

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ
Федеральное государственное автономное образовательное
учреждение высшего образования
«ЮЖНЫЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

Институт математики, механики и компьютерных наук
им. И. И. Воровича

Линейное программирование в примерах и задачах

Учебно-методическое пособие

Ростов-на-Дону – 2022

УДК 519.85

ББК 22.1

Печатается по решению кафедры высшей математики и исследования операций Южного федерального университета (протокол №8 от 4.10.2022)

Рецензенты:

д. ф.-т. н., проф., заведующий кафедры высшей математики и исследования операций Южного федерального университета
Белявский Григорий Исаакович

д.ф.-м.н., проф., заведующий кафедрой прикладной математики и программирования Южного федерального университета
Угольников Геннадий Анатольевич

Составитель: к.ф.-м.н. Землякова И. А.

Линейное программирование в примерах и задачах: учебно-методическое пособие / И. А. Землякова; Южный федеральный университет. – Ростов-на-Дону, 2022. - 81 с.

Учебно-методическое пособие предназначено для студентов бакалавриата, обучающихся по направлению 01.03.02 Прикладная математика и информатика, студентов других специальностей, изучающих курс «Методы оптимизации и исследование операций». Пособие содержит систематическое изложение раздела «Линейное программирование», снабжённое теоретическим материалом, множеством примеров и заданий для самостоятельного закрепления материала. Целью данного пособия является подробное изучение такого раздела математического программирования, как линейное программирование.

УДК 519.85

ББК 22.1

© Составитель: Землякова И.А., 2022

© Южный федеральный университет, 2022

СОДЕРЖАНИЕ

1. Постановка задачи линейного программирования.....	4
2. Примеры классических задач линейного программирования	4
<i>Пример 1. Задача о диете</i>	4
<i>Пример 2. Задача об оптимальном использовании ресурсов</i>	6
3. Виды и формы задач линейного программирования	7
4. Основные определения и теоремы.....	9
5. Графический метод решения задачи линейного программирования.....	10
Особые случаи решения задачи линейного программирования в примерах	15
6. Симплекс-метод решения задачи линейного программирования.	19
Геометрическая интерпретация симплекс-метода	19
Алгоритм симплекс-метода решения задачи линейного программирования	19
7. Метод искусственного базиса решения задачи линейного программирования	26
8. Двойственность в линейном программировании	32
Основные теоремы двойственности и их применение в задачах.....	35
9. Параметрическое линейное программирование	43
10. Транспортная задача	48
Постановка транспортной задачи.....	48
Построение начального опорного плана	50
Метод потенциалов. Алгоритм.....	54
Решение транспортной задачи (метод потенциалов). Примеры	55
10. Решение задачи линейного программирования с помощью системы компьютерной алгебры Maple	64
11. Решение транспортной задачи с помощью системы компьютерной алгебры Maple.....	67
12. Решение задачи линейного программирования на Python	69
13. Решение транспортной задачи на Python	70
Задания для самостоятельного решения.....	72
СПИСОК ЛИТЕРАТУРЫ.....	80

1. Постановка задачи линейного программирования

Линейное программирование (ЛП) представляет собой раздел математического программирования. **Задача математического программирования** заключается в отыскании экстремальных значений функции (целевая функция или функция цели) среди множества её допустимых значений, которое задаётся рядом ограничений. Например:

$$\begin{aligned} f(x, y) &\rightarrow \max \\ g(x, y) &\leq C \\ x, y &\geq 0 \end{aligned}$$

В задаче линейного программирования (ЗЛП) как целевая функция, так и функции, входящие в ограничения (равенства или неравенства), являются линейными.

2. Примеры классических задач линейного программирования

Рассмотрим несколько классических примеров задач линейного программирования. Начнём с «Задачи о диете». Приведём одну из возможных её формулировок.

Пример 1. Задача о диете

По медицинским показаниям человеку прописали придерживаться определённой суточной диеты и следить за тем, чтобы количество питательных веществ в продуктах соответствовало медицинским требованиям. Есть три вида продуктов П1, П2, П3, которые содержат определённые питательные вещества ПИТ1 и ПИТ2. Известно содержание питательных веществ в данных продуктах (мг) (a_{ij} , где i отвечает за i – питательное вещество, j – за j – й продукт, $i = 1, 2$; $j = 1, 2, 3$) и минимальная потребность организма в питательных веществах (мг) ($b_i, i = 1, 2$). Также известна стоимость единицы продукта каждого вида ($c_j, j = 1, 2, 3$). Требуется составить план питания человека следующим образом: все медицинские требования должны быть учтены и стоимость продуктовой корзины должна быть наименьшей. То есть определить, в каком количестве следует приобретать тот или иной продукт с учётом имеющихся ограничений. Оформим условие задачи в виде таблицы:

	П1	П2	П3	Минимальная потребность
ПИТ1	a_{11}	a_{12}	a_{13}	b_1
ПИТ2	a_{21}	a_{22}	a_{23}	b_2
Стоимость единицы продукта	c_1	c_2	c_3	

Табл. 1. Условие «Задачи о диете»

Построим математическую модель данной задачи. Пусть x_1, x_2, x_3 – искомые количества продуктов. Тогда стоимость продуктового набора можно рассчитать следующим образом:

$$c_1x_1 + c_2x_2 + c_3x_3$$

Введём в рассмотрение функцию $f(x_1, x_2, x_3) = c_1x_1 + c_2x_2 + c_3x_3$ (это и есть целевая функция или функция цели). Цель – минимизировать данную функцию, то есть получить наименьшую цену покупки продуктов.

При этом возникают ограничения, связанные с наличием минимальной потребности в соответствующих питательных веществах. Для первого питательного вещества его суммарное содержание во всех продуктах должно быть больше минимальной потребности: $a_{11}x_1 + a_{12}x_2 + a_{13}x_3 \geq b_1$. Аналогично для второго питательного вещества $a_{21}x_1 + a_{22}x_2 + a_{23}x_3 \geq b_2$. Также добавляются ограничения на $x_i, i = 1, 2, 3$. Исходя из смысла нашего обозначения, $x_i \geq 0, i = 1, 2, 3$.

Таким образом, получена следующая математическая модель поставленной задачи:

$$\begin{aligned}
 f(x_1, x_2, x_3) &= c_1x_1 + c_2x_2 + c_3x_3 \rightarrow \min \\
 a_{11}x_1 + a_{12}x_2 + a_{13}x_3 &\geq b_1 \\
 a_{21}x_1 + a_{22}x_2 + a_{23}x_3 &\geq b_2 \\
 x_i &\geq 0, i = 1, 2, 3
 \end{aligned}$$

Заметим, что как целевая функция, так и ограничения, являются линейными функциями. Значит, математической моделью «задачи о диете» является задача линейного программирования.

Второй пример связан с применением задачи линейного программирования в экономике. Стоит отметить, что задача линейного программирования нашла

широкое применение в данной области и представлена, например, такими задачами как транспортная задача, задача об оптимальном использовании ресурсов и так далее. В качестве примера рассмотрим задачу об оптимальном использовании ресурсов.

Пример 2. Задача об оптимальном использовании ресурсов

Фабрика производит три вида продукции: П1, П2, П3 (например, мебельная фабрика может производить стулья, столы, диваны). Для их изготовления затрачиваются три разных вида ресурсов: Р1, Р2, Р3 (например, древесина, ткань, электроэнергия). Известны запасы ресурсов для данной фабрики b_1, b_2, b_3 , где $b_i, i = 1, 2, 3$ – запас i -го вида ресурса. Также известно количество единиц каждого вида сырья, требуемое для производства каждого вида продукции $a_{ij}, i, j = 1, 2, 3$, то есть количество единиц i -го вида ресурса, требуемое для производства j -го вида продукции. Прибыль с продажи j -го вида продукции известна и равна $c_j, j = 1, 2, 3$. Требуется составить план выпуска продукции, позволяющий максимизировать прибыль предприятия с учётом имеющихся ограничений на запасы ресурсов.

Оформим условие задачи в виде таблицы:

	П1	П2	П3	Запасы ресурсов
Р1	a_{11}	a_{12}	a_{13}	b_1
Р2	a_{21}	a_{22}	a_{23}	b_2
Р3	a_{31}	a_{32}	a_{33}	b_3
Прибыль с продажи	c_1	c_2	c_3	

Табл. 2. Условие «Задачи об оптимальном использовании ресурсов»

Построим математическую модель данной задачи. Пусть x_1, x_2, x_3 – искомые количества единиц выпускаемой продукции. Тогда прибыль предприятия будет иметь следующий вид:

$$c_1x_1 + c_2x_2 + c_3x_3$$

Введём в рассмотрение функцию $f(x_1, x_2, x_3) = c_1x_1 + c_2x_2 + c_3x_3$ (функция цели). Цель – максимизировать данную функцию, то есть получить наибольшую прибыль.

При этом возникают условия, связанные с ограниченностью ресурсов. Для первого типа ресурса его суммарный расход не должен превосходить его запас: $a_{11}x_1 + a_{12}x_2 + a_{13}x_3 \leq b_1$. Аналогично для второго и третьего типов ресурсов $a_{21}x_1 + a_{22}x_2 + a_{23}x_3 \leq b_2$ и $a_{31}x_1 + a_{32}x_2 + a_{33}x_3 \leq b_3$. Также добавляются ограничения на $x_i, i = 1, 2, 3$. Исходя из смысла нашего обозначения, $x_i \geq 0, i = 1, 2, 3$.

Таким образом, получена следующая математическая модель поставленной задачи:

$$\begin{aligned} f(x_1, x_2, x_3) &= c_1x_1 + c_2x_2 + c_3x_3 \rightarrow \max \\ a_{11}x_1 + a_{12}x_2 + a_{13}x_3 &\leq b_1 \\ a_{21}x_1 + a_{22}x_2 + a_{23}x_3 &\leq b_2 \\ a_{31}x_1 + a_{32}x_2 + a_{33}x_3 &\leq b_3 \\ x_i &\geq 0, i = 1, 2, 3 \end{aligned}$$

Заметим, что как целевая функция, так и ограничения, являются линейными функциями. Значит, математической моделью «задачи об оптимальном использовании ресурсов» является задача линейного программирования.

3. Виды и формы задач линейного программирования

Общий вид задачи линейного программирования. В общей постановке задача линейного программирования имеет следующий вид. Пусть $x = (x_1, x_2, \dots, x_n)$ – вектор неизвестных переменных; $f(x_1, x_2, \dots, x_n) = f(x)$ – функция цели (целевая функция). Задача заключается в нахождении оптимального значения (максимального или минимального) целевой функции $f(x) = \sum_{j=1}^n c_jx_j$ при наличии ограничений:

$$\begin{aligned} \sum_{j=1}^n a_{ij}x_j &\leq b_i, i = 1, \dots, m_1 \\ \sum_{j=1}^n a_{ij}x_j &\geq b_i, i = m_1 + 1, \dots, m_2 \\ \sum_{j=1}^n a_{ij}x_j &= b_i, i = m_2 + 1, \dots, m \\ x_j &\geq 0, j = 1, \dots, n_1 \end{aligned}$$

$$x_j - \forall, j = n_1 + 1, \dots, n$$

где a_{ij}, b_i, c_j ($i = 1, \dots, m; j = 1, \dots, n$) – действительные числа.

Симметричный вид задачи линейного программирования.

$$f(x) = \sum_{j=1}^n c_j x_j \rightarrow \max$$

$$\sum_{j=1}^n a_{ij} x_j \leq b_i, \quad i = 1, \dots, m$$

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

ИЛИ

$$f(x) = \sum_{j=1}^n c_j x_j \rightarrow \min$$

$$\sum_{j=1}^n a_{ij} x_j \geq b_i, \quad i = 1, \dots, m$$

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

Канонический вид задачи линейного программирования.

$$f(x) = \sum_{j=1}^n c_j x_j \rightarrow \max$$

$$\sum_{j=1}^n a_{ij} x_j = b_i, \quad i = 1, \dots, m$$

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

Любую задачу линейного программирования можно представить в одной из заданных форм. Подробнее об элементарных преобразованиях, позволяющих осуществить данный переход, см. в пункте «Алгоритм симплекс-метода решения задачи линейного программирования».

Также зачастую бывает полезно каноническую форму задачи линейного программирования записывать в **матричной** форме. Для этого введём в рассмотрение вектор-строку коэффициентов целевой функции $c = (c_1, c_2, \dots, c_n)$, вектор-столбец свободных членов ограничений $b =$

$(b_1, b_2, \dots, b_m)$, вектор-строку переменных $x = (x_1, x_2, \dots, x_n)$, матрицу системы ограничений A . Тогда задача линейного программирования примет вид:

$$f(x) = cx \rightarrow \max$$

$$Ax = b$$

$$x \geq 0$$

4. Основные определения и теоремы

Рассмотрим задачу линейного программирования, например, в матричной форме:

$$f(x) = cx \rightarrow \max$$

$$Ax = b$$

$$x \geq 0$$

Определение 1. Вектор $x = (x_1, x_2, \dots, x_n)$, удовлетворяющий ограничениям задачи, называется **допустимым решением** (допустимым планом).

Определение 2. Множество допустимых решений задачи называется **областью допустимых решений (ОДР)** или многогранником решений.

Определение 3. Допустимое решение, доставляющее оптимальное значение целевой функции, называется **оптимальным решением** и обозначается x^* .

Решить задачу линейного программирования – это означает найти оптимальное решение x^* и оптимальное значение целевой функции, то есть $f(x^*)$.

Теорема 1. Множество допустимых решений задачи представляет собой выпуклый многогранник.

Теорема 2. Если решение задачи линейного программирования единственно, то оно совпадает с одной из угловых точек многогранника решений.

Теорема 3. Если решение задачи линейного программирования не единственно, то есть им является сразу несколько угловых точек, то целевая функция принимает оптимальное значение в каждой точке, представляющей выпуклую комбинацию данных угловых.

Теорема 4. Если область допустимых решений ограничена, то задача линейного программирования имеет или единственное решение или бесконечно много решений. В случае неограниченной области допустимых

решений задача будет иметь решение, если целевая функция ограничена сверху (снизу) для задачи максимизации (минимизации).

5. Графический метод решения задачи линейного программирования

Графический метод решения задачи линейного программирования применим лишь в случае двух переменных. В случае трёх переменных данный метод совершенно неудобен, а в случае числа переменных, не меньших, чем три, невозможен.

Рассмотрим задачу линейного программирования:

$$f(x) = c_1x_1 + c_2x_2 \rightarrow \max (\min)$$

$$a_{11}x_1 + a_{12}x_2 \leq b_1$$

$$a_{21}x_1 + a_{22}x_2 \leq b_2$$

$$\dots \dots \dots$$

$$a_{m1}x_1 + a_{m2}x_2 \leq b_m$$

$$x_1 \geq 0, x_2 \geq 0$$

Конечно, ограничения могут быть также и вида "=" или " $\geq$ ", а условие неотрицательности переменных может и не выполняться. На общий ход решения это не влияет и подробно будет рассмотрено в примерах.

Алгоритм решения задачи линейного программирования графическим методом.

1. Построить область допустимых решений (ОДР)

Для построения ОДР каждое неравенство-ограничение заменяем равенством. Так, например, первое ограничение примет вид $a_{11}x_1 + a_{12}x_2 = b_1$. Это линейная функция, график – прямая, строим её. Построение производим в системе координат, где осью абсцисс является ось Ox_1 , осью ординат – ось Ox_2 . Прямая разбивает плоскость на две полуплоскости, одна из которых и содержит решения исходного неравенства. Чтобы понять, какая именно это полуплоскость, выберем любую точку плоскости, не лежащую на данной прямой, например, $O(0,0)$ (если прямая не проходит через начало координат), и подставим в неравенство. Если получилось верное неравенство, то была выбрана точка из нужной полуплоскости, в которой лежат решения. Её и заштриховываем. Если же при подстановке получилось неверное неравенство, то искомая полуплоскость лежит по другую сторону от выбранной точки. Изобразим это графически (рис. 1):

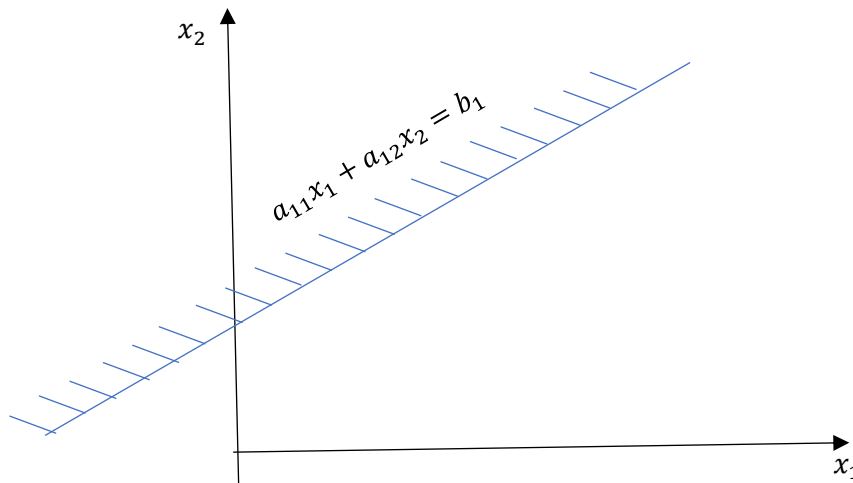


Рис. 1. Выделение полуплоскости для одной прямой

Продолжим аналогичные построения для оставшихся ограничений и получим область, состоящую из точек, удовлетворяющих всем ограничениям. Это и есть область допустимых решений (ОДР). Стоит отметить, что данная область может являться пустой; состоять из одной точки; являться выпуклым многоугольником; являться неограниченным многоугольником. Подробнее об этом будет рассказано в пункте «Особые случаи решения задачи линейного программирования в примерах».

2. Построение градиента целевой функции и линий уровня

Начнём с построения градиента целевой функции. Исходя из геометрического смысла производной, установим, что частные производные целевой функции $c_1 = \frac{\partial f(x)}{\partial x_1}$ и $c_2 = \frac{\partial f(x)}{\partial x_2}$ указывают скорость возрастания функции вдоль координатных осей. Соответственно, вектор, координаты которого представляют собой данные частные производные, будет указывать направление возрастания функции. Этот вектор назовём градиентом целевой функции и обозначим его $\bar{g} = \{c_1; c_2\}$. Таким образом, для нахождения максимума целевой функции следует идти в направлении градиента; для нахождения минимума – в направлении, обратном градиенту, то есть в направлении антиградиента $-\bar{g} = \{-c_1; -c_2\}$. Строим градиент следующим образом: начало вектора совпадает с началом координат, конец вектора находится в точке с координатами $(c_1; c_2)$.

Теперь введём в рассмотрение линии уровня. Возьмём целевую функцию $f(x) = c_1x_1 + c_2x_2$ и приравняем её некоторой константе $c_1x_1 + c_2x_2 = C_1$. Получилась прямая, которую называют линия уровня. На данной прямой целевая функция принимает постоянное значение. Изменяя значение C_1 , получаем семейство параллельных прямых. Градиент целевой функции,

введённый ранее, является нормальным вектором и перпендикулярен линиям уровня.

3. Определение оптимального решения и оптимального значения целевой функции

Таким образом, если идти в направлении градиента, перемещая линию уровня параллельно самой себе, значение целевой функции будет увеличиваться. Перемещение линии уровня в направлении антиградиента приведёт к уменьшению значения целевой функции. Оптимальное значение целевой функции будет достигаться в крайних точках касания линии уровня и ОДР.

Пример 1. Решить задачу линейного программирования графическим методом:

$$\begin{aligned} f(x) &= 3x_1 + 3x_2 \rightarrow \max \\ x_1 + 2x_2 &\leq 8 \\ 2x_1 - x_2 &\geq 1 \\ x_1 - 2x_2 &\leq 2 \\ x_1 &\geq 0, x_2 &\geq 0 \end{aligned}$$

Строим область допустимых решений. Заметим сразу, что $x_1 \geq 0, x_2 \geq 0$, то есть построения достаточно проводить лишь в первой четверти.

