$$\min J = \int_a^в F(x,u)\,dt$$

М. И. Ромакин

ОПТИМИЗАЦИЯ ПЛАНИРОВАНИЯ ПРОИЗВОДСТВА

методы
опти-
мальных
решений

$$x = Z \to \max$$

* МЕТОДЫ ОПТИМАЛЬНЫХ РЕШЕНИЙ *

М. И. Ромакин

ОПТИМИЗАЦИЯ ПЛАНИРОВАНИЯ ПРОИЗВОДСТВА

(экономико-математические модели и методы)

Москва
«Финансы и статистика» 1981

ББК 65.10(2)29
Р69

Книга одобрена Комитетом ВСНТО
по прикладным методам математики
и вычислительной технике

Р $\frac{10803-151}{010(01)-81}$ 26—81 (С) 1502000000

ПРЕДИСЛОВИЕ

Проблемы совершенствования планирования, экономического анализа и управления экономикой постоянно находятся в центре внимания партии и правительства. Необходимость совершенствования управления народным хозяйством определяется ростом масштабов производства, усложнением производственных и экономических связей, ускорением темпов научно-технического прогресса. Без этого немыслимо дальнейшее повышение экономической эффективности народного хозяйства. Важным средством решения этих проблем является постановление партии и правительства «Об улучшении планирования и усилении воздействия хозяйственного механизма на повышение эффективности производства и качества работы».

Этим целям служит совершенствование экономической теории и практики на основе применения экономико-математических методов и электронной вычислительной техники. В названном постановлении указывается на необходимость использования в практике нашей работы оптимизации плановых решений.

Выпускаемые в настоящее время электронно-вычислительные машины третьего поколения (ЕС ЭВМ) имеют развивающееся математическое обеспечение. Для решения оптимизационных задач разработаны пакеты прикладных программ. Пакет математического программирования (ПМП) представляет собой комплекс программ для решения задач линейного, параметрического и сепарабельного программирования. Дальнейшим развитием пакета математического программирования является пакет прикладных программ линейного программирования в автоматизированных системах управления (ППП ЛП АСУ), с помощью которого имеется возможность решать также задачи с обобщенными верхними границами и задачи целочисленного программирования. Под управлением этих пакетов работают вспомогательные процессоры — матричный и генератор отчетов, а также их объединение в один пакет, называемый как генератор матриц и отчетов (ГМО); последний используется для генерации матриц и выдачи на печать отчетов по полученным результатам.

Как видим, возможности оптимизационных пакетов на ЕС ЭВМ большие. Однако в практической работе не все они используются в достаточной мере. Мало еще решается целочисленных задач. Еще меньше — нелинейных задач, прежде всего сепарабельных, методы решения которых реализованы в указанных пакетах прикладных программ на ЕС ЭВМ. О задачах с обобщенными верхними границами в отечественной литературе разработок почти нет. Все это определило содержание настоящей книги.

В книге в краткой и доступной форме излагаются простейшие модели линейных, целочисленных и нелинейных задач, дается описание методов их решения, в ряде случаев приводятся иллюстрированные примеры исходя из допустимых объемов книг данной серии. Раздел, касающийся задач с обобщенными верхними границами, написан более подробно, чем все остальные.

Все это поможет широкому кругу пользователей ЕС ЭВМ, и прежде всего постановщикам оптимизационных задач — экономистам и плановым работникам, практическим работникам и студентам кибернетических специальностей, ознакомиться не только с возможностями оптимизационных пакетов на ЕС ЭВМ и использованными в них математическими методами, но и расширить область практического применения методов оптимальных решений. Такая направленность книги, на наш взгляд, будет отвечать задаче о более полном использовании вычислительной техники, указанной XXVI съездом КПСС в «Основных направлениях экономического и социального развития СССР на 1981—1985 годы и на период до 1990 года».

Настоящую книгу можно рассматривать как продолжение и дальнейшее развитие проблем математического программирования, изложенных в нашей книге [10].

Автор искренне благодарит члена-корреспондента Академии педагогических наук СССР, доктора физико-математических наук, профессора И. С. Бровикова и кандидата экономических наук В. Г. Медницкого, чьи замечания и полезные советы помогли улучшить книгу.

