I_1,\ldots,I_B$ $(I_a\cap I_b=\emptyset$ dla $a\neq b$ oraz $\bigcup_{b=1}^{B} I_b=J)$, które dalej nazywamy grupami. Jeżeli maszyna M_i wykonuje zadanie z grupy I_b bezpośrednio po zadaniu z grupy I_a, wtedy po wykonaniu tego ostaniego zadania musi ona ulec przezbrojeniu, przy czym czas przezbrojenia wynosi $s^i_{ab}\geq 0$, zaś koszt jest równy $c^i_{ab}\geq 0$; $s^i_{bb}=0$ i $c^i_{bb}=0$ dla każdego b oraz i; w tym przypadku mówimy o czasach lub kosztach zależnych od kolejności. Jeżeli czas lub koszt zależy tylko od grupy zadania wykonywanego po przezbrojeniu (grupy I_b), lub jest stały, piszemy, odpowiednio, s_b^i, c_b^i lub s^i, c^i; są to czasy (koszty) niezależne od kolejności.

W dalszym ciągu zakładać będziemy, że czasy i koszty przezbrojeń spełniają nierówność trójkąta: $s_{ab}^i+s_{bc}^i \geq s_{ac}^i$, $c_{ab}^i+c_{bc}^i \geq c_{ac}^i$. Założenie to, powszechnie przyjmowane w literaturze, jest naturalnie spełnione wszędzie tam, gdzie przezbrojenia odbywają się w możliwie najkrótszym czasie lub możliwie najmniejszym kosztem.

Przyjmować będziemy również, że maszyna nie wymaga przezbrojenia przed rozpoczęciem wykonywania pierwszego oraz po zakończeniu ostatniego zadania. Jeżeli jednak wymaganie takie jest narzucone, wówczas możemy je uwzględnić w tym modelu, wprowadzając fikcyjne zadania z zerowymi czasami wykonywania.

Oprócz tego, w opisie zadań może wystąpić relacja częściowego porządku $\prec$ na J określająca ograniczenia kolejnościowe; jeżeli $J_k \prec J_l$, to J_l może być rozpoczęte po wykonaniu J_k.

Opis maszyn: Rozpatrywać będziemy tradycyjne w teorii szeregowania struktury maszynowe (patrz [12]). W przypadku gdy każde z zadań wykonywane jest na jednej maszynie, wyróżniamy struktury: z pojedynczą maszyną (oznaczenie: 1) i z m równoległymi, identycznymi maszynami (P; jeżeli m jest ustalone, to Pm). Gdy zadania wymagają więcej niż jednej maszyny, rozpatrujemy:

- strukturę przepływową (zagadnienie przepływowe, F, ang. flow shop): $m_j=m$, $\mu_{ij}=i$ dla każdego j, ciąg $(O_{1j},...,O_{mj})$ określa porządek operacji dla $j\in J$;
- strukturę otwartą (zagadnienie otwarte, O, ang. open shop): tak jak w strukturze przepływowej, poza tym, że porządek operacji nie jest istotny;
- strukturę gniazdową (zagadnienie gniazdowe, J, ang. job shop): obejmuje ona pozostałe przypadki.

Funkcje celu: Przez uszeregowanie rozumiemy przyporządkowanie każdemu zadaniu przedziałów czasu, w których jest ono wykonywane przez poszczególne maszyny, spełniające sformułowane ograniczenia. Uszeregowanie oznaczamy przez ϕ, a zbiór wszystkich (dopuszczalnych) uszeregowań przez Φ. Jeżeli zadania wykonywane są bez przerwań, w jednakowej kolejności na każdej z maszyn (zagadnienie przepływowe permutacyjne), to każde uszeregowanie charakteryzujące się brakiem zbędnych przestojów maszyn jest w jednoznaczny sposób określone przez podanie kolejności, permutacji π na zbiorze J lub równoważnie, listy zadań L (uporządkowanego zbioru J); przez $\pi(i)$ oznaczamy indeks zadania stojącego na pozycji i w π. Zbiór wszystkich dopuszczalnych permutacji na J oznaczamy przez Π.

Znając uszeregowanie ϕ (π, L), możemy określić dla każdego zadania J_j czas zakończenia wykonywania C_j oraz koszt przezbrojenia maszyny (maszyn) po wykonaniu zadania G_j; w przypadku gdy J_j wykonywane jest przez więcej niż jedną maszynę, G_j jest sumą kosztów przezbrojeń występujących po wykonaniu poszczególnych operacji zadania J_j.

W dotychczasowej literaturze wyróżnia się dwa typy wskaźników jakości uszeregowania. Są one konstruowane w oparciu o odpowiednio C_j i G_j i oznaczane dalej przez F_1 i F_2.

Wskaźnik F_1 (koszt uszeregowania) odpowiada funkcjom celu stosowanym tradycyjnie w teorii szeregowania. Niech f_j będzie funkcją przyporządkowującą każdemu C_j koszt $f_j(C_j)$; f_j jest funkcją niemalejącą. Koszt uszeregowania ϕ definiowany jest jako

$$F_1(\phi) = \max_{j\in J} f_j(C_j) \quad \text{lub} \quad F_1(\phi) = \Sigma_{j\in J} f_j(C_j),$$

przy czym funkcje f_j mają najczęściej postać: $f_j(C_j)=C_j$, $f_j(C_j)=L_j$, gdzie $L_j=C_j-d_j$ - nieterminowość, $f_j(C_j)=T_j$, gdzie $T_j=\max\{0,C_j-d_j\}$ - opóźnienie, $f_j(C_j)=U_j$, gdzie $U_j=0$, jeżeli $C_j\le d_j$, $U_j=1$, w przeciwnym przypadku. Istotnymi wskaźnikami typu F_1 są w naszym opracowaniu: $C_{max}=\max_{j\in J}C_j$ - czas zakończenia wykonywania wszystkich zadań, $L^*_{max}=\max_{j\in J}L_j$ - maksymalna nieterminowość, ΣC_j i Σw_jC_j - łączny i ważony czas przebywania zadań w systemie, ΣU_j - liczba opóźnionych zadań.

Wskaźnik F_2 (koszt przezbrojeń) definiowany jest dla uszeregowania ϕ jako suma kosztów wszystkich przezbrojeń związanych z ϕ,

$$F_2(\phi)=\Sigma_{j\in J}G_j.$$

<u>Problemy szeregowania z przezbrojeniami maszyn:</u> Wśród rozpatrywanych w literaturze sformułowań zagadnień szeregowania z przezbrojeniami maszyn wyróżniają się trzy następujące klasy problemów:

(P1) Minimalizacja kosztu przezbrojeń: Znaleźć $\phi^\circ\in\Phi$, dla którego $F_2(\phi^\circ)=\min_{\phi\in\Phi}F_2(\phi)$.

(P2) Minimalizacja kosztu przezbrojeń przy ograniczonym koszcie uszeregowania: Dla zadanego $\alpha\in R$ znaleźć $\phi^\circ\in\Phi$, dla którego

$$F_2(\phi^\circ)=\min\{F_2(\phi):\ \phi\in\Phi,\ F_1(\phi)\le\alpha\}.$$

(P3) Minimalizacja kosztu uszeregowania: Znaleźć $\phi^\circ\in\Phi$, dla którego $F_1(\phi^\circ)=\min_{\phi\in\Phi}F_1(\phi)$.

