ИЗВЕСТИЯ РАН. ТЕОРИЯ И СИСТЕМЫ УПРАВЛЕНИЯ, 2006, № 6, с. 1–8

СИСТЕМНЫЙ АНАЛИЗ И ИССЛЕДОВАНИЯ ОПЕРАЦИЙ

УДК 519.854.2

ПАРЕТО-ОПТИМАЛЬНОЕ МНОЖЕСТВО NP-ТРУДНОЙ ЗАДАЧИ МИНИМИЗАЦИИ МАКСИМАЛЬНОГО ВРЕМЕНИ

© 2006 г. **А. А. Лазарев**

Москва, ВЦ РАН

Поступила в редакцию 18.05.06 г.

Рассматривается классическая NP-трудная в сильном смысле задача теории расписаний $1|r_j|L_{\max}$. Найдены новые свойства оптимальных расписаний. Выделен полиномиально-разрешимый случай задачи, когда моменты поступлений (r_j), продолжительности обслуживания (p_j) и директивные сроки завершения обслуживания (d_j) требований удовлетворяют ограничениям: $d_1 \le \ldots \le d_n$;$d_1 - r_1 - p_1 \ge \ge \ldots \ge d_n - r_n - p_n$. Алгоритм трудоемкости $O(n^3 \lg n)$ находит Парето-оптимальное множество расписаний по критериям $L_{\max}$ и $C_{\max}$, содержащее не более n вариантов.

Введение. Будем рассматривать следующую задачу теории расписаний. На одном приборе с момента времени t (момент освобождения прибора) необходимо обслужить требования множества $N = \{1, 2, \ldots, n\}$. Запрещается одновременное обслуживание и прерывания при обслуживании требований. Для требований множества N имеем: r_j – минимально возможный момент начала обслуживания, $p_j > 0$ – продолжительность обслуживания, d_j – директивный срок завершения обслуживания требования $j \in N$.

Каждая перестановка π требований множества N однозначно определяет раннее расписание, при котором исключаются искусственные простои прибора. При раннем расписании каждое требование $j \in N$ начинает обслуживаться сразу после окончания обслуживания предыдущего требования в соответствующей перестановке. Если время окончания обслуживания предыдущего требования меньше времени поступления на обслуживание (r_j) текущего требования, то начало обслуживания требования j откладывается до момента поступления. В дальнейшем все рассматриваемые расписания будут подразумеваться ранними. Множество всевозможных ранних расписаний, соответствующих множеству N, обозначим как $\Pi(N)$. Отметим, что число расписаний $\pi \in \Pi(N)$ равняется числу перестановок из n элементов, т. е. $|\Pi(N)| = n!$.

Через $c_j(\pi)$ будем обозначать момент завершения обслуживания требования $j \in N$ при расписании $\pi \in \Pi(N)$. Если $\pi = (j_1, j_2, \ldots, j_n)$, тогда

$$c_{j_1}(\pi) = \max\{r_{j_1} t\} + p_{j_1},$$

$$c_{j_k}(\pi) = \max\{c_{j_{k-1}}(\pi), r_{j_k}\} + p_{j_k}, \; k = 2, \ldots, n.$$

Зависимость $L_j(\pi) = c_j(\pi) - d_j$ отражает временное смещение требования $j \in N$ при расписании π. Максимальное временное смещение требований множества N при расписании π определяется как

$$L_{\max}(\pi) = \max_{j \in N}\{c_j(\pi) - d_j\}.$$

Обозначим момент завершения обслуживания всех требований множества N при расписании π через

$$C_{\max}(\pi) = \max_{j \in N} c_j(\pi).$$

В случае, когда будет рассматриваться расписание π с момента времени $t' > t$, будем использовать обозначения $L_{\max}(\pi, t')$, $C_{\max}(\pi, t')$, $c_j(\pi, t')$.

Задача состоит в нахождении оптимального расписания $\pi^* \in \Pi(N)$, которому соответствует минимальное значение целевой функции

$$L_{\max}(\pi^*) = \max_{j \in N} L_j(\pi) = \max_{j \in N}\{c_j(\pi) - d_j\}. \quad (0.1)$$

В общепринятой нотации теории расписаний Грэхэма и др. [1] данная задача записывается как $1|r_j|L_{\max}$. Интенсивные работы над ее решением продолжаются с начала 50-х годов XX в. В [2] показано, что общий случай задачи $1|r_j|L_{\max}$ является *NP*-трудным в сильном смысле.

Был выделен ряд полиномиально разрешимых случаев задачи, начиная с раннего результата Джексона [3] для варианта $r_j = 0$, $\forall j \in N$, когда решение – расписание, при котором требования упорядочены по неубыванию директивных сроков завершения обслуживания (по правилу EDD). Такое расписание также будет оптимальным для случая, когда времена поступления и директивные сроки согласованы ($r_i \le r_j \Leftrightarrow d_i \le d_j$, $\forall i, j \in N$).

Поттс [4] представил итерационную версию расширенного правила Джексона (*IJ*) и доказал, что $L_{\max}(IJ)/L^*_{\max} \le \dfrac{3}{2}$. Холл и Шмойс [5] модифицировали итерационную версию и создали алгоритм (*MIJ*), который гарантирует $L_{\max}(MIJ)/L^*_{\max} \le \frac{4}{3}$. Также они предложили две аппроксимационные схемы, гарантирующие нахождение ε-приближенного решения за $O(n \lg n + n(1/\varepsilon)^{O(1/\varepsilon^2)}$ и $O((n/\varepsilon)^{O(1/\varepsilon)})$ операций. Мастролилли [6] разработал улучшенную аппроксимационную схему, выполняющуюся за время $O(n + (1/\varepsilon)^{O(1/\varepsilon)})$.

Специальные случаи $1|\,\text{prec}; r_j|C_{\max}$, $1|\text{prec}; p_j = p; r_j|L_{\max}$ и $1|\text{prec}; r_j; \text{pmtn}|L_{\max}$ с ограничениями предшествования при обслуживании требований были рассмотрены в [7–9]. Хогевен [10] обосновал полиномиальный алгоритм (трудоемкость $O(n^2 \lg n)$ операций) для специального случая, в котором параметры требований удовлетворяют ограничениям $d_j - p_j - A \le r_j \le d_j - A$ для некоторой константы A, $\forall j \in N$. Псевдополиномиальный алгоритм для NP-сложного случая, когда времена поступления и директивные сроки расположены в обратном порядке ($d_1 \le \ldots \le d_n$ и $r_1 \ge \ldots \ge r_n$), был разработан в [11, 12], трудоемкость алгоритма $O(nP(n + p_{\max}))$, где $P = \sum_{j \in N} p_j$ и $p_{\max} = \max_{j \in N} p_j$. Результаты, представленные в настоящей статье, являются обобщением решений, полученных в [13].