1. ЛИНЕЙНЫЕ ЗАДАЧИ

В математическом программировании наиболее теоретически разработанными, проверенными и широко используемыми на практике являются задачи линейного программирования как их модели, так и методы решения.

Основная математическая задача линейного программирования формулируется следующим образом:

найти неотрицательное решение (последовательный набор переменных, вектор) $x_1, x_2, \ldots, x_j, \ldots, x_n$, или

$$x_1 \geqslant 0,\ x_2 \geqslant 0, \ldots, x_j \geqslant 0, \ldots, x_n \geqslant 0 \tag{1.1}$$

системы линейных ограничений (уравнений или неравенств)

$$\left.\begin{array}{l} a_{11}x_1 + a_{12}x_2 + \ldots + a_{1j}x_j + \ldots + a_{1n}\omega b_1; \\ a_{21}x_1 + a_{22}x_2 + \ldots + a_{2j}x_j + \ldots + a_{2n}\omega b_2; \\ \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \\ a_{i1}x_1 + a_{i2}x_2 + \ldots + a_{ij}x_j + \ldots + a_{in}\omega b_i; \\ \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \\ a_{m1}x_1 + a_{m2}x_2 + \ldots + a_{mj}x_j + \ldots + a_{mn}\omega b_n \end{array}\right\} \tag{1.2}$$

(где под ω подразумевается знак $=$, $\leqslant$ или $\geqslant$), максимизирующее (или минимизирующее) линейную (или целевую) функцию

$$c_1x_1 + c_2x_2 + \ldots + c_jx_j + \ldots + c_nx_n = z. \tag{1.3}$$

Решение задачи называется также *планом*.

В сокращенном виде система ограничений (1.1) и (1.2) и функция (1.3) могут быть записаны следующим образом:

$$x_j \geqslant 0,\ j = 1, 2, \ldots, j, \ldots, n; \tag{1.1'}$$

$$\sum_{j=1}^{n} a_{ij}x_j\omega b_i, \quad i = 1, 2, \ldots, m; \tag{1.2'}$$

$$\sum_{j=1}^{n} c_jx_j = z \rightarrow \max(\min). \tag{1.3'}$$

Стандартной формой основной задачи линейного программирования будем называть такой вид задачи (1.1)—(1.3), в которой ограничения (1.2) представлены уравнениями, а целевая функция (1.3) минимизируется. При этом свободные члены b_i должны быть неотрицательными (в противном случае обе части уравнения умножаем на -1). Отметим, что решение задачи симплексным методом, реализованным на ЭВМ, начинается со стандартной формы.

В случае максимизации целевой функции, изменив знаки коэффициентов c_j при переменных x_j, $j=1,2,\dots,n$ на противоположные, мы снова приходим к задаче минимизации.

Любая совокупность переменных x_j, $j=1, 2,\dots,n$, удовлетворяющая условиям (1.1) и (1.2), называется *допустимым решением*. Допустимое решение, минимизирующее целевую функцию, называется *оптимальным решением* (или *оптимальным планом*).

Система (1.2), представленная в стандартной форме, путем элементарных преобразований может быть приведена к виду

$$x_i+\sum_{j=m+1}^{n} a'_{ij}x_j=b'_i,\ i=1,2,\dots,m,\quad j=m+1,m+2,\dots,n,$$

которую будем называть *канонической системой*. В такой системе для каждого i базисная переменная x_i имеет коэффициент единицу в i-м уравнении и нуль в остальных уравнениях.

Задачу линейного программирования часто представляют в векторно-матричном виде.

Обозначим через A — матрицу коэффициентов системы линейных уравнений, через x и b — векторы-столбцы переменных и свободных членов соответственно:

$$A=\begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \cdot & \cdot & \cdot & \cdot \\ & & & \\ a_{m1} & a_{m2} & \cdots & a_{mn} \end{bmatrix},\ x=\begin{bmatrix} x_1 \\ x_2 \\ \cdot \\ \cdot \\ \cdot \\ x_n \end{bmatrix},\ b=\begin{bmatrix} b_1 \\ b_2 \\ \cdot \\ \cdot \\ \cdot \\ b_m \end{bmatrix}$$

Введем также n-мерный вектор $c=(c_1, c_2,\dots,c_n)$, компонентами которого являются коэффициенты целевой

функции. Тогда стандартную форму рассматриваемой задачи можно записать следующим образом:
найти вектор

$$x=(x_1, x_2, \ldots, x_n) \tag{1.1''}$$

при ограничениях

$$x \geqslant 0,$$
$$Ax=b, \tag{1.2''}$$

такой, что целевая функция

$$cx=z \longrightarrow \min. \tag{1.3''}$$

Указанные три формы записи основной задачи линейного программирования позволяют выбирать наиболее эффективную форму представления задачи.

