ЭКОНОМИКА И МАТЕМАТИЧЕСКИЕ МЕТОДЫ 1965 том I, вып. 2

О РЕШЕНИИ ЗАДАЧ ДИСКРЕТНОГО ПРОГРАММИРОВАНИЯ СПЕЦИАЛЬНОГО ВИДА

Ю. Ю. ФИНКЕЛЬШТЕЙН

(*МОСКВА*)

Задача дискретного программирования — это такая задача математического программирования, в которой каждая переменная x_j может принимать лишь дискретное множество значений B_j. Наиболее типичным является случай, когда B_j — конечное множество, т. е. $B_j = \{b_{j_1}, \ldots, b_{j_q}\}$. Частным случаем задачи дискретного программирования является задача целочисленного программирования, в которой на переменные накладывается требование целочисленности. Формально любая задача дискретного программирования, в которой все множества B_j конечны, может быть сведена к задаче целочисленного программирования, но такое сведение может привести к увеличению количества переменных и другим трудностям.

Дискретность в задачах математического программирования может появиться по разным причинам, но основные типы дискретных задач следующие:

1. Задачи с неделимыми объектами (нельзя построить 3,5 доменных печи).

2. Комбинаторные задачи (например, задачи теории расписаний; задача коммивояжера).

3. Задачи о выборе варианта (например, выбор варианта размещения предприятий в ряде пунктов на основе проектных вариантов и при наличии ряда ограничений).

Решение задач дискретного программирования связано с большими трудностями. Единственный общий метод решения дискретных (точнее — целочисленных) задач математического программирования — это метод Гомори [1] для решения задач целочисленного линейного программирования. Однако в настоящее время эффективность этого метода пока не ясна, а решение задач на машинах иногда не приводило к результатам даже после весьма большого количества итераций.

Поэтому представляет интерес исследование отдельных классов задач дискретного программирования, для которых могут быть предложены специальные методы решения.

В работе Ю. И. Журавлева и автора [2] был предложен локальный алгоритм, позволяющий в ряде случаев уменьшить перебор при решении задач дискретного* программирования. Более точно: локальный алгоритм позволяет для некоторых задач дискретного программирования вычислить значения отдельных компонент оптимального решения. В настоящей работе выделен класс задач (квазиблочные задачи с не очень большими блоками), для которых дан алгоритм, позволяющий вычислить все компоненты оптимального решения. При этом необходимый перебор оказывается существенно меньше полного перебора, однако несколько возрастают требования

* В работе [2] рассматриваются задачи целочисленного линейного программирования, однако все понятия и результаты этой работы непосредственно переносятся на случай задачи дискретного программирования с сепарабельной целевой функцией.

к объему запоминаемой информации. Определение окрестности, использованное в [2], в данной работе слегка модифицировано.

Постановка задачи. Рассмотрим задачу дискретного программирования с сепарабельной целевой функцией:

$$\sum_{j=1}^{n} f_j(x_j) \to \max, \tag{1}$$

$$(x_{j_{i1}}, \ldots, x_{j_{ip_i}}) \in A_i, \quad i = 1, 2, \ldots, m, \tag{2}$$

$$x_j \in B_j = \{b_{j1}, \ldots, b_{jq_j}\}, \quad j = 1, 2, \ldots, n, \tag{3}$$

где

1) $j_{ir} \in \{1, 2, \ldots, n\}$, $r = 1, 2, \ldots, p_i$, $i = 1, 2, \ldots, m$; 2) $j_{ir} \neq j_{is}$ при $r \neq s$; 3) A_i — некоторое множество p_i-мерных векторов; 4) $b_{jr} = b_{js}$ при $r \neq s$; 5) $p_i \geqslant 2$, $i = 1, 2, \ldots, m$; 6) $q_j \geqslant 2$, $j = 1, 2, \ldots, n$; 7) каждое из переменных x_j входит хотя бы в одно из условий (2), $j = 1, 2, \ldots, n$ *.

Каждое из множеств A_i может быть задано различными способами, например: а) уравнением, б) неравенством, в) логическим условием, г) списком входящих в A_i векторов и т. д. Следует заметить, что при задании списком достаточно перечислить лишь те векторы из A_i, которые удовлетворяют условиям (3) (т. е. вместо множества A_i задать $A_i \cap [B_1, B_2, \ldots, B_n]$, где $[B_1, B_2, \ldots, B_n]$ — прямое произведение множеств $B_1, B_2, \ldots, B_n$).

