ZESZYTY NAUKOWE POLITECHNIKI ŚLĄSKIEJ 2002
Seria: AUTOMATYKA z. 134 Nr kol. 1554

Stanisław GAWIEJNOWICZ, Wiesław KURC*, Lidia PANKOWSKA
Uniwersytet im. Adama Mickiewicza, Poznań

PRZEJŚCIA FAZOWE W SZEREGOWANIU ZADAŃ CZASOWO-ZALEŻNYCH

Streszczenie. W pracy rozważamy złożoność problemu szeregowania postaci $1 \mid p_j(t)=1+\alpha_j t \mid \|C\|_p$ jako funkcję parametru p. Dowodzimy istnienia takich liczb $p_0 < p_1$, że w kolejnych przedziałach: $(1, p_0]$, $(p_0 < p_1]$, $(p_1, +\infty)$ złożoność wyrażona jest odpowiednio wzorami: $O(2^n)$, $O(n^k + n\log n)$, $O(n\log n)$, gdzie $k=1,2,\ldots,O(f(n))$,. Wyniki te ujawniają wewnętrzną naturę badanego problemu dostarczając argumentów, że dla $1 \le p \le p_0$ należy oczekiwać złożoności $O(2^n)$. Zastosowaniem rezultatów są wielomianowe metody przybliżone dla kryterium $\|C\|_1 = \Sigma C_j$.

PHASE TRANSITIONS IN TIME-DEPENDENT SCHEDULING

Summary. In the paper we study the complexity of the linear time-dependent problem $1 \mid p_j(t)=1+\alpha_j t \mid \|C\|_p$ as a function of a parameter p. We show that there exist numbers $p_0 < p_1$ such that in the intervals $(1, p_0]$, $(p_0 < p_1]$, $(p_1, +\infty)$ the complexity is given by $O(2^n)$, $O(n^k + n\log n)$ and $O(n\log n)$, respectively, for $k=1,2,\ldots,O(f(n))$. Such approach explains intrinsic nature of the problem and gives a basis for construction of approximate algorithms in the case of $\|C\|_1=\Sigma C_j$.

1. Wprowadzenie

W pracy rozpatrujemy problem szeregowania zadań czasowo-zależnych, w przypadku liniowym, dla kryteriów normowych. Problem ten będzie oznaczany symbolicznie następująco

$$1 \mid p_j(t)=1+\alpha_j t \mid \|C\|_p \tag{1}$$

gdzie $\alpha_j > 0$ dla j = 0, 1, ... ,n, $C=[C_0,C_1,\ldots,C_n]^T$, t oznacza czas startu j-tego zadania, $C_0=1$, $C_j=C_{j-1}+p_j(C_{j-1})=1+a_jC_{j-1}$ oznaczają czasy zakończenia j-tego zadania, przy czym $a_j=1+\alpha_j$ dla $j=1,2,\ldots,n$. Symbol $\|C\|_p$ oznacza normę Höldera (inaczej l_p-normę) [10] wektora C, to znaczy $\|C\|_p = \left(\sum_{j=0}^{n}|C_j|^p\right)^{1/p}$ dla $1 \le p < +\infty$ oraz $\|C\|_\infty = \max\{|C_j|: j=0,\ldots,n\}$ dla $p = +\infty$. Powyższy problem szeregowania (1) polega na wyznaczeniu takiego uszeregowania n+1 zadań, lub takiej permutacji ciągu $\bar{\alpha}$, gdzie $\bar{\alpha} = (\alpha_0,\alpha_1,\ldots,\alpha_n)$, że wartość normy $\|C\|_p$ jest najmniejsza z możliwych.

