

УДК 338.27:330.43

## ИСПОЛЬЗОВАНИЕ СРЕДЫ МАТЛАВ ПРИ РЕШЕНИИ ЗАДАЧ ЛИНЕЙНОГО ПРОГРАММИРОВАНИЯ

**Бородин Г.А., Титов В.А., Маслякова И.Н.**

*ФГБОУ ВО «Российский экономический университет имени Г.В. Плеханова», Москва,  
e-mail: vtitov213@yandex.ru, maslyakova@gmail.com, georborodin@gmail.com*

Настоящая статья посвящена разбору использования среды программирования MatLab при решении задач линейного программирования. Идея статьи возникла при обсуждении преподавания дисциплины «Линейная алгебра». Квалифицированный специалист должен довольно неплохо разбираться не только в методах решения задач, но и уметь проанализировать полученные результаты. Многочисленные математические пакеты и программы позволяют это сделать довольно легко и быстро. Мы предлагаем на примере одного такого пакета MatLab рассмотреть решение задач линейного программирования и с его помощью сделать анализ полученного результата. Были написаны алгоритмы, позволяющие разрешать задачи линейной оптимизации. Данный код позволяет продемонстрировать практическое применение теоретического подхода на современных вычислительных системах, раскрывая смысл каждого показателя. Была разобрана задача-пример, решенная при помощи данного кода.

**Ключевые слова:** линейное программирование, MatLab, программирование, оптимальное решение, двойственная задача

## SOLVING LINEAR PROGRAMMING PROBLEMS WITH MATLAB

**Borodin G.A., Titov V.A., Maslyakova I.N.**

*Plekhanov Russian University of Economics, Moscow,  
e-mail: vtitov213@yandex.ru, maslyakova@gmail.com, georborodin@gmail.com*

This article is devoted to analysis of using MatLab environment for solving linear programming problems. The idea of the article appeared due to discussion of teaching «Linear Algebra». A qualified person should understand quite well not only the methods of solving problems, but also be ready to analyze results. We offer MatLab environment as a tool for solving linear programming problems, and with his help to make the analysis of the result. Algorithms were created for solving linear optimization problems. The code allows demonstration of practical use of theoretical approach on modern computing systems, explaining every index. Example task was solved, using this code.

**Keywords:** linear programming, MatLab, programming, the optimal solution, dual problem

Еще в XIX в. общество начало предпринимать первые попытки составить математическую модель экономических проблем для создания точных прогнозов и принятия оптимальных решений. В 1939 г. советский математик Л.В. Канторович опубликовал работу «Математические методы организации и планирования производства», в которой заложил основы линейного программирования. Сам термин был предложен в середине 1940-х гг. американским математиком Джорджем Бернардом Данцигом, автором симплекс-метода [1]. В те времена вычислительная техника пребывала в самом начале своего развития, поэтому симплекс-таблицы, содержащие большое количество переменных, требовали большого количества ресурсов. С развитием IT-технологий, и в особенности появлением персональных компьютеров, исчезла проблема большого количества переменных при решении задач математического программирования. Но появилась другая проблема – проблема использования IT-технологий при подготовке специалистов. Ни для кого не

секрет, что во многих российских вузах при подготовке экономистов практически не используются современные математические пакеты и программы для решения тех или иных задач. А это, в свою очередь, снижает конкурентоспособность нашего образования, а экономика нашей страны недополучает высококвалифицированных специалистов [3].

Идея написания данной работы возникла при изучении раздела линейной алгебры «Линейное программирование». Одним из достоинств, но и в то же время недостатков нашего образования является фундаментальность и теоретизированность подачи материала. На наш взгляд, необходим баланс между фундаментальной теорией, решением практических задач и применением современных технологий [2]. В данной работе продемонстрирован такой подход на примере решения задачи линейного программирования и ее постоптимального анализа.