Иногда вместо множества A_i удобнее задавать множество $\bar{A}_i$ (т. е. множество всех p_i-мерных векторов, не входящих в A_i) или соответственно $\bar{A}_i \cap [B_1, B_2, \ldots, B_n]$.

Пример 1.

$$x_1 + 2x_2 + x_3^2 - \frac{12}{x_4} \to \max, \tag{1'}$$

$$\left.\begin{array}{c} x_1^2 + 2x_2^2 = 18, \\ x_1 + x_3 \leqslant 8. \\ \text{Если} \quad x_2 = 3, \quad \text{то} \quad \dfrac{x_4 - 0{,}5}{5} \text{ — целое число,} \\ (x_3, x_4) \in \{(1; 7); (2; 5); (3; 4{,}5); (3; 5{,}5); (8; 4); (8; 2)\}, \end{array}\right\} \tag{2'}$$

$$\left.\begin{array}{l} x_1 \in \{0; 1; 2; 3; 4; 8\}, \\ x_2 \in \{3; 5; 7; 11; 13; 17\}, \\ x_3 \in \{1; 2; 3; 5; 7{,}5; 8\}, \\ x_4 \in \{2; 4{,}5; 5; 5{,}5; 7; 11\}. \end{array}\right\} \tag{3'}$$

Множества условий $U_k(x_j; Z)$ и множества переменных $S_k(x_j; Z)$ [разбиение системы условий (2) на «блоки»]. Рассмотрим некоторую систему условий Z [вида (2)]. Будем i-е условие обозначать через $\langle i \rangle$. Множество этих условий, содержащих хотя бы одно из переменных

---

* Очевидно, что вводя условия 5, 6, 7, мы не ограничиваем класс рассматриваемых задач. Действительно, пусть $p_i = 1$, т. е. соответствующее условие (2) имеет вид $\{x_{j_{i1}}\} \in A_i$. Тогда это условие можно исключить из (2), соответственно изменив (3). Далее пусть $q_r = 1$. Тогда x_r может принимать только одно значение, так что x_r можно зафиксировать, а число переменных уменьшить на единицу. И, наконец, если некоторое переменное x_t не входит ни в одно из условий (2), то можно сразу положить $x_t = \bar{x}_t$ $\left(f_t(\bar{x}_t) = \max\limits_{x_t \in B_t} f_t(x_t)\right)$ и уменьшить число переменных на единицу.

$x_{j_1}, \ldots, x_{j_s}$, будем обозначать через $U(x_{j_1}, \ldots, x_{j_s})$. Множество всех переменных, содержащихся в условиях, $\langle t_1 \rangle, \langle t_2 \rangle, \ldots, \langle t_r \rangle$ будем обозначать через $S\langle t_1, t_2, \ldots, t_r \rangle$.

Выберем некоторое переменное x_j и определим множество условий $U_0'(x_j; Z)$ и переменных $S_0(x_j; Z)$ следующим образом:

$$U_0'(x_j; Z) = U(x_j), \quad S_0(x_j; Z) = S\langle U_0'(x_j; Z) \rangle. \tag{4'}$$

Множества $U_k'(x_j; Z)$ и $S_k(x_j; Z)$ при $k \geqslant 1$ определяются индуктивно. Пусть множества $U'_{k-1}(x_j; Z)$ и $S_{k-1}(x_j; Z)$ уже определены. Тогда по определению

$$U_1'(x_j; Z) = U(S_0(x_j; Z)) \setminus U_0'(x_j; Z); \quad S_1(x_j; Z) = S\langle U_1'(x_j; Z) \rangle,$$

и при $k \geqslant 2$

$$\begin{gathered} U_k'(x_j; Z) = U[S_{k-1}(x_j; Z) \setminus S_{k-2}(x_j; Z)] \setminus U'_{k-1}(x_j; Z), \\ S_k(x_j; Z) = S\langle U_k'(x_j; Z) \rangle. \end{gathered} \tag{4''}$$

Очевидно, что существует такое r, что при $k > r$ имеем $U_k'(x_j; Z) = \Lambda$, $S_k(x_j; Z) = \Lambda$ * (где Λ — пустое множество).

