Управляемые системы, ИМ СО АН СССР,1979 г.,выпуск 19.
УДК 519.854

ЗАДАЧА ВЫБОРА ОБОРУДОВАНИЯ МНОГОРАЗОВОГО ИСПОЛЬЗОВАНИЯ. I

А.С.Кузнецов (Новосибирск)

§1.Постановка задачи

Рассматривается задача выбора оборудования многоразового использования, иначе, задача оптимального выбора и использования парка машин с учетом динамики спроса. Содержательная постановка ее сводится к следующему. Задана совокупность объектов, для каждого из которых ежегодно на планируемый период подаются заявки на выполнение некоторых работ. Каждая работа может быть выполнена с помощью оборудования различных типоразмеров, известны затраты на покупку и эксплуатационные расходы для каждого из них. В конце каждого года можно перераспределять ранее приобретенное оборудование между объектами. При этом транспортные расходы и затраты на монтаж-демонтаж оборудования предполагаются пренебрежимо малыми.

Задача заключается в том, чтобы определить такой парк машин (оборудования) и политику их использования, чтобы расходы на приобретение оборудования и выполнение заявленных работ на весь период планирования были минимальными.

Введем обозначения:

p - продолжительность периода планирования, $P=\{1,\dots,p\}$;
n - число объектов, $N=\{1,\dots,n\}$;
m - число типоразмеров оборудования, $M=\{1,\dots,m\}$;
c_i - стоимость комплекта оборудования i-го типоразмера,
g_{ij}^t - эксплуатационные расходы для оборудования i-го типа на j-м объекте в год t;
y_i - максимальное число одновременно используемых на объектах комплектов оборудования i-го типоразмера;

$$x_{ij}^t = \begin{cases} 1, & \text{если на } j\text{-м объекте в год } t \text{ используется оборудование } i\text{-го типоразмера,} \\ 0 & \text{- в противном случае.} \end{cases}$$

В этих обозначениях математическая модель задачи представляется в виде

$$\sum_{i\in M} c_i y_i + \sum_{i\in M}\sum_{j\in N}\sum_{t\in P} g_{ij}^t x_{ij}^t \longrightarrow \min_{(x_{ij}^t)(y_i)} \qquad /1.1/$$

при ограничениях:

$$\sum_{i\in M} x_{ij}^t = 1 \quad (j\in N,\ t\in P); \qquad /1.2/$$

$$\sum_{j\in N} x_{ij}^{t} \leqslant y_i \qquad (i\in M,\ t\in P); \qquad /I.3/$$

$$x_{ij}^{t} \in \{0,1\} \qquad (i\in M,\ j\in N,\ t\in P). \qquad /I.4/$$

Аналогичными моделями описываются, например, задачи выбора шахтных вентиляторов местного и главного проветривания (при условии заменяемости последних)[I]. Частным случаем задачи /I.I/ - /I.4/ (при $n=1$) является задача стандартизации, которая, как известно [2] , относится к классу полиноминально полных проблем.

Для решения сформулированной задачи разработан приближенный алгоритм с оценкой трудоемкости полиноминального вида. Этот алгоритм проверялся на задачах размерности $(m\times n\times p)=(15\times 40\times 10)$. Значения g_{ij}^{t} и c_i генерировались случайным образом из интервалов (0,I00) и (I0,$\bar{c}$) соответственно. Для $\bar{c}$ задавались значения I00 и I50. Для определения нижней оценки целевой функции задачи /I.I/ - /I.4/ использовался алгоритм решения двойственной задачи к /I.I/ - /I.3/ при $x_{ij}^{t}\geqslant 0$, основанный на идее групповой оптимизации. Всего было решено около ста задач указанных классов, и для каждой из задач относительная погрешность не превосходила 0,0I3.

