УДК 519.852
DOI: 10.26102/2310-6018/2022.38.3.022

Метод проектирования и приращений при решении задач линейного программирования

А.В. Ганичева[1], А.В. Ганичев[2]✉

[1]Тверская государственная сельскохозяйственная академия,
Тверь, Российская Федерация
[2]Тверской государственный технический университет,
Тверь, Российская Федерация
alexej.ganichev@yandex.ru✉

Резюме. В настоящее время проблема выбора оптимального решения является одной из самых важных и актуальных проблем в промышленности, экономике, сельском хозяйстве и военной сфере. Для решения многих прикладных задач оптимизации применяются методы и подходы теории линейного программирования. Основной метод линейного программирования – симплекс-метод – характеризуется большим объемом вычислительных действий и процедур. Поэтому для решения данной проблемы используются модификации основного метода, обладающие более высокой алгоритмической эффективностью. В данной статье разработан новый метод решения задач линейного программирования. Меньшая, чем у симплекс-метода, алгоритмическая сложность обеспечивается рассмотрением класса задач с полностью ограниченными областями допустимых решений. Новый метод обоснован результатами, анонсированными в доказанных утверждениях. Реализация метода описана двумя алгоритмами: 1) поиск квазиоптимального решения с помощью анализа координат проекций на гиперплоскости (алгоритм проектирования); 2) поиск оптимального решения путем задания приращений ограничениям (алгоритм приращений). Для пояснения работы алгоритмов рассмотрены конкретные числовые примеры. Оценка алгоритмической сложности разработанного метода осуществляется подсчетом количества использованных арифметических операций. Получены формульные выражения для расчета сложности вычислений.

Ключевые слова: алгоритм, переменная, гиперплоскость, проекция, неравенство, итерация, число операций, вычислительная сложность.

Для цитирования: Ганичева А.В., Ганичев А.В. Метод проектирования и приращений решения задач линейного программирования. *Моделирование, оптимизация и информационные технологии.* 2022;10(3). Доступно по: https://moitvivt.ru/ru/journal/pdf?id=1223 DOI: 10.26102/2310-6018/2022.38.3.022

The method of design and increments in solving linear programming problems

A.V. Ganicheva[1], A.V. Ganichev[2]✉

[1]Tver State Agricultural Academy, Tver, Russian Federation
[2]Tver State Technical University, Tver, Russian Federation
alexej.ganichev@yandex.ru✉

Abstract. Currently, the issue of choosing the optimal solution is one of the most important and urgent in industry, economy, agriculture, and the military sector. Methods and approaches of linear programming theory are used to solve many applied optimization tasks. The simplex method, which is the principal method of linear programming, is characterized by a large amount of computational actions and procedures. Owing to this, modifications of the main method with higher algorithmic efficiency are employed to address this problem. In this article, a new method for solving linear programming problems has been developed. The algorithmic complexity, which is less than that of the simplex method, is provided by considering a class of problems with completely limited areas of acceptable solutions. The new method is justified by the results announced in the proven statements. The implementation of the method is described by two algorithms: 1) search for a quasi-optimal solution by analyzing the coordinates of projections on hyper planes (design algorithm); 2) search for an optimal solution by setting increments to constraints (increment algorithm). To explain the functioning of the algorithms, specific numerical examples are analyzed. Algorithmic complexity estimates of the developed method are carried out by counting the number of arithmetic operations undertaken. Formula expressions for estimating the complexity of calculations are obtained.

Keywords: algorithm, variable, hyperplane, projection, inequality, iteration, number of operations, computational complexity.

For citation: Ganicheva A.V., Ganichev A.V. The method of design and increments in solving linear programming problems. *Modeling, Optimization and Information Technology*. 2022;10(3). Available from: https://moitvivt.ru/ru/journal/pdf?id=1223 DOI: 10.26102/2310-6018/2022.38.3.022 (In Russ).

Введение

Методы линейного программирования (ЛП) решения прикладных задач широко используются во многих областях знания [1]. Они применяются в факторном анализе [2, 3], моделях обучения [4], управлении портфелями проектов [5], для выбора управленческого решения [6] и других областях. Широкое распространение задач данного класса делает проблему эффективности методов алгоритмов ЛП важной и актуальной проблемой. Можно выделить 2 класса задач ЛП: с неограниченными [7, 8] и полностью ограниченными областями допустимых решений [9]. Важной проблемой является разработка методов ЛП, позволяющих уменьшить вычислительную сложность алгоритмов поиска оптимальных решений по сравнению с симплекс методом и его модификациями [10].

Целью данной работы является разработка нового эффективного метода решения задач ЛП.

Для достижения цели решаются следующие задачи:

1) ищется проекция из заданной точки на область допустимых решений, наиболее близкая к оптимальному решению;

2) задавая приращения ограничениям, определяется оптимальное решение.

Метод проектирования

Общая задача линейного программирования (ОЗЛП) заключается в нахождении максимального (минимального) значения функции

$$L = \sum_{i=1}^{n} c_i \cdot x_i + c_0 \tag{1}$$

при ограничениях

$$\sum_{i=1}^{S_k} d_{ki} x_i = d_k \,, \tag{2}$$

где $k = \overline{1, l}$, $x_i \geq 0$ при $i = 1, \ldots, S_k$.

При этом будем считать, что все переменные из (2) содержатся среди переменных (1).

Рассмотрим максимизацию функции L, т. к. минимизация этой функции сводится к максимизации $-L$.

Пусть $\max_{k=\overline{1,l}} S_k = S$ и $l \le S$. Будем рассматривать задачу при условии неособенности матрицы размерностью $l \times l$ для первых l коэффициентов переменных.

Теорема 1. Путем равносильных элементарных преобразований систему уравнений (2) можно привести к виду:

$$\begin{cases} x_1 = b_{1,l+1}x_{l+1} + ... + b_{1,l+s}x_{l+s} + b_1, \\ x_2 = b_{2,l+1}x_{l+1} + ... + b_{2,l+s}x_{l+s} + b_2, \\ \dots\dots\dots\dots\dots\dots\dots\dots\dots \\ x_l = b_{l,l+1}x_{l+1} + ... + b_{l,l+s}x_{l+s} + b_l. \end{cases} \quad (3)$$

Доказательство очевидно.

При переходе от (2) к (3) некоторые столбцы могут переставляться местами. Однако, во избежание громоздкости и, не нарушая общности, будем считать, что в (1) переменная x_i $(i = \overline{1,S})$ входит с коэффициентом c_i.

Если $l > S$, то система (2) приводится к виду (3) при условии, что все равенства при $l > S$ непротиворечивы. В противном случае система несовместна.

Не нарушая общности, будем рассматривать первый случай и систему (3).

Заметим, что поскольку реальные ситуации в производственной и социальной сфере предполагают ограниченность значений переменных, то система ограничений для x_k $(k = \overline{1,l})$, x_i $(i = \overline{l+1,S})$ запишется в виде:

$$\begin{cases} \gamma_i \le x_i \le \delta_i \ (i = \overline{l+1,S}), \\ x_k = \sum_{i=l+1}^{S} b_{ki}x_i + b_k \ \ k = \overline{1,l}, \\ v_k = x_k \le w_k \ \ k = \overline{1,l}. \end{cases} \quad (4)$$

Третья строка (4) должна быть согласована с первыми двумя. В общем случае будем рассматривать и отрицательные значения переменных, т. е. в случае необходимости можно сделать перенос начала координат. Кроме того, если в (3) для некоторого $k = \overline{1,l}$ все коэффициенты $b_{k,l+i}$ отрицательны, то вводим новую переменную $y_k = -x_k$.

