ЭКОНОМИКА
И МАТЕМАТИЧЕСКИЕ МЕТОДЫ

1972
том VIII, вып. 2

ПРИМЕНЕНИЕ ИТЕРАТИВНЫХ АЛГОРИТМОВ ПРИ РЕШЕНИИ ЗАДАЧ ЛИНЕЙНОГО ПРОГРАММИРОВАНИЯ

В. А. ВОЛКОНСКИЙ, А. Б. ПОМАНСКИЙ, А. Д. ШАПИРО

(Москва)

I. Среди методов решения задач линейного программирования самым распространенным и разработанным являются симплекс-метод и различные его модификации. Принцип движения «от вершины к вершине» многогранника условий с монотонным приближением к оптимуму представляется наиболее естественно вытекающим из свойств задачи линейного программирования.

Симплекс-метод и его модификации позволяют решать задачи весьма большой размерности, однако при этом часто возникают серьезные затруднения. Поэтому представляет интерес исследование алгоритмов, основанных на других принципах.

Симплекс-метод и его модификации являются алгоритмами монотонными и конечно-шаговыми. Методы, не обладающие этими свойствами, называются «итеративными». Хотя этот термин и не вполне корректен (решение при любом алгоритме совершается в форме некоторого итеративного процесса), будем придерживаться его. К таким алгоритмам относятся различные виды градиентных методов и методов теории игр. В ЦЭМИ АН СССР давно ведется теоретическое и экспериментальное исследование методов, основанных на идее известного метода Брауна — Робинсон решения матричных игр (см., например, [1—4]).

В настоящее время накопился значительный опыт решения задач большой размерности итеративным методом. Можно ожидать, что есть такие практически важные классы задач, для которых эти методы окажутся более эффективными. Такая уверенность основывается на следующих свойствах итеративных алгоритмов, которые в некоторых ситуациях могут оказаться их существенными преимуществами.

1. Ошибки не накапливаются в процессе решения, увеличение размерности задачи не требует увеличения точности вычислений.

2. Траектория процесса приближения к оптимуму не определяется параметрами задачи, как в симплекс-методе (движение от вершины к соседней вершине многогранника). «Шаг итерации» является параметром управления процессом и может в широких пределах варьироваться по воле вычислителя. Число итераций, нужных для достижения необходимой точности, в принципе не должно зависеть от размерности задачи или зависит очень слабо. Этот вывод подтверждается проведенными экспериментами.

3. Для итеративных методов нет проблемы вырожденности и плохой обусловленности задачи, которая иногда затрудняет расчеты по симплексным программам.

4. Простота принципа и для задач планирования прозрачность экономической интерпретации облегчает теоретическое изучение и создание новых модификаций метода для специальных классов задач, в частности,

это позволяет широко использовать различные типы блочной структуры задач для создания декомпозиционных алгоритмов.

Недостатки итеративных методов, связанные с невозможностью получения точного решения за конечное число шагов, оказываются несущественными при большой размерности и низкой точности исходных данных, которые характерны для экономических задач. В этих условиях основное значение приобретает задача нахождения приближенного оптимума при не слишком больших затратах машинного времени. Соединение основной схемы итеративного алгоритма с идеей случайного поиска привело к созданию эффективного алгоритма для приближенного решения задач с непрерывными и целочисленными переменными большой размерности [5, 6].

Настоящая статья посвящена описанию опыта применения итеративных алгоритмов к задаче линейного программирования достаточно общего вида, так называемой многовариантной производственной задаче (общая задача линейного программирования, вообще говоря, может рассматриваться как частный случай этого класса). Проведенные эксперименты доказывают, что в применении к этому классу задач итеративные алгоритмы могут конкурировать по эффективности с симплексными.