Модели задач линейного программирования кроме обычных условий неотрицательности переменных часто содержат ограничения на переменные сверху или с обеих сторон, т. е. двусторонние ограничения, что записывают следующим образом:

$$x_j \geqslant \beta_j, \quad j=1, 2, \ldots, n,$$

когда все переменные или часть из них ограничены сверху, и

$$\alpha_j \leqslant x_j \leqslant \beta_j$$

в случае двусторонних ограничений на переменные.

Следует отметить, что задача с двусторонними ограничениями $\alpha_j \leqslant x_j \leqslant \beta_j$ заменой переменных $x_j=\alpha_j+u_j$ может быть сведена к задаче только с верхними границами $0 \leqslant u_j \leqslant \delta_j$, где $\delta_j=\beta_j-\alpha_j$ — верхняя граница на новую переменную u_j.

К задаче линейного программирования сводятся многие планово-экономические модели. Приведем простейшие из них.

Модель задачи планирования производства называют также задачей планирования сырья или производственных мощностей.

Предположим, что предприятие выпускает n наименований продукции. Обозначим через a_{ij} затраты i-го вида ресурсов ($i=1, 2, \ldots, m$) на производство единицы продукции j-го вида ($j=1, 2, \ldots, n$), через b_i — объемы имеющихся ресурсов ($i=1, 2, \ldots, m$), через c_j — прибыль, получаемую предприятием при реализации единицы j-го вида продукции.

Требуется составить план производства продукции, который удовлетворял бы заданным ограничениям по ресурсам на выпуск каждого вида продукции с имеющимися технологическими способами производства и давал бы наибольшую прибыль предприятию.

Математическая формулировка задачи состоит в следующем:
найти план производства продукции

$$x=(x_1, x_2, \ldots, x_n)$$

при ограничениях

$$\sum_{j=1}^{n} a_{ij}x_j \leqslant b_i, \quad i=1, 2, \ldots, m;$$

$$x_j \geqslant 0, \quad j=1, 2, \ldots, n$$

и при этом общая прибыль от производства и реализации продукции была бы максимальной, т. е.

$$\sum_{j=1}^{n} c_j x_j = z \longrightarrow \max.$$

Если предприятию наперед заданы нижняя (α_j) и верхняя (β_j) границы по объему выпуска j-й продукции, то в модель задачи войдут ограничения типа

$$\alpha_j \leqslant x_j \leqslant \beta_j$$

Аналогичный вид имеет задача рационального использования производственных мощностей.

Пусть предприятие выпускает n видов изделий, имея m групп оборудования. Известны нормы времени a_{ij} (например, в минутах) на обработку единицы j-го изделия на i-й группе оборудования и фонд времени работы каждой (i-й) группы оборудования. Известно, что из всех n видов изделий $x=(x_1, x_2, \ldots, x_n)$ наибольшим спросом пользуются r видов.

Требуется составить план производства, при котором выпуск дефицитных изделий будет максимальным.

Необходимое время на обработку всех $x_j (j=1, 2, \ldots, n)$ изделий на i-й группе оборудования будет равно сумме $a_{i1}x_1+a_{i2}x_2+\ldots+a_{in}x_n$, которая не может превышать b_i. А для всех m групп оборудования получим систему неравенств n:

$$\sum_{j=1}^{n} a_{ij}x_j \leqslant b_i, \quad i=1, 2, \ldots, m.$$

Компоненты вектора плана $x=(x_1, x_2,\dots$
ют количества изделий и, следовательно,
отрицательными, т. е.

$$x_j \geqslant 0, \quad j=1, 2, \dots, n.$$

Обозначим через $j_1, j_2, \dots, j_r$ номера наиболее дефицитных изделий, выразим общее количество таких изделий через

$$x_{j1}+x_{j2}+\dots+x_{jr}=z \rightarrow \max$$

(т. е. определим план производства дефицитных изделий с наибольшим возможным значением).