§ 2. Алгоритм T_1 решения транспортной задачи специального вида

При построении алгоритма решения задачи /I.I/ - /I.4/,излагаемого в § 4-5, используется тот факт, что при фиксированном векторе (y_i) исходная задача распадается на последовательность транспортных задач следующего вида:

$$\sum_{i\in M}\sum_{j\in N} g_{ij}x_{ij} \longrightarrow \min_{(x_{ij})}; \qquad /2.I/$$

$$\sum_{i\in M} x_{ij} = 1 \qquad (j\in N); \qquad /2.2/$$

$$\sum_{j\in N} x_{ij} \leqslant y_i \qquad (i\in M); \qquad /2.3/$$

$$x_{ij} \geqslant 0 \qquad (i\in M,\ j\in N); \qquad /2.4/$$

$$x_{ij} \text{ - целые.} \qquad /2.5/$$

В работе [3] рассматривается задача, отличающаяся от /2.I/ - /2.4/ тем, что вместо условий /2.3/ в ней используются ограничения равенства.Ниже мы покажем, что алгоритм T_1 , описанный в [3], позволяет также находить оптимальное целочисленное решение задачи /2.I/ - /2.4/ и решение двойственной к ней задачи, имеющей вид

$$\sum_{j\in N} v_j - \sum_{i\in M} y_i u_i \longrightarrow \max_{(u_i)(v_j)}; \qquad /2.6/$$

$$v_j \leq g_{ij} + u_i \quad (i \in M,\ j \in N); \qquad /2.7/$$

$$u_i \geq 0 \qquad (i \in M). \qquad /2.8/$$

Нетрудно заметить, что в оптимальном решении последней переменные v_j должны удовлетворять соотношениям

$$v_j = \min_{i \in M} (g_{ij} + u_i) \quad (j \in N),$$

т.е. фактически они могут быть определены через компоненты вектора (u_i) $(i \in M)$.

Пусть дан вектор $u = (u_i)$ и матрица (g_{ij}) .

Определение 2.I. Элемент g_{ij} матрицы (g_{ij}) будем называть <u>u -минимальным</u> в столбце j, если для всех $k \in M$ выполняется $g_{ij} + u_i \leq g_{kj} + u_k$.

Пусть (x_{ij}) - булева матрица.

Определение 2.2. u - минимальный элемент g_{ij} матрицы (g_{ij}) , для которого $x_{ij} = 1$, $\sum_{k \in M} x_{kj} = 1$, назовем <u>$u$ -основой j -го столбца</u>. Через $i(j)$ обозначим номер строки, в которой находится основа столбца j .

Определение 2.3. Булеву матрицу (x_{ij}) назовем <u>соответствующей вектору</u> (u_i) , если для всякой пары (i,j) такой, что $x_{ij} = 1$, элемент g_{ij} является u -минимальным в своем столбце.

Определение 2.4. <u>Дефектом</u> булевой матрицы (x_{ij}) назовем величину $z = \sum_{i \in M} max(0, b_i - y_i)$, где $b_i = \sum_{j \in N} x_{ij}$ $(i \in M)$.

Пусть (k, i, j) - тройка индексов, где j - номер столбца матрицы (g_{ij}) , k и i - номера строк этой матрицы, причем одна строка, а именно k -я, помечена. Пусть даны вектор (u_i) и булева матрица (x_{ij}) такие, что из двух u -минимальных элементов j -го столбца, а именно g_{ij} и g_{kj} , первый элемент является u -основой этого столбца.

Определение 2.5. Назовем (k, i, j) -заменой такое преобразование j -го столбца матрицы (x_{ij}) и номера помеченной строки, в результате которого в j -м столбце матрицы (x_{ij}) выполняется $x_{kj} = 1$, $x_{ij} = 0$ для всех $i \in M \setminus \{k\}$, а помеченной становится только i -я строка.

Таким образом, в результате (k, i, j) -замены меняется u -основа j -го столбца: место элемента g_{ij} ею становится элемент g_{kj} .

Алгоритм T_1 состоит из однотипных итераций, на каждой из которых по заданному вектору (u'_i) и соответствующей матрице (x'_{ij}) с дефектом $z' > 0$ находится новый вектор (u_i) и соответствующая ему матрица (x_{ij}) с дефектом $z = z' - 1$. На первой итерации полагаем $u_i = 0 \ (i \in M)$. Итерация состоит из двух этапов.

Этап I. Пусть вектору (u_i) соответствует матрица (x_{ij}) с дефектом $\varkappa' > 0$ и r - номер строки такой, что $b_r > y_r$, $\mathfrak{M} = \varnothing$.