Пусть для $i = \overline{l+1,S}$

$$\begin{cases} \beta_i = \gamma_i, \ \text{если} \ c_i + \sum_{k=1}^{l} c_k b_{ki} < 0, \\ \beta_i = \delta_i \ \text{ в противном случае.} \end{cases} \quad (5)$$

Тогда, согласно теореме 3 из [9], условный максимум функции L при выполнении условий (4) и (5) будет равен:

$$L_{\max} = \sum_{i=l+1}^{S} c_i\beta_i + c_0 + \sum_{k=1}^{l} c_k \cdot (b_k + \sum_{i=l+1}^{S} b_{ki}\beta_i). \quad (6)$$

Данная формула представляет случай, когда переменные в (1) и (2) совпадают.

Если в (1) имеются переменные, не входящие в (2), то в $L_{\max}$ эти переменные принимают значения своей верхней границы, если соответствующий коэффициент положителен, и значение нижней границы в противном случае.

Положим $x_k^* = \sum_{i=l+1}^{S} b_{ki}\beta_i + b_k \ (k=\overline{1,l})$, где β_i определяется из (5). Определим точку $M^*(X^*, B)$, где $X^* = (x_1^*, ..., x_l^*)$, $B = (\beta_{l+1}, ..., \beta_s)$. (7)

Будем называть это решение приближенно максимальным, соответствующее значение L – приближенно максимальным значением.

Если $x_k^* \ (k=\overline{1,l})$ удовлетворяет условиям (4), то точка $M^*(X^*, B)$, как показано в [9], будет точкой максимума функции L (1). Однако для этих β_i может не выполняться для некоторых k третье условие (4). Рассмотрим этот случай.

Поиск максимального решения заключается в поиске точек $M_k \ (k=\overline{1,l})$, близких к M^*, проекции которых на гиперплоскости

$$x_k = \sum_{i=l+1}^{S} b_{ki}x_i + b_k \ (k=\overline{1,l}) \tag{8}$$

удовлетворяют неравенствам (4). Это будет приближенный условный максимум. В дальнейшем для плоскости (8) будем использовать обозначение $A_k \ (k=\overline{1,l})$.

Фиксируем $k \in \overline{1,l}$; на первом этапе k=1. Не нарушая общности, будем использовать обозначение «k». Как будет показано далее (на примере), для некоторых из этих проекций условия (4) не будут выполняться. Эти проекции исключаются из дальнейшего рассмотрения. Если ни одна из проекций не будет удовлетворять системе (4) , как будет показано, система (4) будет несовместной. Если для некоторых $k \ (k=\overline{1,l})$ проекция будет удовлетворять системе (4), то эта проекция будет приближенным решением, которое будет в дальнейшем улучшаться с использованием метода приращений, что будет показано дальше.

Для обозначения проекции точки M будем использовать обозначение ПрM. Пусть точка $M_k = (Y, B)$, где $Y = (y_1, ..., y_l)$ и $B = (\beta_{l+1}, ..., \beta_s)$ $y_p = x_p^* + \varepsilon_p \ (p=\overline{1,l})$, ε_p будут определены позднее. Найдем точку ПрM_k, которая будет основанием перпендикуляра, опущенного из точки $M_k(Y, B)$ на гиперповерхность (A_k).

Запишем уравнение данного перпендикуляра:

$$\frac{x_k - y_k}{-1} = \frac{x_{l+1} - \beta_{l+1}}{b_{k,l+1}} = ... = \frac{x_s - \beta_s}{b_{ks}}. \tag{9}$$

Из (9) находим:

при $i = \overline{l+1, S}$, $x_{l+1} = \beta_{l+1} + b_{k,l+i} \cdot (y_k - x_k)$. (10)

Подставив (10) в (8), получим: $x_k = b_k + \sum_{i=l+1}^{S} b_{ki}\beta_i + \sum_{i=l+1}^{S} b_{ki}^2 (y_k - x_k)$. Отсюда

$$x_k = \left(x_k^* + y_k \cdot \sum_{i=l+1}^{S} b_{ki}^2 \right) \Bigg/ \left(1 + \sum_{i=l+1}^{S} b_{ki}^2 \right), \tag{11}$$

$$x_k = x_k^* + \varepsilon_k \cdot \sum_{i=l+1}^{S} b_{ki}^2 \Bigg/ \left(1 + \sum_{i=l+1}^{S} b_{ki}^2 \right). \tag{12}$$

Подставим (11) в (9), получим:

$$x_{l+i} = \beta_{l+i} + b_{k,l+i}\varepsilon_k \Big/ \left(1 + \sum_{i=l+1}^{S} b_{ki}^2\right). \quad (13)$$

Из (8) и (13) получаем:

$$x_p = \sum_{i=l+1}^{S} b_{pi}x_i + b_p = \sum_{i=l+1}^{S} b_{pi}(\beta_i + b_{ki}\varepsilon_k \Big/ \left(1 + \sum_{i=l+1}^{S} b_{ki}^2\right) + b_p =$$

$$= x_p^* + \varepsilon_k \sum_{i=l+1}^{S} b_{pi}\, b_{ki} \Big/ \left(1 + \sum_{i=l+1}^{S} b_{ki}^2\right), \quad p = \overline{1,l}, \ p \neq k. \quad (14)$$

С другой стороны, поскольку в (A_k) координаты x_p ($p \neq k$, $p = \overline{1,l}$) отсутствуют, то из (9) имеем:

$$\frac{x_k - y_k}{-1} = \frac{x_p - x_p^* - \varepsilon_p}{0}, \text{ т. е. } x_p = x_p^* - \varepsilon_p.$$

Следовательно, $\varepsilon_p = \varepsilon_k \sum_{i=l+1}^{S} b_{pi}\, b_{ki} \Big/ \left(1 + \sum_{i=l+1}^{S} b_{ki}^2\right)$.

Потребуем выполнения условий (4):

$$\begin{cases} v_k \le x_k = x_k^* + \varepsilon_k \sum_{i=l+1}^{S} b_{ki}^2 \Big/ \left(1 + \sum_{i=l+1}^{S} b_{ki}^2\right) \le w_k, \ k = \overline{1,l}, \\ \gamma_{l+i} \le x_{l+i} = \beta_{l+i} + b_{k,l+1}\varepsilon_k \Big/ \left(1 + \sum_{i=l+1}^{S} b_{ki}^2\right) \le \delta_{l+i}, \ i = \overline{1,S}, \\ v_p \le x_p = x_p^* + \varepsilon_k \sum_{i=l+1}^{S} b_{pi}\, b_{ki} \Big/ \left(1 + \sum_{i=l+1}^{S} b_{ki}^2\right) \le w_p, \ p \neq k, \ p = \overline{1,l}. \end{cases}$$

Из первого неравенства имеем:

$$(v_k - x_k^*)\left(1 + \sum_{i=l+1}^{S} b_{ki}^2\right) \Big/ \left(\sum_{i=l+1}^{S} b_{ki}^2\right) \le \varepsilon_k \le (w_k - x_k^*)\left(1 + \sum_{i=l+1}^{S} b_{ki}^2\right) \Big/ \left(\sum_{i=l+1}^{S} b_{ki}^2\right). \quad (15)$$

Заметим, что чем меньше ε_k, тем ближе будут значения y_k и x_k^*. Минимальное значение ε_k равно левой части (15). И тогда $x_k = 0$. Аналогично определяются ограничения для x_j при $i = \overline{l+1,S}$. Здесь различаются два случая.