Рассмотрим первое ограничение $x_1 + 2x_2 \leq 8$, перейдём к равенству $x_1 + 2x_2 = 8$. Построим данную прямую. Прямая разделила координатную плоскость на две полуплоскости, покажем штриховкой ту полуплоскость, которую определяет исходное неравенство. Для этого выберем любую точку плоскости, например $O(0; 0)$ и подставим её координаты в первое ограничение. Получим $0 + 0 \leq 8$, это верное неравенство, значит $O(0; 0)$ лежит в нужной полуплоскости. Покажем её штриховкой.

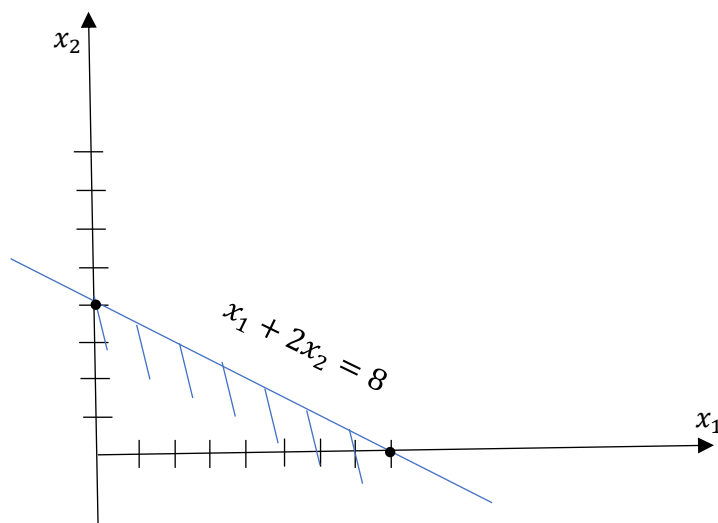


Рис. 2. Графическое изображение первого ограничения системы

Аналогично изобразим оставшиеся ограничения и получим ОДР:

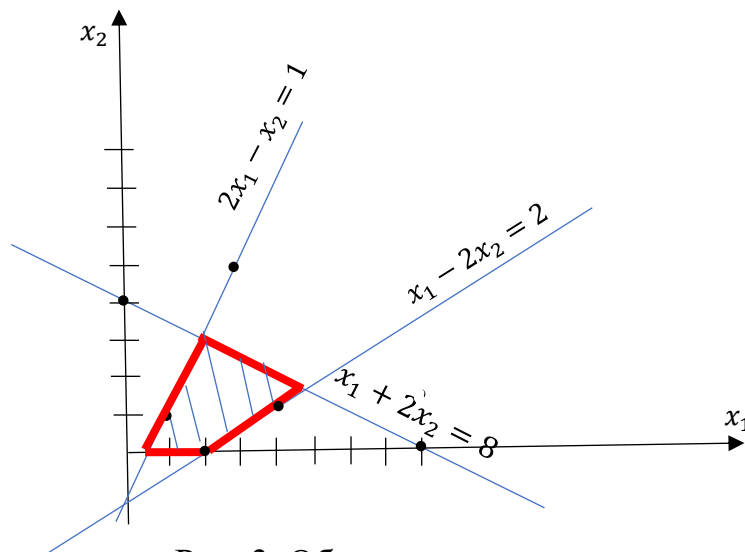


Рис. 3. Область допустимых решений

Построим градиент $\bar{g} = \{3; 3\}$ (вектор с началом в точке $O(0; 0)$ и концом в точке $(3; 3)$) и линию уровня $f(x) = 3x_1 + 3x_2 = 0$. Данная линия уровня проходит через начало координат и перпендикулярна градиенту. Так как в предложенной задаче требуется максимизировать целевую функцию, то идём в направлении градиента, перемещая линию уровня параллельно исходной $3x_1 + 3x_2 = 0$ до тех пор, пока не получим крайнее касание с ОДР. По Рис. 4 видно, что крайнее касание происходит в точке A , значит, её координаты и будут являться оптимальным решением (целевая функция принимает максимальное значение в данной точке).

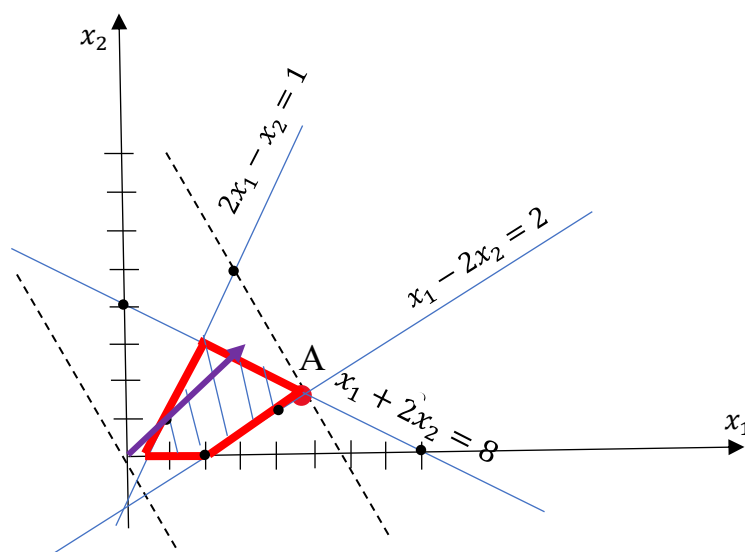


Рис. 4. Оптимальное решение (максимум целевой функции)

Рассчитаем координаты данной точки как точку пересечения прямых $x_1 + 2x_2 = 8$ и $x_1 - 2x_2 = 2$. Получим оптимальное решение (максимум целевой функции) $x^* = (5; 1.5)$. Оптимальное значение целевой функции $f(x^*) = 3 \cdot 5 + 3 \cdot 1.5 = 19.5$.

Ответ: оптимальное решение $x^* = (5; 1.5)$; оптимальное значение целевой функции $f(x^*) = 19.5$.

Пример 2. Решить задачу линейного программирования графическим методом:

$$f(x) = 3x_1 + 3x_2 \rightarrow \min$$

$$x_1 + x_2 \leq 8$$

$$2x_1 - x_2 \geq 1$$

$$x_1 - 2x_2 \leq 2$$

$$x_1 \geq 0, x_2 \geq 0$$

Начинаем решение с построения ОДР (Рис. 5). Отличие от предыдущего примера в том, что целевая функция минимизируется, то есть строим антиградиент $-\bar{g} = \{-3; -3\}$ и идём в направлении антиградиента, перемещая линию уровня параллельно исходной до тех пор, пока не получим крайнее касание с ОДР. По Рис. 5 видно, что крайнее касание происходит в точке В, значит, её координаты и будут являться оптимальным решением (целевая функция принимает минимальное значение в данной точке).

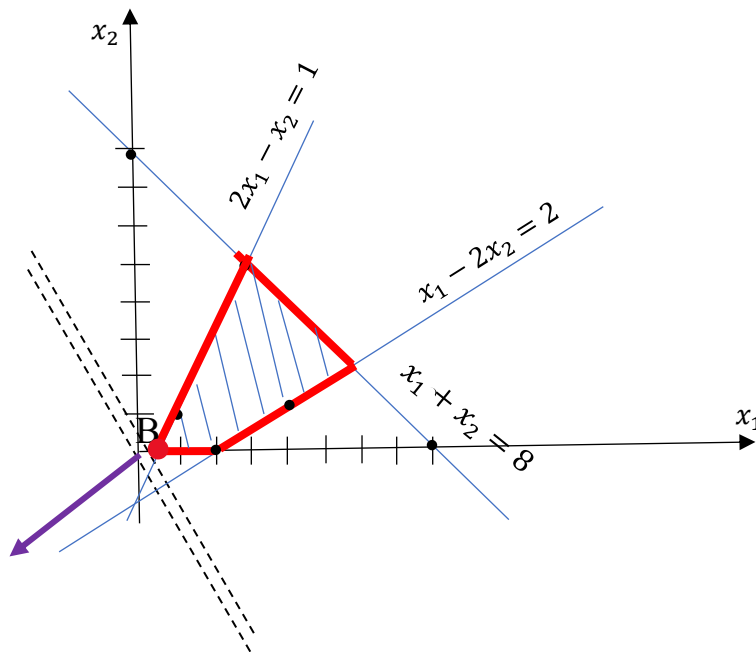


Рис. 5. Оптимальное решение (минимум целевой функции)

Рассчитаем координаты данной точки как точку пересечения прямой $2x_1 - x_2 = 1$ и оси Ox_1 . Получим оптимальное решение (минимум целевой функции) $x^* = (0.5; 0)$. Оптимальное значение целевой функции $f(x^*) = 3 \cdot 0.5 + 3 \cdot 0 = 1.5$.

Ответ: оптимальное решение $x^* = (0.5; 0)$; оптимальное значение целевой функции $f(x^*) = 1.5$.

Особые случаи решения задачи линейного программирования в примерах

В данном пункте на примерах будут разобраны различные случаи графического решения задачи линейного программирования, отличные от Примера 1 и Примера 2 (в которых ОДР – замкнутый многоугольник и оптимальное решение единственно).

Пример 3. Система ограничений несовместна. ОДР представляет собой пустое множество. Задача линейного программирования не имеет решения (Неразрешимость I).

Решить задачу линейного программирования графическим методом:

$$f(x) = 3x_1 + 3x_2 \rightarrow \max$$

$$x_1 + x_2 \leq 8$$

$$2x_1 - x_2 \leq 1$$

$$x_1 - 2x_2 \geq 2$$

$$x_1 \geq 0, x_2 \geq 0$$

Построим область допустимых решений.

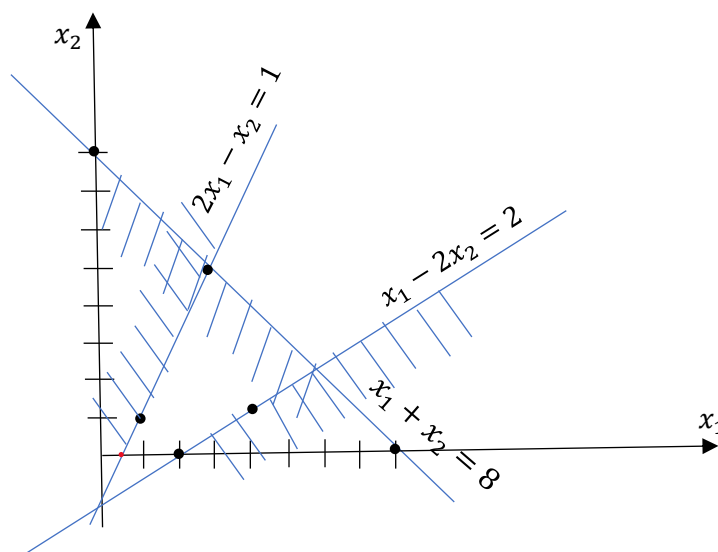


Рис. 6. Область допустимых решений

Видно, что ОДР представляет собой пустое множество. Значит, задача линейного программирования не имеет решения (Неразрешимость 1).

Ответ: нет решения (Неразрешимость 1).

Пример 4. ОДР представляет собой одну точку, которая и будет являться оптимальным решением (как максимумом, так и минимумом).

Решить задачу линейного программирования графическим методом:

$$f(x) = 3x_1 + 3x_2 \rightarrow \max$$

$$x_1 + x_2 = 8$$

$$2x_1 - x_2 = 1$$

$$x_1 - 2x_2 \leq 2$$

$$x_1 \geq 0, x_2 \geq 0$$

Построим область допустимых решений.

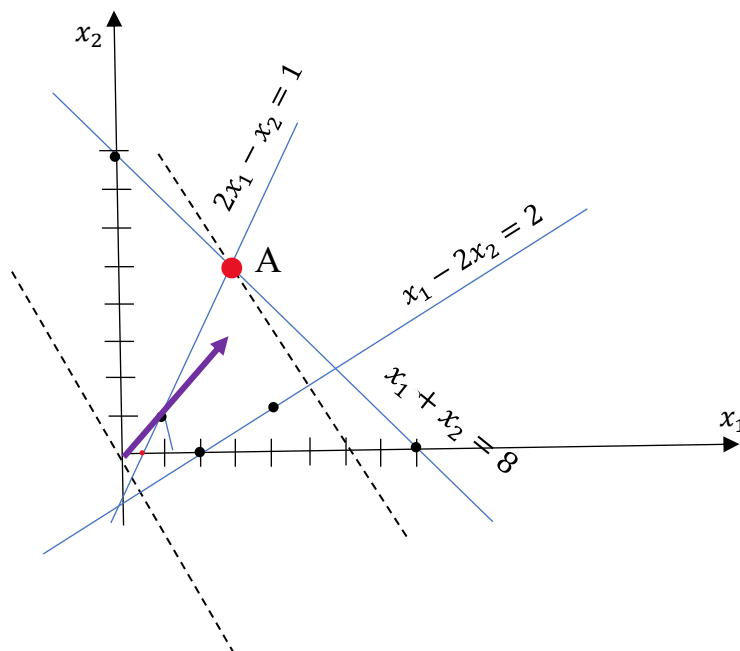


Рис. 7. Оптимальное решение (максимум целевой функции)

Видно, что ОДР состоит из одной точки – точки A. Значит, целевая функция будет принимать оптимальное значение именно в ней. Рассчитаем координаты данной точки как точку пересечения прямых $x_1 + x_2 = 8$ и $2x_1 - x_2 = 1$. Получим оптимальное решение (максимум целевой функции) $x^* = (3; 5)$. Оптимальное значение целевой функции $f(x^*) = 3 \cdot 3 + 3 \cdot 5 = 24$.

Ответ: оптимальное решение $x^* = (3; 5)$; оптимальное значение целевой функции $f(x^*) = 24$.

Пример 5. Задача имеет бесконечное множество решений (отрезок многоугольника решений). Решением является линейная комбинация точек данного отрезка.

Решить задачу линейного программирования графическим методом:

$$f(x) = 3x_1 + 3x_2 \rightarrow \max$$

$$x_1 + x_2 \leq 8$$

$$2x_1 - x_2 \geq 1$$

$$x_1 - 2x_2 \leq 2$$

$$x_1 \geq 0, x_2 \geq 0$$

Построим область допустимых решений. Градиент $\bar{g} = \{3; 3\}$, идём в его направлении, перемещая линию уровня параллельно исходной до тех пор, пока не получим крайнее касание с ОДР. По Рис. 8 видно, что крайнее касание происходит во всех точках отрезка AB (если это неочевидно, то можно найти координаты точки A , координаты точки B , подставить в целевую функцию и убедиться, что функция принимает одинаковые значения в данных точках). Значит, задача имеет бесконечное множество решений, состоящее из всех точек отрезка AB .

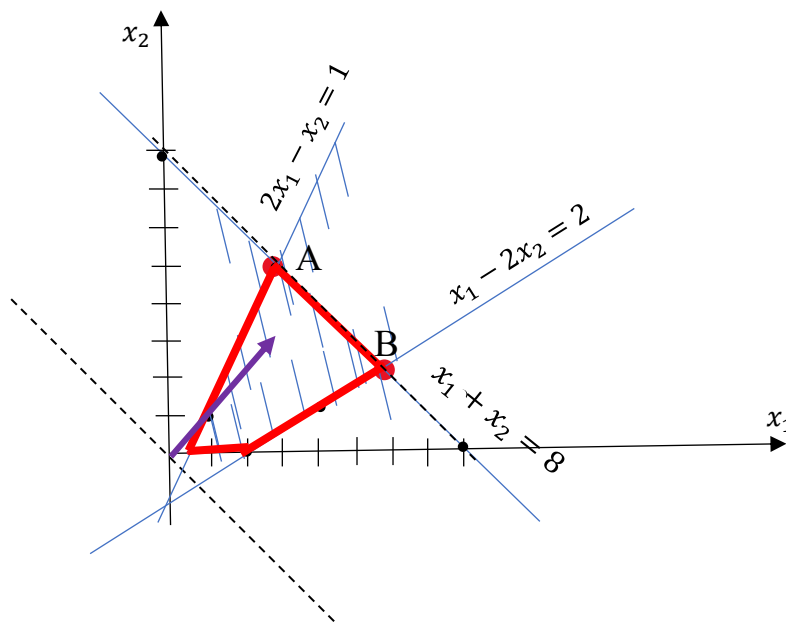


Рис. 8. Оптимальное решение (максимум целевой функции)

Общее решение представляет собой выпуклую комбинацию точек отрезка AB : $x^* = tA + (1 - t)B, t \in [0, 1]$.

Вычислим координаты точек A и B . Точка A является точкой пересечения прямых $x_1 - 2x_2 = 2$ и $x_1 + x_2 = 8$; $A(6; 2)$. Точка B является точкой пересечения прямых $2x_1 - x_2 = 1$ и $x_1 + x_2 = 8$; $B(3; 5)$. Тогда общее решение будет иметь вид: $x^* = t \cdot \begin{pmatrix} 6 \\ 2 \end{pmatrix} + (1 - t) \begin{pmatrix} 3 \\ 5 \end{pmatrix} = \begin{pmatrix} 3t+3 \\ -3t+5 \end{pmatrix}, t \in [0, 1]$; $f(x^*) = 3 \cdot (3t + 3) + 3 \cdot (-3t + 5) = 24$.

Ответ: оптимальное решение $x^* = \begin{pmatrix} 3t+3 \\ -3t+5 \end{pmatrix}, t \in [0,1]$; оптимальное значение целевой функции $f(x^*) = 24$.

Пример 6. Целевая функция не ограничена сверху. Задача линейного программирования не имеет решения (Неразрешимость 2).

Решить задачу линейного программирования графическим методом:

$$f(x) = 3x_1 + 3x_2 \rightarrow \max$$

$$2x_1 - x_2 \geq 1$$

$$x_1 - 2x_2 \leq 2$$

$$x_1 \geq 0, x_2 \geq 0$$

Построим область допустимых решений.

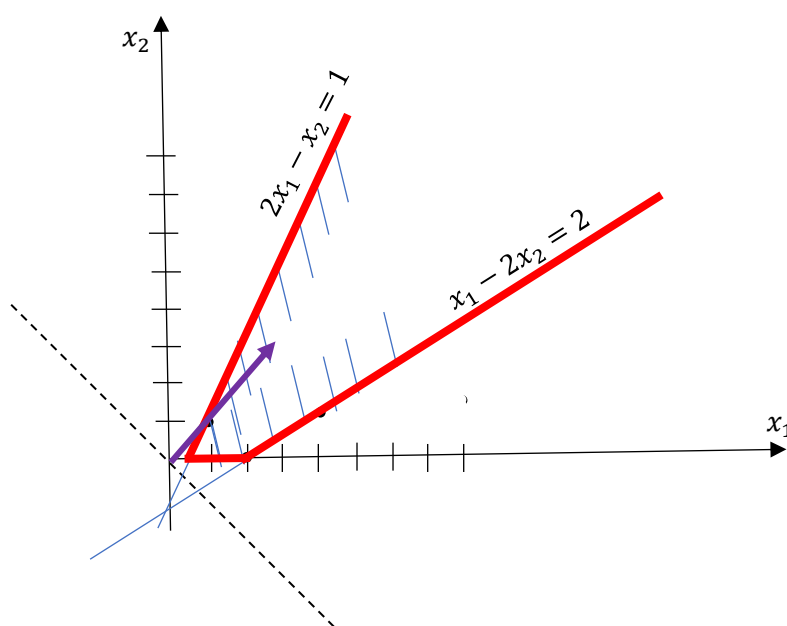


Рис. 9. Оптимальное решение (максимум целевой функции)

По Рис. 9 видно, что ОДР является неограниченной областью. Градиент $\bar{g} = \{3; 3\}$, идём в его направлении, перемещая линию уровня параллельно исходной до тех пор, пока не получим крайнее касание с ОДР. Такое касание не произойдёт, то есть целевая функция не ограничена сверху. Значит, задача линейного программирования не имеет решения (Неразрешимость 2).

Ответ: нет решения (Неразрешимость 2).

Таким образом, графический метод решения задачи линейного программирования довольно прост и удобен, но не универсален – им пользуются лишь в случае двух переменных x_1 и x_2 . Универсальным методом является симплекс-метод, речь о котором пойдёт дальше.

