ЭКОНОМИКА
И МАТЕМАТИЧЕСКИЕ МЕТОДЫ

1969
том V, вып. 2

ЗАДАЧИ СТАНДАРТИЗАЦИИ И НЕКОТОРЫЕ МЕТОДЫ ИХ РЕШЕНИЯ

М. М. БЕРКОВИЧ

(Москва)

1. В этой работе рассматриваются экстремальные задачи, которые на практике называются задачами стандартизации. Несмотря на то, что они часто возникают во многих отраслях хозяйства и решение их имеет большое значение, в математической литературе этот класс задач совершенно не исследован. Их можно формально рассматривать как целочисленные задачи линейного программирования. Однако специфика этих задач позволяет разработать более эффективные методы решения по сравнению с общими методами.

Прежде чем приступить к формальному изложению, приведем несколько конкретных постановок задач стандартизации.

Задача 1. Известно, что промышленность может выпускать n различных сортов стали. Известны потребности b_k, $k = 1, \ldots, n$, по каждому из этих сортов. Сорт k лучше по качеству, чем сорт $k+1$, и может использоваться вместо него и всех последующих сортов. Заданы стоимости производства единицы каждого сорта стали c_k. По техническим причинам можно одновременно выпускать не более m различных сортов. Это может быть связано, например, с требованием стандартизации, либо с тем, что имеется всего m печей для выплавки стали. Требуется определить, какие сорта и в каких количествах необходимо выпускать, чтобы удовлетворить заданные потребности при наименьшей сумме затрат на производство.

Задача 2. При проектировании конструкций часто оказывается, что для реализации необходимо иметь большое количество различных размеров однотипных деталей. Это весьма нежелательно с экономической точки зрения. Можно увеличить стоимость конструкции, сократив количество размеров данных деталей, заменяя детали более дорогими, но более качественными. Однако увеличивать стоимость конструкции можно лишь до некоторой заданной величины. При этом условии требуется минимизировать количество размеров рассматриваемых деталей.

Задача 3. Имеется n станков $j = 1, \ldots, n$ и r видов работ $i = 1, \ldots, r$. Известно, что i-я работа может выполняться на любом станке, номер которого не превосходит $k = k(i)$. Заданы объемы $b_i > 0$ каждого вида работы и стоимости c_{ij} выполнения единицы i-й работы на j-м станке. По техническим причинам может работать не более m станков. Требуется определить, какие станки должны работать, какие виды работ должен выполнять каждый станок, чтобы реализовать весь объем работы при наименьших суммарных затратах.

2. Возьмем за основу задачу 1, хотя аппарат, использованный ниже, будет применим к другим задачам.

Пусть x_{kp} $(k \leqslant p)$ — неизвестное количество стали сорта k, заменяющее количество стали сорта p; $c_{kp}x_{kp}$ — стоимость стали; α_{kp} — коэффициент заменяемости, означающий, что единица сорта k заменяет α_{kp} единиц сорта p. Задача состоит в минимизации функции

$$F = CX = \sum_{k,\,p=1}^{n} c_{kp} x_{kp}$$

$$\left.\begin{array}{ll}
\text{при условии } \displaystyle\sum_{k=1}^{p} \alpha_{kp} x_{kp} \geqslant b_p, & \\
\displaystyle\sum_{p=k}^{n} x_{kp} = x_k, & \text{(а1)} \\
x_{kp} \geqslant 0; \quad k = 1, \ldots, n; \quad p = 1, \ldots, n, & \\
\displaystyle\sum_{k=1}^{n} \operatorname{sgn} x_k \leqslant m. & \text{(б1)}
\end{array}\right\} \mathrm{I}$$

В дальнейшем будем предполагать, что $\alpha_{kp} \geqslant 0$ и $\alpha_{kk} = 1$, $c_{kp} \geqslant 0$; $m < n$.

Если в задаче 1 $\alpha_{kp} \equiv 1$; $c_{kp} = c_k$, то эту же задачу можно записать иначе: минимизировать функцию

$$\left.\begin{array}{ll}
F = \displaystyle\sum_{k=1}^{n} c_k x_k & \\
\text{при условии } \displaystyle\sum_{k=1}^{s} x_k \geqslant \sum_{k=1}^{s} b_k, & \text{(а2)} \\
s = 1, \ldots, n; \quad x_k \geqslant 0; \quad k = 1, \ldots, n, & \\
\displaystyle\sum_{k=1}^{n} \operatorname{sgn} x_k \leqslant m. & \text{(б2)}
\end{array}\right\} \mathrm{II}$$

Эта задача была подробно исследована в [1]. Задачи I и II отличаются от задач линейного программирования условием б. Из-за этого условия множество допустимых планов становится не выпуклым, и поэтому эти задачи являются многоэкстремальными. Условие типа (б) в математическом смысле является очень сложным, так как в нем фигурируют разрывные функции, что значительно затрудняет анализ задачи. Хотя на практике условия такого типа встречаются довольно часто, они совершенно не изучены. Рассмотрим метод решения задачи I, основанный на идеях динамического программирования. Установим сначала основные свойства оптимальных решений задачи I.

Свойство 1. В любом плане задачи I $x_1 > 0$, так как $b_1 > 0$ по условию.

Свойство 2. Если $\alpha_{kp} = 0$, то в оптимальном плане $x_{kp}^0 = 0$. Отсюда вытекает, что если перейти от x_{kp} к x_{kp}' и положить $c_{kp}' = c_{kp} / \alpha_{kp}$ при $\alpha_{kp} \neq 0$ и $c_{kp}' = \infty$ при $\alpha_{kp} = 0$, то полученная задача I′ будет эквивалентна исходной.