Этап I состоит не более чем из m шагов вида $\mathfrak{M} := \mathfrak{M} \cup \{r\}$. Увеличим u_i $(i \in \mathfrak{M})$ на максимальное d такое, что матрица (x_{ij}) остается соответствующей новому вектору (u_i) $(i \in M)$. Для некоторого столбца s, содержащего основу в строке $i(s) \in \mathfrak{M}$, найдется такой u -минимальный элемент в строке $r \in M \setminus \mathfrak{M}$, что $x_{rs} = 0$. Положим $j_r = s$. Если $b_r > y_r$, то шаг повторяется. В противном случае переходим к этапу 2.

Этап 2. Пусть $k = r$ - номер, включенный в множество $\mathfrak{M}$ последним. Производим $(k, i(j_k), j_k)$ -замены до тех пор, пока не придем к строке с номером k такой, что $b_k > y_k$. Итерация закончена, при этом получена новая матрица (x_{ij}), для которой компоненты вектора (b_i) определены через компоненты вектора (b'_i) следующим образом:

$$b_i = \begin{cases} b'_i, & i \neq r, k, \\ b'_i + 1, & i = r, \\ b'_i - 1, & i = k. \end{cases}$$

Дефект новой матрицы (x_{ij}) равен $\varkappa' - 1$. Алгоритм прекращает свою работу, если получена матрица (x_{ij}) с нулевым дефектом.

Укажем способ вычисления d, позволяющий реализовать описанный алгоритм за $O(mn^2)$ действий. Рассмотрим k-й шаг первого этапа итерации. Пусть $\mathfrak{M}_k$ - множество номеров, включенных в $\mathfrak{M}$, $\mathfrak{N}_k = \{j / i(j) \in \mathfrak{M}_k\}$.

Требование, чтобы матрица (x_{ij}) после изменения u_i $(i \in \mathfrak{M}_k)$ на величину $d^{(k)}$ оставалась соответствующей новому вектору (u_i), согласно определениям 2.I и 2.3, математически записывается следующим образом:

$$g_{i(j)j} + u_{i(j)} + d^{(k)} \leq g_{ij} + u_i \quad (i \in M \setminus \mathfrak{M}_k,\ j \in \mathfrak{N}_k).$$

В силу того, что $d^{(k)}$ выбирается максимально возможным, имеем

$$d^{(k)} = \min_{i \in M \setminus \mathfrak{M}_k} \ \min_{j \in \mathfrak{N}_k} \delta_{ij} = \min_{i \in M \setminus \mathfrak{M}_k} q_i^{(k)},$$

где $\delta_{ij} = g_{ij} + u_i - (g_{i(j)j} + u_{i(j)})$ $(i \in M,\ j \in N)$,

$$q_i^{(k)} = \min_{j \in \mathfrak{N}_k} \delta_{ij} = \min(q_i^{(k-1)} - d^{(k-1)},\ \min_{j \in \mathfrak{N}_k \setminus \mathfrak{N}_{k-1}} \delta_{ij}).$$

На первом шаге $q_i^{(0)} = \infty$ $(i \in M)$, $d^{(0)} = 0$, $\mathfrak{N}_0 = \varnothing$.

Покажем теперь, что для получаемых с помощью алгоритма T_1 решений прямой /2.I/ - /2.4/ и двойственной /2.6/ - /2.8/ задач выполняются условия дополняющей нежесткости:

$$(\sum_{i \in M} x_{ij} - 1) v_j = 0 \quad (j \in N); \qquad /2.9/$$

$$(y_i - \sum_{j\in N} x_{ij}) u_i = 0 \qquad (i\in M); \qquad /2.10/$$

$$(g_{ij} + u_i - v_j) x_{ij} = 0 \qquad (i\in M,\ j\in N). \qquad /2.11/$$

Равенства /2.9/ очевидны. Докажем /2.10/. Пусть $\sum_j x_{ij} < y_i$. Тогда,заведомо, $u_i = 0$. Действительно, на каждой итерации увеличиваются значения только тех u_K $(K\in \mathfrak{M})$, для которых $\sum_j x_{Kj} > y_K$, причем изменение u_K заканчивается, как только в множество $\mathfrak{M}$ оказывается включенным некоторый номер i, для которого $\sum_j x_{ij} < y_i$, при этом начальное значение $u_i = 0$ остается неизменным. При $u_i > 0$ равенство $\sum_j x_{ij} = y_i$ следует из того, что на каждой итерации изменяются значения только двух компонент вектора (b_i), причем всегда уменьшается значение b_K только для такого номера K, для которого нарушено условие /2.3/.