II. Рассмотрим пару сопряженных задач линейного программирования.

Задача A: минимизировать

$$A(x) = \sum_{j=1}^{n} \sum_{k=1}^{r_j} c_{kj} x_{kj}$$

при

$$\sum_{j=1}^{n} \sum_{k=1}^{r_j} a_{ij}^{k} x_{jk} \geqslant b_i, \qquad i = 1, \ldots, m, \tag{1}$$

$$\sum_{k=1}^{r_j} x_{kj} = 1, \qquad j = 1, \ldots, n, \tag{2}$$

$$x_{kj} \geqslant 0. \tag{3}$$

Двойственная задача к A

Задача B: максимизировать величину $$B(p, q) = \sum_{i=1}^{m} p_i b_i - \sum_{j=1}^{n} q_j$$

при

$$\sum_{i=1}^{m} p_i a_{ij}^{k} - q_j \leqslant c_{kj}, \tag{4}$$

$$p_i \geqslant 0, \quad i = 1, \ldots, m. \tag{5}$$

Если $p^* = \{p_i^*\}$ и $q^* = \{q_j^*\}$ оптимальное решение задачи B, то справедливы соотношения

$$q_j^* = \max_{1 \leqslant k \leqslant r_j} \left\{ \sum_{i=1}^{m} p_i^* a_{ij}^{k} - c_{jk} \right\}, \qquad j = 1, \ldots, n.$$

Отсюда следует, что задача B может быть переформулирована следующим образом: максимизировать

$$B(p)=\sum_{i=1}^{m}p_ib_i-\sum_{j=1}^{n}\max_{1\leqslant h\leqslant r_j}\left\{\sum_{i=1}^{m}p_ia_{ij}^h-c_{jh}\right\}$$

при условии, что $p_i\geqslant 0$.

Легко проверить, что пара двойственных (сопряженных) задач A и B эквивалентна (см. [1]) игре Г двух лиц с нулевой суммой, в которой множеством стратегий первого (минимизирующего) игрока является множество $X=\sum x_{jh}=1,\ x_{jh}\geqslant 0)$, множеством стратегий второго игрока (максимизирующего) — множество $P=\{p_i\geqslant 0\}$, а функция выигрыша (второго игрока) есть функция Лагранжа задачи A

$$\tilde{\varphi}(x,p)=\sum_{i=1}^{m}p_ib_i+\sum_{j=1}^{n}\sum_{k=1}^{r_j}c_{jk}x_{jk}-\sum_{i=1}^{m}\sum_{j=1}^{n}\sum_{k=1}^{r_j}p_ia_{ij}^kx_{jk}.$$

Этой игре можно дать наглядную экономическую интерпретацию. Условимся, что первый игрок — совокупность предприятий, выбирающих план производства продукции $x=\{x_{jk}\}$, второй — планирующий орган, отвечающий за сбыт продукции, стратегиями которого являются цены на продукцию и сырье. Первый выбирает план x, максимизирующий его прибыль $\sum\limits_{jk}(\sum\limits_{i}p_ia_{ij}^kx_{jk}-c_{jk}x_{jk})$, второй — цены p с целью максимизировать свою прибыль от реализации продукции $\sum\limits_{i}(p_ib_i-\sum\limits_{jk}p_ia_{ij}^kx_{jk})$. Такая интерпретация согласуется с задачей линейного программирования и с ее игровой формализацией. На каждую стратегию p второго игрока существует «оптимальный ответ» $\hat{x}(p)$ первого игрока. Для нахождения этого ответа первый игрок должен минимизировать свою платежную функцию $\tilde{\varphi}(x,p)$. Из формулы, определяющей функцию $\tilde{\varphi}(x,p)$, следует, что оптимальный ответ $\hat{x}(p)$ имеет структуру

$$\hat{x}_{jk}(p)=\begin{cases}1, \text{ если } -c_{jk}+\sum\limits_{i=1}^{m}p_ia_{ij}^k=S_j(p),\\[2ex] 0, \text{ если } -c_{jk}+\sum\limits_{i=1}^{m}p_ia_{ij}^k<S_j(p),\end{cases}$$

$$\sum_{k=1}^{r_j}\hat{x}_{jk}(p)=1,$$

где

$$S_j(p)=\max_{1\leqslant k\leqslant j}\left\{\sum_{i=1}^{m}p_ia_{ij}^k-c_{jk}\right\}.$$

Если максимум выражения $\sum\limits_{i=1}^{m}p_ia_{ij}^k-c_{jk}$ достигается при нескольких значениях k, то оптимальный ответ определяется неоднозначно.