Кроме задачи $1|r_j|L_{\max}$ в данной работе рассматривается задача построения Парето-оптимального множества расписаний по критериям $C_{\max}$ и $L_{\max}$. Будет сформулирован алгоритм построения множества расписаний $\Phi(N, t) = \{\pi'_1, \pi'_2, \ldots, \pi'_m\}$, для которых

$$C_{\max}(\pi'_1) < C_{\max}(\pi'_2) < \ldots < C_{\max}(\pi'_m),$$

$$L_{\max}(\pi'_1) < L_{\max}(\pi'_2) < \ldots < L_{\max}(\pi'_m).$$

1. Свойства задач. Обозначим предшествование требования i требованию j при расписании π как $(i \longrightarrow j)_\pi$. Введем также

$$r_j = \max\{r_j, t\} \quad \forall j \in N, \tag{1.1}$$

$$r(N, t) = \min_{j \in N}\{r(t)\}. \tag{1.2}$$

В тех случаях, когда очевидно, о каком множестве требований идет речь, будем записывать вместо $r(N, t)$ просто $r(t)$.

Пусть π – расписание обслуживания требований некоторого подмножества из N, для определения которого в дальнейшем используется обозначение $\{\pi\}$.

Предполагается, что параметры требований удовлетворяют ограничениям

$$d_1 \le \ldots \le d_n,\ d_1 - r_1 - p_1 \ge \ldots \ge d_n - r_n - p_n. \tag{1.3}$$

Например, данным ограничениям соответствует случай, когда $d_j = r_j + p_j + z$, $j = 1, \ldots, n$, где z – константа, т. е. когда все требования имеют одинаковый запас времени до своего директивного срока.

Допустим $|N| > 1$ и t – момент освобождения прибора. Выделим из множества N два требования $f = f(N, t)$ и $s = s(N, t)$ следующим образом:

$$f(N, t) = \operatorname*{argmin}_{j \in N}\{d_j | r_j(t) = r(N, t)\}, \tag{1.4}$$

$$s(N, t) = \operatorname*{argmin}_{j \in N\setminus f}\{d_j | r_j(t) = r(N\setminus f, t)\}, \tag{1.5}$$

где $f = f(N, t)$. Если $N = \{i\}$, то полагаем $f(N, t) := i$, $s(N, t) := 0$, $\forall t$. Также определим $d_0 = +\infty$, $f(\emptyset, t) = 0$, $s(\emptyset, t) = 0$, $\forall t$. Для требований f и s выполняются следующие свойства.

Л е м м а 1. Если для требований множества N справедливо (1.3), то при любом расписании $\pi \in \Pi(N)$ для всех $j \in N\setminus\{f\}$, для которых $(j \longrightarrow f)_\pi$ имеет место

$$L_j(\pi) < L_f(\pi) \tag{1.6}$$

и для всех $j \in N\setminus\{f, s\}$, отвечающих условию $(j \longrightarrow s)_\pi$,

$$L_j(\pi) < L_s(\pi) \tag{1.7}$$

где $f = f(N, t)$ и $s = s(N, t)$

Д о к а з а т е л ь с т в о. Для всех требований j, таких, что $(j \longrightarrow f)_\pi$, имеем $c_j(\pi) < c_f(\pi)$. Если $d_j \ge d_f$, то, очевидно

$$L_j(\pi) = c_j(\pi) - d_j < c_f(\pi) - d_f = L_f(\pi),$$

поэтому справедливо (1.6).

Пусть для требования $j \in N$, $(j \longrightarrow f)_\pi$, имеем $d_j < d_f$. Тогда $r_j > r_f$. Если бы $r_j \le r_f$, то и $r_j(t) \le r_f(t)$, и $r_f(t) = r(t)$, как следует из (1.1) и (1.4). Тогда $r_j(t) = r(t)$ и $d_j < d_f$, но это противоречит определению требования f (1.4). Поэтому $r_j > r_f$. Очевидно, что $c_j(\pi) - p_j < c_f(\pi) - p_f$, и так как $r_j > r_f$, то

$$c_j(\pi) - p_j - r_j < c_f(\pi) - p_f - r_f,$$

$$c_j(\pi) - c_f(\pi) < p_j + r_j + p_f - r_f.$$

Если $d_j < d_f$, тогда из (1.3) $d_j - r_j - p_j \ge d_f - r_f - p_f$ или $d_j - d_f \ge r_j + p_j - r_f - p_f$, поэтому $c_j(\pi) - c_f(\pi) < p_j + r_j - p_f - r_f \le d_j - d_f$. Отсюда $L_j(\pi, t) < L_f(\pi, t)$ для всех требований j, $(j \longrightarrow f)_\pi$.

Аналогично докажем неравенство (1.7).

Для всех требований j, удовлетворяющих условию $(j \longrightarrow s)_\pi$, имеем $c_j(\pi) < c_s(\pi)$. Если $d_j \ge d_s$, то $L_j(\pi, t) = c_j(\pi) - d_j < c_s(\pi) - d_s = L_s(\pi, t)$, поэтому справедливо (1.7).

Пусть для требования $j \in N\setminus\{f\}$, $(j \longrightarrow s)_\pi$, справедливо $d_j < d_s$, тогда $r_j > r_s$. Действительно, если предположить, что $r_j \le r_s$, то $r_j(t) \le r_s(t)$, что следует из (1.1). Кроме того, для требования s выполнено $r_s(t) \ge r(t)$ в соответствии с определениями (1.2) и (1.5). Если $r_s(t) = r(t)$, тогда для j и s можно записать $r_j(t) = r_s(t) = r(t)$ и $d_j < d_s$, что противоречит определению (1.5) требования $s(N, t)$. Если же $r_s(t) > r(t)$, т. е. $r_s > r(t)$, тогда нет требования $j \in N\backslash\{f, s\}$, для которого $r_s > r_i > r(t)$. Следовательно, для j и s получим $r_j(t) = r_s(t)$ и $d_j < d_s$, что противоречит определению (1.5) требования $s(N, t)$. Поэтому $r_j > r_s$.

Так как $c_j(\pi) \le c_s(\pi) - p_s$ и $p_j > 0$, то $c_j(\pi) - p_j < c_s(\pi) - p_s$ и так как $r_j > r_s$, поэтому $c_j(\pi) - p_j - r_j < c_s(\pi) - p_s - r_s$ и

$$c_j(\pi) - c_s(\pi) < p_j + r_j + p_s - r_s. \quad (1.8)$$

Поскольку $d_j < d_s$, тогда из (1.3) имеем $d_j - r_j - p_j \ge d_s - r_s - p_s$ или

$$c_j(\pi) - c_s(\pi) < p_j + r_j + p_s - r_s \le d_j - d_s. \quad (1.9)$$

Отсюда, $L_j(\pi) < L_s(\pi)$ для всех требований $j \in N\backslash\{f\}$, $(j \longrightarrow s)_\pi$.