Можно составить задачу производственного планирования с учетом дополнительных условий.

Пусть k_j — оптовая цена единицы j-го изделия, а q — заданный объем выпуска изделий в рублях, тогда условие запишем в виде

$$k_1x_1+k_2x_2+\dots+k_nx_n \geqslant q.$$

Если известна потребность в каждом изделии на планируемый период, например, l_j — заявка на j-е изделие, то требование выполнения заявок запишем в виде $x_j \geqslant l_j$, $j=1, 2, \dots, n$. Это будет еще одним дополнительным условием нашей задачи.

Рассмотрим и другие задачи линейного программирования, например, *задачу на составление смесей* (шихты, различных марок бензинов и т. п.). Она называется также задачей о диете, о кормовом рационе.

Известно, что каждый вид корма содержит определенное количество питательных веществ — белков, жиров, углеводов, минеральных солей, витаминов и т. д. Эти величины при планировании кормового рациона известны.

Пусть в хозяйстве имеется $j=1, 2, \dots, n$ видов кормов, каждый из которых содержит $i=1, 2, \dots, m$ разновидностей питательных веществ. Обозначим удельное содержание i-го питательного вещества в корме j-го вида через a_{ij}. Имеются зоотехнические нормы на потребление питательных веществ каждого вида, которые обозначим через b_i. Условие того, что набор кормов $x= (x_1, x_2, \dots, x_n)$, причем

$$x_j \geqslant 0 \text{ для всех } j=1, 2, \dots, n \tag{1.4}$$

удовлетворяет нормам по всем видам питательных веществ, можно записать в виде системы линейных уравнений:

$$\sum_{j=1}^{n} a_{ij}x_j=b_i, \ i=1, 2, \ldots, m. \qquad (1.5)$$

Обозначим стоимость одной единицы j-го корма через c_j. Тогда стоимость всего рациона z через искомые величины x_j выразится следующим образом:

$$\sum_{j=1}^{n} c_jx_j=z. \qquad (1.6)$$

Требуется найти такие значения переменных x_j, которые удовлетворяли бы системе ограничений (1.5) и обращали бы в минимум целевую функцию (1.6).

Таким образом, рассмотренная модель приводит нас к стандартной форме задачи линейного программирования:

$$\sum_{j=1}^{n} a_{ij}x_j=b_i, \ i=1, 2, \ldots, m;$$

$$x_j \geqslant 0, \quad j=1, 2, \ldots, n;$$

$$\sum_{j=1}^{n} c_jx_j=z \to \min.$$

Очевидно, ограничения (1.5) требуют точного выполнения норм по каждому из питательных веществ. В действительности, для некоторых питательных веществ их количество может превзойти норму. Тогда система ограничений (1.5) будет иметь вид:

$$\sum_{j=1}^{n} a_{ij}x_j \geqslant b_i, \ i=1, 2, \ldots, m. \qquad (1.7)$$

Требования разнообразия рациона и ограничений на количество корма каждого вида дают возможность учесть в модели и другие ограничения. Так, если через α_j обозначим обязательное минимальное количество корма вида j в рационе, а через β_j — ограничение на количество корма вида j сверху, то вместо (1.4) запишем

$$\alpha_j \leqslant x_j \leqslant \beta_j, \ j=1, 2, \ldots, n. \qquad (1.8)$$

Возможны и другие дополнения и уточнения, направленные на более точное отображение в модели реальной ситуации.

Простейшей и наиболее известной моделью планирования перевозок является *транспортная задача* планирования перевозок. Рассмотрим эту задачу в матричной постановке по критерию стоимости.