Величину $\sum_{i=1}^{m} p_i a_{ij}^k - c_{jk}$ можно интерпретировать как прибыль j-го предприятия при реализации k-го варианта, а $S_j(p)$ — как максимальную прибыль. Таким образом, «оптимальный ответ» $\hat{x}(p)$ на стратегию второго игрока есть план, реализующий варианты с максимальной прибылью при ценах p. На стратегии $x = \{x_{jk}\}$ первого игрока, не удовлетворяющей условиям

$$\sum_{j=1}^{n} \sum_{k=1}^{r_j} a_{ij}^k x_{jk} \geqslant b_i,$$

существуют оптимальные ответы второго игрока, приводящие к сколь угодно большому значению функции $\tilde{\varphi}(x, p)$. Поэтому множество стратегий второго игрока должно быть сужено неравенствами $p_i \leqslant M_i$, где M_i любое число, превосходящее оптимальное значение p_i^*.

Описываемый ниже алгоритм α основан на применении идеи метода Брауна к игре Г.

III. Перейдем к описанию алгоритма L для решения задачи A.

Задаемся начальными значениями переменных прямой и двойственной задач: $x^0 = \{x^0_{jk}\}$ и $p^0 = \{p_i^0\}$, которые удовлетворяют соотношениям

$$\sum_{k=1}^{r_j} x_{jk}^0 = 1, \qquad \text{для всех } j = 1, \ldots, n, \tag{6}$$

$$x^0_{jk} \geqslant 0, \tag{7}$$

$$0 < p_i^0 \leqslant M_i. \tag{8}$$

После проведения предписанных алгоритмом действий получим значения переменных на первой итерации $x' = \{x'_{jk}\}$ и $p' = \{p_i'\}$, затем на второй и т. д. Таким образом, получим последовательность векторов $x^t = \{x_{jk}^t\}$ и $p^t = \{p_i^t\}$, которую назовем траекторией с начальными значениями x^0 и p^0 и обозначим через $T\{x^0, p^0\}$.

Опишем теперь совокупность действий, производимых на $(t+1)$-й итерации.

1) Нахождение «оптимального ответа». Для каждого j-го блока переменных определяем номер $k = k_j$, для которого

$$S_{jk}^t = \sum_{i=1}^{m} p_i^t a_{ij}^k - c_{jk}$$

максимально: $S_j^t = \max S_{jk}^t$. Если максимум S_j^t достигается для нескольких k, то выбираем произвольно одно из них. Затем полагаем, что

$$\hat{x}_{jk} = \begin{cases} 1, & \text{если } k = k_j, \\ 0, & \text{если } k \neq k_j. \end{cases}$$

2) Формирование нового «текущего плана» x^{t+1}. Для каждых j и k полагаем

$$x_{jk}^{t+1} = (1 - \alpha_t) x_{jk}^t + \alpha_t \hat{x}_{jk}^t, \tag{9}$$

где α_t больше нуля, но меньше 1.

3) Вычисление «невязок» для планов x^{t+1} и $\hat{x}^t$. Для каждого i вычисляем «невязки»

$$\Delta_i^{t+1} = \sum_{j=1}^{n} \sum_{k=1}^{r_j} a_{ij}^k x_{jk}^{t+1} - b_i, \tag{10}$$

$$\hat{\Delta}_i^t = \sum_{j=1}^{n} \sum_{k=1}^{r_j} a_{ij}^k \hat{x}_{jk}^t - b_i.$$

4) Формирование нового вектора двойственных переменных p^{t+1}. Для каждого i полагаем

$$p_i^{t+1} = (1 - h_t) p_i^t + h_t \hat{p}_i^t, \tag{11}$$

где

$$\hat{p}_i^t = \begin{cases} M_i, \text{ если } \Delta_i^{t+1} < 0, \\ 0, \text{ если } \Delta_i^{t+1} \geqslant 0. \end{cases}$$

Для сходимости такого процесса к точке равновесия (совпадающей с решением исходной задачи) последовательности «демпфирующих параметров» $\{\alpha_t\}$ и $\{h_t\}$ должны удовлетворять условиям

$$\alpha_t, h_t \to 0, \quad \sum_{t=1}^{\infty} \alpha_t = \sum_{t=1}^{\infty} h_t = \infty.$$

