Управляемые системы, ИМ СО АН СССР, 1984 г., вып. 24

УДК 519.8+519.854

АЛГОРИТМ РЕШЕНИЯ ЗАДАЧИ ОПТИМАЛЬНОГО ВЫБОРА ДИНАМИЧЕСКОГО РЯДА ИЗДЕЛИЙ

В.Л.Береснев, Г.И.Ибрагимов, Ю.А.Кочетов

Модели оптимального выбора динамических рядов изделий отличаются от моделей, рассмотренных в [1, 2], тем, что описывают ситуацию многократного пересмотра состава системы изделий (состава группы однородной продукции), происходящего под влиянием изменения с течением времени различных технико-экономических факторов, связанных с данной системой изделий (группой однородной продукции).

В работе приводится математическая модель оптимизации выбора динамического ряда, аналогичная рассмотренной в [3], и предлагается алгоритм решения этой задачи. В основе алгоритма лежит вычислительная схема типа ветвей и границ. Описание алгоритма ведется применительно к некоторой задаче, к которой сводится исследуемая модель и которая названа обобщенной задачей. Предлагаемый алгоритм оптимизации выбора динамического ряда изделий реализован в виде программы для ЭВМ БЭСМ-6. Приводятся некоторые показатели качества этого алгоритма, обобщающие накопленный опыт работы с ним.

§ 1. Математическая модель оптимизации выбора динамического ряда изделий

1. Содержательная формулировка задачи оптимального выбора динамического ряда изделий выглядит следующим образом.

Пусть на фиксированном отрезке времени, для которого рассматривается вопрос о выборе (пересмотре) состава исследуемой системы изделий, фиксированы моменты времени, называемые <u>контрольными</u>. Считаем, что в эти моменты производится проверка дееспособности системы.

Предположим, что известен перечень образцов изделий, которые в принципе могут быть использованы для формирования системы на рассматриваемом отрезке времени. Этот список может включать в себя образцы как существующих изделий, так и перспективных (подготовленных или намеченных к производству). Такой перечень образцов назовем <u>исходным рядом</u> изделий.

Предположим также, что для всякого контрольного момента задан перечень видов работ (с указанием количества единичных работ каждого вида), выполнение которых должно быть обеспечено исследуемой системой изделий. Этот перечень

назовем областью применения изделий в данный контрольный момент. Объединение таких областей по всем контрольным моментам будем называть областью применения изделий на всем отрезке времени.

Рядом изделий в данный контрольный момент назовем некоторую совокупность образцов изделий из исходного ряда, предназначенных для выполнения работ области применения этого контрольного момента, а динамическим рядом – последовательность рядов изделий по всем контрольным моментам. Динамический ряд изделий будем считать допустимым, если система изделий соответствующего количественного состава способна обеспечить выполнение всех работ области применения каждого контрольного момента. При этом предполагается, что известен критерий выполнения работ каждого вида из области применения.

Суммарными затратами, связанными с динамическим рядом, считаем сумму затрат на всех этапах жизненного цикла изделий, формирующих систему.

Задача оптимального выбора динамического ряда изделий состоит в отыскании такого варианта допустимого динамического ряда изделий и соответствующего количественного состава системы изделий и такого назначения исполнителей работ каждого вида области применения, чтобы суммарные затраты были минимальными.

2. Занумеруем контрольные моменты рассматриваемого отрезка времени числами $1, 2, \dots, T$. Интервал времени до 1-го контрольного момента назовем 1-м промежутком, а между $t-1$-м и t-м контрольными моментами – t-м промежутком.

Множеством номеров $I = \{1, \dots, m\}$ зададим исходный ряд изделий, а множеством номеров $J(t)$ – область их применения в t-й контрольный момент. Считаем, что множества $J(t)$, $t = 1, \dots, T$, попарно не пересекаются и, следовательно, множество $J = \bigcup J(t) = \{1, \dots, n\}$ задает область применения изделий на всем отрезке времени. Для $j \in J$ обозначим через $t(j)$ такой номер t, что $j \in J(t)$.

Этапы разработки, производства и эксплуатации изделий в предлагаемой модели представляются через соответствующие затраты. Для $i \in I$ считаем известными величину $c_i^o \geq 0$, равную затратам на разработку изделия i-го образца, и величину $c_i^п$, равную затратам на производство одного изделия этого образца. Для $i \in I$, $t = 1, \dots, T$ считаем известной величину $c_{it}^э$, равную затратам на эксплуатацию одного изделия i-го образца на t-м промежутке. При этом предполагаем, что на t-м промежутке эксплуатируются только те изделия, которые используются для выполнения работ области применения t-го контрольного момента.

Относительно этапа использования примем следующее предположение. Считаем, что единичная работа каждого вида выполняется нарядом изделий одного образца. В соответствии с этим для $i \in I$, $j \in J$ считаем известными

величину $p_{ij} > 0$, равную количеству изделий i-го образца, потребных для выполнения работ j-го вида, и величину $c^{в}_{ij} \geqslant 0$, равную необходимым при этом затратам.

Введем переменные $x_i \in \{0, 1\}$, $i \in I$, определяющие образцы, которые формируют динамический ряд изделий, и переменные $x_{ij} \geqslant 0$, $i \in I$, $j \in J$, задающие исполнителей работ области применения. Величина x_i принимает значение 1, если изделие i-го образца входит в ряд хотя бы в один контрольный момент, и 0 - в противном случае. Величина x_{ij} равняется доле единичных работ j-го вида, выполняемых изделиями i-го образца. Введем, кроме того, переменные $v_{i\tau} \geqslant 0$, $i \in I$, $\tau = 1, \dots, T$, и переменные $\bar{v}_{i\tau} \geqslant 0$, $i \in I$, $\tau = 1, \dots, T$, задающие соответственно объемы производства изделий и количественный состав системы изделий. Величина $v_{i\tau}$ равняется объему производства изделий i-го образца на τ-м промежутке, а величина $\bar{v}_{i\tau}$ — требуемому количеству изделий этого образца в τ-й контрольный момент.

