# Лабораторная работа №3
# Линейное программирование: симплекс метод

## Цель работы

Приобрести практические навыки решения задач линейного программирования с использованием симплекс-метода.

## Теоретическая часть

Рассмотрим задачу поиска максимума функции $f : \mathbb{R}^n \to \mathbb{R}$

$$f(x) = \sum_{i=1}^{n} c_i x_i \to \max\,, \qquad x \in \mathbb{R}^n \tag{1}$$

при следующих ограничениях

$$\sum_{i=1}^{n} a_j^i x_i = b_j\,, \qquad x_i \geqslant 0\,, \quad j = 1, \ldots, m\,, \quad i = 1, \ldots, n\,, \quad m < n \tag{2}$$

Задача (1)-(2) может быть записана в матричной форме

$$\langle c, x \rangle \to \max\,, \qquad \mathbf{A}x = b\,, \quad x \geqslant 0\,, \tag{3}$$

где $x, c \in \mathbb{R}^n$, $b \in \mathbb{R}^m$, $\langle c, x \rangle := \sum\limits_{j=1}^{n} c_j x_j$, $\mathbf{A} = \begin{pmatrix} a_1^1 & \cdots & a_1^n \\ \vdots & \ddots & \vdots \\ a_m^1 & \cdots & a_m^n \end{pmatrix}$. При этом, вектор $c$ принято называть вектором стоимости, $b$ — вектором ограничений, $\mathbf{A}$ — матрицей условий, а задача (3) называется канонической задачей линейного программирования.

Задача линейного программирования может быть задана в общей форме

$$\langle c, x \rangle \to \max\,, \qquad \mathbf{A}x \leqslant b\,, \quad x \geqslant 0\,. \tag{4}$$

Переход от общей формы к канонической осуществляется путем замены исходной задачи (4) задачей на максимум и введением дополнительных координат $\tilde{x}(x_{n+1}, \ldots, x_{n+m})$:

$$\langle -c, x \rangle \to \max\,, \qquad \mathbf{A}x + \mathbf{E}\tilde{x} = b\,, \quad x \geqslant 0\,, \quad \tilde{x} \geqslant 0\,,$$

где E — единичная матрица.

Точка $d$ выпуклого множества $D$ называется крайней, если не существует точек $d_1, d_2 \in D$, $d_1 \neq d_2$ и числа $t \in (0, 1)$, таких что $d = td_1 + (1 - t)d_2$. Очевидно, что если множество $D$ является многогранником, то крайние точки — вершины этого многогранника.

Эктсремум линейной функции $\langle c, x\rangle$ достигается в крайней точке выпуклого многогранника $D := \{x \in \mathbb{R}^n | Ax = b, x >= 0\}$. Число крайних точек множества $D$ конечно. Поэтому для решения задачи линейного программрования (3), если оно существует, достаточно перебрать значения функции $\langle c, x\rangle$ во всех крайних точках множества $D$. Опишем симплекс-метод решения задачи линейного программирования, который позволяет, начиная с исходной крайней точки, переходить к другой в направлении возрастния целевой функии $\langle c, x\rangle$.

Задача (3) называется невырожденной задачей линейного программирования, если любая крайняя точка множества ограничений (2) содержит ровно $m$ положительных координат.

Пусть $x$ — крайняя точка в невырожденной задаче (3) с $m$ положительными координатами. Для определенности будем считать, что положительными являются первые $m$ координат. Тогда вектор $x$ можно представить в виде $x = (x_b, x_k)$, где $x_b = (x_1, \ldots, x_m)$, $x_i > 0$ — базисный вектор, а $x_k = (0, \ldots, 0) \in \mathbb{R}^{n-m}$. Аналогично матрицу $\mathbf{A}$ можно представить в виде $\mathbf{A} = (\mathbf{A}_b, \mathbf{A}_k)$.

### Алгоритм решения задач линейного программирования симплекс-методом

1. Привести задачу к канонической форме.

2. Отыскать крайнюю точку $x = (x_1, \ldots, x_m, 0, \ldots, 0)$, $x_i > 0$, $i = \overline{1, m}$ множества допустимых элементов $D$.