Первый случай. Пусть $b_{k,l+j} > 0$, тогда

$$\frac{\gamma_{l+i} - \beta_{l+i}}{b_{k,l+i}}\left(1 + \sum_{i=l+1}^{S} b_{ki}^2\right) \le \varepsilon_k \le \frac{\delta_{l+i} - \beta_{l+i}}{b_{k,l+i}}\left(1 + \sum_{i=l+1}^{S} b_{ki}^2\right). \quad (16)$$

Второй случай. Пусть $b_{k,l+j} < 0$, тогда

$$\frac{\delta_{l+i} - \beta_{l+i}}{b_{k,l+i}}\left(1 + \sum_{i=l+1}^{S} b_{ki}^2\right) \le \varepsilon_k \le \frac{\gamma_{l+i} - \beta_{l+i}}{b_{k,l+i}}\left(1 + \sum_{i=l+1}^{S} b_{ki}^2\right). \quad (17)$$

Из (14) для x_p при $\sum_{i=l+1}^{S} b_{pi}b_{ki} > 0$ имеем:

$$(w_p - x_p^*)\left(1+\sum_{i=l+1}^{S} b_{ki}^2\right)\Bigg/\left(\sum_{i=l+1}^{S} b_{pi}b_{ki}\right) \geq \varepsilon_k \geq (v_p - x_p^*)\left(1+\sum_{i=l+1}^{S} b_{ki}^2\right)\Bigg/\left(\sum_{i=l+1}^{S} b_{pi}b_{ki}\right), \qquad (18)$$

здесь $p = \overline{1,l}$, $p \neq k$.

Если $\sum_{i=l+1}^{S} b_{pi}b_{ki} < 0$, то имеем противоположное неравенство:

$$(w_p - x_p^*)\left(1+\sum_{i=l+1}^{S} b_{ki}^2\right)\Bigg/\left(\sum_{i=l+1}^{S} b_{pi}b_{ki}\right) \leq \varepsilon_k \leq (v_p - x_p^*)\left(1+\sum_{i=l+1}^{S} b_{ki}^2\right)\Bigg/\left(\sum_{i=l+1}^{S} b_{pi}b_{ki}\right). \qquad (19)$$

В дальнейшем, если неравенство имеет вид: $\alpha \leq \beta$, то α будем называть левой частью (ЛЧ) неравенства, а β – правой частью (ПЧ) неравенства. При этом в скобках после символов ЛЧ и ПЧ будем указывать номер неравенства, например, ЛЧ (18) - левая часть неравенства (18). Положим,

$$\varepsilon_{k1} = \max\{ЛЧ(15), \max_{i:b_{k,l+i}>0} ЛЧ(16), \max_{i:b_{k,l+i}<0} ЛЧ(17), \max_{p\neq k:\ \sum_{i=l+1}^{S} b_{pi}b_{ki}>0} ЛЧ(18), \max_{p\neq k:\ \sum_{i=l+1}^{S} b_{pi}b_{ki}<0} ЛЧ(19)\}; \qquad (20)$$

$$\varepsilon_{k2} = \min\{ПЧ(15), \min_{i:b_{k,l+i}>0} ПЧ(16), \min_{i:b_{k,,l+i}<0} ПЧ(17), \min_{p\neq k:\ \sum_{i=l+1}^{S} b_{pi}b_{ki}>0} ПЧ(18), \min_{p\neq k:\ \sum_{i=l+1}^{S} b_{pi}b_{ki}<0} ПЧ(19)\}. \qquad (21)$$

Отсюда следует, что для выполнения системы (14) необходимым и достаточным условием является условие:

$$\varepsilon_{k1} \leq \varepsilon_k \leq \varepsilon_{k2}. \qquad (22)$$

Будем называть это условие основным. Заметим, что если максимальный элемент из (20) – это *ЛЧ*(15), а минимальный элемент из (21) – это $\min_{p\neq k:\ \sum_{i=l+1}^{S} b_{pi}b_{ki}<0} ПЧ(19)$, то (22) всегда выполняется в случае, если $x_k^* > 0$ и $x_p^* > 0$.

При остальных значениях ε_{k1} и ε_{k2} (22) может выполняться, а может и не выполняться. Если (22) выполняется, то минимальное значение ε_k, при котором точка M_k максимально приближена к точке $M_k(X^*, A)$, находится из равенства $\varepsilon_k = \varepsilon_{k1}$, при этом координаты точки M_k определяются следующим образом: x_k находится из формулы (12), x_p $(p \neq k)$ – из формулы (14), x_{p+i} $(i = \overline{1,S})$ – из формулы (13).

Однако, точка M_k дает только приближенное значение условного максимума функции L.

Если условие (22) не выполнено, то присваиваем k значение k+1, и весь процесс повторяется. Если (22) не будет выполнено ни при каком k, то это означает несовместимость условий (4).

Практическая реализация метода проектирования

Рассмотрим пример.

Требуется найти максимальное значение функции $L = -x_1 + 2x_2 + 3x_3 + x_4$ при ограничениях:

$$x_1 = 3x_3 - x_4 - 0,1;$$
$$x_2 = -3x_3 + x_4 + 0,2;$$
$$0,3 \le x_3 \le 2;\ 0 \le x_4 \le 1;\ x_1 \ge 0;\ x_2 \ge 0,$$

т. е. $\delta_3 = 2$; $\delta_4 = 1$; $\gamma_3 = 0,3$; $\gamma_4 = 0$.

С учетом ограничений на x_3 и x_4 получаем, что $-0,2 \le x_1 \le 5,5$; $-5,8 \le x_2 \le 0,6$. Тогда ограничения для x_1 и x_2 будут: $0 \le x_1 \le 5,9$; $0 \le x_2 \le 0,6$. Таким образом, $v_1 = 0$; $w_1 = 5,9$; $v_2 = 0$; $w_2 = 0,6$.

Шаги Алгоритма 1:

1. Определяем β_i из (5).

Имеем: $i = 3$, $c_3 + \sum_{k=1}^{2} c_k b_{k3} = 3 + (-1)\cdot 3 + 2\cdot(-2) < 0$, поэтому $\beta_3 = \gamma_3 = 0,3$;

$i = 4$, $c_4 + \sum_{k=1}^{2} c_k b_{k4} = 1 + (-1)\cdot(-1) + 2\cdot 1 > 0$, значит, $\beta_4 = \delta_4 = 1$.