Sporadycznie rozważano również minimalizację $F_1(\phi)+F_2(\phi)$ na zbiorze Φ. We wczesnych pracach badano głównie przypadki, kiedy ilość grup równała się ilości zadań, $B=n$; każde zadanie wymagało przezbrojenia maszyny. Problemy z $B<n$ są z punktu widzenia teorii złożoności obliczeniowej trudniej-

sze. Na przykład, jednomaszynowy problem szeregowania z czasami przezbrojeń niezależnymi od kolejności i z funkcją celu L_{max} jest NP-trudny dla B<n, [3], podczas gdy dla B=n pozostaje rozwiązywalny w wielomianowym czasie. Problemy te są jednak istotne dla planowania i sterowania w systemach wytwarzających wiele grup (partii) produktów za pomocą tych samych urządzeń (np. w elastycznych systemach produkcyjnych). Systemy takie rodzą zapotrzebowanie na jeszcze bardziej skomplikowane modele. W pracach [23], [24] badane są sytuacje, w których występuje zróżnicowanie typów czasów przezbrojeń. "Duże czasy" występują przy przejściu pomiędzy grupami zadań, ale ponadto, każda grupa składa się z wielu partii jednakowych zadań i przy przejściu pomiędzy dwoma różnymi partiami tej samej grupy występują dodatkowe "małe czasy" przezbrojeń.

W dalszym ciągu przedstawimy problemy z poszczególnych klas, rozpoczynając od najprostszych. Dla zapisu problemów szeregowania z przezbrojeniami maszyn stosować będziemy zapis trójpolowy $\alpha|\beta|\gamma$, wprowadzony przez Grahama et al. [12] dla tradycyjnych zagadnień, uzupełniony o informacje dotyczące czasów i kosztów przezbrojeń. Przypomnijmy, że w zapisie tym w polu α określona jest struktura maszyn (np. 1, P, F, Pm, Fm, itd.), w polu β podane są informacje dotyczące zadań (np. pmtn - zadania podzielne, prec - ograniczenia kolejnościowe $\prec$, r_i - istnieje przynajmniej jedno zadanie J_j, dla którego $r_j>0$, s_{ab} - występują czasy przezbrojeń zależne od kolejności zadań, itp.), natomiast w polu γ podawana jest funkcja celu (np. ΣG_j, C_{max}, itp.). Na przykład, $1|pmtn,prec,s_{ab}|L_{max}$ oznacza problem jednomaszynowy z podzielnymi zadaniami, z ograniczeniami kolejnościowymi, w którym $r_j=0$ dla każdego j, występują czasy przezbrojeń zależne od kolejności zadań, a funkcją celu jest L_{max}. Ten sam problem z zadaniami niepodzielnymi i bez ograniczeń kolejnościowych zapisany będzie jako $1|s_{ab}|L_{max}$.

3. Problemy jednomaszynowe

W punkcie tym dokonamy przeglądu problemów jednomaszynowych. Zauważmy najpierw, że każdą listę zadań L, reprezentującą uszeregowanie, można przedstawić jako ciąg list częściowych $L=(L_1,L_2,...,L_v)$, gdzie L_i zawiera zadania z tej samej grupy, różnej od grup, do których należą zadania z list L_{i-1} i L_{i+1}. Ciąg $(L_1,...,L_v)$ nazywać będziemy grupową reprezentacją listy L. Oczywiste jest, że $v \geq B$. Jeżeli v=B, to listę $L=(L_1,...,L_B)$ nazywamy uszeregowaniem grupowym. W uszeregowaniu takim zadania z każdej grupy wykonywane są kolejno po sobie.

3.1. Problemy $1|c_{ab}|\Sigma G_j$ i $1|s_{ab}|C_{max}$

Zachodzi następująca, oczywista własność.

<u>Własność 3.1.</u> W problemach $1|c_{ab}|\Sigma G_j$ i $1|s_{ab}|C_{max}$, jeżeli koszty i czasy przezbrojeń spełniają nierówność trójkąta, to zbiór uszeregowań optymalnych zawiera przynajmniej jedno uszeregowanie grupowe.

Z powyższego wynika, że problemy $1|c_{ab}|\Sigma G_j$ i $1|s_{ab}|C_{max}$ sprowadzają się (przy założeniu podanym we Własności 3.1) do zagadnienia szeregowania list częściowych odpowiadających grupom; porządek zadań wewnątrz tych list nie jest istotny. Niech δ będzie permutacją zbiotu $\{1,\ldots,B\}$ a Δ, zbiorem wszystkich permutacji δ. W $1|c_{ab}|\Sigma G_j$ należy zatem znaleźć $\delta^{\circ}\in\Delta$, dla którego $\sum_{i=1}^{B-1} c_{\delta(i)\delta(i+1)}$ osiąga minimum na zbiorze Δ. To samo zagadnienie, tylko z s_{ab} w miejscu c_{ab}, należy rozwiązać dla $1|s_{ab}|C_{max}$. Jak widać, problem ten jest równoważny problemowi komiwojażera, w którym $\{0,1,\ldots,B\}$ jest zbiorem miast, a c_{ab}, $a,b=0,1,\ldots,B$, są odległościami pomiędzy miastami; $c_{b0}=c_{0b}=0$ dla $b=1,\ldots,B$. Ponieważ ten ostatni problem jest NP-trudny [7], zatem $1|c_{ab}|\Sigma G_j$ i $1|s_{ab}|C_{max}$ są również NP-trudne.

Gilmore i Gomory [10] przedstawili algorytm o złożoności $O(n^2)$ dla szczególnego przypadku problemu $1|c_{ab}|\Sigma G_j$. W problemie tym B=n, $c_{ij}=\int_{B_i}^{A_j} f(x)dx$, jeżeli $A_j\geq B_i$, $c_{ij}=\int_{A_j}^{B_i} g(x)dx$, w przeciwnym przypadku, gdzie A_k jest stanem maszyny wymaganym przed rozpoczęciem wykonywania zadania J_k, natomiast B_k, stanem jaki maszyna osiąga po wykonaniu J_k. Zakłada się, że $f(x)+g(x)\geq 0$ dla każdego x, przy czym f(x) i g(x) należy interpretować jako gęstości kosztu występujące przy, odpowiednio, zwiększaniu i zmniejszaniu zmiennej stanu. Założenie to jest odpowiednikiem nierówności trójkąta dla c_{ab}; zwiększenie zmiennej stanu i następnie zmniejszenie jej nie przynosi zysku. Powyższy model znajduje zastosowanie w procesach obróbki cieplnej zachodzących w piecach; temperatura pieca jest zmienną opisującą stan maszyny, natomiast f(x) i g(x) są kosztami jednostkowymi, odpowiednio, grzania i studzenia pieca w sytuacji, kiedy jego temperatura wynosi x. Algorytm oraz pewne własności problemu podane są w [10].

Ponieważ istnieje oddzielna, obszerna literatura poświęcona dokładnym i aproksymacyjnym algorytmom dla problemu komiwojażera, patrz np. [16], dalszych aspektów obliczeniowych problemów $1|c_{ab}|\Sigma G_j$ i $1|s_{ab}|C_{max}$ nie będziemy tu poruszać. Należy tylko zauważyć, że szczególne przypadki $1|s_b|\Sigma G_j$ i $1|c_b|C_{max}$ są trywialne; każde uszeregowanie grupowe jest optymalne.