3. Построить симплексную таблицу для начальной крайней точки $x$:

| | $c$ | | $c_1$ | $\ldots$ | $c_m$ | $c_{m+1}$ | $\ldots$ | $c_{j^*}$ | $\ldots$ | $c_n$ | $t$ |
|---|---|---|---|---|---|---|---|---|---|---|---|
| базис | | $b$ | $a^1$ | $\ldots$ | $a^m$ | $a^{m+1}$ | $\ldots$ | $a^{j^*}$ | $\ldots$ | $a^n$ | |
| $a_1$ | $c_1$ | $b_1$ | 1 | $\ldots$ | 0 | $a_1^{m+1}$ | $\ldots$ | $a_1^{j^*}$ | $\ldots$ | $a_1^n$ | $t_1$ |
| $\vdots$ | $\vdots$ | $\vdots$ | $\vdots$ | $\vdots$ | $\vdots$ | $\vdots$ | $\vdots$ | $\vdots$ | $\vdots$ | $\vdots$ | $\vdots$ |
| $a_{i^*}$ | $c_{i^*}$ | $b_{i^*}$ | 0 | $\ldots$ | 0 | $a_{i^*}^{m+1}$ | $\ldots$ | $a_{i^*}^{j^*}$ | $\ldots$ | $a_{i^*}^n$ | $t_{i^*}$ |
| $\vdots$ | $\vdots$ | $\vdots$ | 0 | $\vdots$ | 1 | $\vdots$ | $\vdots$ | $\vdots$ | $\vdots$ | $\vdots$ | $\vdots$ |
| $a_m$ | $c_m$ | $b_m$ | 1 | $\ldots$ | 0 | $a_m^{m+1}$ | $\ldots$ | $a_m^{j^*}$ | $\ldots$ | $a_m^n$ | $t_m$ |
| $z$ | | $\sum\limits_{i=1}^m c_i b_i$ | $c_1$ | $\ldots$ | $c_m$ | $\sum\limits_{i=1}^m c_i a_i^{m+1}$ | $\ldots$ | $\sum\limits_{i=1}^m c_i a_i^{j^*}$ | $\ldots$ | $\sum\limits_{i=1}^m c_i a_i^n$ | |
| $\Delta$ | | | 0 | $\ldots$ | 0 | $z_m - c_m$ | $\ldots$ | $z_{j^*} - c_{j^*}$ | $\ldots$ | $z_n - c_n$ | |

4. Исследовать симплексную таблицу.

   - Если вектор $\Delta \geqslant 0$, то крайняя точка $x$ — решение задачи.
   - Если для некоторого $j$ выполняется $\Delta_j < 0$ и $x^j < 0$ то решение задачи (3) $\langle c, x\rangle = +\infty$.
   - Пусть в строке $\Delta$ имеются отрицательные числа, а соответствующие столбцы $x^j$ содержат положительные числа.

Предположим, что $\min \Delta_j = \Delta_{j^*} < 0$. Ясно, что $m+1 \leqslant j^* \leqslant n$. Столбец, соответствующий индексу $j^*$ называется разрешающим столбцом. Если $\min \Delta_j$ достигается на нескольких значениях $j$, то в качестве разрешающего столбца выбираем столбец с любым таким индексом.

Обозначим $t_i := \left\{ \dfrac{b_i}{a_i^{j^*}} \middle| a_i^{j^*} > 0 \right\} > 0$. Эти значения $t_i$ ставим соответственно в последнем столбце симплексной таблицы.

Пусть $t_{i^*} = \min_i t_i > 0$. Строка вектора $a_{i^*}$ называется разрешающей. Если $\min_i t_i$ достигается на нескольких значениях $i$, то в качестве разрешающей строки выбираем любую такую строку. Элемент $a_{i^*}^{j^*}$ называется *разрешающим элементом* симплексной таблицы.

Далее из числа базисных векторов исключаем вектор $a_{i^*}$, вместо него берем вектор $a_{j^*}$. Значение функционала на новой крайней точке с новыми базисными векторами $a_1, \ldots, a_{j^*-1}, a_{j^*}, a_{j^*+1}, \ldots, a_m$ возрастет на величину $-t_{i^*}\Delta_{j^*}$ .

5. Построить новую симплексную таблицу для нового базиса.