Докажем равенства /2.11/. Поскольку каждая основа является u -минимальным элементом, ясно, что если по окончании работы алгоритма T_1 для некоторой пары (i,j) выполняется $(x_{ij}) = 1$, то $g_{ij} + u_i \leq g_{Kj} + u_K$ $(K\in M\setminus\{i\})$ и $v_j = g_{ij} + u_i$, так как по условиям задачи $v_j = \min_{K\in M}(g_{Kj} + u_K)$; и наоборот, если $u_j < g_{ij} + u_i$; то элемент g_{ij} не является u -основой j - го столбца,т.е. $x_{ij} = 0$.

Таким образом, доказана

Т е о р е м а 2.I. Алгоритм T_1 за $O(mn^2)$ действий дает оптимальное целочисленное решение задачи /2.I/ - /2.5/.

Пусть (x_{ij}) и (u_i) являются результатом работы алгоритма T_1 для некоторого вектора (y_i), причем матрица (x_{ij}) соответствует (u_i). Пусть после изменения значения одной из компонент вектора (y_i) на -I компоненты u_i получили приращения Δu_i и пусть $\bar{x}_{ij}$ - новые значения переменных прямой задачи. Тогда справедливо равенство

$$\sum_{j\in N} \min_{i\in M}(g_{ij} + u_i + \Delta u_i) - \sum_{j\in N} \min_{i\in M}(g_{ij} + u_i) = \sum_{i\in M} y_i \Delta u_i, \qquad /2.12/$$

которое понадобится нам при построении алгоритма решения задачи /I.I/ - /I.4/. Докажем его.

Имеем

$$\Delta\varphi = \sum_j \min_{i\in M}(g_{ij} + u_i + \Delta u_i) - \sum_j \min_{i\in M}(g_{ij} + u_i) =$$

$$= \sum_j \sum_i (g_{ij} + u_i + \Delta u_i)\bar{x}_{ij} - \sum_j \sum_i (g_{ij} + u_i) x_{ij}.$$

В силу того, что замена основ осуществляется при выполнении равенств $g_{i'j} + u_{i'} = g_{ij} + u_i$, где j - столбец, в котором осуществляется замена основы, g_{ij} и $g_{i'j}$ - старая и новая основы,можем написать

$$\Delta\varphi = \sum_i \sum_j (g_{ij}+u_i+\Delta u_i)x_{ij} - \sum_i \sum_j (g_{ij}+u_i)x_{ij} =$$
$$= \sum_i \Delta u_i \sum_j x_{ij} - \sum_{i/\Delta u_i>0} \Delta u_i y_i - \sum_{i\in M} y_i \Delta u_i ,$$

что и требовалось.

§3.Алгоритм T_2 решения задачи /2.I/ - /2.5/ при увеличении значения одной из компонент вектора (y_i).

Описанный в предыдущем пункте алгоритм T_1 за $O(mn)$ действий позволяет находить решение задачи /2.I/ - /2.5/ при уменьшении значения одной из переменных y_i на I (при условии, что известны решения (x_{ij}) и (u_i) при исходных значениях y_i). Однако в процессе решения задачи /I.I/ - /I.4/ значения y_i могут увеличиваться. Поэтому рассмотрим следующую задачу $\mathcal{P}$. Пусть мы имеем решение (u_i') задачи /2.6/ - /2.8/ и соответствующее решение (x_{ij}') задачи /2.I/ - /2.5/ для вектора (y_i') $(i\in M)$. Требуется найти решения (x_{ij}) и (u_i) для $y_i = y_i'$ $(i\in M\setminus\{i_1\})$ и $y_{i_1} = y_{i_1}' + 1$ $(u_{i_1}>0)$.

Приведем описание алгоритма T_2, позволяющего решить задачу $\mathcal{P}$ за $O(m^2+mn)$ действий. Алгоритм T_2 состоит из трех этапов, из которых первые два представляют собой незначительную модификацию алгоритма, описанного в работе [4].

Будем использовать определения и обозначения, введенные в предыдущем пункте, с той лишь разницей, что (k,w,j) -замену определим как последовательность операций $x_{kj}:=0$, $x_{wj}:=1$, $i(j):=w$, $k:=w$.

