УЧЕНЫЕ ЗАПИСКИ КАЗАНСКОГО ГОСУДАРСТВЕННОГО УНИВЕРСИТЕТА
Том 150, кн. 4 Физико-математические науки *2008*

УДК 519.854

ПСЕВДОПОЛИНОМИАЛЬНЫЙ ПРИБЛИЖЕННЫЙ АЛГОРИТМ РЕШЕНИЯ NP-ПОЛНОЙ ЗАДАЧИ МИНИМИЗАЦИИ МАКСИМАЛЬНОГО ВРЕМЕННОГО СМЕЩЕНИЯ

О.Н. Шульгина, Н.К. Щербакова

Аннотация

В статье предлагается и обосновывается приближенный алгоритм псевдополиномиальной трудоемкости для решения известной NP-полной в сильном смысле задачи теории расписаний – минимизации максимального временного смещения для одного прибора при запрещении прерываний в обслуживании требований. Получена оценка абсолютной погрешности значения целевой функции расписания, построенного с помощью предложенного алгоритма.

Ключевые слова: расписание, временное смещение, псевдополиномиальный алгоритм, NP-полнота, трудоемкость.

Введение

Одной из известных задач теории расписаний является задача минимизации максимального временного смещения для одного прибора. Указанная задача является NP-полной [1] в сильном смысле, то есть не существует псевдополиномиального алгоритма ее решения в предположении, что классы P и NP не совпадают. Если допускаются прерывания в обслуживании требований, то задача минимизации максимального временного смещения разрешима за полиномиальное время [2–4]. Алгоритмы трудоемкости $O(n \log n)$ решения задачи в случае одновременно поступающих требований или одинаковых директивных сроков были предложены в [4, 5]. Получено полиномиальное решение задачи в случае одинаковых продолжительностей обслуживания требований [3, 6, 7]. В [8] разработан и обоснован алгоритм псевдополиномиальной трудоемкости решения NP-полного [9, с. 293] частного случая задачи, когда требования можно перенумеровать одновременно по неубыванию директивных сроков и невозрастанию моментов поступления. Этот алгоритм используется для построения приближенного метода решения задачи, предложенного и обоснованного в данной статье.

1. Постановка задачи и обозначения

На одном приборе не ранее момента времени t необходимо обслужить n требований. Пронумеруем требования числами $1, 2, \ldots, n$ и в дальнейшем будем говорить об обслуживании требований множества $N = \{1, 2, \ldots, n\}$. Запрещаются одновременное обслуживание более одного требования и прерывания при обслуживании требований. Для каждого требования j, $j \in N$, заданы следующие параметры: момент поступления требования на обслуживание r_j; продолжительность обслуживания $p_j \geq 0$; желательный (директивный) срок завершения обслуживания d_j. Числа t, r_j, p_j, d_j являются целыми. Под расписанием будем понимать некоторую перестановку элементов любого подмножества множества N. Будем обозначать через $\Pi(N', t')$ множество всех расписаний обслуживания требований множества $N' \subseteq N$ с момента времени $t' \geq t$. Расписание обслуживания требований любого подмножества множества $N' \subseteq N$ будем называть частичным на множестве N'.

Пусть $\pi = (j_1, j_2, \ldots, j_{n'})$ – некоторое расписание из множества $\Pi(N', t')$, где $n' = |N'|$ – количество элементов в множестве N, j_k – номер требования, которое обслуживается k-м по порядку при расписании π. Момент $t_{j_k}(\pi)$ завершения обслуживания требования j_k, $k = 1, \ldots, n'$, находится следующим образом: $t_{j_1}(\pi) = \max\{t', r_{j_1}\} + p_{j_1}$; $t_{j_k}(\pi) = \max\{t_{j_{k-1}}(\pi), r_{j_k}\} + p_{j_k}$, $k = 2, 3, \ldots, n'$. Обозначим через $L_j(\pi)$ временное смещение требования $j \in N'$ при расписании π, то есть $L_j(\pi) = t_j(\pi) - d_j$.

