

|  |
| --- |
| МИНОБРНАУКИ РОССИИ |
| Федеральное государственное бюджетное образовательное учреждение  высшего образования  **"МИРЭА - Российский технологический университет"**  **РТУ МИРЭА** |

**Институт** Информационных Технологий

**Кафедра** Вычислительной Техники

**ПРАКТИЧЕСКАЯ РАБОТА №5**

**по дисциплине**

**«Теория принятия решений»**

**Симплексный метод**

Студент группы:ИКБО-14-21 \_\_Даурбеков М.И.\_\_\_\_\_\_ *(Ф. И.О. студента)*

Преподаватель \_\_Железняк Л.М.\_\_

*(Ф.И.О. преподавателя)*

|  |  |  |
| --- | --- | --- |
|  |  |  |

Москва 2023

**СОДЕРЖАНИЕ**

[ВВЕДЕНИЕ 3](#_Toc133282937)

[1 СИМПЛЕКСНЫЙ МЕТОД 4](#_Toc133282938)

[1.1 Постановка задачи 4](#_Toc133282939)

[1.2 Математическая модель задачи 4](#_Toc133282940)

[1.2 Программная реализация симплексного метода 11](#_Toc133282941)

[ЗАКЛЮЧЕНИЕ 12](#_Toc133282942)

[СПИСОК ИНФОРМАЦИОННЫХ ИСТОЧНИКОВ 13](#_Toc133282943)

[ПРИЛОЖЕНИЯ 14](#_Toc133282944)

ВВЕДЕНИЕ

Симплексный метод является одним из самых важных алгоритмов в области оптимизации и математического программирования. Он широко используется в различных областях, таких как экономика, инженерия, физика и другие науки. Суть метода заключается в том, что он ищет оптимальное решение задачи линейного программирования путем последовательного перемещения по угловым точкам многогранника, ограничивающего область допустимых решений задачи.

Симплексный метод может быть использован для решения широкого спектра задач линейного программирования, включая задачи с большим числом переменных и ограничений. Он позволяет найти оптимальное решение задачи за конечное число шагов и является одним из наиболее эффективных алгоритмов в этой области.

Одной из главных проблем, с которой сталкиваются при использовании симплексного метода, является возможность возникновения вырожденности. Это происходит, когда два или более ограничения имеют одинаковые коэффициенты. В таком случае, симплексный метод может зациклиться и не найти оптимальное решение задачи. Однако, существуют различные методы, которые позволяют избежать этой проблемы[1].

Симплексный метод имеет множество приложений в различных областях. Например, он может быть использован для оптимизации производственных процессов, распределения ресурсов, планирования бюджета и других задач. Он также может быть использован для решения задач, связанных с линейной регрессией и анализом данных.

**1 СИМПЛЕКСНЫЙ МЕТОД**

* 1. **Постановка задачи**

Решить прямую ЗЛП с помощью симплексного метода и обратную с помощью теорем двойственности. Определить интервалы устойчивости.

Для изготовления двух видов тары (бочек и ящиков) употребляется два вида древесины. Расход древесины каждого вида на каждое изделие, объём ресурсов и прибыль на единицу изделия заданы в таблице 1.

*Таблица 1 Условия задачи*

|  |  |  |  |
| --- | --- | --- | --- |
| Изделия | Расход древесины | | Прибыль на единицу продукции, ден. ед. |
| I | II |
| Бочки | 0.15 | 0.2 | 1.5 |
| Ящики | 0.2 | 0.1 | 1.2 |
| Объём ресурсов | 60 | 40 |  |

Требуется определить сколько ящиков и бочек должен изготовить завод, чтобы прибыль была максимальной.

**1.2 Математическая модель задачи**

Пусть х1 – количество выпускаемых бочек, х2 – количество выпускаемых ящиков. Прибыль от продажи сырья составит 1.5х1 + 1.2х2, прибыль требуется максимизировать.

Ограничения задачи:

Таким образом, переходим к задаче линейного программирования:

Приведем задачу к канонической форме. Для этого в левые части ограничений вводим дополнительные переменные: x3 ≥ 0, х4 ≥ 0. Эти переменные выбираются так, чтобы они обращали неравенства в равенства.