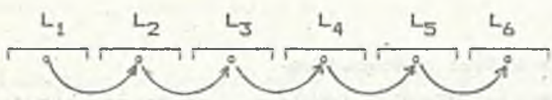
3.2. Problem $1|prec,c_{ab}=1|\Sigma G_j$

Problem minimalizacji ilości przezbrojeń przy zadanych ograniczeniach kolejnościowych (rozważany był przez Richtera [22]. Ponieważ należy on do podstawowych zagadnień szeregowania z przezbrojeniami maszyn, omówimy go szczegółowo. W tym celu pomocne będą następujące oznaczenia i definicje.

Niech $P(j)$ i $S(j)$ będą zbiorami wszystkich, odpowiednio, poprzedników i następników zadania j, bezpośrednich lub pośrednich, oraz niech dla $J' \subset J$, $P(J')=\cup_{j\in J'}P(j)$, $S(J')=\cup_{j\in J'}S(j)$. Lista $L=(j_1,\dots,j_n)$ jest zgodna, jeżeli dla każdej pary (j_k,j_l), $j_k<j_l$ implikuje $k<l$. Częściowe listy L_k, L_l, $k<l$, nazywamy b-sąsiadami, jeżeli $L_k,L_l \subset I_b$ i $(L_{k+1}\cup \dots \cup L_{l-1})\cap I_b=\emptyset$. Zadania k i l należące do b-sąsiadów, odpowiednio, L_k i L_l są separowane, jeżeli $(S(k)\cap P(l))\setminus I_b \neq \emptyset$, to znaczy, istnieje zadanie $j \notin I_b$, które jest następnikiem k i poprzednikiem l, bezpośrednim lub pośrednim. Niech dalej $L'_1,\dots,L'_s$ będą listami częściowymi reprezentacji grupowej listy L zawierającymi wszystkie zadania z grupy I_b. Zgodną listę L nazywamy b-trywialną, jeżeli istnieje podciąg zadań $j_1 \in L'_1,\dots,j_s \in L'_s$, w którym zadania j_k,j_{k+1}, $k=1,\dots,s-1$, są separowane.

<u>Własność 3.2.</u> [22] W problemie $1|prec,c_{ab}=1|\Sigma G_j$, jeżeli lista L jest b-trywialna dla każdego b, to jest ona listą optymalną.

Sytuację, dla której zachodzi powyższa własność, ilustruje rys.1



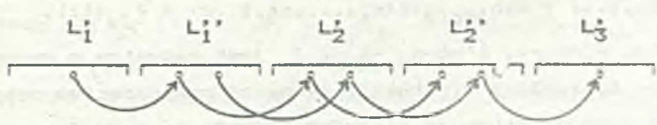
Rys.1. Lista 1-, 2-trywialna dla problemu z B=2:

Fig.1. The 1-, 2-trivial list for a problem with B=2.

Niech zgodna lista L dana będzie w postaci: $L=(\dots L'_1 \dots L''_1 \dots L'_2 \dots L''_2 \dots$ $\dots L'_{s-1} \dots L''_{s-1} \dots L'_s)$, gdzie $\cup_{l=1}^{s} L'_l=I_a$ oraz $\cup_{l=1}^{s-1} L''_l=I_b$, $a \neq b$. Listę tę nazywamy a,b-trywialną, jeżeli istnieją podciągi zadań: $j'_1 \in L'_1$, $l'_1,l''_1 \in L''_1$, $j'_2,j''_2 \in L'_2,\dots,j'_{s-1},j''_{s-1} \in L'_{s-1}$, $l'_{s-1},l''_{s-1} \in L''_{s-1}$, $j'_s \in L'_s$ takie, że

$$j'_k \in P(l'_k) \text{ i } l'_k \in P(j'_{k+1}) \text{ dla } k=1,\dots,s-1,$$

$$l''_k \in P(j''_{k+1}) \text{ i } j''_{k+1} \in P(l''_{k+1}) \text{ dla } k=1,\dots,s-1.$$



Rys.2. Lista 1,2-trywialna dla problemu z B=2; $L'_1 \cup L'_2 \cup L'_3=I_1$, $L''_1 \cup L''_2=I_2$.

Fig.2. The 1,2-trivial list for a problem with B=2; $L'_1 \cup L'_2 \cup L'_3=I_1$, $L''_1 \cup L''_2=I_2$.

Własność 3.3. [22] W problemie $1|prec,c_{ab}=1|\Sigma G_j$, jeżeli lista L jest a,b-trywialna dla każdego a, to jest ona listą optymalną.

Rys.2 ilustruje powyższą własność dla B=2. Własności 3.2 i 3.3 sugerują następujący

Algorytm.

1) Utwórz listę zgodną $L=(j_1,...,j_n)$.

2) Dla $t=1,...,n-2$ znajdź I_b takie, że $j_t \in I_b$, oraz $l=\min\{q: j_q \in I_b, q>t, j_r \not< j_q, t<r<q\}$, i jeżeli $l>t+1$, to przesuń j_l bezpośrednio za j_t.
Wynik: $L=(j_1,...,j_n)$

3) Dla $t=n,...,3$ znajdź I_b takie, że $j_t \in I_b$ oraz $l=\max\{q: j_q \in I_b, q<t, j_q \not< j_r, q<r<t\}$, i jeżeli $l<t-1$, to przesuń j_l bezpośrednio przed j_t.

W [22] pokazano, że jeżeli B=2, to algorytm ten generuje optymalną listę. Dla B>2 algorytm daje listę optymalną, jeżeli zagadnienie nie zawiera nietrywialnej pary zadań; parę zadań $k,l \in I_b$ nazywamy nietrywialną, jeżeli $l \notin P(k)$, k i l nie są separowane oraz $S(k) \cap I_a \neq \emptyset$ i $P(l) \cap I_c \neq \emptyset$ dla różnych a,b, c.

Pokazano również [22], że dla dowolnego B problem $1|prec,c_{ab}=1|\Sigma G_j$ jest NP-trudny. Brak jest w literaturze algorytmów dokładnych i aproksymacyjnych dla przypadku B>2.

3.3. Problem $1|s_{ab}|\Sigma w_j C_j$

Monma i Potts [20] zauważyli następującą

Własność 3.4. W problemie $1|s_{ab}|\Sigma w_j C_j$, jeżeli s_{ab} spełniają nierówność trójkąta, to istnieje uszeregowanie optymalne, w którym zadania każdej grupy występują względem siebie w kolejności zgodnej z niemalejącymi p_j/w_j.

Własność ta umożliwia konstrukcję algorytmu programowania dynamicznego o wielomianowej złożoności obliczeniowej dla ustalonego B. Niech $|I_b|=N_b$, $b=1,...,B$, oraz niech zadania każdej grupy będą ponumerowane oddzielnie, zgodnie z niemalejącymi p_j/w_j. Oznaczmy przez $\ell(n_1,...,n_B,t,a)$ minimalny koszt częściowego uszeregowania zawierającego n_b pierwszych zadań z każdej grupy I_b, w którym ostatnio ustawione zadanie kończy się w chwili t i pochodzi z grupy I_a; $\ell(0,...0,0,0)=0$. Dla t>0 i a>0

$$\ell(n_1,...,n_B,t,a) = \min_{1 \le c \le B}(\ell(n_1',...,n_B',t',c) + f_{j_c}(t)), \tag{1}$$

gdzie $n_b'=n_b$ dla $b \neq a$, $n_a'=n_a-1$, $t'=t-p_{j_c}-s_{ca}$, j_c jest zadaniem o numerze n_a z grupy I_a, $f_{j_c}(t)=w_{j_c}t$; zakłada się tutaj, że przed rozpoczęciem wykonywania pierwszego zadania maszyna wykonuje fikcyjne zadanie z grupy I_0 o zerowym czasie trwania, a s_{0b} należy interpretować jako czas początkowego przezbrojenia. Optymalne uszeregowanie znajdujemy wybierając minimalne $g(N_1,\dots,N_B,t,a)$ dla pewnego czasu zakończenia wykonywania zadań t i pewnej grupy I_a, do której należy ostatnie zadanie. Zwróćmy jeszcze uwagę na to, że wymagania dotyczące pamięci m.c. możemy obniżyć zastępując zmienną t zmiennymi t_{bc}, $b=0,\dots,B$, $c=1,\dots,B$, reprezentującymi liczby przezbrojeń z grupy b do grupy c w częściowyn uszeregowaniu. Zmienną t można teraz prosto wyliczyć: $t=\Sigma_{b=0}^{B}\Sigma_{b=1}^{B}t_{bc}s_{bc} + \Sigma_{b=1}^{B}\Sigma_{i=1}^{n_b}p_{ib}$.