2. Вычисляем x_1^* и x_2^*:

$$x_1^* = \sum_{i=1}^{S} b_{1i}\beta_i + b_1 = 3\cdot 0,3 - 1\cdot 1 - 0,1 = -0,2;\ x_2^* = \sum_{i=1}^{S} b_{2i}\beta_i + b_2 = -2\cdot 0,3 + 1 + 0,2 = 0,6.$$

Таким образом, $M^*(X^*, A) = (-0,2;\ 0,6;\ 0,3;\ 1)$. $L_{\max} = 2,9$ без учета условий (4).

3. Находим точки M_k (k=1,2). Пусть k=1. Определяем ε_{11} и ε_{12}.

4. Находим $ЛЧ\ (15) = -x_k^* \cdot \left(1 + \sum_{i=l+1}^{S} b_{1i}^2\right) \Big/ \left(\sum_{i=l+1}^{S} b_{1i}^2\right) = 0,2\cdot(1+9)/(9+1) = 0,22$.

5. Находим $\max\limits_{i:b_{1,l+i}>0} ЛЧ\ (16) = \max\limits_{i:b_{1,l+i}>0} \frac{\gamma_{l+i} - \beta_{l+i}}{b_{1,l+i}}\left(1 + \sum_{i=l+1}^{S} b_{1i}^2\right) = \frac{0,3-0,3}{3}\cdot 11 = 0$.

6. Определяем $\max\limits_{i:b_{k,l+i}<0} ЛЧ\ (17) = \max\limits_{i:b_{k,l+i}<0} \frac{\delta_{l+i} - \beta_{l+i}}{b_{k,l+i}}\left(1 + \sum_{i=l+1}^{S} b_{1i}^2\right) = \frac{1-1}{-1}\cdot 11 = 0$.

7. Находим $\max\limits_{p=2} ЛЧ\ (19) = 0,6 - 0,6\cdot 11/10 = -0,06$.

8. Поскольку $\sum_{i=l+1}^{S} b_{2i} b_{1i} < 0$, то ЛЧ (18) не существует.

Вывод: $\varepsilon_{11} = \max\{0,22;\ 0;\ -0,06\} = 0,22$.

9. Находим: $\min\limits_{i:b_{1,l+i}>0} ПЧ\ (16) = \min\limits_{i:b_{1,l+i}>0} \frac{\delta_{l+i} - \beta_{l+i}}{b_{1,l+i}}\left(1 + \sum_{i=l+1}^{S} b_{1i}^2\right) = \frac{2-0,3}{3}\cdot 11 = 6,23$.

10. Имеем: $\min\limits_{i:b_{1,l+i}<0} ПЧ\ (17) = \min\limits_{i:b_{1,l+i}<0} \frac{\gamma_{l+i} - \beta_{l+i}}{b_{1,l+i}}\left(1 + \sum_{i=l+1}^{S} b_{1i}^2\right) = \frac{0-1}{-1}\cdot 11 = 11$.

11. Находим:

$$\min_{p\ne 1:\ \sum_{i=l+1}^{S} b_{pi}b_{1i}<0} ПЧ(19) = \min_{p\ne 1:\ \sum_{i=l+1}^{S} b_{pi}b_{1i}<0} -x_p^*\left(1 + \sum_{i=l+1}^{S} b_{1i}^2\right) \Big/ \left(\sum_{i=l+1}^{S} b_{pi}b_{1i}\right) = -0,6\cdot 11/(-7) = 0,943.$$

Вывод: $\varepsilon_{12} = \min\{6,233;\ 11;\ 0,943\} = 0,943$.

12. Сравниваем ε_{11} и ε_{12}: $\varepsilon_{11} = 0,22 < \varepsilon_{12} = 0,943$. Значит, $\varepsilon_{11} = 0,22$, $M_1(0,02;0,6;0,3;0,1)$.

13. Находим:

$$x_1 = x_1^* + \varepsilon_{11}\sum_{i=l+1}^{S} b_{1i} / (1+\sum_{i=l+1}^{S} b_{1i}^2) = -0,2 + 0,22 \cdot 10/11 = 0;$$

$$x_2 = x_2^* + \varepsilon_{11}\sum_{i=l+1}^{S} b_{2i}b_{1i} / (1+\sum_{i=l+1}^{S} b_{1i}^2)_1 = 0,6 + 0,22\cdot(-7)\cdot 10/11 = 0,36;$$

$$x_3 = \beta_{l+1} + b_{1,l+1}\cdot\varepsilon_{11} / (1+\sum_{i=l+1}^{S} b_{1i}^2) = 0,3 + 3\cdot 0,22/11 = 0,36;$$

$$x_4 = \beta_{l+2} + b_{1,l+2}\cdot\varepsilon_{11} / (1+\sum_{i=l+1}^{S} b_{1i}^2) = 1 - 1\cdot 0,22/11 = 0,98.$$

Итог: $\textit{Пр}M_1(0;0,46;0,36;0,98)$.

Вывод: приближенная точка максимума функции L, удовлетворяющая ограничениям (4), это точка $\text{Пр}M_1$.

Найдем $L(\text{Пр}M_1) = 2\cdot 0,466 + 3\cdot 0,36 + 0,98 = 2,98$.

Метод приращений

Возникает вопрос: как улучшить найденное значение $L(\text{Пр}M_1)$ и найти точную точку максимума. Для этого рассматривается итерационный процесс – назовем его Алгоритм 2, суть которого заключается в следующем. Известны интервальные оценки для координат найденной точки $\textit{Пр}M_1$. На каждой итерации координаты деформируются за счет соответствующих приращений с последующей проверкой значения L для этих приращений. Если L будет неположительна, то, значит, полученная на предыдущем шаге приближенная точка максимума будет являться оптимальным решением. В противном случае новая точка, полученная из приближенной точки прибавлением к ее координатам соответствующих приращений, будет точкой максимума, если ее координаты будут удовлетворять заданным ограничениям. В противном случае ищется проекция этой точки, и работает Алгоритм 1.

Итак, рассмотрим сущность Алгоритма 2 сначала для первой итерации, а затем в общем виде. В разделе практической реализации будет приведен соответствующий пример.

Алгоритм 2.

1. Дадим каждой переменной x_{l+i} $(i = \overline{1,S})$ приращение $\Delta^1 x_{l+i}$.

2. Тогда каждая переменная x_p $(i = \overline{1,l})$ получит приращение $\Delta^1 x_p = \sum_{i=p+1}^{S} b_{pi}\Delta^1 x_i + b_p$.

Замечание. Здесь нижний индекс «k» связан с перебором всех плоскостей (A_k) при $k = \overline{1,l}$. Для определенности сначала положим k=1, знак Δ означает, что рассматриваются приращения; верхний индекс «1» обозначает номер итерации, т. к. Алгоритм 2 итерационный процесс добавления к каждой переменной x_i последовательности приращений $\Delta^1 x_i$, затем $\Delta^2 x_i$, $\Delta^3 x_i$ и т. д. Знак «*» обозначает значения координат без проектирования.