Т е о р е м а 1. Если для требований подмножества $N' \subseteq N$ справедливо (1.3), то для любого момента времени $t' \ge t$ и всякого раннего расписания $\pi \in \Pi(N')$ существует $\pi' \in \Pi(N')$, что

$$\begin{aligned} &L_{\max}(\pi', t') \le L_{\max}(\pi, t'),\\ &C_{\max}(\pi', t') \le C_{\max}(\pi, t') \end{aligned} \quad (1.10)$$

и первым при расписании π' обслуживается требование $f = f(N', t')$ или $s = s(N', t')$. Если $d_f \le d_s$, то первым при расписании π' обслуживается требование f.

Д о к а з а т е л ь с т в о. Пусть $\pi = (\pi_1, f, \pi_2, s, \pi_3)$, где π_1, π_2, π_3 – частичные расписания π. Построим расписание $\pi' = (f, \pi_1, \pi_2, s, \pi_3)$. Из определений (1.1), (1.2), (1.4) получаем $r_f(t') \le r_j(t')$, $j \in N'$, отсюда $C_{\max}((f, \pi_1), t') \le C_{\max}((\pi_1, f), t')$ и

$$C_{\max}(\pi', t') \le C_{\max}(\pi, t') \quad (1.11)$$

$$L_j(\pi', t') \le L_j(\pi, t'),\ \forall j \in \{(\pi_2, s, \pi_3)\}. \quad (1.12)$$

Из леммы 1 (1.7) имеем

$$L_j(\pi', t') < L_s(\pi', t'),\ \forall j \in \{\pi_1\} \cup \{\pi_2\}. \quad (1.13)$$

Очевидно, что для требования f справедливо

$$L_f(\pi', t') \le L_f(\pi, t'). \quad (1.14)$$

Из (1.11)–(1.14) приходим к $C_{\max}(\pi', t') \le C_{\max}(\pi, t')$ и $L_{\max}(\pi', t') \le L_{\max}(\pi, t')$.

Пусть $\pi = (\pi_1, s, \pi_2, f, \pi_3)$, т. е. требование s обслуживается до требования f. Построим расписание $\pi' = (s, \pi_1, \pi_2, f, \pi_3)$. Далее доказательство может быть повторено, как и для f. Первая часть теоремы доказана.

Предположим $d_f \le d_s$ и расписание $\pi = (\pi_1, s, \pi_2, f, \pi_3)$. Построим расписание $\pi' = (f, \pi_{11}, \pi_{12}, \pi_3)$, где π_{11}, π_{12} – расписания из требований множеств $\{j : j \in \{(\pi_1, s, \pi_2)\}, d_j < d_f\}$ и $\{j : j \in \{(\pi_1, s, \pi_2)\}, d_j \ge d_f\}$. Требования в π_{11} и π_{12} упорядочены по не убыванию моментов поступлений r_j. Из $d_s \ge d_f$ следует, что $s \in \{\pi_{12}\}$.

Для каждого требования $j \in \{\pi_{11}\}$ имеем $d_j < d_f$. Из (1.3) получаем $d_j - r_j - p_j \ge d_f - r_f - p_f$, отсюда $r_j + p_j < r_f + p_f$, $\forall j \in \{\pi_{11}\}$, и $C_{\max}((f, \pi_{11}), t') = r_f(t') + p_f + \sum_{j \in \pi_{11}} p_j$. Так как требования расписания $\{\pi_{12}\}$ упорядочены по не убыванию моментов поступлений, то $C_{\max}((f, \pi_{11}, \pi_{12}), t') \le C_{\max}((\pi_1, s, \pi_2, f), t')$. В результате

$$C_{\max}(\pi', t') \le C_{\max}(\pi, t') \quad (1.15)$$

$$L_j(\pi', t') \le L_j(\pi, t'),\ \forall j \in \{\pi_3\}. \quad (1.16)$$

Требование $j \in \{\pi_{12}\}$ удовлетворяет $d_j \ge d_f$ и $c_j(\pi', t') \le c_f(\pi, t')$, а значит

$$L_j(\pi', t') \le L_f(\pi, t'),\ \forall j \in \{\pi_{12}\}. \quad (1.17)$$

Так как $s \in \{\pi_{12}\}$, то

$$L_s(\pi', t') \le L_f(\pi, t'). \quad (1.18)$$

Из леммы 1

$$L_j(\pi', t') \le L_s(\pi, t'),\ \forall j \in \{\pi_{11}\}. \quad (1.19)$$

Более того, очевидно, что

$$L_f(\pi', t') \le L_f(\pi, t'). \quad (1.20)$$

Из (1.15)–(1.20) следует $C_{\max}(\pi', t') \le C_{\max}(\pi, t')$ и $L_{\max}(\pi', t') \le L_{\max}(\pi, t')$, что и требовалось доказать.

Будем называть расписание $\pi' \in \Pi(N)$ эффективным, если не существует расписания $\pi \in \Pi(N)$, что $L_{\max}(\pi) \le L_{\max}(\pi')$ и $C_{\max}(\pi) \le C_{\max}(\pi')$, причем хотя бы одно неравенство строгое.

Таким образом, когда для требований множества N выполняются ограничения (1.3), то найдется эффективное расписание π', при котором первым обслуживается либо требование $f = f(N, t)$, либо $s = s(N, t)$. Более того, если $d_f \le d_s$, то существует эффективное расписание π' с первоочередным обслуживанием требования f.

Определим $\Omega(N, t)$ как подмножество $\Pi(N)$. Расписание $\pi = (i_1, i_2, \ldots, i_n)$ принадлежит $\Omega(N, t)$, если требование i_k, $k = 1, 2, \ldots, n$, выбирается из $f_k = f(N_{k-1}, c_{i_{k-1}})$ и $s_k = s(N_{k-1}, c_{i_{k-1}})$, где $N_{k-1} = N\backslash\{i_1, i_2, \ldots, i_{k-1}\}$, $c_{i_{k-1}} = c_{i_{k-1}}(\pi)$ и $N_0 = N$, $c_{i_0} = t$. Для $d_{f_k} \le d_{s_k}$ выполняется $i_k = f_k$, если $d_{f_k} > d_{s_k}$, то $i_k = f_k$ или $i_k = s_k$. Очевидно, что множество расписаний $\Omega(N, t)$ содержит не более 2^n расписаний. Для примера, когда

$$\begin{cases} n = 2(m, t \le r_1 < r_2 < \ldots < r_n,) \\ r_{2i-1} < r_{2i} + p_{2i} < r_{2i-1} + p_{2i-1}, 1 < i \le m, \\ r_{2i} + p_{2i} + p_{2i-1} \le r_{2i+1}, 1 < i \le m-1, \\ d_j = r_j + p_j, 1 \le j \le n, \end{cases} \quad (1.21)$$

множество $\Omega(N, t)$ включает 2^m расписаний.