Задача оптимального выбора динамического ряда изделий записывается следующим образом:

$$F\left[(x_i)\,(x_{ij})\,(v_{i\tau})\right] =$$

$$= \sum_{i \in I} \left\{ c^{о}_i x_i + \sum_{\tau=1}^{T} \left\{ c^{п}_i v_{i\tau} + c^{э}_{i\tau} \bar{v}_{i\tau} \right\} + \sum_{j \in J} c^{в}_{ij} x_{ij} \right\} \longrightarrow$$

$$\longrightarrow \min_{(x_i)\,(x_{ij})\,(v_{i\tau})} ; \qquad (1)$$

$$\sum_{i \in I} x_{ij} = 1, \quad j \in J; \qquad (2)$$

$$x_i \geqslant x_{ij}, \quad j \in J, \quad i \in I; \qquad (3)$$

$$\bar{v}_{i\tau} = \sum_{j \in J(\tau)} p_{ij} x_{ij}, \quad i \in I, \ \tau = 1, \dots, T; \qquad (4)$$

$$\bar{v}_{i\tau} \leqslant \sum_{t=1}^{\tau} v_{it}, \quad i \in I, \quad \tau = 1, \dots, T; \qquad (5)$$

$$x_i \in \{0, 1\}, \quad i \in I; \qquad (6)$$

$$x_{ij}, \ v_{i\tau} \geqslant 0, \quad i \in I, \quad \tau = 1, \dots, T. \qquad (7)$$

Если $(x_i^*)\,(x_{ij}^*)\,(v_{i\tau}^*)$ - оптимальное решение данной задачи, то соответствующий динамический ряд изделий представляет собой последовательность

$$\{i\in I\mid \bar{v}_{i1}^*>0\},\ \{i\in I\mid \bar{v}_{i2}^*>0\},\dots,\{i\in I\mid \bar{v}_{iT}^*>0\}.$$

§ 2. Вычислительная схема типа ветвей и границ

1. В настоящем параграфе предлагается общая вычислительная схема типа ветвей и границ с односторонним ветвлением [1, 3, 4], которая лежит в основе построения алгоритма решения задачи (1)-(7).

Описание вычислительной схемы будем вести применительно к экстремальной задаче общего вида:

$$f(x)\to\min_x;$$

$$x\in\mathcal{D}.$$

Считаем, что задано разбиение множества $\mathcal{D}$ на конечное число подмножеств, называемых <u>базисными</u>. Далее будем рассматривать только такие подмножества $d\subset\mathcal{D}$, которые являются объединением некоторого числа базисных подмножеств. Если $\mathcal{D}$ - конечное множество, то в качестве базисных могут выступать одноэлементные подмножества. Пусть $d\subset\mathcal{D}$. <u>Оптимальным решением</u> задачи <u>на подмножестве</u> d^* будем называть оптимальное решение исходной задачи с дополнительным ограничением $x\in d$.

<u>Нижней границей</u> назовем вещественную функцию $H(d)$, определенную на множествах $d\subset\mathcal{D}$, такую, что $H(d)\le f(x)$ при $x\in d$. <u>Ветвлением</u> назовем функцию $B(d)$, определенную на небазисных множествах $d\subset\mathcal{D}$ и ставящую в соответствие множеству d некоторое его собственное подмножество d'. Функцией <u>выбора выделенного решения</u> назовем функцию $x(d)$, определенную на базисных и некоторых небазисных множествах $d\subset\mathcal{D}$ и ставящую в соответствие множеству d некоторое решение $x\in d$, близкое по целевой функции к оптимальному решению задачи на подмножестве d. В качестве выделенного решения может выступать, в частности, и само это оптимальное решение.

2. Вычислительная схема состоит из некоторого числа однотипных шагов, на каждом из которых рассматривается множество $t\subset\mathcal{D}$ не отброшенных к данному шагу решений, подмножество решений $t'\subset t$, проверяемых на данном шаге, и рекордное решение $x^0\in\mathcal{D}$, являющееся наилучшим из просмотренных к данному шагу решений. На первом шаге $t=t'=\mathcal{D}$, а x^0 - произвольный элемент множества $\mathcal{D}$.

Шаг начинается с проверки множества t', цель которой установить, можно ли отбросить это подмножество. Множество считается проверенным в одном из двух случаев: либо $H(t') \geqslant f(x^0)$, либо на t' определена функция выбора выделенного решения. При этом во втором случае, если $f(x^0) > f(x(t'))$, полагаем $x^0 = x(t')$.

Пусть множество t' проверено. Если $t \setminus t' = \emptyset$, то алгоритм заканчивает работу. Если $t \setminus t' \neq \emptyset$, то полагаем $t = t \setminus t'$, $t' = B(t)$ и переходим к следующему шагу. Пусть множество t' проверить не удается. Тогда полагаем $t' = B(t')$ и переходим к следующему шагу.

3. Заметим, что алгоритм, работающий по описанной схеме, останавливается через конечное число шагов. Это следует из того, что через конечное число шагов множество неотброшенных решений становится объединением меньшего числа базисных подмножеств. Последнее вытекает из следующих фактов. Во-первых, на каждом шаге либо отбрасываем проверяемое множество, либо переходим к рассмотрению подмножества этого множества, являющегося объединением меньшего числа базисных подмножеств. Во-вторых, базисное подмножество после проверки всегда отбрасывается.