В начале работы алгоритма полагаем $\mathfrak{M}=\{i_1\}$.

Этап I состоит не более чем из m шагов следующего вида: уменьшим u_i $(i\in\mathfrak{M})$ на максимальное $d\leqslant u_i$ $(i\in\mathfrak{M})$ так, чтобы матрица (x_{ij}) соответствовала новым значениям u_i $(i\in M)$. В силу выбора d, либо для некоторого столбца s, содержащего основу в строке $i(s)\in M\setminus\mathfrak{M}$, найдется такой u -минимальный элемент в строке $l\in\mathfrak{M}$, что $x_{ls}=0$, либо для некоторого номера $i_0\in\mathfrak{M}$ будет выполняться равенство $u_{i_0}=0$. В последнем случае считаем, что номер i_0 включен в $\mathfrak{M}$ последним, и переходим к этапу 2. В противном случае полагаем $\mathfrak{M}:= \mathfrak{M}\bigcup\{i(s)\}$, $w_{i(s)}:=l$, $j_{i(s)}:=s$, и если $u_{i(s)}>0$, то шаг повторяется, если же $u_{i(s)}=0$, то переходим к этапу 2.

Этап 2. Пусть k - номер, включенный в $\mathfrak{M}$ последним. Производим последовательность (k, w_k, j_k) -замен до тех пор, пока не придем к номеру $k=i_1$.

Этап 3. Пусть $\mathcal{I}=\{i/u_i>0\}$, $\mathcal{J}=\{i/i(j)\in M\setminus\mathcal{I}\}$. Этап состоит из последовательности шагов следующего вида: уменьшим u_i $(i\in\mathcal{I})$ на максимальное $\hat{d}\leqslant u_i$ $(i\in\mathcal{I})$ такое, что матрица (x_{ij}) остается соответствующей новому вектору (u_i) $(i\in M)$. В силу выбора $\hat{d}$ выполняется один из двух случаев:

а) $u_i>0$ $(i\in \mathcal{I})$, и для некоторого $j_1\in \mathcal{J}$ существует номер $i_1\in \mathcal{I}$ такой, что $g_{i_1 j_1}+u_{i_1}=g_{i(j)j}+u_{i(j)}$ при $i(j)\in M\setminus \mathcal{I}$;

б) существует номер $i_1\in \mathcal{I}$, для которого $u_{i_1}=0$.

Исключим из $\mathcal{I}$ элемент i_1 и положим $u_{i_1}=0$. Если $\mathcal{I}\neq\varnothing$, то шаг повторяется, в противном случае алгоритм заканчивает свою работу.

Укажем способ вычисления $\bar{d}$. Рассмотрим K-й шаг первого этапа. Пусть $\mathfrak{M}_K$ - множество номеров, включенных в $\mathfrak{M}$, $\mathfrak{N}_K=\{j/i(j)\in M\setminus \mathfrak{M}_K\}$. Тогда

$$\bar{d}^{(K)}=\min\Big\{\min_{i\in\mathfrak{M}_K}u_i,\ \min_{j\in\mathfrak{N}_K}\min_{i\in\mathfrak{M}_K}\delta_{ij}\Big\}=$$

$$=\min\Big\{\check{u}^{(K)},\ \min_{j\in\mathfrak{N}_K}(q_j^{(K)}-g_{i(j)j}-u_{i(j)})\Big\},\quad \text{где}$$

$$\check{u}^{(K)}=\min_{i\in\mathfrak{M}_K}u_i=\min\Big\{u_{i_K},\ \check{u}^{(K-1)}-\bar{d}^{(K-1)}\Big\},$$

$$q_j^{(K)}=\min_{i\in\mathfrak{M}_K}(g_{ij}+u_i)=\min\Big\{q_j^{(K-1)}-\bar{d}^{(K-1)},\ g_{i_K j}+u_{i_K}\Big\},$$

i_K - номер, включенный в $\mathfrak{M}$ последним, $q_j^{(0)}=\check{u}^{(0)}=\infty$, $\bar{d}^{(0)}=0$. Использование рекуррентных соотношений дает возможность вычислить все значения $\bar{d}$ за $O(mn)$ операций.