Теорема 1. *Предположим, что*

$$\alpha_{kp} c_{k+1\,p} < \alpha_{k+1\,p} c_{kp} \qquad (1)$$

для всех k, p, X^0 — оптимальный план задачи I *и*

$$\overset{0}{x}_i \neq 0, \quad \overset{0}{x}_{i+1} = \ldots \overset{0}{x}_{i+t-1} = 0, \quad \overset{0}{x}_{i+t} \neq 0. \tag{2}$$

Тогда

$$\overset{0}{x}_{i,\, i+q} = 0; \quad q \geqslant t. \tag{3}$$

Если (1) выполняется как нестрогое неравенство, то существует оптимальный план, для которого выполнено условие (3).

Доказательство. Предположим, что $\overset{0}{x}_{i,\, i+q} \neq 0$; $q > t$. Составим новый план X'

$$X' = \begin{cases} x'_{i,\, i+q} = 0, \\ x'_{i+t,\, i+q} = \overset{0}{x}_{i+t,\, i+q} + \overset{0}{x}_{i,\, i+q} \dfrac{a_{i,\, i+q}}{a_{i+t,\, i+q}}, \\ x'_{kp} = \overset{0}{x}_{kp} \quad \text{во всех остальных случаях.} \end{cases} \tag{4}$$

Если $\overset{0}{x}_{i,i+q} \neq 0$, то $a_{i,\, i+q} \neq 0$ согласно свойству 2. Тогда из (1) следует, что $a_{i+t,\, i+q} \neq 0$. Следовательно, (4) имеет смысл. Легко проверить, что X' — план задачи I. Вычислим разность

$$CX^0 - CX' = \left[c_{i,\, i+q} - c_{i+t,\, i+q} \frac{a_{i,\, i+q}}{a_{i+t,\, i+q}} \right]. \tag{5}$$

Из (1) следует, что выражение (5) строго больше нуля; это противоречит оптимальности X^0. Если же в (1) стоит нестрогое неравенство и $c_{i,\, i+q} a_{i+t,\, i+q} = c_{i+t,\, i+q} a_{i,\, i+q}$, то построенный вектор X' будет удовлетворять условию (3) и $CX^0 = CX'$. Таким образом, X' — оптимальный план задачи I и теорема доказана полностью.

Из этой теоремы вытекает ряд важных следствий.

Следствие 1. *Если X^0 — оптимальный план, x_i^0 и $\overset{0}{x}_{i+t}$ удовлетворяет* (2), *то*

$$\overset{0}{x}_{is} = \frac{b_s}{a_{is}}; \quad s = i, \ldots, i+t-1, \tag{6}$$

$$x_i^0 = \sum_{s=i}^{i+t-1} \frac{b_s}{a_{is}}; \quad x_i^0 = \sum_{s=1}^{n} \frac{b_s}{a_{is}}, \quad \text{если} \quad x_{i+1} = \ldots = x_n = 0.$$

Другими словами, в каждом уравнении (a1) в оптимальном плане только одна компонента отлична от нуля.

Следствие 2. *Если X^0 — оптимальный план, x_i^0, $\overset{0}{x}_{i+t}$ удовлетворяют условию* (2) *и $a_{is} = 0$ для некоторого $s > i$, то $i + t \leqslant s$.*

Доказательство этих следствий непосредственно вытекает из теоремы 1.

Теорема 2. *Если коэффициенты задачи* I *удовлетворяют* (1), *то для любого оптимального плана $X^0 \sum_{k=1}^{n} \operatorname{sgn} x_k^0 = m$. Если* (1) *выполняется как нестрогое неравенство, то существует оптимальный план с тем же свойством.*

Доказательство. Будем вести доказательство от противного. Предположим для определенности, что $x_i^0 \neq 0$, $x_{i+1}^0 = 0$ и что $\sum_{k=1}^{n} \operatorname{sgn} x_k = r < m$. Построим другой план X'

$$X' = \begin{cases} x'_{i,\, i+1} = 0, \\ x'_{i+1,\, i+1} = b_{i+1}, \\ x'_{kp} = x^0_{kp} \text{ во всех остальных случаях.} \end{cases}$$

Легко видеть, что X' — план задачи I. Из (6) имеем

$$CX^0 - CX' = \left[c_{i,\, i+1} \frac{b_{i,\, i+1}}{a_{i,\, i+1}} - b_{i+1}\, c_{i+1,\, i+1} \right] = b_{i,\, i+1} \left[\frac{c_{i,\, i+1}}{a_{i,\, i+1}} - \frac{c_{i+1,\, i+1}}{a_{i+1,\, i+1}} \right]. \tag{7}$$

Из (1) следует, что (7) строго больше нуля. Это противоречит оптимальности X^0. На этом первая часть теоремы доказана. Доказательство второй части теоремы непосредственно вытекает из (7).

Из теорем 1 и 2 следует, что если коэффициенты задачи удовлетворяют условию (1), то в любом оптимальном плане $\sum_{k=1}^{n} \operatorname{sgn} x_k^0 = m$ и компоненты этого плана удовлетворяют (6).

Таким образом, оптимальный план полностью определяется номерами своих ненулевых компонент x_i^0. Из этого можно заключить, что необходимо исследовать лишь планы, построенные по формулам (6), причем в каждом плане должно быть ровно m отличных от нуля компонент. Поскольку известно, что $x_1 > 0$ в любом плане, то для нахождения оптимального плана достаточно перебрать C_{n-1}^{m-1} различных планов и выбрать из них оптимальный. К сожалению, при больших n и m этот перебор практически неосуществим. Поэтому для нахождения решения нужно использовать методы, отличные от полного перебора.