W [20] pokazano, że algorytm ten posiada złożoność obliczeniową $O(B^2n^B\min(n^\tau,T))$, gdzie τ jest liczbą różnych wartości s_{ab}, $\tau\le B^2+B$, a $T=\Sigma_{j\in J}p_j + \Sigma_{b=1}^{B}N_b\max_{0\le a\le B}s_{ab}$.

Fakt, że dla ustalonego B przedstawiony algorytm rozwiązuje problem $1|s_{ab}|\Sigma w_jC_j$ w wielomianowym czasie, ma znaczenie wyłącznie teoretyczne. Złożoność obliczeniowa (w najgorszym przypadku) $O(B^2n^{B^2})$ wyklucza jego praktyczne zastosowanie, nawet w problemach o bardzo małych rozmiarach.

Złożoność obliczeniowa problemu $1|s_{ab}|\Sigma w_jC_j$ dla dowolnego B pozostaje kwestią otwartą. Nie jest również znana złożoność przypadków szczególnych, kiedy czasy przezbrojeń nie zależą od kolejności i kiedy są stałe.

Gupta [13] przedstawił algorytm o złożoności $O(n\log n)$ dla przypadku, gdy $B=2$ i $w_j=1$, $j\in J$. Oto szkic tego podejścia. Rozważmy dwie listy $L_1=(\sigma\rho ij\pi)$ i $L_2=(\sigma i\rho j\pi)$, gdzie σ, ρ, π są listami zadań z wzajemnie rozłącznych podzbiorów zbioru J, odpowiednio, k_1, k, $n-k_1-k-2$ -elementowych, przy czym ostatnie zadanie w σ należy do I_a, $\rho\subset I_b$; $i\in I_a$, j jest zadaniem z dowolnej grupy. Niech $k_2=k_1+k$ oraz niech F(L) oznacza wartość funkcji celu dla L; jeżeli $k_2=n-1$, zakłada się, że $j\in I_a$.

<u>Własność 3.5.</u> [13] A) Jeżeli $j\in I_a$, to $F(L_2)\le F(L_1)$ wtedy i tylko wtedy, gdy
(i) $kp_i \le s_{ab}+s_{ba}+\Sigma_{r\in\rho}p_r$, $1\le k_1<k_2\le n-1$,
(ii) $kp_i+(n-1)s_{ab} \le s_{ab}+\Sigma_{r\in\rho}p_r$, $k_1=0$, $1\le k_2\le n-1$.
B) Jeżeli $j\in I_b$, to $F(L_2)\le F(L_1)$ wtedy i tylko wtedy, gdy
(iii) $kp_i \le (n-k_2)(s_{ab}+s_{ab})+\Sigma_{r\in\rho}p_r$, $1\le k_1<k_2\le n-2$,
(iv) $k(p_i+s_{ab}) \le (n-k)s_{ab}+\Sigma_{r\in\rho}p_r$, $k_1=0$, $1\le k_2\le n-2$.

<u>Algorytm</u>
0) (Inicjacja) Uporządkuj zadania według niemalejących p_j; wynik: $L=(j_1,\dots,j_n)$. Podstaw $k_1:=0$, b:=indeks grupy, do której należy j_1, $a\ne b$. Znajdź k, dla którego $j_t\in I_b$, $t=1,\dots,k$, $j_{k+1}\notin I_b$, i podstaw $\rho:=(j_1,\dots,j_k)$, $i:=j_{k+1}$, $j:=j_{k+2}$, $\pi:=(j_{k+3},\dots,j_n)$.
1) (Przypadek $k_1=0$) Jeżeli $j\in I_a$, przejdź do 1A; w przeciwnym przypadku. przejdź do 1B.

1A) Jeżeli zachodzi (ii), podstaw $L:=(i\rho j\pi)$, $k_1:=1$; w przeciwnym przypadku $k_1:=k$ i $a:=b$. Przejdź do 2.

1B) Jeżeli zachodzi (iv), podstaw $L:=(i\rho j\pi)$, $k_1:=1$; w przeciwnym przypadku $k_1:=k$ i $a:=b$. Przejdź do 2.

2) (Przypadek $k_1>0$) Znajdź k_2, dla którego $j_{k_2}\in I_a$ i $j_t\in I_b$ dla $t=k_1+1,..,k_2$. Jeżeli $k_2<n$ nie istnieje, koniec; w przeciwnym przypadku podstaw $\sigma:=(j_1,...,j_{k_1})$, $\rho:=(j_{k_1+1},...,j_{k_2})$, $\pi:=(j_{k_2+3},...,j_n)$, $i:=j_{k_2+1}$, $j:=j_{k_2+2}$, $k:=k_2-k_1$. Jeżeli $j\in I_a$, przejdź do 2A; w przeciwnym przypadku, do 2B.

2A) Jeżeli zachodzi (i), podstaw $L:=(\sigma i\rho j\pi)$, $k_1:=k_1+1$; w przeciwnym przypadku, $k_1:=k_2$ i $a:=b$. Przejdź do 2.

2B) Jeżeli zachodzi (iii), podstaw $L:=(\sigma i\rho j\pi)$, $k_1:=k_1+1$; w przeciwnym przypadku, $k_1:=k_2$, $a:=b$. Przejdź do 2.

Brak jest w literaturze algorytmów aproksymacyjnych dla przypadku $B>2$.

3.4. Problem $1|s_{ab}|L_{max}$

W pracy [3] pokazano, że dla jednostkowych czasów przezbrojeń problem wyznaczenia uszeregowania spełniającego warunek $L_j\le 0$ dla każdego j jest NP-zupełny. Stąd wynika, że problem $1|s|L_{max}$ jest NP-trudny, i w konsekwencji do klasy problemów NP-trudnych należą również $1|s_b|L_{max}$ i $1|s_{ab}|L_{max}$. W [20] wykazano własność analogiczną do Własności 3.4.

<u>Własność 3.6</u>. W problemie $1|s_{ab}|L_{max}$, jeżeli s_{ab} spełniają nierówność trójkąta, to istnieje uszeregowanie optymalne, w którym zadania każdej grupy występują względem siebie w kolejności zgodnej z niemalejącymi d_j.

Zatem algorytm programowania dynamicznego sformułowany dla $1|s_{ab}|\Sigma w_jC_j$ pozostaje ważny dla $1|s_{ab}|L_{max}$; w tym przypadku w (1) należy przyjąć $f_{j_c}(t)=t-d_{j_c}$, a operację sumowania (po prawej stronie (1)) należy zastąpić operacją "maksimum". Podobnie jak w przypadku $1|s_{ab}|\Sigma w_jC_j$, algorytm ma znaczenie wyłącznie teoretyczne.