Укажем формулу для вычисления $\hat{d}$. Рассмотрим K-й шаг третьего этапа. Пусть $\mathcal{I}_K$, $\mathcal{J}_K$ - множества $\mathcal{I}$ и $\mathcal{J}$ к моменту вычисления $\hat{d}^{(K)}$. Тогда

$$\hat{d}^{(K)}=\min\Big\{\min_{j\in\mathcal{J}_K}\min_{i\in\mathcal{I}_K}\delta_{ij},\ \min_{i\in\mathcal{I}_K}u_i\Big\}=$$

$$=\min_{i\in\mathcal{I}_K}\min\Big\{q_i^{(K)},\ u_i\Big\},\qquad \text{где}$$

$$q_i^{(K)}=\min_{j\in\mathcal{J}_K}\delta_{ij}=\min\Big\{q_i^{(K-1)}-\hat{d}^{(K-1)},\ \min_{j\in\mathcal{J}_K\setminus\mathcal{J}_{K-1}}\delta_{ij}\Big\},$$

$$q_i^{(0)}=\infty\ (i\in M),\quad \hat{d}^{(0)}=0,\quad \mathcal{J}_0=\varnothing.$$

Если элементы множеств $\{j/x_{ij}=1\}$ для каждого i такого, что $u_i>0$, выписаны в отдельные списки, то использование приведенного рекуррентного соотношения для вычисления q_i дает возможность реализовать третий этап за $O(m^2+mn)$ действий, что и составляет оценку трудоемкости алгоритма T_2.

Т е о р е м а 3.I. Результатом первых двух этапов алгоритма T_2 является оптимальное решение задачи $\mathcal{P}$.

Доказательство этой теоремы мы опускаем, так как она доказывается аналогично теореме 2.I.

Однако может оказаться, что значения u_i, полученные в результате работы первого этапа алгоритма T_2, можно уменьшить таким образом, что новые значения u_i также будут оптимальными.

Действительно, рассмотрим такой случай. Предположим, что в начале работы алгоритма T_2 существует пара (i_0, j_0), для которой $x_{i_0 j_0} = 1$, $u_{i_0} = 0$, $g_{i_0 j_0} = g_{i_1 j_0} + u_{i_1}$, и для всех $j \in N \setminus \{j_0\}$ таких, что $i(j) \in M \setminus \{i_1\}$, выполняются соотношения $g_{i_1 j} + u_{i_1} > g_{i(j)j} + u_{i(j)}$ (i_1 - номер переменной y_{i_1}, значение которой увеличено на 1). В этом случае, в силу выбора $\bar{d}$, значения u_i не меняются, и первый этап после одного шага заканчивает свою работу. Однако очевидно, что после замены основы в столбце j_0, значение u_{i_1} может быть уменьшено на величину

$$\hat{d} = \min\left\{u_{i_1}, \min_{\substack{j / i(j) \neq i_1 \\ j \in N \setminus \{j_0\}}} (g_{i_1 j} + u_{i_1} - v_j)\right\} > 0,$$

при этом каждая основа остается u-минимальным элементом в соответствующем столбце. Для того чтобы исключить все случаи, когда u_i можно уменьшить, сохранив при этом оптимальность, мы используем завершающий этап алгоритма T_2. Остается доказать следующую теорему.

Теорема 3.2. Значения u_i, полученные в результате работы алгоритма T_2, являются оптимальным решением задачи /2.6/-/2.8/.

Действительно, по теореме /3.I/, решение (u_i), полученное в результате работы первых двух этапов алгоритма T_2, оптимально. На третьем этапе осуществляется уменьшение u_i, при этом $u_i > 0$; кроме того, в силу выбора $\hat{d}$ все основы остаются u - минимальными элементами в соответствующих столбцах, следовательно, алгоритм T_2 дает оптимальное решение задачи /2.6/ - /2.8/.

§4. Основная идея алгоритма решения задачи /I.I/ - /I.4/

Алгоритм решения задачи /I.I/ - /I.4/ представляет собой итерационный процесс, в основу которого положено следующее свойство: при фиксированных значениях переменных y_i задача /I.I/ - / I.4/ распадается на последовательность транспортных задач специального вида.