Заметим далее, что если функция выбора выделенного решения не дает, вообще говоря, оптимального решения на подмножестве, то алгоритм, работающий по приведенной схеме, является приближенным. Точность получаемого с его помощью решения определяется "точностью" функции выбора выделенного решения. Если эта функция такова, что $f(x(t)) \leqslant (1+\varepsilon) f(x^*(t))$, где $x^*(t)$ - оптимальное решение на подмножестве t, то $f(x^0) \leqslant (1+\varepsilon) f(x^*)$, где x^* - оптимальное решение задачи.

Действительно, предположим противное, и пусть $f(x^0) > (1+\varepsilon) f(x^*)$. Тогда решение x^* на каком-то шаге алгоритма отброшено вместе с некоторым подмножеством t'. Если множество t' проверено по первому признаку, то получаем $f(x^0) \leqslant H(t') \leqslant f(x^*) < (1+\varepsilon) f(x^*)$, а если по второму, то $f(x^0) \leqslant f(x(t')) \leqslant (1+\varepsilon) f(x^*(t')) = (1+\varepsilon) f(x^*)$. В обоих случаях вступаем в противоречие с принятым неравенством, что и показывает требуемое.

Из доказанного следует, что если функция выбора выделенного решения дает оптимальное решение на подмножестве, то алгоритм является точным и в результате его работы получаем оптимальное решение.

В любом случае для решения x^0, полученного с помощью алгоритма, работающего по приведенной схеме, имеет место апостериорная оценка точности

$$f(x^0) \leqslant (1+\varepsilon) f(x^*), \quad \text{где } \varepsilon = \frac{f(x^0) - H(\mathcal{D})}{H(\mathcal{D})}.$$

§ 3. Алгоритм ветвей и границ для обобщенной задачи

Рассмотрим задачу, обобщающую сформулированную задачу оптимального выбора динамического ряда изделий, и построим алгоритм ее решения, приняв за основу описанную в предыдущем параграфе вычислительную схему.

Пусть заданы множества $A=\{1,\dots,\alpha\}$, $Y=\{1,\dots,n\}$, подмножества $A_s \subset A$, $s=1,\dots,S$, и величины C_s^0, $s=1,\dots,S$. Пусть, кроме того, заданы функции $\varphi_i(y_1,\dots,y_n)\geq 0$, $i\in A$, где $0\leq y_j\leq 1$, $j\in Y$. Относительно каждой из функций будем считать, что:

а) $\varphi_i(0,\dots,0)=0$;

б) $\varphi_i(y_1,\dots,y_n)+\varphi_i(y'_1,\dots,y'_n)\geq\varphi_i(y_1+y'_1,\dots,y_n+y'_n)$;

в) $\varphi_i(y_1,\dots,y_n)\geq\sum\limits_{j\in Y}c_{ij}y_j$, где $c_{ij}\geq 0, j\in Y$ -

некоторые величины, не все равные нулю.

Рассмотрим следующую задачу:

$$\Phi[(x_i)(x_{ij})]=$$

$$=\sum_{s=1}^{S}C_s^0\Big(1-\prod_{i\in A_s}(1-x_i)\Big)+\sum_{i\in A}\varphi_i(x_{i1},\dots,x_{in})\to\min_{(x_i)(x_{ij})}; \tag{8}$$

$$\sum_{i\in A}x_{ij}=1,\quad j\in Y; \tag{9}$$

$$x_i\geq x_{ij}\geq 0,\quad i\in A,\ j\in Y; \tag{10}$$

$$x_i\in\{0,1\},\quad i\in A. \tag{11}$$

Построим алгоритм решения данной задачи, в котором перебор (ветвление) ведется только по переменным x_i, а значения остальных переменных отыскиваются с помощью более эффективной процедуры. Для этого конкретизируем элементы общей вычислительной схемы ветвей и границ следующим образом.

1. Подмножества решений будем задавать с помощью частичных решений. Частичным решением (вектором) назовем булев вектор $(x_{i_1},x_{i_2},\dots,x_{i_q})$, $0\leq q\leq\alpha$, где $\langle i_1,i_2,\dots,i_q\rangle$ – упорядоченная выборка элементов множества A, а продолжением этого частичного вектора – всякое решение $(x'_i)\,(x'_{ij})$ задачи (8)-(11) такое, что $x'_{i_\ell}=x_{i_\ell}$, $\ell=1,\dots,q$. Множество продолжений частичного решения $(x_{i_1},\dots,x_{i_q})$

обозначим $\pi(x_{i_1}, \ldots, x_{i_q})$. Базисными подмножествами решений будем считать множества вида $\pi(x_1, \ldots, x_\alpha)$. Понятно, что всякое множество $\pi(x_{i_1}, \ldots, x_{i_q})$ есть объединение базисных множеств $\pi(\bar{x}_1, \ldots, \bar{x}_\alpha)$ таких, что $\bar{x}_{i_\ell} = x_{i_\ell}$, $\ell = 1, \ldots, q$. Для частичного решения $(x_{i_1}, \ldots, x_{i_q})$ положим $\bar{A} = \{i_1, \ldots, i_q\}$, $\bar{A}_1 = \{i \in \bar{A} \mid x_i = 1\}$, $\bar{A}_0 = \{i \in \bar{A} \mid x_i = 0\}$, $A' = A \setminus \bar{A}$, $A'_0 = A \setminus \bar{A}_0$.

2. <u>Нижняя граница.</u> Построим функцию $H(x_{i_1}, \ldots, x_{i_q})$ такую, что

$$H(x_{i_1}, \ldots, x_{i_q}) \leq \Phi[(x_i)(x_{ij})],$$

где (x_i) (x_{ij}) – решение из множества $\pi(x_{i_1}, \ldots, x_{i_q}), 0 \leq q \leq \alpha$.
Пусть $\mathcal{S}_1 = \{s \mid \bar{A}_1 \cap A_s \neq \emptyset\}$, $\mathcal{S}' = \{s \mid s \notin \mathcal{S}_1\}$.