Пусть l — наименьшее из целых неотрицательных чисел k, обладающих тем свойством, что $S_{k+1}(x_j; Z) \subseteq S_k(x_j; Z)$. Дадим следующее определение конечной последовательности множеств $U_k(x_j; Z)$ (где $k = 0, 1, \ldots, l$):

$$U_k(x_j; Z) = \begin{cases} U_k'(x_j; Z) & \text{при } 0 \leqslant k < l, \\ U_l'(x_j; Z) \cup U'_{l+1}(x_j; Z) & \text{при } k = l. \end{cases} \tag{5}$$

Будем считать в дальнейшем, что $\bigcup_{k=0}^{l} U_k(x_j; Z) = \langle 1, 2, \ldots, m \rangle$ и соответственно $\bigcup_{k=0}^{l} S_k(x_j; Z) = \{x_1, x_2, \ldots, x_n\}$.

Действительно, если предположить, что это не так, то можно без труда найти такие переменные

$x_{j1} = x_j,\ x_{j2}, \ldots, x_{j_r} (j_s \in \{1, 2, \ldots, n\},\ s = 1, 2, \ldots, r;\ j_s \neq j_t$ при $s \neq t)$, что

$$\text{а)} \quad \left[\bigcup_{k=0}^{l_t} U_k(x_{j_t}; Z) \right] \cap \left[\bigcup_{k=0}^{l_s} U_k(x_{j_s}; Z) \right] = \Lambda$$

при $t \neq s$ и соответственно

$$\left[\bigcup_{k=0}^{l_t} S_k(x_{j_t}; Z) \right] \cap \left[\bigcup_{k=0}^{l_s} S_k(x_{j_s}; Z) \right] = \Lambda \quad \text{при} \quad t \neq s,$$

$$\text{б)} \quad \bigcup_{t=1}^{r} \left[\bigcup_{k=0}^{l_t} U_k(x_{j_t}; Z) \right] = \langle 1, 2, \ldots, m \rangle \quad \text{при} \quad t \neq s$$

и соответственно

$$\bigcup_{t=1}^{r} \left[\bigcup_{k=0}^{l_t} S_k(x_{j_t}; Z) \right] = \{x_1, x_2, \ldots, x_n\}.$$

Но в силу сепарабельности целевой функции задача (1) — (3) в этом случае «распадается» на r независимых задач того же типа, каждую из которых можно решать отдельно.

---

* Мы считаем, что если Λ — пустое множество, то $S(\Lambda) = \Lambda$ и $U(\Lambda) = \Lambda$; Λ, как обычно, считается подмножеством любого другого множества.

Следующие свойства множеств $U_k(x_j; Z)$ и $S_k(x_j; Z)$ $(k = 0, 1, \ldots, l)$ вытекают непосредственно из их определений:

1) множество всех переменных, входящих хотя бы в одно из условий системы $U_k(x_j; Z)$, есть $S_k(x_j; Z)$;

2) $U_k(x_j; Z) \cap U_t(x_j; Z) = \Lambda$ при $k \neq t$ $(k = 0, 1, \ldots, l;\ t = 0, 1, \ldots, l)$, т. е. системы $U_k(x_j; Z)$ и $U_t(x_j; Z)$ при $k \neq t$ не имеют общих условий;

3) $S_k(x_j; Z) \cap S_{k+1}(x_j; Z) \neq \Lambda$ $(k = 0, 1, \ldots, l)$, т. е. системы $U_k(x_j; Z)$ и $U_{k+1}(x_j; Z)$ обязательно имеют общие переменные;

4) $S_k(x_j; Z) \cap S_r(x_j; Z) = \Lambda$ при $|k - r| \geqslant 2$ $(k = 0, 1, \ldots, l;\ r = 0, 1, \ldots, l)$, т. е. при $|k - r| \geqslant 2$ системы $U_k(x_j; Z)$ и $U_r(x_j; Z)$ не имеют общих переменных.