3. Рассмотрим функциональные уравнения и методы их решения. Применим для задачи I метод последовательного анализа вариантов, основанный на решении функциональных уравнений. Обозначим через $F_j(i)$ функцию, значение которой определяется решением задачи

$$F_j(i) = \min \sum_{k,\, p=i}^{n} c_{kp} x_{kp}$$

при условии $\sum_{k=i}^{p} a_{kp} x_{kp} \geqslant b_p; \quad p = i, \ldots, n,$

$$\sum_{p=k}^{n} x_{kp} = x_k; \quad x_{kp} \geqslant 0; \quad k = i, \ldots, n; \quad p = i, \ldots, n,$$

$$\sum_{k=i}^{n} \operatorname{sgn} x_k \leqslant m - j. \tag{8}$$

Аргумент i в функции $F_j(i)$ указывает, что в (8) рассматриваются переменные, начиная с x_i. Индекс j определяет количество отличных от нуля переменных.

Пусть

$$f_j(il) = \sum_{k=i}^{l-1} \frac{c_{ik}}{a_{ik}} b_k + F_{j+1}(l). \tag{9}$$

Тогда рекуррентные соотношения для функций $F_j(i)$ будут иметь такой же вид, как и в [1]

$$F_j(i) = \min_{l>i} f_j(il), \qquad i = 1, \ldots, n; \qquad j = 0, \ldots, m-2,$$

$$F_{m-1}(i) = \sum_{k=i}^{n} \frac{c_{ik}}{a_{ik}} b_k. \tag{10}$$

Из определения функции $F_j(i)$ следует, что $F_0(1)$ — оптимальное значение функционала задачи I. Так же, как в [1], можно показать, что при фиксированном $j \neq 0$ достаточно рассматривать $i \in [j+1, n-m+j+1]$. Поэтому в (10) достаточно рассмотреть $l \leqslant l^0$, где $l^0 = n - m + j + 2$. Если для рассматриваемого i существует $a_{ik} = 0$ и $l' = \min_l \{l : a_{il} = 0\}$, то из следствия 2 теоремы 1 легко заключить, что в (10) достаточно рассмотреть $l \leqslant l^*(i) = \min\{l^0, l'\}$.

Заметим, что $l^*(i) \leqslant l^*(i-1)$. Решение уравнений (10) может быть получено любым методом динамического программирования [2, 3]. Реализация этих методов весьма проста на ЭВМ и не требует большого объема машинной памяти. Перед тем как непосредственно переходить к решению уравнений (10), целесообразно вычислить заранее коэффициенты $d_{ik} = c_{ik} / a_{ik}$ и числа $l'(i)$ для всех индексов i, k. Эта несложная предварительная процедура значительно сократит общее количество вычислений в решении.

Оказывается, что уравнения (10) обладают очень важным свойством, благодаря которому процесс решения можно еще значительно сократить.

Лемма 1. *Рассмотрим j-е уравнение системы* (10)

$$F_j(i) = \min_{i<l\leqslant l^*} f_j(il) = \min \left[\sum_{k=i}^{l-1} d_{ik} b_k + F_{j+1}(l) \right]. \tag{11}$$

Предположим, что решение этого уравнения для $i = i_0$ есть l_i. Заменим в этом уравнении d_{ik} на $\bar{d}_{ik} = d_{ik} + \Delta_{ik}$, где $\Delta_{ik} > 0$. Пусть l_2 — решение полученного уравнения для $i = i_0$, тогда $l_1 \leqslant l_2$.

Доказательство. Положим

$$f_1(l) = \sum_{k=i}^{l-1} d_{ik} b_k + F_{j+1}(l), \qquad f_2(l) = \sum_{k=i}^{l-1} \bar{d}_{ik} b_k + F_{j+1}(l). \tag{12}$$

Из оптимальности l_1 и l_2 имеем

$$f_1(l_1) \leqslant f_1(l_2); \quad f_2(l_2) \leqslant f_2(l_1). \tag{13}$$

Преобразуем $f_2(l)$ следующим образом

$$f_2(l) = \sum_{k=i}^{l-1} d_{ik} b_k + F_{j+1}(l) + \sum_{k=i}^{l-1} \Delta_{ik} b_k = f_1(l) + \sum_{k=i}^{l-1} \Delta_{ik} b_k. \tag{14}$$

Следовательно,

$$f_2(l_2) = f_1(l_2) + \sum_{k=i}^{l_2-1} \Delta_{ik} b_k \leqslant f_2(l_1) = f_1(l_1) + \sum_{k=i}^{l_1-1} \Delta_{ik} b_k,$$

$$0 \leqslant f_1(l_2) - f_1(l_1) \leqslant \sum_{k=i}^{l_1-1} \Delta_{ik} b_k - \sum_{k=i}^{l_2-1} \Delta_{ik} b_k. \qquad (15)$$

Так как $\Delta_{ik} b_k > 0$ для всех i, k, то из (15) следует, что $l_2 \geqslant l_2$. Лемма доказана.

Теорема 3. *Пусть $l(i)$ и $l(i-1)$ — произвольные решения (11) для i и $i-1$ соответственно, тогда $l(i) \geqslant l(i-1)$ для любых i.*

Доказательство. Очевидно, что для доказательства теоремы достаточно рассмотреть случай, когда $l(i) \leqslant l^*(i-1)$. Поэтому будем считать, что $l^*(i) = l^*(i-1) = l^*$. Обозначим через $l_1 = \max\limits_{l}\{l(i-1)\}$. Если $l_1 = i$, то утверждение теоремы очевидно. Поэтому в дальнейшем будем считать, что $l_1 > i$. Пусть l_0 — решение (11), в котором d_{ik} заменены на $d_{i-1,k}$, т. е.

$$F_j(i) = \min_{i<l\leqslant l^*} \left[\sum_{k=i}^{l-1} d_{i-1,\,k} b_k + F_{j+1}(l) \right] = \sum_{k=i}^{l_0-1} d_{i-1,\,k} b_k + F_{j+1}(l_0). \quad (16)$$

Так как $l_1 > i$, то

$$f_j(i-1, l_1) = d_{i-1,\,i-1} b_{i-1} + \sum_{k=i}^{l_1-1} d_{i-1,\,k} b_k + F_{j+1}(l_1).$$