Пусть $\pi^* \in \Pi(N', t')$ – расписание, при котором функция

$$F(\pi) = \max_{j \in N'} L_j(\pi), \quad \pi \in \Pi(N', t') \tag{1}$$

достигает минимального значения на множестве $\Pi(N', t')$. Тогда расписание π^* будем называть оптимальным на множестве $\Pi(N', t')$. Если $N' = \oslash$, то полагаем $F(\pi) = -\infty$, $\pi \in \Pi(N', t')$, а расписание на пустом множестве будем обозначать через $\pi^{\oslash}$. Таким образом, задача минимизации максимального временного смещения для одного прибора заключается в отыскании расписания, оптимального на множестве $\Pi(N, t)$.

Введем необходимые в дальнейшем обозначения. Пусть $N' \subseteq N$, $N' \neq \oslash$, $t' \geq t$, $\pi \in \Pi(N', t')$. Положим

$$r_{\min}(N') = \min_{i \in N'} r_i;$$

$$r_{\max}(N') = \max_{i \in N'} r_i;$$

$$p_{\max}(N') = \max_{i \in N'} p_i;$$

$$T(\pi) = \max_{j \in N'} t_j(\pi);$$

$$J^*(\pi) = \{j \in N' : F(\pi) = L_j(\pi)\};$$

$$J_d(N') = \{j \in N' : d_j = \min_{i \in N'} d_i\};$$

$\Pi^*(N', t') = \{\pi^* \in \Pi(N', t') : F(\pi^*) = \min\limits_{\pi \in \Pi(N', t')} F(\pi)\}$ – множество оптимальных расписаний на множестве $\Pi(N', t')$;

$\vec{\Pi}_r(N', t')$, $\vec{\Pi}_d(N', t')$ – множества расписаний обслуживания требований множества N' с момента времени t', составленных в порядке неубывания моментов поступления требований и в порядке неубывания директивных сроков завершения обслуживания требований соответственно.

Если требование i предшествует требованию j, $i \neq j$, при расписании π, то это соотношение номеров будем обозначать через $i \xrightarrow{\pi} j$. Запись $i \xrightarrow{\pi} \bar{N}$, где $\bar{N} \subseteq N'$, $i \notin \bar{N}$, означает, что $i \xrightarrow{\pi} j$ для любого $j \in \bar{N}$, а запись $\bar{N} \xrightarrow{\pi} \bar{\bar{N}}$, где $\bar{N}, \bar{\bar{N}} \subseteq N'$, $\bar{N} \cap \bar{\bar{N}} = \oslash$, означает, что для всех пар i, j таких, что $i \in \bar{N}$, $j \in \bar{\bar{N}}$, выполняется соотношение $i \xrightarrow{\pi} j$.

2. Вспомогательные результаты

Пусть обслуживаемые требования можно перенумеровать так, чтобы

$$d_1 \leq \cdots \leq d_n, \quad r_1 \geq \cdots \geq r_n. \tag{2}$$

Опишем процедуру h_0 построения такого расписания $\pi_{h_0} \in \Pi(N,t)$, что значение $F(\pi_{h_0})$ будет отличаться от оптимального значения $F(\pi^*)$, $\pi^* \in \Pi^*(N,t)$, не более, чем на величину $p_{\max}(N)$ [10], то есть

$$F(\pi_{h_0}) - F(\pi^*) \leq p_{\max}(N). \tag{3}$$

Процедура h_0 [10]. Перенумеруем требования множества N таким образом, чтобы соблюдались неравенства (2).

Полагаем $\bar{t} = \max\{r_n, t\}$, $N_1 = \{1\}$, $P_1 = \max\{r_1, t\} + \sum_{j \in N} p_j - p_1$, $\pi_1^i = (1)$, $\pi_1^i \in \Pi(N_1, i)$ для всех $i = \bar{t}, \ldots, P_1$.