Пусть имеется A_i, $i=1,2,\dots,m$ пунктов производства (поставщиков) некоторого однородного продукта и B_j, $j=1,2,\dots,n$ пунктов его потребления. Для каждого пункта производства A_i известен объем производства a_i, а для каждого пункта потребления B_j — объем потребления b_j. При этом предполагается, что суммарное производство и суммарное потребление сбалансированы, т. е.

$$\sum_{i=1}^{m} a_i = \sum_{j=1}^{n} b_j \tag{1.9}$$

Затраты на перевозку единицы продукта от каждого пункта производства A_i до каждого пункта потребления B_j составляют c_{ij}. Требуется составить план перевозок, равный производительности поставщиков и полностью обеспечивающий потребителей, дающий минимум издержек транспортных затрат на перевозку, т. е.

$$\sum_{i=1}^{m} \sum_{j=1}^{n} c_{ij}x_{ij} = z \longrightarrow \min. \tag{1.10}$$

Задача определяется следующими условиями:

$$\sum_{j=1}^{n} x_{ij} = a_i, \quad i=1,2,\dots,m; \tag{1.11}$$

$$\sum_{i=1}^{m} x_{ij} = b_j, \quad j=1,2,\dots,n; \tag{1.12}$$

$$x_{ij} \geqslant 0, \quad i=1,2,\dots,m, \quad j=1,2,\dots,n, \tag{1.13}$$

где переменные x_{ij} представляют собой объемы перевозок от каждого поставщика A_i к каждому потребителю B_j.

Стоимость c можно рассматривать и как расстояние c_{ij} от пункта A_i до пункта B_j.

Забегая вперед скажем, что при целочисленных исходных данных $a_1, a_2, \dots, a_m$ и $b_1, b_2, \dots, b_n$ транспортная задача всегда обладает целочисленными решениями-планами.

Транспортная задача в матричной постановке по критерию времени совпадает с постановкой транспортной задачи, рассмотренной выше, с той разницей, что вместо матрицы транспортных издержек задается матрица времени доставки t_{ij} груза от каждого поставщика к каждому потребителю.

Мы рассмотрели так называемую классическую транспортную задачу. Ее обобщением является *распределительная задача,* или, как ее еще называют, лямбда (λ)-задача. Ее постановка следующая.

Необходимо составить план поставок x_{ij} неоднородной продукции потребителям при условии полного удовлетворения их спроса с учетом минимальных транспортных издержек.

Заданы множество поставщиков A_i, $i=1,2,\ldots,m$ и объемы поставляемой продукции a_i; множество потребителей B_j, $j=1,2,\ldots,n$ и объемы потребления b_j; затраты на перевозку единицы продукции от поставщиков к потребителям c_{ij}; коэффициенты взаимозаменяемости поставляемой продукции λ_{ij}.

Обозначим через x_{ij} количество продукции, поставляемой i-м поставщиком j-му потребителю. Если между видом продукции i-го поставщика и видом продукции j-го потребителя нет взаимозаменяемости ($\lambda_{ij}=0$), то на перевозку между ними накладывается запрет.

Указанная задача формулируется следующим образом:

найти минимальное значение целевой функции, т. е.

$$\sum_{i=1}^{m}\sum_{j=1}^{n} c_{ij}x_{ij}=z\longrightarrow\min \tag{1.14}$$

при ограничениях

$$\sum_{j=1}^{n} x_{ij}\leqslant a_i,\quad i=1,2,\ldots,m; \tag{1.15}$$

$$\sum_{i=1}^{m} \lambda_{ij}x_{ij}\geqslant b_j,\quad j=1,2,\ldots,n; \tag{1.16}$$

$$x_{ij}\geqslant 0,\quad i=1,2,\ldots,m,\quad j=1,2,\ldots,n. \tag{1.17}$$

Ограничение (1.15) означает, что плановые ограничения поставщиков не нарушаются. Ограничение (1.16) означает, что заявки потребителей должны быть выполнены.

Производственно-транспортная задача нашла широкое применение в практике отраслевого планирования.

Примем следующие обозначения: i — номер поставщика (предприятия); j — номер потребителя; k — номер вида продукции; c_{ij} — расстояние от i-го поставщика до j-го потребителя (при решении задачи по критерию расстояния) или стоимость перевозки единицы продукции (при решении задачи по критерию стоимости); x_{ijk} — количество продукции k-го вида, изготовленное i-м предприятием для j-го потребителя; y_{jk} — потребность j-го потребителя в k-м виде продукции; t_{ik} — удельное время изготовления i-м предприятием единицы k-го вида продукции; T_i — ограничения i-го поставщика по времени в планируемый период; b_i — ограничения i-го поставщика по ресурсам в планируемый период.