Пример 2. В качестве Z возьмем следующую систему неравенств:

$$f_1(x_2, x_3, x_5) \leqslant b_1, \quad \langle 1\rangle$$

$$f_2(x_2, x_7, x_8) \leqslant b_2, \quad \langle 2\rangle$$

$$f_3(x_3, x_4, x_5, x_6) \leqslant b_3, \quad \langle 3\rangle$$

$$f_4(x_1, x_7, x_8) \leqslant b_4, \quad \langle 4\rangle$$

$$f_5(x_4, x_6) \leqslant b_5. \quad \langle 5\rangle$$

Выпишем множества $U_k'(x_1; Z)$, $U_k(x_1; Z)$, $S_k(x_1; Z)$.

k	$U_k'(x_1; Z)$	$S_k(x_1; Z)$	$U_k(x_1; Z)$	k	$U_k'(x_1; Z)$	$S_k(x_1; Z)$	$U_k(x_1; Z)$
0	⟨4⟩	$\{x_1, x_7, x_8\}$	⟨4⟩	3	⟨3⟩	$\{x_3, x_4, x_5, x_6\}$	⟨3,5⟩
1	⟨2⟩	$\{x_2, x_7, x_8\}$	⟨2⟩	4	⟨5⟩	$\{x_4, x_6\}$	—
2	⟨1⟩	$\{x_2, x_3, x_5\}$	⟨1⟩				

По определению получаем $l = 3$.

Алгоритм. При изложении алгоритма будем пользоваться следующими обозначениями:

$U_k = U_k(x_j; Z)$; $S_k = S_k(x_j; Z)$; $\{x_j\}_k$ — набор, состоящий из всех переменных, входящих в $S_k \cap S_{k+1}$ $(k = 0, 1, \ldots, l-1)$; $\{x_j\}^k$ — набор, состоящий из всех переменных, входящих в $S_k \setminus S_{k+1}$ $(k = 0, 1, \ldots, l-1)$; $\{x_j\}^l$ — набор, состоящий из всех переменных, входящих в S_l; $\{x_j\}_\Sigma{}^k$ — набор, состоящий из всех переменных, входящих в $[\bigcup\limits_{t=1}^{k} S_t] \setminus S_{k+1}$*, $0 \leqslant k \leqslant l-1$.

Переходим непосредственно к изложению алгоритма.

1) Произвольно выбираем некоторое переменное x_j и строим множества $U_k = U_k(x_j; Z)$ и $S_k = S_k(x_j; Z)$ $(k = 0, 1, \ldots, l)$.

2) Если $l = 0$, то решаем задачу с помощью полного перебора всех наборов $\{x_1, \ldots, x_n\}$, удовлетворяющих условиям (3) (тривиальный случай, задача не имеет блочной структуры).

3) Пусть $l > 0$.

Тогда для каждого набора переменных $\{x_p\}_0 = \{\alpha_p\}_0$, удовлетворяющего условиям (3), решаем следующую задачу максимизации:

$$\sum_{x_p \in [S_0 \setminus S_1]} f_p(x_p) \to \max, \tag{6}$$

$$\{x_p\}_0 = \{\alpha_p\}_0, \tag{7}$$

* Соответствующие фиксированные наборы будем обозначать, например, через $\{\alpha_j\}_k$, $\{\alpha_j\}^k$, $\{\alpha_j\}^l$, $\{\alpha_j\}_\Sigma{}^k$.

$$\{x_{j_{i1}}, \ldots, x_{j_{ip_i}}\} \in A_i \qquad (\text{для всех } \langle i \rangle \in U_0), \tag{8}$$

$$x_p \in B_p \quad (\text{для всех } x_p \in S_0). \tag{9}$$

Для некоторых наборов $\{\alpha_p\}_0$ задача (6) — (9) может оказаться неразрешимой. Такие наборы назовем неотмеченными. Для других наборов $\{\alpha_p\}_0$ задача (6) — (9) имеет решение $\{x_t\}_0 = \{\beta_t[\{\alpha_p\}_0]\}^0$. Такие наборы назовем отмеченными. Множество отмеченных наборов $\{\alpha_p\}_0$ обозначим через R_0. (Очевидно, если множество R_0 пусто, то исходная задача (1) — (3) не имеет решения.)