Пусть уже известны N_k, P_k, π_k^i для всех $i = \bar{t}, \ldots, P_k$, и $1 \leq k < n$. Полагаем

$$N_{k+1} = N_k \cup \{k+1\}, \quad P_{k+1} = P_k - p_{k+1}.$$

Для всех $i = \bar{t}, \ldots, P_{k+1}$ строим

$$\pi_i' = (k+1, \pi_k^{\max\{r_{k+1}, i\} + p_{k+1}}), \quad \pi_i'' = (\pi_k^i, k+1), \quad \pi_i''' = (k+1, \pi_k^i),$$

$$\pi_i', \pi_i'', \pi_i''' \in \Pi(N_{k+1}, i), \quad \Pi_i = \{\pi \in \{\pi_i', \pi_i'', \pi_i'''\} : F(\pi) = \min_{\bar{\pi} \in \{\pi_i', \pi_i'', \pi_i'''\}} F(\bar{\pi})\},$$

$$\pi_{k+1}^i = \arg\min\{T(\pi) | \pi \in \Pi_i\}.$$

При $k = n$ полагаем $\pi_{h_0} = \pi_k^{\bar{t}}$, и процесс заканчивается.

Трудоемкость процедуры h_0 составляет $O(n^2 P)$ операций [10].

Пусть γ – вещественное число. Расписание $\pi \in \Pi(N,t)$ будем называть допустимым относительно γ, если $F(\pi) \leq \gamma$. Опишем процедуру, которая строит допустимое относительно заданного значения γ расписание $\pi_h \in \Pi(N,t)$ либо устанавливает, что такого расписания не существует [11].

Процедура h [11]. Перенумеруем требования множества N таким образом, чтобы соблюдались неравенства (2). Полагаем $\bar{t} = \max\{r_n, t\}$, $N_1 = \{1\}$, $P_1 = \max\{r_1, t\} + \sum_{j \in N} p_j - p_1$, $\pi_1^i = \pi^{\oslash}$, если $\max\{r_1, i\} + p_1 - d_1 > \gamma$, и $\pi_1^i = (1)$, если $\max\{r_1, i\} + p_1 - d_1 \leq \gamma$ для всех $i = \bar{t}, \ldots P_1$.

Пусть $1 \leq k < n$ и известны N_k, P_k, π_k^i для всех $i = \bar{t}, \ldots, P_k$. Полагаем $N_{k+1} = N_k \cup \{k+1\}$, $P_{k+1} = P_k - p_{k+1}$. Далее, для каждого $i = \bar{t}, \ldots, P_{k+1}$ строим расписания $\pi_i', \pi_i'', \pi_{k+1}^i \in \Pi(N_{k+1}, i)$ следующим образом.

Полагаем $\pi_i' = \pi^{\oslash}$, если $F(k+1, \pi_k^{\max\{r_{k+1}, i\} + p_{k+1}}) > \gamma$, и $\pi_i' = (k+1, \pi_k^{\max\{r_{k+1}, i\} + p_{k+1}})$ в противном случае, $\pi_i'' = \pi^{\oslash}$, если $F(\pi_k^i, k+1) > \gamma$, и $\pi_i'' = (\pi_k^i, k+1)$ в противном случае.

Полагаем $\Pi_i = \{\pi \in \{\pi_i', \pi_i''\} : \pi \neq \pi^{\oslash}\}$, $\pi_{k+1}^i = \pi^{\oslash}$, если $\Pi_i = \oslash$, и $\pi_{k+1}^i = \arg\min\{T(\pi) | \pi \in \Pi_i\}$, если $\Pi_i \neq \oslash$.

При $k = n$ полагаем $\pi_h = \pi_k^{\bar{t}}$, и процесс заканчивается.

Трудоемкость процедуры h составляет $O(nP)$ операций [11].