Для любого решения $(x_i)(x_{ij})$ из множества $\pi(x_{i_1}, \ldots, x_{i_q})$ можем, очевидно, написать

$$\Phi[(x_i)(x_{ij})] \geq$$

$$\geq \sum_{s \in \mathcal{S}_1} c_s^0 + \sum_{s \in \mathcal{S}'} c_s^0 \Big(1 - \prod_{i \in A_s \cap A'_0} (1 - x_i)\Big) + \sum_{i \in A'_0} \sum_{j \in Y} c_{ij} x_{ij} \geq$$

$$\geq \sum_{s \in \mathcal{S}_1} c_s^0 + \sum_{s \in \mathcal{S}'} \max_{i \in A'_0 | x_i = 1} f_{is} + \sum_{j \in Y} \min_{i \in A'_0 | x_i = 1} c_{ij},$$

где

$$f_{is} = \begin{cases} c_s^0, & \text{если } i \in A_s, \\ 0, & \text{если } i \notin A_s. \end{cases}$$

Последнее выражение есть целевая функция задачи выбора оптимального набора строк пары матриц (см. [1, 3, 4]), и, стало быть, функция $H(x_{i_1}, \ldots, x_{i_q})$ может быть определена следующим образом:

$$H(x_{i_1}, \ldots, x_{i_q}) = \sum_{s \in \mathcal{S}_1} c_s^0 + \mu(W),$$

где $W = (w_{ij})(i \in A'_0, j \in Y)$ – оценочная матрица (см. [3, 4]) для пары матриц (F, C), $F = (f_{is})(i \in A'_0, s \in \mathcal{S}')$; $C = (c_{ij})$ $(i \in A'_0, j \in Y)$. Среди оценочных матриц выделяются тупиковые матрицы, которые и используются при вычислении нижней границы. Положим

$$\mathcal{I}_0(W) = \Big\{i \in A'_0 \,\Big|\, \sum_{s \in \mathcal{S}'} f_{is} = \sum_{j \in Y} (w_{ij} - c_{ij})\Big\},$$

$$\mathcal{I}_j(W) = \Big\{i \in A'_0 \,\Big|\, w_{ij} = \min_{k \in A'_0} w_{kj}\Big\}, \ j \in Y.$$

Оценочная матрица W называется <u>тупиковой</u>, если $\mathcal{J}_0(W) \cap \mathcal{J}_j(W) = \emptyset$ при любом $j \in Y$. Алгоритмы построения "хороших" тупиковых матриц, дающих значение величины $\mu(W)$, близкое к максимальному, рассмотрены в [3, 4].

3. <u>Выбор выделенного решения</u>. Определим прежде всего функцию выбора на базисных подмножествах. Для этого свяжем с подмножеством $\Pi(X_1, \dots, X_\alpha)$ следующую задачу:

$$\sum_{i \in A} \varphi_i(x_{i1}, \dots, x_{in}) \to \min_{(x_{ij})} ;$$

$$\sum_{i \in A} x_{ij} = 1, \quad j \in Y;$$

$$x_{ij} \geq 0, \quad i \in A, \ j \in Y;$$

$$x_{ij} = 0, \quad i \notin \bar{A}_1, \ j \in Y.$$

Очевидно, каждое решение (X_i) (x_{ij}) из множества $\Pi(X_1, \dots, X_\alpha)$ определяется соответствующим решением (x_{ij}) этой задачи.

Поскольку нетрудоемкий алгоритм отыскания оптимального решения сформулированной задачи, по которому естественно было бы определять выделенный элемент множества $\Pi(X_1, \dots, X_\alpha)$, удается построить только в некоторых частных случаях функций $\varphi_i(y_1, \dots, y_n)$, то в общем случае в качестве такого решения примем решение, которое получается посредством алгоритма, представляющего собой комбинацию градиентного и биградиентного алгоритмов.

Пусть $\tilde{A} \subset A$. Матрицу (x_{ij}) $(i \in A, \ j \in Y)$ такую, что

$$\sum_{i \in A} x_{ij} \leq 1, \quad j \in Y;$$

$$x_{ij} \geq 0, \quad i \in A, \ j \in Y;$$

$$x_{ij} = 0, \quad i \notin \tilde{A}, \ j \in Y,$$

назовем матрицей <u>частичных назначений множества</u> $\tilde{A}$. Матрицу частичных назначений множества $\tilde{A}$ такую, что

$$\sum_{i \in A} x_{ij} = 1, \quad j \in Y,$$

будем называть матрицей <u>назначений множества</u> $\tilde{A}$.

Пусть для всякого $j \in Y$ фиксирована целая положительная величина Δ_j. При заданной матрице частичных назначений для всякого $i \in A$ положим

$$\Delta\varphi^{+}_{ij} = \varphi_i(x_{i1},\dots,x_{ij-1}, x_{ij}+\frac{1}{\Delta_j}, x_{ij+1},\dots,x_{in}) - \\ - \varphi_i(x_{i1},\dots,x_{in}),$$

$$\Delta\varphi^{-}_{ij} = \varphi_i(x_{i1},\dots,x_{ij-1}, x_{ij}-\frac{1}{\Delta_j}, x_{ij+1},\dots,x_{in}) - \\ - \varphi_i(x_{i1},\dots,x_{in}).$$

Предлагаемый алгоритм получения решения рассматриваемой задачи состоит из двух этапов: построения начальной матрицы назначений множества $\bar{A}_1$ и ее улучшения.

Первый этап включает в себя $\sum\limits_{j\in Y}\Delta_j$ шагов, к началу каждого из которых имеется матрица (x_{ij}) частичных назначений множества $\bar{A}_1$. К первому шагу имеем $x_{ij}=0,\ i\in\bar{A}_1,\ j\in Y$. Шаг состоит в отыскании элементов $i_0\in\bar{A}_1,\ j_0\in\{j\in Y \mid \sum\limits_{i\in A} x_{ij} < 1\}$ таких, что

$$\Delta\varphi^{+}_{i_0 j_0} = \min_i \min_j \Delta\varphi^{+}_{ij}.$$

Далее полагаем $x_{i_0 j_0} = x_{i_0 j_0} + \frac{1}{\Delta_{j_0}}$ и переходим к следующему шагу.