   Способ построения новой таблицы по предыдущей (*правило прямоугольника*):

$$b_i{}' = b_i - \frac{b_{i^*} a_i^{j^*}}{a_{i^*}^{j^*}}, \qquad a_i^{j\prime} = a_i^l - \frac{a_{i^*}^j a_i^{j^*}}{a_{i^*}^{j^*}}, \qquad i \neq j^*\,,$$

$$a_{i^*}^{j^*\prime} = 1\,, \qquad a_i^{j^*\prime} = 0\,, \qquad i \neq i^*, \quad i = 1, \ldots, m\,,$$

$$b_{i^*}{}' = \frac{b_{i^*}}{a_{i^*}^{j^*}}, \qquad a_{i^*}^{j}{}' = \frac{a_{i^*}^j}{a_{i^*}^{j^*}}, \qquad = ji = 1, \ldots, n\,.$$

   Далее возвращаемся к пункту 4, пока не придем к решению задачи.

## Методы нахождения начальной крайней точки

### Метод Жордана—Гаусса

Сводит систему $m$ уравнений с $n$ неизвестными к каноническому виду при помощи элементарных операций над строками. При использовании первых $m$ переменных $(x_1, x_2, \ldots, x_m)$ такая система имеет вид

$$\begin{gathered} x_1 + a_1^{m+1} x_{m+1} + \cdots + a_1^n x_n = b_1\,, \\ \vdots \\ x_k + a_k^{m+1} x_{m+1} + \cdots + a_k^n x_n = b_k\,, \\ \vdots \\ x_m + a_m^{m+1} x_{m+1} + \cdots + a_m^n x_n = b_m\,, \end{gathered} \tag{5}$$

Переменные $(x_1, x_2, \ldots, x_m)$, входящие с единичными коэффициентами только в одно уравнение системы (5) и с нулевыми в остальные, называются базисными. В канонической системе каждому уравнению соответствует одна базисная переменная.

Базисным решением системы (5) называется решение, полученное при нулевых небазисных переменных.

Базисное решение называется допустимым базисным решением, если значения, входящих в него базисных переменных $x_i \geqslant 0$, $i = \overline{1,m}$. Последнее условие эквивалентно условию $b_i \geqslant 0$, $i = \overline{1,m}$.

Допустимое базисное решение является крайней точкой множества допустимых решений задачи (3).

### Метод искусственного базиса

Будем считать, что все координаты вектора неотрицательны. Если, например, $b_i \leqslant 0$, то умножаем $j$-е уравнение на $-1$.

Введем искусственные переменные $\hat{x} = (x_{n+1}, \ldots, x_{n+m})$. Симплекс методом решаем вспомогательную задачу

$$-\sum_{i=1}^{m} x_{n+1} \to \max, \qquad \mathbf{A}x + \mathbf{I}\hat{x} = b\,, \qquad x \geqslant 0\,, \quad \hat{x} \geqslant 0\,. \tag{6}$$

Если при решении задачи (6) получаем, что не все искусственные переменные равны нулю, то задача (3) не имеет допустимых элементов.

Если решением задачи (6) является вектор $(x_1, \ldots, x_n, 0, \ldots, 0)$: все искусственные переменные равны нулю и среди базисных векторов нет векторов, соответствующих искусственным переменным, то найденная точка $(x_1, \ldots, x_n, )$ является крайней точкой множества допустимых решений задачи (3).

Если все искусственные переменные равны нулю и среди базисных векторов есть вектора, соответствующие искусственным переменным, то нужно исключить из числа базисных вектора, соответствующие искусственным переменным следующим образом. Пусть в последней таблице есть строка с искусственной переменной $x_{i^*}$. Эта строка считается разрешающей. В качестве разрешающего столбца берется столбец $a^{j^*}$, $j^* \leqslant n$, $a^{j^*}_{n+i^*} \neq 0$. Этот процесс закончится удалением всех искусственных переменных из базисных или окажется что $a^{j}_{n+i^*} = 0$, $j = \overline{1,n}$, что означает $i^*$-я строка исходной системы уравнений линейно зависима от остальных строк. Линейно зависимые равенства исключаются, и исходная задача решается симплекс методом с найденной крайней точкой, имеющей менее $m$ положительных элементов.

## Двойственность задач линейного программирования

Рассмотрим две задачи линейного программирования

$$\langle c, x\rangle \to \max; \qquad \mathbf{A}x \leqslant b\,, \quad x \geqslant 0\,, \tag{7}$$

$$\langle b^T, y\rangle \to \min; \qquad \mathbf{A^T} y \geqslant c^T\,, \quad y \geqslant 0\,, \tag{8}$$

Задача (8) является двойственной по отношению к задаче (7), и наоборот, задача (7) является двойственной по отношению к задаче (8) [1].