6. Симплекс-метод решения задачи линейного программирования.

Геометрическая интерпретация симплекс-метода

Из Теоремы 2 следует, что решение ЗЛП можно найти следующим образом: перебрать все вершины многогранника решений, вычислить значения целевой функции в них, выбрать вершину, в которой функция принимает оптимальное значение (максимум или минимум). Но данный приём является крайне сложным с вычислительной точки зрения, ведь происходит простой перебор всех вершин, которых может быть достаточно много, а то, насколько каждая последующая вершина улучшает оптимальное решение, не берётся во внимание.

Более оптимальным будет подход, в котором идёт не просто перебор всех вершин многогранника решений, а вершины целенаправленно перебираются так, чтобы каждое последующее решение было лучше (не хуже) предыдущего. Это позволит значительно сократить число шагов в решении. При этом на каждом последующем шаге значение целевой функции улучшается (не ухудшается).

Данная идея, основанная на последовательном улучшении решения, легла в основу универсального метода решения ЗЛП – симплекс-метода. Геометрический смысл симплекс-метода заключается в последовательном переходе от одной вершины многогранника (исходной) к соседней (в которой целевая функция должна принимать лучшее или нехудшее значение). Данный перебор осуществляется до тех пор, пока не будет найдено оптимальное решение.

Алгоритм симплекс-метода решения задачи линейного программирования

Симплекс-метод решения задачи линейного программирования позволяет за конечное число шагов найти оптимальное решение задачи линейного программирования или прийти к выводу, что решения не существует.

Для использования симплекс-метода задача линейного программирования обязательно должна быть приведена к канонической форме. Напомним вид ЗЛП в канонической форме:

$$f(x) = \sum_{j=1}^n c_j x_j \rightarrow \max$$

$$\sum_{j=1}^n a_{ij} x_j = b_i, \quad i = 1, \dots, m$$

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

Алгоритм решения ЗЛП симплекс-методом можно представить в виде следующих основных шагов:

1. Привести задачу к канонической форме
2. Построить начальную симплекс-таблицу
3. Если начальная симплекс-таблица не является оптимальной, переходим к построению новой таблицы с помощью метода Жордана-Гаусса. Данные переходы осуществляем до тех пор, пока текущая симплекс-таблица не будет оптимальной.

Опишем каждый из предложенных пунктов подробнее.

1. Привести задачу к канонической форме.

- 1.1. Если в исходной задаче присутствует ограничение вида $\sum_{j=1}^n a_{ij}x_j \leq b_i, b_i \geq 0, i = 1, \dots, m_1$, то добавим к левой части дополнительные (балансовые) переменные $x_{n+i} \geq 0, i = 1, \dots, m_1$ получим ограничение равенство:

$$\sum_{j=1}^n a_{ij}x_j + x_{n+i} = b_i, b_i \geq 0, x_{n+i} \geq 0, i = 1, \dots, m_1$$

- 1.2. Если в исходной задаче присутствует ограничение вида $\sum_{j=1}^n a_{ij}x_j \geq b_i, b_i \geq 0, i = 1, \dots, m_2$, то вычтем из левой части дополнительные (балансовые) переменные $x_{n+i} \geq 0, i = 1, \dots, m_2$ получим ограничение равенство:

$$\sum_{j=1}^n a_{ij}x_j - x_{n+i} = b_i, b_i \geq 0, x_{n+i} \geq 0, i = 1, \dots, m_2$$

- 1.3. Если в исходной задаче требуется минимизировать целевую функцию $f(x) = \sum_{j=1}^n c_jx_j \rightarrow \min$, то, умножив её на -1, получим задачу на нахождение максимума:

$$-f(x) = -\sum_{j=1}^n c_jx_j \rightarrow \max$$

- 1.4. Если в исходной задаче на некоторый $x_j, j = 1 \dots n$ не наложено условие неотрицательности, то представим $x_j = x_j^1 - x_j^2$, $x_j^1 \geq 0, x_j^2 \geq 0$.

Пример 7. Привести задачу линейного программирования к канонической форме:

$$f(x) = 3x_1 + 3x_2 \rightarrow \min$$

$$x_1 + x_2 \leq 8$$

$$2x_1 - x_2 \geq 1$$

$$x_1 - 2x_2 \leq 2$$

$$x_1 \geq 0, x_2 \geq 0$$

Целевая функция минимизируется, а в канонической форме ЗЛП следует искать максимум целевой функции, значит, умножим её на -1 и получим $-f(x) = -3x_1 - 3x_2 \rightarrow \max$.

Первое ограничение – ограничение-неравенство типа меньше или равно, прибавим дополнительную переменную x_3 , $x_3 \geq 0$ (x_1, x_2 в задаче присутствуют, следующая по нумерации переменная – x_3) и получим ограничение-равенство: $x_1 + x_2 + x_3 = 8$.

Второе ограничение – ограничение-неравенство типа больше или равно, вычтем дополнительную переменную x_4 , $x_4 \geq 0$ и получим ограничение-равенство: $2x_1 - x_2 - x_4 = 1$.

Третье ограничение аналогично первому, прибавляем x_5 , $x_5 \geq 0$ и получим ограничение-равенство: $x_1 - 2x_2 + x_5 = 2$.

Требование неотрицательности для всех $x_j, j = 1, \dots, 5$ выполняется.

Таким образом, получен канонический вид ЗЛП:

$$\begin{aligned} -f(x) &= -3x_1 - 3x_2 \rightarrow \max \\ x_1 + x_2 + x_3 &= 8 \\ 2x_1 - x_2 - x_4 &= 1 \\ x_1 - 2x_2 + x_5 &= 2 \\ x &\geq 0 \end{aligned}$$

2. Построить начальную симплекс-таблицу

- 2.1. Преобразовать все ограничение-равенства так, чтобы правая часть являлась неотрицательным числом (если в правой части находится отрицательное число, умножаем ограничение на -1).
- 2.2. Выписать матрицу ограничений. Если в ней присутствует единичная матрица, то разбить $x_j, j = 1, \dots, n$ на базисные (соответствующие данной матрице, вектор x^B), и небазисные (остальные, вектор x^N). Если единичной матрицы нет, то следует воспользоваться методом искусственного базиса (пункт «Метод искусственного базиса решения задачи линейного программирования»).
- 2.3. Положив базисную переменную равной правой части ограничения, а небазисную равной 0, получим опорное решение, которое будем улучшать.
- 2.4. Строим начальную симплекс-таблицу, она выглядит следующим образом:

	x^N	
x^B	A	b
f	Δ^N	$f(\tilde{x})$

Таблица 3. Общий вид симплекс-таблицы

где x^N – вектор небазисных переменных; x^B – вектор базисных переменных; A – столбцы матрицы ограничений, соответствующие небазисным переменным; b – вектор правых частей ограничений; Δ^N – строка оценок ($\Delta^N = c^B A - c^N$); $f(\tilde{x})$ – значение целевой функции в полученном опорном решении $\tilde{x}$.

3. Проверка симплекс-таблицы на оптимальность и преобразования Жордана-Гаусса

- 3.1. Если все оценки $\Delta^N \geq 0$, то данная таблица является оптимальной и можно выписывать оптимальное решение. Оптимальное решение строится следующим образом: полагаем $x^N = 0$, а x^B приравниваем правой части ограничений. Задача решена.
- 3.2. Если в симплекс-таблице есть столбец с отрицательной оценкой и в нём все элементы меньше или равны 0, то делаем вывод, что задача не имеет решения (целевая функция неограничена). Данный случай называется Неразрешимость 2.
- 3.3. Если среди оценок есть хотя бы одна нулевая, которая соответствует небазисной дополнительной переменной, а в столбце, содержащем этот нуль имеется хотя бы один положительный элемент, то задача имеет бесконечное множество решений.
- 3.4. Если есть столбцы с отрицательными оценками, то выбираем среди них разрешающий. Разрешающий столбец имеет наибольшую по модулю оценку.
- 3.5. В разрешающем столбце выбираем главный элемент. Для этого находим частные от деления правой части ограничений на положительные элементы столбца (симплексные отношения). В качестве главного элемента выбираем число, которому соответствует наименьшее симплексное отношение. Главный элемент определяет главную строку и главный столбец (находится на их пересечении).
- 3.6. Составляем новую симплекс-таблицу по следующим правилам (преобразования Жордана-Гаусса):

- поменять местами переменные, стоящие в главной строке и главном столбце;
- заменить главный элемент обратным (изменить число, являющееся главным элементом, на число, обратное ему);
- оставшиеся элементы главной строки разделить на главный элемент;
- оставшиеся элементы главного столбца разделить на главный элемент и изменить знаки на противоположные;
- остальные элементы новой симплекс-таблицы пересчитать по правилу, напоминающему правило вычисления определителей 2×2 , только в качестве главной диагонали выбирается та, на которой расположен главный элемент **и разделить на главный элемент**. В данный «определитель» обязательно входят главный и пересчитываемый элементы. Для удобства нахождения данного определителя можно выделить главный столбец, главную строку, строку и столбец, на пересечении которых находится пересчитываемый элемент. На пересечении выделенных строк и столбцов и находятся элементы, попадающие в данный «определитель».

Данные шаги симплекс-метода и переходы от одной симплекс-таблицы к другой выполняем до тех пор, пока не получим оптимальную симплекс-таблицу или убедимся, что решений нет.

Пример 8. Решить задачу линейного программирования симплекс-методом:

$$\begin{aligned}
 f(x) &= 2x_1 + x_2 \rightarrow \max \\
 -x_1 + 3x_2 &\leq 6 \\
 2x_1 + x_2 &\leq 9 \\
 x_1 - x_3 &\leq 3 \\
 x &\geq 0
 \end{aligned}$$

1. Приводим задачу к канонической форме. Так как присутствуют ограничения вида «меньше или равно», то прибавим к каждому из них соответствующую дополнительную переменную и получим искомую каноническую форму:

$$\begin{aligned}
 f(x) &= 2x_1 + x_2 \rightarrow \max \\
 -x_1 + 3x_2 + x_4 &= 6 \\
 2x_1 + x_2 + x_5 &= 9 \\
 x_1 - x_3 + x_6 &= 3 \\
 x &\geq 0
 \end{aligned}$$

2. Правые части ограничений представляют собой положительные числа, значит можно переходить к матрице ограничений (элементы которой представляют собой коэффициенты при соответствующем $x_i, i = 1, \dots, 6$:

$$A = \begin{pmatrix} x_1 & x_2 & x_3 & x_4 & x_5 & x_6 \\ -1 & 3 & 0 & 1 & 0 & 0 \\ 2 & 1 & 0 & 0 & 1 & 0 \\ 1 & 0 & -1 & 0 & 0 & 1 \end{pmatrix}$$

Обратим внимание, что столбцы, соответствующие переменным x_4, x_5, x_6 образуют единичную матрицу, то есть являются базисными $x^B = (x_4, x_5, x_6)$, тогда оставшиеся переменные x_1, x_2, x_3 являются небазисными $x^N = (x_1, x_2, x_3)$.

Строим начальную симплекс-таблицу. Для вычисления строки оценок удобно подписать рядом с $x_i, i = 1..6$ их коэффициенты из целевой функции ($f(x) = 2x_1 + x_2$), то есть коэффициент при x_1 равен 2, коэффициент при x_2 равен 1, остальные равны 0:

		2	1	0	
		x_1	x_2	x_3	b
0	x_4	-1	3	0	6
0	x_5	2	1	0	9
0	x_6	①	0	-1	3
	Δ^N	-2	-1	0	

Таблица 4. Начальная симплекс-таблица

Подробно распишем получение оценок. Оценка для первого столбца $\Delta_1^N = 0 \cdot (-1) + 0 \cdot 2 + 0 \cdot 1 - 2 = -2$; для второго столбца $\Delta_2^N = 0 \cdot 3 + 0 \cdot 1 + 0 \cdot 0 - 1 = -1$; для третьего столбца $\Delta_3^N = 0 \cdot 0 + 0 \cdot 0 + 0 \cdot (-1) - 0 = 0$.

3. Заметим, что среди оценок есть отрицательные, значит данная симплекс-таблица не является оптимальной. То есть решение можно улучшить. Среди отрицательных оценок выбираем большую по модулю, -2, она определяет главный столбец (столбец, соответствующий x_1). В данном столбце есть два положительных элемента - 2 и 1, для каждого из них находим симплексное отношение, то есть делим правую часть ограничения на соответствующий элемент и выбираем наименьшее частное. $9:2 = 4.5$; $3:1 = 3$, то есть наименьшее частное равно 3 и оно соответствует числу 1, значит 1 из первого столбца выбирается в качестве главного элемента. Выделим её. Переходим к новой симплекс-таблице.

4. Заполнение новой симплекс-таблицы. Меняем местами переменные, стоящие на пересечении главного элемента, то есть x_1 и x_6 . Главный элемент

заменяем обратным ему числом ($1:1 = 1$). Элементы главной строки делим на главный элемент. Элементы главного столбца делим на главный элемент и меняем знаки на противоположные. Все остальные элементы пересчитываем по правилу, напоминающему правило вычисления определителей 2×2 . Распишем подробно. Пересчитаем элемент, который находится на пересечении x_4 и x_2 . Выделим одним цветом главную строку и главный столбец (их определил на прошлом шаге главный элемент), другим цветом строку и столбец на пересечении которых находится пересчитываемый элемент:

		x_1	x_2
0	x_4	-1	3
0	x_5	2	1
0	x_6	1	0

Таблица 5. Часть таблицы для пересчёта элемента на пересечении x_4 и x_2

Элементы, находящиеся на пересечение указанных прямых и образуют тот самый «определитель»:

$$\begin{vmatrix} -1 & 3 \\ 1 & 0 \end{vmatrix}$$

Главная диагональ должна содержать главный элемент, покажем её:

$$\begin{vmatrix} -1 & 3 \\ 1 & 0 \end{vmatrix}$$

Пересчитываем элемент: $1 \cdot 3 - (-1) \cdot 0 = 3$. Не забываем разделить это число на главный элемент, то есть $3:1 = 3$. Записываем это число в новую симплекс-таблицу.

Аналогично пересчитываются все оставшиеся элементы. Этот пересчёт рекомендуется проделать самостоятельно и сверить свои результаты с результатами, приведёнными в таблице:

	x_6	x_2	x_3	b
x_4	1	3	-1	9
x_5	-2	1	2	3
x_1	1	0	-1	3
Δ^N	2	-1	-2	

Таблица 6. Следующий шаг симплекс-метода

Снова в строке оценок есть отрицательные числа, значит, проделываем ещё один шаг симплекс-метода (аналогично предыдущему). Теперь в качестве главного элемента выступает число 2 из столбца, соответствующего переменной x_3 , выделим его. Заменяем главный элемент ему обратным числом, то есть $\frac{1}{2}$, меняем x_3 и x_5 местами и пересчитываем таблицу аналогично предыдущей:

	x_6	x_2	x_5	b
x_4	0	$7/2$	$\frac{1}{2}$	$21/2$
x_3	-1	$\frac{1}{2}$	$\frac{1}{2}$	$3/2$
x_1	0	$\frac{1}{2}$	$\frac{1}{2}$	$9/2$
Δ^N	0	0	1	

Таблица 7. Следующий шаг симплекс-метода

В таблице 7 все оценки неотрицательны, значит, она является оптимальной и можно выписывать решение. x_1, x_3 – базисные переменные, приравниваем их правой части ограничения, $x_1 = \frac{9}{2}, x_3 = \frac{3}{2}$. x_2 – небазисная переменная, приравниваем её нулю, $x_2 = 0$. То есть оптимальное решение имеет вид $x^* = (\frac{9}{2}, 0, \frac{3}{2})$. Осталось найти оптимальное значение целевой функции. $f(x^*) = 2 \cdot \frac{9}{2} + 0 = 9$.

Ответ: оптимальное решение $x^* = (\frac{9}{2}, 0, \frac{3}{2})$, оптимальное значение целевой функции $f(x^*) = 9$.

7. Метод искусственного базиса решения задачи линейного программирования

Зачастую при решении задачи линейного программирования в матрице ограничений отсутствует единичная матрица, то есть исходного допустимого базиса нет. В этом случае используется метод искусственного базиса, который позволяет создать данный базис искусственным образом.

Алгоритм метода искусственного базиса.

1. Свести задачу к канонической форме
2. Убедиться, что правые части ограничений – положительные числа, если нет – умножить обе части равенства на -1.
3. Выписать матрицу ограничений. Убедиться, что в ней нет единичной матрицы.

4. Добавить в ограничения искусственные переменные $w_i, i = 1, \dots, k$ для создания базиса. Переходим к решению вспомогательной задачи.
5. Вспомогательная задача заключается в нахождении максимума целевой функции $F(w) = -\sum_{i=1}^k w_i$ при ограничениях из пункта 4.
6. Решаем вспомогательную задачу симплекс-методом. Цель решения состоит в следующем: превратить базисные переменные $w_i, i = 1, \dots, k$ в небазисные. Если в оптимальной симплекс-таблице это сделать не удалось (то есть оптимальное значение целевой функции вспомогательной задачи является отрицательным числом) и остались базисные искусственные переменные, то делаем вывод, что исходная задача не имеет решения (Неразрешимость 1).
7. Если все базисные переменные $w_i, i = 1, \dots, k$ превратились в небазисные, то есть оптимальное значение целевой функции вспомогательной задачи равно нулю, то используем полученную оптимальную таблицу вспомогательной задачи для перехода к начальной симплекс-таблице исходной задачи. Для этого вычёркиваем столбцы, соответствующие искусственным небазисным переменным, и пересчитываем оценки, используя целевую функцию исходной задачи. Построена начальная симплекс-таблица исходной задачи.
8. Решаем исходную задачу симплекс-методом.

Пример 9. Решить задачу линейного программирования методом искусственного базиса:

$$\begin{aligned}
 f(x) &= 2x_1 + 3x_2 \rightarrow \max \\
 x_1 + x_2 + x_3 + x_4 &= 1 \\
 x_1 - x_2 + x_3 - x_4 &= 1 \\
 x &\geq 0
 \end{aligned}$$

1. Заметим, что задача уже находится в канонической форме.
2. Правые части ограничений представляют собой положительные числа, значит можно переходить к матрице ограничений.
3. Выпишем матрицу ограничений, её элементы представляют собой коэффициенты при соответствующем $x_i, i = 1..4$:

$$A = \begin{pmatrix} x_1 & x_2 & x_3 & x_4 \\ 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 \end{pmatrix}$$

Заметим, что данная матрица не содержит в себе единичную, именно поэтому добавляем в ограничения искусственные переменные $w_i, i = 1, \dots, 2$ для создания базиса.

4. Для того, чтобы матрица ограничений содержала в себе единичную матрицу, следует добавить два столбца, как раз и являющихся единичной

матрицей. То есть в задаче будет две искусственные переменные w_1 и w_2 , матрица ограничений должна иметь следующий вид:

$$A = \begin{pmatrix} x_1 & x_2 & x_3 & x_4 & w_1 & w_2 \\ 1 & 1 & 1 & 1 & 1 & 0 \\ 1 & -1 & 1 & -1 & 0 & 1 \end{pmatrix}$$

Тогда преобразуем ограничения исходной задачи. В первое ограничение добавим искусственную переменную w_1 , во второе w_2 . Тогда система ограничений примет вид:

$$x_1 + x_2 + x_3 + x_4 + w_1 = 1$$

$$x_1 - x_2 + x_3 - x_4 + w_2 = 1$$

$$x \geq 0, w \geq 0$$

5. Построим вспомогательную задачу

$$F(w) = -w_1 - w_2 \rightarrow \max$$

$$x_1 + x_2 + x_3 + x_4 + w_1 = 1$$

$$x_1 - x_2 + x_3 - x_4 + w_2 = 1$$

$$x \geq 0, w \geq 0$$

Решаем вспомогательную задачу симплекс-методом.

6. Обратим внимание, что столбцы, соответствующие переменным w_1, w_2 образуют единичную матрицу, то есть являются базисными $x^B = (w_1, w_2)$, тогда оставшиеся переменные x_1, x_2, x_3, x_4 являются небазисными $x^N = (x_1, x_2, x_3, x_4)$.