В начале работы алгоритма выбирается некоторое начальное приближение $y^{(0)}, x^{(0)}$. Далее следуют однотипные итерации, каждая из которых условно может быть разбита на два этапа:

I) изменение вектора (y_i);

2) решение задачи /I.I/ - /I.4/ при новых значениях y_i.

Для простоты на каждой итерации изменяется значение только одной из переменных y_i, причем новое значение отличается от старого на 1.

Чтобы определить правило, согласно которому на каждой итерации выбирается одна из переменных y_i, значение которой подвергается изменению, рассмотрим двойственную задачу к /I.I/ - /I.3/ при $x_{ij}^t > 0$:

$$\varPhi(u) = \sum_{j\in N} \sum_{t\in P} \min_{i\in M}(g_{ij}^t + u_{it}) \longrightarrow \max_{(u_{it})} ; \qquad /4.I/$$

$$\sum_{t\in P} u_{it} < c_i \qquad (i \in M); \qquad /4.2/$$

$$u_{it} \geqslant 0 \qquad (i\in M,\ t\in P). \qquad /4.3/$$

Нетрудно показать, что задача /4.I/ - /4.3/ является задачей выпуклого программирования. Это обстоятельство значительно облегчает поиск ее решения,близкого к оптимальному. Однако в силу целочисленности переменных задачи /I.I/ - /I.4/, имея даже оптимальное решение задачи /4.I/ - /4.3/, мы не можем достаточно эффективно организовать поиск приемлемого по точности решения $(y_i,\ x_{ij}^t)$. В этой связи представляется естественным искать решение исходной задачи, ориентируясь на двойственную. Это оказывается возможным, если на каждой итерации в качестве приближенных решений задачи /4.I/ - /4.3/ использовать решения задач /2.6/ - /2.8/ при каждом значении $t \in P$.

Таким образом, решая на каждой итерации задачу /I.I/ - /I.4/ при некоторых фиксированных значениях y_i , мы одновременно получаем приближенное решение задачи /4.I/ - /4.3/, при этом, в силу того, что $\varPhi(u)$ - неубывающая функция, с уменьшением значений y_i (см.описание алгоритма T_1) значение $\varPhi(u)$ не убывает.Поэтому,выбрав в качестве начального приближения значения $u_{it}=0$ и определив соответствующие значения x_{ij}^t и $y_i = \max_{t\in P} \sum_j x_{ij}^t$, попытаемся так организовать выбор переменной y_k , значение которой мы будем уменьшать, чтобы приращение целевой функции задачи /4.I/- /4.3/ было максимальным или близким к максимальному.

Пусть $\Delta u_{it}(k)$ и $\Delta\varPhi(\Delta u(k))$ - соответственно приращения переменных u_{it} и функции $\varPhi(u)$ при уменьшении на 1 значения k-й компоненты вектора (y_i) . С учетом соотношения /2.I2/ мы можем написать

$$\begin{aligned}\Delta\varPhi(\Delta u(k)) &= \varPhi(u+\Delta u(k)) - \varPhi(u) = \\ &= \sum_t \Big(\sum_j \min_{i\in M}(g_{ij}^t + u_{it} + \Delta u_{it}(k)) - \\ &- \sum_j \min_{i\in M}(g_{ij}^t + u_{it})\Big) = \sum_i \sum_t \Delta u_{it}(k)\, y_i .\end{aligned}$$

Таким образом, приходим к задаче

$$\sum_i y_i \sum_t \Delta u_{it}(k) \longrightarrow \max_{k\in M} . \qquad /4.4/$$

В силу соотношений $\Delta u_{it}(K) < \Delta u_{Kt}(K) (i \in M, t \in P)$, справедливость которых вытекает из правил вычисления Δu_{it} (см. первый этап алгоритма T_1), величина критерия /4.4/ существенно зависит от $\Delta u_{Kt}(K)$, однако в силу ограничений /4.2/ сумма $\sum_t \Delta u_{Kt}(K)$ не может быть больше $c_K - \sum_t u_{Kt}$. С учетом сказанного критерий выбора номера K может быть представлен в виде

$$y_i (c_i - \sum_t u_{it}) \longrightarrow \max_{i \in M}.$$

Предположим теперь, что на некоторой итерации после уменьшения значения одной из переменных y_i для некоторого номера i' выполняется неравенство $\sum_t u_{i't} > c_{t'}$. Чтобы вернуть u_{it} в область допустимых решений, увеличим на I значение переменной $y_{i'}$. Если при этом потребовать, чтобы значение $y_{i'}$ далее не уменьшалось, то через некоторое число итераций нельзя будет уменьшить значение ни одной переменной y_i. В этом случае алгоритм заканчивает свою работу.