Т е о р е м а 2. Если для требований подмножества $N' \subseteq N$, $|N'| = n'$, справедливо (1.3), то для любых момента времени $t' \ge t$ и расписания $\pi \in \Pi(N')$ существует такое расписание $\pi' \in \Omega(N', t')$, что

$$L_{\max}(\pi', t') \le L_{\max}(\pi, t') \text{ и } C_{\max}(\pi', t') \le C_{\max}(\pi, t')$$

Д о к а з а т е л ь с т в о. Пусть $\pi = (j_1, j_2, \ldots, j_n')$ – произвольное расписание. Будем обозначать первые l требований расписания π как π_l, $l = 0, 1, 2, \ldots, n'$, где π_0 – пустое расписание, а $\bar{\pi}_l = (j_{l+1}, \ldots, j_n')$, тогда $\pi = (\pi_l, \bar{\pi}_l)$. Введем $N_l = N' \setminus \{\pi_l\}$ и $C_l = C_{\max}(\pi_l, t')$. Предположим, что для некоторого l, $0 \le l$, $\pi_l < n'$, π_l является наибольшим начальным частичным расписанием некоторого расписания из $\Omega(N', t')$. Если $j_1 \ne f(N', t')$ и $j_1 \ne s(N', t')$, то $\pi_l = \pi_0$, $l = 0$, т. е. наибольшим частичным расписанием будет пустое. Допустим $f = f(N_l, C_l)$ и $s = s(N_l, C_l)$. Если $d_f > d_s$, то $j_{l+1} \ne f$ и $j_{l+1} \ne s$, напротив когда $d_f \le d_s$, то $j_{l+1} \ne f$, так как π_{l+1} не является начальным расписанием некоторого расписания из $\Omega(N', t')$.

Согласно теореме 1 для требований множества $\{\bar{\pi}_l\}$, $\bar{\pi}_l \in \Pi(N_l)$, с момента времени C_l существует расписание $\bar{\pi}'_l$, для которого выполняется $L_{\max}(\bar{\pi}'_l, C_l) \le L_{\max}(\bar{\pi}_l, C_l)$ и $C_{\max}(\bar{\pi}'_l, C_l) \le C_{\max}(\bar{\pi}_l, C_l)$, и $[\bar{\pi}_l]_1 = f$ или s, более того, при $d_f \le d_s$, справедливо $[\bar{\pi}_l]_1 = f$, где $[\sigma]_k$ – требование на k-ом месте при расписании σ. Отсюда $L_{\max}((\pi_l, \bar{\pi}'_l), t') \le L_{\max}((\pi_l, \bar{\pi}_l), t')$ и $C_{\max}((\pi_l, \bar{\pi}'_l), t') \le C_{\max}((\pi_l, \bar{\pi}_l), t')$.

Обозначим $\pi' = (\pi_l, \bar{\pi}'_l)$. Получим расписание π', отличающееся тем, что обслуживание первых $l + 1$ требований будет совпадать с обслуживанием первых $l + 1$ требований некоторого расписания из множества $\Omega(N', t')$ и $L_{\max}(\pi', t') \le L_{\max}(\pi, t')$, $C_{\max}(\pi', t') \le C_{\max}(\pi, t')$.

После менее чем n' последовательных преобразований (так как длина расписания $n' \le n$) исходного произвольно выбранного расписания π приходим к расписанию $\pi' \in \Omega(N', t')$, обеспечивающему $L_{\max}(\pi', t') \le L_{\max}(\pi, t')$ и $C_{\max}(\pi', t') \le C_{\max}(\pi, t')$, что и требовалось доказать.

Сформируем следующее частичное расписание: $\omega(N, t) = (i_1, i_2, \ldots, i_l)$. Для каждого требования i_k, $k = 1, 2, \ldots, l$, имеем $i_k = f_k$ и $d_{f_k} \le d_{s_k}$, где $f_k = f(N_{k-1}, C_{k-1})$ и $s_k = s(N_{k-1}, C_{k-1})$. Для $f = f(N_l, C_l)$ и $s = s(N_l, C_l)$ выполняется $d_f > d_s$. Если $d_f > d_s$ для $f = f(N, t)$ и $s = s(N, t)$, то $\omega(N, t) = \emptyset$. Таким образом, $\omega(N, t)$ – "максимальное" расписание, при построении которого однозначно выбирается требование (f) на очередное место расписания. С помощью алгоритма 1 для требований множества N с момента времени t может быть построено расписание $\omega(N, t)$.

А л г о р и т м 1. Предварительно положим $\omega = \emptyset$. Найдем требования $f = f(N, t)$ и $s = s(N, t)$. Если $d_f \le d_s$, то $\omega = (\omega, f)$, иначе алгоритм заканчивает работу; положим $N = N \setminus \{f\}$, $t = r_f(t) + p_f$ и повторяем операции для следующего шага.

Л е м м а 2. Трудоемкость алгоритма 1 нахождения расписания $\omega(N, t)$ для любых N и t составляет не более $O(n \lg n)$ операций.

Д о к а з а т е л ь с т в о. На каждой итерации алгоритма 1 находим два требования: $f = f(N, t)$ и $s = s(N, t)$. Если требования упорядочены по временам поступления r_j (и соответственно момент времени $r(t)$ находится за $O(1)$ операций), то для нахождения двух требований (f и s) необходимо $O(\lg n)$ операций. Всего итераций не более n. Таким образом, для построения расписания $\omega(N, t)$ требуется $O(n \lg n)$ операций.

Л е м м а 3. Если для требований множества N справедливо (1.3), то любое расписание $\pi \in \Omega(N, t)$ начинается с расписания $\omega(N, t)$.

Д о к а з а т е л ь с т в о. Когда $\omega(N, t) = \emptyset$, т. е. $d_f > d_s$, где $f = f(N, t)$, $s = s(N, t)$, утверждение леммы справедливо, так как любое расписание начинается с пустого.