Строим начальную симплекс-таблицу:

		0	0	0	0	
		x_1	x_2	x_3	x_4	b
-1	w_1	1	1	1	1	1
-1	w_2	①	-1	1	-1	1
	Δ^N	-2	0	-2	0	

Таблица 8. Начальная симплекс-таблица вспомогательной задачи

Заметим, что среди оценок есть отрицательные, значит данная симплекс-таблица не является оптимальной. То есть решение можно улучшить. Среди отрицательных оценок выбираем большую по модулю, -2, она определяет главный столбец. В таблице содержится две одинаковые отрицательные оценки, выберем любую из них, например, первую (столбец, соответствующий x_1). В данном столбце есть два положительных элемента 1 и 1, для каждого из них находим симплексное отношение, то есть делим правую часть ограничения на соответствующий элемент и выбираем наименьшее частное. $1:1 = 1; 1:1 = 1$, симплексные отношения равны. В этом случае следует воспользоваться правилом Креко для выбора главного элемента. По данному

правилу следует поочерёдно делить элементы строк с одинаковыми симплексными отношениями на предполагаемые соответствующие главные элементы. Та строка, в которой раньше встретится наименьшее частное (при проходе слева направо) и содержит в себе главный элемент. В данном примере это вторая строка, значит, в качестве главного элемента выбираем 1 из второй строки и выделяем его. Переходим к новой симплекс-таблице и заполняем её:

	w_2	x_2	x_3	x_4	b
w_1	-1	2	0	2	0
x_1	1	-1	1	-1	1
Δ^N	2	-2	0	-2	

Таблица 9. Следующий шаг симплекс-метода вспомогательной задачи

Снова в строке оценок есть отрицательные числа, значит, проделываем ещё один шаг симплекс-метода (аналогично предыдущему). Теперь в качестве главного элемента выступает число 2 из столбца, соответствующего переменной x_2 , выделяем его. Пересчитываем таблицу:

	w_2	w_1	x_3	x_4	b
x_2	$-1/2$	$1/2$	0	1	0
x_1	$1/2$	$1/2$	1	0	1
Δ^N	1	1	0	0	

Таблица 10. Следующий шаг симплекс-метода вспомогательной задачи

В таблице 10 все оценки неотрицательны, значит, она является оптимальной. Вспомогательная задача решена. На следующем шаге будет произведён анализ получившегося решения для получения вывода о разрешимости исходной задачи.

7. Заметим, что все искусственные переменные стали небазисными, то есть $F(w^*) = 0$. Значит вспомогательная задача решена, исходная задача будет иметь решение, переходим от решения вспомогательной задачи к построению начальной симплекс-таблицы исходной. Для этого вычёркиваем столбцы, соответствующие искусственным переменным:

	w_2	w_1	x_3	x_4	b
x_2	$-1/2$	$1/2$	0	1	0
x_1	$1/2$	$1/2$	1	0	1
Δ^N	1	1	0	0	

Таблица 11. Вычёркиваем столбцы, соответствующие искусственным переменным

Для составления начальной симплекс-таблицы исходной задачи пересчитываем строку оценок, используя её целевую функцию $f(x) = 2x_1 + 3x_2$:

	0	0	
	x_3	x_4	b
2 x_2	0	1	0
3 x_1	1	0	1
Δ^N	3	2	

Таблица 12. Начальная симплекс-таблица исходной задачи

Полученная симплекс-таблица является оптимальной, значит, можно выписывать решение исходной задачи: $x^* = (1, 0, 0, 0)$, $f(x^*) = 2$.

Ответ: оптимальное решение $x^* = (1, 0, 0, 0)$, оптимальное значение целевой функции $f(x^*) = 2$.

Рассмотрим ещё один пример решения задачи линейного программирования методом искусственного базиса.

Пример 10. Решить задачу линейного программирования методом искусственного базиса:

$$\begin{aligned}
 f(x) &= 2x_1 + x_2 + x_3 \rightarrow \max \\
 2x_1 - x_2 + x_3 &= 6 \\
 x_1 - 2x_2 - x_3 &= 4 \\
 x &\geq 0
 \end{aligned}$$

1. Заметим, что задача уже находится в канонической форме.
2. Правые части ограничений представляют собой положительные числа, значит можно переходить к матрице ограничений.
3. Выпишем матрицу ограничений, её элементы представляют собой коэффициенты при соответствующем $x_i, i = 1, \dots, 3$:

$$A = \begin{pmatrix} x_1 & x_2 & x_3 \\ 2 & -1 & 1 \\ 1 & -2 & -1 \end{pmatrix}$$

Заметим, что данная матрица не содержит в себе единичную, именно поэтому добавим в ограничения искусственные переменные $w_i, i = 1, 2$ для создания базиса.

4. Добавим две искусственные переменные w_1 и w_2 , матрица ограничений должна иметь следующий вид:

$$A = \begin{pmatrix} x_1 & x_2 & x_3 & w_1 & w_2 \\ 2 & -1 & 1 & 1 & 0 \\ 1 & -2 & -1 & 0 & 1 \end{pmatrix}$$

Система ограничений примет вид:

$$2x_1 - x_2 + x_3 + w_1 = 6$$

$$x_1 - 2x_2 - x_3 + w_2 = 4$$

$$x \geq 0, w \geq 0$$

5. Построим вспомогательную задачу:

$$F(w) = -w_1 - w_2 \rightarrow \max$$

$$2x_1 - x_2 + x_3 + w_1 = 6$$

$$x_1 - 2x_2 - x_3 + w_2 = 4$$

$$x \geq 0, w \geq 0$$

Решаем вспомогательную задачу симплекс-методом.

6. Строим начальную симплекс-таблицу:

		0	0	0	0
		x_1	x_2	x_3	b
-1	w_1	②	-1	1	6
-1	w_2	1	-2	-1	4
	Δ^N	-3	3	0	

Таблица 13. Начальная симплекс-таблица вспомогательной задачи

Заметим, что среди оценок есть одна отрицательная, значит данная симплекс-таблица не является оптимальной. Отрицательная оценка соответствует первому столбцу, в данном столбце есть два положительных элемента 2 и 1, для каждого из них находим симплексное отношение. $6:2 = 3$; $4:1 = 4$, меньшее частное соответствует числу 2, это главный элемент, выделяем его. Переходим к новой симплекс-таблице и заполняем её:

[illegible]

Таким образом, была построена задача, двойственная к исходной.
Удобно записать построенную пару двойственных задач в векторной форме:

Таблица 15. Векторная форма записи пары двойственных задач

Исходная задача	Двойственная задача
$f(x) = (c, x) \rightarrow \max$	$g(y) = (b, y) \rightarrow \min$
$(A_i, x) \leq b_i, i = 1, \dots, m_1$	$y_i \geq 0, i = 1, \dots, m_1$
$(A_i, x) = b_i, i = m_1 + 1, \dots, m$	$y_i - \text{любое}, i = m_1 + 1, \dots, m$
$x_j \geq 0, j = 1, \dots, n_1$	$(A^j, y) \geq c_j, j = 1, \dots, n_1$
$x_j - \text{любое}, j = n_1 + 1, \dots, n$	$(A^j, y) = c_j, j = n_1 + 1, \dots, n$

33

Пример 11. Построить двойственную задачу к следующей задаче линейного программирования:

$$\begin{aligned} f(x) &= 2x_1 + x_2 \rightarrow \max \\ -x_1 + 3x_2 &\leq 6 \\ 2x_1 + x_2 &\leq 9 \\ x_1 - x_3 &\leq 3 \\ x &\geq 0 \end{aligned}$$

Оформим решение в стиле Таблицы 16:

Исходная задача	Двойственная задача
$f(x) = 2x_1 + x_2 \rightarrow \max$	$g(y) = 6y_1 + 9y_2 + 3y_3 \rightarrow \min$
$-x_1 + 3x_2 \leq 6$	$y_1 \geq 0$
$2x_1 + x_2 \leq 9$	$y_2 \geq 0$
$x_1 - x_3 \leq 3$	$y_3 \geq 0$
$x_1 \geq 0$	$-y_1 + 2y_2 + y_3 \geq 2$
$x_2 \geq 0$	$3y_1 + y_2 \geq 1$
$x_3 \geq 0$	$-y_3 \geq 0$

Таблица 17. Построение двойственной задачи

Пример 12. Построить двойственную задачу к следующей задаче линейного программирования:

$$\begin{aligned} f(x) &= -x_1 + 3x_2 \rightarrow \max \\ 5x_1 - 4x_2 &\leq 6 \\ 2x_1 + x_2 &\leq 3 \\ -7x_1 + 3x_2 &= 8 \\ x_1 &\geq 0 \\ x_2 &\text{— любое} \end{aligned}$$

Оформим решение в стиле Таблицы 16:

Исходная задача	Двойственная задача
$f(x) = -x_1 + 3x_2 \rightarrow \max$	$g(y) = 6y_1 + 3y_2 + 8y_3 \rightarrow \min$
$5x_1 - 4x_2 \leq 6$	$y_1 \geq 0$
$2x_1 + x_2 \leq 3$	$y_2 \geq 0$
$-7x_1 + 3x_2 = 8$	$y_3 \text{— любое}$
$x_1 \geq 0$	$5y_1 + 2y_2 - 7y_3 \geq -1$
$x_2 \text{— любое}$	$-4y_1 + y_2 + 3y_3 = 3$

Таблица 18. Построение двойственной задачи

Основные теоремы двойственности и их применение в задачах

Первая теорема двойственности. Если одна из пары двойственных задач имеет решение, то и вторая задача также имеет решение, причём оптимальные значения целевых функций совпадают, то есть $f(x^*) = g(y^*)$. Если одна из пары двойственных задач не имеет решения вследствие неограниченности целевой функции, то это означает несовместность системы ограничений другой задачи.

Вторая теорема двойственности. Чтобы допустимые решения x^* и y^* были оптимальными, необходимо и достаточно, чтобы выполнялись условия дополняющей нежёсткости:

Если $\sum_{j=1}^n a_{ij}x_j^* < b_i \Rightarrow y_i^* = 0$; если $y_i^* > 0 \Rightarrow \sum_{j=1}^n a_{ij}x_j^* = b_i$.

Если $\sum_{i=1}^m a_{ij}y_j^* > c_j \Rightarrow x_j^* = 0$; если $x_j^* > 0 \Rightarrow \sum_{i=1}^m a_{ij}y_j^* = c_j$.

Рассмотрим примеры, иллюстрирующие Первую теорему двойственности.

Пример 13. Решить задачу линейного программирования, построить двойственную к ней и решить её.

$$\begin{aligned} f(x) &= 3x_1 + 3x_2 \rightarrow \max \\ 2x_1 - x_2 &\geq 1 \\ x_1 - 2x_2 &\leq 2 \\ x_1 \geq 0, x_2 &\geq 0 \end{aligned}$$

Данная задача была решена графически в Примере 6. Имела место Неразрешимость 2. Построим двойственную задачу. Оформим построение двойственной задачи в виде таблицы:

Исходная задача	Двойственная задача
$f(x) = 3x_1 + 3x_2 \rightarrow \max$	$g(y) = -y_1 + 2y_2 \rightarrow \min$
$-2x_1 + x_2 \leq -1$	$y_1 \geq 0$
$x_1 - 2x_2 \leq 2$	$y_2 \geq 0$
$x_1 \geq 0$	$-2y_1 + y_2 \geq 3$
$x_2 \geq 0$	$y_1 - 2y_2 \geq 3$

Таблица 19. Построение двойственной задачи

Решим графически построенную двойственную задачу:

Построим область допустимых решений.

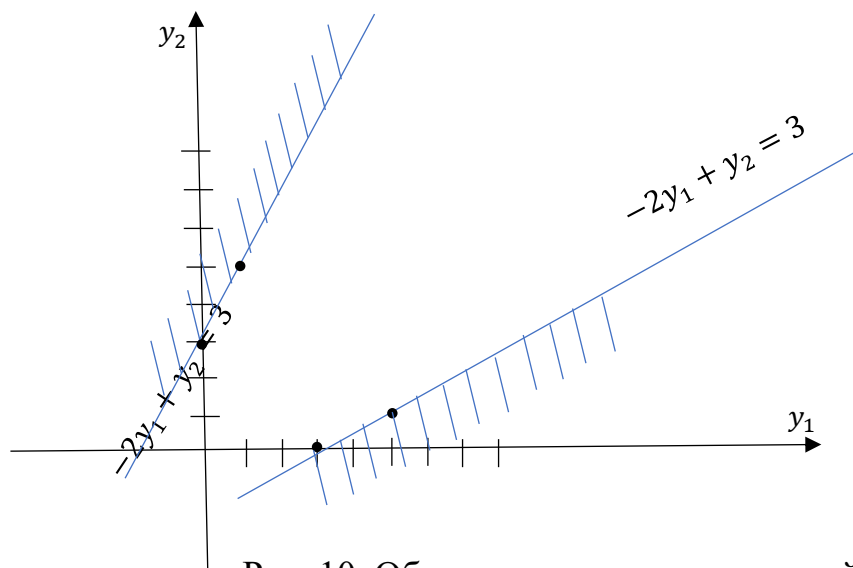


Рис. 10. Область допустимых решений

Видим, что ОДР представляет собой пустое множество, то есть система ограничений двойственной задачи несовместна.

Пример 14. Решить задачу линейного программирования, построить двойственную к ней и решить её.

$$f(x) = 3x_1 + 3x_2 \rightarrow \max$$

$$x_1 + 2x_2 \leq 8$$

$$x_1 - 2x_2 \leq 2$$

$$x_1 \geq 0, x_2 \geq 0$$

Решим исходную задачу графически:

Область допустимых решений:

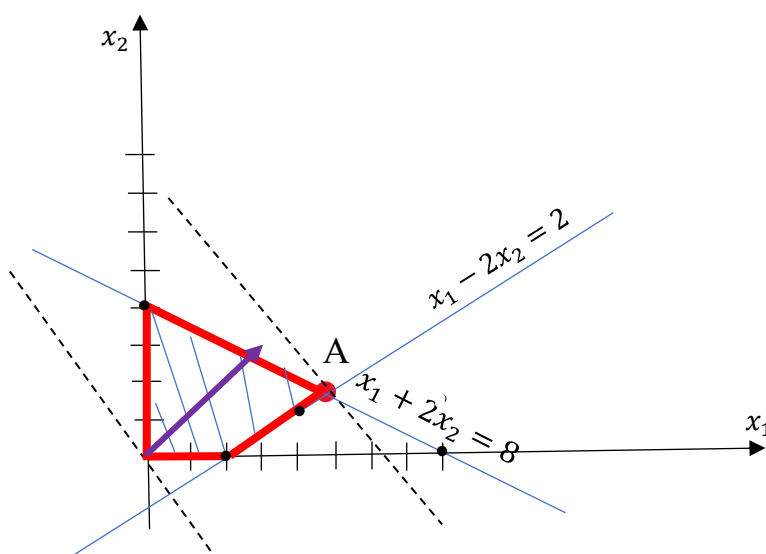


Рис. 11. Оптимальное решение (максимум целевой функции)

Рассчитаем координаты оптимального решения как точку пересечения прямых $x_1 + 2x_2 = 8$ и $x_1 - 2x_2 = 2$. Получим оптимальное решение (максимум целевой функции) $x^* = (5; 1.5)$. Оптимальное значение целевой функции $f(x^*) = 3 \cdot 5 + 3 \cdot 1.5 = 19.5$.

Построим задачу, двойственную к исходной:

Исходная задача	Двойственная задача
$f(x) = 3x_1 + 3x_2 \rightarrow \max$	$g(y) = 8y_1 + 2y_2 \rightarrow \min$
$x_1 + 2x_2 \leq 8$	$y_1 \geq 0$
$x_1 - 2x_2 \leq 2$	$y_2 \geq 0$
$x_1 \geq 0$	$y_1 + y_2 \geq 3$
$x_2 \geq 0$	$2y_1 - 2y_2 \geq 3$

Таблица 20. Построение двойственной задачи

Решим графически построенную двойственную задачу:

Область допустимых решений:

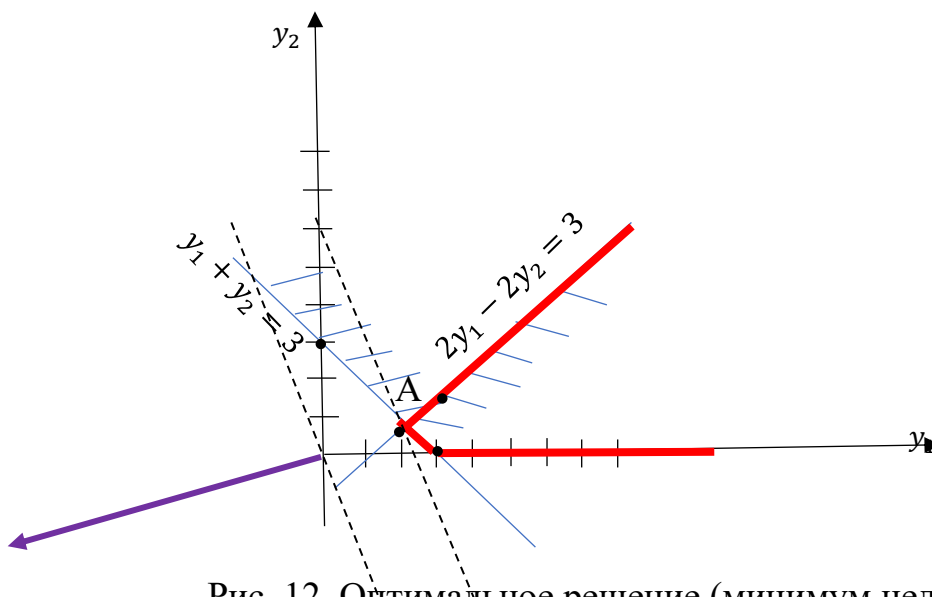


Рис. 12. Оптимальное решение (минимум целевой функции)

Рассчитаем координаты оптимального решения как точку пересечения прямых $y_1 + y_2 = 3$ и $2y_1 - 2y_2 = 3$. Получим оптимальное решение (минимум целевой функции) $y^* = (9/4; 3/4)$. Оптимальное значение целевой функции $g(y^*) = 8 \cdot 9/4 + 2 \cdot 3/4 = 19.5$.

Видим, что значения целевых функций совпадают, как и утверждалось в Первой теореме двойственности.

Рассмотрим примеры, иллюстрирующие Вторую теорему двойственности.

Пример 15. Решить задачу линейного программирования, составить двойственную к ней и найти её решение с помощью Второй теоремы двойственности (не решая в явном виде).

$$f(x) = -x_1 + 2x_2 \rightarrow \max$$

$$x_1 - 3x_2 \leq 3$$

$$-x_1 + x_2 \leq 1$$

$$-x_1 + 2x_2 \leq 4$$

$$x_1 \geq 0, x_2 \geq 0$$

Решим данную задачу симплекс-методом.

Каноническая форма:

$$f(x) = -x_1 + 2x_2 \rightarrow \max$$

$$x_1 - 3x_2 + x_3 = 3$$

$$-x_1 + x_2 + x_4 = 1$$

$$-x_1 + 2x_2 + x_5 = 4$$

$$x \geq 0$$

Матрица ограничений:

$$A = \begin{pmatrix} x_1 & x_2 & x_3 & x_4 & x_5 \\ 1 & -3 & 1 & 0 & 0 \\ -1 & 1 & 0 & 1 & 0 \\ -1 & 2 & 0 & 0 & 1 \end{pmatrix}$$

Начальная симплекс-таблица:

		-1	2	
		x_1	x_2	b
0	x_3	1	-3	3
0	x_4	-1	1	1
0	x_5	-1	2	4
	Δ^N	1	-2	

Таблица 21. Начальная симплекс-таблица

Переходим к новой симплекс-таблице:

	x_1	x_4	b
x_3	-2	3	6
x_2	-1	1	1
x_5	1	-2	2
Δ^N	-1	2	

Таблица 22. Следующий шаг симплекс-метода

Пересчитываем таблицу аналогично предыдущей:

	x_5	x_4	b
x_3	2	-1	10
x_2	1	-1	3
x_1	1	-2	2
Δ^N	1	0	

Таблица 23. Следующий шаг симплекс-метода

В таблице 23 все оценки неотрицательны, значит, она является оптимальной и можно выписывать решение. Оптимальное решение имеет вид $x^* = (2; 3)$. Осталось найти оптимальное значение целевой функции. $f(x^*) = -2 + 2 \cdot 3 = 4$.