W [27] przedstawiono wyniki analizy najgorszego przypadku pewnych prostych algorytmów aproksymacyjnych dla problemu $1|s,q_j|C_{max}$, który jest równoważny problemowi $1|s|L_{max}$. W $1|s,q_j|C_{max}$ każde zadanie J_j wykonywane jest najpierw na M_1, o przepustowości 1, przez p_j jednostek czasu, a następnie, począwszy od momentu zakończenia C_j (na M_1), na M_2, o nieskończonej przepustowości, przez $q_j = \max_{i\in J}d_i-d_j \triangleq D-d_j$ jednostek czasu. Przezbrojeniom podlega tylko maszyna M_1, przy czym $s^1_{ab}=s$ dla każdego a i b. Równoważność wynika z faktu, że $L_{max}=\max_{j\in J}(C_j-d_j)=\max_{j\in J}(C_j+D-d_j)-D=\max_{j\in J}(C_j+q_j)-D$. Oznaczmy przez K_A i K^*, odpowiednio, wartość funkcji celu otrzymaną w wyniku zastosowania algorytmu aproksymacyjnego A i minimum funkcji celu.

<u>Własność 3.7</u>. W problemie $1|s,q_j|C_{max}$,
a) dla każdego algorytmu A generującego uszeregownie grupowe, $K_A/K^* \le 2$, i ograniczenie to jest najlepsze z możliwych;
b) dla algorytmu A' porządkującego zadania według nierosnących q_j, $K_{A'}/C^* \le n-1$, i ograniczenie to jest najlepsze z możliwych.

Należy zauważyć, że gdy $s=0$, to algorytm A' generuje uszeregowanie optymalne (algorytm Jacksona). Opierając się na powyższych obserwacjach, w [27] zbadano całą klasę algorytmów łączących cechy szeregowania grupowego z cechami szeregowania według nierosnących q_j. Analiza najgorszego przypadku wykazała, że dla każdego z nich $K_A/K^* \le 2$, i że ograniczenie to jest osiągalne.

3.5. Problem $1|s_{ab}|\Sigma w_jU_j$

Problem minimalizacji ważonej liczby opóźnionych zadań rozważany był w pracy [20]. Problem ten jest NP-trudny, począwszy już od przypadku, gdy $s_{ab}=s$ dla każdego a i b, oraz $w_j=1$ dla każdego j (to znaczy, $1|s|\Sigma U_j$ jest NP-trudny). Jest to, podobnie jak w poprzednio rozważanym problemie, konsekwencja faktu wykazanego w [3], że problem wyznaczenia uszeregowania spełniającego warunek $L_j \le 0$ dla każdego $j \in J$ jest NP-zupełny. W [20] wykazano
<u>Własność 3.8</u>. W problemie $1|s_{ab}|\Sigma w_jU_j$, jeżeli s_{ab} spełniają nierówność trójkąta, to istnieje uszeregowanie optymalne, w którym nie opóźnione zadania każdej grupy występują względem siebie w kolejności niemalejących d_j.

W oparciu o powyższą własność zaproponowano algorytm programowania dynamicznego, który tworzy uszeregowanie zadań nieopóźnionych; zadania opóźnione wykonywane są na końcu w dowolnej kolejności. Jego złożoność obliczeniowa wynosi $O(B^2n^B\min\{W,D,T\})$, gdzie $W=\Sigma_{j\in J}w_j$, $D=\max_{j\in J}d_j$, $T=\Sigma_{j\in J}p_j+\Sigma_{b=1}^{B}N_b\max_{0\le b\le B}s_{ab}$, przy czym $N_b=|I_b|$, zaś s_{0b} jest czasem przezbrojenia początkowego. W przypadku gdy $w_j=1$ ($j\in J$), złożoność algorytmu wynosi $O(B^2n^{B+1})$. Oto szkicowe przedstawienie tego podejścia.

Przyjmujemy, że zadania z każdej grupy ponumerowane są oddzielnie (J_{jb}, p_{jb}, w_{jb}, d_{jb}, $j=1,\dots,N_b$, $b=0,1,\dots,B$) w kolejności niemalejących d_j. Niech $f(n_1,\dots,n_B,t,a)$ będzie minimalną wartością funkcji celu dla częściowego uszeregowania n_b pierwszych zadań z każdej grupy I_b, w którym ostatnie nieopóźnione zadanie pochodzi z grupy I_a i kończy się w chwili t. Wartości początkowe dane są następująco: $f(n_1,\dots,n_B,0,0)=\Sigma_{b=1}^{B}\Sigma_{i=1}^{n_b}w_{ib}$, $0\le n_b\le N_b$, $1\le b\le B$. Dla $t>0$ i $a>0$,

$$\mathcal{E}(n_1,\dots,n_B,t,a)=\begin{cases}\min(\min_{0\le c\le B}\mathcal{E}(n_1',\dots,n_B',t',c),\\ \mathcal{E}(n_1',\dots,n_B',t,a), & t\le d_{n_a a}\\ \mathcal{E}(n_1',\dots,n_B',t,a)) & t>d_{n_a a},\end{cases} \tag{2}$$

gdzie $n_b'=n_b$ dla $b\neq a$, $n_a'=n_a-1$, $t'=t-p_{n_a a}-s_{ca}$. Pierwszy element zbioru, dla którego szukamy minimum po prawej stronie (2), odpowiada sytuacji, kiedy zadanie n_a grupy I_a nie jest opóźnione i poprzednie zadanie, z grupy I_c, również nie jest opóźnione. Drugi i trzeci element odpowiadają sytuacji, w której zadanie n_a grupy I_a jest opóźnione, przy czym w ostatnim przypadku jest to oczywiste. Minimalną wartość funkcji celu i optymalne uszeregowanie zadań (nieopóźnionych) znajdujemy wybierając minimalne $\mathcal{E}(N_1,\dots,N_B,t,a)$ dla pewnego czasu t zakończenia wykonywania zadań nieopóźnionych i a, numeru grupy, z której pochodzi ostatnie nieopóźnione zadanie. Jeżeli czasy wykonywania zadań lub czasy przezbrojeń są duże, w [20] proponuje się zamienić zmienną t przez w, wartość funkcji celu uszeregowania częściowego, definiując równocześnie $\mathcal{E}(n_1,\dots,n_B,w,a)$ jako minimalny czas zakończenia wykonywania nieopóźnionych zadań uszeregowania częściowego. Równanie rekurencyjne dla tego wariantu można znaleźć w [20] (w ocenie złożoności obliczeniowej bierze się pod uwagę obydwa warianty).

W podsumowaniu należy stwierdzić, że podobnie jak dla $1|s_{ab}|\Sigma w_jC_j$ i $1|s_{ab}|L_{max}$, algorytm ten ma znaczenie raczej teoretyczne. Interesujący jest fakt, że problem $1|s_{ab}|\Sigma U_j$ okazuje się łatwiejszy (w sensie złożoności obliczeniowej) niż jego odpowiednik z funkcją celu L_{max}; co przeczy dotychczasowym doświadczeniom z badań poświęconych tym dwóm funkcjom celu.

Brak jest w literaturze algorytmów aproksymacyjnych dla $1|s_{ab}|\Sigma w_jU_j$.