Problem szeregowania zadań czasowo-zależnych ma już bogatą literaturę. Szczególny przypadek rozpatrywanego problemu dotyczący kryterium C_{max}, czyli problem $1 \mid p_j(t)=\beta_j+\alpha_j t \mid \|C\|_\infty$, gdzie α_j, $\beta_j > 0$ dla $j = 0,1,..n$, jest rozwiązywalny w czasie $O(n\log n)$ poprzez takie uszeregowanie zadań, że ułamki α_j/β_j tworzą ciąg nierosnący ([1], [3], [9], [14]). Znane są także wyniki dotyczące problemu (1) dla kryterium ΣC_j, tzn. $1 \mid p_j(t)=1+\alpha_j t \mid \|C\|_1$ ([2], [4], [6], [8], [11], [12], [13]; por. też [5] i [7]). Jednak nadal nie jest znany wielomianowy algorytm rozwiązujący problem (1) dla kryterium $\|C\|_1=\Sigma C_j$. Co więcej, nie ma jak dotąd dowodu na to, że taki algorytm nie istnieje. W [12] dokonano dwóch istotnych spostrzeżeń. Po pierwsze, warunkiem koniecznym na to, aby ciąg $\bar{\alpha} = (\alpha_0,\alpha_1,\ldots,\alpha_n)$ tworzył optymalne uszeregowanie dla kryterium $\|C\|_1=\Sigma C_j$, jest to, aby był V-ciągiem ([4], [12]). Po drugie, najlepszą strategią, także w przypadku ciągu optymalnego, jest, aby zadanie odpowiadające maksymalnemu elementowi w ciągu było przetwarzane w pierwszej kolejności. Oznacza to, że bez zmniejszenia ogólności zawsze możemy przyjmować, że α_0 jest największym elementem ciągu α i rozpatrywać jedynie permutacje pozostałych elementów ciągu $\bar{\alpha}$ w celu wyznaczenia optymalnego uszeregowania zadań.

W pracy przedstawiamy metodę, która wypełniając lukę dotyczącą statusu problemu (1) dla pośrednich kryteriów normowych, czyli dla norm Höldera $\|\cdot\|_p$, gdzie $1 < p < +\infty$, dostarcza dodatkowych informacji wspierających hipotezę o wykładniczej złożoności problemu (1) dla kryterium $\|C\|_1=\Sigma C_j$. Przy okazji rozwijamy analityczne środki badania rozpatrywanego problemu oraz wskazujemy na metodę konstrukcji przybliżonych

algorytmów o złożoności wielomianowej dla kryterium $\|C\|_1=\Sigma C_j$. Metody te przenoszą się na ogólniejszy przypadek $1 \mid p_j = \beta_j + \alpha_j t \mid \|C\|_\varphi$ dla tzw. norm modularnych $\|C\|_\varphi$ w R^{n+1}.

W dalszym ciągu wygodniej będzie zastąpić tradycyjne oznaczenia i pisać x_j zamiast C_j, x w miejsce symbolu C oraz przedstawić problem (1) w postaci matematycznej, którą nazywać będziemy kanoniczną. Dzięki możliwości ustalenia pierwszego elementu w ciągu $\bar{\alpha} = (\alpha_0,\alpha_1,\ldots,\alpha_n)$ jako $max_j\{a_j\}$ (por. [12]) i rozpatrywania permutacji tylko pozostałych elementów przyjmijmy, że $a = (a_1, a_2, \ldots ,a_n)$, gdzie $a_j = 1 + \alpha_j$, $a_j > 1$, dla $j = 1, 2, \ldots ,n$. Problem (1) można zatem zredukować do zagadnienia wyznaczania takiej permutacji $\pi^* \in \Pi_n$, że ([4])

$$\|x(a_\pi^*)\|_p = min\{\|x(a_\pi)\|_p : \pi \in \Pi_n\}, \tag{2}$$

przy czym a_π oznacza tutaj permutację π ciągu $a = (a_1, a_2, \ldots ,a_n)$, wektor $x(a_\pi)$ jest rozwiązaniem układu równań $A(a_\pi)\, x(a_\pi) = d(1)$, gdzie $d(1)=[1,1,\ldots,1]^T \in R^{n+1}$, a macierz $A(a)$, dla ciągu $a = (a_1, a_2, \ldots ,a_n)$, zdefiniowana jest następująco

$$A(a) = \begin{bmatrix} 1 & 0 & \ldots & 0 & 0 \\ -a_1 & 1 & \ldots & 0 & 0 \\ 0 & -a_2 & \ldots & 0 & 0 \\ \ldots & \ldots & \ldots & \ldots & \ldots \\ 0 & 0 & \ldots & -a_n & 1 \end{bmatrix}.$$