Задача состоит в том, чтобы оптимально распределить заказы потребителей на производство многономенклатурной продукции между предприятиями-изготовителями (отраслями), учитывая при этом ограничения предприятий по времени и ресурсам в планируемый период, а также транспортный фактор на доставку продукции потребителям. Задача формулируется следующим образом:

найти минимальное значение целевой функции

$$\sum_{i=1}^{m} \sum_{j=1}^{n} \sum_{k=1}^{L} c_{ij} x_{ijk} = z \longrightarrow \min \qquad (1.18)$$

при ограничениях

$$\sum_{i=1}^{m} x_{ijk} = y_{jk}, \quad j = 1, 2, \ldots, n; \quad k = 1, 2, \ldots, L. \qquad (1.19)$$

$$\sum_{j=1}^{n} \sum_{k=1}^{L} t_{ik} x_{ijk} \leqslant T_i, \quad i = 1, 2, \ldots, m; \qquad (1.20)$$

$$\sum_{j=1}^{n} \sum_{k=1}^{L} x_{ijk} \leqslant b_i, \quad i = 1, 2, \ldots, m; \qquad (1.21)$$

$$x_{ijk} \geqslant 0, \quad i = 1, 2, \ldots, m; \quad j = 1, 2, \ldots, n; \quad k = 1, 2, \ldots, L.$$

Ограничение (1.19) означает, что заявки потребителей должны быть полностью удовлетворены.

Неравенства (1.20) и (1.21) отражают ограничения предприятий в планируемый период по времени и ресурсам.

Любая задача линейного программирования, в каком бы она виде ни была записана, может быть приведена к стандартной и канонической форме и решена симплексным методом, который в определенном смысле является универсальным методом в линейном программировании. Прямой алгоритм (включая алгоритм с искусственными переменными) симплексного метода рассмотрен нами в книгах [10], [11] и др.

На блок-схеме 1 (рис. 1) показан прямой алгоритм симплексного метода.

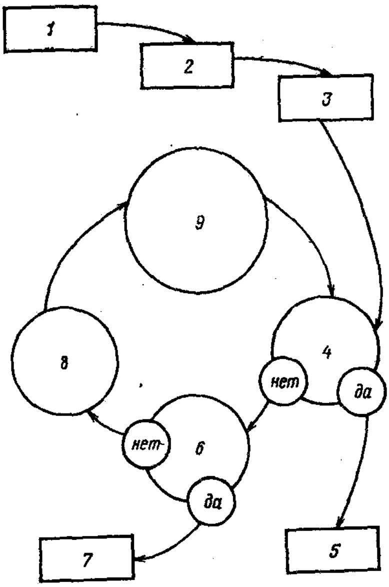

Рис. 1. Блок-схема 1

Блок-схема 1. Прямой алгоритм симплексного метода:

1. Начинаем со стандартной формы.

2. Делаем свободные члены неотрицательными:

3. Преобразуем задачу в каноническую форму относительно x и $(-z)$.

4. Находим ключевой столбец, выбирая k из условия $c_k = \min\{c_j\}$. Устанавливаем, получен ли $\min z$, т. е. правильно ли, что $c_k \geqslant 0$?

| ДА | | НЕТ |

5. **Конец 1.** Получено оптимальное (минимальное) базисное решение. (В случае ДА п. 4.)

6. Устанавливаем, все ли $a_{ik} < 0$? (В случае НЕТ п. 4.)

| ДА | | НЕТ |

7. **Конец 2.** Можно построить множество допустимых решений, для которых целевая функция z неограничена снизу. (В случае ДА п. 6.)

8. Находим ключевую строку, выбирая r из условия $\frac{b_r}{a_{rk}} = \min\{\frac{b_i}{a_{ik}}\}$, $a_{rk} > 0$, $a_{ik} > 0$. (В случае НЕТ п. 6.) (При равенстве отношений берем последнее с наибольшим знаменателем).