Из оптимальности l_0 имеем

$$f_j(i-1, l_1) \geqslant d_{i-1,\,i-1} b_{i-1} + \sum_{k=i}^{l_0-1} d_{i-1,\,k} b_k + F_{j+1}(l_0) =$$

$$= \sum_{k=i-1}^{l_0-1} d_{i-1,\,k} b_k + F_{j+1}(l_0) = f_j(i-1, l_0). \qquad (17)$$

Из оптимальности l_1 имеем $f_j(i-1,\ l_1) \leqslant f_j(i-1,\ l_0)$, следовательно,

$$f_j(i-1,\ l_1) = f_j(i-1,\ l_0). \qquad (18)$$

Из (17) и (18) следует, что l_1 — решение уравнения (16). Из леммы вытекает, что $l_1 \leqslant l(i)$, но $l_1 \geqslant l(i-1)$ и, значит, $l(i-1) \leqslant l(i)$. Теорема доказана.

Следствие 3. *Если для некоторого i_0 $l(i_0) = l^*$, $l^*(i_0) = \ldots = l^*(i_0 + r) = l^*$, то для всех $i = i_0 + 1, \ldots, i_0 + r$; $l(i) = l^*$.*

В любом методе динамического программирования порядок вычисления величин $F_j(i)$ для различных i не имеет никакого значения. Выше была исследована зависимость между различными решениями $l(i)$ уравнения (11). Доказанные результаты позволяют значительно сократить количество вычислений (в среднем), необходимое для вычисления одного значения $F_j(i)$. Для этого нужно производить вычисления $F_j(i)$ для различных i в определенном порядке. Например, можно предложить следующий порядок вычислений.

Предположим, что все допустимые i заполняют отрезок $[ab]$, где a и

b — натуральные числа. Все целые точки этого отрезка разбиваются на группы. Первая группа состоит из средней точки i_1' отрезка $[ab]$ *. Вторая группа точек состоит из средних точек i_1^2, i_2^2 отрезков $[ai_1']$ и $[i_1'b]$ и т. д. Пусть определена $(k-1)$-я группа точек $\{i_\xi^{k-1}\}$, при этом отрезок $[ab]$ оказался разбит на систему отрезков Δ_ξ^{k-1}, образуемых точками из групп $1, \ldots, k-1$. Тогда k-я группа точек $\{i_\xi^k\}$ будет состоять из середин отрезков $\Delta_\eta^{\xi-1}$. Этот процес закончится, когда все точки отрезка $[ab]$ разобьются на группы. После этого, используя полученное решение для i_1', вычисляется сначала значение $F_j(i)$ для $i = i_1'$, потом для i_1^2, i_2^2 и т. д. На k-м шаге вычисляются значения $F_j(i)$ для всех $i \in \{i_\xi^k\}$ на основе результатов, полученных для предшествующих групп. Такой порядок вычислений позволяет оптимально использовать информацию о вычислениях, произведенных ранее.

4. Проведем дальнейшее исследование задачи II. Во всех результатах, изложенных выше, условие (1) было весьма существенным. В самом деле, без него не выполнялось бы основное свойство оптимальных планов, установленное теоремой 1, не были бы справедливы функциональные уравнения (10). В то же время это условие является вполне естественным и с экономической точки зрения. Ниже будет показано, что это условие можно элиминировать для задачи II. Более того, будет показано, что если в задаче II заменить линейные **функции на вогнутые, то алгоритм решения** полученной задачи не изменится. Приступим к доказательству этих утверждений.

Лемма 2. *Если в задаче* II *найдутся такие* c_i *и* c_{i-1}, *что* $c_{i-1} \leqslant c_i$, *то существует оптимальный план, в котором* $x_i^0 = 0$.

Доказательство. Пусть X — оптимальный план, а $x_i \neq 0$. Построим новый план X^0

$$X^0 = \begin{cases} x_i^0 = 0, \\ \overset{0}{x}_{i-1} = x_i + x_{i-1}, \\ x_k^0 = x_k, \quad k \neq i \neq i-1, \end{cases}$$

который удовлетворяет условиям (а2), (б2), кроме того, $CX - CX^0 \geqslant 0$. Из оптимальности X следует, что $CX - CX^0 \leqslant 0$. Отсюда вытекает, что если $c_i = c_{i-1}$, то X^0 — оптимальный план и $x_i^0 = 0$. Если $c_i > c_{i-1}$, то $CX - CX^0 > 0$. Это противоречит оптимальности X; значит, предположение о том, что $x_i \neq 0$, в этом случае было неверно. Лемма доказана.

Основываясь на этом результате, построим задачу II′, которая будет эквивалентна в известном смысле задаче II. Рассмотрим величины $c_1 - c_i$, $i = 1, \ldots, n$. Пусть $c_1 - c_i \leqslant 0$, $i = 1, \ldots, r-1$; $c_1 - c_r > 0$. Положим $x_1' = x_1$; $c_1' = c_1$;

$$x_2' = x_r; \quad c_2' = c_r; \quad b_1' = \sum_{k=1}^{r-1} b_k; \quad I_2 = \{i_1 i_2\} = \{1, r\}.$$

Пусть определены уже $x_1', \ldots, x_s'$; $c_1', \ldots, c_s'$; $b_1', \ldots, b_{s-1}'$; $I_s = \{i_1, \ldots, i_s\}$ и $x_s' = x_t$. Рассмотрим величины $c_t - c_i$; $i = t, \ldots, n$. Предположим, что $c_t - c_i \leqslant 0$; $i = t, \ldots, q-1$; $c_t - c_q > 0$. Если не существует такого q, то полагаем $b_s' = \sum_{k=t}^{n} b_k$, $n' = s$. На этом формирование задачи II′

---

* Если m и n — целые числа, то средней точкой отрезка $[mn]$ назовем число $[(m+n)/2]$.