3.6. Problem $1|s_{ab}=0,c_{ab},p_j=1,\hat{d}_j|\Sigma G_j$

Problem minimalizacji kosztu przezbrojeń, z zerowymi czasami przezbrojeń, jednostkowymi czasami trwania zadań i ograniczeniami wynikającymi z zadanych linii krytycznych $\hat{d}_j$, ma ważne zastosowanie praktyczne w planowaniu produkcji i był rozpatrywany przez wielu autorów; do istotnych prac należą [11], [19], [6] i [3].

Złożoność obliczeniowa problemu $1|s_{ab}=0,c_{ab},p_j=1,\hat{d}_j|\Sigma G_j$ nie jest znana. Bruno i Downey [3] pokazali, że $1|s_{ab}=0,c_{ab}=1,\hat{d}_j|\Sigma G_j$ jest problemem NP-trudnym, począwszy już od sytuacji, gdy $|I_b|=3$ i dwie linie krytyczne są liczbami skończonymi, natomiast problem $1|s_{ab}=0,c_b,\hat{d}_j|\Sigma G_j$ jest NP-trudny już dla $|I_b|=2$ i dla jednej skończonej linii krytycznej.

Interpretacja związana z planowaniem produkcji jest następująca. Pojedyncza maszyna wytwarza w jednostce czasu (np. dzień) jednostkę produktu typu b ($1\le b\le B$). Dla każdego typu produktu określone jest zapotrzebowanie (zamówienie) rozłożone w czasie, to znaczy podane są terminy i ilości jednostek produktów, które należy wykonać w tych terminach (terminy dostaw wyrażone są jednostkach czasu pracy maszyny). Czas trwania przezbrojenia z produktu typu a na typ b nie jest istotny, ponieważ odbywa się ono poza efektywnym czasem pracy maszyny. Istotny jest natomiast koszt przezbrojenia. Należy znaleźć harmonogram pracy maszyny zapewniający wykonanie wszystkich zamówień w zadanych terminach i minimalizujący łączny koszt przezbrojeń.

Zgodnie z tą interpretacją, zadaniu odpowiada jednostka produktu, zaś linia krytyczna zadania jest terminem dostawy tej jednostki.

Niech $t=1,\ldots,T$ będą numerami kolejnych jednostek czasu pracy maszyny. Oznaczmy przez $x(t)$ wektor B-wymiarowy, w którym jedynka na miejscu b oznacza, że w jednostce czasu t wykonywane jest zadanie z grupy (typu) b; pozostałe składowe są zerowe. Niech $D(t)$ będzie również wektorem B-wymiarowym określającym łączne zapotrzebowanie na produkty poszczególnych typów w przedziale czasu $1,2,\ldots,t$. Niech dalej $C=[c_{ab}]^{B\times B}$ będzie macierzą kosztów przezbrojeń. Powyższy problem można zapisać następująco

$$\min(\Sigma_{t=1}^{T-1}x'(t)Cx(t+1):\ \Sigma_{k=1}^{t}x(k)\ge D(t),\ t=1,\ldots,T), \qquad (3)$$

gdzie $x\ge D$ oznacza $x_b\ge D_b$ dla $b=1,\ldots,B$. Oczywiste jest [11], że rozwiązanie dopuszczalne dla tego problemu istnieje wtedy i tylko wtedy, gdy $\Sigma_{b=1}^{B}D_b(t)\le t$ dla $t=1,\ldots,T$.

Zagadnienie $1|s_{ab}=0,c_{ab},p_j=1,\hat{d}_j|\Sigma G_j$ formułowane było jak dotąd jako problem minimalizacyjny (3). Glassey [11] podał algorytm typu podziału i ograniczeń dla przypadku $c_{ab}=1$ (minimalizacja liczby przezbrojeń), Mitsumori [19] sformułował ulepszoną wersję obejmującą przypadek $c_{ab}=c_b$, natomiast Driscoll i Emmons [6] zaproponowali algorytm programowania dynamicznego dla ogólnej sytuacji, gdy koszty przezbrojeń zależą od kolejności. Ten ostatni, i równocześnie najlepszy z wymienionych, był z powodzeniem testowany dla problemów z n=60 i B=6, przy czym zauważono umiarkowany, liniowy wzrost czasu obliczeń dla rosnącego n oraz gwałtowny wzrost czasu przy rosnącym B.

3.7. Problem $1|s_{ab}=0,c_{ab}|\Sigma w_jC_j+\Sigma G_j$

Problem ten był rozpatrywany w pracy [1] przy założeniu, że B=n. Zauważmy, że gdy $w_j=0$, to jest on równoważny problemowi komiwojażera (patrz punkt 3.1); zatem należy on do klasy NP-trudnych. Jeżeli zaś $c_{ab}=0$, to mamy do czynienia z klasycznym zagadnieniem $1||\Sigma w_j C_j$, dla którego uporządkowanie zadań według niemalejących p_j/w_j daje minimum funkcji celu (reguła Smitha).

W [1] przedstawiono wyniki badań eksperymentalnych przeprowadzonych dla pewnej grupy algorytmów typu podziału i ograniczeń. i algorytmu hybrydowego wykorzystującego elementy programowania dynamicznego i metody podziału i ograniczeń. Badania wykazały wyższość podejścia hybrydowego, jednakże i ono pozwalało rozwiązywać, w rozsądnym czasie, problemy o wymiarach co najwyżej n=20. W [1] można znaleźć szczegóły tego algorytmu.

W [1] zbadano również przydatność dwóch algorytmów aproksymacyjnych, opartych na procedurze "najbliższy sąsiad" (NN), znanym algorytmie aproksymacyjnym dla problemu komiwojażera (patrz np. [7]). Oto szkic tego podejścia. Niech J_0 będzie zadaniem fikcyjnym o zerowym czasie wykonywania, które musi być wykonane jako pierwsze; c_{0j} jest kosztem początkowego przezbrojenia. Przyjmujemy, że zadania $J_1,\dots,J_n$ ponumerowane są zgodnie z niemalejącymi p_j/w_j. Niech dalej $0,1,\dots,n$ będą numerami miast w problemie komiwojażera, a $g(i,j)$, $0\le i,j\le n$, odległościami pomiędzy miastami. Algorytm NN, mając częściową trasę $\pi(1),\dots,\pi(k)$, gdzie $\pi(1)=0$, wybiera jako $\pi(k+1)$ miasto j°, dla którego $g(\pi(k),j)$ osiąga minimum na zbiorze $J\setminus\{\pi(1),\dots,\pi(k)\}=J\setminus P_k$. W pierwszym algorytmie, zaproponowanym w [1], stosuje się procedurę NN do problemu komiwojażera, w którym odległości $g(\pi(k),j)$, $j\in J\setminus P_k$, zdefiniowane są w następujący sposób:

$$g(\pi(k),j) = c_{\pi(k),j} + p_j(\Sigma_{i\in P_k,i<j} w_i) - w_j(\Sigma_{i\in P_k,i<j} p_i)$$

Zauważmy, że jeżeli $w_j=0$ dla $j\in J$, to otrzymujemy procedurę NN w problemie komiwojażera z $g(i,j)=c_{ij}$, natomiast gdy $c_{ij}=0$, $0\le i,j\le n$, to procedura ta działa tak samo jak reguła Smitha. Ostatnie stwierdzenie wynika z faktu, że zadania są ponumerowane zgodnie z niemalejącymi p_j/w_j. Mamy bowiem

$g(\pi(k),j)=0$, jeżeli $j=\min P_k$, oraz

$g(\pi(k),j)=\Sigma_{i\in P_k,i<j}(p_j w_i - w_j p_i)\ge 0$, jeżeli $j>\min P_k$,

przy czym w ostatnim wyrażeniu równość zachodzi wtedy i tylko wtedy, gdy $p_j/w_j=p_i/w_i$ dla każdego $i\in P_k$, $i<j$. Badania ekperymentalne [1] wykazały nadspodziewanie wysoką efektywność tego algorytmu. Średnie odchylenie od minimum wyniosło 2.49%. Drugi z rozpatrywanych algorytmów, pewna modyfikacja opisanego wyżej, okazał się jeszcze lepszy dając średnie odchylenie 0.66%. Wyniki te są zaskakujące, ponieważ analiza najgorszego przypadku dla NN daje następujący rezultat (patrz np. [7]): $K_{NN} \le \frac{1}{2}(\lceil\log_2 n\rceil+1)K^*$, gdzie K_{NN} i K^* oznaczają, odpowiednio, wartość funkcji celu otrzymaną w wyniku zastosowania NN i minimum funkcji celu, natomiast n jest liczbą miast.