Целью данной работы является создание универсального инструмента на базе среды Matlab для работы с задачами по-

стоптимального анализа, который за счёт своей открытости не только может легко быть перестроен под специализированные задачи, но и стать иллюстрацией методов данного вида анализа экономических задач. Студенты смогут легко проследить, а следовательно, и понять, каким образом при изменении базисных переменных меняется значение целевой функции.

**Общая постановка задачи оптимизации**

В ходе решения задачи оптимизации необходимо будет выполнить следующие этапы решения:

- Получить оптимальный план прямой задачи
- Определить дефицитные ресурсы
- Получить оптимальный план двойственной задачи
- Найти интервалы устойчивости ресурсов
- Предоставить данные пользователю

**Постановка и решение прямой задачи линейного программирования**

Общей задачей линейного программирования является задача, которая состоит в определении максимального или же минимального значения целевой функции при заданной системе ограничений (рис. 1).

$$Z(X) = c_1x_1 + c_2x_2 + \dots + c_nx_n \rightarrow \max(\min)$$

$$\begin{cases} a_{11}x_1 + a_{12}x_2 + \dots + a_{1n}x_n = b_1 \\ a_{21}x_1 + a_{22}x_2 + \dots + a_{2n}x_n = b_2 \\ \dots \dots \dots \\ a_{m1}x_1 + a_{m2}x_2 + \dots + a_{mn}x_n = b_m \end{cases}$$

$$x_j \geq 0, j = 1, 2, \dots, n$$

Рис. 1. Общий вид задачи линейного программирования

```
f = [1; 0.7; 1.1; 2; 0.6];
A = [0.8 0.5 1 2 1.1; 0.2 0.1 0.1 0.1 0.2; 0.3 0.4 0.6 1.3 0.05; 0.2 0.3 0.3 0.7 0.5; 0.7 0.1 0.9 1.5 0];
b = [14111; 138; 815.5; 466; 1080];
Aeq=[];
beq=[];
lb = [0; 0; 0; 0; 0];
ub=[];
task =1;
if task == 1
f = -f;
end

[x,fval] = linprog(f,A,b,Aeq,beq,lb,ub);
```

Рис. 3. Запись системы ограничений примера в среде MatLab и нахождение решения

К данным задачам сводятся многочисленные задачи экономического содержания [5]. Для решения этих задач в среде Matlab используется функция linprog из дополнения OptimizationToolbox. Работа с данной функцией производится путём задания пользователем данных о задаче следующими условиями:

- $A*x \leq b$  (где A – матрица коэффициентов при переменных в ограничениях-неравенствах, b – вектор свободных членов в ограничениях-неравенствах, x – вектор значений переменных);
- $Aeq*x = beq$  (где Aeq – матрица коэффициентов при переменных в ограничениях-равенствах, beq-вектор свободных членов в ограничениях-равенствах, x – вектор значений переменных);
- $lb \leq x \leq ub$  (ограничения на координаты, заданные двумя векторами).

Таким образом, для нахождения оптимального решения прямой задачи следует обратиться к функции в форме

$$[x, fval] = \text{linprog}(f, A, b, Aeq, beq, lb, ub)$$

После выполнения данной функции, в массиве x будут находиться значения аргументов функции для оптимального решения, а переменная fval будет равна значению целевой функции при данных значениях.

Приведем пример решения задачи:

$$\begin{aligned} & x_1 + 0,7x_2 + 1,1x_3 + 2x_4 + 0,6x_5 \rightarrow \max \\ & \begin{cases} 0,8x_1 + 0,5x_2 + x_3 + 2x_4 + 1,1x_5 \leq 1411 \\ 0,2x_1 + 0,1x_2 + 0,1x_3 + 0,1x_4 + 0,2x_5 \leq 149 \\ 0,3x_1 + 0,4x_2 + 0,6x_3 + 1,3x_4 + 0,05x_5 \leq 815,5 \\ 0,2x_1 + 0,3x_2 + 0,3x_3 + 0,7x_4 + 0,5x_5 \leq 466 \\ 0,7x_1 + 0,1x_2 + 0,9x_3 + 1,5x_4 \leq 1080 \end{cases} \end{aligned}$$

Рис. 2. Система ограничений примера

Соответственно, на экране данная задача будет выглядеть так:

Здесь task = 1 (переменная-маркер поиска максимума/минимума целевой функции).