Построим задачу, двойственную к исходной:

Исходная задача	Двойственная задача
$f(x) = -x_1 + 2x_2 \rightarrow \max$	$g(y) = 3y_1 + y_2 + 4y_3 \rightarrow \min$
$x_1 - 3x_2 \leq 3$	$y_1 \geq 0$
$-x_1 + x_2 \leq 1$	$y_2 \geq 0$
$-x_1 + 2x_2 \leq 4$	$y_3 \geq 0$
$x_1 \geq 0$	$y_1 - y_2 - y_3 \geq -1$
$x_2 \geq 0$	$-3y_1 + y_2 + 2y_3 \geq 2$

Таблица 24. Построение двойственной задачи

Запишем в удобном табличном виде условия дополняющей нежёсткости таким образом, чтобы при подстановке оптимального решения в одну из пары двойственных задач, получалось решение другой. Пусть x^* – решение исходной задачи, y^* – решение двойственной:

Результаты подстановки	Вывод
$(A_i, x^*) < b_i, i = 1, \dots, m_1$	$y_i^* = 0, i = 1, \dots, m_1$
$(A_i, x^*) = b_i, i = m_1 + 1, \dots, m$	$y_i^* \geq 0, i = m_1 + 1, \dots, m$
$x_j^* > 0, j = 1, \dots, n_1$	$(A^j, y^*) = c_j, j = 1, \dots, n_1$
$x_j^* = 0, j = n_1 + 1, \dots, n$	$(A^j, y^*) \geq c_j, j = n_1 + 1, \dots, n$

Таблица 25. Решение двойственной задачи

Воспользуемся данной таблицей и продолжим рассмотрение примера. Подставим полученное решение $x^* = (2; 3)$ в исходную задачу и получим:

Результаты подстановки	Вывод
$2 - 3 \cdot 3 < 3$	$y_1^* = 0$
$-2 + 3 = 1$	$y_2^* \geq 0$
$-2 + 2 \cdot 3 = 4$	$y_3^* \geq 0$
$2 > 0$	$y_1^* - y_2^* - y_3^* = -1$
$3 > 0$	$-3y_1^* + y_2^* + 2y_3^* = 2$

Таблица 26. Результаты подстановки

Во втором столбце таблицы получилась система, решив которую, можно построить решение двойственной задачи y^* . Подставим $y_1^* = 0$ в четвёртое и пятое равенства, получим:

$$\begin{cases} -y_2^* - y_3^* = -1 \\ y_2^* + 2y_3^* = 2 \end{cases}$$

Откуда $y_2^* = 0, y_3^* = 1$. Таким образом, построено решение двойственной задачи $y^* = (0, 0, 1)$. Оптимальное значение целевой функции $g(y^*) = 4$.

Ответ: оптимальное решение исходной задачи $x^* = (2; 3)$, оптимальное решение двойственной задачи $y^* = (0, 0, 1)$. Оптимальное значение целевой функции $f(x^*) = g(y^*) = 4$.

Также с помощью Второй теоремы двойственности можно проверять, является ли точка оптимальным решением задачи линейного программирования. Рассмотрим следующий пример.

Пример 16. Найти значение параметра λ , при котором точка $x^* = (2; 1)$ является оптимальным решением задачи линейного программирования:

$$\begin{aligned} f(x) &= x_1 + \lambda x_2 \rightarrow \max \\ x_1 + x_2 &\leq 3 \\ x_1 - x_2 &\leq 1 \\ -x_1 - x_2 &\leq 2 \\ x_1 &\geq 0, x_2 \geq 0 \end{aligned}$$

Воспользуемся Таблицей 25 и определим, для каких значений параметра λ двойственная задача имеет решение. Если двойственная задача разрешима для полученных значений параметра, значит для этих же значений параметра точка $x^* = (2; 1)$ будет являться оптимальным решением ЗЛП.

Начнём с проверки точки на допустимость. Подставив координаты $x^* = (2; 1)$ во все ограничения задачи, видим, что все неравенства справедливы, значит точка является допустимым решением. Продолжаем проверку на оптимальность. Подставим координаты точки в исходную задачу:

Результаты подстановки	Вывод
$2 + 1 = 3$	$y_1^* \geq 0$
$2 - 1 = 1$	$y_2^* \geq 0$
$-2 - 1 < 2$	$y_3^* = 0$
$2 > 0$	$y_1^* + y_2^* - y_3^* = 1$
$1 > 0$	$y_1^* - y_2^* - y_3^* = \lambda$

Таблица 26. Результаты подстановки

Решаем систему, образовавшуюся во втором столбце таблицы. Подставим $y_3^* = 0$ в четвертое и пятое равенства:

$$\begin{cases} y_1^* + y_2^* = 1 \\ y_1^* - y_2^* = \lambda \end{cases}$$

$y_1^* = \frac{1+\lambda}{2}, y_2^* = \frac{1-\lambda}{2}$, потребуем выполнения условий $y_1^* \geq 0, y_2^* \geq 0$:

$$\begin{cases} \frac{1+\lambda}{2} \geq 0 \\ \frac{1-\lambda}{2} \geq 0 \end{cases}$$

Система имеет решение, если $\lambda \in [-1; 1]$. То есть двойственная задача имеет решение при $\lambda \in [-1; 1]$, значит $x^* = (2; 1)$ является оптимальным решением исходной задачи при $\lambda \in [-1; 1]$.

Ответ: $x^* = (2; 1)$ является оптимальным решением при $\lambda \in [-1; 1]$.

Также теорию двойственности линейного программирования удобно применять в следующем случае. Предположим, что исходная задача достаточно громоздкая (имеет большое количество переменных), но количество ограничений не так велико. Это указывает на то, что составленная двойственная задача будет более простой, так как в её состав будет входить меньшее число переменных. То есть удобно от исходной задачи перейти к более простой двойственной, решить её, а затем с помощью Второй теоремы двойственности восстановить решение исходной. Рассмотрим пример.

Пример 17. Решить задачу линейного программирования с использованием теории двойственности:

$$\begin{aligned} f(x) &= 3x_1 + 4x_2 + 5x_3 + 6x_4 \rightarrow \min \\ 2x_1 + x_2 - x_3 + 5x_4 &\geq 5 \\ 3x_1 - 2x_2 + x_3 + 4x_4 &\geq 4 \\ x &\geq 0 \end{aligned}$$

Данную задачу стоит решать методом искусственного базиса. Но заметим, что в задаче содержится всего два ограничения, то есть в двойственной задаче будет лишь две переменные. От исходной задачи перейдём к двойственной:

Исходная задача	Двойственная задача
$f(x) = 3x_1 + 4x_2 + 5x_3 + 6x_4$ $\rightarrow \min$	$g(y) = 5y_1 + 4y_2 \rightarrow \max$
$2x_1 + x_2 - x_3 + 5x_4 \geq 5$	$y_1 \geq 0$
$3x_1 - 2x_2 + x_3 + 4x_4 \geq 4$	$y_2 \geq 0$
$x_1 \geq 0$	$2y_1 + 3y_2 \leq 3$
$x_2 \geq 0$	$y_1 - 2y_2 \leq 4$
$x_3 \geq 0$	$-y_1 + y_2 \leq 5$
$x_4 \geq 0$	$5y_1 + 4y_2 \leq 6$

Таблица 27. Переход от исходной задачи к двойственной

Двойственная задача содержит всего две переменные, значит, будет удобно решать её графически:

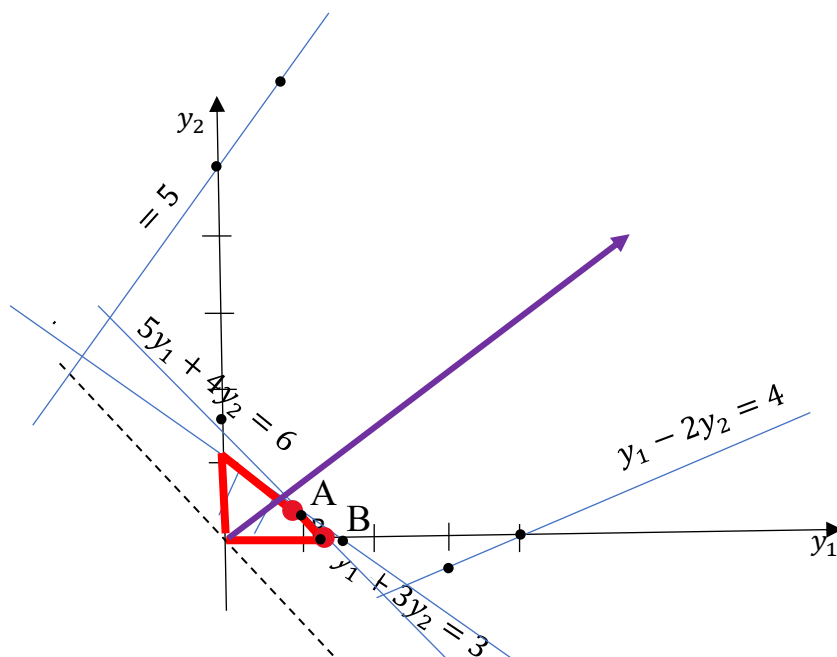


Рис. 13. Оптимальное решение (максимум целевой функции)

Оптимальное решение – или точка A, или точка B (неочевидно). Найдём значение целевой функции в каждой из точек и определим, где функция достигает своё наибольшее значение. Координаты точки A найдём как координаты точки пересечения прямых $5y_1 + 4y_2 = 6$ и $2y_1 + 3y_2 = 3$, получим $A\left(\frac{6}{7}; \frac{3}{7}\right)$. Координаты точки B найдём как координаты точки

пересечения прямой $5y_1 + 4y_2 = 6$ и оси абсцисс $y_2 = 0$, получим $B\left(\frac{6}{5}; 0\right)$.
 $g(A) = 5 \cdot \frac{6}{7} + 4 \cdot \frac{3}{7} = 6$, $g(B) = 5 \cdot \frac{6}{5} = 6$. Так как значения целевой функции в точках A и B совпали, то делаем вывод, что функция $g(y)$ принимает максимальное значение на всём отрезке AB .

Подставим координаты точки B в решённую задачу:

Результаты подстановки	Вывод
$2 \cdot \frac{6}{5} < 3$	$x_1^* = 0$
$\frac{6}{5} < 4$	$x_2^* = 0$
$-\frac{6}{5} < 5$	$x_3^* = 0$
$5 \cdot \frac{6}{5} = 6$	$x_4^* \geq 0$
$\frac{6}{5} > 0$	$2x_1 + x_2 - x_3 + 5x_4 = 5$
$0 = 0$	$3x_1 - 2x_2 + x_3 + 4x_4 \geq 4$

Таблица 28. Результаты подстановки

Решаем систему, образовавшуюся во втором столбце таблицы. Очевидно, решение имеет вид $x^* = (0; 0; 0; 1)$. $f(x^*) = 6$.

Ответ: оптимальное решение $x^* = (0; 0; 0; 1)$; оптимальное значение целевой функции $f(x^*) = 6$.

9. Параметрическое линейное программирование

В задаче параметрического линейного программирования возможно наличие параметра как в целевой функции, так и в ограничениях. Необходимость рассмотрения задач данного класса связана с тем, что исходные данные реальных задач могут быть или приближёнными или меняться в зависимости от некоторых факторов. То есть для того, чтобы учесть возможные изменения, следует указывать не конкретное решение задачи, а диапазон возможного изменения данных.

Для решения задач данного типа будем использовать следующий подход: находить оптимальное решение для конкретных исходных данных и определять, для каких значений параметра данное решение будет оставаться оптимальным.

Рассмотрим класс задач, в которых параметр λ содержится только в целевой функции и сформулируем алгоритм решения данных задач:

1. Фиксируем некоторое значение параметра λ , для него находим оптимальное решение x^* , если это сделать невозможно, то делаем вывод о неразрешимости задачи.
2. Определяем множество значений параметра λ , для которых найденное решение x^* остаётся оптимальным или задача неразрешима. Используемое множество значений параметра λ убираем из рассмотрения.
3. Выбираем некоторое значение параметра λ из оставшегося множества его значений, находим оптимальное решение x^* , определяем множество значений параметра λ , для которых найденное решение x^* остаётся оптимальным.
4. Данные исследования продолжаем до тех пор, пока не будут рассмотрены все возможные значения параметра λ .

Рассмотрим пример решения задачи параметрического линейного программирования

Пример 18. Решить задачу параметрического линейного программирования:

$$\begin{aligned} f(x) &= 2x_1 + (3 + 4\lambda)x_2 \rightarrow \max \\ x_1 + x_2 + x_3 &= 12 \\ x_1 - x_2 + x_4 &= 10 \\ -x_1 + x_2 + x_5 &= 6 \\ x &\geq 0 \\ \lambda &\in (-\infty; +\infty) \end{aligned}$$

1. Зафиксируем некоторое начальное значение параметра λ . Пусть, например, $\lambda = 0$. Решим исходную задачу симплекс-методом. Заметим, что задача уже находится в канонической форме. Выпишем матрицу ограничений:

$$A = \begin{pmatrix} x_1 & x_2 & x_3 & x_4 & x_5 \\ 1 & 1 & 1 & 0 & 0 \\ 1 & -1 & 0 & 1 & 0 \\ -1 & 1 & 0 & 0 & 1 \end{pmatrix}$$

Начальная симплекс-таблица. Помним предположение, что $\lambda = 0$.

		2	$3 + 4\lambda$	
		x_1	x_2	b
0	x_3	1	1	12
0	x_4	1	-1	10
0	x_5	-1	(1)	6
	Δ^N	-2	$-3-4\lambda$	

Таблица 29. Начальная симплекс-таблица

Переходим к новой симплекс-таблице:

		x_1	x_5	b
	x_3	(2)	-1	6
	x_4	0	1	16
	x_2	-1	1	6
	Δ^N	$-5-4\lambda$	$3+4\lambda$	

Таблица 30. Следующий шаг симплекс-метода

Пересчитываем таблицу аналогично предыдущей.

		x_3	x_5	b
	x_1	$1/2$	$-1/2$	3
	x_4	0	1	16
	x_2	$1/2$	$1/2$	9
	Δ^N	$\frac{5}{2} + 2\lambda$	$\frac{1}{2} + 2\lambda$	

Таблица 31. Следующий шаг симплекс-метода

Так как задача решалась в предположении, что $\lambda = 0$, то полученная симплекс-таблица является оптимальной. Оптимальное решение $x^* = (3; 9; 0; 16; 0)$.

2. Определим множество значений параметра λ , для которых найденное решение x^* остаётся оптимальным. Решение оптимально, если оценки неотрицательны, то есть:

$$\begin{cases} \frac{5}{2} + 2\lambda \geq 0 \\ \frac{1}{2} + 2\lambda \geq 0 \end{cases}$$

Решив систему, получим, что $\lambda \in \left[-\frac{1}{4}; +\infty\right)$. То есть при $\lambda \in \left[-\frac{1}{4}; +\infty\right)$ ЗЛП имеет оптимальное решение $x^* = (3; 9; 0; 16; 0)$.

3. Возьмём $\lambda < -\frac{1}{4}$. Конкретное значение можно не фиксировать, а принять равным числу, меньшему на некоторое ε , чем $-\frac{1}{4}$. Тогда Таблица 31 перестает быть оптимальной вследствие отрицательности оценки $\frac{1}{2} + 2\lambda$. Выбираем в ней главный элемент.

	x_3	x_5	b
x_1	$1/2$	$-1/2$	3
x_4	0	(1)	16
x_2	$1/2$	$1/2$	9
Δ^N	$\frac{5}{2} + 2\lambda$	$\frac{1}{2} + 2\lambda$	

Таблица 32. Следующий шаг симплекс-метода

И переходим к следующему шагу симплекс-метода:

	x_3	x_4	b
x_1	$1/2$	$1/2$	11
x_5	0	1	16
x_2	$1/2$	$-1/2$	1
Δ^N	$\frac{5}{2} + 2\lambda$	$-\frac{1}{2} - 2\lambda$	

Таблица 33. Следующий шаг симплекс-метода

Здесь оптимальное решение $x^* = (11; 1; 0; 0; 16)$. Определим множество значений параметра λ , для которых найденное решение x^* остаётся оптимальным. Решение оптимально, если оценки неотрицательны, то есть:

$$\begin{cases} \frac{5}{2} + 2\lambda \geq 0 \\ -\frac{1}{2} - 2\lambda \geq 0 \end{cases}$$

Решив систему, получим, что $\lambda \in \left[-\frac{5}{4}; -\frac{1}{4}\right)$. Правая граница не включается, так как $\lambda = -\frac{1}{4}$ рассмотрено на предыдущем шаге. То есть при $\lambda \in \left[-\frac{5}{4}; -\frac{1}{4}\right)$ ЗЛП имеет оптимальное решение $x^* = (11; 1; 0; 0; 16)$.

4. Возьмём $\lambda < -\frac{5}{4}$. Тогда Таблица 33 перестает быть оптимальной вследствие отрицательности оценки $\frac{5}{2} + 2\lambda$. Выбираем в ней главный элемент.

	x_3	x_4	b
x_1	$1/2$	$1/2$	11
x_5	0	1	16
x_2	$1/2$	$-1/2$	1
Δ^N	$\frac{5}{2} + 2\lambda$	$-\frac{1}{2} - 2\lambda$	

Таблица 34. Следующий шаг симплекс-метода

И переходим к следующему шагу симплекс-метода:

	x_2	x_4	b
x_1	-1	1	10
x_5	0	1	16
x_3	2	-1	2
Δ^N	$-5 - 4\lambda$	2	

Таблица 35. Следующий шаг симплекс-метода

Здесь оптимальное решение $x^* = (10; 0; 2; 0; 16)$. Определим множество значений параметра λ , для которых найденное решение x^* остаётся оптимальным. Решение оптимально, если оценки неотрицательны, то есть:

$$\begin{cases} -5 - 4\lambda \geq 0 \\ 2 \geq 0 \end{cases}$$

Решив систему, получим, что $\lambda \in (-\infty; -\frac{5}{4})$. Правая граница не включается, так как $\lambda = -\frac{5}{4}$ рассмотрено на предыдущем шаге. То есть при $\lambda \in (-\infty; -\frac{5}{4})$ ЗЛП имеет оптимальное решение $x^* = (10; 0; 2; 0; 16)$.

Заметим, что на этом шаге рассмотрен весь промежуток с возможными значениями параметра λ , $\lambda \in (-\infty; +\infty)$. Значит, можно записать ответ.

Ответ: при $\lambda \in (-\infty; -\frac{5}{4})$ оптимальное решение $x^* = (10; 0; 2; 0; 16)$;

при $\lambda \in [-\frac{5}{4}; -\frac{1}{4})$ оптимальное решение $x^* = (11; 1; 0; 0; 16)$;

при $\lambda \in [-\frac{1}{4}; +\infty)$ оптимальное решение $x^* = (3; 9; 0; 16; 0)$.

10. Транспортная задача

Постановка транспортной задачи

Транспортная задача является задачей линейного программирования и её смысл заключается в определении наиболее экономичного плана перевозок однородного товара из пунктов отправления (склады) в пункты назначения (магазины).

Пусть имеется m пунктов отправления $A_1, A_2, \dots, A_m$ из которых следует перевезти однородный груз в n пунктов назначения $B_1, B_2, \dots, B_n$. Пусть $c_{ij}, i = 1, \dots, m, j = 1, \dots, n$ – стоимость перевозки единицы груза из i –го пункта отправления в j –й пункт назначения (тариф перевозки). Обозначим через $a_i, i = 1, \dots, m$ – запасы груза на i –м пункте отправления; $b_j, j = 1, \dots, n$ – потребности в грузе на j –м пункте назначения; $x_{ij}, i = 1, \dots, m, j = 1, \dots, n$ – количество единиц груза, перевозимое из i –го пункта отправления в j –й пункт назначения. Матрица $X = (x_{ij})_{m \times n}$ называется **планом перевозок**. Матрица $C = (c_{ij})_{m \times n}$ называется **матрицей тарифов**.