3.8. Inne problemy jednomaszynowe

Lockett i Muhleman [18] rozpatrują problem minimalizacji sumy czasów przezbrojeń w specyficznym zagadnieniu, w którym B=n oraz czas przezbrojenia maszyny w celu wykonania zadania J_j zależy od kolejności zadań poprzedzających J_j, a nie tylko od zadania bezpośrednio poprzedzającego J_j. Zależność ta bierze się stąd, że maszyna wyposażona jest w stację narzędziową o skończonej pojemności. i do wykonania zadania wymagane jest wypełnienie stacji odpowiednim zestawem narzędzi. Niektóre narzędzia występują w zestawach odpowiadających wielu zadaniom, a ponadto zestaw nie musi wypełniać stacji w całości. Zatem kolejność zadań wykonywanych przed J_j wpływa na zawartość stacji przed rozpoczęciem wykonywania J_j i tym samym określa niezbędne zmiany narzędzi. W [18] przedstawiono algorytmy dokładne i aproksymacyjne oraz podano wyniki analiz ekperymentalnych.

Moore [21] zajmuje się zagadnieniem, w którym: B=n, występują czasy przezbrojeń s_{ab}, określone są przedziały $[r_j,\hat{d}_j]$, $j\in J$, każdemu zadaniu J_j przyporządkowana jest wartość $V_j>0$, oraz określony jest przedział czasu $[T_s,T_e]$ (okno). Należy znaleźć $J'\subset J$ oraz uszeregowanie $\alpha=(\alpha(1),\ldots,\alpha(k)$ zadań z J' takie, że czas rozpoczęcia $\alpha(1)$ nie jest mniejszy niż T_s, $C_{\alpha(k)}\le T_e$, każde zadanie $\alpha(i)$, $i=1,\ldots,k$, wykonywane jest w przedziale $[r_{\alpha(i)},\hat{d}_{\alpha(i)}]$ oraz $\Sigma_{j=1}^{n}x_jV_j$ osiąga maksimum, gdzie $x_j=1$, jeżeli $j\in J'$; $x_j=0$ w przeciwnym przypadku. Dla zagadnienia tego podano algorytm typu podziału i ograniczeń, który, jak wykazały badania eksperymentalne, rozwiązuje w krótkim czasie (rzędu 1 s.) problemy z $n\le 200$.

Wyniki badań nad podobnymi zagadnieniami, wybiegającymi swoimi sformułowaniami poza przyjęty w tym przeglądzie zakres tematyczny, oraz wyniki nieistotne przestawione są w pracach [4], [8], [15], [25], przy czym lista ta nie jest wyczerpująca.

4. Problemy wielomaszynowe

Literatura dotycząca wielomaszynowych problemów szeregowania z przezbrojeniami maszyn jest zdecydowanie uboższa. Przeważają prace, w których rozpatrywane są modele planowania produkcji w zadanym horyzoncie czasowym, często ujmowane jako zagadnienie całkowitoliczbowego programowania liniowego, na przykład prace [23], [24], [26]. W [9], pewien problem planowania produkcji na równoległych liniach produkcyjnych sprowadzony został do kwadratowego zagadnienia przydziału pracy.

4.1. Problemy $P2|pmtn,s_b|C_{max}$, $P2|s_b|C_{max}$, $P2|pmtn,s_b|\Sigma w_jC_j$, $P2|s_b|\Sigma w_jC_j$

W [20] pokazano, że dwumaszynowy problem z identycznymi, równoległymi maszynami, podzielnymi zadaniami i czasami przezbrojeń niezależnymi od kolejności jest NP-trudny; dla $s_b=0$ problem ten jest rozwiązywalny w wielomianowym czasie. Ten sam problem z niepodzielnymi zadaniami, $P2|s_b|C_{max}$, jest NP-trudny, nawet dla $s_b=0$ (patrz np.[12]). Stąd wynika, że odpowiednie problemy z funkcjami celu L_{max} i ΣU_j są również NP-trudne. Problem $P2|s_b|\Sigma w_jC_j$ jest NP-trudny dla $s_b=0$ [17], natomiast przynależność $1|pmtn,s_b|\Sigma w_jC_j$ do klasy problemów NP-trudnych wynika z faktu, że dla problemu tego istnieje uszeregowanie optymalne bez przerwań.

Brak jest w literaturze algorytmów dokładnych i aproksymacyjnych dla omawianej klasy problemów.

4.2. Problem $F2|s^i_{ab}|C_{max}$

Zarówno permutacyjna, jak i niepermutacyjna wersja dwumaszynowego problemu przepływowego z B=n jest NP-trudna, począwszy już od przypadku, gdy czasy przezbrojeń na jednej z maszyn są zerowe; są one wtedy równoważne problemowi komiwojażera [14]. Przypomnijmy, że gdy $s^i_{ab}=0$ dla każdego i, a, b, to problem ten jest rozwiązywalny w wielomianowym czasie (algorytm Johnsona). Monma i Potts [20] stwierdzili, że zachodzi następująca własność.

<u>Własność 4.1.</u> W problemie permutacyjnym $F2|s^i_{ab}|C_{max}$, jeżeli s^i_{ab} spełniają nierówność trójkąta dla i=1,2, to istnieje uszeregowanie optymalne, w którym zadania każdej grupy występują względem siebie w kolejności: J_j poprzedza J_k, jeżeli $\min(p_{1j},p_{2k}) \le \min(p_{2j},p_{1k})$ (reguła Johnsona).

Gupta i Darrow [14] przedstawili szereg metod aproksymacyjnych dla przypadku B=n, wykorzystujących warunki dominacji wzorowane na regule Johnsona. W jednym z tych algorytmów proponują poprawianie uszeregowania (otrzymanego w wyniku zastosowania "warunków Johnsona") poprzez zamianę kolejności sąsiednich zadań w sytuacji, kiedy zmniejsza to wartość funkcji celu; to ostatnie podejście okazało się w badaniach eksperymentalnych najbardziej efektywne , w sensie średniej odległości otrzymywanych wartości funkcji celu od minimum. Stwierdzają również, że zaproponowane algorytmy zachowują się dobrze, gdy czasy przezbrojeń nie przekraczają 1/10 sumy czasów wykonywania zadań, w sensie średnim. W [14] zaproponowano również algorytm typu podziału i ograniczeń, w którym dolne ograniczenia wylicza się poprzez relaksację jednej z maszyn. Ponieważ problemy zrelaksowane są także NP-trudne (problemy komiwojażera), przydatność tej metody ogranicza się do zagadnień o małych rozmiarach; rozwiązywano zagadnienia dla n=3,...,7.