Второй этап включает в себя конечное число шагов, к началу каждого из которых имеется матрица (x_{ij}) назначений множества $\bar{A}_1$. Шаг состоит в отыскании элементов $i_0\in\bar{A}_1,\ j_0\in Y,\ K_0\in\{i\in\bar{A}_1 \mid x_{ij_0}>0\}$ таких, что

$$\Delta\varphi^{+}_{i_0 j_0} + \Delta\varphi^{-}_{K_0 j_0} = \min_j \min_i \min_K \{\Delta\varphi^{+}_{ij} + \Delta\varphi^{-}_{Kj}\}.$$

Если $\Delta\varphi^{+}_{i_0 j_0} + \Delta\varphi^{-}_{K_0 j_0} < 0$, то полагаем $x_{i_0 j_0} = x_{i_0 j_0} + \frac{1}{\Delta_{j_0}}$, $x_{K_0 j_0} = x_{K_0 j_0} - \frac{1}{\Delta_{j_0}}$ и переходим к следующему шагу. В противном случае алгоритм заканчивает работу.

3. Поскольку выбор выделенного решения в проверяемом подмножестве предотвращает его дальнейшее ветвление, то укажем на способ отыскания выделенных решений в некоторых подмножествах $\pi(x_{i_1},\dots,x_{i_q})$, отличных от базисных. Сформулируем достаточное условие того, что оптимальным решением задачи (8)-(11) на подмножестве $\pi(x_{i_1},\dots,x_{i_q})$ является решение $(x_i)\ (x_{ij})$ такое, что $x_i=0$, если $i\in A'$.

Пусть $i' \in A'$, (x'_{ij}) - матрица частичных назначений множества $\{i'\}$. Пусть $(\bar{x}_{ij})$ - матрица частичных назначений множества $\bar{A}_1$ такая, что

$$\sum_{i \in A} x'_{ij} = \sum_{i \in A} \bar{x}_{ij}, \quad j \in Y.$$

Положим

$$V_{i'}\ [(\bar{x}_{ij})] = \sum_{i \in \bar{A}_1} \varphi_i(\bar{x}_{i1}, \ldots, \bar{x}_{in}) - \varphi_{i'}(x'_{i'1}, \ldots, x'_{i'n}).$$

Л е м м а 1. (Критерий оптимальности.) Если при любом $i' \in A'$ для всякой матрицы (x'_{ij}) частичных назначений множества $\{i'\}$ найдется соответствующая матрица $(\bar{x}_{ij})$ частичных назначений множества $\bar{A}_1$, при которой

$$V_{i'}\ [(\bar{x}_{ij})] \leq \sum_{s \in S' | i' \in A_s} \frac{c_s^0}{|A_s \setminus \bar{A}_0|},$$

то существует оптимальное решение $(x_i^*)\ (x_{ij}^*)$ задачи (8)-(11) на подмножестве $\pi\,(x_{i_1}, \ldots, x_{i_q})$ такое, что $x_i^* = 0$ при $i \in A'$.

Д о к а з а т е л ь с т в о. Рассмотрим некоторое решение $(x_i)\ (x_{ij})$ из множества $\pi\,(x_{i_1}, \ldots, x_{i_q})$ и предположим, что множество $A'_1 = \{i \in A' | x_i = 1\}$ непусто. Построим новое решение $(\tilde{x}_i)(\tilde{x}_{ij})$, занулив компоненты x_i, $i \in A'_1$, исходного решения, и покажем, что разность значений целевой функции Φ на построенном и исходном решениях неположительна. Отсюда и будет следовать утверждение леммы.

Для каждого $i' \in A'_1$ рассмотрим матрицу назначений $(x'_{ij}(i'))$ множества $\{i\}$, порожденную матрицей назначений (x_{ij}), т.е. такую, что $x'_{i'j}(i') = x_{i'j}$, $j \in Y$, и соответствующую матрицу частичных назначений $(\bar{x}_{ij}(i'))$ множества $\bar{A}_1$, обладающую указанным в условии леммы свойством. Положим

$$\tilde{x}_i = \begin{cases} x_i & \text{, если } i \notin A'_1, \\ 0 & \text{, если } i \in A'_1; \end{cases}$$

$$\tilde{x}_{ij} = x_{ij} + \sum_{i' \in A'_1} (\bar{x}_{ij}(i') - x'_{ij}(i')).$$

Для разности значений функции Φ на построенном решении $(\tilde{x}_i)(\tilde{x}_{ij})$ и исходном $(x_i)\ (x_{ij})$ можем написать

$$\Phi[(\tilde{x}_i)(\tilde{x}_{ij})] - \Phi[(x_i)(x_{ij})] =$$

$$= \sum_{i\in\bar{A}_1} \varphi_i(\tilde{x}_{i1},\dots,\tilde{x}_{in}) - \sum_{i\in\bar{A}_1\cup A'_1} \varphi_i(x_{i1},\dots,x_{in}) -$$

$$- \sum_{s\in S'|A_s\cap A'_1\neq\emptyset} c_s^0 = \sum_{i\in\bar{A}_1} \varphi_i\Big(x_{i1} + \sum_{i'\in A'_1} \bar{x}_{i1}(i'),\dots, x_{in} +$$

$$+ \sum_{i'\in A'_1} \bar{x}_{in}(i')\Big) - \sum_{i\in\bar{A}_1} \varphi_i(x_{i1},\dots,x_{in}) -$$

$$- \sum_{i'\in A'_1} \varphi_{i'}(x'_{i'1}(i'),\dots,x'_{i'n}(i')) - \sum_{s\in S'|A_s\cap A'_1\neq\emptyset} c_s^0 \le$$

$$\le \sum_{i'\in A'_1} \Big\{ \sum_{i\in\bar{A}_1} \varphi_i(\bar{x}_{i1}(i'),\dots,\bar{x}_{in}(i')) -$$

$$- \varphi_i(x'_{i'1}(i'),\dots,x'_{i'n}(i')) - \sum_{s\in S'|i'\in A_s} \frac{c_s^0}{|A_s\setminus\bar{A}_0|} \Big\} \le 0.$$

Лемма доказана.