**Пример**[2]. Предприятие выпускает три вида продукции количества $x_1, x_2, x_3$ на трех различных типах установок. Количество ресурса времени установки $j$ для производства единицы продукции $i$ определяется величиной $a_i^j$. Ресурс времени установок каждого типа ограничен величиной $b_j$. Прибыль от единицы каждого вида продукции составляет $c_1, c_2, c_3$. Необходимо максимизировать прибыль.

Задача линейного программирования запишется в виде

$$\begin{array}{l} c_1x_1 + c_2x_2 + c_3x_3 \to \max, \\ a_1^1x_1 + a_1^2x_2 + a_1^3x_3 \leqslant b_1\,, \\ a_2^1x_1 + a_2^2x_2 + a_2^3x_3 \leqslant b_2\,, \\ a_3^1x_1 + a_3^2x_2 + a_3^3x_3 \leqslant b_3\,, \\ x_i \geqslant 0\,, \qquad i = 1, \ldots, 3\,, \end{array}$$

а двойственная к ней задача будет иметь вид

$$\begin{array}{l} b_1y_1 + b_2y_2 + b_3y_3 \to \min, \\ a_1^1y_1 + a_1^2y_2 + a_1^3y_3 \leqslant c_1\,, \\ a_2^1y_1 + a_2^2y_2 + a_2^3y_3 \leqslant c_2\,, \\ a_3^1y_1 + a_3^2y_2 + a_3^3y_3 \leqslant c_3\,, \\ y_i \geqslant 0\,, \qquad i = 1, \ldots, 3\,. \end{array}$$

Здесь $y_i$ — это оценка, соответствующая одной единице ограниченного ресурса по $i$-ой установке. Она равна величине, на которую могла бы увеличиться суммарная прибыль, если бы количество $i$-го ресурса увеличилось на единицу, и если это увеличение было бы использовано оптимально. Иными словами, $y_i$ — это количество прибыли, недополученной из-за нехватки единицы ограниченного ресурса $b_i$.

### Теорема двойственности линейного программирования [1]

Для пары двойственных задач справедлива альтернатива:

- если значение одной из задач конечно, то значение другой тоже конечно и эти значения совпадают;
- если множество допустимых элементов в одной из задач пусто, то другая задача либо несовместна, либо имеет бесконечное значение.

### Средства пакета Maple для решения задач линейного программирования

В систему математических пакетов Maple для решения задач линеного программирования входит пакет simplex

>with(simplex)

В пакет входят следующие команды

- basis возврат списка основных переменных для множества линейных уравнений;
- convexhull — вычисление выпуклой оболочки для набора точек;
- cterm — задание констант для системы уравнений или неравенств;
- display — вывод системы уравнений или неравенств в матричной форме;
- dual — выдача сопряженных выражений;
- equality — параметр для функции convert, указывающая на эквивалентность;
- feasible — выяснение возможности решения заданной задачи:
- maximize — вычисление максимума функции;
- minimize — вычисление минимума функции;
- pivot — создание новой системы уравнений с заданным главным элементом;
- pivoteqn — выдача подсистемы уравнений для заданного главного элемента;
- pivotvar — выдача переменных с положительными коэффициентами в целевой функции;
- ratio — выдача отношений для определения наиболее жесткого ограничения;
- setup — задание системы линейных уравнений;
- standardize — приведение заданной системы уравнений или неравенств к стандартной форме неравенств типа «меньше или равно».

Приведем пример поиска максимума линейной функции с линейными ограничениями.

```
>with(simplex);
>cnsts := 3*x+4*y-3*z <= 23, 5*x-4*y-3*z <= 10, 7*x+4*y+11*z <= 30;
>obj := -x+y+2*z;
>maximize(obj, cnsts, NONNEGATIVE)
```

Задачи линейного программирования можно решать в пакете Optimization, в котором введена функция

LPSolve(obj [, constr, bd, opts])

Она имеет следующие параметры:

- obj — алгебраическое выражений, целевая функция;
- constr — множество или список линейных ограничений;

- bd — последовательность вида name=range, задающая границы одной или многих переменных;

- opts — равенство или равенства в форме option=value, где option одна из опций assume, feasibilitytolerance, infinitebound, initialpoint, iterationlimit или maximize, специализированных для команды LPSolve.