В качестве критерия оптимальности выберем минимальную стоимость перевозок всего груза. Таким образом, решить транспортную задачу – это определить план перевозок $x_{ij}, i = 1, \dots, m, j = 1, \dots, n$, то есть количество груза, которое следует перевезти из i –го пункта отправления в j –й пункт назначения, чтобы все пункты отправления избавились от запасов груза, а все пункты назначения получили требуемое количество груза и при этом стоимость перевозки всего груза была наименьшей.

Математическая модель транспортной задачи примет вид:

$$\begin{aligned} F &= \sum_{i=1}^m \sum_{j=1}^n c_{ij} x_{ij} \rightarrow \min \\ \sum_{i=1}^m x_{ij} &= b_j, j = 1, \dots, n \\ \sum_{j=1}^n x_{ij} &= a_i, i = 1, \dots, m \\ x_{ij} &\geq 0, i = 1, \dots, m, j = 1, \dots, n \end{aligned}$$

Любое неотрицательное решение $X_{ij} = \|x_{ij}\|, i = 1, \dots, m, j = 1, \dots, n$, удовлетворяющее ограничениям, называется **допустимым планом**.

Допустимый план $X_{ij}^* = \|x_{ij}^*\|, i = 1, \dots, m, j = 1, \dots, n$, при котором целевая функция принимает наименьшее значение, называется **оптимальным планом**.

Модель транспортной задачи называется **закрытой (сбалансированной)**, если суммарные потребности равны суммарным запасам, то есть $\sum_{i=1}^m a_i = \sum_{j=1}^n b_j$. В противном случае модель называется **открытой (несбалансированной)**.

Необходимым и достаточным условием разрешимости транспортной задачи является равенство суммарных потребностей суммарным запасам. Если это не так, то есть $\sum_{i=1}^m a_i \neq \sum_{j=1}^n b_j$, то задачу следует преобразовать следующим образом:

1. Если $\sum_{i=1}^m a_i > \sum_{j=1}^n b_j$, то есть суммарные запасы больше суммарных потребностей, то в этом случае следует добавить фиктивный пункт назначения B_{n+1} , потребность которого $b_{n+1} = \sum_{i=1}^m a_i - \sum_{j=1}^n b_j$.
2. Если $\sum_{i=1}^m a_i < \sum_{j=1}^n b_j$, то есть суммарные запасы меньше суммарных потребностей, то в этом случае следует добавить фиктивный пункт отправления A_{m+1} , запасы которого $a_{m+1} = \sum_{j=1}^n b_j - \sum_{i=1}^m a_i$.

При этом соответствующие тарифы считаются равными нулю. После данных преобразований модель становится закрытой.

При решении транспортных задач используют лишь закрытые модели, если модель является открытой, то с помощью преобразований, описанных выше, модель превращают в закрытую и только после этого переходят к решению задачи.

Условие транспортной задачи удобно записывать в виде таблицы:

Пункты отправления	Пункты назначения				Запасы
	B_1	B_2	...	B_n	
A_1	c_{11} x_{11}	c_{12} x_{12}	...	c_{1n} x_{1n}	a_1
A_2	c_{21} x_{21}	c_{22} x_{22}	...	c_{2n} x_{2n}	a_2
...	...	...	...	...	...
A_m	c_{m1} x_{m1}	c_{m2} x_{m2}	...	c_{mn} x_{mn}	a_m
Потребности	b_1	b_2	...	b_n	

Таблица 36. Условие транспортной задачи в табличной форме

Решение транспортной задачи начинается с построения начального опорного плана. Рассмотрим два метода – метод минимального элемента и метод северо-западного угла.

Построение начального опорного плана Метод минимального элемента

Рассмотрим на примере построение начального опорного плана методом минимального элемента.

Пример 19. Транспортная задача задана в виде таблицы. Построить начальный опорный план методом минимального элемента.

Пункты отправления	Пункты назначения				Запасы
	B_1	B_2	B_3	B_4	
A_1	2 x_{11}	3 x_{12}	4 x_{13}	2 x_{14}	140
A_2	8 x_{21}	4 x_{22}	1 x_{23}	4 x_{24}	160
A_3	9 x_{31}	7 x_{32}	3 x_{33}	6 x_{34}	120
Потребности	150	90	100	80	

Таблица 37. Условие транспортной задачи

1. Проверяем, является ли модель закрытой. Суммарные потребности $150 + 90 + 100 + 80 = 420$; суммарные запасы $140 + 160 + 120 = 420$. Суммарные потребности совпадают с суммарными запасами, значит, можно переходить к построению начального опорного плана.

2. Выбираем клетку с минимальным тарифом, в данном случае минимальный тариф равен 1 (пересечение A_2 и B_3). Если есть несколько клеток с одинаковыми минимальными тарифами, то можно выбрать любую из них. Далее записываем в эту клетку меньшее число из запасов и потребностей, на пересечении которых находится выбранная клетка (меньшее из чисел 100 и 160, то есть 100), и вычитаем из запасов и потребностей записанное число. Заметим, что потребности третьего потребителя обнулились, убираем из рассмотрения третий столбец, нерассмотренные клетки остаются пустыми.

Пункты отправления	Пункты назначения				Запасы
	B_1	B_2	B_3	B_4	
A_1	2 x_{11}	3 x_{12}	4 x_{13}	2 x_{14}	140
A_2	8 x_{21}	4 x_{22}	1 100	4 x_{24}	60
A_3	9 x_{31}	7 x_{32}	3 x_{33}	6 x_{34}	120
Потребности	150	90	0	80	

Таблица 38. Построение начального опорного плана

3. Теперь минимальный тариф равен 2 (выберем, например, пересечение A_1 и B_1). Далее записываем в эту клетку меньшее число из запасов и потребностей, на пересечении которых находится выбранная клетка (меньшее из чисел 140 и 150, то есть 140), и вычитаем из запасов и потребностей записанное число. Заметим, что запасы первого пункта отправления обнулились, убираем из рассмотрения первую строку, нерассмотренные клетки остаются пустыми.

Пункты отправления	Пункты назначения				Запасы
	B_1	B_2	B_3	B_4	
A_1	2 140	3	4	2	0
A_2	8 x_{21}	4 x_{22}	1 100	4 x_{24}	60
A_3	9 x_{31}	7 x_{32}	3 x_{33}	6 x_{34}	120
Потребности	10	90	0	80	

Таблица 39. Построение начального опорного плана

4. Продолжаем аналогичные действия и на последнем шаге получим следующий опорный план:

Пункты отправления	Пункты назначения				Запасы
	B_1	B_2	B_3	B_4	
A_1	2 140	3	4	2	0
A_2	8	4 60	1 100	4	0
A_3	9 10	7 30	3	6 80	0
Потребности	0	0	0	0	

Таблица 40. Начальный опорный план

Получен следующий план перевозок:

$$X = \begin{pmatrix} 140 & 0 & 0 & 0 \\ 0 & 60 & 100 & 0 \\ 10 & 30 & 0 & 80 \end{pmatrix}$$

План перевозок показывает, какое количество груза требуется перевезти из i —го пункта отправления в j —й пункт назначения, $i = 1, \dots, 3; j = 1, \dots, 4$.

Рассчитаем стоимость данной перевозки: $F = 2 \cdot 140 + 9 \cdot 10 + 4 \cdot 60 + 7 \cdot 30 + 1 \cdot 100 + 6 \cdot 80 = 1400$.

Метод северо-западного угла

Рассмотрим на примере построение начального опорного плана методом северо-западного угла.

Пример 20. Транспортная задача задана в виде таблицы. Построить начальный опорный план методом северо-западного угла.

Пункты отправления	Пункты назначения				Запасы
	B_1	B_2	B_3	B_4	
A_1	2 x_{11}	3 x_{12}	4 x_{13}	2 x_{14}	140
A_2	8 x_{21}	4 x_{22}	1 x_{23}	4 x_{24}	160
A_3	9 x_{31}	7 x_{32}	3 x_{33}	6 x_{34}	120
Потребности	150	90	100	80	

Таблица 41. Условие транспортной задачи

1. Проверяем, является ли модель закрытой. Проверка выполнена в предыдущем примере.

2. Выбираем самую северо-западную клетку. Эта клетка находится на пересечении A_1 и B_1 . Записываем в эту клетку меньшее число из запасов и потребностей, на пересечении которых находится выбранная клетка (меньшее из чисел 140 и 150, то есть 140), и вычитаем из запасов и потребностей записанное число. Заметим, что запасы первого пункта отправления обнулились, убираем из рассмотрения первую строку, нерассмотренные клетки остаются пустыми.

Пункты отправления	Пункты назначения				Запасы
	B_1	B_2	B_3	B_4	
A_1	2 140	3	4	2	0
A_2	8 x_{21}	4 x_{22}	1 x_{23}	4 x_{24}	160
A_3	9 x_{31}	7 x_{32}	3 x_{33}	6 x_{34}	120
Потребности	10	90	100	80	

Таблица 42. Построение начального опорного плана

3. Теперь самая северо-западная клетка находится на пересечении A_2 и B_1 . Далее записываем в эту клетку меньшее число из запасов и потребностей, на пересечении которых находится выбранная клетка (меньшее из чисел 10 и 160, то есть 10), и вычитаем из запасов и потребностей записанное число. Заметим, что потребности первого пункта назначения обнулились, убираем из рассмотрения первый столбец, нерассмотренные клетки остаются пустыми.

Пункты отправления	Пункты назначения				Запасы
	B_1	B_2	B_3	B_4	
A_1	2 140	3	4	2	0
A_2	8 10	4 x_{22}	1 x_{23}	4 x_{24}	150
A_3	9	7 x_{32}	3 x_{33}	6 x_{34}	120
Потребности	0	90	100	80	

Таблица 43. Построение начального опорного плана

4. Продолжаем аналогичные действия и на последнем шаге получим следующий опорный план:

Пункты отправления	Пункты назначения				Запасы
	B_1	B_2	B_3	B_4	
A_1	2 140	3	4	2	0
A_2	8 10	4 90	1 60	4	0
A_3	9	7	3 40	6 80	0
Потребности	0	0	0	0	

Таблица 44. Начальный опорный план

Получен следующий план перевозок:

$$X = \begin{pmatrix} 140 & 0 & 0 & 0 \\ 10 & 90 & 60 & 0 \\ 0 & 0 & 40 & 80 \end{pmatrix}$$

План перевозок показывает, какое количество груза требуется перевезти из i –го пункта отправления в j –й пункт назначения, $i = 1, \dots, 3; j = 1, \dots, 4$.

Рассчитаем стоимость данной перевозки: $F = 2 \cdot 140 + 8 \cdot 10 + 4 \cdot 90 + 1 \cdot 60 + 3 \cdot 40 + 6 \cdot 80 = 1380$.

Таким образом, был построен начальный опорный план. Следует проверить его оптимальность и дальнейшее решение задачи будет заключаться в усовершенствовании опорного плана с помощью метода потенциалов.

Стоит отметить, что для построения начального опорного плана можно использовать как метод минимального элемента, так и метод северо-западного угла. В итоге в результате применения метода потенциалов получится один и тот же ответ.

Метод потенциалов. Алгоритм

Определение 4. Клетки таблицы, содержащие не равные нулю перевозки, называют **базисными**, остальные – **свободными**.

Определение 5. План называется **вырожденным**, если количество базисных клеток в нём меньше, чем $m + n - 1$.

Определение 6. **Цикл** – несколько клеток, соединённых замкнутой ломаной линией таким образом, что две соседние вершины ломаной лежат или в одной строке, или в одном столбце. Одна вершина находится в свободной клетке, остальные – в базисных. Ломаная может пересекать себя, но только не в клетках цикла.

Алгоритм метода потенциалов.

1. Добавить в транспортную задачу потенциалы $u_i, i = 1, \dots, m; v_j, j = 1, \dots, n$. Потенциалы добавляются по базисным клеткам таким образом, что $u_i + v_j = c_{ij}$. То есть для определения потенциалов решается система уравнений, в которой одному из потенциалов придаём произвольное значение. Будем полагать $u_1 = 0$.

2. Проверка плана на оптимальность. Для проверки на оптимальность для всех свободных клеток рассчитываются оценки $\Delta c_{ij} = c_{ij} - (u_i + v_j)$ (для **небазисных клеток**). Если встречена хотя бы одна отрицательная оценка, то план не является оптимальным и его следует улучшать за счёт загрузки клетки с отрицательной оценкой. Если в таблице присутствует несколько клеток с отрицательными оценками, то выбираем клетку с наибольшей по модулю оценкой. Если все оценки неотрицательны, то план является оптимальным и задача решена. Если все $\Delta c_{ij} > 0$, то решение единственно, если присутствует оценка $\Delta c_{ij} = 0$, то задача имеет бесконечное множество решений, дающих одно и то же значение целевой функции.

3. Перераспределение поставок. Если таблица не является оптимальной, то улучшаем её путём перераспределения поставок. Строим цикл (Определение б), в который входит выбранная в предыдущем пункте оценка, а остальные клетки в нём – базисные. Цикл всегда существует и единственен. Помечаем вершины цикла поочерёдно знаками «+» и «-» и поочерёдно расставляем знаки, начиная со знака «+», начиная с рассматриваемой оценки. Среди клеток цикла, помеченных знаком «-», выбираем минимальное значение груза и прибавляем его в клетки, помеченные знаком «+» и вычитаем из клеток, помеченных знаком «-».

4. Получилась таблица с перераспределёнными поставками. Важно помнить, что количество базисных клеток должно быть равно $m + n - 1$. Если это не так, то одну из свободных клеток превращаем в базисную, определив в ней нулевое количество груза. Так как целью задачи является минимизация стоимости перевозки груза, то целесообразнее превратить в базисную клетку свободную клетку с наименьшей стоимостью перевозки. Когда количество базисных клеток стало равно $m + n - 1$, проверяем таблицу на оптимальность. Повторяем все описанные выше шаги до тех пор, пока не получится оптимальная таблица.

Решение транспортной задачи (метод потенциалов). Примеры

Вернёмся к рассмотрению Примера 20, в котором была рассмотрена транспортная задача и построен начальный опорный план методом северо-западного угла:

Пункты отправления	Пункты назначения				Запасы
	B_1	B_2	B_3	B_4	
A_1	2 140	3	4	2	0
A_2	8 10	4 90	1 60	4	0
A_3	9	7	3 40	6 80	0
Потребности	0	0	0	0	

Таблица 45. Начальный опорный план

Стоимость перевозки: $F = 1380$.

Продолжим решение методом потенциалов.

Пример 21 (метод потенциалов для Примера 20).

1. Добавляем в таблицу столбцы потенциалов:

Пункты отправления	Пункты назначения				U
	B_1	B_2	B_3	B_4	
A_1	2 140	3	4	2	0
A_2	8 10	4 90	1 60	4	6
A_3	9	7	3 40	6 80	8
V	2	-2	-5	-2	

Таблица 46. Метод потенциалов. Введение потенциалов

Для определения потенциалов составляем по базисным клеткам систему $(u_i + v_j = c_{ij})$:

$$\begin{cases} u_1 + v_1 = 2 \\ u_2 + v_1 = 8 \\ u_2 + v_2 = 4 \\ u_2 + v_3 = 1 \\ u_3 + v_3 = 3 \\ u_3 + v_4 = 6 \end{cases}$$

Пусть $u_1 = 0$ (такое предположение будем делать всегда), тогда $v_1 = 2$; $u_2 = 6$; $v_2 = -2$; $v_3 = -5$; $u_3 = 8$; $v_4 = -2$.

2. Для свободных клеток рассчитаем оценки $\Delta c_{ij} = c_{ij} - (u_i + v_j)$ (для небазисных клеток) и запишем их в левый нижний угол (выделены полужирным шрифтом):

Пункты отправления	Пункты назначения				U
	B_1	B_2	B_3	B_4	
A_1	2 140	3 5	4 9	2 4	0
A_2	8 10	4 90	1 60	4 0	6
A_3	9 -1	7 1	3 40	6 80	8
V	2	-2	-5	-2	

Таблица 47. Метод потенциалов. Расчёт оценок

В таблице присутствует отрицательная оценка, значит, полученное решение не является оптимальным и его можно улучшить, перераспределив поставки.

3. Выделяем цикл. В него берём клетку с отрицательной оценкой и, согласно определению цикла, базисные клетки. Цикл выделен в Таблице 46. Помечаем клетку с отрицательной оценкой знаком «+», и для остальных клеток цикла чередуем знаки с «+» на «-»:

Пункты отправления	Пункты назначения				U
	B_1	B_2	B_3	B_4	
A_1	2 140	3 5	4 9	2 4	0
A_2	8 10 -	4 90	1 + 60	4 0	6
A_3	9 -1 +	7 1 -	3 - 40	6 80	8
V	2	-2	-5	-2	

Таблица 48. Метод потенциалов. Цикл

Переходим к новому решению. Для этого выбираем наименьшее значение груза в клетках, помеченных знаком «-», это 10. Вычитаем это значение из клеток цикла со знаком «-» и прибавляем его к клеткам цикла, помеченным знаком «+». Данное перераспределение груза будем записывать в новой таблице, оценки и потенциалы с прошлого шага не переписываем, они будут

рассчитаны заново для получившегося решения. Получили, что клетка, стоящая на пересечении A_2 и B_1 стала свободной.

4. Проверяем количество базисных клеток, их действительно $4 + 3 - 1 = 6$. Получилось новое решение. Проверим его на оптимальность. Для этого введём в рассмотрение потенциалы и рассчитаем оценки (выполнить самостоятельно аналогично шагам 1,2):

Пункты отправления	Пункты назначения				U
	B_1	B_2	B_3	B_4	
A_1	2 140	3 4	4 8	2 3	0
A_2	8 1	4 90	1 70	4 0	5
A_3	9 10	7 1	3 30	6 80	7
V	2	-1	-4	-1	

Таблица 49. Метод потенциалов. Перераспределение поставок

Все оценки неотрицательны, значит таблица является оптимальной и можно выписывать решение.

Получен следующий план перевозок:

$$X = \begin{pmatrix} 140 & 0 & 0 & 0 \\ 0 & 90 & 70 & 0 \\ 10 & 0 & 30 & 80 \end{pmatrix}$$

Рассчитаем стоимость данной перевозки: $F = 2 \cdot 140 + 9 \cdot 10 + 4 \cdot 90 + 1 \cdot 70 + 3 \cdot 30 + 6 \cdot 80 = 1370$.

Ответ: план перевозок: $X = \begin{pmatrix} 140 & 0 & 0 & 0 \\ 0 & 90 & 70 & 0 \\ 10 & 0 & 30 & 80 \end{pmatrix}$; стоимость перевозки $F = 1370$.

Рассмотрим следующий пример.

Пример 22. Решить транспортную задачу:

Пункты отправления	Пункты назначения					Запасы
	B_1	B_2	B_3	B_4	B_5	
A_1	3 x_{11}	7 x_{12}	1 x_{13}	5 x_{14}	4 x_{15}	30
A_2	7 x_{21}	5 x_{22}	8 x_{23}	6 x_{24}	3 x_{25}	5
A_3	6 x_{31}	4 x_{32}	8 x_{33}	3 x_{34}	2 x_{35}	45
A_4	3 x_{41}	1 x_{42}	7 x_{43}	4 x_{44}	2 x_{45}	70
Потребности	10	35	45	25	35	

Таблица 50. Условие транспортной задачи

1. Проверяем, является ли модель закрытой. Суммарные потребности 150; суммарные запасы 150. Суммарные потребности совпадают с суммарными запасами, переходим к построению начального опорного плана.

2. Строим начальный опорный план методом минимального элемента.

