ДИСКРЕТНЫЙ АНАЛИЗ И ИССЛЕДОВАНИЕ ОПЕРАЦИЙ
Апрель—июнь 1996. Том 3, № 2, 15–32

УДК 519.8

КОМБИНАТОРНАЯ СЛОЖНОСТЬ СОСТАВЛЕНИЯ РАСПИСАНИЙ ДЛЯ РАБОТ С ПРОСТЫМ ЛИНЕЙНЫМ РОСТОМ ДЛИТЕЛЬНОСТЕЙ*)

А. В. Кононов

В работе изучаются системы поточного типа (flow shop), открытого типа (open shop) и система из параллельных машин. Длительность операций прямо пропорциональна времени постановки операции на выполнение. Рассматриваются следующие критерии оптимальности: минимизация общего времени выполнения всех работ, минимизация суммарного времени выполнения всех работ, минимизация максимального запаздывания. Изучается комбинаторная сложность задач.

Задачи теории расписаний характеризуются списком работ или операций, списком машин, на которых они выполняются, и временем выполнения каждой операции. В классическом случае предполагается, что длительность работы постоянна и не зависит от момента времени, с которого она начнет выполняться. Однако часто на практике работа, которая начинает выполняться позже, требует больше времени на свое выполнение. Например, в химической промышленности, в работах, связанных с очисткой окружающей среды, и т. д. В [1–5] изучались различные задачи, в которых длительность операций зависит от времени начала их выполнения. В [1] рассматривалась задача, в которой время выполнения операции j равно $v_j t_j$. Такие работы будем называть *работами с простым линейным ростом длительностей.* Для различных критериев в [1] построены полиномиальные алгоритмы нахождения оптимального расписания для n работ на одной машине. Далее там же отмечено, что полиномиально-разрешимые случаи для задач на одной машине с постоянными длительностями остаются полиномиально-разрешимыми и в рассматриваемом случае.

Целью данной статьи является изучение этой задачи при наличии нескольких машин. В ней рассматриваются системы поточного типа

*) Работа выполнена при финансовой поддержке Российского фонда фундаментальных исследований (код проекта 95–01–00989).

(flow shop), открытого типа (open shop) и система из M параллельных машин. Устанавливаются NP-трудность задач минимизации общего времени выполнения всех работ для многооперационных систем на трех машинах и для системы из параллельных машин; NP-трудность задач минимизации максимального запаздывания для многооперационных систем на двух машинах; NP-трудность задачи построения оптимального расписания на двух параллельных машинах с критерием минимизации суммарного времени выполнения всех работ, хотя в классическом случае эта задача полиномиально-разрешима [6]. В § 1 сформулированы основные определения и обозначения, в § 2 приведены вспомогательные результаты, § 3 посвящен задачам с параллельными машинами, § 4 — задачам построения расписания в системах поточного и открытого типа. В § 5 приведены полиномиально-разрешимые случаи.

§ 1. Основные обозначения и определения

Пусть задано множество работ $I = \{1, 2, \ldots, n\}$. Каждая работа $i \in I$ состоит из последовательности операций Ω_i. Если t_j — момент начала выполнения операции j, то длительность выполнения операции j равна $p_j = v_j t_j$, где $v_j \geqslant 0$ — постоянная величина. Если для операции указан номер машины, на которой она выполняется, то в дальнейшем для ее параметров будет использоваться двухиндексная запись, например $v_{i,j}$. Все работы поступают на выполнение в момент времени $t = 1$. Прерывания при выполнении работ не допускаются. Операции выполняются на M машинах. Одновременное обслуживание любой работы несколькими машинами не допускается. В каждый момент на машине выполняется не более одной операции. Рассматриваются следующие задачи теории расписаний.

Задача P (система с параллельными машинами). Каждая работа $i \in I$ состоит из одной операции, которая может выполняться на любой машине

Задача F (система потокового типа). Каждая работа $i \in I$ состоит из M последовательных операций. Операция j_i, $i \in I$, выполняется на машине j. Порядок выполнения операций одинаков для всех работ и определяется последовательностью машин $1, 2, \ldots, M$.

Задача O (система открытого типа). Каждая работа $i \in I$ состоит из M операций. Операция j_i выполняется на машине j. Порядок выполнения операций произвольный.

Обозначим через $\bar{t}_i = t_i + p_i$ время завершения работы i, а через d_i — директивный срок для окончания работы i. В качестве целевой функции f будем использовать один из следующих функционалов:

$f = C_{\max} = \max\limits_{i \in I} \bar{t}_i$ — время завершения выполнения всех работ;

$f = \Sigma C = \sum_{i \in I} \bar{t}_i$ — суммарное время завершения выполнения всех работ;

$f = L_{\max} = \max_{i \in I}\{0; \bar{t}_i - d_i\}$ — максимальное запаздывание.

Расписанием в рассматриваемых задачах будем называть задание для каждой операции j момента t_j начала выполнения этой операции и номера машины, на которой она будет выполняться. В дальнейшем через s будем обозначать расписание в различных задачах, а через $f(s)$ — значение целевой функции в расписании s.

При обозначении конкретной задачи для удобства будем использовать трехиндексную запись $\langle\alpha|\beta|\gamma\rangle$, введенную в [6]. Пусть $\circ$ обозначает пустой символ. Первый индекс $\alpha = \alpha_1\alpha_2$ связан с ограничениями на машины и означает следующее:

— если $\alpha_1 = \circ$, то работы выполняются на одной машине;

— если $\alpha_1 = P$, то работы выполняются в системе с параллельными машинами;

— если $\alpha_1 = F$, то работы выполняются в системе потокового типа;

— если $\alpha_1 = O$, то работы выполняются в системе открытого типа.

Если α_2 является натуральным числом, то число машин равно α_2. Если $\alpha_2 = \circ$, то число машин произвольно. Второй индекс связан с указанием различных характеристик работ. Если $\beta = vt$, то длительность выполнения операции j равна $p_j = v_j t_j$. Третий индекс $\gamma \in \{C_{\max}, \Sigma C, L_{\max}\}$ определяет вид целевой функции.

§ 2. Вспомогательные результаты

Рассмотрим две вспомогательные задачи, которые в дальнейшем будем использовать для доказательства NP-трудности задач теории расписаний, изучаемых в настоящей статье.

Задача SP (произведение размеров). Заданы конечное множество A, «размеры» $\eta(j) \in Z^+$ всех элементов $j \in A$ и положительное целое число b. Существует ли такое подмножество $A' \subseteq A$, что произведение размеров его элементов равно b.