W dalszym ciągu zarówno permutację optymalną, jak i dowolną, choć ustaloną, permutację wyjściowego ciągu a oznaczać będziemy tą samą literą a, co nie doprowadzi do nieporozumień. Nietrudno zauważyć, że macierz $A^{-1}(a) = [c_{i,j}]$ istnieje, przy czym $c_{i,j}=0$ dla $i<j$ oraz $c_{i,j}=a_j \ldots a_{i-1}$ dla $i \geq j$ gdzie $i=1,\ldots,n$ oraz $j=1,\ldots,n+1$. Jest to macierz unimodalna, podczas gdy macierz $A(a)$ jest M-macierzą. Wektor $x(a) = [x_0(a), x_1(a),\ldots,x_n(a)]^T$ stanowi tylko inne oznaczenie wektora $C = [C_0, C_1,\ldots,C_n]^T$ z problemu (1). Układ równań $A(a)\, x(a) = d(1)$ jest macierzowym zapisem związku rekurencyjnego $x_0=1$, $x_1=a_1x_0+1$, $x_2=a_2x_1+1$, ..., $x_n=a_nx_{n-1}+1$ ze sformułowania (1) problemu określającego czasy zakończenia kolejnych zadań w rozpatrywanym porządku. Ponieważ $1=x_0(a)\leq x_1(a)\leq\ldots\leq x_n(a)$, więc $\|x(a)\|_\infty = x_n(a)$.

2. Podstawowe rezultaty

Niech $b = a(a_q \leftrightarrow a_{q+1})$ oznacza wyjściowy ciąg a = (a_1, a_2, ... , a_n), w którym przestawiono elementy a_q oraz a_{q+1}. Niech x(b) oznacza rozwiązanie układu A(b) x(b) = d(1).

Lemat 1 ([4],[6]). Niech $q \in \{1,...,n-1\}$. *Wówczas dla* $i = 1,...,n$ *oraz* $n \geq 2$

$$x_i(b) - x_i(a) = \begin{cases} (a_q - a_{q+1})a_{q+2}...a_i & ,1 \leq q \leq i, \\ (a_{q+1} - a_q)\sum_{j=0}^{q-1} a_{j+1}...a_{q-1} & ,q = i, \\ 0 & ,q > i. \end{cases}$$

Ponieważ dla $1 \leq p < +\infty$ norma Höldera $\|\cdot\|_p$ jest funkcją różniczkowalną we wnętrzu dodatniego stożka w R^{n+1}, dlatego można zastosować twierdzenie o wartości średniej. Mamy $\|y\|_p = \|x\|_p + \nabla\|x^\theta\|_p$ (y-x), gdzie $x,y \in R^{n+1}$, x > 0 i y > 0, $x^\theta = \theta x + (1-\theta)y$ dla pewnego $\theta \in (0,1)$ natomiast $\nabla\|x^\theta\|_p$ oznacza gradient normy $\|\cdot\|_p$ w punkcie x^θ. Ponieważ w rozpatrywanym przypadku $x = x(a) > 0$ oraz $y = x(b) > 0$, więc także $x_i^\theta(a) > 0$ dla j=0,1,...n. Ostatecznie możemy otrzymać wzór Taylora w postaci ([4], [6]) $\|x(b)\|_p = \|x(a)\|_p + \nabla\|x^\theta(a)\|_p$ (x(b) - x(a)), gdzie $x^\theta(a) = \theta x(a) + (1-\theta)x(b)$ dla pewnego $\theta \in (0,1)$ oraz $\nabla\|x^\theta(a)\|_p = \left[...,\left(\frac{x_j^\theta(a)}{\|x^\theta(a)\|_p}\right)^{p-1},...\right]$. Stąd wynika ważny dla naszych rozważań lemat ([4], [6], [7]).