Представим полученное достаточное условие оптимальности в форме, более удобной для построения алгоритмов проверки его выполнения. Пусть

$$V_{i'}^* = \max_{(x'_{ij})} \min_{(\bar{x}_{ij})} \Big\{ \sum_{i\in\bar{A}_1} \varphi_i(\bar{x}_{i1},\dots,\bar{x}_{in}) - \varphi_{i'}(x'_{i1},\dots,x'_{in}) \Big\}.$$

Понятно, что условие леммы выполняется тогда и только тогда, когда

$$V_{i'}^* \le \sum_{s\in S'|i'\in A_s} \frac{c_s^0}{|A_s\setminus\bar{A}_0|}$$

для всякого $i'\in A'$. Следовательно, схему проверки выполнения рассматриваемого критерия оптимальности можно представить следующим образом. Для всякого $i'\in A'$ последовательно вычисляем величины $V_{i'}^*$ и проверяем справедливость указанного неравенства. Если все неравенства выполняются, то оптимальное решение с нужным свойством существует, в противном случае, неизвестно, существует оно или нет.

К сожалению, нетрудоемкий алгоритм вычисления величины $V_{i'}^*$ удается построить только в некоторых частных случаях функций $\varphi_i(y_1,\dots,y_n)$. В общем случае предлагается приближенный алгоритм, аналогичный рассмотренному выше алгоритму отыскания выделенного решения.

Алгоритм состоит из двух этапов: построения начальной матрицы частичных назначений множества $\{i'\}$ и соответствующей матрицы частичных назначений множества $\bar{A}_1$ и улучшения матрицы частичных назначений множества $\{i'\}$.

Первый этап включает в себя $\sum\limits_{j\in Y}\Delta_j$ шагов, к началу каждого из них известны текущая матрица $(\bar{x}_{ij})$ частичных назначений множества $\bar{A}_1$ и наилучшая к данному шагу матрица $(\bar{x}^*_{ij})$ частичных назначений множества $\bar{A}_1$. К первому шагу имеем $\bar{x}_{ij}=\bar{x}^*_{ij}=0,\ i\in\bar{A}_1,\ j\in Y$. Шаг состоит в отыскании элементов $j_0\in\{j\in Y\,|\sum\limits_{i\in\bar{A}_1}\bar{x}_{ij}<1\},\ i_0\in\bar{A}_1$ таких, что

$$\Delta\varphi^+_{i_0j_0}-\Delta\varphi^+_{i'j_0}=\max_j\ \min_i\{\Delta\varphi^+_{ij}-\Delta\varphi^+_{i'j}\}.$$

Далее полагаем $\bar{x}_{i_0j_0}=x_{i_0j_0}+\dfrac{1}{\Delta_{j_0}}$ и, если $V_{i'}[(\bar{x}_{ij})]>V[(\bar{x}^*_{ij})]$, то $\bar{x}^*_{ij}=x_{ij},\ i\in\bar{A}_1,\ j\in Y$. После этого переходим к следующему шагу.

Второй этап включает в себя конечное число шагов, к началу каждого из них имеется матрица $(\bar{x}^*_{ij})$ частичных назначений множества $\bar{A}_1$. Шаг состоит в отыскании элементов $j_0\in Y,\ i_0\in\{i\in\bar{A}_1|\ \bar{x}^*_{ij_0}>0\}$ таких, что

$$\Delta\varphi^-_{i_0j_0}-\Delta\varphi^-_{i'j_0}=\max_j\ \min_i\{\Delta\varphi^-_{ij}-\Delta\varphi^-_{i'j}\}.$$

Если $\Delta\varphi^-_{i_0j_0}-\Delta\varphi^-_{i'j_0}>0$, то полагаем $\bar{x}^*_{i_0j_0}=\bar{x}^*_{i_0j_0}-\dfrac{1}{\Delta_{j_0}}$ и переходим к следующему шагу. В противном случае алгоритм заканчивает работу.

4. <u>Функция ветвления</u>. Рассмотрим частичное решение $(x_{i_1},\dots,x_{i_q})$ и свяжем с ним, кроме множества $\Pi(x_{i_1},\dots,x_{i_q})$, еще и множество, являющееся объединением множества $\Pi(x_{i_1},\dots,x_{i_q})$ и всех множеств вида $\Pi(x_{i_1},\dots,x_{i_{q'-1}},0)$, где номер $q'\le q$ такой, что $x_{i_{q'}}=1$. Предположим, что первое из этих множеств является множеством проверяемых решений, а второе – множеством неотброшенных решений. Результатом применения функции ветвления ко второму множеству будем считать подмножество $\Pi(x_{i_1},\dots,x_{i_q})$, а к первому – подмножество $\Pi(x_{i_1},\dots,x_{i_q},1)$, где номер $i_{q+1}\in A'$ определяется следующим образом.