Введем следующее обозначение:

$$\varphi_0[\{\alpha_p\}_0] = \sum_{x_t \in [S_0 \setminus S_1]} f_t(\beta_t[\{\alpha_p\}_0]) + \sum_{x_p \in [S_0 \cap S_1]} f_p(\alpha_p). \tag{10}$$

Для каждого отмеченного набора $\{\alpha_p\}_0$ запомним следующий набор величин:

$$\{\{\beta_t[\{\alpha_p\}_0]\}_{\Sigma^0};\ \{\alpha_p\}_0;\ \varphi_0[\{\alpha_p\}_0]\}, \text{ где } \{\beta_t[\{\alpha_p\}_0]\}_{\Sigma^0} = \{\beta_t[\{\alpha_p\}_0]\}^0.$$

4) Пусть $0 < k < l$. Предположим, что все наборы $\{x_p\}_{k-1} = \{\alpha_p\}_{k-1}$ разбиты на множество неотмеченных и множество отмеченных R_{k-1}, причем для каждого отмеченного набора $\{\alpha_p\}_{k-1}$ уже определено и хранится в памяти следующий набор величин:

$$\{\{\beta_t[\{\alpha_p\}_{k-1}]\}_{\Sigma^{k-1}};\ \{\alpha_p\}_{k-1};\ \varphi_{k-1}[\{\alpha_p\}_{k-1}]\}.$$

Тогда для каждого набора $\{x_p\}_k = \{\alpha_p\}_k$, удовлетворяющего условиям (3), решаем следующую задачу максимизации:

$$\sum_{x_p \in [S_k \setminus (S_{k-1} \cup S_{k+1})]} f_p(x_p) + \varphi_{k-1}[\{x_p\}_{k-1}] \to \max, \tag{11}$$

$$\{x_p\}_k = \{\alpha_p\}_k, \tag{12}$$

$$\{x_p\}_{k-1} \in R_{k-1}, \tag{13}$$

$$\{x_{j_{i_1}}, \ldots, x_{j_{ip_i}}\} \in A_i \ (\text{для всех } \langle i \rangle \in U_k), \tag{14}$$

$$x_p \in B_p \quad (\text{для всех } x_p \in S_k). \tag{15}$$

Для некоторых наборов $\{\alpha_p\}_k$ задача (11) — (15) может оказаться неразрешимой. Такие наборы назовем неотмеченными. Для других наборов $\{\alpha_p\}_k$ задача (11) — (15) имеет решение $\{x_t\}^k = \{\beta_t'[\alpha_p\}_k]\}^k$. Такие наборы назовем отмеченными. Множество отмеченных наборов $\{\alpha_p\}_k$ обозначим через R_k. (Очевидно, если множество R_k пусто, то исходная задача (1) — (3) не имеет решения.)

Введем следующие обозначения:

$$\{\beta_t[\{\alpha_p\}_k]\}_{\Sigma^k} = \{\{\beta_t[\{\beta_r'[\{\alpha_p\}_k]\}_{k-1}]\}_{\Sigma^{k-1}},\quad \{\beta_t'[\{\alpha_p\}_k]\}^k\}, \tag{16}$$

$$\varphi_k[\{\alpha_p\}_k] = \varphi_{k-1}[\{\beta_t[\{\alpha_p\}_k]\}_{k-1}] + \sum_{x_t \in [S_t \setminus (S_{k-1} \cup S_{k+1})]} f_t(\beta_t[\{\alpha_p\}_k]) + $$
$$+ \sum_{x_p \in [S_k \cap S_{k+1}]} f_p(\alpha_p). \tag{17}$$

Поясним содержательный смысл набора величин $\{\beta_t[\{\alpha_p\}_k]\}_{\Sigma^k}$. Рассмотрим следующую задачу:

$$\sum_{x_l \in \left[\bigcup\limits_{q=1}^{k} S_q\right] \setminus S_{k+1}} f_l(x_l) \to \max,$$

$$x_p = \alpha_p (x_p \in S_k \cap S_{k+1}),$$

причем для всех переменных $x_t \in \bigcup\limits_{q=1}^{k} S_q$ соблюдаются условия (2), входящие в $\bigcup\limits_{q=0}^{k} U_q$, и условия (3). Одним из решений этой задачи и является набор

$$\{\{\beta_t[\{\alpha_p\}_k]\}_{\Sigma^k},\ \{\alpha_p\}_k\}.$$

Запоминаем для всех отмеченных наборов следующие наборы величин:

$$\{\{\beta_t[\{\alpha_p\}_k]\}_{\Sigma^k};\ \{\alpha_p\}_k;\ \varphi_k[\{\alpha_p\}_k]\},$$