Значения в x на выходе будут:

$$\begin{aligned} x_1 &= 454,2553 \\ x_2 &= 58,5106 \\ x_3 &= 0,0000 \end{aligned}$$

$$\begin{aligned} x_4 &= 504,1135 \\ x_5 &= 9,4326 \end{aligned}$$

При выполнении данной функции также можно сразу же получить значения теневых цен ресурсов, заменив запись на:

```
[x,fval,exitflag,output,lambda] = linprog(f,A,b,Aeq,beq,lb,ub);
```

Рис. 4. Вызов функции с введением дополнительных аргументов для получения теневых цен

```
disp('Значения теневых цен ресурсов: ');
disp(lambda.ineqlin);
max=lambda.ineqlin(1);
maxn=1;
for i=1:length(lambda.ineqlin)
    if lambda.ineqlin(i)>max
        max=lambda.ineqlin(i);
        maxn=i;
    end
end

disp('Самым дефицитным товаром является товар под номером: ')
disp(maxn);
disp('Его теневая цена: ');
disp(max);
```

Рис. 5. Автоматизированный поиск наиболее дефицитного товара

В таком случае (рис. 4) в матрице lambda.ineqlin будут содержаться значения множителей Лагранжа для данной задачи. По своему экономическому смыслу множители Лагранжа одинаковы с теневыми ценами, которые показывают, насколько дефицитным является товар [6].

Код, представленный на рис. 5, показывает, как на основании полученных значений в матрице lambda.ineqlin определить наиболее дефицитный товар и его теневую цену. В теле цикла в переменной max содержится последнее известное максимальное значение, а в maxn – его порядковый номер.

Составление двойственной задачи линейного программирования

прямая	двойственная
$(c, x) - \min$	$(b, y) - \max$
$Ax \geq b$	$AT y \leq c$
$x \geq 0$	$y \geq 0$

Рис. 6. Взаимосвязь коэффициентов прямой и двойственной задач

Двойственную задачу к любой прямой задаче можно вывести, руководствуясь данными правилами:

- Если прямая задача является задачей максимизации, то двойственная будет задачей минимизации, и наоборот.

- Каждому ограничению типа стандартного неравенства прямой задачи соответствует неотрицательная двойственная переменная двойственной задачи, и наоборот.

- Каждому ограничению типа равенства прямой задачи соответствует двойственная переменная без ограничения на знак, и наоборот.

- Коэффициенты целевой функции прямой задачи становятся свободными членами ограничений двойственной задачи, и наоборот.

- Матрицу ограничений двойственной задачи получают транспонированием матрицы ограничений прямой задачи.

Вспользуемся функцией:

```
[y,doubleval]=linprog(b,-A',f,-Aeq',[],lb);
```

Рис. 7. Код для решения двойственной задачи на основе заданных коэффициентов прямой задачи

В коде (рис. 7) используется тот же метод linprog, однако меняются аргументы функции в соответствии с теорией. В матрице y будут находиться значения неизвестных двойственной задачи, а в переменной doubleval – значение целевой функции двойственной задачи.

### Нахождение интервалов устойчивости ресурсов

Для нахождения интервалов устойчивости ресурсов будем изменять количество каждого из ресурсов поочередно с заданным шагом до момента, когда решение перестанет быть оптимальным. Для этого будем, изменив значение свободного члена ограничений, сверять значения в матрице оптимальных оценок с изначальной матрицей оптимальных оценок.

```
%Анализируем интервалы чувствительности ресурсов:
oldy=y;
safeb=b;
for i=1:length(b)
    while (y==oldy)
        b(i)=b(i)-approxim;
    end
    disp(b(i));
    b=safeb;
end
```