заканчивается. Если существует указанное выше q, то полагаем $x'_{s+1} = x_q$,

$$c'_{s+1} = c_q; \quad b_s' = \sum_{t=i}^{q-1} b_t; \quad I_{s+1} = \{i_1, \ldots, i_{s+1}\}, \quad i_{s+1} = q.$$

Ясно, что такой процесс после конечного числа шагов закончится. Построим новую задачу минимизации функции

$$\left.\begin{aligned} F &= \sum_{k=1}^{n'} c_k' x_k' \\ \text{при условии } \sum_{k=1}^{s} x_k' &\geqslant \sum_{k=1}^{s} b_k' \\ x_s' \geqslant 0, \quad s &= 1, \ldots, n', \\ \sum_{k=1}^{n'} \operatorname{sgn} x_k' &\leqslant m. \end{aligned}\right\} \text{II}'$$

Пусть X' — оптимальный план задачи II′. Определим X^0

$$X^0 = \begin{cases} x_s^0 = x_k', & s = i_k \in I_{n'}, \\ x_s^0 = 0, & s \in \overline{I}_{n'}. \end{cases}$$

Основываясь на лемме 2, легко показать, что X^0 — оптимальный план задачи II. Таким образом, любой оптимальный план задачи II′ можно сопоставить с оптимальным планом задачи II. В задаче II′ коэффициенты удовлетворяют условию (1). Поэтому если $m < n'$, то оптимальный план задачи II′ удовлетворяет соотношениям (6), которые в данном случае будут иметь вид

$$x'_{\gamma_i} = \sum_{k=\gamma_i}^{\gamma_{i+1}-1} b_k'; \quad x'_{\gamma_m} = \sum_{k=\gamma_m}^{n} b_k', \tag{19}$$

где γ_i — номер ненулевой компоненты вектора X'. Если же $m \geqslant n'$, то вектор $X' = (b_1', \ldots, b_{n'}')$ — оптимальный план задачи II′. Подставляя вместо b_s' их выражения через b_k в формулах (19), получим, что X^0 удовлетворяет соотношениям (6) при любом n'. Таким образом, доказана нижеследующая теорема.

Теорема 4. *Для произвольных $c_i \geqslant 0$ существует оптимальное решение задачи* II, *удовлетворяющее условиям* (6).

Заметим, что если хотя бы одно $c_i < 0$, то функционал задачи II неограничен снизу.

Пусть G — множество планов задачи II, а $\hat{G}$ — выпуклая оболочка этого множества, тогда

$$\min_{X \in G} c(X) = \min_{X \in \hat{G}} c(X), \qquad X \in R^n$$

для любых вогнутых функций $c(X)$, так как G — многогранное множество. Более того, минимум $c(X)$, если он существует, достигается в вершине $\hat{G}$. Пусть X^0 — вершина $\hat{G}$, тогда очевидно, что $X^0 \in G$. Существует линейный функционал $c^0(X)$, такой, что $\min_{X \in \hat{G}} c^0(X) = c^0(X^0)$ и этот минимум единственный. Из теоремы 4 вытекает, что X^0 удовлетворяет соотношениям (6). Следовательно, любая вершина $\hat{G}$ удовлетворяет (6).

Рассмотрим теперь задачу II с произвольными вогнутыми функциями $c_i(x_i)$. Согласно сказанному выше, существует оптимальный план этой задачи, удовлетворяющий условиям (6). Поэтому для решения можно воспользоваться методом функциональных уравнений, изложенным в п. 3. Функциональные уравнения для этой задачи будут иметь вид

$$\begin{gathered} F_j(i) = \min_{l \geqslant i} f_j(il); \quad j = 0, \ldots, m-2; \quad i = 1, \ldots, n, \\ F_{m-1}(i) = c_i \left(\sum_{k=i}^{n} b_k \right); \quad f_j(il) = c_i \sum_{k=1}^{l-1} b_k + F_{j+1}(l). \end{gathered} \tag{20}$$

Эти уравнения почти не отличаются от (10), и метод их решения остается прежним.

Заметим, что если $c_i(x_i)$ — произвольные линейные функции, тогда для уравнений (20) справедлива теорема 3. Это непосредственно вытекает из леммы 2 и теоремы 4. Если же $c_i(x_i)$ вогнуты, то доказательство для этого случая не годится.

В описанной задаче II с произвольными вогнутыми функциями $c_i(x_i)$ было показано, что для нахождения оптимального решения можно применить метод функциональных уравнений, изложенный ранее. Для произвольных функций $c_i(x_i)$, в частности для выпуклых, аналогичные результаты неверны, так как оптимальные планы в этих задачах не являются вершинами многогранника $\hat{G}$.

5. Рассмотрим другой подход к исследованию задачи II. Возьмем опять задачу II при следующих дополнительных условиях

$$\begin{gathered} c_i(x_i) \text{ — произвольные функции}, \\ c_i(0) = 0^*; \quad x_i \leqslant d_i; \quad x_i \in R, \end{gathered} \tag{21}$$

R — подмножество в R^1, например, R может являться множеством целых чисел.

Задачу II с условиями (21) назовем задачей III. Введем замену переменных: $u_s = \sum_{k=1}^{s} x_k$. Тогда задача III примет следующий вид. Минимизировать функцию

$$\left.\begin{array}{ll} F = \sum_{s=1}^{n} c_s(u_s - u_{s-1}) & \\ \text{при условии } 0 \leqslant u_s - u_{s-1} \leqslant d_s, & \\ u_s \geqslant B_s;\ u_0 = 0, & (\text{а}3) \\ u_s \in R;\ s = 1, \ldots, n, & \\ \sum_{s=1}^{n} \operatorname{sgn}(u_s - u_{s-1}) \leqslant m, & (\text{б}3) \end{array}\right\} \text{(III)}$$

где $B_s = \sum_{k=1}^{s} b_k$.

---

* Это условие является несущественным и введено для упрощения записи.