и вычеркиваем из памяти наборы величин

$$\{\{\beta_t[\{\alpha_p\}_{k-1}]\}_{\Sigma^{k-1}};\ \{\alpha_p\}_{k-1};\ \varphi_{k-1}[\{\alpha_p\}_{k-1}]\}.$$

5) Пусть $k = l$. Предположим, что все наборы $\{x_p\}_{l-1} = \{\alpha_p\}_{l-1}$ разбиты на множество неотмеченных и множество отмеченных R_{l-1}, причем для каждого отмеченного набора $\{\alpha_p\}_{l-1}$ помним следующие наборы величин:

$$\{\{\beta_t[\{\alpha_p\}_{l-1}]\}_{\Sigma^{l-1}};\qquad \{\alpha_p\}_{l-1};\qquad \varphi_{l-1}[\{\alpha_p\}_{l-1}]\}.$$

Тогда решаем следующую задачу максимизации:

$$\sum_{x_p \in [S_l \setminus S_{l-1}]} f_p(x_p) + \varphi_{l-1}[\{x_p\}_{l-1}] \to \max, \tag{18}$$

$$\{x_p\}_{l-1} \in R_{l-1}, \tag{19}$$

$$\{x_{j_{i_1}}, \ldots, x_{j_{ip_i}}\} \in A_i\ (\text{для всех } \langle i\rangle \in U_l), \tag{20}$$

$$x_p \in B_p\ \ (\text{для всех } x_p \in S_l). \tag{21}$$

Пусть $\{x_p\}^l = \{\gamma_p'\}^l$ — решение задачи (18) — (21). Тогда решение первоначальной задачи [(1) — (3)] $\{x_p\} = \{\gamma_p\}$ выражается следующим образом:

$$\{\gamma_p\} = \{\{\beta_p\{\gamma_p'\}_{l-1}]\}_{\Sigma^{l-1}},\ \{\gamma_p'\}^l\}. \tag{22}$$

П р и м е р 3. Конкретизируем систему условий, рассмотренную в примере 2 (это позволит нам использовать множества $U_k(x_1;\ Z)$ и $S_k(x_1;\ Z)$, определенные в примере 2). Введем целевую функцию и определим множества B_j; получим задачу максимизации, выписанную ниже.

При этом переставим условия и переменные (не меняя их нумерации). Это позволит выявить «квазиблочную» структуру задачи и пояснить определение множеств $U_k(x_j;\ Z)$ и $S_k(x_j;\ Z)$:

$$10x_1 + x_7 + 8x_8 + 2x_2 + 9x_3 + 3x_5 + 11x_4 + 7x_6 \to \max,$$

$$x_1 + x_7 + x_8 \leqslant 1, \qquad \langle 4\rangle$$

$$x_7 + 2x_8 + x_2 \leqslant 2, \qquad \langle 2\rangle$$

$$2x_2 + 3x_3 + 4x_5 \leqslant 8, \qquad \langle 1\rangle$$

$$2x_3 + x_5 + 2x_4 + x_6 \leqslant 3, \qquad \langle 3\rangle$$

$$x_4 + x_6 \leqslant 1, \qquad \langle 5\rangle$$

$$x_j \in \{0, 1,\},\ j = 1, 2, \ldots, 8.$$

1) $U_0(x_1;\ Z) = \langle 4 \rangle$; $U_1(x_1;\ Z) = \langle 2 \rangle$; $U_2(x_1;\ Z) = \langle 1 \rangle$;

$$U_3(x_1;\ Z) = \langle 3, 5 \rangle;\quad S_0(x_1; Z) = \{x_1, x_7, x_8\};$$

$$S_1(x_1, Z) = \{x_7, x_8, x_2\};\quad S_2\{x_1;\ Z\} = \{x_2, x_3, x_5\};$$

$$S_3(x_1; Z) = \{x_3, x_5, x_4, x_6\}.$$

2) $l = 3 \neq 0$.