Пусть $\omega(N, t) = (i_1, i_2, \ldots, i_l)$, а значит для каждого i_k, $k = 1, 2, \ldots, l$, имеем $i_k = f_k$ и $d_{f_k} \le d_{s_k}$, где $f_k = f(N_{k-1}, C_{k-1})$ и $s_k = s(N_{k-1}, C_{k-1})$. Для $f = f(N_l, C_l)$ и $s = s(N_l, C_l)$ справедливо $d_f > d_s$. Как видно из определения множества расписаний $\Omega(N, t)$, все расписания этого подмножества будут начинаться с частичного расписания $\omega(N, t)$.

Воспользуемся следующими обозначениями $\omega^1(N, t) = (f, \omega(N', t'))$ и $\omega^2(N, t) = (s, \omega(N'', t''))$, где $f = f(N, t)$, $s = s(N, t)$, $N' = N \setminus \{f\}$, $N'' = N \setminus \{s\}$, $t' = r_f(t) + p_f$, $t'' = r_s(t) + p_s$. Очевидно, что алгоритм для нахождения ω^1 (а также ω^2) выполняется за $O(n \lg n)$ операций также, как и алгоритм для построения $\omega(N, t)$.

С л е д с т в и е и з л е м м ы 3. Если для требований множества N справедливо (1.3), то каждое расписание $\pi \in \Omega(N, t)$ начинается или с $\omega^1(N, t)$, или с $\omega^2(N, t)$.

Т е о р е м а 3. Если требования множества N удовлетворяют (1.3), то для любого расписания $\pi \in \Omega(N, t)$ выполняется $(i \longrightarrow j)_\pi$ для каждого $i \in \{\omega^1(N, t)\}$ и $j \in N \setminus \{\omega^1(N, t)\}$.

Доказательство. В случае $\{\omega^1(N, t)\} = N$ утверждение теоремы, очевидно, справедливо. Пусть $\{\omega^1(N, t)\} \neq N$. Далее в тексте доказательства будем использовать обозначение $\omega^1 = \omega^1(N, t)$.

Если $f = f(N, t)$ и $s = s(N, t)$ таковы, что $d_f \le d_s$, то все расписания из множества $\Omega(N, t)$ начинаются с частичного расписания $\omega(N, t) = \omega^1$, поэтому утверждение теоремы также верно.

Рассмотрим случай $d_f > d_s$. Все расписания множества $\Omega(N, t)$, начинающиеся с требования f, имеют частичное расписание $\omega(N, t) = \omega^1$. Возьмем произвольное расписание $\pi \in \Omega(N, t)$ с требованием s на первом месте, $[\pi]_1 = s$, и расписание $|\omega^1| = l$, $l < n$, содержащее l требований. Пусть $l = (j_1, j_2, \dots, j_l)$ – частичное расписание длины l расписания π, $j_1 = s$. Докажем, что $\{\pi_l\} = \{\omega^1\}$. Предположим противное, что существует требование $j \in \{\pi_l\}$, но $j \notin \{\omega^1\}$.

Допустим, что $(j \longrightarrow f)_\pi$. Если $d_j < d_f$, то из (1.3) имеем $d_j - r_j - p_j \ge d_f - r_f - p_f$, поэтому $r_j + p_j < r_f + p_f$. Следовательно, требование j будет включено в расписание ω^1 согласно определению $\omega(N, t)$ и ω^1, но по предположению $j \notin \{\omega^1\}$. Если же $d_j \ge d_f$, то из того, что $\pi \in \Omega(N, t)$ следует $(f \longrightarrow j)_\pi$, но это противоречит $(j \longrightarrow f)_\pi$.

Пусть $(f \longrightarrow j)_\pi$. Тогда для каждого требования $i \in \{\omega^1\}$, для которого $i \notin \{\pi_l\}$, $r_i < r_i + p_i C_{\max}(\omega^1) < r_{s_{l+1}} \le r_j$, так как $j \notin \{\omega^1\}$, где $s_{l+1} = s(N \setminus \{\omega^1\}, C_{\max}(\omega^1))$. Требования s_{l+1} и j не были упорядочены при расписании ω^1, поэтому $C_{\max}(\pi_1) < r_{s_{l+1}} \le r_j$. Кроме того, $d_i \le d_j$. Если бы $d_i > d_j$, то $r_i + p_i \ge r_j + p_j$, но выполняется $r_i + p_i < r_j$. Следовательно, $(i \longrightarrow j)_{\pi_l}$, так как $\pi = (\pi_l, \bar{\pi}_l) \in \Omega(N, t)$, но тогда будет нарушено предположение $i \notin \{\pi_l\}$ и $j \in \{\pi_l\}$.

Значит, наше предположение не верно, поэтому $\{\omega^1\} = \{\pi_l\}$, что и требовалось доказать.

Таким образом, обслуживание требований множества $\{\omega^1(N, t)\}$ предшествует обслуживанию требований множества $N \setminus \{\omega^1(N, t)\}$ при любом расписании из множества $\Omega(N, t)$.

2. Задача на быстродействие при ограничении на максимальное временное смещение. Сформулируем задачу $1|d_i \le d_j,\ d_i - r_i - p_i \ge d_j - r_j - p_j;\ L_{\max} \le y|C_{\max}$: необходимо для некоторого действительного числа y найти расписание θ с $C_{\max}(\theta) = \min\{C_{\max}(\pi) : L_{\max}(\pi) \le y\}$. Если $L_{\max}(\pi) > y$ для любого $\pi \in \Pi(N)$, то $\theta = \emptyset$.

Алгоритм 2. Положим $\theta = \omega(N, t)$. Если $L_{\max}(\pi) > y$, то $\theta = \emptyset$ и алгоритм заканчивает работу, иначе продолжим построение расписания θ. Найдем $N' = N \setminus \{\theta\}$ и $t' := C_{\max}(\theta)$. Если $N' = \emptyset$, то выполнение алгоритма прекращается. Иначе:

если $L_{\max}(\omega^1, t') \le y$, где $\omega^1 = \omega^1(N', t')$, то $\theta = (\theta, \omega^1)$ и переходим к следующему шагу;

если $L_{\max}(\omega^1, t') > y$ и $L_{\max}(\omega^2, t') \le y$, где $\omega^2 = \omega^2(N', t')$, то $\theta = (\theta, \omega^2)$ и переходим к следующему шагу;

если $L_{\max}(\omega^1, t') > y$ и $L_{\max}(\omega^2, t') > y$, то полагаем $\theta = \emptyset$ и алгоритм заканчивает работу.

Лемма 4. Трудоемкость алгоритма 2 не превышает $O(n^2 \lg n)$ операций.