3. При этом

$$\gamma_{l+i} - \tilde{x}_{l+i} = \Delta^1 x_{l+i} \le \delta_{l+i} - \tilde{x}_{l+i},$$
$$v_p - \tilde{x}_p = \Delta^1 x_p \le w_p - \tilde{x}_p,$$

$i = \overline{1,S}$, $p = \overline{1,l}$, $\tilde{x}_{l+i}$, $\tilde{x}_p$ _ значение соответственно переменной x_{l+i} и x_p точки ПрM_k.

4. Положим: $\gamma^1_{l+i} = \gamma_{l+i} - \tilde{x}_{l+i}$; $\delta^1_{l+i} = \delta_{l+i} - \tilde{x}_{l+i}$; $v^1_p = v_p - \tilde{x}_p$; $w^1_p = w_p - \tilde{x}_p$.

5. Определим по формулам (5)

$\beta^1_{l+i} = \gamma^1_{l+i}$, если $c_{l+i} + \sum_{k=1}^{l} c_k \cdot b_{k,l+i} < 0$, и $\beta^1_{l+i} = \delta^1_{l+i}$, если $c_{l+i} + \sum_{k=1}^{l} c_k \cdot b_{k,l+i} \ge 0$.

6. Находим $x_i = \Delta^1 x_i = \beta^1_i$ $(i = \overline{l+1,S})$, $x^*_p = \Delta^1 x^*_p = \sum_{i=l+1}^{S} b_{pi}\beta_i + b_p$ $(p = \overline{1,l})$, т. е. точку $M^*_{k,\Delta}$.

7. Если выполняется условие: $v^1_p \le x^*_p \le w^1_p$ $(p = \overline{1,l})$, то проверяется условие $L(M^{*1}_{k,\Delta}) > 0.$

8. Если это условие выполняется, то переход на начало Алгоритма 2.

9. Если $L(M^{*1}_{k,\Delta}) \le 0$, то найденная ранее точка ПрM_k будет точкой максимума.

10. Если условие $v^1_p \le x^*_p \le w^1_p$ не выполняется для некоторого p $(p = \overline{1,l})$, то происходит проектирование точки $M^{*1}_{k,\Delta}$, которая получается из точки $M^{*1}_{k,\Delta}$ прибавлением к каждой координате x^*_p числа ε_p, т. е. повторяется Алгоритм 1 для $x_p = \Delta^1 x_i$, которые находятся по формулам (12)-(14) для $x_i = \Delta^1 x_i = \beta^1_i$ и нового $\varepsilon_{k1} = \Delta\varepsilon^1_{k1}$.

11. При этом проверяется основное условие. Если оно не выполняется, то полагаем k=k+1, и весь процесс (Алгоритм 1 и 2) повторяется с самого начала для гиперплоскости (A_{k+1}).

12. Если основное условие выполняется, вычисляем значение L в точке ПрM^1_1, Δ.

13. Если $L \le 0$, то найденная ранее точка M^1_1 будет являться точкой максимума.

14. Если L>0, то точка M^1_1 будет приближенным максимальным решением, которое может быть улучшено за счет уменьшения соответствующего приращения первой координаты.

15. Это итерационный процесс, для которого

$$\gamma^1_{l+i} - \sum_{j=1}^{q} \tilde{x}^j_{l+i} = \Delta^{q+1} x_{l+i} \le \delta^1_{l+i} - \sum_{j=1}^{q} \tilde{x}^j_{l+i}; \tag{23}$$

$$v^1_p - \sum_{j=1}^{q} \tilde{x}^j_p \le \Delta^{q+1} x_p \le w^1_p - \sum_{j=1}^{q} \tilde{x}^j_p; \tag{24}$$

здесь $\tilde{x}^j_{l+i}$ $(\tilde{x}^j_p)$ – значение переменной Δx^j_{l+i} (Δx^j_p) точки $M^{*j}_{k,\Delta}$ или, если не выполняется (4), Пр$M^j_{k,\Delta}$, $i = 1,\ldots,S$, $p = 1,\ldots,l$.

Данный процесс сходится, если правая часть (23) или (24) – убывающая относительно q величина для некоторого i или p. Будем рассматривать именно этот случай. При этом

$$\tilde{x}_{l+k}^{j} = \tilde{x}_{k}^{j-1} + b_{k,l+i} \cdot \Delta\varepsilon_{k}^{j-1} \Bigg/ \left(1 + \sum_{i=l+1}^{S} b_{ki}^{2}\right). \quad (25)$$

Аналогично определяется $\tilde{x}_{p}^{j}$.

16. Если на некоторой итерации с номером q $ПЧ(24) \le 0$, $(ПЧ(23))$, то $\text{Пр}M_{k,\Delta}^{q}$ будет точкой максимума.

17. Если $ПЧ(24) > 0$, то по формулам (5) определяем $\beta_{l+i}^{q} = \delta_{l+i}^{q}$.

18. Находим значения $\Delta^{q} x_{l+i} = \beta_{l+i}^{q}$ $(i = \overline{1,S})$, $\Delta^{q} x_{p}^{*} = \sum_{i=l+1}^{S} b_{pi}\beta_{i}^{j} + b_{p}$ $(p = \overline{1,l})$. Получаем точку $M_{k,\Delta}^{*q}$.

19. Находим $L(M_{k,\Delta}^{*q}) = L_{k,\Delta}^{q}$.

20. Если $L \le 0$, то $\text{Пр}M_{k,\Delta}^{q}$ будет точкой максимума. Если $L > 0$, то при $\Delta^{q+1} x_{p}^{*}$ $(p = \overline{1,l})$, удовлетворяющем (24), точка $M_{k}^{q} = (\tilde{x}_{1} + \sum_{j=1}^{q} \Delta^{j} x_{1},\ \tilde{x}_{2} + \sum_{j=1}^{q} \Delta^{j} x_{2}, ..., \tilde{x}_{n} + \sum_{j=1}^{q} \Delta^{j} x_{n})$ - подозрительная на экстремум, и повторяется Алгоритм 2.

21. Если $L > 0$, но условие (24) выполняется не для всех p, то повторяется Алгоритм 1 для $x_{p} = \Delta^{q} x_{p}^{*}$, $x_{l+i} = \Delta^{q} x_{l+i}^{*} = \beta_{l+i}^{q}$ и нового $\varepsilon_{k1} = \Delta^{q} \varepsilon_{k1}$.

22. Если для некоторого j не будет выполняться основное неравенство, то полагаем k=k+1 (т. е. переходим к другой плоскости) и весь процесс повторяется, начиная с шага 3 Алгоритма 1.

Практическая реализация метода приращений

Вернемся к рассмотренному ранее примеру. Попробуем приблизить найденное решение к истинному.

Согласно п. 3 Алгоритма 2 имеем:

$0,3 - 0,36 = -0,06 \le \Delta^{1} x_{3} \le 2 - 0,36 = 1,64$; $0 - 0,98 = -0,98 \le \Delta^{1} x_{4} \le 1 - 0,98 = 0,02$; $0 \le \Delta^{1} x_{1} \le 5,9$; $-0,46 \le \Delta^{1} x_{2} \le 0,14$.

4. Положим $\gamma_{3}^{1} = -0,06$; $\delta_{3}^{1} = 1,64$; $\gamma_{4}^{1} = -0,98$; $\delta_{4}^{1} = 0,02$.