Алгоритм 1 [8]. Полагается $\gamma^0 = F(\pi_{h_0})$. При помощи процедуры h, где $\gamma = \gamma^0$, строится допустимое относительно γ^0 расписание $\pi_h^0 \in \Pi(N,t)$.

Пусть уже построено расписание π_h^{k-1}, и $k \geq 1$. Тогда расписание π_h^k строится следующим образом. Полагается $\gamma^k = \gamma^{k-1} - 1$, с помощью процедуры h, где $\gamma = \gamma^k$, строится допустимое относительно γ^k расписание $\pi_h^k \in \Pi(N,t)$. Если $\pi_h^k = \pi^{\oslash}$, то полагается $\pi^* = \pi_h^{k-1}$, и алгоритм заканчивает работу.

Теорема 1 [8]. *Пусть на параметры требований множества N накладываются ограничения* (2). *Тогда алгоритмом* 1 *трудоемкости $O(n^2 P + n p_{\max} P)$ будет построено расписание, оптимальное на множестве $\Pi(N,t)$.*

3. Алгоритм решения

Идея предлагаемого в статье алгоритма заключается в следующем. Директивные сроки требований изменяются так, чтобы новые параметры требований удовлетворяли ограничениям (2). Далее задача решается с помощью алгоритма 1. Полученное расписание является приближенным расписанием с минимальным значением границы абсолютной погрешности оптимального значения целевой функции (1) на множестве оптимальных расписаний частного случая при условиях (2). Минимизация границы абсолютной погрешности обеспечивается использованием алгоритма изменения директивных сроков.

Пусть $\pi^*, \pi' \in \Pi(N,t)$ – оптимальные расписания обслуживания требований множества N с момента времени t при директивных сроках d_j и $d^{'}_j$ соответственно, то есть $\max\limits_{j\in N} L_j(\pi^*) = F(\pi^*)$, $\max\limits_{j\in N} L_j(\pi') = \min\limits_{\pi\in\Pi(N,t)} \max\limits_{j\in N}\{t_j(\pi) - d^{'}_j\}$. Очевидно, что $F(\pi') \geq F(\pi^*)$.

Теорема 2 [12, с. 36]. *Для задач минимизации максимального временного смещения с директивными сроками обслуживания требований d_j и $d^{'}_j$, $j \in N$, имеет место $F(\pi') \leq F(\pi^*) + \rho$, где $\rho = \max\limits_{j\in N}\{d_j - d^{'}_j\} - \min\limits_{j\in N}\{d_j - d^{'}_j\}$.*

Из теоремы 2 следует, что значение целевой функции (1) оптимального расписания задачи с директивными сроками $d^{'}_j$, $j \in N$, отличается от значения целевой функции (1) оптимального расписания задачи с директивными сроками d_j, $j \in N$, не более, чем на ρ. Поэтому если можно подобрать директивные сроки требований $d^{'}_j$ так, чтобы задача с новыми параметрами требований решалась эффективным алгоритмом, то полученное расписание будет приближенным для общей задачи с оценкой абсолютной погрешности оптимального значения целевой функции (1), не превышающей ρ.

Задачу минимизации границы абсолютной погрешности ρ можно поставить как задачу математического программирования.

Не ограничивая общности, будем полагать, что требования множества N пронумерованы так, что

$$r_1 \leq r_2 \leq \cdots \leq r_n, \tag{4}$$

причем

$$r_j = r_{j+1} \Rightarrow d_j \geq d_{j+1} \quad \forall\, j = 1,\ldots,n-1. \tag{5}$$

Рассмотрим задачу:

$$\min_{d^{'}_1, d^{'}_2, \ldots, d^{'}_n} \left(\max_{j\in N}\{d_j - d^{'}_j\} - \min_{j\in N}\{d_j - d^{'}_j\} \right) \tag{6}$$

$$d^{'}_1 \geq d^{'}_2 \geq \cdots \geq d^{'}_n, \tag{7}$$

Решение задачи (6), (7) может быть найдено при помощи следующего алгоритма.