Пункты отправления	Пункты назначения					Запасы
	B_1	B_2	B_3	B_4	B_5	
A_1	3 30	7	1 30	5	4	0
A_2	7	5	8 5	6	3	0
A_3	6	4	8	3 10	2 35	0
A_4	3 10	1 35	7 10	4 15	2	0
Потребности	0	0	0	0	0	

Таблица 51. Начальный опорный план

Получен следующий план перевозок:

$$X = \begin{pmatrix} 0 & 0 & 30 & 0 & 0 \\ 0 & 0 & 5 & 0 & 0 \\ 0 & 0 & 0 & 10 & 35 \\ 10 & 35 & 10 & 15 & 0 \end{pmatrix}$$

Рассчитаем стоимость данной перевозки: $F = 3 \cdot 10 + 1 \cdot 35 + 1 \cdot 30 + 8 \cdot 5 + 7 \cdot 10 + 3 \cdot 10 + 4 \cdot 15 + 2 \cdot 35 = 365$.

3. Переходим к методу потенциалов.

Пункты отправления	Пункты назначения					U
	B_1	B_2	B_3	B_4	B_5	
A_1	3 6	7 12	1 30	5 7	4 7	0
A_2	7 3	5 3	8 5	6 1	3 -1	7
A_3	6 4	4 4	8 2	3 + 10	2 35 -	5
A_4	3 10	1 35	7 10	4 - 15	2 -1 +	6
V	-3	-5	1	-2	-3	

Таблица 52. Метод потенциалов

Пункты отправления	Пункты назначения					U
	B_1	B_2	B_3	B_4	B_5	
A_1	3 6	7 12	1 30	5 8	4 8	0
A_2	7 3	5 3	8 5	6 2	3 0	7
A_3	6 3	4 3	8 1	3 25	2 20	6
A_4	3 10	1 35	7 10	4 1	2 15	6
V	-3	-5	1	-3	-4	

Таблица 53. Метод потенциалов

Таблица является оптимальной. Получен следующий план перевозок:

$$X = \begin{pmatrix} 0 & 0 & 30 & 0 & 0 \\ 0 & 0 & 5 & 0 & 0 \\ 0 & 0 & 0 & 25 & 20 \\ 10 & 35 & 10 & 0 & 15 \end{pmatrix}$$

Рассчитаем стоимость данной перевозки: $F = 3 \cdot 10 + 1 \cdot 35 + 1 \cdot 30 + 8 \cdot 5 + 7 \cdot 10 + 3 \cdot 25 + 2 \cdot 20 + 2 \cdot 15 = 350$.

Ответ: Оптимальный план перевозок:

$$X = \begin{pmatrix} 0 & 0 & 30 & 0 & 0 \\ 0 & 0 & 5 & 0 & 0 \\ 0 & 0 & 0 & 25 & 20 \\ 10 & 35 & 10 & 0 & 15 \end{pmatrix}$$

Стоимость данной перевозки $F = 350$.

Перейдём к рассмотрению ещё одного примера решения транспортной задачи, рассмотрим открытую модель.

Пример 23. Решить транспортную задачу:

Транспортная задача задана в виде таблицы. Построим начальный опорный план методом северо-западного угла.

Пункты отправления	Пункты назначения				Запасы
	B_1	B_2	B_3	B_4	
A_1	2 x_{11}	5 x_{12}	3 x_{13}	4 x_{14}	45
A_2	6 x_{21}	1 x_{22}	2 x_{23}	5 x_{24}	35
A_3	3 x_{31}	4 x_{32}	3 x_{33}	8 x_{34}	70
Потребности	20	60	55	45	

Таблица 54. Условие транспортной задачи. Открытая модель

1. Проверяем, является ли модель закрытой. Суммарные потребности 180; суммарные запасы 150. Суммарные потребности не совпадают с суммарными запасами, значит, модель является открытой. Чтобы сделать модель закрытой, следует ввести в рассмотрение фиктивный пункт отправления A_4 с запасом груза $180 - 150 = 30$. Тариф на перевозку груза из фиктивного пункта отправления полагаем равным нулю. Получаем следующую модель, которая уже является закрытой:

Пункты отправления	Пункты назначения				Запасы
	B_1	B_2	B_3	B_4	
A_1	2 x_{11}	5 x_{12}	3 x_{13}	4 x_{14}	45
A_2	6 x_{21}	1 x_{22}	2 x_{23}	5 x_{24}	35
A_3	3 x_{31}	4 x_{32}	3 x_{33}	8 x_{34}	70
A_4	0 x_{41}	0 x_{42}	0 x_{43}	0 x_{44}	30
Потребности	20	60	55	45	

Таблица 55. Переход от открытой модели к закрытой

2. Строим начальный опорный план методом минимального элемента

Пункты отправления	Пункты назначения				Запасы
	B_1	B_2	B_3	B_4	
A_1	2 20	5	3 25	4	0
A_2	6	1 35	2	5	0
A_3	3	4 25	3 30	8 15	0
A_4	0	0	0	0 30	0
Потребности	0	0	0	0	

Таблица 56. Начальный опорный план

Получен следующий план перевозок:

$$X = \begin{pmatrix} 20 & 0 & 25 & 0 \\ 0 & 35 & 0 & 0 \\ 0 & 25 & 30 & 15 \end{pmatrix}$$

и при этом потребность 4-го пункта назначения в грузе не будет полностью удовлетворена, не хватает 30 единиц груза.

Стоимость данной перевозки: $F = 2 \cdot 20 + 1 \cdot 35 + 4 \cdot 25 + 3 \cdot 25 + 3 \cdot 30 + 8 \cdot 15 = 460$.

3. Переходим к методу потенциалов:

Пункты отправления	Пункты назначения				U
	B_1	B_2	B_3	B_4	
A_1	2 20	5 1	3 - 25	4 -4 +	0
A_2	6 7	1 35	2 2	5 0	-3
A_3	3 1	4 25	3 + 30	8 15 -	0
A_4	0 6	0 4	0 5	0 30	-8
V	2	4	3	8	

Таблица 57. Метод потенциалов

Пункты отправления	Пункты назначения				U
	B_1	B_2	B_3	B_4	
A_1	2 20	5 1	3 10	4 15	0
A_2	6 7	1 35	2 2	5 4	-3
A_3	3 1	4 25	3 45	8 4	0
A_4	0 2	0 0	0 1	0 30	-4
V	2	4	3	4	

Таблица 58. Метод потенциалов

Полученная таблица является оптимальной. План перевозок имеет вид:

$$X = \begin{pmatrix} 20 & 0 & 10 & 15 \\ 0 & 35 & 0 & 0 \\ 0 & 25 & 45 & 0 \end{pmatrix}$$

и при этом потребность 4-го пункта назначения в грузе не будет полностью удовлетворена, не хватает 30 единиц груза.

Рассчитаем стоимость данной перевозки: $F = 2 \cdot 20 + 1 \cdot 35 + 4 \cdot 25 + 3 \cdot 10 + 3 \cdot 45 + 4 \cdot 15 = 400$.

Ответ: план перевозок: $X = \begin{pmatrix} 20 & 0 & 10 & 15 \\ 0 & 35 & 0 & 0 \\ 0 & 25 & 45 & 0 \end{pmatrix}$

и при этом потребность 4-го пункта назначения в грузе не будет полностью удовлетворена, не хватает 30 единиц груза. Стоимость данной перевозки $F = 400$.

10. Решение задачи линейного программирования с помощью системы компьютерной алгебры Maple

В системе компьютерной алгебры Maple для решения задачи линейного программирования используется пакет *simplex*. Подключаем этот пакет:

`>with(simplex):`

Нас будут интересовать функции *maximize* и *minimize*, позволяющие найти соответственно максимум и минимум функции симплекс-методом. Они имеют следующий вид:

`maximize(f,C)`

`maximize(f,C,vartype)`

`minimize(f,C)`

`minimize(f,C,vartype)`

Здесь f —целевая функция, C — множество ограничений, $vartype$ — тип переменных (принимает два значения: NONNEGATIVE и UNRESTRICTED, необязательный параметр).

Рассмотрим пример решения задачи линейного программирования средствами системы компьютерной алгебры Maple.

Пример 24. Решить задачу линейного программирования:

$$\begin{aligned} f(x) &= 3x_1 + 3x_2 \rightarrow \max \\ x_1 + 2x_2 &\leq 8 \\ 2x_1 - x_2 &\geq 1 \\ x_1 - 2x_2 &\leq 2 \\ x_1 \geq 0, x_2 &\geq 0 \end{aligned}$$

```

> restart;
  with(simplex) :

> f:= 3· x1 + 3· x2;

> maximize(f, {x1 + 2· x2 ≤ 8, 2· x1 - x2 ≥ 1, x1 - 2· x2 ≤ 2}, NONNEGATIVE);
                                     {x1 = 5, x2 = 3/2}

> #подставим оптимальное решение в целевую функцию
  subs(%, f);

                                     39
                                     2

```

Получили оптимальное решение $x_1 = 5, x_2 = \frac{3}{2}$, оптимальное значение целевой функции $\frac{39}{2}$.

Также для решения задачи линейного программирования можно использовать пакет *Optimization*. Это новый пакет, использующий улучшенные алгоритмы оптимизации. Подключаем этот пакет:

>with(Optimization):

Нас будет интересовать функция *LPSolve*, предназначенная для решения задачи линейного программирования. Она имеет следующий вид:

LPSolve(obj, constr, 'type')

Здесь *obj* –целевая функция, *constr* – множество ограничений, 'type' –тип задачи ('maximize' или 'minimize').

Рассмотрим пример решения задачи линейного программирования средствами системы компьютерной алгебры Maple с использованием пакета *Optimization*. Обратим внимание на графическую визуализацию решения.

Пример 25. Решить задачу линейного программирования:

$$\begin{aligned}
 f(x) &= 3x_1 + 3x_2 \rightarrow \max \\
 x_1 + 2x_2 &\leq 8 \\
 2x_1 - x_2 &\geq 1 \\
 x_1 - 2x_2 &\leq 2 \\
 x_1 \geq 0, x_2 &\geq 0
 \end{aligned}$$

Подключаем необходимые библиотеки, задаём количество знаков после запятой.

```

> restart;
> with(plots) :
  with(Optimization);
  Digits := 16;

```

Целевая функция и набор ограничений:

```

> f := 3·x1 + 3·x2;
  init := [x1 + 2·x2 ≤ 8, 2·x1 - x2 ≥ 1, x1 - 2·x2 ≤ 2, x1 ≥ 0, x2 ≥ 0] :

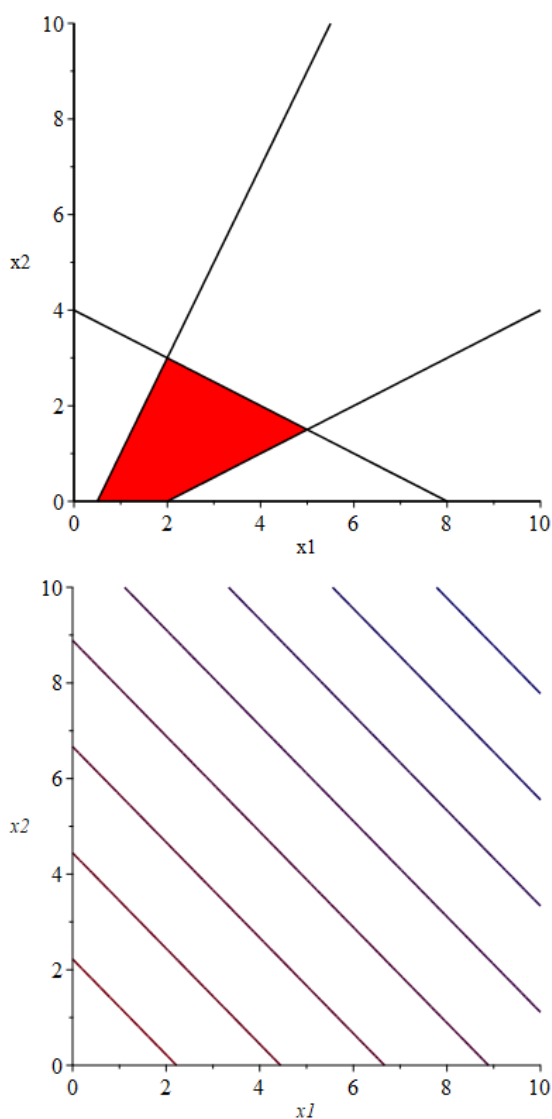
```

Изображаем графически ОДР и линии уровня:

```

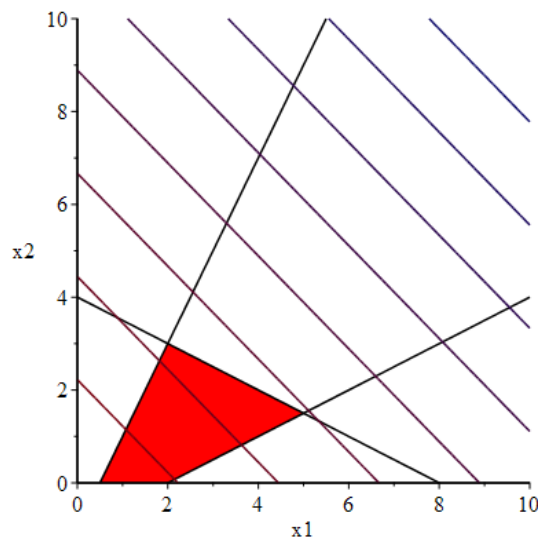
> p1 := inequal(init, x1 = 0 .. 10, x2 = 0 .. 10, optionsfeasible = (color = red));
  p2 := contourplot(f, x1 = 0 .. 10, x2 = 0 .. 10);

```



Объединим ОДР и линии уровня:

```
> display([p1, p2]);
```



Максимизируем целевую функцию на построенном множестве:

```
> LPSolve(f, init,'maximize');
```

```
[19.500000000000000, [x1 = 5.000000000000000, x2 = 1.500000000000000]]
```

Здесь была построена область допустимых решений, линии уровня и найдено оптимальное решение $x_1 = 5, x_2 = 1.5$, оптимальное значение целевой функции 19.5.

11. Решение транспортной задачи с помощью системы компьютерной алгебры Maple.

Транспортная задача является задачей линейного программирования и может быть решена в системе компьютерной алгебры Maple с помощью описанного ранее пакета *simplex* и функции *minimize*. Рассмотрим решение транспортной задачи на следующем примере.

Пример 26. Решить транспортную задачу:

Пункты отправления	Пункты назначения					Запасы
	B_1	B_2	B_3	B_4	B_5	
A_1	3 x_{11}	7 x_{12}	1 x_{13}	5 x_{14}	4 x_{15}	30
A_2	7 x_{21}	5 x_{22}	8 x_{23}	6 x_{24}	3 x_{25}	5
A_3	6 x_{31}	4 x_{32}	8 x_{33}	3 x_{34}	2 x_{35}	45
A_4	3 x_{41}	1 x_{42}	7 x_{43}	4 x_{44}	2 x_{45}	70
Потребности	10	35	45	25	35	

Таблица 59. Условие транспортной задачи

Пусть функция f – суммарная стоимость перевозки, которую согласно условию задачи, следует минимизировать; $init$ – набор ограничений, отражающий суммарные потребности каждого пункта назначения и суммарные запасы каждого пункта отправления.

```

> restart;
> with(simplex) :
> f:= 3·x11 + 7·x12 + 1·x13 + 5·x14 + 4·x15 + 7·x21 + 5·x22
    + 8·x23 + 6·x24 + 3·x25 + 6·x31 + 4·x32 + 8·x33 + 3·x34 + 2·x35 +
    3·x41 + 1·x42 + 7·x43 + 4·x44 + 2·x45 :
> init := {x11 + x21 + x31 + x41 = 10, x12 + x22 + x32 + x42 = 35,
    x13 + x23 + x33 + x43 = 45, x14 + x24 + x34 + x44 = 25,
    x15 + x25 + x35 + x45 = 35, x11 + x12 + x13 + x14 + x15 = 30,
    x21 + x22 + x23 + x24 + x25 = 5, x31 + x32 + x33 + x34 + x35 = 45,
    x41 + x42 + x43 + x44 + x45 = 70} :

```

Таким образом, возникает задача минимизации функции f при условии ограничений $init$.

```

> minimize(f, init, NONNEGATIVE);

```

Решение – элементы матрицы перевозок $x_{ij}, i = 1, \dots, 4; j = 1, \dots, 5$.

$\{x_{11} = 0, x_{12} = 0, x_{13} = 30, x_{14} = 0, x_{15} = 0, x_{21} = 0, x_{22} = 0, x_{23} = 0, x_{24} = 0,$
 $x_{25} = 5, x_{31} = 0, x_{32} = 0, x_{33} = 0, x_{34} = 25, x_{35} = 20, x_{41} = 10, x_{42} = 35, x_{43} =$
 $15, x_{44} = 0, x_{45} = 10\}$

Подставим получившееся решение в функцию f для определения наименьшей стоимости перевозок.

```
> subs(%, f);
```

350

12. Решение задачи линейного программирования на Python

Для решения оптимизационных задач будем использовать пакет *SciPy*, предназначенный для решения математических задач. Нас будет интересовать его внутренний пакет *optimize*, предназначенный для решения оптимизационных задач и функция *linprog* данного пакета, предназначенная для решения задач линейного программирования.

Рассмотрим решение задачи линейного программирования на Python на следующем примере.

Пример 27. Решить задачу линейного программирования:

$$\begin{aligned} f(x) &= 3x_1 + 3x_2 \rightarrow \max \\ x_1 + 2x_2 &\leq 8 \\ 2x_1 - x_2 &\geq 1 \\ x_1 - 2x_2 &\leq 2 \\ x_1 \geq 0, x_2 &\geq 0 \end{aligned}$$

Так как *linprog* решает только задачи минимизации и не допускает ограничений вида $\geq$, то преобразуем исходную задачу к виду:

$$\begin{aligned} -f(x) &= -3x_1 - 3x_2 \rightarrow \min \\ x_1 + 2x_2 &\leq 8 \\ -2x_1 + x_2 &\leq -1 \\ x_1 - 2x_2 &\leq 2 \\ x_1 \geq 0, x_2 &\geq 0 \end{aligned}$$

Подключаем пакет *SciPy*, и функцию *linprog* из *scipy.optimize*:

```
import scipy
from scipy.optimize import linprog
```

Задаём коэффициенты целевой функции при переменных x_1, x_2 :

```
funct = [-3, -3]
```

Коэффициенты ограничений при переменных x_1, x_2 :

```
left_ineq = [[ 1, 2], [-2, 1], [ 1, -2]]
```

Правые части ограничений:

```
right_ineq = [8, -1, 2]
```

Задаём ограничения на x_1, x_2 :

```
var = [(0, float("inf")), (0, float("inf"))]
```

Решаем задачу симплекс-методом:

```

opt = linprog(c=funct, A_ub=left_ineq, b_ub=right_ineq,
              bounds=var,
              method="simplex")

opt

```

Получаем следующее решение:

```

con: array([], dtype=float64)
fun: -19.5
message: 'Optimization terminated successfully.'
nit: 3
slack: array([0. , 7.5, 0. ])
status: 0
success: True
x: array([5. , 1.5])

```

Оптимальное решение (5,1.5), оптимальное значение следует умножить на -1 и получим 19.5.

`linprog()` возвращает структуру данных со следующими атрибутами:

- `.con` – остатки ограничения-равенства;
- `.fun` – оптимальное значение целевой функции (если найдено);
- `.message` – словесный статус решения;
- `.nit` – количество итераций, необходимых для завершения расчета;
- `.slack` – значения так называемых дополнительных переменных – разниц между значениями левой и правой сторонами ограничений;
- `.status` – целое число от 0 до 4, отражающих результат решения:
0 – поиск оптимального решения завершился успешно;
1 – достигнут лимит на число итераций;
2 – задача не имеет решений;
3 – целевая функция не ограничена;
- `.success` – логическое значение, показывающее, найдено ли оптимальное решение;
- `.x` – массив NumPy, содержащий оптимальные значения переменных решения.

13. Решение транспортной задачи на Python

Для решения транспортной задачи также будем использовать пакет *SciPy*, предназначенный для решения математических задач. Нас будет интересовать его внутренний пакет *optimize*, предназначенный для решения оптимизационных задач и функция *linprog* данного пакета, предназначенная для решения задач линейного программирования.