Доказательство. На каждой итерации основного шага алгоритма 2 находятся расписания ω^1 и при необходимости ω^2 за $O(n \lg n)$ операций. Так как ω^1 и ω^2 состоят по крайней мере из одного требования, то на каждой итерации алгоритма к расписанию θ добавляется одно или несколько требований или полагаем $\theta = \emptyset$ и завершаем работу. Поэтому общее число шагов алгоритма не более n. Таким образом, алгоритм 2 выполняется за $O(n^2 \lg n)$ операций.

Будем обозначать через $\theta(N, t, y)$ расписание, построенное алгоритмом 2 с момента времени t из требований множества N с максимальным временным смещением не больше y. Если $N = \emptyset$, то $\theta(\emptyset, t, y) = \emptyset$ для любого и y.

Теорема 4. Пусть для требований множества N справедливо (1.3). Если с помощью алгоритма 2 будет построено расписание $\theta(N, t, y) = \emptyset$, то $C_{\max}(\theta) = \min\{C_{\max}(\pi) : L_{\max}(\pi) \le y,\ \pi \in \Pi(N)\}$. Если же в результате работы алгоритма 2 не будет сформировано расписание, т. е. $\theta(N, t, y) = \emptyset$, то $L_{\max}(\pi) > y$ для каждого $\pi \in \Pi(N)$.

Доказательство. В ситуации, когда для расписания $\pi \in \Pi(N)$ выполняется $L_{\max}(\pi) \le y$, существует такое расписание $\pi' \in \Omega(N, t)$, что $L_{\max}(\pi') \le L_{\max}(\pi) \le y$ и $C_{\max}(\pi') \le C_{\max}(\pi)$,– согласно теореме 2. Поэтому искомое расписание θ находится среди расписаний множества $\Omega(N, t)$.

По лемме 3, все расписания множества $\Omega(N, t)$ начинаются с $\omega(N, t)$. Примем $\theta_0 = \omega(N, t)$.

После k, $k \ge 0$, основных шагов алгоритма 2 получим расписание θ_k и $N' = N \setminus \{\theta_k\}$, $t' = C_{\max}(\theta_k)$. Предположим, что существует оптимальное по быстродействию расписание θ, начинающееся с θ_k. Согласно теореме 2, возможно оптимальное продолжение расписания θ_k среди расписаний из множества $\Omega(N', t')$.

Пусть $\theta_{k+1} = (\theta_k, \omega^1(N', t'))$, т. е. $L_{\max}(\theta_{k+1}) \le y$. При расписании ω^1, $\omega^1 = \omega^1(N', t')$, нет искусственных простоев прибора и все расписания из множества $\Omega(N', t')$ начинаются с требований множества $\{\omega^1(N', t')\}$,– по теореме 3. Поэтому $\omega^1(N', t')$ наилучшее по быстродействию ($C_{\max}$) среди всех допустимых по максимальному временному смещению ($L_{\max}$) продолжений частичного расписания θ_k.

На очередном шаге $\theta_{k+1} = (\theta_k, \omega^2(N', t'))$, т. е. $L_{\max}(\omega^1, t') > y$ и $L_{\max}(\omega^2, t') \le y$. Все расписания множества $\Omega(N', t')$ начинаются с расписания $\omega^1(N', t')$ или с $\omega^2(N', t')$. Так как $L_{\max}(\omega^1, t') > y$, то существует только одно подходящее продолжение – $\omega^2(N', t')$.

Таким образом, на каждом основном шаге алгоритма выбираем наилучшее по быстродействию продолжение частичного расписания θ_k среди всех допустимых по максимальному временному смещению. После не более чем n основных шагов алгоритма будет построено искомое расписание.

Допустим, что после k + 1-го шага алгоритма $L_{\max}(\omega^1, t') > y$ и $L_{\max}(\omega^2, t') > y$. Если расписание θ существовало бы, т. е. $\theta = \emptyset$, то θ начиналось бы с θ_k. Тогда для любого расписания $\pi \in \Pi(N', t')$ существует такое расписание $\pi' \in \Omega(N', t')$, что $L_{\max}(\pi, t') \ge L_{\max}(\pi', t') \ge L_{max}(\omega^1, t') > y$ или $L_{\max}(\pi, t') \ge L_{\max}(\pi', t') \ge L_{\max}(\omega^2, t') > y$. Поэтому $\theta = \emptyset$.

Повторив наши рассуждения столько раз, сколько выполнялся основной шаг алгоритма 2 (не более чем n), придем к истинности утверждения теоремы.

3. Алгоритм построения множества Парето-расписаний по критериям $C_{\max}$ и $L_{\max}$. Ниже приводится алгоритм построения множества Парето-расписаний $\Phi(N, t) = \{\pi'_1, \pi'_2, \ldots, \pi'_m\}$ по критериям C_{max} и L_{max}, таких, что

$$C_{\max}(\pi'_1) < C_{\max}(\pi'_2) < \ldots < C_{\max}(\pi'_m),$$

$$L_{\max}(\pi'_1) < L_{\max}(\pi'_2) < \ldots < L_{\max}(\pi'_m).$$

Расписание π'_m будет решением задачи $1|r_j|\,L_{\max}$ при условии, что выполняется (1.3).

А л г о р и т м 3. Положим $y = +\infty$, $\pi^* = \omega(N, t)$, $\Phi = \emptyset$, $m = 0$.

Найдем $N' = N\backslash\{\pi^*\}$ и $t' = C_{max}(\pi^*)$. Если $N' = \emptyset$, то $\Phi = \Phi \cup (\pi^*)$, $m = 1$ и алгоритм заканчивает работу. Иначе:

если $L_{\max}(\omega^1, t') \le L_{\max}(\pi^*)$, то $\pi^* = (\pi^*, \omega^1)$, где $\omega^1 = \omega^1(N', t')$ и переходим к следующему шагу;

в случае $L_{\max}(\omega^1, t') > L_{\max}(\pi^*)$ возможны варианты:

1) $L_{\max}(\omega^1, t') < y$, тогда находим $\theta = \theta(N', t', y')$ с помощью алгоритма 2, где $y' = L_{\max}(\omega^1, t')$, и проверяем:

если $\theta = \emptyset$, то $\pi^* = (\pi^*, \omega^1)$ и переходим к следующему шагу;

иначе полагаем $\pi' = (\pi^*, \theta)$. Сравниваем $C_{\max}(\pi'_m) < C_{max}(\pi')$ и, убедившись в истинности этого неравенства выполняем: $m = m + 1$, $\pi'_m = \pi'$, $\Phi = \Phi \cup (\pi'_m)$, $y = L_{\max}(\pi'_m)$; в противном случае $\pi'_m = \pi'$ и переходим к следующему шагу;

2) $L_{max}(\pi_1, t')y$; находим $\omega^2 = \omega^2(N', t')$:

если $L_{max}(\omega^2, t') < y$, то $\pi^* = (\pi^*, \omega^2)$ и переходим к следующему шагу;

иначе $\pi^* = \pi'_m$ и алгоритм заканчивает работу.