Алгоритм 2. Перенумеруем требования множества N согласно (4), (5). Полагаем $N_1 = N$.

Пусть уже построено множество N_k, $k \geq 1$. Если $N_k = \oslash$, то алгоритм заканчивает работу. В противном случае находим

$$l_k = \max\{j \in N_k : d_j = \max_{i\in N_k} d_i\}, \tag{8}$$

$\bar{N}_k = \{j \in N_k : j \leq l_k\}$, и полагаем $\bar{d}^{'}_j = d_{l_k}$ для всех $j \in \bar{N}_k$, $N_{k+1} = N_k \setminus \bar{N}_k$.

Лемма 1. *Решение задачи* (6), (7) *отыскивается с помощью алгоритма 2 трудоемкости* $O(n\log n)$ *операций.*

Доказательство. Оценим трудоемкость алгоритма 2. Для перенумерации требований понадобится $O(n\log n)$ операций [13, гл. 5, § 5.3]. Поскольку $\bar N_k \cap \bar N_{k+1} = \oslash$, то для нахождения $\bar d_j^{'}$ для всех $j \in N$ потребуется не более $O(n)$ операций. Следовательно, трудоемкость алгоритма 2 составляет $O(n\log n)$ операций.

Пусть число итераций алгоритма 2 равно m. Очевидно, что согласно алгоритму 2 значения $\bar d_j^{'}$ для всех $j = 1,\dots,n$ удовлетворяют ограничениям (7), то есть

$$\bar d_1^{'} \ge \dots \ge \bar d_n^{'}, \tag{9}$$

и, кроме того,

$$\bar d_j^{'} \ge d_j, \quad j = 1,2,\dots,n. \tag{10}$$

Покажем, что

$$\max_{j\in N}\{\bar d_j^{'} - d_j\} - \min_{j\in N}\{\bar d_j^{'} - d_j\} \le \max_{j\in N}\{d_j^{'} - d_j\} - \min_{j\in N}\{d_j^{'} - d_j\} \tag{11}$$

для любых целых чисел $d_j^{'}$, удовлетворяющих (7). Тогда утверждение леммы будет доказано. Для этого выберем произвольно целые значения $\tilde d_j^{'}$, $j = 1,\dots,n$, такие, что

$$\tilde d_1^{'} \ge \dots \ge \tilde d_n^{'}, \tag{12}$$

и проверим справедливость неравенства (11) при $d_j^{'} = \tilde d_j^{'}$ для всех $j = 1,\dots,n$.

Очевидно, что $\max\limits_{j\in N}\{d_j - d_j^{'}\} = -\min\limits_{j\in N}\{d_j^{'} - d_j\}$ и $\min\limits_{j\in N}\{d_j - d_j^{'}\} = -\max\limits_{j\in N}\{d_j^{'} - d_j\}$ для всех $d_1^{'},\dots,d_n^{'}$. Отсюда

$$\max_{j\in N}\{d_j - \bar d_j^{'}\} - \min_{j\in N}\{d_j - \bar d_j^{'}\} = \max_{j\in N}\{\bar d_j^{'} - d_j\} - \min_{j\in N}\{\bar d_j^{'} - d_j\} \tag{13}$$

$$\max_{j\in N}\{d_j - \tilde d_j^{'}\} - \min_{j\in N}\{d_j - \tilde d_j^{'}\} = \max_{j\in N}\{\tilde d_j^{'} - d_j\} - \min_{j\in N}\{\tilde d_j^{'} - d_j\}. \tag{14}$$