Приведем пример поиска целочисленоого решения задачи линейного программирования
>with(Optimization):
>LPSolve(2*x+5*y, 3*x-y = 1, x-y <= 5, assume = integer, nonnegative)

## Порядок выполнения работы

1. Ознакомиться с постановкой задачи, определяемой вариантом задания к лабораторной работе.

2. При выполнении задания необходимо предусмотреть:

   - переменные по выпуску продукции каждого вида;
   - ограничения по использованию сырого молока и по времени загрузки автоматизированных фасовочных линий;
   - другие ограничения и переменные согласно индивидуальным вариантам задания.

3. Решить задачу симплексным методом. Разрешается получить начальное базисное решение любым известным методом.

4. По завершении расчётов выполнить проверку правильности численного решения с использованием программных средств линейной оптимизации пакета Maple (Simplex).

5. Составить и решить задачу, двойственную к задаче, соответствующей индивидуальному варианту задания.

6. Дать экономическую интерпретацию двойственной задачи и ее решения.

7. По результатам выполненной лабораторной работы составьте отчет, содержащий:

   - цель работы;
   - математическую запись задачи линейного программирования с указанием названий и единиц измерения переменных и ограничений;
   - поиск начальной крайней точки;

- решение поставленной задачи симплекс методом: привести все симплексные таблицы (исходную, все промежуточные, заключительную);
- оптимальное решение, оптимальное значение целевой функции и экономическую интерпретацию оптимального плана;
- листинг проверки решения поставленной задачи с помощью пакета Maple;
- математическая запись двойственной задачи линейного программирования с указанием названий и единиц измерения переменных и ограничений, оптимальное решение и оптимальное значение целевой функции, экономическая интерпретация;
- ответы на контрольные вопросы, приведенные в задании,

## Контрольные вопросы

1. Приведите три формы основной задачи линейного программирования.
2. Какая точка выпуклого множества называется крайней?
3. Какая задача линейного программирования называется невырожденной?
4. Назовите известные Вам методы нахождения начальной крайней точки.
5. Приведите основные этапы решения задачи линейного программирования симплекс методом.
6. Какие задачи линейного программирования называются двойственными?
7. Теорема двойственности линейного программирования.

## Задание к лабораторной работе

Составить и решить симплексным методом задачу линейного программирования (с учётом изменений, предусмотренных индивидуальным вариантом задания), предназначенную для составления оптимальной производственной программы молокоперерабатывающего предприятия при следующих условиях [2].

- Ассортимент выпускаемой продукции включает пастеризованное молоко, кефир и сметану, а также дополнительную продукцию согласно индивидуальному варианту задания.
- Затраты сырого молока составляют:
  - на пастеризованное молоко — 1,01 кг/кг;
  - на кефир — 1,01 кг/кг;

  - на сметану — 9,45 кг/кг.

- Поставщики в состоянии поставить не более 140 ц молока в сутки.

- Фасовка молока и кефира осуществляется на автоматизированной линии производительностью 5 ц молока или 6 ц кефира в час. В течение суток линия может эксплуатироваться не более 21 часа.

- Фасовка сметаны осуществляется на другой автоматизированной линии производительностью 30 кг сметаны в час. В течение суток линия может эксплуатироваться не более 16 часов.

- Цена реализации пастеризованного молока — 2,4, кефира — 2,7, сметаны — 13,8 тыс. руб./ц.

- План должен обеспечивать максимальную выручку от реализации молочной продукции (контракт на поставку молока уже оплачен).

**Варианты заданий**

1. Дополнительный вид продукции — творог. Цена — 5400 руб./ц. Затраты сырого молока — 13 ц/ц. Производительность фасовки на оборудовании для фасовки творога — 0,35 ц/ч. Оборудование может работать не более 17 ч/сут.

2. Дополнительный вид продукции — йогурт. Цена — 2500 руб./ц. Затраты сырого молока — 0,85 ц/ц. Производительность фасовки на оборудовании для фасовки сметаны — 0,15 ц/ч. Йогурта должно производиться не меньше, чем сметаны.

3. Дополнительный вид продукции — кефир обезжиренный. Цена — 770 руб./ц. Затраты сырого молока — 0,3 ц/ц. Производительность фасовки на оборудовании для фасовки молока и кефира — 6 ц/ч. Суммарный максимальный выпуск кефира обоих видов — 40 ц/сут. (минимальный выпуск кефира жирного не регламентируется).