5. Определим $\beta_{3}^{1} = \gamma_{3}^{1} = -0,6$; $\beta_{4}^{1} = \gamma_{4}^{1} = 0,02$.

6. Находим $\Delta x_{1}^{*} = -3 \cdot 0,06 - 0,02 - 0,1 = -0,3$; $\Delta x_{2}^{*} = 2 \cdot 0,06 + 0,02 + 0,2 = 0,34$.

Точка $M_{k,\Delta}^{*q} = (\Delta x_{1}^{*}, \Delta x_{2}^{*} \beta_{3}^{1}, \beta_{4}^{1})$.

7–8. Условия $\Delta^{1} x_{1}^{*} \ge 0$ и $\Delta^{1} x_{2}^{*} \le 0,14$ не выполнены. Работает Алгоритм 1.

11. Ищем $\text{Пр}M_{1,\Delta}^{1}$ – проекцию точки $M_{1,\Delta}^{1}$ на плоскость (A_{1}).

4–7. Находим

$$\varepsilon_{11}^1 = \max\{0,3\cdot 11/10;(-0,06-1,64\cdot 11)/3;(0,02-0,02)\cdot 11/(-1);(0,14-0,34)\cdot 11/(-7)\}=0,33.$$

8-10. Находим $\varepsilon_{12}^1 = \min\{(5,9+0,3)\cdot 11/10;(1,64+0,06)\cdot 11/3;$

$(-0,98-0,02)\cdot 11/(-1);(0,46-0,34)\cdot 11/(-7)\}=1,25.$

11. Имеем: $\varepsilon_{11}^1 = 0,33 < 1,25 = \varepsilon_{12}^1$. Значит, $\varepsilon_{11}^1 = 0,33$.

12-15. Находим: $\Delta^1 x_1 = -0,3+0,3-0,33\cdot 10/11=0;$

$\Delta^1 x_2 = 0,34-0,33\cdot 7/11=0,13;$ $\Delta^1 x_3 = -0,06+0,33\cdot 13/11=0,03;$

$\Delta^1 x_4 = 0,02-0,33/11=-0,01;$ $\text{Пр}M_{1,\Delta}^1 = (0;\ 0,13;\ 0,03;\ -0,01)$.

Вывод: точка

$M_1^1 = (0;\ 0,46+0,1; 0,36+0,03;\ 0,98-0,01) = (0; 0,59; 0,39; 0,97)$ – приближенное решение рассмотренной задачи ЛП, при котором $L(M_1^1)=3,32$.

23. При успешном нахождении максимального значения функции *L* с использованием проектирования только на одну плоскость (A_1), *k* присваивается значение *k*+1 и весь процесс отыскания максимума осуществляется по отношению к следующей гиперплоскости.

24. После перебора всех плоскостей и нахождения соответствующих максимальных значений функции *L* из них выбирается максимум, который и представляет собой максимальное значение *L*.

Для рассмотренного примера при *k*=2 имеем: $\varepsilon_{21}=0$; $\varepsilon_{22}=-1$. Таким образом, не выполнено основное условие (22). Поэтому максимальное значение *L* определяется только первой гиперплоскостью.

Покажем, что найденное приближенное решение можем превратить в оптимальное посредством еще двух итераций (*q*=2, *q*=3), результаты которых представлены следующим образом (Таблица 1).

Таблица 1 – Результаты двух итераций
Table 1 – Results of two iterations

q	γ_3^2	δ_3^2	γ_4^2	δ_4^2	β_3^2	β_4^2	v_1^2	w_1^2	v_2^2	w_2^2
2	-0,09	1,61	-0,97	0,03	-0,09	0,03	0	5,9	-0,59	0,01
	$\Delta^2 x_1^*$	$\Delta^2 x_2^*$	ε_{11}^2	ε_{12}^2	$\Delta^2 x_1$	$\Delta^2 x_2$	$\Delta^2 x_3$	$\Delta^2 x_4$	$L_{1\Delta}^2$	
	-0,4	0,41	0,6285	1,57	0,1714	0,0101	0,8141	-0,271	0,066	
	$M_1^2 = (0,1714;\ 0,6;\ 0,4714;\ 0,943)$									
3	γ_3^3	δ_3^3	γ_4^3	δ_4^3	β_3^3	β_4^3	v_1^3	w_1^3	v_2^3	w_2^3
	-0,1714	1,5286	-0,9429	0,0571	-0,1714	0,0571	-0,1714	-0,6	5,729	0
	$\Delta^3 x_1^*$	$\Delta^3 x_2^*$	ε_{11}^3	ε_{12}^3	$\Delta^3 x_1$	$\Delta^3 x_2$	$\Delta^3 x_3$	$\Delta^3 x_4$	$L_{1\Delta}^3$	
	-0,557	0,5999	0,9427	1,8856	0,2999	0	0,0857	-0,03	-0,071	

Из этой таблицы видно, что при q=2 $w_1^2 > 0$ и $L_{1\Delta}^2 > 0$. Поэтому q присваивается значение $q = q + 1 = 3$, и весь процесс повторяется. При q=3 имеем: $w_1^3 < 0$ и $L_{1\Delta}^3 < 0$. Это означает, что дальнейшего увеличения функции L не может быть и точка максимума – это точка $M_1^2 = (0{,}1714;\ 0{,}6;\ 0{,}4714;\ 0{,}9429)$. При этом $L_{\max} = L(M_1^2) = L(M_1^1) + L_{1\Delta}^2 = 3{,}32 + 0{,}0658 = 3{,}3858$.

Таким образом, имеет место следующее утверждение.

Теорема 2.

Пусть функция L представлена формулой (1) при ограничениях (4). Если условие (4) выполнено для точки $M^*(X^*, B) = (x_1^*, ..., x_l^*, \beta_{l+1}, ..., \beta_{l+S})$, то это точка максимума, причем β_i $(i = \overline{1,S})$ определяются из (5). В противном случае максимальное значение $L_{\max} = \max\limits_{\Omega}(L(ПрM_k(Y,B)) + \sum\limits_{j=1}^{q} L_{k,\Delta}^j)$, где множество Ω представляет собой множество тех значений $k = \overline{1,l}$, для которых условие (22) выполняется на каждом шаге проектирования на соответствующую гиперплоскость (A_k), $ПрM_k(Y,B)$ - проекция точки $M_k(Y,B) = (x_1^* + \varepsilon_1, ..., x_k^* + \varepsilon_k, ..., x_l^* + \varepsilon_l, \beta_{l+1}, ..., \beta_{l+S})$ на гиперплоскость A_k $(k = \overline{1,l})$, $L_{k,\Delta}^j = L(\tilde{x}_1^j, ..., \tilde{x}_n^j)$ значения функции L, где $\tilde{x}_l^j = \Delta^j x_k^*$, если выполняется условие (22), в противном случае $\tilde{x}_l^j$ – это $\Delta^j x_i$ – координаты проекции $Пр(\Delta^j x_j^*, ..., \Delta^j x_n^*)$ на гиперплоскость (A_k), если для нее выполняется условие (22). Если это условие не выполняется, то данное k исключается из рассмотрения.