3) $10x_1 \to \max$,

$(x_7,\ x_8) = \{\alpha_7,\ \alpha_8\}$,

$x_1 + x_7 + x_8 \leqslant 1$,

$x_j \in \{0,\ 1\},\ j = 1,\ 7,\ 8.$

α_7	0	0	1
α_8	0	1	0
$\beta_1(\alpha_7,\ \alpha_8)$	1	0	0
$\varphi_0(\alpha_7,\ \alpha_8)$	10	8	1

4) $\varphi_0(x_7,\ x_8) \to \max$,

$x_2 = \alpha_2$

$x_7 + 2x_8 + x_2 \leqslant 2$,

$(x_7,\ x_8)$ — отмеченные,

$x_j \in \{0,\ 1\},\ j = 7,\ 8,\ 2.$

α_2	0	1
$\beta_8(\alpha_2)$	0	0
$\beta_7(\alpha_2)$	0	0
$\beta_1(\alpha_2)$	1	1
$\varphi_1(\alpha_2)$	10	12

5) $\varphi_1(x_2) \to \max$,

$2x_2 + 3x_3 + 4x_5 \leqslant 8$,

$(x_3,\ x_5) = (\alpha_3,\ \alpha_5)$,

x_2 — отмеченные,

$x_j \in \{0,\ 1\},\ j = 2,\ 3,\ 5$

α_3	0	0	1	1
α_5	0	1	0	1
$\beta_2(\alpha_3,\ \alpha_5)$	1	1	1	0
$\beta_8(\alpha_3,\ \alpha_5)$	0	0	0	0
$\beta_7(\alpha_3,\ \alpha_5)$	0	0	0	0
$\beta_1(\alpha_3,\ \alpha_5)$	1	1	1	1
$\varphi_2(\alpha_3,\ \alpha_5)$	12	15	21	22

6) $\varphi_2(x_3, x_5) + 11x_4 + 7x_6 \to \max$,

$$2x_3 + x_5 + 2x_4 + x_6 \leqslant 3,$$

$$x_4 + x_6 \leqslant 1,$$

$$(x_3,\ x_5) \quad \text{— отмеченные},$$

$$x_j \in \{0,\ 1\},\ j = 3,\ 5,\ 4,\ 6.$$

Решение: $\{\gamma_1, \gamma_7, \gamma_8, \gamma_2, \gamma_3, \gamma_5, \gamma_4, \gamma_6\} = \{1, 0, 0, 1, 1, 0, 0, 1\}$. $\sum\limits_j c_j\gamma_j = 10\cdot 1 + 2\cdot 1 + 9\cdot 1 + 7\cdot 1 = 28$.

Объем перебора и памяти. Оценим эффективность алгоритма. Объем перебора $P(x_j;\ Z)$ будем оценивать общим количеством перебираемых наборов, а объем памяти $Q(x_j;\ Z)$ — максимальным количеством одновременно запоминаемых наборов. Напомним, что q_t — это количество различных значений, которые может принимать переменная $x_t (t = 1, 2, \ldots, n)$.

Пусть по-прежнему $S_h = S_h(x_j;\ Z)$.

Тогда получаем

$$P(x_j; Z) \leqslant \sum_{h=0}^{l} \Big(\prod_{x_t \in S_h} q_t \Big), \tag{23}$$

$$Q(x_j;\,Z)\leqslant\begin{cases}\prod\limits_{x_t\in S_0} q_t & \text{при} \quad l=1,\\ \max\limits_{1\leqslant k\leqslant l-1}\left(\prod\limits_{x_t\in[S_k\cap S_{k-1}]} q_t+\prod\limits_{x_t\in[S_k\cap S_{k+1}]} q_t\right) & \text{при} \quad l>1.\end{cases}\tag{24}$$

Для сравнения отметим, что при решении методом полного перебора требования к памяти невелики, а количество перебираемых наборов P равно:

$$P=\prod_{t=1}^{n} q_t.\tag{25}$$

Обозначим через $\pi(S_k)$ количество переменных, входящих в множество $S_k=S_k(x_j;\,Z)$. Допустим, что

$$\max_{0\leqslant k\leqslant l}\pi(S_k)=C\tag{26}$$

и что

$$q_t=q,\ t=1,\,2,\,\ldots,\,n.\tag{27}$$

Из (26) вытекает:

$$l+1\leqslant n-C+1.\tag{28}$$

Отсюда получаем, что если выполнены условия (26) и (27), то для $P(x_j;\,Z)$ и P имеют место следующие оценки:

$$P(x_j;Z)\leqslant(n-C+1)\,q^C,\tag{23$'$}$$

$$P=q^n.\tag{25$'$}$$

Для $Q(x_j;\,Z)$ (при условии выполнения условий (26) и (27)) имеет место следующая оценка:

$$Q(x_j;\,Z)\leqslant q^{C-1}+q.\tag{24$'$}$$

Допустим теперь, что

$$\max_{0\leqslant k\leqslant l-1}\pi(S_k\cap S_{k+1})=B.\tag{26$'$}$$

Тогда для $Q(x_j;\,Z)$ имеет место следующая оценка:

$$Q(x_j;\,Z)\leqslant\min\{2q^B;\ (q^B+q^{C-B})\}.\tag{24$''$}$$

Следует заметить, что формула (23) может быть использована для выбора в некотором смысле наилучшего $x_j=x_{j_1}$.

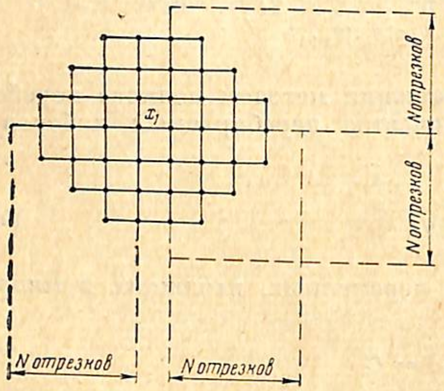
О возможных приложениях алгоритма. Мы ограничимся указанием двух классов задач, для которых изложенный алгоритм может дать значительное снижение перебора.

а) Задача о наилучшем выборе специализации предприятий для выпуска ряда продуктов.

Для каждого из n строящихся предприятий возможны несколько вариантов специализации. Каждый из вариантов (для данного предприятия) характеризуется определенным уровнем расходов и выпуском определенных количеств разных продуктов. Заданы потребности во всех продуктах. Требуется так выбрать варианты строительства для всех предприятий, чтобы потребности во всех продуктах были полностью удовлетворены, а сумма расходов была минимальной.

Изложенный выше алгоритм может дать существенное уменьшение перебора при соблюдении следующих условий: 1) продуктов и предприятий сравнительно много; 2) все варианты строительства предприятий предусматривают достаточно высокий уровень специализации (т. е. выпуск относительно небольшого количества продуктов); 3) каждый из продуктов производится сравнительно небольшим количеством предприятий.

б) Задачи сетевого типа. Имеются в виду задачи, в которых узлы сети интерпретируются как уравнения (или группы уравнений), а дуги — как переменные (или группы переменных), причем каждая переменная может принимать лишь конечное число значений. Для ряда задач такого типа вышеизложенный алгоритм может дать существенное уменьшение перебора.



Приведем простой пример. Рассмотрим сеть, изображенную на рисунке.

Если интерпретировать узлы данной сети как уравнения, а дуги — как входящие в эти уравнения переменные, то нетрудно видеть (считая $N \geqslant 2$), что общее количество переменных равно

$$n = 4N(N+1) - 1 \tag{29}$$

$$\pi(S_k(x_1; Z)) = \begin{cases} 7 & \text{при} \quad k = 0, \\ 16k + 6 & \text{при} \quad 1 \leqslant k \leqslant N - 1 = l. \end{cases} \tag{30}$$

Допустим теперь, что каждая из переменных может принимать ровно q значений. Тогда объем полного перебора P равен:

$$P = q^n = q^{4N(N+1)-1}. \tag{31}$$

Если же воспользоваться изложенным выше алгоритмом, то получаем следующий объем перебора $P(x_1; Z)$:

$$P(x_1; Z) = \sum_{k=0}^{N-1} q^{\pi(S_k(x_1;\, Z))} = q^7 + \sum_{k=1}^{N-1} q^{16k+6},$$

$$P(x_1; Z) = q^7 - q^6 + q^6 \frac{q^{16N} - 1}{q^{16} - 1} \sim \frac{q^{16N+6}}{q^{16} - 1}. \tag{32}$$

ЛИТЕРАТУРА

1. R. E. Gomory. Outline of an algorithm for integer solution to linear programs. Bull. Amer. Math. Soc., 1958, vol. 64, No. 5.
2. Ю. И. Журавлев, Ю. Ю. Финкельштейн. Локальные алгоритмы для задач линейного целочисленного программирования. В сб. Проблемы кибернетики (в печати).

Поступила в редакцию
14 XII 1964