Пусть W – тупиковая матрица для соответствующей пары матриц. Если $\mathcal{I}_0(W)\cap A'\ne\emptyset$, то в качестве i_{q+1} берется такой номер $i'\in\mathcal{I}_0(W)\cap A'$, для которого число элементов в множестве

$\{j \in Y \mid i' \in \mathcal{J}_j(W)\}$ наибольшее. Если $\mathcal{J}_0(W) \cap A' = \varnothing$ и проверка выполнения критерия оптимальности заканчивается отрицательным исходом, то в качестве i_{q+1} берется такой номер $i' \in A'$, для которого величина

$$V_{i'}\,[(\bar{x}^*_{ij})] - \sum_{s \in S' \mid i' \in A_s} \frac{c^0_s}{|A_s \setminus \bar{A}_0|}$$

наибольшая.

Покажем теперь, что при так определенной функции ветвления на каждом шаге алгоритма ветвей и границ множество t неотброшенных решений и множество t' проверяемых решений можно задавать некоторым частичным решением.

На первом шаге таким частичным решением является, очевидно, пустой частичный вектор $(q = 0)$. Предположим, что на некотором шаге множества t и t' задаются частичным решением $(x_{i_1}, \dots, x_{i_q})$. Возможны два исхода проверки множества t': либо его удается отбросить, и тогда функция ветвления применяется к множеству $t \setminus t'$, либо нет, и тогда функция ветвления применяется к множеству t'. Пусть имеет место первый случай и пусть $q' \leq q$ – наибольший номер такой, что $x_{i_{q'}} = 1$. Рассмотрим частичное решение $(x_{i_1}, \dots, x_{i_{q'-1}}, 0)$ и заметим, что оно задает, во-первых, множество $t \setminus t'$ и, во-вторых, множество $B(t \setminus t')$. Если номера q' с указанным свойством не существует, то, очевидно, $t = t'$ и алгоритм заканчивает работу. Пусть реализуется второй случай. Рассмотрим тогда частичное решение $(x_{i_1}, \dots, x_{i_q}, 1)$, где номер i_{q+1} определяется функцией ветвления, и заметим, что оно задает множество t и множество $B(t')$. Таким образом, в обоих случаях на следующем шаге алгоритма информация о множестве неотброшенных решений и множестве решений задается некоторым частичным решением.

§ 4. Сведение задачи оптимального выбора динамического ряда изделий к обобщенной задаче

1. Покажем, что для решения сформулированной в § 1 задачи оптимального выбора динамического ряда изделий может быть использован построенный в предыдущем параграфе алгоритм.

Имеет место

Л е м м а 2. Задача (1)-(7) сводится к задаче (8)-(11).

Д о к а з а т е л ь с т в о. Рассмотрим задачу (8)-(11) вида

$$\Phi[(x_i)(x_{ij})] = \sum_{i \in I} c^0_i x_i + \sum_{i \in I} \varphi_i(x_{i1}, \dots, x_{in}) \rightarrow \min_{(x_i)(x_{ij})}; \quad (12)$$

$$\sum_{i\in I} x_{ij} = 1, \quad j \in J; \tag{13}$$

$$x_i \geq x_{ij} \geq 0, \quad i \in I, \ j \in J; \tag{14}$$

$$x_i \in \{0,1\}, \quad i \in I, \tag{15}$$

где

$$\varphi_i\,(x_{i1}, \dots, x_{in}) =$$

$$= c_i^{п} \max_t \sum_{j\in J(t)} p_{ij}\, x_{ij} + \sum_{j\in J} (c_{ij}^{в} + c_{it(j)}^{э} p_{ij})\, x_{ij},$$

и построим по оптимальному решению данной задачи оптимальное решение задачи (1)-(7). Для этого заметим прежде всего, что если $(x_i)\ (x_{ij})$ и $(x_i)\ (x_{ij})\ (v_{i\tau})$ - решения соответственно задач (12)-(15) и (1)-(7) такие, что

$$\sum_{\tau=1}^{T} v_{i\tau} = \max_t \sum_{j\in J(t)} p_{ij}\, x_{ij}, \quad i \in I, \tag{16}$$

то

$$\Phi\,[(x_i)\,(x_{ij})] = F[(x_i)\,(x_{ij})\,(v_{i\tau})].$$

Пусть $(x_i^*)\ (x_{ij}^*)$ - оптимальное решение задачи (12)-(15). Для всякого $i \in I$ положим

$$v_{i1} = \sum_{j\in J(1)} p_{ij}\, x_{ij}^*,$$

$$v_{i\tau} = \max\Big\{0, \sum_{j\in J(\tau)} p_{ij}\, x_{ij}^* - \sum_{t=1}^{\tau-1} v_{it}\Big\}, \quad \tau = 2, \dots, T.$$

Очевидно, $(x_i^*)\ (x_{ij}^*)\ (v_{i\tau})$ - решение задачи (1)-(7), для которого справедливо равенство (16). Это решение является и оптимальным. Действительно, рассмотрим $(\tilde{x}_i)\ (\tilde{x}_{ij})\ (\tilde{v}_{i\tau})$ - оптимальное решение задачи (1)-(7) и заметим, что для него выполняется равенство (16). Следовательно, можем написать

$$F\,[(\tilde{x}_i)(\tilde{x}_{ij})(\tilde{v}_{i\tau})] = \Phi[(\tilde{x}_i)\,(\tilde{x}_{ij})] \geq \Phi[(x_i^*)\,(x_{ij}^*)] =$$

$$= F\,[(x_i^*)\,(x_{ij}^*)\,(v_{i\tau})].$$

Заметим далее, что функции $\varphi_i\,(y_1, \dots, y_n)$ рассмотренного вида удовлетворяют всем требуемым свойствам (см. § 3). Действительно, первые два свойства, очевидно, выполняются, а третье имеет место в силу неравенства

$$\max_t a_t \geq \sum_t \lambda_t a_t ,$$

где $\lambda_t \geq 0$ – произвольные величины такие, что $\sum_t \lambda_t \leq 1$. С учетом этого неравенства можем написать

$$\varphi_i(y_1,\dots,y_n) \geq \sum_{j\in J} (\lambda_{i\tau(j)}\, c_i^{п} p_{ij} + c_{ij}^{в} + c_{i\tau(j)}^{э} p_{ij})\, y_j .$$

Лемма доказана.