Lemat 2. *Niech* $A(a)x(a) = d(1)$, $A(b)x(b) = d(1)$ *oraz* $b = a(a_q \leftrightarrow a_{q+1})$. *Wówczas dla wszystkich* $1 \leq p < +\infty$

$$\|x(b)\|_p - \|x(a)\|_p = \left(\frac{x_j^\theta(a)}{\|x^\theta(a)\|_p}\right)^{p-1}(a_{q+1} - a_q)\cdot\Delta(a,p,q),$$

gdzie a_1, a_2, ... ,a_{q-1} *są zawarte w pierwszej sumie,* a_{q+2},... ,a_n *są zawarte w drugiej sumie w wyrażeniu* $\Delta(a,q,p) = \sum_{j=0}^{q-1} a_{j+1}...a_{q-1} - \sum_{i=q+1}^{n}\left(\frac{x_j^\theta(a)}{x_q^\theta(a)}\right)^{p-1} a_{q+2}...a_i$. *Elementy* a_q, *i* a_{q+1} *występują tylko w* $(a_q - a_{q+1})$. *Jeśli* $p = +\infty$, *to* $\|x(b)\|_p - \|x(a)\|_p = (a_q - a_{q+1})\cdot a_{q+2}...a_n$.

Jako wniosek otrzymujemy twierdzenie stanowiące pierwszy etap naszych rozważań.

Twierdzenie 1 ([4],[6],[7]). *Niech $1 \le p < +\infty$ oraz niech a jest uporządkowaniem wyjściowego ciągu realizującym minimum dla kryterium $\|x(a_\pi)\|_p$ $(\pi \in \Pi_n)$ w rozpatrywanym zagadnieniu (2). Wówczas dla każdego $q=1,2,\ldots,n-1$ istnieje $\theta \in (0,1)$ takie, że albo*

(i) $a_{q+1} - a_q \le 0$ *oraz* $\Delta(a,q,p) \le 0$ *albo*

(ii) $a_{q+1} - a_q \ge 0$ *oraz* $\Delta(a,q,p) \ge 0$,

gdzie wyrażenie $\Delta(a,q,p)$ zostało zdefiniowane w Lemacie 2.

Dowód. Skoro $\|x(a)\|_p$ osiaga najmniejszą z możliwych wartości, to dla dowolnej transpozycji $b = a(a_q \leftrightarrow a_{q+1})$ w ciągu a otrzymujemy $\|x(b)\|_p$ - $\|x(a)\|_p \ge 0$. Teza wynika wprost z Lematu 2.

W dalszym ciągu niezbędne są jednostajne oszacowania ułamków postaci $\dfrac{x_j^\theta(a)}{x_q^\theta(a)}$ występujących w wyrażeniu $\Delta(a,q,p)$. Zachodzi następujący Lemat.

Lemat 3 ([4],[6]). *Istnieje $\sigma_n(a) > 1$, niezależne od θ, j oraz q $(j \ge q)$ takie, że spełniona jest nierówność*

$$1 + \sigma_n^{-1}(a) \le \frac{x_j^\theta(a)}{x_q^\theta(a)} \le \sigma_n(a).$$

W szczególności można przyjąć, że $\sigma_n(a) = \max_{\pi \in \Pi_n} \|x(a_\pi)\|_p$, czyli maksymalną z możliwych wartości rozpatrywanego kryterium uszeregowania.

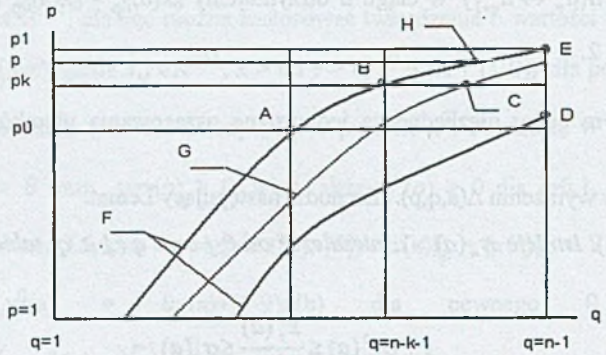
Na mocy tych oszacowań oraz Twierdzenia 1 otrzymujemy wynik dotyczący istnienia wspomnianych wyżej liczb $1 < p_0 < p_1 < +\infty$, stanowiących punkty przejść fazowych (rys. 1).