Построим начальную симплекс-таблицу. Запишем систему в векторной форме:

Векторы 𝐴3, 𝐴4 являются линейно независимыми единичными векторами 3х-мерного пространства и образуют базис этого пространства [2].

Поэтому за базисные переменные выбираем переменные 𝑥3, 𝑥4. Небазисными переменными являются 𝑥1, 𝑥2. Разложение позволяет найти первое базисное допустимое решение.

Для этого свободные переменные 𝑥1, 𝑥2 приравниваем нулю. В результате получим разложение:

Которому соответствует первоначальный опорный план

Для проверки плана 𝑥(0) на оптимальность построим первую симплекс-таблицу. Введем в рассмотрение вектор коэффициентов целевой функции при базисных переменных.

В левый столбец Таблицы 1.2 запишем переменные 𝑥3, 𝑥4 образующие базис, в верхней строке – небазисные переменные 𝑥1, 𝑥2. В строке 𝑐j запишем коэффициенты целевой функции, соответствующие небазисным переменным с1 = 1.5, с2 = 1.2. В столбце запишем коэффициенты целевой функции, соответствующие базисным переменным Столбец, определяемый переменной 𝑥1, состоит из коэффициентов вектора . Аналогично, столбец, определяемый переменной 𝑥2, состоит из коэффициентов вектора . Крайний правый столбец заполняется элементами столбца , в нем же в результате вычислений получаем оптимальный план.

Заполнение f-строки (Таблица 1.3). Найдем относительные оценки ∆1, ∆2 и значение целевой функции 𝑄.

*Таблица 1.2 – Начальная симплекс-таблица задачи о максимальном доходе*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | сj | 1.5 | 1.2 |  |
|  |  | X1 | X2 |  |
| 0 | X3 | 0.15 | 0.12 | 60 |
| 0 | X4 | 0.2 | 0.1 | 40 |
|  | F | -1.5 | -1.2 | 0 |
|  |  | Δ1 | Δ2 | Q |

*Таблица 1.3 – Заполнение f-строки*

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | сj | 1.5 | 1.2 |  |  |
|  |  | X1 | X2 |  |  |
| 0 | X3 | 0.15 | 0.2 | 60 | 60 / 0.15 = 400 |
| 0 | X4 | 0.2 | 0.1 | 40 | 40 / 0.2 = 200 *min* |
|  | f | -1.5 | -1.2 | 0 |  |
|  |  | Δ1 | Δ2 | Q |  |

Для оптимальности опорного решения в задаче на максимум требуется выполнение не отрицательности всех относительных оценок ∆i ≥ 0. Так как оценки ∆1= −1.5, ∆2= −1.2 в f-строке отрицательны, то это свидетельствуют о возможности улучшения полученного решения. Наибольшая по модулю отрицательная оценка ∆1= −1.5. В базис будет включена соответствующая ей небазисная переменная 𝑥1. Составим отношения свободных членов к положительным элементам разрешающего столбца. Данные отношения приведены справа от таблицы. Наименьшему частному соответствует строка с переменной 𝑥4. Эта переменная исключается из базиса. В Таблице 1.3 разрешающий столбец и разрешающая строка выделены. Разрешающим элементом является число 𝑎21 = 0.2.

Далее построим новую симплекс-таблицу. Ниже поэтапно демонстрируется процесс заполнения новой симплекс-таблицы (Таблицы 1.4 ).

*Таблица 1.4 – Новая симплекс-таблица*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | сj | 0 | 1.2 |  |
|  |  | X4 | X2 |  |
| 0 | X3 |  |  |  |
| 1.5 | X1 | 5 |  |  |
|  | f |  |  |  |
|  |  | Δ1 | Δ2 | Q |

В Таблице 1.4 переменные 𝑥1 и 𝑥4 меняются местами вместе с коэффициентами 𝑐𝑗. Разрешающий элемент заменяется на обратный. В Таблице 1.5 элементы разрешающей строки делятся на разрешающий элемент. Элементы разрешающего столбца делятся на разрешающий элемент и меняют знак.