Задача 4-P (4-произведение). Заданы множество A, состоящее из $4n$ элементов, граница $b \in Q^+$ и «размеры» $\eta(j) \in Q^+$, $\eta(j) \geqslant 1$ всех элементов, $a \in A$, причем $\sqrt[5]{b} < \eta(j) < \sqrt[3]{b}$ и $\prod_{j \in A} \eta(j) = b^n$. Можно ли A разбить на n непересекающихся подмножеств $A_1, A_2, \dots, A_n$ таких, что $\prod_{j \in A_i} \eta(j) = b$, $1 \leqslant i \leqslant n$.

Отметим, что указанные ограничения на размеры элементов приводят к тому, что каждое множество A_i должно содержать точно четыре элемента из множества A.

Задача SP приведена в списке NP-полных задач в [8].

Теорема 1. *Задача 4-P является NP-полной в сильном смысле.*

Доказательство. Поскольку за полиномиальное время легко проверить, что заданное разбиение множества A удовлетворяет необходимым требованиям, то легко видеть, что задача 4-P лежит в классе NP. Убедимся в том, что данная задача является NP-полной в сильном смысле. Для этого сведем следующую NP-полную в сильном смысле задачу о трехмерном сочетании к задаче 4-P.

Задача 3-С (трехмерное сочетание). Дано множество $\Phi \subseteq W \times X \times Y$, где W, X и Y — непересекающиеся множества, содержащие одинаковое число элементов q. Верно ли, что Φ содержит трехмерное сочетание, т. е. подмножество $\Phi' \subseteq \Phi$ такое, что $|\Phi'| = q$ и никакие два разных элемента Φ' не имеют ни одной равной координаты.

NP-полнота в сильном смысле задачи 3-С установлена в [7].

Пусть $W = \{w_1, w_2, \dots, w_q\}$, $X = \{x_1, x_2, \dots, x_q\}$, $Y = \{y_1, y_2, \dots, y_q\}$ и $\Phi \subseteq W \times X \times Y$ определяют произвольную индивидуальную задачу из 3-С. Не ограничивая общности, можно считать, что $|\Phi| \geqslant q$. Построим множество A из $4|\Phi|$ элементов. Каждому элементу $g \in W \bigcup X \bigcup Y$ будет соответствовать $N(g)$ элементов из множества A, где $N(g)$ — число троек из Φ, в которые входит g. Элементы, соответствующие фиксированному $g \in W \bigcup X \bigcup Y$, обозначим через $g[1], g[2], \dots, g[N(g)]$. Размеры этих элементов определим следующим образом:

$$\begin{aligned}
\eta(w_i[1]) &= \varphi_Q^{10}\varphi_i, && 1 \leqslant i \leqslant q;\\
\eta(w_i[l]) &= \varphi_Q^{11}\varphi_i, && 1 \leqslant i \leqslant q, \quad 2 \leqslant l \leqslant N(w_i);\\
\eta(x_j[1]) &= \varphi_Q^{10}\varphi_{q+j}, && 1 \leqslant j \leqslant q;\\
\eta(x_j[l]) &= \varphi_Q^{11}\varphi_{q+j}, && 1 \leqslant j \leqslant q, \quad 2 \leqslant l \leqslant N(x_j);\\
\eta(y_k[1]) &= \varphi_Q^{10}\varphi_{2q+k}, && 1 \leqslant k \leqslant q;\\
\eta(y_k[l]) &= \varphi_Q^{8}\varphi_{2q+k}, && 1 \leqslant k \leqslant q, \quad 2 \leqslant l \leqslant N(y_k),
\end{aligned}$$

где φ_k есть k-е простое число, а $Q = 3q + 1$. Единственный элемент, соответствующий каждой тройке $\phi_l = (w_i, x_j, y_k) \in Z$, обозначается через u_l, а его размер $\eta(u_l)$ зависит от индексов элементов тройки следующим образом: $\eta(u_l) = \varphi_Q^{12}/(\varphi_i\varphi_j\varphi_k)$. В качестве границы возьмем $b = \varphi_Q^{42}$. Из неравенства Чебышева для распределения простых чисел [9] следует, что $k < \varphi_k < 10k\ln k/9$, выбранные размеры элементов удовлетворяют неравенству $\sqrt[5]{b} < \eta(a) < \sqrt[3]{b}$. Размер максимального элемента не превосходит $c_1 q^{42} \ln q + c_2$ и ограничен полиномом от общего числа элементов задачи 3-С.

Для доказательства сильной NP-полноты задачи 4-P достаточно показать, что искомое 4-разбиение существует тогда и только тогда, когда Φ содержит трехмерное сочетание.

1. Предположим сначала, что имеется трехмерное сочетание $\Phi' \subseteq \Phi$. Соответствующее 4-разбиение получается из $|\Phi|$ подмножеств, каждое из которых содержит 4 элемента — один вида u_l, один вида w_i, один вида x_j, один вида y_k, где $(w_i, x_j, y_k) = \phi_l \in \Phi$. Если $\phi_l \in \Phi'$, то u_l группируем с элементами $w_i[1]$, $x_j[1]$, $y_k[1]$. Если $\phi_l \notin \Phi'$, то u_l группируем с элементами $w_i[r]$, $x_j[r]$, $y_k[r]$, $r \neq 1$. Заметим, что в полученном разбиении произведение элементов в каждой четверке равно $b = \varphi_Q^{42}$.

2. Предположим, что задано 4-разбиение нужного вида. Рассмотрим любое 4-множество из этого 4-разбиения. Размер любого элемента $g \in W \bigcup X \bigcup Y$ имеет вид $\eta(g) = \phi_Q^r \phi_g$, где r — целое, а φ_g — простое число, не равное φ_Q. Так как $b = \varphi_Q^{42}$, то в 4-разбиение входит элемент u_l. Размер любого элемента u_l имеет вид $\eta(u_l) = \varphi_Q^{12}/(\varphi_i \phi_j \varphi_k)$, где $1 \leqslant i \leqslant q$, $q+1 \leqslant j \leqslant 2q$, $2q+1 \leqslant i \leqslant 3q$. Поэтому в 4-разбиение входит по одному элементу, соответствующему каждому из составляющих тройки, причем все три элемента имеют вид либо $w_i[1]$, $x_j[1]$, $y_k[1]$, либо $w_i[r]$, $x_j[r]$, $y_k[r]$, $r \neq 1$. Пусть w_i, x_j, y_k обозначают соответствующие элементы множеств W, X и Y, а $\phi_l = (w_{i'}, x_{j'}, y_{k'})$ обозначает соответствующую тройку из Φ. Из простоты чисел φ_i, φ_j, φ_k, $\varphi_{i'}$, $\varphi_{j'}$, $\varphi_{k'}$ следует, что $i = i'$, $j = j'$, $k = k'$. Таким образом, тройки вида $w_i[1]$, $x_j[1]$, $y_k[1]$, входящие в заданное 4-разбиение, содержатся в Φ и образуют искомое трехмерное сочетание. Теорема 1 доказана.