В настоящее время сходимость строго доказана для случая $\alpha_t = h_t$ (см. [7]). Однако можно ожидать, что соответствующая теорема остается справедливой, когда последовательности $\{\alpha_t\}$ и $\{h_t\}$ различны, но удовлетворяют условию, что для любой последовательности номеров итераций $\{t_k\}$ ряды $\sum_{k=1}^{\infty} \alpha_{t_k}$ и $\sum_{k=1}^{\infty} h_{t_k}$ либо сходятся, либо расходятся одновременно. Применение алгоритма к игре, описанной в п. II, допускает следующую условную экономическую интерпретацию. Каждое предприятие отрасли выбирает свой оптимальный вариант развития, руководствуясь критерием максимума прибыли в ценах на продукцию, сырье и материалы, установленных планирующим органом. Затем предприятия корректируют свои планы в направлении увеличения прибыли и передают свои предложения в планирующий орган, который корректирует цены, повышая их на дефицитные виды продукции и сырье и понижая на продукты (сырье), предложение которых превышает потребности (лимиты).

В процессе разработки практических методов решения задач линейного программирования оказалось целесообразным видоизменить алгоритм [3, 4]. Так как обычно априорно не известны более или менее точные оценки сверху P_i для двойственных переменных p_i, то соотношение (11) было заменено следующим

$$p_i^{(t+1)} = (1 + \theta_i^t h_t) p_i^t, \tag{12}$$

где

$$\theta_i^t = \begin{cases} 1, \text{ если } \Delta_i \;\; < 0, \\ -1, \text{ если } \Delta_t^{t+1} \geqslant 0. \end{cases} \tag{13}$$

Большой эффект в смысле ускорения сходимости дало предложенное А. Н. Пителиным использование такого способа изменения, который учитывает не только сам факт выполнения или невыполнения ограничений, но и тенденцию к изменениям невязок. Вместо (13) используется соотношение

$$\theta_i^t = \begin{cases} 1, \text{ если } \Delta_i^{t+1} < 0 \text{ и } \hat{\Delta}_i^t < 0, \\ -1, \text{ если } \Delta_i^{t+1} \geqslant 0 \text{ и } \hat{\Delta}_i^t \geqslant 0, \\ 0 \text{ — в остальных случаях,} \end{cases} \tag{14}$$

где $\hat{\Delta}_i^t$ — невязки, построенные по «оптимальному ответу» $\hat{x}(p^t)$. В дальнейшем алгоритм, использующий такие формулы усреднения p_i, будем называть алгоритмом L.

Как уже отмечалось, необходимыми условиями сходимости алгоритма L является стремление демпфирующих параметров α_t и h_t к нулю и расходимость рядов $\sum_{t=1}^{\infty} \alpha_t$ и $\sum_{t=1}^{\infty} h_t$. Это можно пояснить следующим образом. Если α_t и h_t не стремятся к нулю, то процесс может не установиться, т. е. может не существовать предела. А если ряды $\sum_{t=1}^{\infty} \alpha_t$ и $\sum_{t=1}^{\infty} h_t$ сходятся, то последовательность $\{(x^t, p^t)\}$ сойдется «раньше времени», не дойдя до точки равновесия. Проведенные эксперименты показывают зависимость скорости сходимости от характера изменения демпфирующих параметров $\{\alpha_t\}$ и $\{h_t\}$. Возможно, не существует единого пригодного для всех задач способа изменений демпфирующих параметров.

В расчетах по алгоритму L был использован следующий способ изменения параметров $\{\alpha_t\}$ и $\{h_t\}$. Сначала полагали $\alpha_1 = {}^1/_2$, и до итерации с номером d_1 α_t сохраняло это значение. Затем полагали $\alpha_{d_1} = {}^1/_4$ и не меняли его до итерации с номером $2d_1$, на которой полагали $\alpha_{2d_1} = {}^1/_8$. Следующие изменения происходили на итерациях с номерами $4d_1$, $8d_1$, $16d_1$ и т. д. На всех этих итерациях α_t уменьшалось вдвое. Последовательность h_t строилась по такому же правилу, только деления происходили на итерациях d_2, $2d_2$, $4d_2$, $8d_2$ и т. д. Ясно, что таким образом построенные последовательности демпфирующих параметров удовлетворяют необходимым условиям, которые обсуждались выше.

IV. Разберем подробнее механизм действия описываемого итеративного процесса. Для всех «текущих планов» $x^t = \{x_{jk}^t\}$ выполняются специальные ограничения, т. е.

$$\sum_{k=1}^{r_j} x_{jk}^t = 1.$$

Это следует из того, что начальный план x^0 и все «оптимальные ответы» $\hat{x}^t$ удовлетворяют этим условиям.