Это типичная задача динамического программирования [2], несмотря на то, что в ней присутствует условие (63).

Следуя основным идеям динамического программирования, рассмотрим частичные задачи типа III, заключающиеся в минимизации функции

$$\left.\begin{aligned} &F(u^0_{i-1}, i, j) = \sum_{s=1}^{n} c_s(u_s - u_{s-1}) \\ &\text{при условии } 0 \leqslant u_s - u_{s-1} \leqslant d_s, \\ &\qquad u_s \geqslant B_s;\ u_s \in R, \\ &\qquad u_{i-1} = u^0{}_{i-1};\ s = i, \ldots, n, \\ &\sum_{s=i}^{n} \operatorname{sgn}(u_s - u_{s-1}) \leqslant m - j. \end{aligned}\right\} \text{(III}'\text{)}$$

Так как задача III является обобщением задачи II, то желательно было бы получить рекуррентные соотношения, которые являлись бы непосредственным обобщением соотношений (20). Обозначим оптимальное значение функционала задачи III′ при фиксированных $u^0{}_{i-1}$, i, j через $F_j(i, u_{i-1})$. При этом обозначении $F_0(1, 0)$ является оптимальным значением функционала задачи III.

Нетрудно показать, что рекуррентные соотношения для функций $F_j(i, u_{i-1})$ будут иметь вид

$$F_j(i, u_{i-1}) = \min_{u_i;\, l} [c_i(u_i - u_{i-1}) + F_{j+1}(l, u_{l-1})],$$

$$0 < u_i \ \ldots\ u_{i-1} \leqslant d_i; \quad u_i \in R, \tag{22}$$

$$u_i = u_{i+1} = \ldots = u_{l-1} \geqslant B_{l-1}; \quad l = i+1, \ \ldots, \ n+1; \quad i = j+1, \ldots, n;$$

$$F_j(n+1, u_n) = 0; \quad j = 0, \ldots, m-2; \qquad F_{m-1}(i, u_{i-1}) = \min c_i(u_i - u_{i-1});$$

$$u_i = u_{i+1} = \ldots = u_n \geqslant B_n.$$

Эти уравнения решаются теми же методами, что и (20). Однако для решения (22) требуется большее количество вычислений и больший объем машинной памяти, так как на каждом этапе нужно минимизировать функцию двух переменных. Если в (22) заменить $c_i(x_i)$ на линейные или вогнутые функции, положить $d_i = \infty$ и отбросить условие $u_i \in R$, то вид этих уравнений совершенно не изменится. Все же эти уравнения являются обобщением уравнений (20), т. е. они могут быть приведены к (20). Чтобы показать это, нужно воспользоваться результатами, изложенными в п. 4, из которых вытекает, что $u_{l-1}{}^0 - u_i{}^0 = B_{l-1} - B_i$ при фиксированных i и l. Поэтому аргумент u_{i-1} в функциях $F_j(i, u_{i-1})$ лишний.

Оказывается, что для задачи III можно выписать рекуррентные соотношения, отличные от (22). Для этого обозначим оптимальное значение функционала задачи III′ через $F_i(u_{i-1}, j)$. Тогда рекуррентные соотношения для этих функций будут иметь вид

$$F_i(u_{i-1}, j) = \left\{ \begin{aligned} &\left.\begin{aligned} &\min_{u_i} [c_i(u_i - u_{i-1}) + F_{i+1}(u_i, j+1)]; \\ &u_i \in R;\ 0 \leqslant u_i - u_{i-1} \leqslant d_i,\ u_i \geqslant B_i; \end{aligned}\right\} \text{если } u_i \neq u_{i-1} \\ &F_{i+1}(u_i, j),\ u_i \geqslant B_i; \qquad\qquad \text{если } u_i = u_{i-1} \end{aligned} \right. \tag{23}$$

$$F_i(u_{i-1}, m) \equiv 0; \quad F_{n+1}(u_n, j) \equiv 0; \quad i = 1, \ \ldots, \ n; \quad j = 0, \ \ldots, \ m.$$

С точки зрения удобства вычислений эти уравнения предпочтительнее уравнений (22), так как на каждом этапе нужен меньший объем машинной памяти. Однако количество этапов при решении (23) равно n, в то время, как в (22) количество этапов равно m.

Основное достоинство уравнений (22), (23) — их нечувствительность к виду функций $c_i(x_i)$ и допущение произвольных ограничений на переменные u_i. Чем больше ограничений типа $u_i \in R$, тем эффективнее решаются эти уравнения. Этот факт является общим для задач динамического программирования, и поэтому не имеет смысла заострять на нем внимание. Все же, несмотря на все свои положительные качества, уравнения (22), (23) хуже, чем уравнения (20).

6. Перейдем к обобщенной задаче стандартизации. Естественным обобщением задачи I является задача следующего вида:

$$\left.\begin{aligned} &\sum_{i=1}^{n} c_i x_i \to \min, && \text{(а4)} \\ &AX = b,\ X \geqslant 0, && \\ &\sum_{i=1}^{n_1} \operatorname{sgn} x_i \leqslant m,\ m \leqslant n_1 \leqslant n. && \text{(б4)} \end{aligned}\right\} \text{IV}$$

Матрица A имеет произвольную структуру. Будем предполагать, что задача IV разрешима. Как было отмечено ранее, задача IV является многоэкстремальной и аппарат линейного программирования для ее анализа неприменим.

Однако для этой задачи справедлив аналог одной из основных теорем линейного программирования о том, что решение задачи линейного программирования достигается на опорном плане задачи. Задачу, получающуюся из задачи IV отбрасыванием условия (б4), назовем усеченной к задаче IV и обозначим ее через IV$'$. Для удобства иногда будем говорить о задаче IV с матрицей A, а соответствующую ей задачу IV$'$ называть усеченной задачей с матрицей A.