4. Дополнительный вид продукции — творожные сырки. Цена — 7800 руб./ц. Затраты сырого молока — 14 ц/ц. Производительность фасовки на оборудовании для фасовки творожных сырков — 0,25 ц/ч. Оборудование может работать не более 18 ч./сут.

5. Дополнительный вид продукции — йогурт. Цена — 2750 руб./ц. Затраты сырого молока — 0,95 ц/ц. Производительность фасовки на оборудовании для фасовки сметаны — 0,25 ц/ч. Максимальный выпуск — 15 ц/сут.

6. Дополнительный вид продукции — кефир обезжиренный. Цена — 770 руб./ц. Затраты сырого молока — 0,27 ц/ц. Производительность фасовки на оборудовании для фасовки молока и кефира — 5.5 ц/ч. Суммарный минимальный выпуск кефира обоих видов — не более 8 ц/сут.

7. Дополнительный вид продукции — творог. Цена — 5500 руб./ц. Затраты сырого молока — 18 ц/ц. Производительность фасовки на оборудовании для фасовки творога — 0,3 ц/ч. Оборудование может работать не более 16 ч./сут.

8. Дополнительный вид продукции — йогурт. Цена — 2000 руб./ц. Затраты сырого молока — 0,8 ц/ц. Производительность фасовки на оборудовании для фасовки сметаны — 0,2 ц/ч. Максимальный выпуск — 12 ц/сут.

9. Дополнительный вид продукции — кефир обезжиренный. Цена — 770 руб./ц. Затраты сырого молока — 0,3 ц/ц. Производительность фасовки на оборудовании для фасовки молока и кефира — 6 ц/ч. Суммарный максимальный выпуск кефира обоих видов — 40 ц/сут. (минимальный выпуск кефира жирного не регламентируется).

10. Дополнительный вид продукции — творожные сырки. Цена — 7500 руб./ц. Затраты сырого молока — 16 ц/ц. Производительность фасовки на оборудовании для фасовки творожных сырков — 0,25 ц/ч. Оборудование может работать не более 17 ч./сут.

11. Дополнительный вид продукции — йогурт. Цена — 2200 руб./ц. Затраты сырого молока — 0,9 ц/ц. Производительность фасовки на оборудовании для фасовки сметаны — 0,2 ц/ч. Максимальный выпуск — 20 ц/сут.

12. Дополнительный вид продукции — кефир обезжиренный. Цена — 790 руб./ц. Затраты сырого молока — 0,31 ц/ц. Производительность фасовки на оборудовании для фасовки молока и кефира — 6 ц/ч. Доля обезжиренного кефира в общем производстве кефира должна составлять не менее трети.

13. Дополнительный вид продукции — творожные сырки. Цена — 7200 руб./ц. Затраты сырого молока — 15 ц/ц. Производительность фасовки на оборудовании для фасовки творожных сырков — 0,2 ц/ч. Оборудование может работать не более 16 ч./сут.

14. Дополнительный вид продукции — йогурт. Цена — 2000 руб./ц. Затраты сырого молока — 0,8 ц/ц. Производительность фасовки на оборудовании для фасовки сметаны — 0,2 ц/ч. Максимальный выпуск — 12 ц/сут.

15. Дополнительный вид продукции — творог. Цена — 5300 руб./ц. Затраты сырого молока — 17 ц/ц. Производительность фасовки на оборудовании для фасовки творога — 0,4 ц/ч. Оборудование может работать не более 16 ч./сут.

16. Дополнительный вид продукции — творожные сырки. Цена — 7500 руб./ц. Затраты сырого молока — 16 ц/ц. Производительность фасовки на оборудовании для фасовки творожных сырков — 0,25 ц/ч. Оборудование может работать не более 17 ч./сут.

# Литература


1. Рамазанов М.Д. Лекции по теории линейного программирования. Уфа: ИМВЦ УНЦ РАН. 2005. 66 с.

2. Светлов Н.М. Задания и методические указания к лабораторным работам по курсу «Экономико-математическое моделирование» для студентов бакалавриата по направлению «Менеджмент», обучающихся в сельскохозяйственных образовательных учреждениях высшего профессионального образования. М.: Издательство ФГОУ ВПО РГАУ-МСХА имени К.А. Тимирязева. 2009. 78 с.