Из формул, определяющих изменение двойственных переменных p_i, следует, что если на t-й итерации не выполняются для какого-нибудь i условия

$$\sum_{j=1}^{n} \sum_{k=1}^{r_j} a_{ij}^k x_{jk}^{t+1} \geqslant b_i \quad \text{и} \quad \sum_{j=1}^{n} \sum_{k=1}^{r_j} a_{ij}^k \hat{x}_{jk}^{t+1} \geqslant b_i,$$

то p_i^{t+1} больше, чем p_i^t. Если эти условия не выполняются на протяжении еще некоторого числа итераций, то p_i (в силу расходимости ряда $\sum\limits_{t=1}^{\infty} h_t$) может стать сколь угодно большим. Но тогда в «оптимальном ответе» значение компоненты, равной единице, будет соответствовать вариантам с большим значением элементов a_{ij}^k (при данном i). Более точно это можно объяснить следующим образом. Пусть $p_{i_0}^t$ очень велико по сравнению с остальными компонентами вектора двойственных переменных p^t. Тогда максимум выражения

$$\sum_{i=1}^{m} p_i^t a_{ij}^k - c_{jk}$$

будет достигаться при тех значениях k_j, при которых

$$a_{ij}^{k_j} = \max_{1 \leqslant k \leqslant r_j} a_{ij}^k.$$

Но для такого «оптимального ответа» обязательно выполняется условие

$$\sum_{j=1}^{n} \sum_{k=1}^{r_j} a_{ij}^k \hat{x}_{jk}^t > b_i,$$

если, конечно, задача имеет допустимое решение. Из формулы, определяющей итеративный процесс, следует, что левые части ограничений удовлетворяют соотношениям

$$\sum_{j=1} \sum_{k=1}^{r_j} a_{ij}^k x_{jk}^{t+1} = (1 - \alpha_t) \sum_{j=1}^{n} \sum_{k=1}^{r_j} a_{ij}^k x_{jk}^t + d_t \sum_{j=1}^{n} \sum_{k=1}^{r_j} a_{ij}^k \hat{x}_{jk}^t.$$

Так как $\sum\limits_{t=1}^{\infty} \alpha_t = \infty$ и если для «оптимальных ответов» на протяжении достаточно большого числа итераций выполняется i-е ограничение, то наступит момент, когда это ограничение выполняется и для «текущего плана» $\{x_{jk}^t\}$. Если же какое-нибудь i-е ограничение все время выполняется для «текущих планов» x^t и «оптимальных ответов» $\hat{x}^t$, то значение оценки p_i^t, соответствующей этому ограничению, будет уменьшаться и может быть сколь угодно мало (опять же в силу расходимости ряда $\sum\limits_{t=}^{\infty} h_t$). А это означает, что при формировании «оптимальных ответов» это ограничение не будет учитываться и если оно вообще в задаче существенно, то в конце концов оно не будет выполняться для «оптимальных ответов». Проиллюстрируем вышесказанное на простейшем случае однопродуктовой модели.

Предположим, что требуется минимизировать $\sum\limits_{j=1}^{n} c_j x_j$ при условиях

$$\sum_{j=1} a_j x_j \geqslant b, \tag{15}$$

$$0 \leqslant x_j \leqslant 1. \tag{16}$$

Обозначим через $\bar{p}$ минимальное значение управляющего параметра p, при котором для «оптимального ответа» выполняется ограничение (15), т. е.

$$\sum_{j=1}^{n} a_j \hat{x}_j(p) \geqslant b.$$

Можно показать, что такое $\bar{p}$ существует и равно оптимальному значению p для соответствующей двойственной задачи.