§ 3. Параллельные машины

Установим NP-трудность задач построения расписаний для параллельных машин с критериями $C_{\max}$ и ΣC.

Лемма 1. *Пусть с момента времени t на некоторой машине без простоя выполняется n работ. Тогда время $\overline{T}$ завершения выполнения всех работ равно*

$$\overline{T} = t \prod_{i=1}^{n} (v_i + 1)$$

и не зависит от порядка выполнения работ.

Справедливость леммы 1 непосредственно следует из определения длительности работ $p_j = v_j t_j$.

Лемма 2. *Пусть n операций в расписании s выполняются на m машинах, причем машина k, $1 \leqslant k \leqslant m$, работает без простоев в проме-*

жутке времени $[T_k, \bar{T}_k]$. *Тогда*

$$\prod_{i=1}^{n}(v_i+1)=\frac{\prod_{k=1}^{m}\overline{T}_k}{\prod_{k=1}^{m}T_k}.$$

Справедливость леммы 2 непосредственно вытекает из леммы 1.

Теорема 2. *Задача* $P|vt|C_{\max}$ *является* NP-*трудной в сильном смысле.*

Доказательство. Задачу 4-Р сведем к задаче $P|vt|C_{\max}$. Положим $M = n$ и $v_i = \eta(i) - 1$. Покажем, что в построенной задаче расписание s, на котором $C_{\max}(s) \leqslant b$, существует тогда и только тогда, когда в A существует разбиение на n непересекающихся подмножеств $A_1, A_2, \dots, A_n$ таких, что $\prod\limits_{i\in A_j} \eta(i) = b$ для $1 \leqslant j \leqslant n$.

1. Пусть такое разбиение существует. Тогда расписание s строится следующим образом. На машине m выполняются работы с номерами из множества A_m в произвольном порядке. Обозначим через $\overline{T}_m$ время завершения обслуживания работ на машине m. По лемме 1 получаем

$$\overline{T}_m=\prod_{i\in A_m}(v_i+1)=\prod_{i\in A_m}\eta(i)=b.$$

Следовательно, $C_{\max}(s) = \max\limits_{i\in I} \overline{\eta}_i = \max_{m=1\dots n} \overline{T}_m = b$.

2. Пусть существует расписание s, на котором $C_{\max}(s) \leqslant b$. Обозначим через A'_m множество работ, выполняемых на машине m. Покажем, что разбиение множества A на подмножества A'_j является решением задачи 4-Р. Для этого покажем, что все машины работают в промежутке времени $[1, b]$ без простоев. Предположим, что это не так. Пусть на машине m существует простой. Так как $\overline{T}_m \leqslant b$, то $\prod\limits_{i\in A'_m} (1 + v(i)) < b$. Тогда

$$\prod_{i\in A}\eta(i)=\prod_{i\in A}(1+v(i))=\prod_{m=1}^{n}\prod_{i\in A'_m}(1+v(i))<b^n.$$

Получили противоречие. Следовательно, $\prod\limits_{i\in A'_m} \eta(i) = b$ и разбиение множества A на подмножества A'_m является решением задачи 4-Р. Теорема 2 доказана.

Теорема 3. *Задача* $P2|vt|C_{\max}$ *является* NP-*трудной.*

Доказательство. Задачу SP сведем к задаче $P2|vt|C_{\max}$. Пусть $a = \prod\limits_{i\in A} \eta(i)$. Положим $M = n + 2$, $v_{n+1} = 2(a/b) - 1$, $v_{n+2} = 2b - 1$, $v_i = \eta(i) - 1$, где $i \in A$. Покажем, что в построенной задаче расписание s, на котором $C_{\max}(s) \leqslant 2a$, существует тогда и только тогда, когда в A есть подмножество $A' \subseteq A$ такое, что произведение размеров его элементов равно b.

1. Пусть такое подмножество существует. Тогда расписание s строится следующим образом. На первой машине выполняются работа $n+1$ и работы множества A' в произвольном порядке, а на второй — остальные работы. Тогда

$$\overline{T}_1 = (v_{n+1}+1)\prod_{i\in A'}(v_i+1) = 2\frac{a}{b}\prod_{i\in A'}\eta(i) = 2a,$$

$$\overline{T}_2 = (v_{n+2}+1)\prod_{i\in A\setminus A'}(v_i+1) = 2b\prod_{i\in A\setminus A'}\eta(i) = 2a.$$

Следовательно, $C(s) = \max\{\overline{T}_1, \overline{T}_2\} \leqslant 2a$ и расписание s искомое.

2. Пусть существует расписание s такое, что $C_{\max}(s) \leqslant 2a$. В расписании s работы $n+1$ и $n+2$ выполняются на разных машинах. Действительно, если это не так, то $C_{\max}(s) \geqslant (v_{n+1}+1)(v_{n+2}+1) = 2(a/b)2b = 4a > 2a$. Получаем противоречие. Пусть для определенности работа $n+1$ в расписании s выполняется на первой машине. Обозначим через I_1 множество работ, выполняемых на этой машине. Положим $A' = I_1 \setminus \{n+1\}$. Так как $C_{\max}(s) = \max\{T_1, T_2\} \leqslant 2a$ и $\prod\limits_{i\in I}(v_i+1) = 4a^2$, то из леммы 2 следует, что в промежутке времени $[1, 2a]$ все машины работают без простоев. Тогда $2a = \overline{T}_1 = 2\frac{a}{b}\prod\limits_{i\in A'}.\eta(i)$. Следовательно, $\prod\limits_{i\in A'}\eta(i) = b$ и множество A' искомое. Теорема 3 доказана.

Перейдем к рассмотрению задач с критерием ΣC.

Лемма 3 [1]. *Пусть с момента времени t на некоторой машине без простоя обслуживается подряд k работ. Тогда*

$$\sum_{i=1}^{k}\overline{T}_i = t\sum_{i=1}^{k}\prod_{j=1}^{i}(v_j+1).$$

Утверждение леммы 3 следует из определения длительности работ $p_j = v_j t_j$.

Лемма 4. *Пусть $a_i \geqslant 2$ для всех $1 \leqslant i \leqslant n, n \geqslant 2$. Тогда*

$$\sum_{k=1}^{n-1}\prod_{i=1}^{k}a_i \leqslant \prod_{i=1}^{n}a_i.$$

Доказательство проводится по индукции. При $n = 2$ утверждение справедливо, так как $a_1 \leqslant a_1a_2$. Пусть для $n = l-1$ неравенство выполнено. Тогда

$$\sum_{k=1}^{l-1}\prod_{i=1}^{k}a_i=\prod_{i=1}^{l-1}a_i+\sum_{k=1}^{l-2}\prod_{i=1}^{k}a_i\leqslant 2\prod_{i=1}^{l-1}a_i\leqslant\prod_{i=1}^{l}a_i.$$

Лемма 4 доказана.