Dla przypadku gdy B=n oraz $s^{1}_{ab}=0$ dla jednej z maszyn, w [5] (patrz również [2]) podano algorytm programowania dynamicznego; rozwiązywano zagadnienia o wymiarach do n=15.

5. Uwagi końcowe

- Problemy szeregowania zadań z przezbrojeniami maszyn stanowią bardzo trudną klasę zagadnień optymalizacji kombinatorycznej, niemniej jednak okazuje się, że i tu można znaleźć interesujące własności, ułatwiające konstrukcję algorytmów dokładnych i aproksymacyjnych. Ta dziedzina wydaje się być daleka od wyczerpania.
- Biorąc pod uwagę to, że problemy te są uogólnieniami klasycznych zagadnień szeregowania (przynależność tych ostatnich do klasy problemów NP-trudnych implikuje natychmiastowo przynależność do tej klasy problemów omawianych), to sporo kwestii związanych z przynależnością do klasy problemów NP-trudnych zostało już rozwiązanych. Brak jest wyników wyjaśniających wpływ parametru B na złożoność obliczeniową; z wyjątkiem pracy [22].
- Odczuwa się wyraźnie brak systematycznych badań nad algorytmami aproksymacyjnymi, przyjmujących analizę dokładności - analizę najgorszego przypadku, analizę probabilistyczną, jako zasadniczą metodę oceny jakości algorytmu. Jest to istotny niedostatek, szczególnie w świetle faktu, że proponowane algorytmy dokładne mają bardzo wąski zasięg, jeżeli chodzi o rozmiary rozwiązywanych problemów.

Praca została wykonana w ramach programu RP.I.02 "Teoria sterowania i optymalizacji ciągłych układów dynamicznych i procesów dyskretnych".

LITERATURA

[1] Barnes J.W., Vanston L.K., Scheduling Jobs with Linear Delay Penalties and Sequence Dependent Setup Costs, Ops Res. 29(1981), 146-159.

[2] Bellman R., Esogbue A.O., Nabeshima I., Mathematical Aspects of Scheduling and Applications, Pergamon Press, New York, 1982.

[3] Bruno J., Downey P., Complexity of Task Sequencing with Deadlines, Set-Up Times and Changeover Costs, SIAM J. Comput. 7(1978), 393-404.

[4] Buzacott J.A., Dutta S.K., Sequencing many jobs on multipurpose facili-

ty, Nav.Res.Logist.Q. 18(1971), 75-82.

[5] Corwin B.D., Esogbue A.O., Two-machine flowshop scheduling problems with sequence dependent setup times: A dynamic programming approach, Nav.Res.Logist.Q. 21(1974), 515-524.

[6] Driscoll W.C., Emmons H., Scheduling Production on One Machine with Changeover Costs, AIIE Trans. 9(1977), 188-395.

[7] Garey M.R.,Johnson D.S.: Computers and Intractability, W.H. Freeman and Company, San Francisco 1979.

[8] Gavett J.W.,Three heuristic rules for sequencing jobs to a single production facility, Mgmt Sci. 12(1965), B166-B176.

[9] Geoffrion A.M., Graves G.W., Scheduling Parallel Production Lines with Changeover Costs: Practical Application of a Quadratic Assignment/LP Approach, Ops Res. 24(1976), 595-610.

[10] Gilmore P.C., Gomory R.E., Sequencing a one state-variable machine: a solvable case of the travelling salesman problem, Ops Res. 12(1964), 655-679.

[11] Glassey C.R. Minimum changeover scheduling of several products on one machine, Ops Res. 16(1968), 342-352.

[12] Graham R.L. et al., Optimization and approximation in deterministic sequencing and scheduling: a survey, Ann. Discrete Math. 5(1979),287-326.

[13] Gupta J.N.D., Optimal Scheduling for Single Facility with Two Job Classes, Comput.& Ops Res. 11(1984), 409-413.

[14] Gupta J.N.D., Darrow W.P., The two-machine sequence dependent flowshop scheduling problem, European J. Operational Res. 24(1986), 439-446.

[15] Haynes R.D., Komar C.A., Byrd J. The effectiveness of three heuristic rules for job sequencing in a single production facility, Mgmt Sci. 19(1973), 575-580.

[16] Lawler E.L., Lenstra J.K., Rinnooy Kan A.H.G., Shmoys D.B. (ed.), The Traveling Salesman Problem. A Guidec Tour of Combinatorial Optimization, John Wiley and Sons, Chichester 1985.

[17] Lenstra J.K., Rinnooy Kan A.H.G., Brucker P., Complexity of Machine Scheduling Problems, Ann. Discrete Math. 1(1977), 343-363.

[18] Lockett A.G. Muhlemann A.P. A scheduling problem involving sequence dependent changeover costs, Ops Res. 20(1972), 895-902.

[19] Mitsumori S. Optimal production scheduling of multi-commodity in flow line, IEEE Trans. Systems Man and Cyber. SMC-2(1972),486-493.

[20] Monma C.L., Potts C.N., On the complexity of scheduling with batch set-up times, Research Report, University of Southampton 1987.

[21] Moore J.E. An Algorithm for a Single Machine Scheduling Problem with Sequence Dependent Setup Times and Scheduling Windows, AIIE Trans. 7(1975), 35-41.

[22] Richter K. The robot sequencing problem: polynomial algorithm and complexity, Optimization 16(1985),597-605.

[23] Tang C.S., Wittrock R.J., Parallel Machine Scheduling with Major and Minor Setups, Research Report RC 11412 9/10/85, IBM T.J. Watson Research Center, Yorktown Heights, 1985.

[24] Wittrock R.J. , Scheduling Parallel Machines with Setups, Research Report RC 11740 3/3/86, IBM T.J. Watson Resarch Center, Yorktown Heights, 1986.

[25] Sahney V., Single serever,two machine sequencing with switching times, Ops Res. 22(1972),24-36.

[26] Sumichrast R.T. Scheduling Parallel Processors: An Integer Linear Prgramming Based Heuristic for Minimizing Setup Time, Research Report, Virginia Polytechnic Institute and State University, Blacksburg, Virginia, 1987.

[27] S. Zdrzałka, Analiza najgorszego przypadku algorytmów aproksymacyjnych dla zagadnień szeregowania zadań z przezbrojeniami maszyn, Komunikat, Politechnika Wrocławska, Instytut Cybernetyki Technicznej, 1988.

Recenzent:Doc.dr hab.inż.J.Klamka

Wpłynęło do Redakcji do 1988-04-30.

ПРОБЛЕМЫ РАСПИСАНИЯ ЗАДАЧ С ПЕРЕНАЛАДКАМИ

Р е з ю м е

В работе рассмотрен класс задач расписания , в котором множество заданий разбито на группы (подмножества) . С переходом из задания данной группы к заданию инной группы связано время переналадки или стоимость переналадки. Приводится обзор проблем этого класса , алгоритмы и результаты анализа вычислительной сложности.

PROBLEMS OF SCHEDULING WITH SETUPS

S u m m a r y

We consider a class of scheduling problems in which a job set is partitioned into batches (subsets) and a setup time or setup cost are incurred whenever there is a switch from processing a job in one batch to a job in another batch. Problems of this class, their properties, algorithms and results of computational complexity analysis are surveyed.