Рис. 8. Код для нахождения нижней границы устойчивости ресурсов

Представленный на рис. 8 код для нахождения нижней границы устойчивости ресурсов легко можно использовать для установления верхней границы устойчивости, заменив операцию вычитания на операцию сложения. Смысл данного алгоритма состоит в том, что при изменении значения какого-либо ресурса в пределах его интервала устойчивости, значение целевой функции не изменяется. В переменной `approxim` задается значение приближения, чем оно меньше – тем точнее получаемый результат, однако медленнее вычисления. Таким образом, каждый раз, когда мы изменяем значение какого-то из ресурсов на значение переменной `approxim`, мы сравниваем его со значением целевой функции при начальных условиях. Как только значения перестают совпадать – обнаруживает себя граница интервала устойчивости, и данную процедуру можно повторять для следующего ресурса [4].

### Выводы

Среда Matlab – крайне гибкий инструмент. Её использование в задачах оптимизации позволяет создать не просто приложение, но и наглядное пособие при обучении

работе с задачами линейного программирования. На основании составленного «каркаса» возможно изменение используемых алгоритмов, подстройка кода под определенную задачу, а также настройка работы с большими данными, оформленными в форматах текстов файлов или записей БД. Заметим, что это только первый шаг на пути к созданию нового подхода в преподавании математических дисциплин.

### Список литературы

1. Данциг Д. Линейное программирование, его применения и обобщения. – М., Прогресс, 1966. – 600 с.
2. Маслякова И.Н. Применение процедуры Байесовского оценивания в последовательности тестов при подготовке специалистов // European researcher. Series A. – 2011. – № 5–1. – С. 509–510.
3. Маслякова И.Н., Мочалина Е.П., Татарников О.В., Иванкова Г.Н. Модель рекуррентного оценивания уровня знаний студента // Менеджмент и бизнес-администрирование. – 2015. – № 3. – С. 71–76.
4. Титов В.А., Долгополов А.А. Применение симплекс-метода для решения задач динамического управления запасами организации // Современные проблемы науки и образования. – 2014. – № 3; URL: <http://www.science-education.ru/article/view?id=13074>.
5. Титов В.А., Долгополов А.А. Линейные математические модели управления запасами производственной компании с учетом фактора времени // Успехи современного естествознания. – 2015. – № 1–1. – С. 170–171.
6. Титов В.А., Максимов Д.А. Методы оценки экономической безопасности производственной сферы интегрированной производственной структуры // Путеводитель предпринимателя. – 2013. – № 18. – С. 188–205.

### References

1. Dancig D. Linejnoe programmirovaniye, ego primeneniya i obobshheniya. M., Progress, 1966. 600 p.
2. Masljakova I.N. Primeneniye procedury Bajesovskogo ocenivaniya v posledovatelnosti testov pri podgotovke specialistov // European researcher. Series A. 2011. no. 5–1. pp. 509–510.
3. Masljakova I.N., Mochalina E.P., Tatarnikov O.V., Ivankova G.N. Model rekurrentnogo ocenivaniya urovnya znaniy studenta // Menedzhment i biznes-administrirovaniye. 2015. no. 3. pp. 71–76.
4. Titov V.A., Dolgopolov A.A. Primeneniye simpleks-metoda dlja resheniya zadach dinamicheskogo upravleniya zapasami organizacii. //Sovremennyye problemy nauki i obrazovanija. 2014. no. 3. pp. 321; URL: <http://www.science-education.ru/article/view?id=13074>.
5. Titov V.A., Dolgopolov A.A. Linejnye matematicheskie modeli upravleniya zapasami proizvodstvennoj kompanii s uchetom faktora vremeni // Uspehi sovremennogo estestvoznaniya. 2015. no. 1–1. pp. 170–171.
6. Titov V.A., Maksimov D.A. Metody ocenki jekonomicheskoj bezopasnosti proizvodstvennoj sfery integrirovannoj proizvodstvennoj struktury // Putevoditel predprinimatelja. 2013. no. 18. pp. 188–205.