Поскольку требования множества N пронумерованы согласно (4), (5), то требования $l_1,\dots,l_m$ и множества $\bar N_1,\dots,\bar N_m$ можно выбрать с помощью алгоритма 2. Очевидно, что $\bar N_k \cap \bar N_{k+1} = \oslash$ и $\bar N_1 \cup \dots \cup \bar N_m = N$. Отсюда

$$\max_{j\in N}\{\bar d_j^{'} - d_j\} - \min_{j\in N}\{\bar d_j^{'} - d_j\} = \max_{1\le k\le m}(\max_{j\in \bar N_k}\{\bar d_j^{'} - d_j\}) - \min_{1\le k\le m}(\min_{j\in \bar N_k}\{\bar d_j^{'} - d_j\}) \tag{15}$$

$$\max_{j\in N}\{\tilde d_j^{'} - d_j\} - \min_{j\in N}\{\tilde d_j^{'} - d_j\}) = \max_{1\le k\le m}(\max_{j\in \bar N_k}\{\tilde d_j^{'} - d_j\}) - \min_{1\le k\le m}(\min_{j\in \bar N_k}\{\tilde d_j^{'} - d_j\}). \tag{16}$$

Обозначим через $s_k \in \bar N_k$ такое требование, что

$$d_{s_k} \le d_j \quad \forall\, j \in \bar N_k, \quad k = 1,\dots,m. \tag{17}$$

Согласно алгоритму 2

$$\bar d_j^{'} = \bar d_{l_k}^{'} = d_{l_k} \quad \forall\, j \in \bar N_k, \quad k = 1,\dots,m. \tag{18}$$

Согласно выбору (8) требования l_k с учетом (17), (18)

$$\max_{j\in\bar{N}_k}\{\bar{d}_j^{'} - d_j\} = d_{l_k} - d_{s_k} \quad \forall\, k = 1,\ldots,m, \tag{19}$$

$$\min_{j\in\bar{N}_k}\{\bar{d}_j^{'} - d_j\} = 0 \quad \forall\, k = 1,\ldots,m, \tag{20}$$

и $d_{l_k} \geq d_j$ для всех $j \in \bar{N}_k$. Кроме того, в силу (12) $\tilde{d}_{l_k}^{'} \leq \tilde{d}_j^{'}$ для всех $j \in \bar{N}_k$, $k = 1,\ldots,m$. Отсюда

$$\min_{j\in\bar{N}_k}\{\tilde{d}_j^{'} - d_j\} = \tilde{d}_{l_k}^{'} - d_{l_k} \quad \forall\, k = 1,\ldots,m. \tag{21}$$

Пусть числа C_j, $j = 1,\ldots,n$, таковы, что $\tilde{d}_j^{'} = \bar{d}_j^{'} + C_j$. Тогда с учетом (18)

$$\tilde{d}_j^{'} = d_{l_k} + C_j \quad \forall\, j \in \bar{N}_k, \quad k = 1,\ldots,m, \tag{22}$$

а из (12), (22) следует, что

$$i \leq j \Rightarrow C_i \geq C_j \quad \forall\, i,j \in \bar{N}_k, \quad k = 1,\ldots,m. \tag{23}$$

Так как $\max\limits_{j\in\bar{N}_k}\{\tilde{d}_j^{'} - d_j\} \geq \tilde{d}_j^{'} - d_j$ для всех $j \in \bar{N}_k$, то $\max\limits_{j\in\bar{N}_k}\{\tilde{d}_j^{'} - d_j\} \geq \tilde{d}_{s_k}^{'} - d_{s_k}$. Отсюда с учетом (22)

$$\max_{j\in\bar{N}_k}\{\tilde{d}_j^{'} - d_j\} \geq d_{l_k} + C_{s_k} - d_{s_k} \quad \forall\, k = 1,\ldots,m. \tag{24}$$

Из (19)–(21), (24) следует, что

$$\max_{j\in\bar{N}_k}\{\bar{d}_j^{'} - d_j\} - \min_{j\in\bar{N}_k}\{\bar{d}_j^{'} - d_j\} - (\max_{j\in\bar{N}_k}\{\tilde{d}_j^{'} - d_j\} - \min_{j\in\bar{N}_k}\{\tilde{d}_j^{'} - d_j\}) \leq \tilde{d}_{l_k}^{'} - d_{l_k} - C_{s_k}$$