Теорема 5. *Если X^0 — решение задачи* IV, *то можно указать такой опорный план X' усеченной задачи* IV$'$, *что $CX^0 = CX'$ и* $\sum\limits_{i=1}^{n} \operatorname{sgn} x_i{}^1 \leqslant m.$

Доказательство. Пусть X^0 — решение задачи IV и $I = \{i = x_i{}^0 = 0,\ i \leqslant n_1\}$; $\bar{A} = \{A^i\}\ i \,\bar{\in}\, I$, где A^i — i-й столбец матрицы A. Пусть I — опорное решение усеченной задачи с матрицей $\bar{A}$. Построим вектор X'

$$x_i{}' = \begin{cases} y_i; & i \,\bar{\in}\, I, \\ 0; & i \in I, \end{cases}$$

Очевидно, X' — удовлетворяет условию а4. Так как допустимое множество задачи IV содержится в допустимом множестве задачи IV$'$, то $CX' \leqslant CX^0$. С другой стороны, X' — план задачи IV, так как условие (б4) для X' выполнено по построению; отсюда, учитывая, что X^0 — решение задачи IV, получаем $CX' = CX^0$, и теорема доказана.

Отметим еще одно важное свойство задачи IV. Пусть задача IV$'$ не вырождена и ее решение не удовлетворяет условию (б4). Тогда справедлива следующая теорема.

Теорема 6. *Если* X^0 — *оптимальный опорный план задачи* IV, *то*

$$\sum_{i=1}^{n_1} \operatorname{sgn} x_i{}^0 = m.$$

Доказательство. Пусть X^0 — решение задачи IV и опорный план задачи IV$'$. Так как по условию X^0 не является решением задачи IV$'$, то, выполнив один шаг симплекс-метода, начиная с X^0, найдем X', для которого $CX^0 > CX'$ в силу невырожденности задачи IV$'$ [4].

Допустим, что утверждение теоремы неверно, т. е. $\sum\limits_{i=1}^{n_1} \operatorname{sgn} x_i{}^0 \leqslant m-1$, и покажем, что X' — план задачи IV$'$. В самом деле, X' однозначно определяется сменой базисов $Б^0$ на $Б^1$, которые соответствуют X^0 и X'. При этой смене одна переменная (вектор A^ξ) выводится из базиса $Б^0$ и становится равной нулю. Другая переменная (вектор A^η), не входящая ранее в базис $Б^0$, вводится вместо первой и получает отличное от нуля значение. Следовательно, число ненулевых компонент с номерами от 1 до n при переходе от $Б^0$ к $Б'$ может увеличиться не более, чем на единицу, т. е. $\sum\limits_{i=1} \operatorname{sgn} x_i{}' \leqslant m$. Следовательно, X' — план задачи IV; причем по построению $CX' < CX^0$, что невозможно, так как X^0 — решение задачи IV. Полученное противоречие доказывает теорему.

Итак, можно заключить, что для решения задачи IV необходимо исследовать только опорные планы усеченной задачи IV$'$, для которых выполняется условие (б4), Но количество этих планов, даже строго удовлетворяющих (б4), чрезвычайно велико, хотя и ограничено сверху числом $C_{n_1}^m \cdot C_{n-n}^{r-m}$, где r — количество строк матрицы A. При больших r, n, m было бы непосильно отыскать оптимальный план путем перебора всех допустимых опорных планов. Поэтому желательно иметь вычислительную схему, позволяющую избегать полный перебор.

Таблица 1

i	1	2	3	4	5	6	7	8	9	10
c	20	18	15	14	10	7	5	4	2	1
b	3	8	5	5	3	6	6	1	10	3

Следует отметить, что задача IV в общем виде еще не была исследована. В литературе встречались лишь частные модели, которые сводятся к этой задаче. Причем, как правило, эти модели исследовались только с экономической точки зрения. Во всех работах, посвященных этим задачам, предлагалось находить решение методом Гомори либо методом «ветвей и границ». Но эти методы для практических задач совершенно неприменимы. К сожалению, подход, изложенный в этой работе, тоже оказался неприменим к задаче IV.

7. Проиллюстрируем метод, изложенный выше, на небольшом примере. Рассмотрим задачу II, в которой $n = 10$, $m = 4$, c_i и b_i даны в табл. 1. Выпишем сначала рекуррентные соотношения для $F_j(i)$.

Таблица 2

		$F_1(8)=27$		$F_2(9)=23$		$F_3(10)=3$		1
		$i=8$	$l=9$	$i=9$	$l=10$	$i=10$		
		$F_1(7)=58$		$F_2(8)=30$		$F_3(9)=26$		2
		$i=7$	$l=9$	$i=8$	$l=9$	$i=9$		
		$F_1(6)=103$		$F_2(7)=61$		$F_3(8)=56$		3
		$i=6$	$l=7$	$i=7$	$l=9$	$i=8$		
		$F_1(5)=144$		$F_2(6)=117$		$F_3(7)=100$		4
		$i=5$	$l=6$	$i=6$	$l=9$	$i=7$		
		$F_1(4)=229$		$F_2(5)=186$		$F_3(6)=182$		5
		$i=4$	$l=6$	$i=5$	$l=9$	$i=6$		
		$F_1(3)=312$		$F_2(4)=294$		$F_3(5)=290$		6
		$i=3$	$l=6$	$i=4$	$l=6$	$i=5$		
$F_0(1)=532$		$F_1=495$		$F_2(3)=377$		$F_3(4)=474$		7
$i=1$	$l=3$	$i=2$	$l=6$	$i=3$	$l=6$	$i=4$		
$j=0$		$j=1$		$j=2$		$j=3$		N

$$F_0(1)=\min_{l>1}\left[c_1\sum_{k=1}^{l-1}b_k+F_1(l)\right];\quad l=2,\ldots,8;$$

$$F_1(i)=\min_{l>i}\left[c_i\sum_{k=i}^{l-1}b_k+F_2(l)\right];\quad l=3,\ldots,9;\quad i=2,\ldots,8;$$

$$F_2(i)=\min_{l>i}\left[c_i\sum_{k=i}^{l-1}b_k+F_3(l)\right];\quad l=4,\ldots,10;\quad i=3,\ldots,9;$$

$$F_3(i)=c_i\sum_{k=i}^{n}b_k;\quad i=4,\ldots,10.$$

Все вычисления удобно проводить при помощи табл. 2, в которую будут заноситься значения $F_j(i)$ и $l_j(i)$ — оптимальное l, на котором достигается минимум $f_j(i, l)$, т. е. $l_j(i) = l^* : F_j(i) = f_j(i, l^*)$.