*Таблица 1.5 – Симплекс преобразования*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | сj | 7.5 | 3 |  |
|  |  | X4 | X2 |  |
| 0 | X3 | -0.75 |  |  |
| 1.5 | X1 | 5 | 0.5 | 200 |
|  | F | 7.5 |  |  |
|  |  | Δ1 | Δ2 | Q |

*Таблица 1.6 – Итерация 0*

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | сj | 0 | 1.2 |  |  |
|  |  | X4 | X2 |  |  |
| 0 | X4 | -0.75 | 0.125 | 30 | 30 / 0.125 = 240 |
| 1.5 | X6 | 5 | 0.5 | 200 | 200 / 0.5 = 400 |
|  | f | 7.5 | -0.45 | 300 |  |
|  |  | Δ1 | Δ2 | Q |  |

Остальные элементы (Таблица 1.6) рассчитываются по «правилу прямоугольника».

Базисное решение, которое дает последняя таблица

Это решение не является оптимальным, так как в f-строке имеются отрицательная оценка ∆2.

Выполняем следующую итерацию до тех пор, пока в таблице f-строка не будет отрицательных оценок.

Так как оценка ∆2= -0.45 в f-строке отрицательна, то это свидетельствуют о возможности улучшения полученного решения. Наибольшая по модулю отрицательная оценка ∆1= −0.45. В базис будет включена соответствующая ей небазисная переменная 𝑥2. Составим отношения свободных членов к положительным элементам разрешающего столбца. Данные отношения приведены справа от таблицы. Наименьшему частному соответствует строка с переменной 𝑥4. Эта переменная исключается из базиса. В Таблице 1.6 разрешающий столбец и разрешающая строка выделены. Разрешающим элементом является число 𝑎12 = 0.125.

Далее построим новую симплекс-таблицу. Ниже поэтапно демонстрируется процесс заполнения новой симплекс-таблицы (Таблицы 1.7 ).

*Таблица 1.7 – Новая симплекс-таблица*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | сj | 0 | 0 |  |
|  |  | X4 | X3 |  |
| 1.2 | X2 |  |  |  |
| 1.5 | X1 | 5 |  |  |
|  | F |  |  |  |
|  |  | Δ1 | Δ2 | Q |

В Таблице 1.7 переменные 𝑥3 и 𝑥7 меняются местами вместе с коэффициентами 𝑐𝑗. Разрешающий элемент заменяется на обратный. В Таблице 1.8 элементы разрешающей строки делятся на разрешающий элемент. Элементы разрешающего столбца делятся на разрешающий элемент и меняют знак.

*Таблица 1.8 – Симплекс преобразования*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | сj | 0 | 0 |  |
|  |  | X4 | X3 |  |
| 1.2 | X1 | -6 | 8 | 240 |
| 1.5 | X2 |  | -4 |  |
|  | F |  | 3.6 |  |
|  |  | Δ1 | Δ2 | Q |

*Таблица 1.9 – Итерация 1*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | сj | 0 | 0 |  |
|  |  | X4 | X3 |  |
| 1.2 | X1 | -6 | 8 | 240 |
| 1.5 | X2 | 8 | -4 | 80 |
|  | f | 4.8 | 3.6 | 408 |
|  |  | Δ1 | Δ2 | Q |

Остальные элементы (Таблица 1.9) рассчитываются по «правилу прямоугольника».

Базисное решение, которое дает последняя таблица

Если в последней таблице f-строке не содержит отрицательных оценок, то это свидетельствует об оптимальности полученного решения:

Подставляем базисное решение, которое дает последняя таблица

Где n – количество итераций

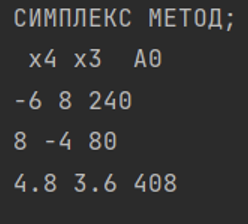
Проверим решение по «правилу прямоугольника».

((0.125\*300) - (-0.45 \* 30)) / (0.125) = 408

Таким образом, фабрика должна выпускать в течении недели 𝑥1 = 240 шт. ящиков и x2 = 80 шт. бочек. Тогда фабрика получит максимальный доход от продажи 456 [ден.ед].

* 1. **Программная реализация симплексного метода**

Результат работы программы представлен на рисунке 1.