Twierdzenie 2 ([4],[6],[7]). *Istnieją liczby p_0, p_1 spełniające warunek $1 < p_0 < p_1 < +\infty$ takie, że: (i) dla wszystkich $p_1 < p \le +\infty$ (skończonych lub nie) problem (2) dla kryterium $\|\cdot\|_p$ może być rozwiązany w czasie $O(n\log n)$; (ii) dla wszystkich $1 \le p \le p_0$ problem (2) dla kryterium $\|\cdot\|_p$ może być rozwiązany w czasie $O(2^n)$ (lub w czasie wielomianowym, o ile taki algorytm istnieje).*

Aby dokładniej zbadać złożoność problemu (2) dla kryterium $\|\cdot\|_p$ przy $p_0 < p \le p_1$, należy dokładniej zbadać zachowanie się funkcji $\Delta(a,q,p)$, uwzględniając wszystkie możliwe

permutacje ciągu a = (a_1, a_2, ... ,a_n) oraz wszelkie nie znane a priori wartości parametru $\theta \in [0,1]$. Z uwagi na obecność wielu technicznych lematów przedstawimy metodę postępowania wykorzystując poniższy szkic (rys. 1). Na osi poziomej odłożono wartości parametru q=1,2,...,n-1 związanego z transpozycjami $b = a(a_q \leftrightarrow a_{q+1})$ optymalnego dla kryterium $\|\cdot\|_p$ ciągu a. Zwracamy uwagę na trudność polegającą na tym, że mówiąc o permutacji optymalnej jeszcze nie wiemy, jaka konkretnie wartość parametru p jest rozpatrywana. Na osi pionowej odłożono wartości parametru $1 \le p < +\infty$.

Krzywa G obrazuje miejsca zerowe funkcji $(q,p) \to \Delta(a_\pi, q, p)$ dla konkretnej permutacji $\pi \in \Pi_n$, podczas gdy wyraźniej zaznaczone krzywe F obrazują górne i dolne obwiednie tych miejsc zerowych. Ich przebieg jest już niezależny od $\theta \in [0,1]$.



Rys. 1. Obwiednie miejsc zerowych funkcji D(a,q,p)
Fig. 1. Envelopes of zeros of the functions D(a,q,p)

Obwiednie te wyznaczamy obliczając explicite miejsca zerowe dla minoranty i majoranty funkcji $(q,p) \to \Delta(a_\pi, q, p)$. Minoranty i majoranty tej funkcji uzyskujemy na podstawie oszacowań z Lematu 3. Można pokazać, że te obwiednie można wyznaczyć punktowo w sposób dokładny w czasie O(n log n), poprzez odpowiedni wybór permutacji wyjściowego ciągu a ([7]). Skoro tak jest, to możemy taką wartość p_k górnej obwiedni wyznaczyć w punkcie q=n-k+1, odległym o k punktów (k transpozycji) od q = n-1 (k=0,1,2,...). Zwracamy uwagę, że biorąc teraz $p_k \le p \le p_1$ oraz odpowiadające mu kryterium normowe $\|\cdot\|_p$ w (2) możemy przecinać (dzięki opisanej konstrukcji) zbiór miejsc zerowych funkcji $(q,p) \to \Delta(a_\pi, q, p)$, a zatem i zbiór miejsc zerowych dla przypadku optymalnej permutacji π, w ten sposób, że na prawo od punktu H (rys. 1) pozostaje zawsze nie więcej punktów niż w przypadku punktu B - czyli co najwyżej k punktów. Zwracamy

także uwagę, że liczbę tych punktów k wybieramy a priori i dlatego będzie ona swoistym regulatorem złożoności dla wspomnianych wyżej kryteriów normowych w (2).