Допустим, что на некоторой итерации t_0 выполнены соотношения

$$\sum_{j=1}^{n} a_j x_j^{t_0} < b \quad \text{и} \quad p^{t_0} < \bar{p}.$$

Так как на t_0-й итерации не выполняются ограничения для «текущего» плана и «оптимального ответа», то p^{t_0+1} будет больше, чем p^{t_0}. И вообще, p^t будет возрастать до тех пор, пока либо $\sum_{j=}^{n} a_j x_j^t$ станет больше, чем b, либо p^t станет не меньше $\bar{p}$, но пока $p^t < \bar{p}$, $\sum_{j=1}^{\infty} a_j x_j^t$ не может стать больше b. Пусть t_1 — номер итерации, на которой впервые $p^{t_1} \geqslant \bar{p}$. С этого момента p_t не изменяется до тех пор, пока $\sum_{j=1}^{n} a_j x_j < b$. Пусть t_2 — номер итерации, на которой впервые $\sum_{j=1}^{n} a_j x_j \geqslant b$. Для $t > t_2$ p^t будет уменьшаться и для некоторого t_3 p^t станет меньше $\bar{p}$. С этого момента $\sum_{j=1}^{n} a_j x_j^t$ начнет уменьшаться, так как $\sum_{j=1}^{n} a_j \hat{x}_j(p^t) < b$. На итерации с номером t_4 $\sum_{j=1}^{n} a_j x_j^t$ станет меньше b и будет продолжать уменьшаться до момента t_5. На этом заканчивается полный «период» колебаний.

На этом примере хорошо видно преимущество использованного в алгоритме L способа изменения управляющих параметров по сравнению со способом, задаваемым формулами (11) и (13). Действительно, если не учитывать «оптимальный ответ» при изменении p^t, то в промежутке от t_1 до t_2 p^t будет возрастать (соответственно в промежутке от t_3 и t_4 уменьшаться), что приводит к увеличению «амплитуды» и «длины волны» колебаний.

На рис. 1 и 2 показаны различия двух способов изменения управляющего параметра p, изображены условные графики $\sum_{j=1}^{n} a_j x_j^t$ и p^t как функции от t. Рисунок 1 соответствует (13), а рис. 2 — (14).

V. Приведенное описание позволяет сформулировать ряд частных утверждений о поведении траекторий $T(x^0, p^0)$.

Утверждение 1. Если задача A совместна и существует план $\{z^{i_0}\}$ такой, что $\Delta_i(z^{i_0}) > 0$, то для любой траектории $T(x^0, p^0)$ найдется номер итерации t_{i_0}, на которой $\Delta_i(x_{t\,i}^{\,0}) > 0$. Другими словами, для любой траектории либо, начиная с некоторого t_0, $\Delta_i^t > 0$ для всех $t > t_0$, либо Δ_i^t бесконечное число раз меняет знак, т. е. колеблется вокруг нуля.

Утверждение 2. Если траектория $T(x^0, p^0)$ имеет предел $(\tilde{x}, \tilde{p})$, то $\tilde{x}$ и $\tilde{p}$ оптимальные решения задач A и B. Это показывает «корректность» алгоритма, т. е. если существует сходимость, то к оптимуму.

Утверждение 3. Для задачи с одним ограничением ($m = 1$) каждая траектория $T(x^0, p^0)$ имеет предел. Доказательство этого утверждения является продолжением тех качественных соображений, которые приводились в п. IV.

Эти утверждения являются вполне достаточными теоретическими обоснованиями для применения описываемого алгоритма L. Однако необходимо отметить, что этот алгоритм совсем не всегда способен дать хорошие результаты. Чтобы он оказался эффективным, задача должна иметь достаточное число «степеней свободы»: число ограничений общего вида (1) должно быть меньше числа переменных $\{x\}$. Смысл этого условия может быть, например, таков: число различных потребностей хозяйства, обращенных к данному объекту, значительно меньше, чем число способов их удовлетворения, так что остается существенная свобода выбора.

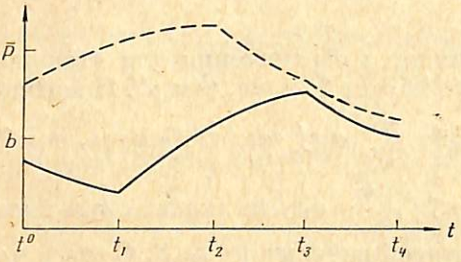

Рис. 1

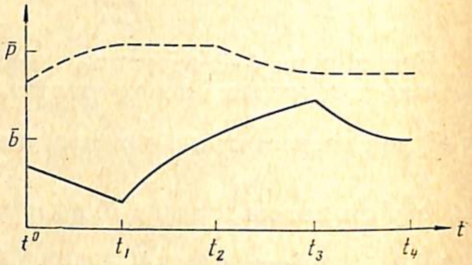

Рис. 2

Часто при построении моделей и постановке задач вводятся несущественные условия, которые резко сужают область допустимых решений, что обычно затрудняет использование алгоритма L.