Лемма 5 [1]. *Пусть множество работ I упорядочено в порядке возрастания величин v_i, $i\in I$. Тогда данный порядок задает оптимальное расписание для задачи $1|vt|\Sigma C$.*

Справедливость этого утверждения следует из леммы 3.

Теорема 4. *Задача $P2|vt|\Sigma C$ является NP-трудной.*

Доказательство. Задачу SP сведем к задаче $P2|vt|\Sigma C$. Положим $M=n+4$ и $v_{n+1}=2(a/b)-1$, $v_{n+2}=2b-1$, $v_{n+3}=v_{n+4}=6a-1$, $v_i=\eta(i)-1$, $i\in A$. Покажем, что в этой задаче расписание s, при котором $\Sigma C(s)\leqslant y=24a^2+8a$, существует тогда и только тогда, когда в A есть подмножество $A'\subseteq A$ такое, что произведение размеров его элементов равно b.

1. Пусть такое подмножество существует. Тогда расписание s строится следующим образом. На первой машине выполняются работы $n+1$, $n+3$ и работы множества A', а на второй — остальные работы. Каждая машина выполняет работы без простоев в порядке возрастания величин v_i. Тогда работы $n+3$, $n+4$ начинают выполняться в моменты времени

$$t_{n+3}=(v_{n+1}+1)\prod_{i\in A'}(v_i+1)=2\frac{a}{b}\prod_{i\in A'}\eta(i)=2a,$$

$$t_{n+4}=(v_{n+2}+1)\prod_{i\in A\setminus A'}(v_i+1)=2b\prod_{i\in A\setminus A'}\eta(i)=2a.$$

Обозначим через σ_i суммарное время выполнения работ на машине i. Пусть $|A'|=k$. Используя лемму 3, получаем

$$\sigma_1=\sum_{i=1}^{k}\bar{t}_i+t_{n+1}+t_{n+3}=\sum_{i=1}^{k}\prod_{j=1}^{i}(v_j+1)+2a+12a^2.$$

Из леммы 4 следует, что $\sum\limits_{i=1}^{k}\prod\limits_{j=1}^{i}(v_j+1)\leqslant 2a$ и $\sigma_1\leqslant 12a^2+4a$. Проделав аналогичные вычисления для машины 2, получаем $\sigma_2\leqslant 12a^2+4a$. Следовательно, $\Sigma C(s)\leqslant 24a^2+8a$ и расписание s искомое.

2. Пусть существует расписание s такое, что $\Sigma C(s) \leqslant y$. В расписании s работы $n+3$, $n+4$ не могут выполняться на одной машине, так как $\bar{t}_{n+3}+\bar{t}_{n+4} \geqslant (6a+1)+(6a+1)^2 > y$. Покажем, что в расписании s выполнение этих работ должно начаться не позднее момента времени $2a$. Пусть это не так. Согласно лемме 5 оптимальное расписание работ на каждой машине получается упорядочением их в порядке возрастания величин v_i. Поэтому можно считать, что работы $n+3$, $n+4$ выполняются последними. Предположим, что в расписании s работа $n+3$ начинает выполняться в момент времени $2ax$, где $x > 1$. Тогда работа $n+4$ начинает выполняться не ранее момента времени $2a/x$. Так как $2ax + 2a/x > 4a$ и все параметры задачи целые, то $2ax + 2a/x \geqslant 4a+1$,

$$\begin{aligned}\Sigma C(s) &\geqslant t_{n+3} + t_{n+4} + \bar{t}_{n+3} + \bar{t}_{n+4}\\ &\geqslant 2ax + \frac{2a}{x} + 2ax6a + \frac{2a}{x}6a \geqslant (6a+1)(4a+1) = 24a^2 + 10a + 1.\end{aligned}$$

Получили противоречие. Следовательно, к моменту времени $2a$ все работы, кроме работ $n+3$ и $n+4$, должны быть выполнены. Тогда в расписании s работы $n+1$ и $n+2$ выполняются на разных машинах, так как иначе $C_{\max}(s) \geqslant (v_{n+1}+1)(v_{n+2}+1) = 2(a/b)2b = 4a > 2a$. Не ограничивая общности, можно считать, что работы $n+1$ и $n+3$ выполняются на машине 1, и пусть I_1 — множество работ, выполняемых на этой машине. Положим $A' = I_1 \setminus \{n+1, n+3\}$. Так как $\prod_{i=1}^{n+2}(v_i+1) = 4a^2$, то из леммы 2 следует, что все машины выполняют работы без простоев в промежутке времени $[1, 2a]$. Тогда $2a = \overline{T}_1 = 2a/b \prod_{i\in A'} \eta(i)$. Следовательно, $\prod_{i\in A'} \eta(i) = b$ и множество A' искомое. Теорема 4 доказана.

§ 4. Многооперационные системы

Установим NP-трудность ряда задач построения оптимального расписания работ в многооперационных системах. Сначала рассмотрим задачи с системами потокового типа.

Теорема 5. *Задача $F3|vt|C_{\max}$ является NP-трудной в сильном смысле.*

Доказательство. Укажем сведение задачи 4-P к $F3|vt|C_{\max}$. Положим $|I| = 5n+1$. Обозначим через $v_{i,j}$ скорость роста длительности операции работы $i \in I$ на машине j, $1 \leqslant j \leqslant 3$.

Положим

$$\begin{array}{llll}
v_{i,1} = 0, & v_{i,2} = \eta(i) - 1, & v_{i,3} = 0, & 1 \leqslant i \leqslant 4n; \\
v_{4n+1,1} = 0, & v_{4n+1,2} = 0, & v_{4n+1,3} = 2b - 1; & \\
v_{4n+2,1} = b - 1, & v_{4n+2,2} = 1, & v_{4n+2,3} = 2b - 1; & \\
v_{4n+i,1} = 2b - 1, & v_{4n+i,2} = 1, & v_{4n+i,3} = 2b - 1, & 3 \leqslant i \leqslant n - 1; \\
v_{5n,1} = 2b - 1, & v_{5n,2} = 1, & v_{5n,3} = b - 1; & \\
v_{5n+1,1} = 2b - 1, & v_{5n+1,2} = 0, & v_{5n+1,3} = 0. &
\end{array}$$

Покажем, что в этой задаче оптимальное расписание не превосходит величины $y = 2^{n-1}b^n$ тогда и только тогда, когда существует разбиение множества A на n непересекающихся подмножеств $A_1, A_2, \dots, A_n$ таких, что $\prod\limits_{i \in A_k} \eta(i) = b$ для $1 \leqslant k \leqslant n$.