Wniosek ten jest kluczowy dla naszej metody. Wobec Twierdzenia 1 fakt przecinania linią poziomą miejsc zerowych funkcji $(q,p) \to \Delta(a_\pi, q, p)$ dla optymalnej permutacji π ciągu a oznacza, że na lewo od tego punktu przecięcia (to znaczy dla q mniejszych) funkcja $\Delta(a_\pi, q, p)$ przyjmuje wartości ujemne, co wobec optymalności ciągu $a = a_\pi$, a więc i nierówności $\|x(b)\|_p - \|x(a)\|_p \geq 0$, pociąga konkluzję, iż ciąg optymalny $a = a_\pi$ dla tych wartości q musi być malejący. Odwrotnie, na prawo od tego punktu przecięcia (dla q większych) $\Delta(a_\pi, q, p)$ przyjmuje wartości dodatnie, a więc ponownie wobec optymalności rozpatrywanego ciągu (tj. wobec $\|x(b)\|_p - \|x(a)\|_p \geq 0$, przy wszystkich transpozycjach $b = a(a_q \leftrightarrow a_{q+1})$) na mocy Twierdzenia 1 otrzymujemy, że druga część optymalnego ciągu $a = a_\pi$ musi być niemalejąca. Oznacza to, że optymalna permutacja musi być V-ciągiem.

W rozpatrywanym jednak przypadku, ustalając k=0,1,2,... i rozpatrując kryterium normowe $\|\cdot\|_p$ w (2), dla $p_k \leq p \leq p_1$, doprowadzamy do tego, że wynikający optymalny V-ciąg będzie miał prawą gałąź utworzoną z co najwyżej k=0,1,2,... elementów. Będziemy takie ciągi nazywać V_k-ciągami. Oznacza to, że dla kryterium $\|\cdot\|_p$ w (2) ($p_k \leq p \leq p_1$) wystarczy wyznaczyć wszystkie V_k-ciągi. Takie V_k-ciągi można wyznaczyć w czasie $O(n^k + n\log n)$. Istotnie, najpierw porządkujemy wyjściowy ciąg a w porządku nierosnącym, a następnie kosztem $O(n^k)$ operacji wyznaczamy wymagane V_k-ciągi. Stąd wynika wspomniana wyżej złożoność. Otrzymujemy ostatecznie następujące istotne uzupełnienie Twierdzenia 2.

Twierdzenie 3 ([7]). *Niech dany będzie ciąg $a = (a_1, a_2, \ldots, a_n)$ taki, że $a_1 > 1$ oraz rozważmy liczbę naturalną k taką, że $q_k = n - k + 1$ jest dostatecznie bliskie n-1. Wówczas istnieją takie liczby $1 < p_0 < p_1 < +\infty$, które wyznaczyć można w czasie O(nlogn), że*

(i) *jeśli $1 \leq p \leq p_0$, to problem (2) dla kryterium $\|\cdot\|_p$ może być rozwiązany w czasie co najwyżej $O(2^n)$,*

(ii) *jeśli $p_k \leq p \leq p_1$, to problem (2) dla kryterium $\|\cdot\|_p$ może być rozwiązany w czasie $O(n^k + n\log n)$,*

(iii) *jeśli $p_1 < p \leq +\infty$ (p skończone lub nie), to problem (2) dla kryterium $\|\cdot\|_p$ może być rozwiązany w czasie O(nlogn).*

3. Uwagi końcowe

Założenie, że $q_k = n - k + 1$ jest dostatecznie bliskie n-1, oznacza na rys. 1, że dolna obwiednia F przecina oś q pomiędzy q=1 i q=n-1 i że rozpatrujemy tylko ten zakres wartości q, dla których ta obwiednia jest dodatnia. Można udowodnić ([7]), że funkcja $(q, p) \to \Delta(a_\pi, q, p)$ jest dodatnia dla q=n-1 przy dowolnej permutacji π. A zatem założenie to zawsze jest spełnione. Z drugiej strony praktyczne znaczenie mają tylko małe wartości parametru k, gdyż oznaczają mniejszą złożoność algorytmu polegającego na wyznaczeniu wszystkich V_k-ciągów.