Рассмотрим решение транспортной задачи на Python на следующем примере.

Пример 28. Решить транспортную задачу:

Пункты отправления	Пункты назначения					Запасы
	B_1	B_2	B_3	B_4	B_5	
A_1	3 x_{11}	7 x_{12}	1 x_{13}	5 x_{14}	4 x_{15}	30
A_2	7 x_{21}	5 x_{22}	8 x_{23}	6 x_{24}	3 x_{25}	5
A_3	6 x_{31}	4 x_{32}	8 x_{33}	3 x_{34}	2 x_{35}	45
A_4	3 x_{41}	1 x_{42}	7 x_{43}	4 x_{44}	2 x_{45}	70
Потребности	10	35	45	25	35	

Таблица 60. Условие транспортной задачи

Задаём вектор тарифов c , разбиваем массив переменных по строкам *line* и по столбцам *column*, задаём запасы A и потребности B . Минимизируем затраты на перевозку с помощью функции *linprog*.

```
import scipy
from scipy.optimize import linprog
c = [3,7,1,5,4,7,5,8,6,3,6,4,8,3,2,3,1,7,4,2]
line = [[1,1,1,1,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0],
         [0,0,0,0,0,1,1,1,1,1,0,0,0,0,0,0,0,0,0,0],
         [0,0,0,0,0,0,0,0,0,0,1,1,1,1,1,0,0,0,0,0],
         [0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,1,1,1]]
A = [30,5,45,70]
column = [[1,0,0,0,0,1,0,0,0,0,1,0,0,0,0,1,0,0,0,0],
          [0,1,0,0,0,0,1,0,0,0,0,1,0,0,0,0,1,0,0,0],
          [0,0,1,0,0,0,0,1,0,0,0,0,1,0,0,0,0,1,0,0],
          [0,0,0,1,0,0,0,0,1,0,0,0,0,1,0,0,0,0,1,0],
          [0,0,0,0,1,0,0,0,0,1,0,0,0,0,1,0,0,0,0,1]]
B = [10,35,45,25,35]
print(linprog(c, line, A, column, B))
```

Получили следующее решение, соответствующее решениям, полученным ранее:

```

con: array([4.80042672e-11, 2.48022047e-10, 3.27972316e-10, 1.67990066e-10,
2.47986520e-10])
fun: 349.9999999823945
message: 'Optimization terminated successfully.'
nit: 6
slack: array([1.97367456e-10, 1.82076576e-12, 3.20419247e-10, 5.20373078e-10])
status: 0
success: True
x: array([1.22431513e-11, 6.70713495e-12, 3.00000000e+01, 9.95248056e-12,
9.96958435e-12, 3.39260584e-12, 2.37923907e-11, 3.26688384e+00,
2.95746635e-11, 1.73311616e+00, 2.32728236e-11, 2.62069605e-11,
6.94802026e-11, 2.50000000e+01, 2.00000000e+01, 1.00000000e+01,
3.50000000e+01, 1.17331162e+01, 7.29063593e-11, 1.32668838e+01])

```

Задания для самостоятельного решения

1. Решить задачу линейного программирования графическим методом:

$f(x) = x_1 + x_2 \rightarrow \max$ $x_1 + 4x_2 \leq 6$ $2x_1 + x_2 \leq 4$ $x_1 \geq 0$	$f(x) = 3x_1 + 2x_2 \rightarrow \max$ $x_1 + 2x_2 \geq 2$ $x_1 + x_2 \leq 10$ $x_1 \leq 6$ $x_2 \leq 8$ $x \geq 0$
$f(x) = 2x_1 - 3x_2 \rightarrow \min$ $x_1 + x_2 \geq 1$ $4x_1 + 3x_2 \leq 16$ $4x_1 - 3x_2 \geq 0$ $x \geq 0$	$f(x) = x_1 + 3x_2 \rightarrow \max$ $x_1 - x_2 \leq 1$ $2x_1 + x_2 \leq 2$ $x_1 - x_2 \geq 0$ $x \geq 0$
$f(x) = x_1 \rightarrow \max$ $2x_1 + x_2 \leq 2$ $x_1 + x_2 \leq 1$ $x \geq 0$	$f(x) = -x_1 + x_2 \rightarrow \max$ $-x_1 + x_2 \leq 2$ $2x_1 + x_2 \geq 2$ $x \geq 0$
$f(x) = 3x_2 \rightarrow \max$ $x_1 + 2x_2 \geq 2$ $x_1 + x_2 \leq 7$ $x_2 \leq 4$ $x_1 - 2x_2 \leq 4$ $5x_1 - 2x_2 \geq 0$ $x \geq 0$	$f(x) = 3x_1 + 3x_2 \rightarrow \max$ $x_1 + x_2 \leq 0.25$ $2x_1 - x_2 \geq 1$ $x_1 - 2x_2 \leq 2$ $x \geq 0$

$f(x) = 2x_2 \rightarrow \max$ $x_1 + x_2 \leq 6$ $x_1 - 2x_2 \leq 4$ $x_2 \leq 3$ $x \geq 0$	$f(x) = 5x_1 + 3x_2 \rightarrow \max$ $x_1 + 3x_2 \leq 6$ $3x_1 + x_2 \leq 6$ $-x_1 + 2x_2 \leq 1$ $x \geq 0$
---	---

2. Решить задачу линейного программирования симплекс-методом:

$f(x) = 2x_1 + 4x_2 + x_3 \rightarrow \max$ $x_1 + 2x_2 + x_3 = 6$ $-x_1 + x_2 \leq 2$ $x \geq 0$	$f(x) = 2x_1 + x_2 \rightarrow \max$ $2x_1 + 3x_2 \leq 6$ $x_1 \leq 2$ $x \geq 0$
$f(x) = 10x_1 - 7x_2 - 5x_3 \rightarrow \min$ $6x_1 + 15x_2 + 6x_3 \leq 9$ $14x_1 + 42x_2 + 16x_3 \leq 21$ $2x_1 + 8x_2 + 2x_3 \leq 4$ $x \geq 0$	$f(x) = x_4 - 2x_5 + 1 \rightarrow \min$ $x_1 - x_4 + x_5 = 2$ $x_2 - 2x_4 + 3x_5 = 7$ $x_3 + x_4 - 2x_5 = 1$ $x \geq 0$
$f(x) = 3x_1 + 2x_2 \rightarrow \max$ $-4x_1 + 3x_2 \leq 8$ $x_1 - 4x_2 \leq 2$ $2x_1 + 3x_2 \leq 12$ $x \geq 0$	$f(x) = 6x_1 + 5x_2 \rightarrow \max$ $3x_1 + x_2 \leq 9$ $x_1 + 3x_2 \leq 6$ $x_1 - 3x_2 \leq 2$ $x \geq 0$
$f(x) = 4x_1 + 5x_2 \rightarrow \max$ $5x_1 + 4x_2 \leq 15$ $x_1 + 2x_2 \leq 7$ $2x_1 - 3x_2 \leq 3$ $x \geq 0$	$f(x) = 3x_1 + 2x_2 \rightarrow \max$ $5x_1 + x_2 \leq 10$ $2x_1 + 3x_2 \leq 12$ $-2x_1 + x_2 \leq 3$ $x \geq 0$
$f(x) = x_1 + 2x_2 \rightarrow \max$ $5x_1 - 2x_2 \leq 4$ $-x_1 + 2x_2 \leq 8$ $x_1 + 2x_2 \leq 10$ $x \geq 0$	$f(x) = 6x_1 + 5x_2 \rightarrow \max$ $2x_1 + 5x_2 \leq 10$ $5x_1 + 2x_2 \leq 10$ $2x_1 - x_2 \leq 2$ $x \geq 0$

3. Решить методом искусственного базиса:

$f(x) = 3x_1 + 2x_2 - 3x_3 \rightarrow \min$ $2x_1 - x_2 + x_3 \leq 12$ $2x_1 - x_2 - 2x_3 = 3$ $-x_1 + 2x_2 + x_3 \geq 6$ $x \geq 0$	$f(x) = x_1 + x_2 \rightarrow \max$ $-x_1 + x_2 \leq 2$ $2x_1 + x_2 \geq 2$ $x \geq 0$
$f(x) = x_1 + 4x_2 - 7x_3 \rightarrow \max$ $3x_1 - 2x_2 + 14x_3 = 2$ $x_1 - 2x_2 + 10x_3 = 0$ $x \geq 0$	$f(x) = x_1 + x_2 + x_3 \rightarrow \max$ $-2x_1 + x_2 + x_3 = 4$ $-x_1 + 2x_2 - x_3 = -1$ $x \geq 0$
$f(x) = -x_1 + 2x_2 + 4x_3 \rightarrow \max$ $x_1 - x_2 - x_3 = 1$ $x_1 + x_2 + x_3 = 3$ $x \geq 0$	$f(x) = 5x_1 + x_2 - 2x_3 \rightarrow \max$ $2x_1 - 3x_2 + x_3 = 2$ $-x_1 + 5x_2 + x_3 = -4$ $x \geq 0$
$f(x) = x_1 - 3x_2 + 2x_3 \rightarrow \max$ $-2x_1 + 3x_2 + x_3 = 0$ $-3x_1 + 2x_2 - 6x_3 = -1$ $x \geq 0$	$f(x) = 3x_1 + x_2 - x_3 \rightarrow \max$ $5x_1 + 2x_2 - x_3 = 4$ $3x_1 - 2x_2 + 5x_3 = 2$ $x \geq 0$
$f(x) = 5x_1 + 4x_2 - x_3 \rightarrow \max$ $3x_1 - 2x_2 + 7x_3 = 4$ $2x_1 + x_2 + x_3 = 3$ $x \geq 0$	$f(x) = 2x_1 - 3x_2 + x_3 \rightarrow \max$ $2x_1 - 3x_2 + x_3 = 0$ $-x_1 + 2x_2 - x_3 = 1$ $x \geq 0$

4. Определить, при каких значениях параметров a и b имеет место:

а) Неразрешимость 1

б) Неразрешимость 2

в) задача имеет бесконечное количество оптимальных решений

г) задача имеет единственное оптимальное решение

$f(x) = x_1 + 3x_2 \rightarrow \max$ $2x_1 + x_2 \leq a$ $x_1 + bx_2 \geq 1$ $x \geq 0$
$f(x) = -x_1 + x_2 \rightarrow \max$ $-x_1 + ax_2 \leq 2$ $2x_1 + x_2 \geq b$ $x \geq 0$

5. В №1 заменить в целевой функции задачи коэффициенты при x_1 и x_2 на α_1 и α_2 соответственно и определить, при каких значениях данных параметров найденное оптимальное решение будет оставаться оптимальным (использовать теорию двойственности).

6. Проверить, используя теорию двойственности линейного программирования, являются ли точки $A(4; 1; 2), B(2; 0; 2)$ оптимальными решениями задачи линейного программирования:

$$\begin{aligned} f(x) &= 3x_1 + x_2 + x_3 \rightarrow \max \\ 2x_1 + x_2 + 3x_3 &\leq 10 \\ 2x_2 + x_3 &\leq 6 \\ x &\geq 0 \end{aligned}$$

7. Проверить, используя теорию двойственности линейного программирования, являются ли точки $A(4; 4), B(0; 1)$ оптимальными решениями задачи линейного программирования:

$$\begin{aligned} f(x) &= -x_1 + 2x_2 \rightarrow \max \\ x_1 - 3x_2 &\leq 3 \\ -x_1 + x_2 &\leq 1 \\ -x_1 + 2x_2 &\leq 4 \\ x &\geq 0 \end{aligned}$$

8. Для данной задачи линейного программирования составить двойственную задачу, решить её графически и найти решение исходной, применив вторую теорему двойственности:

$$\begin{aligned} f(x) &= 8x_1 + 6x_2 - 3x_3 \rightarrow \max \\ 3x_1 + x_2 - x_3 &\leq 1 \\ x_1 + 2x_2 + x_3 &\leq 1 \\ x &\geq 0 \end{aligned}$$

9. Решить задачу параметрического линейного программирования:

$$\begin{aligned} f(x) &= (2 + \lambda)x_1 + (1 + 2\lambda)x_2 \rightarrow \max \\ -x_1 + 2x_2 &\leq 2 \\ x_1 + x_2 &\leq 4 \\ x_1 &\leq 3 \\ x &\geq 0 \\ \lambda &\in [0; +\infty) \end{aligned}$$

$f(x) = (1 + 2\lambda)x_1 + (-2 + \lambda)x_2 \rightarrow \max$ $x_1 + x_2 \leq 4$ $x_1 - x_2 \leq 1$ $x_1 \leq 2$ $x \geq 0$ $\lambda \in (-\infty; +\infty)$
$f(x) = (-2 + \lambda)x_1 + (-1 + 2\lambda)x_2 \rightarrow \max$ $x_1 + x_2 \geq 1$ $x_1 - x_2 \leq 2$ $x \geq 0$ $\lambda \in (-\infty; +\infty)$

10. Решить транспортную задачу:

Пункты отправления	Пункты назначения				Запасы
	B_1	B_2	B_3	B_4	
A_1	1 x_{11}	8 x_{12}	2 x_{13}	3 x_{14}	30
A_2	4 x_{21}	7 x_{22}	5 x_{23}	1 x_{24}	50
A_3	5 x_{31}	3 x_{32}	4 x_{33}	4 x_{34}	20
Потребности	15	15	40	30	

Пункты отправления	Пункты назначения				Запасы
	B_1	B_2	B_3	B_4	
A_1	2 x_{11}	4 x_{12}	5 x_{13}	1 x_{14}	60
A_2	2 x_{21}	3 x_{22}	9 x_{23}	4 x_{24}	70
A_3	3 x_{31}	4 x_{32}	22 x_{33}	5 x_{34}	20
Потребности	40	30	30	50	

Пункты отправления	Пункты назначения				Запасы
	B_1	B_2	B_3	B_4	
A_1	2 x_{11}	4 x_{12}	3 x_{13}	2 x_{14}	60
A_2	3 x_{21}	1 x_{22}	2 x_{23}	3 x_{24}	65
A_3	5 x_{31}	4 x_{32}	1 x_{33}	5 x_{34}	70
Потребности	40	60	70	25	

Пункты отправления	Пункты назначения				Запасы
	B_1	B_2	B_3	B_4	
A_1	3 x_{11}	2 x_{12}	4 x_{13}	1 x_{14}	50
A_2	2 x_{21}	3 x_{22}	1 x_{23}	5 x_{24}	40
A_3	3 x_{31}	2 x_{32}	7 x_{33}	4 x_{34}	20
Потребности	30	25	35	10	

Пункты отправления	Пункты назначения				Запасы
	B_1	B_2	B_3	B_4	
A_1	10 x_{11}	5 x_{12}	7 x_{13}	4 x_{14}	40
A_2	7 x_{21}	4 x_{22}	9 x_{23}	10 x_{24}	25
A_3	6 x_{31}	14 x_{32}	8 x_{33}	7 x_{34}	35
Потребности	15	40	30	15	

Пункты отправления	Пункты назначения					Запасы
	B_1	B_2	B_3	B_4	B_5	
A_1	3 x_{11}	7 x_{12}	1 x_{13}	5 x_{14}	4 x_{15}	20
A_2	7 x_{21}	5 x_{22}	8 x_{23}	6 x_{24}	3 x_{25}	15
A_3	6 x_{31}	4 x_{32}	8 x_{33}	3 x_{34}	2 x_{35}	55
A_4	3 x_{41}	1 x_{42}	7 x_{43}	4 x_{44}	2 x_{45}	60
Потребности	20	25	45	35	25	

Пункты отправления	Пункты назначения				Запасы
	B_1	B_2	B_3	B_4	
A_1	1 x_{11}	3 x_{12}	3 x_{13}	8 x_{14}	20
A_2	8 x_{21}	6 x_{22}	2 x_{23}	6 x_{24}	20
A_3	7 x_{31}	7 x_{32}	3 x_{33}	8 x_{34}	40
A_4	5 x_{41}	2 x_{42}	4 x_{43}	4 x_{44}	45
Потребности	25	30	40	15	

Пункты отправления	Пункты назначения				Запасы
	B_1	B_2	B_3	B_4	
A_1	2 x_{11}	3 x_{12}	9 x_{13}	7 x_{14}	20
A_2	3 x_{21}	4 x_{22}	6 x_{23}	1 x_{24}	16
A_3	5 x_{31}	1 x_{32}	2 x_{33}	2 x_{34}	14
A_4	4 x_{41}	5 x_{42}	8 x_{43}	1 x_{44}	11
Потребности	16	18	12	15	

Пункты отправления	Пункты назначения				Запасы
	B_1	B_2	B_3	B_4	
A_1	1 x_{11}	3 x_{12}	3 x_{13}	4 x_{14}	50
A_2	5 x_{21}	2 x_{22}	7 x_{23}	5 x_{24}	20
A_3	6 x_{31}	4 x_{32}	8 x_{33}	2 x_{34}	30
A_4	7 x_{41}	1 x_{42}	5 x_{43}	7 x_{44}	20
Потребности	40	30	35	15	

Пункты отправления	Пункты назначения				Запасы
	B_1	B_2	B_3	B_4	
A_1	10 x_{11}	5 x_{12}	7 x_{13}	4 x_{14}	40
A_2	7 x_{21}	4 x_{22}	9 x_{23}	10 x_{24}	25
A_3	6 x_{31}	14 x_{32}	8 x_{33}	7 x_{34}	35
Потребности	15	40	30	15	

СПИСОК ЛИТЕРАТУРЫ

1. Веремеенко, Т.В. Высшая математика. Ч. 4. Математическое программирование [Текст]: учеб.-метод. пособие. в 4 ч. / Т. В. Веремеенко ; под ред. Л. Г. Третьяковой. – 2-е изд., испр., Минск : ГИУСТ БГУ, 2010, 159 стр.
2. Линейное программирование: симплекс-метод и двойственность [Текст]: учебное пособие / О. В. Болотникова, Д. В. Тарасов, Р. В. Тарасов ; М-во образования и науки РФ, Федеральное гос. бюджетное образовательное учреждение высш. проф. образования "Пензенский гос. ун-т" (ПГУ). - Пенза : Изд-во ПГУ, 2015. - 82, [1] с.
3. Зотова С. А., Агишева, Д. К., Светличная, В. Б., Матвеева, Т. А. «Линейное программирование»: учебное пособие / С. А. Зотова, Д. К. Агишева, В. Б. Светличная, Т. А. Матвеева / ВПИ (филиал) ВолгГТУ. – Волгоград, 2009. – 79 с.
4. Сантылова Л.И. Вариационное исчисление и методы оптимизации. Руководство по решению задач. Часть 1. [Электронный ресурс] / Л. И. Сантылова; - URL: <https://bookscat.org/book/506053> (дата обращения 01.09.2022)
5. Самаров К.Л. МАТЕМАТИКА. Учебно-методическое пособие по разделу ТРАНСПОРТНАЯ ЗАДАЧА [Электронный ресурс] / К.Л. Самаров, - URL: <https://www.resolventa.ru/data/metodstud/transproblem.pdf> (дата обращения 01.09.2022)
6. Широкова О.А. Решение задач линейного программирования в пакете Maple [Электронный ресурс] / О.А. Широкова, - URL: https://dspace.kpfu.ru/xmlui/bitstream/handle/net/108784/iton2016_77_81.pdf?sequence=-1 (дата обращения 01.09.2022)
7. Линейное программирование. Практика решения задач оптимизации на Python [Электронный ресурс], - URL: <https://proglib.io/p/lineynoe-programmirovanie-praktika-resheniya-zadach-optimizacii-na-python-2020-11-26> (дата обращения 01.09.2022)

Учебно-методическое издание

И.А. Землякова

Линейное программирование в примерах и задачах

Заказ № ____ от _____ г.

Усл. печ. лист. ____ Уч. изд. л. ____.

Отдел полиграфической, корпоративной и сувенирной продукции
Издательско-полиграфического комплекса КИБИ МЕДИА ЦЕНТРА ЮФУ.
344090, г. Ростов-на-Дону, пр. Стачки, 200/1, тел (863) 243-41-66.