В результате работы алгоритма 3 для множества требований N с момента времени t будет построено множество расписаний $\Phi(N, t)$, для которого верно $1 \le |(N, t)| \le n$. Множество $\Phi(N, t)$ примера (1.21) будет состоять не более чем из двух расписаний.

Л е м м а 5. Трудоемкость алгоритма 3 не превышает $O(n^3 \lg n)$ операций.

Д о к а з а т е л ь с т в о. На каждой итерации описанного шага алгоритма 3 находятся расписания ω^1 и, при необходимости, ω^2 за $O(n \lg n)$ операций согласно лемме 2, а также расписание θ за $O(n^2 \lg n)$ операций. Так как ω^1 и ω^2 состоят по крайней мере из одного требования, то на любой итерации алгоритма к расписанию π^* добавляется одно или больше требований или происходит остановка на последнем эталонном расписании π'. Поэтому общее число итераций не более n. Таким образом, алгоритм 3 выполняется не более чем за $O(n^3 \lg n)$ операций.

Т е о р е м а 5. Предположим, что для требований множества N справедливо (1.3), тогда расписание π^*, построенное алгоритмом 3, является оптимальным по критерию $L_{\max}$. Кроме того, для любого расписания $\pi \in \Pi(N)$ имеется расписание $\pi' \in \Phi(N, t)$, что $L_{\max}(\pi') \le L_{\max}(\pi)$ и $C_{\max}(\pi') \le C_{\max}(\pi)$.

Д о к а з а т е л ь с т в о. Согласно теореме 2 существует оптимальное (по $L_{\max}$) расписание, принадлежащее множеству $\Omega(N, t)$. Все расписания множества $\Omega(N, t)$ начинаются с частичного расписания $\omega(N, t)$, – согласно лемме 3.

Пусть $\pi_0 = \omega(N, t)$. После k, $k \ge 0$, основных шагов алгоритма 3 имеем частичное расписание k. Допустим наличие оптимального (по $L_{\max}$) расписания, начинающегося с k. Обозначим $N' = N\backslash\{\pi_k\}$ и $t' = C_{max}(\pi_k)$.

Если $\pi_{k+1} = (\pi_k, \omega^1)$, где $\omega^1 = \omega^1(N', t')$, тогда или $L_{max}(\omega^1, t') \le L_{max}(\pi_k)$, или $L_{max}(\pi_k) < L_{max}(\omega^1, t') < y$ – текущее значение критерия и максимальное временное смещение появится на следующих шагах алгоритма 3, т. е. $\theta(N', t', y') \ne \emptyset$, где $y' = L_{max}(\omega^1, t')$. Если же $\theta = \theta(N', t', y') \ne \emptyset$, то улучшаем текущее значение максимального временного смещения: $\pi' = (\pi_k, \theta)$ и $y = L_{max}(\pi') = L_{max}(\omega^1, t')$. Расписание π' добавляется в множество расписаний $\Phi(N, t)$. Более того, обслуживание требований множества $\{\omega^1\}$ предшествуют обслуживанию требований множества $N' \backslash\{\omega^1\}$,– согласно теореме 3. Таким образом, расписание ω^1 без искусственных простоев прибора будет лучшим продолжением для π_k.

Рассмотрим случай $\pi_{k+1} = (\pi_k, \omega^2)$, где $\omega^2 = \omega^2(N', t')$, т. е. $L_{\max}(\omega^2, t') < L_{\max}(\pi') \le L_{\max}(\omega^1, t')$, согласно алгоритму 3. Поэтому продолжение ω^2 “лучше”, чем ω^1. Отсюда частичное расписание π_{k+1} является частью некоторого оптимального расписания.

Повторяя подобные рассуждения не более n раз, мы приходим к оптимальности (по $L_{\max}$) расписания π^*. Множество расписаний $\Phi(N, t)$ будет содержать не более n расписаний, так как на каждом основном шаге алгоритма в множество $\Phi(N, t)$ добавляется не более одного расписания, а этот шаг выполняется не более n раз.

Предположим, что существует расписание $\pi \in \Pi(N)$, $\pi \notin \Phi(N, t)$, такое, что либо $C_{\max}(\pi) \le C_{\max}(\pi')$ и $L_{\max}(\pi) \ge L_{\max}(\pi')$, либо $C_{\max}(\pi) \ge C_{\max}(\pi')$ и $L_{\max}(\pi) \le L_{\max}(\pi')$ для каждого расписания $\pi' \in \Phi(N, t)$, причем в каждой паре неравенств хотя бы одно из неравенств строгое. Из теоремы 2 следует наличие расписания $\pi'' \in \Omega(N, t)$ такого, что $L_{\max}(\pi'') \le L_{\max}(\pi)$ и $C_{\max}(\pi'') \le C_{\max}(\pi)$. Если $\pi'' \in \Phi(N, t)$, то очевидно, что наше предположение не верно. Пусть $\pi'' \in \Omega(N, t)\backslash\Phi(N, t)$. Из алгоритма 3 видно, что структуру каждого расписания $\pi' \in \Phi(N, t)$ можно представить как последовательность частичных расписаний $\pi' = (\omega'_0, \omega'_1, \omega'_2, \ldots, \omega'_{k'})$, где $\omega'_0 = \omega(N, t)$, ω'_i равно либо $\omega^1(N'_i, C'_i)$, либо $\omega^2(N'_i, C'_i)$, а $N'_i = N\backslash\{\omega'_0, \ldots, \omega'_{i-1}\}$, $C'_i = C_{\max}((\omega'_0, \ldots, \omega'_{i-1}), t)$, $i = 1, 2, \ldots, k'$. Расписание π'' будет обладать аналогичной структурой, согласно определению множества $\Omega(N, t)$, т. е. $\pi = (\omega''_0, \omega''_1, \omega''_2, \ldots, \omega'_{k''})$, возможно, $k'' \ne k'$, где $\omega''_0 = \omega'_0 = \omega(N, t)$, ω''_i равно либо $\omega^1(N''_i, C''_i)$, либо $\omega^2(N''_i, C''_i)$, а $N''_i = N\backslash\{\omega''_0, \ldots, \omega''_{i-1}\}$, $C''_i = C_{\max}((\omega''_0, \ldots, \omega''_{i-1}), t)$, $i = 1, 2, \ldots, k''$.