2. Понятно, что значение нижней границы зависит от выбранных величин λ_{it}, $i \in I$, $t = 1,\dots,T$. Ясно также, что поскольку мы заинтересованы в использовании "хорошей" нижней границы, значение которой близко к оптимальному значению целевой функции, то эти величины должны соответствующим образом изменяться в зависимости от проверяемого подмножества. Рассмотрим поэтому некоторый нетрудоемкий способ вычисления матрицы $(\lambda_{it})(i \in I_0',\ t = 1,\dots,T)$, дающей величину $H(x_{i_1},\dots,x_{i_q})$, близкую к максимальной. Здесь множество I_0' определяется по аналогии с множеством A_0'.

В основе этого способа лежит процедура пересчета при фиксированном частичном решении $(x_{i_1},\dots,x_{i_q})$ i-й строки, $i_0 \in I_0'$, матрицы $(\lambda_{it})(i \in I_0',\ t = 1,\dots,T)$. Пусть (F, C) – соответствующая пара матриц, вычисленная при данных величинах λ_{it}. Обозначим через F' и C' соответственно матрицы F и C, у которых вычеркнута i_0-я строка, и пусть $W' = (w_{ij}')$ – тупиковая матрица для пары (F', C'). Обозначим через W матрицу, полученную в результате приписывания к ней i_0-й строки с элементами вида

$$w_{i_0 j} = \lambda_{\tau(j)}\, c_{i_0}^{п} p_{i_0 j} + c_{i_0 j}^{в} + c_{i_0 \tau(j)}^{э} .$$

В качестве новых элементов i_0-й строки возьмем величины λ_t, $t = 1,\dots,T$, при которых величина $\mu(W)$ не сильно отклоняется от величины $\mu(W')$.

Для отыскания таких величин рассмотрим задачу

$$\sum_{j\in J} \psi_j(\lambda_{\tau(j)}) \to \max_{(\lambda_t)} ;$$

$$\sum_{t=1}^{T} \lambda_t = 1 ;$$

$$\lambda_t \geq 0, \quad t = 1,\dots,T ,$$

где

$$\psi_j(\lambda) = \min\{\min_i w_{ij}',\ w_{i_0 j}\} .$$

Решать эту задачу предлагается градиентным алгоритмом.

Пусть фиксирована целая положительная величина Δ. При заданном векторе (λ_t) для всякого $j \in J$ положим

$$\Delta \psi_j = \psi_j(\lambda_{\tau(j)} + \frac{1}{\Delta}) - \psi_j(\lambda_{\tau(j)}) .$$

Алгоритм включает в себя Δ шагов, к началу каждого из которых имеется вектор (λ_t). К первому шагу имеем $\lambda_t = 0, t = 1, \ldots, T$. Шаг состоит в отыскании такого номера t_0, что

$$\sum_{j \in J(t_0)} \Delta \psi_j = \max_t \sum_{j \in J(t)} \Delta \psi_j .$$

Далее полагаем $\lambda_{t_0} = \lambda_{t_0} + \frac{1}{\Delta}$ и переходим к следующему шагу.

Описанную процедуру предлагается использовать для построения (пересчета) матрицы (λ_{it}) $(i \in I_0', \ t = 1, \ldots, T)$ на всех шагах алгоритма решения задачи (1)-(7). На первом шаге (при проверке всего множества решений задачи) матрица (λ_{it}) $(i \in I, t = 1, \ldots, T)$ строится в результате последовательного пересчета строк некоторой начальной матрицы с одинаковыми элемантами, равными $1/T$. На последующих шагах матрица пересчитывается, если множество $\Pi(x_{i_1}, \ldots, x_{i_q})$ проверяемых решений на данном шаге есть результат ветвления множества проверяемых решений на предыдущем шаге. При этом пересчитывается только i_q-я строка.

3. Приведенная схема алгоритма решения задачи (1)-(7) реализована в виде программы на языке АЛГОЛ. Оценки памяти алгоритма и трудоемкости одного шага алгоритма имеют вид $O(|I| + |J|)$ и $O(|I||J|(|I| + |J|))$. Накопленный к настоящему времени опыт использования алгоритма для решения практических задач и численные эксперименты с ним позволяют дать некоторую характеристику тем параметрам алгоритма, для которых не удается установить нетривиальных априорных оценок. Имеющаяся информация свидетельствует о том, что число шагов алгоритма не превышает величины $10|I|$, а апостериорная оценка точности не выходит за рамки 10% ($\varepsilon \le 0{,}1$). При этом для задач средней размерности $(|I| = 40 \div 50, |J| = 50 \div 100, T = 3 \div 5)$ время счета на ЭВМ БЭСМ-6 составляет в среднем около 30 мин. Наблюдаемое число шагов, необходимое для получения последнего рекордного значения, в среднем составляет менее 20 % от общего числа шагов, а работа процедуры проверки выполнения критерия оптимальности более чем в половине случаев заканчивается положительным исходом.

В целом можно сделать вывод о работоспособности предложенного алгоритма, возможности решать с его помощью задачи реальных размеров за практически приемлемое время.

Поступила в ред.-изд.отдел
4 марта 1984 г.

Литература

1. Береснев В.Л. Алгоритм неявного перебора для задачи типа размещения и стандартизации. - В кн.: Управляемые системы. Новосибирск, 1974, вып. 12, с. 24-34.

2. Береснев В.Л. О задаче выбора оптимальных рядов изделий и комплектующих узлов. В кн.: Управляемые системы. Новосибирск, 1977, вып. 16, с. 35-45.

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

4. Береснев В.Л. Алгоритмы минимизации полиномов от булевых переменных. - В кн.: Проблемы кибернетики. М., 1979, вып. 36, с. 225-246.