9. Преобразуем симплексную таблицу:

а) заменяем r-ю базисную переменную на x_k;

б) делим элементы ключевой строки на a_{rk};

в) заменяем элементы ключевого столбца нулями (кроме $a_{rk}=1$);

г) вычисляем все остальные элементы по формулам:

$$a'_{ij}=a_{ij}-a_{ik}\frac{a_{rj}}{a_{rk}}, \quad b'_i=b_i-a_{ik}\frac{b_r}{b_{rk}}, \quad i\neq r, \quad a_{rk}>0.$$

Если в ключевой строке окажется нуль, то соответствующий ему столбец оставляем в следующей таблице без изменения (то же относится и к ключевому столбцу, если в нем окажется нуль).

На блок-схеме 2 (рис. 2) показан прямой алгоритм с искусственными переменными.

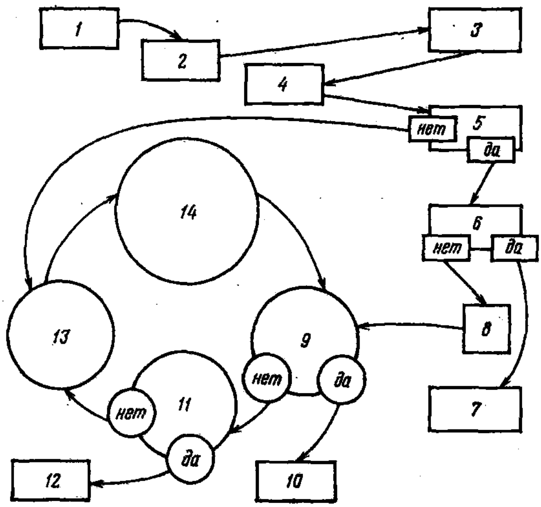

Рис. 2. Блок-схема 2

Блок-схема 2. Алгоритм с искусственными переменными симплексного метода:

1. Начинаем со стандартной формы.

2. Делаем свободные члены неотрицательными.

3. Вводим в уравнения искусственные переменные x_{n+i}. Составляем искусственную форму $u=x_{n+1}+$ $+x_{n+2}+\ldots+x_{n+m}$.

4. Преобразуем задачу в каноническую форму относительно x, $(-z)$ и $(-u)$.

5. Находим ключевой столбец, выбирая k из условия $d_k = \min\{d_j\}$. Устанавливаем, получен ли $\min u$, т. е. правильно ли, что $d_k \geqslant 0$?

ДА НЕТ

6. Устанавливаем, верно ли неравенство $u_0 > 0$? (В случае ДА п. 5.)

7. **Конец 3.** Нет допустимого решения. (В случае ДА п. 6.)

8. Исключаем все такие x_i, что $d_j \geqslant 0$. Исключаем u-строку. (В случае НЕТ п. 6.)

9, 10, 11, 12, 13 и 14 процедуры аналогичны процедурам 4, 5, 6, 7, 8 и 9 (соответственно) в блок-схеме 1.

Процедура 13 выполняется также в случае НЕТ п. 5.

Симплексный метод основан на том, что если в матрице ограничений имеется m строк и они линейно независимы, то существует набор из m столбцов, называемых базисными, которые также линейно независимы. Следовательно, при любой правой части констант можно однозначно найти совокупность значений переменных, соответствующих базисным столбцам. Эта совокупность называется *базисным решением.*

Симплексный метод использует базисные решения, шагая от одного базисного решения к другому посредством замены одного столбца в базисе на один небазисный столбец за каждый шаг (итерацию) до тех пор, пока не будет найдено допустимое решение. После того как найдено хотя бы одно допустимое решение, исследуется ряд этих решений, чтобы найти такое, которое удовлетворяет заданному требованию минимизации целевой функции. Это решение будет *оптимальным.*

Не все задачи имеют оптимальное решение. Если не существует решения с неотрицательными значениями переменных, то найденное решение называется *недопустимым.* Если существует допустимое решение, но целевая функция оказывается неограниченной, то решение называется *неограниченным.*

В симплексном методе на каждой итерации отыскиваются значения m базисных переменных, удовлетворяющие m линейным ограничениям. При этом остальные переменные полагаются равными нулю. Для решения задачи требуется обращение матрицы коэффициентов размером $m \times m$.