Примем, что первые k частичных расписаний π'' и π' совпадают, т. е. $\omega''_i = \omega'_i = \omega_i$, $i = 0, 1, \ldots, k-1$, $\omega''_k \ne \omega'_k$. Пусть $y = L_{\max}(\omega_0, \ldots, \omega_{k-1})$, построим расписание θ с помощью алгоритма 2, $\theta = \theta(N_k, C_k, y)$. Если $\theta = \emptyset$, то в соответствии с алгоритмом 3, $\omega'_k = \omega^1(N_k, C_k)$, так как $\omega''_k \ne \omega'_k$, то $\omega''_k = \omega^2(N_k, C_k)$. Значение целевой функции ($L_{\max}$) будет достигаться на требовании из множества N_k, так как $\theta = \emptyset$. Вся структура алгоритма 3 построена таким образом, чтобы до “критического” требования (по $L_{\max}$) упорядочить требования как можно более “плотно”, поэтому достраиваем расписание ω^1, после чего $C_{\max}(\pi') \le C_{\max}(\pi'')$ и $L_{\max}(\pi') \le L_{\max}(\pi'')$. Если $\theta \ne \emptyset$, тогда для расписаний π' и π'' справедливо $C_{\max}(\pi') \le C_{\max}(\pi'')$ и $L_{\max}(\pi') = L_{\max}(\pi'')$. Таким образом, для любого расписания $\pi'' \in \Omega(N, t)\backslash\Phi(N, t)$ существует расписание $\pi' \in \Phi(N, t)$, что $C_{\max}(\pi') \le C_{\max}(\pi'')$ и $L_{\max}(\pi') \le L_{\max}(\pi'')$. Из этого следует, что для любого расписания $\pi \in \Pi(N)$ существует расписание $\pi' \in \Phi(N, t)$, что $L_{\max}(\pi') \le L_{\max}(\pi)$ и $C_{\max}(\pi') \le C_{\max}(\pi)$. На рисунке схематично показаны рассмотренные расписания.

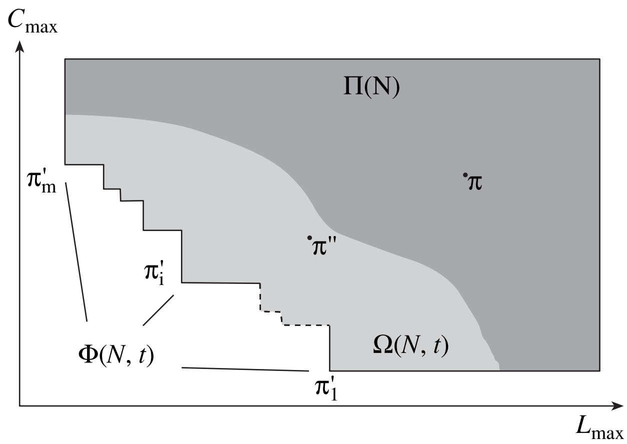

Рисунок 1. Множество Парето-оптимальных расписаний.

Для множества расписаний $\Phi(N, t) = \{\pi'_1, \pi'_2, \ldots, \pi'_m\}$ имеем

$$C_{\max}(\pi'_1) < C_{\max}(\pi'_2) < \ldots < C_{\max}(\pi'_m),$$

$$L_{\max}(\pi'_1) < L_{\max}(\pi'_2) < \ldots < L_{\max}(\pi'_m).$$

Расписание π'_1 является оптимальным по быстродействию (по $C_{\max}$), а π'_m – по максимальному временному смещению (по $L_{\max}$), если параметры требований множества N удовлетворяют условиям (1.3).

Экспериментальное исследование алгоритма 3 показало, что с его помощью может быть построены оптимальные расписания (по $L_{\max}$) и для примеров, не удовлетворяющих условиям (1.3).

СПИСОК ЛИТЕРАТУРЫ

1. *Graham R.L., Lawler E.L., Lenstra J.K. et al.* Optimization and approximation in deterministic sequencing and scheduling: a survey // Ann. Discrete Math. 1979. V. 5. P. 287–326.
2. *Lenstra J.K., Rinnooy Kan A.H.G., Brucker P.* Complexity of machine scheduling problems // Annals of Oper. Res. 1975. V. 1. P. 343–362.
3. *Jackson J.R.* Scheduling a production line to minimize maximum tardiness // Manag. Sci. 1955. V. 43.
4. *Potts C.N.* Analysis of a heuristic for one machine sequencing with release dates and delivery times // Oper. Res. 1980. V. 28. P. 1436–1441.
5. *Hall L.A., Shmoys D.B.* Jackson’s rule for one-machine scheduling: making a good heuristic better // Math. Oper. Res. 1992. V. 17. P. 22–35.
6. *Mastrolilli M.* Efficient approximation schemes for scheduling problems with release dates and delivery times // J. Scheduling. 2003. V. 6. P. 521–531.

7. *Lawler E.L.* Optimal sequencing of a single machine subject to precedence constraints // Manag. Sci. 1973. V. 19. P. 544–546.

8. *Simons B.B.* A fast algorithm for single processor scheduling // 19-th Annu. Symp. Found. Comput. Sci. N. Y. 1978. P. 246–252.

9. *Baker K.R., Lawler E.L., Lenstra J.K. et al.* Preemptive scheduling of a single machine to minimize maximum cost subject to release dates and precedence constraints // Oper. Res. 1983. V. 31. P. 381–386.

10. *Hoogeveen J.A.* Minimizing maximum promptness and maximum lateness on a single machine // Math. Oper. Res. 1996. V. 21. P. 100–114.

11. *Лазарев А.А., Шульгина О.Н.* Псевдополиномиальный алгоритм решения NP-трудной задачи минимизации максимального временного смещения // Тр. 11-й междунар. Байкальской школы-семинара “Методы оптимизации и их приложения”, 5–12 июля. Иркутск. 1998. С. 163–167.

12. *Лазарев А.А., Шульгина О.Н.* Полиномиально разрешимые частные случаи задачи минимизации максимального временного смещения // Изв. вузов. Математика. 2000. 11 с.

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

Pareto optimal set schedules for NP – hard minimizing maximum lateness problem.

© 2006 г. Lazarev A. A.

We consider NP-hard in strong sense problem for single machine $1|r_j|L_{\max}$. New properties of optimum schedules are found. It is allocated new polynomial case when r_j (release times), p_j (processing times and d_j (due dates) satisfy to restrictions $d_1 \le \ldots \le d_n$;$d_1 - r_1 - p_1 \ge \ldots \ge d_n - r_n - p_n$. The algorithm with run times $O(n^3 \lg n)$ operations finds Pareto optimum set of schedules by criteria $L_{\max}$ and $C_{\max}$, containing no more n schedules.