VI. Вывод о колебании невязок Δ_i вокруг нулевых значений, сделанный в п. V, приводит к следующему специальному приему, резко повышающему скорость сходимости. Естественно предположить, что указанные процессы колебания невязок Δ_i для разных ограничений (i) слабо зависят друг от друга. Поэтому ситуации, при которых выполнены одновременно все ограничения (1), происходят очень редко. Амплитуды этих колебаний (с ростом номера итерации) должны уменьшаться. Поэтому, учитывая неточность исходных данных, можно было бы считать приемлемыми те планы x^t, для которых невязки Δ_i^t удовлетворяют не неравенствам $\Delta_i^t \geqslant 0$, а неравенствам

$$\Delta_i^t \geqslant -\delta_i, \tag{17}$$

где δ_i — достаточно малое положительное число.

Если при этом сам алгоритм и процесс не изменяются, т. е. ослабление ограничений учитывается только при выборе момента окончания вычислений, а не при формировании величин p_i, то планы, допустимые с точки зрения ограничений (9), будут встречаться гораздо чаще, чем допускаемые с точки зрения исходной задачи. Тот же прием можно использовать и для решения исходной задачи (с неослабленными ограничениями). Для этого, наоборот, надо внести изменение в алгоритм (и в траекторию процесса) и не менять определение допустимых планов. А именно, при определении величин θ_i^t по формуле (14) вместо невязок Δ_i^t и $\hat{\Delta}_i^t$, вычисленных по формулам (10), надо использовать величины на δ_i меньше: $\Delta_i^t - \delta_i$ и $\hat{\Delta}_i^t - \delta_i$. При этом согласно рассуждениям, приведенным в п.п. III, IV, невязки Δ_i^t будут колебаться вокруг значений δ_i, а так как $\delta_i > 0$, то будут чаще появляться допустимые планы (для которых $\Delta_i^t \geqslant 0$ одновременно для всех i). Как было сказано выше, эксперименты подтверждают эффективность этого приема.

VII. Описанные выше алгоритмы L и его модификации использовались для решения целого ряда практических задач отраслевого планирования. Эти задачи были разными по числу производственных ограничений (от 3 до 174), по числу блоков (от 5 до 68) и по общему числу переменных (от 77 до 711). Этими же алгоритмами были решены несколько задач с плохой обусловленностью.

Алгоритмы реализованы на машине БЭСМ-6. Запрограммированы они для решения задачи на минимум, поэтому в тех задачах, где требуется максимизировать функционал, необходимо было изменить знаки его коэффициентов на противоположные. Независимо от размерности задач решения, близкие к оптимуму, получались за относительно небольшое число итераций (300—600). Выдача результата производилась в момент, когда разность критериев прямой и двойственной задач становилась меньше заданного E. Число E колебалось в пределах 0,01—1% от величины критерия.

ЛИТЕРАТУРА

1. В. А. Волконский. Оптимальное планирование в условиях большой размерности (итеративные методы и принцип декомпозиции). Экономика и матем. методы, 1965, т. I, вып. 2.
2. В. Г. Медницкий. О методе оптимального распределения плановых заданий в отрасли. Экономика и матем. методы, 1965, т. I, вып. 6.
3. В. А. Волконский, В. И. Данилов-Данильян, В. Г. Медницкий. Модели и методы оптимального планирования. М., 1966 (ЦЭМИ АН СССР).
4. А. К. Пителин. Построение итеративных алгоритмов для решения задач оптимального планирования. В сб. Материалы к Всесоюзной конференции по применению экономико-математических методов в отраслевом планировании и управлении. Секция № 1. М., 1966 (ЦЭМИ АН СССР).
5. А. Б. Поманский, А. Д. Шапиро. Итеративные методы для решения отраслевых задач, содержащих дискретные переменные. Докл. АН СССР, 1969, т. 186, № 1.
6. Оптимальный план отрасли. М., «Экономика», 1970.
7. С. А. Иванков. Итеративный метод решения игр. В сб. Труды Третьей зимней школы по математическому программированию. 23 января — 5 февраля 1970 г., г. Дрогобыч. М., 1970 (Научный совет по комплексной проблеме «Оптимальное планирование и управление народным хозяйством»).

Поступила в редакцию
9.IV.1970