1. Пусть такое разбиение существует. Построим расписание s следующим образом. На первой машине выполняются в порядке возрастания номеров работы с номерами от $4n + 2$ до $5n + 1$. На второй машине работы выполняются в последовательности $(\pi_k, 4n + k + 1)$, $1 \leqslant k \leqslant n$, где π_k — произвольная перестановка работ с номерами из множеств A_k. На третьей машине выполняются в порядке возрастания номеров работы с номерами от $4n + 1$ до $5n$. Легко проверить, что в построенном расписании машины работают с момента времени 1 до момента времени $2^{n-1}b^n$ без простоев. Следовательно, расписание s оптимально и не превосходит величины $y = 2^{n-1}b^n$.

2. Пусть существует расписание s такое, что $C(s) \leqslant y$. Так как длительность выполнения всех операций на первой, второй, и третьей машинах соответственно равна

$$V_1 = \prod_{i=1}^{n} (v_{i,1}^2 + 1) = 2^{n-1}b^n,$$

$$V_2 = \prod_{i=1}^{n} (v_{i,2}^2 + 1) \prod_{i=1}^{4n} (v_{i,2}^1 + 1) = 2^{n-1}b^n,$$

$$V_3 = \prod_{i=0}^{n-1} (v_{i,3}^2 + 1) = 2^{n-1}b^n,$$

то по лемме 2 все машины работают без простоев с момента времени 1 до момента времени $2^{n-1}b^n$. Покажем, что машина 3 работает без простоя только в том случае, если она выполняет работы в следующем порядке: первой выполняется работа $4n+1$, затем $4n+2$, затем работы с номерами от $4n + 3$ до $5n - 1$ и последней — работа $5n$. Так как длительности операций работ с номерами от $4n+3$ до $5n-1$ одинаковы, то можно считать, что они всегда выполняются в порядке возрастания номеров.

Обозначим через $t_{i,j}$ время начала выполнения работы i на машине j, а через $\bar{t}_{i,j}$ — время завершения выполнения работы i на машине j. В момент времени 1 к выполнению на машине 3 готова только работа $4n+1$. Тогда $t_{4n+1,3} = 1$, $\bar{t}_{4n+1,3} = 2b$. К этому времени только работу $4n+2$ можно успеть выполнить на двух первых машинах. Следовательно, $t_{4n+2,1} = 1$, $\bar{t}_{4n+2,1} = t_{4n+2,2} = b$, $\bar{t}_{4n+2,2} = t_{4n+2,3} = 2b$, $\bar{t}_{4n+2,3} = 4b^2$. К моменту времени $4b^2$ только одну из работ с номерами от $4n+3$ до $5n-1$ можно успеть выполнить на двух первых машинах. Следовательно, получим $t_{4n+3,1} = b$, $\bar{t}_{4n+3,1} = t_{4n+3,2} = 2b^2$, $\bar{t}_{4n+3,2} = t_{4n+3,3} = 4b^2$, $\bar{t}_{4n+3,3} = 8b^3$.

Повторив это рассуждение для оставшихся работ, получаем $t_{4n+i,1} = 2^{i-3}b^{i-2}$, $\bar{t}_{4n+i,1} = t_{i,2} = 2^{i-2}b^{i-1}$, $\bar{t}_{i,2} = t_{i,3} = 2^{i-1}b^{i-1}$, $\bar{t}_{i,3} = 2^i b^i$, $4 \leqslant i \leqslant n-1$, $t_{5n,1} = 2^{n-3}b^{n-2}$, $\bar{t}_{5n,1} = t_{5n,2} = 2^{n-2}b^{n-1}$, $\bar{t}_{5n,2} = t_{5n,3} = 2^{n-1}b^{n-1}$, $\bar{t}_{5n,3} = 2^{n-1}b^n$. Из приведенных рассуждений следует, что на машине 2 работы с номерами от 1 до n выполняются в промежутки времени $[2^{i-1}b^{i-1}; 2^{i-1}b^i]$, $1 \leqslant i \leqslant n$. Пусть A_i — множество работ, выполняемых в промежуток времени $[2^{i-1}b^{i-1}; 2^{i-1}b^i]$. Тогда из леммы 2 следует, что

$$b = \prod_{j\in A_i} (v_{j,2}^2 + 1) = \prod_{j\in A_i} \eta(j),$$

и набор множеств A_i задает искомое разбиение. Теорема 5 доказана.

Теорема 6. *Задача* $F2|vt|L_{\max}$ *является* NP-*трудной.*

Доказательство. Построим сведение задачи 4-P к $F2|vt|L_{\max}$. Положим $|I| = 5n-1$,

$$\begin{array}{llll}
v_{i,1} = 0, & v_{i,2} = \eta(i) - 1, & d_i = 2^{n-1}b^n, & 1 \leqslant i \leqslant 4n;\\
v_{4n+1,1} = b-1, & v_{4n+1,2} = 1, & d_{4n+1} = 2b; & \\
v_{4n+i,1} = 2b-1, & v_{4n+i,2} = 1, & d_{4n+i} = 2^i b^i, & 2 \leqslant i \leqslant n-1.
\end{array}$$

Покажем, что в этой задаче существует расписание с целевой функцией $L(s) = 0$ тогда и только тогда, когда существует разбиение множества A на n непересекающихся подмножеств $A_1, A_2, \ldots, A_n$ таких, что $\prod\limits_{i\in A_k} \eta(i) = b$ для $1 \leqslant k \leqslant n$.

1. Пусть такое разбиение существует. Рассмотрим следующее расписание. На первой машине выполняются работы с номерами от $4n+1$ до $5n-1$ в порядке возрастания номеров. На второй машине работы выполняются в последовательности $(\pi_k, 4n+k)$, $1 \leqslant k \leqslant n$, где π_k — произвольная перестановка работ с номерами из множеств A_k. Нетрудно убедиться, что такое расписание является допустимым.

2. Пусть существует расписание s такое, что $L(s) = 0$. Так как $\max_{i\in I} d_i = 2^{n-1}b^n$ и $\prod_{i=1}^{5n-1}(v_{i,2}+1) = 2^{n-1}b^n$, то машина 2 работает без простоев в промежутке времени $[1, 2^{n-1}b^n]$. Так как работа $4n+1$ заканчивается до директивного срока $d_{4n+1} = 2b$, то она выполняется без задержек в промежутке времени $[1, 2b]$. Следовательно, $t_{1,1} = 1$, $\bar{t}_{1,1} = b$, $t_{1,2} = b$, $\bar{t}_{1,2} = 2b$. Работа $4n+i$, $2 \leqslant i \leqslant n-1$ заканчивается до директивного срока $d_{4n+i} = 2^i b^i$, и аналогично изложенному выше на машине 1 она начинает выполняться не позже момента $t_{i,1} = 2^{i-2}b^{i-1}$. С другой стороны, выполнение данной работы не может начаться раньше этого срока, так как на машине 1 в это время выполняется работа $i-1$. Следовательно, $t_{i,1} = 2^{i-2}b^{i-1}$, $\bar{t}_{i,1} = t_{i,2} = 2^{i-1}b^i$, $\bar{t}_{i,2} = 2^i b^i$, $2 \leqslant i \leqslant n-1$.