**Рисунок 1 – Результат работы программы**

**ЗАКЛЮЧЕНИЕ**

Симплексный метод имеет ряд преимуществ по сравнению с другими методами оптимизации. Во-первых, он гарантирует нахождение оптимального решения за конечное число шагов (если такое решение существует). Во-вторых, он может использоваться для решения задач с большим числом переменных и ограничений. Наконец, симплексный метод может быть применен к различным типам задач, в том числе к задачам с равенствами и неравенствами, а также к задачам с целочисленными переменными.

Однако симплексный метод также имеет некоторые недостатки. В частности, он может столкнуться с проблемой вырожденности, когда две или более переменных имеют одинаковое значение в оптимальной точке. Это может привести к тому, что метод застрянет в локальном минимуме вместо глобального минимума. Есть несколько методов, которые позволяют избежать этой проблемы, например, методы искусственного базиса и методы регуляризации [3].

В целом, симплексный метод остается одним из наиболее эффективных и широко используемых алгоритмов в области оптимизации и математического программирования. Он может быть использован для решения многих практических задач, таких как планирование производства, распределение ресурсов и оптимизация финансовых портфелей.

**СПИСОК ИНФОРМАЦИОННЫХ ИСТОЧНИКОВ**

1. Болотова Л. С. Многокритериальная оптимизация. Болотова Л. С., Сорокин А. Б. [Электронный ресурс] / Метод. указания по вып. курсовой работы — М.: МИРЭА, 2015.
2. Сорокин А. Б. Методы оптимизации: гибридные генетические алгоритмы. Сорокин А. Б. [Электронный ресурс] / Учебно-метод. пособие — М.: МИРЭА, 2016.
3. Сорокин А. Б. Линейное программирование: практикум. Сорокин А. Б., Бражникова Е. В., Платонова О. В. [Электронный ресурс] / Учебно-метод. пособие — М.: МИРЭА, 2017.

**ПРИЛОЖЕНИЯ**

Приложение А – Код реализации симплексного метода на языке python

**Приложение А**

Код реализации симплексного метода на языке python

*Листинг А.1. Реализация симплексного метда.*

st=2

n = st + 1

sv=2

k = sv + n

mas = [[0.15, 0.2, 1, 0, 60], [0.2, 0.1, 0, 1, 40], [-1.5, -1.2, 0, 0, 0]]

print("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*")

print(" Таблица: ")

for i in range(n):

for j in range(k):

print(" "+str(mas[i][j]),end = ' ')

print(" ")

print("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*")

kpol=0

for j in range (sv):

if (mas[n-1][j] > 0):

kpol+=1

print("целевая функция равняется "+str(mas[n-1][k-1])+"\n")

count=0

while (kpol != sv):

print("\nИТЕРАЦИЯ "+str(count)+"\n")

kpol = 0

max = 0

for j in range (sv):

if (abs(mas[n-1][j]) > abs(mas[n-1][max])):

max = j

print("этот столбец ведущий "+str(mas[n-1][max]))

t=mas[n-1][max]

print("\n\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\n")

myn=0

r=0

z=(mas[myn][k-1]) / (mas[myn][max])

for i in range(st):

if (mas[i][max])>0:

r= (mas[i][k-1]) / (mas[i][max])

if (z > r):

myn=i

z=r

print("эта строка ведущая "+str(mas[myn][k-1]))

print("\n\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\n")

print("коэффицент "+str(mas[myn][max]))

l=mas[myn][max]

for i in range(n):

p=mas[i][max]

if (i != myn):

for j in range(k):

if j!=max:

mas[i][j]=(mas[i][j]\*mas[myn][max]-mas[myn][j] \* p)/mas[myn][max]

for j in range(k):

if j!=max:

mas[myn][j]=mas[myn][j] / l

for i in range(n):

if i!=myn:

mas[i][max]=-mas[i][max]/l

print("\n\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\n")

print(" Таблица: ")

for i in range(n):

for j in range(k):

print(" " + str(round(mas[i][j],1)), end=" ")

print("")

print("\n\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\n")

print("целевая функция равняется " + str(round(mas[n - 1][k - 1],1)) + "\n")

for j in range(sv):

§ (mas[n-1][j] > 0):

kpol+=1

count+=1

print("решение оптимально, целевая функция равняется "+str(round(mas[n - 1][k - 1],1))+"\n")