Количество итераций q связано с выполнением одного из условий: либо $ПЧ(24) \le 0$; $(ПЧ(23) = 0)$, либо $L_{k,\Delta}^q \le 0$.

Оценка сложности метода проектирования и приращений

Одна из важнейших задач линейного программирования заключается в оценке числа операций сложения, умножения, деления, сравнения и присваивания. Это число характеризует сложность алгоритмов (методов). Рассмотрим решение этой задачи согласно разработанному в статье методу с иллюстрацией на конкретном примере для описанных алгоритмов 1 и 2.

1. i=l+1, число операций умножения равно S, сложений – S+1; сравнения – 1, присваивания -1, т. е. всего – 2S+3 операций.

При i, изменяющемуся от l+1 до S, общее число операций составит (2S+3)S. Для рассматриваемого примера это число будет равно 14.

2. При вычислении x_k^* используется S операции умножения, S-1 – сложения, т. е. всего -2S. Тогда при $k = \overline{1,l}$ будет (2S-1)l операций.

3. 4. Для $ЛЧ$(15) – умножения – S+1, сложения – S, деления – 1. Итого: 2S+2. Для рассматриваемого примера – 6.

5. 2S операций умножения и сложения, которые использовались на шаге 3, 2, новые операции – вычитания 1, умножения – 1, деления – 1, т. е. 3 операции, и таких операций для i: $b_{k,l+i} > 0$, т. е. не более S. Далее из этих чисел выбирается максимум. Для

этого тратится не более $S(S+1)/2$ операций. Всего на шаге 5 получается $S(S+7)/2$ операций. Для рассматриваемого примера будет 9 операций.

6. Всего будет не более $S(S+7)/2$ операций.

7. Имеем S операций умножения, S-1 – сложения, 1 – сравнения. Итого: $2S$ операций. Если $\sum_{i=l+1}^{S} b_{pi}b_{ki} > 0$, то вычисляется *ЛЧ*(19), для этого используется $2S+2$ операций. Всего: $4S+2$. Находится $\varepsilon_k = \max\{ЛЧ(15), ЛЧ(16), ЛЧ(19)\}$ – 3 операции.

8, 9. Аналогично п. 5, т. е. $S(S+7)/2$ операций.

10. Всего - $4S+2$ операций. Находится ε_k^2 – 3 операции.

11. Сравнение ε_k^1 и ε_k^2 – 1 операция.

12. Если $\varepsilon_k^1 \le \varepsilon_k^2$, то вычисляем x_i $(i=\overline{l+1,S})$, x_k $(k=\overline{1,l})$ по формулам (12)-(14). Имеем $3S+3+4(l-1)$ новых операций. Итог: точка M_k – 1 операция написания вектора.

Переходим к Алгоритму 2.

13. Если $\varepsilon_k^1 > \varepsilon_k^2$, то полагаем $k=k+1$ и весь процесс повторяется, пока не будет выполняться условие $\varepsilon_k^1 \le \varepsilon 2$. Если это условие не будет выполняться ни для какого k, то это означает несовместность системы ограничений.

Подведем итог. Согласно перечисленным пунктам при реализации Алгоритма 1 (при k=1) потребуется общее количество операций сложения, умножения, деления, сравнения и присваивания, равное

$$R_1=(2S+3)S+(2S-1)l+2S+2+S(S+7)/2+S(S+7)/2++4S+2+3+S(S+7)/2+S(S+7)/2+4S+2+3+1+ \\ +3S+3+4(l-1)+1=4S^2+2Sl+27S+3l+13. \quad (26)$$

Для рассматриваемого примера общее число операций составит 97. Для k=2 учитываются только 11 пунктов. Поэтому здесь будет 57 операций.

При переходе от системы (3) к системе (4) число операций не превосходит величины

$$R_2=(l-1)(2nl-4l^2/3+14l/3+1). \quad (27)$$

Для рассматриваемого примера это будет 21 операция.
Таким образом, общее число операций на данном этапе по порядку не превосходит n^2l.

Рассмотрим оценку δ_{l+i}^1 (или W_p^1), определяющую число шагов Алгоритма 2. Должно выполняться условие $\delta_{i+1}^1 \ge \sum_{j=1}^{q} ПЧ(25)$.

Усиливая данное неравенство, считаем, что выполняется условие:

$$\sum_{j=1}^{q} \frac{u_i}{r^j} = u_i \frac{1-1/r^q \, 1/2}{1-1/r} \ge \delta_{l+i}^1 \ge \sum_{j=1}^{q} \frac{a_i}{p^j} = a_i \frac{1-1/p^q}{1-1/p}, \quad (28)$$

где $a_i(u_i)$ - минимальное (максимальное) значение, такое, что слагаемое с номером j $(j=\overline{1,q})$ в (28) больше (меньше) по абсолютной величине соответствующего слагаемого $ПЧ(25)$ $(p>1,\ r>1)$. Тогда $\frac{a_i}{\delta_{l+i}} \ge 1-\frac{1}{p}$ $\left(\frac{u_i}{\delta_{l+i}} \ge 1-\frac{1}{r}\right)$. Докажем это.

Покажем, что $a_i \Big/ \sum_{j=1}^{q} \frac{a_i}{p^j} \ge 1 - \frac{1}{p}$. Имеем: $\frac{(1-1/p)\,p^q}{p^q - 1} \ge 1 - 1/p$, т. е. $1 \ge 1/p$.

Аналогично доказывается второе неравенство.

Кроме того, $1/p^q \le 1 - \frac{\delta_{l+i}}{u_i} \cdot (1 - 1/p)$ и $1/r^q \ge 1 - \frac{\delta_{l+i}}{u_i} \cdot (1 - 1/r)$.

Таким образом, логарифмируя, получаем

$$-\log_r(1 - \frac{\delta_{l+i}}{u_i}(1 - \frac{1}{r})) \le q \le -\log_p(1 - \frac{\delta_{l+i}}{a_i}(1 - \frac{1}{p})). \qquad (29)$$

Выражения в скобках (29) дробные. Если точность вычислений равна 0,0001, то тогда наибольшее значение –log будет, если выражение в скобках равно 0,0001, а наименьшее, если 1. Отсюда с учетом (29) получаем, что $q \le \log_p 10000$. Например, если p=10, то $q \le 4$.

Для рассматриваемого примера оказалось достаточно одной итерации Алгоритма 1 (l=1), переходить к проекциям на другую плоскость стало нецелесообразно. В Алгоритме 2 использовалась q=4 итерации.

К трудоемкости Алгоритма 1 прибавляется число шагов, связанных с переходом от системы (2) к системе (4), т. е. для приведенного примера это составит 21 операцию. Итого всего – не более $97 \cdot 4 + 21 + 57 = 466$ операций.

Таким образом, общее число операций в соответствии с Алгоритмами 1 и 2 не превосходит $R_1 ql + R_2$.

Для рассматриваемого примера использовалось проектирование только на одну плоскость (A_1). В общем случае весь процесс может повториться для всех l плоскостей. Для каждой гиперплоскости будет получен свой максимум. Выбор максимального из них можно осуществлять поэтапно, сравнивая последующее значение с предыдущим и оставляя наибольшее. Поэтому к общему числу шагов прибавляется еще значение l. Для рассматриваемого примера l=2.