Из приведенных выше рассуждений следует, что на машине 2 работы с номерами от 1 до n выполняются в промежутки времени $[2^{i-1}b^{i-1}; 2^{i-1}b^i]$, $1 \leqslant i \leqslant n$. Пусть A_i — множество работ, выполняемых в промежуток времени $[2^{i-1}b^{i-1}; 2^{i-1}b^i]$. Тогда из леммы 2 следует, что

$$b = \prod_{j\in A_i}(v_{j,2}^2 + 1) = \prod_{j\in A_i}\eta(j)$$

и набор множеств A_i задает искомое разбиение. Теорема 6 доказана.

Перейдем к рассмотрению задач с системами открытого типа.

Теорема 7. *Задача $O3|vt|C_{max}$ является* NP*-трудной.*

Доказательство. Построим сведение задачи SP к $O3|vt|C_{\max}$. Положим $|I| = n+3$, $v_{i,j} = \eta(i)-1$, $1 \leqslant i \leqslant n$, $v_{n+1,j} = 2a/b-1$, $v_{n+2,j} = 2b-1$, $v_{n+3,j} = 2a-1$, $1 \leqslant j \leqslant 3$. Покажем, что в построенной задаче расписание s, при котором $C(s) \leqslant 8a^3$, существует тогда и только тогда, когда в A есть подмножество $A' \subseteq A$ такое, что произведение размеров его элементов равно b.

1. Пусть такое подмножество существует. Тогда расписание s может быть построено следующим образом. Сгруппируем работы в три множества: $N_1 = \{i | i \in A'\} \cup \{n+1\}$, $N_2 = \{n+3\}$, $N_3 = \{i | i \in A \setminus A'\} \cup \{n+2\}$. На первой машине работы выполняются в последовательности (π_1, π_2, π_3), на второй — (π_2, π_3, π_1) и на третьей машине — в последовательности (π_3, π_1, π_2), где π_k — произвольная перестановка работ из множества N_k, $1 \leqslant k \leqslant 3$. Так как $\prod_{i\in N_k} v_{i,j} + 1 = 2a$, $1 \leqslant j \leqslant 3$, то все машины работают без простоев с момента времени 1 до момента времени $8a^3$ и $C(s) \leqslant 8a^3$.

2. Пусть существует расписание s такое, что $C(s) \leqslant 8a^3$. Так как $\prod_{i=1}^{n+3} v_{i,j} + 1 = 8a^3$, $1 \leqslant j \leqslant 3$, то по лемме 2 все машины работают без простоев с момента времени 1 до момента времени $8a^3$. Так как выполнение

работы $n+3$ в расписании s завершается к моменту времени $8a^3$, то она выполняется с момента времени 1 без задержек. Следовательно, существует машина, которая выполняет работу $n+3$ в промежутке времени $[2a, 4a^2]$. Без ограничения общности можно считать, что это машина 1. Тогда остальные работы выполняются на ней в промежутке времени $[1, 2a]$, $[4a^2, 8a^3]$. Так как $(v_{n+1,1}+1)(v_{n+2,1}+1) = 4a > 2a$, то работы $n+1$ и $n+2$ на машине 1 выполняются в разные промежутки времени. Пусть A' — множество номеров работ, которые выполняются в том же промежутке, что и работа $n+1$. Так как машина работает без простоев, то по лемме 2 имеем

$$2a = (v_{n+1,1}+1)\prod_{i\in A'}(v_{i,1}+1) = 2\frac{a}{b}\prod_{i\in A'}\eta(i).$$

Следовательно, $\prod\limits_{i\in A'}\eta(i) = b$ и множество A' искомое. Теорема 7 доказана.

Теорема 8. *Задача* $O2|vt|L_{\max}$ *является* NP*-трудной.*

Доказательство. Построим сведение задачи SP к $O2|vt|L_{max}$. Положим $T=1$, $I=|n+3|$, $v_{i,j}=\eta(i)-1$, $1\leqslant i\leqslant n$, $v_{n+1,j}=2a/b-1$, $v_{n+2,j}=2b-1$, $v_{n+3,j}=2a-1$, $1\leqslant j\leqslant 2$, $d_i=8a^3$, $1\leqslant i\leqslant n+2$, $d_{n+3}=4a^2$.

Покажем, что в построенной задаче существует расписание s, не нарушающее директивных сроков тогда и только тогда, когда в A есть подмножество $A'\subseteq A$ такое, что произведение размеров его элементов равно b.

1. Пусть такое подмножество существует. Построим расписание s следующим образом. Сгруппируем работы в три множества: $N_1=\{i|i\in A'\}\cup\{n+1\}$, $N_2=\{n+3\}$, $N_3=\{i|i\in A\setminus A'\}\cup\{n+2\}$. На первой машине работы выполняются в последовательности (π_1,π_2,π_3), на второй — в последовательности (π_2,π_3,π_1), где π_k — произвольная перестановка работ из множества N_k, $1\leqslant k\leqslant 3$. Так как $\prod\limits_{i\in N_k}(v_{i,j}+1)=2a$, $1\leqslant j\leqslant 3$, то в построенном расписании все машины работают без простоев в промежутке $[1, 8a^3]$ и выполнение всех работ завершается до директивных сроков.

2. Пусть существует расписание s такое, что $L(s)=0$. Так как выполнение работы $n+3$ в расписании s завершается к моменту времени $4a^2$, то она выполняется с момента времени 1 без задержек. Следовательно, существует машина, которая выполняет работу $n+3$ в промежуток времени $[2a, 4a^2]$. Без ограничения общности можно считать, что это машина 1. Тогда остальные работы выполняются на ней в промежутки времени $[1, 2a]$, $[4a^2, 8a^3]$. Так как $(v_{n+1,1}+1)(v_{n+2,1}+1)=4a>2a$, то работы $n+1$ и $n+2$ на машине 1 выполняются в разные промежутки времени. Пусть A' — множество номеров работ, которые выполняются

в том же промежутке, что и работа $n+1$. Так как машина 1 работает без простоев, то из леммы 2 следует, что

$$2a = (v_{n+1,1}+1)\prod_{i\in A'}(v_{i,1}+1) = 2\frac{a}{b}\prod_{i\in A'}\eta(i).$$

Поэтому $\prod\limits_{i\in A'}\eta(i) = b$ и множество A' искомое. Теорема 8 доказана.

Отметим, что для большинства NP-трудных задач с простым линейным ростом длительностей NP-трудны и аналогичные задачи с постоянными длительностями. Исключением является задача с критерием ΣC на параллельных машинах, которая полиномиально-разрешима в классическом случае и NP-трудна для задач с простым линейным ростом длительностей.