Jeżeli parametr p dla kryterium $\|\cdot\|_p$ w zagadnieniu (2) zmniejsza swoją wartość, począwszy od p = $+\infty$ do wartości p = 1, to złożoność problemu jest stała do pewnego momentu i wynosi O(nlogn). Jeśli p jest nie większe od p_1, to przy dalszym zmniejszaniu p złożoność powiększa się zgodnie ze wzorem $O(n^k + n\log n)$ dla k=1,2,.... Jeśli k jest niezależne od n, mamy oczywiście do czynienia ze złożonością wielomianową. Jednak nie można wykluczyć, że warunek: "$q_k = n - k + 1$ jest dostatecznie bliskie n-1" jest spełniony nawet wtedy, gdy $k = 1,2,\ldots,O(f(n))$, gdzie np. $f(n) = \left[\frac{n}{2}\right]$. Oznacza to, że w końcu (z dokładnością do odpowiedniego czynnika, zależnego od n) dochodzimy do złożoności $O(2^n)$. Powyższa obserwacja dotycząca możliwości przyjmowania przez k wartości zależnych od n jest argumentem na to, iż problem (2), a zatem i problem (1), dla kryterium $\|C\|_1 = \Sigma C_j$ ma aktualnie przestrzeń poszukiwań rozwiązań optymalnych o mocy rzędu $O(2^n)$. Oczywiście nie wyklucza to istnienia algorytmu wielomianowego rozwiązującego ten problem.

Przedstawione rozważania (por. też [7]) umożliwiają konstrukcję wielomianowych algorytmów przybliżonych (zamiast normy $\|\cdot\|_1$ rozpatrujemy normy Höldera $\|\cdot\|_p$) o złożoności $O(n^k + n\log n)$, z regulatorem tej złożoności $k=1,2,\ldots,O(f(n))$ poprzez konstrukcję wszystkich V_k-ciągów.

*) Adres do korespondencji: Uniwersytet im. A. Mickiewicza, Wydział Matematyki i Informatyki, 61-614 Poznań, Umultowska 87. e-mail: wkurc@main.amu.edu.pl

LITERATURA

1. Browne S., Yechiali U.: Scheduling deteriorating jobs on a single processor. Operations Research, 38, 1990, s. 495-498.

2. Chen Z-L.: Parallel machine scheduling with time dependent processing times. Discrete Applied Mathematics, 70, 1996, s. 81-93.(Erratum: Discrete Applied Mathematics, 75, 1996, p.103).

3. Gupta J.N.D., Gupta S.K.: Single facility scheduling with nonlinear processing times. Computers and Industrial Engineering 14, 1988, s. 387-393.

4. Gawiejnowicz S., Kurc W., Pankowska L., Suwalski C.: Approximate solution of time-dependent scheduling problem for l_p-norm based criteria. In: Fleischmann B. et al. (eds), Operations Research OR2000. Springer, Berlin, 2001, s. 372-377.

5. Gawiejnowicz S., Kurc W., Pankowska L.: Bicriterion approach to a single machine time-dependent scheduling problem. In: Chamoni P. et al. (eds.), Operations Research Proceedings 2001. Springer, Berlin, 2002, s. 199-206.

6. Gawiejnowicz S., Kurc W., Pankowska L.: Approximate O(n log n) solution of linear time-dependent scheduling problem subject to the total completion time minimization. Report No. 113, 2001, Faculty of Mathematics and Computer Science, Poznań, Poland.

7. Gawiejnowicz S., Kurc W., Pankowska L.: A Linear Deterioration and Phase Transitions. Report No. 114, 2002, Faculty of Mathematics and Computer Science, Poznań, Poland.

8. Gawiejnowicz S., Lai T-C, Chiang M-H.: Polynomially solvable cases of scheduling deteriorating jobs to minimize total completion time. In: Brucker P., et al. (eds.): Seventh International Workshop on Project Management and Scheduling. Extended Abstracts, University of Osnabrück, 2000, s. 131-134.

9. Gawiejnowicz S., Pankowska L.: Scheduling jobs with varying processing times. Information Processing Letters, 54, 1995, s. 175-178.