§5. Формальное описание алгоритма $\mathcal{A}$ решения задачи /I.I/ - /I.4/

В качестве начального приближения выберем следующие значения переменных:

$$x_{ij}^t = \begin{cases} 1, & \text{если } g_{ij}^t = \min_{i \in M} g_{ij}^t \text{ (если минимальных значений несколько,} \\ & \text{то выбираем одно из них);} \\ 0 & \text{– в противном случае,} \end{cases}$$

$$y_i = \max_{t \in P} \sum_{j \in N} x_{ij}^t, \qquad u_{it} = 0.$$

Опишем алгоритм по пунктам. Через $\mathcal{J}\!\mathcal{L}$ обозначено множество строк, для которых к данному моменту работы алгоритма значение y_i увеличилось. В начале работы алгоритма $\mathcal{J}\!\mathcal{L} = \varnothing$.

П.I. Если $\sum_i y_i = n$ либо $\mathcal{J}\!\mathcal{L} = M$, то алгоритм заканчивает свою работу.

П.2. Определим номер K такой, что

$$y_K (c_K - \sum_t u_{Kt}) = \min_{i \in M \setminus \mathcal{J}\!\mathcal{L}} y_i (c_i - \sum_t u_{it}).$$

П.3. Если $y_K = 0$, то алгоритм прекращает свою работу, в противном случае положим $y_K := y_K - 1$.

П.4. Используя алгоритм T_1, решаем задачу /I.I/ - /I.4/ при фиксированных значениях y_i.

П.5. Пусть получены новые значения x_{ij}^t и u_{it}. Если

$$\sum_t u_{Kt} > c_K,$$

то вернемся к старым значениям переменных x_{ij}^t и u_{it}, положим $y_K := y_K + 1$, $\mathcal{J}\!\mathcal{L} := \mathcal{J}\!\mathcal{L} \cup \{K\}$ и перейдем к п.I.

П.6. Если для всех значений i выполняются условия $\sum_t u_{it} < c_i$, то перейдем к п.I.

П.7. Пусть $R=\{i / \sum_t u_{it} > c_i\}$. Определим номер r такой, что

$$\sum_t u_{rt} - c_r = \max_{i \in R} \left(\sum_t u_{it} - c_i \right).$$

П.8. Если $r \in M \setminus \mathcal{J}$, то положим $\mathcal{J} := \mathcal{J} \bigcup \{r\}$.

П.9. Положим $y_r := y_r + 1$ и решим задачу /I.I/ - /I.4/ при фиксированных значениях y_i.

П.IO. Перейдем к п.6.

Оценим трудоемкость алгоритма. При каждом изменении на I значения некоторой переменной y_i на решение последовательности задач вида /2.I/ - /2.4/ затрачивается $O(mp(m+n))$ действий. Общее число изменений значений компонент вектора (y_i) не превосходит величины Cmn, где $C = const$, следовательно, трудоемкость алгоритма $\mathcal{A}$ оценивается величиной $O(m^2 np(m+n))$.

Поступила в ред.-изд.отдел
I октября 1978 г.

Литература

I. Петров Н.Н.,Кузнецов А.С. К вопросу выбора оборудования для главных вентиляторных установок шахт. - В кн.: Управление вентиляцией и газодинамическими явлениями в шахтах. Новосибирск, 1977, с. 71-80.

2. Береснев В.Л.,Гимади Э.Х.,Дементьев В.Т. Экстремальные задачи стандартизации.-Новосибирск.:Наука, 1978.-333 с.

3. Кравцов М.К.,Шерман А.Х. Алгоритм решения транспортной задачи специального вида. - В кн.:Математические модели и методы в автоматизированных системах. Минск, 1975, вып. I, с. 44-5I.

4. Диниц Е.А., Кронрод М.А. Один алгоритм решения задачи о назначениях. - Докл. АН СССР , 1969, т. 189, № I, с. 10-15.