При решении данного примера симплекс-методом потребуется не менее 90 операций сложения, умножения, сравнения, присваивания только при приведенных к стандартному виду данных и построении первой симплекс-таблицы.

Теорема 3.

Общее количество операций сложения, умножения, деления, сравнения и присваивания предложенного метода не превосходит величины

$$R_1 ql + R_2 + l, \qquad (30)$$

где R_1 определяется из (26), q из (29), R_2 из (27).

Замечание 1. В общем случае будут перебираться не все гиперплоскости, т. к. для некоторых из них может не выполняться основное условие. Таким образом, число операций будет меньше, чем в формуле (30).

Замечание 2. Величина q является константой, при грубых приближениях достаточно большой. Однако, при достаточно большом числе переменных n сумма (30) по порядку будет сопоставима с числом $n^2 l$.

Вычислительная сложность разработанного метода меньше, чем симплекс-метода, приведенная в статье [10].

Заключение

Основными результатами работы являются:

1) разработан новый метод решения задач ЛП с полностью ограниченными областями допустимых решений, обладающий меньшей вычислительной сложностью, чем симплекс-метод;

2) приведены 2 алгоритма поиска опорного и оптимального решений;

3) оценена вычислительная сложность разработанного метода.

СПИСОК ИСТОЧНИКОВ

1. Жевнеров В.А. Модификация симплекс-метода на основе принципа эволюции. *Проблемы управления*. 2004;1:28–31.
2. Базилевский М.П. Отбор информативных регрессоров с учетом мультиколлинеарности между ними в регрессионных моделях как задача частично-булевого линейного программирования. *Моделирование, оптимизация и информационные технологии*. 2018;2(22):104–118.
3. Базилевский М.П. Сведение задачи отбора информативных регрессоров при оценивании линейной регрессионной модели по методу наименьших квадратов к задаче частично-булевого линейного программирования. *Моделирование, оптимизация и информационные технологии.* 2018;1(20):108–117.
4. Сумин В.И., Кузнецова Л.Д., Лукин М.А. Определение коэффициентов математической модели управления качеством обучения методом линейного программирования. *Моделирование, оптимизация и информационные технологии*. 2018;3(22):214–222.
5. Шаповалов А.В., Преображенский А.П., Чопоров О.Н. Возможности применения методов оптимизации в управлении портфелями проектов. *Моделирование, оптимизация и информационные технологии*. 2020;8(1).
6. Жилина А.А., Кострова В.Н., Преображенский Ю.П. Разработка методики постановки задачи выбора управленческого решения на основе оптимизационного подхода. *Моделирование, оптимизация и информационные технологии*. 2018;6(1):243–253.
7. Ганичева А.В., Ганичев А.В. *Математическое программирование*. СПб.: Лань; 2020. 88 с.
8. Хасанов А.С. Об особенностях алгоритмов решения задач линейного программирования с неограниченными областями допустимых решений. *Вестник Московского государственного областного университета. Серия: Физика-математика*. 2017;1:113–123.
9. Ганичева А.В., Ганичев А.В. Метод решения некоторых классов оптимизационных задач. *Моделирование, оптимизация и информационные технологии.* 2019;7(2):43–54.
10. Березнев В.А. О полиномиальной сложности одной модификации симплекс-метода. *Журнал вычислительной математики и математической физики.* 2004;44(7):1244–1260.

REFERENCES

1. Zhevnerov V.A. Modification of the simplex method based on the evolution principle. *Problemy upravleniya = Control Sciences*. 2004;1:28–31. (In Russ.).
2. Bazilevskij M.P. Selection of Informative Regressors Taking into Account Multicollinearity Between Them in Regression Models as a Problem of Partial-Boolean Linear Programming. *Modelirovaniye, optimizatsiya i informatsionnyye tekhnologii = Modeling, optimization and information technology*. 2018;2(22):104–118. (In Russ.).

3. Bazilevskij M.P. Reduction of the problem of selecting informative regressors in estimating a linear regression model by the least squares method to a problem of partially Boolean linear programming. *Modelirovaniye, optimizatsiya i informatsionnyye tekhnologii = Modeling, optimization and information technology*. 2018;1(20):108–117. (In Russ.).
4. Sumin V.I., Kuznecova L.D., Lukin M.A. Determination of the coefficients of the mathematical model of education quality control by the method of linear programming. Modeling, optimization and information technology. 2018;3(22):214–222. (In Russ.).
5. Shapovalov A.V. Preobrazhenskij A.P., Choporov O.N. Possibilities of using optimization methods in project portfolio management. *Modelirovaniye, optimizatsiya i informatsionnyye tekhnologii = Modeling, optimization and information technology*. 2020;8(1). (In Russ.).
6. Zhilina A.A., Kostrova V.N., Preobrazhenskij Ju.P. Development of a methodology for setting the problem of choosing a management decision based on an optimization approach. *Modelirovaniye, optimizatsiya i informatsionnyye tekhnologii = Modeling, optimization and information technology*. 2018;6(1):243–253. (In Russ.).
7. Ganicheva A.V., Ganichev A.V. *Mathematical programming*. SPb.: Lan'; 2020. 88 p. (In Russ.).
8. Hasanov A.S. On the Features of Algorithms for Solving Linear Programming Problems with Unlimited Domains of Admissible Solutions. *Bulletin of Moscow Region State University*. *Series: Physics and Mathematics*. 2017;1:113–123. (In Russ.).
9. Ganicheva A.V., Ganichev A.V. A method for solving some classes of optimization problems. *Modelirovaniye, optimizatsiya i informatsionnyye tekhnologii = Modeling, optimization and information technology*. 2019;7(2):43–54. (In Russ.).
10. Bereznev B. A. On the polynomial complexity of a modification of the simplex method. *Zhurnal Vychislitel'noi Matematiki i Matematicheskoi Fiziki = Computational Mathematics and Mathematical Physics*. 2004;44(7):1244–1260. (In Russ.).

ИНФОРМАЦИЯ ОБ АВТОРАХ / INFORMATION ABOUT THE AUTHORS

Ганичева Антонина Валериановна, кандидат физико-математических наук, доцент, кафедра физико-математических дисциплин и информационных технологий, Тверская государственная сельскохозяйственная академия, Тверь, Российская Федерация.
e-mail: TGAN55@yandex.ru
ORCID: 0000-0002-0224-8945

Antonina V. Ganicheva, Candidate of Physical and Mathematical Sciences, Associate Professor, Department of Physical and Mathematical Disciplines and Information Technology, Tver State Agricultural Academy, Tver, Russian Federation.

Ганичев Алексей Валерианович, доцент, кафедра информатики и прикладной математики, Тверской государственный технический университет, Тверь, Российская Федерация.
e-mail: alexej.ganichev@yandex.ru
ORCID: 0000-0003-3389-7582

Aleksey V. Ganichev, Associate Professor, Department of Informatics and Applied Mathematics, Tver State Technical University, Tver, Russian Federation.

Статья поступила в редакцию 30.08.2022; одобрена после рецензирования 12.09.2022; принята к публикации 19.09.2022.

The article was submitted 30.08.2022; approved after reviewing 12.09.2022; accepted for publication 19.09.2022.