§ 5. Полиномиально-разрешимые случаи

Как было отмечено выше, для большинства критериев результаты по NP-трудности задач с постоянными длительностями переносятся на задачи с простым линейным ростом длительностей. Аналогичная картина наблюдается и для некоторых полиномиально-разрешимых задач. Для каждой из таких задач можно построить полиномиальный алгоритм, в некотором смысле эквивалентный алгоритму для решения такой же задачи с постоянными длительностями. Ниже делается попытка предложить некоторый метод, позволяющий по уже существующему алгоритму построить другой алгоритм, решающий эквивалентную задачу.

Пусть Π — оптимизационная задача, I_Π — индивидуальная задача, s_i — допустимое решение задачи I_Π, $1 \leqslant i \leqslant n$, и $f(s_i)$ — значение целевой функции на допустимом решении s_i. Пусть $\gamma(x) : R \to R$ — строго возрастающая функция.

Определение 1. Будем говорить, что задача Π изоморфна задаче $\overline{\Pi}$ относительно функции γ, если

1) для любой индивидуальной задачи I_Π существует индивидуальная задача $I_{\overline{\Pi}}$ такая, что для любого допустимого решения s_i задачи I_Π решение $\overline{s}_i = \gamma(s_i)$ является допустимым для задачи $I_{\overline{\Pi}}$ и для любого допустимого решения $\overline{s}_i$ задачи $I_{\overline{\Pi}}$ решение $s_i = \gamma^{-1}(\overline{s}_i)$ является допустимым решением задачи I_Π;

2) $\bar{f}(\gamma(s_i)) = \gamma(f(s_i))$ для любого допустимого решения s_i задачи I_Π.

Индивидуальные задачи, описанные в определении, будем называть *изоморфными*. Рассмотрим задачи $F2||C_{\max}$ $O2||C_{\max}$.

Лемма 6. 1. *Относительно функции* $\gamma = 2^x$ *задача* $F2|vt|C_{\max}$ *изоморфна задаче* $F2||C_{\max}$.

2. *Относительно функции* $\gamma = 2^x$ *задача* $O2|vt|C_{\max}$ *изоморфна задаче* $O2||C_{\max}$.

Доказательство. Докажем пункт 1. Числовыми параметрами, характеризующими каждую индивидуальную задачу с постоянными длительностями, являются длительности выполнения операций $\overline{p}_i$. В задачах с растущими длительностями числовыми параметрами являются величины v_i. Обозначим через Ω и $\overline{\Omega}$ множества всех операций в задачах $F2||C_{\max}$ и $F2|vt|C_{\max}$ соответственно. Пусть $\overline{s}_i$, $i \in \overline{\Omega}$, — допустимое расписание некоторой индивидуальной задачи $I_{\overline{\Pi}}$. Построим индивидуальную задачу I_{Π}, изоморфную индивидуальной задаче $I_{\overline{\Pi}}$. Положим $v_i = 2^{\overline{p}_i} - 1$. Убедимся, что в построенной задаче $s_i = 2^{\overline{s}_i}$ является допустимым решением. Обозначим через τ_i и $\overline{\tau}_i$ момент завершения операций в задачах I_{Π} и $I_{\overline{\Pi}}$ соответственно:

$$\tau_i = s_i + s_i v_i = 2^{\overline{s}_i} 2^{\overline{p}_i} = 2^{\overline{s}_i + \overline{p}_i} = 2^{\overline{\tau}_i}. \tag{1}$$

Для допустимости решения необходимо выполнение неравенств, связанных с величинами s_i и τ_i. Поэтому из (1) и монотонности функции 2^x следует, что если $\overline{s}_i$ — допустимое решение в задаче $I_{\overline{\Pi}}$, то s_i — допустимое решение в задаче I_{Π}. Аналогично проверяется и обратное утверждение. Совпадение целевых функций следует из равенства $\max\limits_{i \in I} 2^{\overline{\tau}_i} = 2^{\max_{i \in I} \overline{\tau}_i}$. Второе утверждение леммы 6 доказывается аналогично. Лемма 6 доказана.

Следствие 1. *Если $\overline{s}_i$ — оптимальное решение в задаче $I_{\overline{\Pi}}$, то s_i — оптимальное решение в задаче I_{Π}, изоморфной задаче $I_{\overline{\Pi}}$.*

В работах [10, 11] для точного решения задач $F2||C_{\max}$, $O2||C_{\max}$ приведены полиномиальные алгоритмы с трудоемкостью $O(n \log n)$ и $O(n)$ соответственно. Обозначим их через $\overline{AF}$ и $\overline{AO}$ соответственно. Обозначим длительности операций работы i на первой и второй машинах через a_i и b_i.

Алгоритм $\overline{AF}$.

Шаг 1. Множество работ I разбивается на два подмножества $I_1 = \{i | a_i \leqslant b_i\}$ и $I_2 = \{i | a_i > b_i\}$.

Шаг 2. Находится перестановка π, в которой сначала идут операции множества I_1 в порядке неубывания значений a_i, а затем операции множества I_2 в порядке неубывания значений b_i.

Шаг 3. Находится расписание, в котором операции выполняются в порядке, заданном перестановкой π на обеих машинах.

Конец.

Алгоритм $\overline{AO}$.

Шаг 1. Множество работ I разбивается на два подмножества $I_1 = \{i|a_i \leqslant b_i\}$ и $I_2 = \{i|a_i > b_i\}$.

Шаг 2. Среди операций множества I_1 выбирается операция j_1 такая, что $b_{j_1} \geqslant \max\limits_{i\in I_1} a_i$, а среди операций множества I_2 выбирается операция j_2 такая, что $a_{j_2} \geqslant \max\limits_{i\in I_1} b_i$.

Шаг 3. Вычисляются $\alpha = \sum\limits_{i\in I} a_i$, $\beta = \sum\limits_{i\in I} b_i$.

Шаг 4. В силу симметрии задачи можно считать, что $\alpha + b_{j_2} \leqslant \beta + a_{j_1}$. Находится перестановка $\pi_a = (\pi_{I_1\setminus j_1}, \pi_{I_2\setminus j_2}, j_2, j_1)$ и $\pi_b = (j_1, \pi_{I_1\setminus j_1}, \pi_{I_2\setminus j_2}, j_2)$, где $\pi_{I_1\setminus j_1}$ — произвольная перестановка из $I_1 \setminus j_1$, а $\pi_{I_2\setminus j_2}$ — произвольная перестановка из $I_2 \setminus j_2$.

Шаг 5. Строится расписание, в котором операции на машине a выполняются в порядке, заданном перестановкой π_a, а на машине b — перестановкой π_b.

Конец.

В обоих алгоритмах операция сразу ставится на выполнение, если все предшествующие операции выполнены и машина свободна.