Вся таблица состоит из четырех столбцов $j = 0, 1, 2, 3$. Каждая клетка j-го столбца разбита на три подклетки. В верхнюю подклетку записывается значение $F_j(i)$, номер i — в левую подклетку; в правой подклетке помещается число $l_j(i)$.

Сначала в таблицу записываются числа i. После этого начинается процесс заполнения таблицы, состоящий из четырех этапов. Каждый этап имеет семь шагов.

Этап 1. Вычисляются значения $F_3(i)$; $i = 4, \ldots, 10$ по формулам (24) и результаты заносятся в соответствующие клетки последнего столбца.

Этап 2. Заполняется следующий столбец $j = 2$. В п. 3 был описан порядок вычислений $F_j(i)$ для различных i, при котором требуется наименьшее количество вычислений (в среднем) для всех i. Для простоты изложения выберем не самый оптимальный путь заполнения таблицы. Будем заполнять таблицу «сверху вниз». Для вычисления $F_2(i)$ вычислим значения $f_2(i, l)$ для всех $l > i$ и выберем из них наименьшее. На первом шаге вычисляем $F_2(9)$, которое будет равно $f_2(9, 10) = c_9 \sum\limits_{k=9}^{9} b_k + F_3(10) = 23$. Поэтому заносим в верхнюю подклетку число 23, а в правую подклетку число 10. На этом первый шаг второго этапа заканчивается. На втором шаге вычисляются $f_2(8, l)$; $l = 9, 10$ и выбирается наименьшее. Оно будет равно f_2 (8, 9) $= 30$. Поэтому в верхнюю подклетку записываем число 30, а в правую — 9 и переходим к следующему шагу. На следующем шаге нужно вычислить $f_2(7, l)$; $l = 8, 9, 10$; но из теоремы 3 следует, что достаточно рассмотреть только $l = 8, 9$, так как $l_2(8) = 9$ и т. д. На последнем шаге второго этапа нужно было бы вычислить $f_2(3, l)$; $l = 4, \ldots, 10$. Но на предыдущем шаге было найдено, что $l_2(4) = 6$; поэтому для вычисления $F_2(3)$ достаточно рассмотреть $l = 4, 5, 6$.

Этап 3. Совершенно так же, как и на предыдущем этапе, заполняется второй столбец таблицы $j = 1$.

Этап 4. Этот этап состоит из одного шага. Вычисляются значения $f_0(1, l)$; $l = 2, \ldots, 8$ и выбирается наименьшее. Получаем $F_0(1) = f_0(1, 3) = 532$. После этого заполняется последняя клетка таблицы. Таким образом, на последнем этапе заполнения таблицы вычисляется оптимальное значение функционала задачи, т. е. значение $F_0(1)$. Для того, чтобы закончить решение задачи, нужно еще указать оптимальный план. Это делается следующим образом. Выбирается число в правой подклетке первого столбца ($j = 0$) $l = 3$, затем во втором столбце ($j = 1$) выбирается клетка, в которой $i = 3$, и выбирается из этой клетки число, стоящее в правой подклетке, $l = 6$. Затем аналогичным приемом в следующем столбце выбирается число $l = 9$. Выделенные таким образом числа 1, 3, 6, 9 являются номерами компонент, отличных от нуля, в оптимальном плане. Значения этих компонент вычисляются по формулам (6), т. е.

$$x_1{}^0 = \sum_{k=1}^{2} b_k = 11; \quad x_3{}^0 = \sum_{k=3}^{5} b_k = 13;$$

$$x_6{}^0 = \sum_{k=6}^{8} b_k = 13; \quad x_9{}^0 = \sum_{k=9}^{10} b_k = 13.$$

Итак, оптимальный план задачи есть вектор $X^0 = (11, 0, 13, 0, 0, 13, 0, 0, 13, 0)$, оптимальное значение функционала равно 532.

Здесь была использована не самая оптимальная схема заполнения таблицы, однако эта схема более проста для расчетов на ЭВМ. Оптимальная схема заполнения таблицы состоит в том, что строчки в каждом столбце заполняются в следующем порядке: 4, 2, 1, 3, 6, 5, 7.

Если за единицу вычисления принять вычисление одного значения $f_j(i, l)$, то количество вычислений без учета теоремы 3 равнялось бы 70, по схеме, примененной выше,— 51, по оптимальной — 50.

Следовательно, даже в малой задаче выигрыш получился значительный. Если же n, m, $n - m$ большие, то выигрыш будет гораздо больше.

Автор выражает искреннюю благодарность А. А. Фридману за постановку задачи и постоянное внимание к данной работе.

ЛИТЕРАТУРА

1. М. М. Беркович. Решение одной комбинаторной задачи. Экономика и матем. методы, 1968, т. IV, вып. 2.
2. Р. Беллман. Динамическое программирование. М., Изд. иностр. лит., 1960.
3. Р. Беллман, С. Дрейфус. Прикладные задачи динамического программирования. М., «Наука», 1965.
4. Д. Б. Юдин, Е. Г. Гольштейн. Задачи и методы линейного программирования. М., «Сов. радио», 1961.

Поступила в редакцию
31 V 1968