10. Hirsch F., Lacombe G.: Elements of Functional Analysis. Springer, Berlin 1999.

11. Kononov A.: Scheduling problems with linear increasing processing times. In: Zimmermann U. et al. (eds), Operations Research 1996. Springer, Berlin 1997, s.208-212.

12. Mosheiov G.: V-shaped policies for scheduling deteriorating jobs. Operations Research, 39, 1991, s. 979-991.

13. Mosheiov G.: Scheduling jobs under simple linear deterioration. Computers and Operations Research, 21, 1994, s. 653-659.

14. Tanaev V.S., Gordon V.S., Shafransky Y.M.: Scheduling Theory. Single-stage Systems. Kluwer, Dordrecht 1994.

Recenzent: Dr hab. inż. Czesław Smutnicki

Abstract

In this paper (cf. [4], [7]) we study a time dependent scheduling problem for norm-based criteria. The problem can be denoted shortly as

$$1 \mid p_j(t)=1+\alpha_j t \mid \|C\|_p \tag{1}$$

where $\alpha_j>0$, $C = [C_0,C_1,\ldots,C_n]$, $C_0\equiv 1$, $C_j=C_{j-1}+p_j(C_{j-1})=1+a_jC_{j-1}$ with $a_j=1+\alpha_j$ ($a_j > 1$) and $j=0, 1,2,\ldots,n$ where t is the starting time of a job. The optimization is carried over all permutations of a given sequence $\alpha = (\alpha_0,\ldots,\alpha_n)$. It is well known ([1],[3],[9]) that a single machine scheduling problem $1|\, p_j(t)=\beta_j+\alpha_j t|\, \|C\|_\infty$, where α_j, $\beta_j>0$ for $j=1,2,\ldots,n$, is solvable in $O(n\log n)$ time by scheduling the jobs in an nonincreasing order of α_j/β_j ratios. Hence, to get the minimum of the completion time $\|C\|_\infty=C_n$ in (1) it is sufficent to arrange the sequence $\alpha = (\alpha_0,\ldots,\alpha_n)$ in an nonincreasing way, which can be done in $O(n\log n)$ time. However, an efficient (i.e. polynomial-time) algorithm for solving the problem (1) with $\|C\|_1=\Sigma C_j$ is not known.

Our first goal in this paper is to fill the gap betwen the opposite norm criteria $\|C\|_\infty$ and $\|C\|_1$. We call this shortly a "phase transition" since the intermediate norms $\|\cdot\|_p$ link quite different states of the complexity: polynomial for $\|C\|_\infty$ and the (actually recognized) complexity $O(2^n)$ for $\|C\|_1$ criteria. Our main result in this direction is the following theorem.

Theorem. *Let $a=(a_1,\ldots,a_n)$ be such that $a_i>1$. Let moreover a natural number k be such that $q_k = n-k+1$ is sufficiently close to $n-1$. Then there exist numbers $1 < p_0 < p_1 < +\infty$, which can be determined in $O(n\log n)$ time, such that*

(i) *if $1 \le p \le p_0$, then the problem (1) can be solved in at most $O(2^n)$ time,*
(ii) *if $p_k \le p \le p_1$, then the problem (1) can be solved in $O(n^k+n\log n)$ time,*
(iii) *if $p_1 < p \le +\infty$, then the problem (1) can be solved in $O(n\log n)$ time.*

Next, since we cannot exclude the case when $k=0,1,2,\ldots,\left[\frac{n}{2}\right]$, we conclude that the complexity of the problem (1) can increase according to the formula $O(n^k+n\log n)$ to $O(2^n)$ (clearly up to some, dependent on n, coefficient). This is our argument on the final complexity (i.e. $O(2^n)$) of the problem (1) with $\|C\|_1=\Sigma C_j$.

Finally, by careful study of envelopes of prescribed functions related to our norm-related measures, we obtain a family of approximate but polynomial-time algorithms solving the problem (1) with $\|C\|_1=\Sigma C_j$. These algorithms are based on the construction of V-shaped sequences with k-elements (k=0,1,2,...) right branch. In this way we obtain $O(n^k+n\log n)$ element set of such V-shaped sequence in which optimal solution can be easily identified.