Обозначим через v_i^a и v_i^b скорости роста длительностей операций на первой и второй машинах в задачах $F2|vt|C_{\max}$ и $O2|vt|C_{\max}$. Алгоритмы AO и AF работают следующим образом. В алгоритмах $\overline{AO}$ и $\overline{AF}$ величины a_i заменяются на $\alpha_i = v_i^a + 1$, b_i заменяются на $\beta_i = v_i^b + 1$ и каждая операция сложения — на операцию умножения.

Алгоритм AF.

Шаг 1. Множество работ I разбивается на два подмножества $I_1 = \{i|\alpha_i \leqslant \beta_i\}$ и $I_2 = \{i|\alpha_i > \beta_i\}$.

Шаг 2. Находится перестановка π, в которой сначала идут операции множества I_1 в порядке неубывания значений α_i, а затем операции множества I_2 в порядке неубывания значений β_i.

Шаг 3. Находится расписание, в котором операции выполняются в порядке, заданном перестановкой π на обеих машинах.

Конец.

Алгоритм AO.

Шаг 1. Множество работ I разбивается на два подмножества $I_1 = \{i|\alpha_i \leqslant \beta_i\}$ и $I_2 = \{i|\alpha_i > \beta_i\}$.

Шаг 2. Среди операций множества I_1 выбирается операция j_1 такая, что $\beta_{j_1} \geqslant \max\limits_{i\in I_1} \alpha_i$, а среди операций множества I_2 выбирается операция j_2 такая, что $\alpha_{j_2} \geqslant \max\limits_{i\in I_1} \beta_i$.

Шаг 3. Вычисляются $\alpha = \prod\limits_{i\in I} \alpha_i$, $\beta = \prod\limits_{i\in I} \beta_i$.

Шаг 4. В силу симметрии задачи можно считать, что $\alpha\beta_{j_2} \leqslant \beta\alpha_{j_1}$. Находятся перестановки $\pi_a = (\pi_{I_1\setminus j_1}, \pi_{I_2\setminus j_2}, j_2, j_1)$ и $\pi_b = (j_1, \pi_{I_1\setminus j_1}, \pi_{I_2\setminus j_2}, j_2)$, где $\pi_{I_1\setminus j_1}$ — произвольная перестановка из $I_1 \setminus j_1$, а $\pi_{I_2\setminus j_2}$ — произвольная перестановка из $I_2 \setminus j_2$.

Шаг 5. Строится расписание, в котором операции на машине a выполняются в порядке, заданном перестановкой π_a, а на машине b — перестановкой π_b.

Конец.

Пусть I_Π — произвольная индивидуальная задача $F2|vt|C_{\max}$, а $I_{\overline{\Pi}}$ — изоморфная ей индивидуальная задача $F2||C_{\max}$. Алгоритм $\overline{AF}$ находит оптимальное решение $\overline{s}_i$ задачи $F2||C_{\max}$. Покажем, что алгоритм AF находит решение $s_i = 2^{\overline{s}_i}$, которое, как было отмечено выше, также является оптимальным. Действительно, в начальный момент значению каждого числового параметра x в задаче $I_{\overline{\Pi}}$ соответствует значение 2^x в задаче I_Π. Так как функция 2^x строго возрастающая, операция сравнения двух величин дает одинаковый результат. Пусть на некотором шаге в алгоритме $\overline{AF}$ вычисляется число $\overline{x} = x_1 + x_2$. Тогда в алгоритме AF на этом шаге получается число $x = 2^{x_1}2^{x_2} = 2^{x_1+x_2} = 2^x$. Следовательно, алгоритм AF получит решение $s_i = 2^{\overline{s}_i}$. Отметим, что замена одной элементарной операции на другую, а именно сложения на умножение, не увеличивает трудоемкости алгоритма. Проведя аналогичные рассуждения и для алгоритма AO, получаем следующее утверждение.

Теорема 9. 1. *Алгоритм AF находит оптимальное расписание задачи $F2|vt|C_{\max}$ с трудоемкостью $O(n\log n)$.*

2. *Алгоритм AO находит оптимальное расписание задачи $O2|vt|C_{\max}$ с трудоемкостью $O(n)$.*

§ 6. Заключение

Из полученных результатов следует, что задачи с простым линейным ростом длительностей по своей структуре сходны с классическими задачами теории расписаний. Поэтому для критерия минимизации длины расписания следует ожидать эквивалентных полиномиальных алгоритмов и для других постановок задач с простым линейным ростом длительностей, если такие существуют в аналогичных классических задачах. По-видимому, следует ожидать такие результаты для задачи построения расписания на двух машинах с системой рабочего типа и задачи построения расписания на двух машинах с системой, в которой присутствуют работы из системы как потокового типа, так и открытого.

ЛИТЕРАТУРА

1. **Mosheiov G.** Scheduling jobs under simple linear deterioration // Comput. Oper. Res. 1994. V. 21, N 6. P. 653–659.
2. **Mosheiov G.** V-shaped policies for scheduling deteriorating jobs // Oper. Res. 1991. V. 39, N 6. P. 979–991.
3. **Browne S., Yechiali U.** Scheduling deterioration jobs on a single processor // Oper. Res. 1990. V. 38, N 3. P. 495–498.
4. **Kunnarthur A. S., Gupta S. K.** Minimizing the makespan with late start penalties added to processing times in a single facility scheduling problem // Europian J. Oper. Res. 1990. V. 47, N 1. P. 56–64.
5. **Кононов А. В.** О расписаниях работ на одной машине с длительностями, нелинейно зависящими от времени // Дискрет. анализ и исслед. операций. 1995. Т. 2, № 1. С: 21–35.
6. **Conway R. W., Maxwell W. L., Miller L. W.** Theory of Scheduling. Reading, MA: Addison-Wesley, 1967.
7. **Karp R. M.** Reducibility among combinatorial problems // Complexity of Computer Computations. New York: Plenum Press, 1972. P. 85–103.
8. **Гэри М., Джонсон Д.** Вычислительные машины и труднорешаемые задачи. М.: Мир, 1982.
9. **Бухштаб А. А.** Теория чисел. М.: Учпедгиз, 1960.
10. **Jonson S. M.** Optimal two and three stage production schedules with set up times included // Naval. Res. Logist. Quart. 1954. V. 1, N 1. P. 61–68.
11. **Graham R. L., Lawler E. L., Lenstra J. K., Rinnooy Kan A. H. G.** Optimization and approximation in deterministic sequencing and scheduling: a survey // Ann. Discrete Math. Amsterdam: Horth-Holland, 1979. V. 5. P. 287–326.

Адрес автора:

Россия,
630090 Новосибирск,
Университетский пр., 4,
Институт математики
им. С. Л. Соболева СО РАН
E-mail: alvenko@math.nsk.su

Статья поступила

12 марта 1996 г.