для всех $k = 1,\ldots,m$. Согласно выбору (8) требований l_k и s_k имеем, что $s_k \leq l_k$. Тогда из (22) с учетом (23) следует, что $\tilde{d}_{l_k}^{'} \leq d_{l_k} + C_{s_k}$ для всех $k = 1,\ldots,m$. Поэтому

$$\max_{j\in\bar{N}_k}\{\bar{d}_j^{'} - d_j\} - \min_{j\in\bar{N}_k}\{\bar{d}_j^{'} - d_j\} \leq \max_{j\in\bar{N}_k}\{\tilde{d}_j^{'} - d_j\} - \min_{j\in\bar{N}_k}\{\tilde{d}_j^{'} - d_j\} \quad \forall\, k = 1,\ldots,m$$

и в силу (15), (16)

$$\max_{j\in N}\{\bar{d}_j^{'} - d_j\} - \min_{j\in N}\{\bar{d}_j^{'} - d_j\} \leq \max_{j\in N}\{\tilde{d}_j^{'} - d_j\} - \min_{j\in N}\{\tilde{d}_j^{'} - d_j\}.$$

Отсюда с учетом (13), (14) следует неравенство (11) при $d_j^{'} = \tilde{d}_j^{'}$. Лемма доказана. □

Опишем приближенный алгоритм решения общей задачи минимизации максимального временного смещения.

Алгоритм 3. Для требований множества N при помощи алгоритма 2 находим новые директивные сроки $\bar{d}_j^{'}$, $j \in N$. Далее с помощью алгоритма 1 решаем задачу с условиями 2 при $d_j = \bar{d}_j^{'}$.

Теорема 3. *При помощи алгоритма 3 трудоемкости $O(n^2P + np_{\max}P)$ операций строится расписание $\pi' \in \Pi(N,t)$ такое, что $F(\pi') - F(\pi^*) \leq \bar{\rho}$, где $\pi^* \in \in \Pi^*(N,t)$,*

$$\bar{\rho} = \max_{j\in N}\{d_j - \bar{d}^{'}_j\} - \min_{j\in N}\{d_j - \bar{d}^{'}_j\},$$

причем $\bar{\rho}$ – минимальное значение разности

$$\max_{j\in N}\{d_j - d^{'}_j\} - \min_{j\in N}\{d_j - d^{'}_j\}$$

по всем $d^{'}_1, \ldots, d^{'}_n$, удовлетворяющих неравенствам (7).

Доказательство. Из теоремы 2 имеем, что $F(\pi') \leq F(\pi^*) + \rho$, где $\pi^*, \pi' \in \in \Pi(N,t)$ – оптимальные расписания обслуживания требований множества N с момента времени t при директивных сроках d_j и $d^{'}_j$ соответственно. Из леммы 1 следует, что минимальное значение ρ достигается при $d^{'}_j = \bar{d}^{'}_j$, $j = 1, \ldots n$.

Трудоемкость алгоритма 2 не превышает $O(n \log n)$ операций, а трудоемкость алгоритма 1 – $O(n^2P + np_{\max}P)$ операций, поэтому трудоемкость алгоритма 3 составляет $O(n^2P + np_{\max}P)$ операций. Теорема доказана. □

Таким образом, изменив директивные сроки исходной задачи и решая полученную задачу с помощью алгоритма 1 за $O(n^2P + np_{\max}P)$ операций, получим приближенное решение исходной задачи с минимальной величиной границы абсолютной погрешности оптимального значения целевой функции (1) на множестве оптимальных расписаний задачи при условиях (2).

Алгоритм 3 показал неплохие результаты при экспериментальном исследовании. Из 1000 проведенных экспериментов с размерностями $3 \leq n \leq 10$ в 220 примерах было построено оптимальное расписание, в оставшихся 780 случаях $F(\pi_{A3})/F(\pi^*) < 1.06$, где $\pi_{A3} \in \Pi(N,t)$ построено алгоритмом 3, $\pi^* \in \Pi^*(N,t)$. Минимальное значение отношения теоретически известной величины абсолютной погрешности ρ к практически полученному значению составило 1.3, максимальное – 2.

Summary

O.N. Shulgina, N.K. Sherbakova. Pseudopolynomial Approximation Algorithm for Solving the NP-Complete Problem of Minimizing Maximum Lateness.

The article states and proves pseudopolynomial complexity approximation algorithm for solving the scheduling theory known as NP-complete problem, namely minimizing maximum lateness on a single machine, interruption in job processing being banned. The bound value absolute error of criterion function for schedule constructed by algorithm is received.

Key words: schedule, lateness, pseudopolynomial algorithm, NP-complete, complexity.

Литература

1. *Bruker P., Lenstra J.K., Rinnooy Kan A.H.G.* Complexity of machine scheduling problems. – Amsterdam: Math. Cent. Afd. Math. Beslisk, 1975, BW 43. – 29 p.

2. *Лебединская Н.Б.* Минимизация максимального отклонения в случае прерывания работ // Зап. науч. семинаров. Ленингр. отд. Матем. ин-та АН СССР. – 1978. – С. 117–124.

3. *Horn W.A.* Some simple scheduling algorithms // Nav. Res. Log. Quart. 21. – 1974. – No 1. – P. 177–185.

4. *Lageweg B.J., Lenstra J.K., Rinnooy Kan A.H.G.* Minimizing maximum lateness on one machine: computational experience and some applications // Statist. Neer. – 1976. – No 1. – P. 25–41.

5. *Jackson J.R.* Scheduling a production line to minimize maximum tardiness // Res. Report 43, Manag. Sci. Res. Project. – Los Angeles: Univ. of California, 1955.

6. *Frederickson G.N.* Scheduling unit-time tasks with integer release times and deadlines // Inform. Process. Lett. – 1983. – V. 16, No 4. – P. 171–173.

7. *Simons B.A.* A fast algorithm for single processor scheduling // 19th Annu. Symp. Found. Comput. Sci., Ann. Arbor, Mich. – N. Y., 1978. – P. 246–252.

8. *Шульгина О.Н.* Точный псевдополиномиальный алгоритм решения одной NP-трудной задачи теории расписаний // Исслед. по прикл. матем. и информ. – Казань: Казан. гос. ун-т, 2004. – Вып. 25. – С. 148–151.

9. *Танаев В.С., Гордон В.С., Шафранский Я.М.* Теория расписаний. Одностадийные системы. – М.: Наука, 1984. – 384 с.

10. *Шульгина О.Н., Щербакова Н.К.* Об одном приближенном алгоритме решения NP-трудной задачи теории расписаний // Исслед. по прикл. матем. и информ. – Казань: Казан. гос. ун-т, 2003. – Вып. 24. – С. 146–155.

11. *Шульгина О.Н.* Процедура построения допустимого расписания для задачи минимизации максимального временного смещения // Исслед. по прикл. матем. и информ. – Казань: Изд-во Казан. матем. об-ва, 2001. – Вып. 23. – С. 150–158.

12. *Лазарев А.А.* Эффективные алгоритмы решения некоторых задач теории расписаний для одного прибора с директивными сроками обслуживания требований: Дис. ... канд. физ.-мат. наук. – Казань, 1989. – 108 с.

13. *Кнут Д.* Искусство программирования для ЭВМ. Т. 3: Сортировка и поиск. – М.: Мир, 1973. – 348 с.

Поступила в редакцию
01.10.08

Шульгина Оксана Николаевна – кандидат физико-математических наук, доцент кафедры экономической кибернетики Казанского государственного университета.

E-mail: *ONSHUL@mail.ru*, *Oksana.Shulgina@ksu.ru*

Щербакова Наталья Казбековна – кандидат физико-математических наук, доцент кафедры экономической кибернетики Казанского государственного университета.

E-mail: *nata6060@mail.ru*