Санкт-Петербургский
Политехнический университет Петра Великого

Отчет по лабораторной работе №1
по дисциплине
"Методы оптимизации"

Решение задач линейного программирования. Реализация симплекс-метода для решения прямой и двойственной задачи линейного программирования.

Выполнили:

Марков Михаил Денисович
Аптуков Михаил Ильдусович
группа: 5030102/10201

Преподаватель:

Родионова Елена Александровна

Санкт-Петербург
2024

Содержание

1 Постановка задачи **2**

2 Условие применимости метода **3**
2.1 Метод перебора крайних точек . . . 3
2.2 Симплекс-метод . . . 3

3 Описание алгоритма **4**
3.1 Построение двойственной задачи . . . 4
3.2 Перевод в каноническую форму . . . 4
3.3 Метод перебора крайних точек . . . 5
3.4 Симплекс-метод . . . 6

4 Результаты **8**
4.1 Результат построения двойственной задачи к данной . . . 8
4.2 Результат перевода задачи в каноническую форму . . . 8
4.3 Результат перевода двойственной задачи в каноническую форму . . . 8
4.4 Результаты решения задач . . . 9
4.4.1 Прямая задача . . . 9
4.4.2 Двойственная задача . . . 9

5 Обоснование достоверности полученных результатов **10**

6 Дополнительные исследования **12**

1 Постановка задачи

В данной работе поставлена задача линейного программирования, которая содержит 5 неизвестных, а также систему из трех равенств и трех неравенств. Также в задаче три переменные имеют ограничения на знак. Задача выглядит следующим образом:

$$\begin{cases} 2x_1 + 1x_2 + 4x_3 + 2x_4 + 5x_5 \le 23 \\ -1x_1 + 1x_2 + 3x_3 + 12x_4 + 5x_5 \le 12 \\ 3x_1 + 9x_2 - 2x_3 + 15x_4 - 5x_5 \ge 10 \\ 4x_1 + 12x_2 + 4x_3 - 1x_4 + 4x_5 = 439 \\ 2x_1 - 2x_2 + 2x_3 + 1x_4 + 4x_5 = 11 \\ -1x_1 + 4x_2 + 2x_3 + 4x_4 + 4x_5 = 13 \end{cases} \tag{1}$$

$$x_2 \ge 0;\ x_4 \ge 0;\ x_5 \ge 0;\ x_i \in R,\ i = 1, 3 \tag{2}$$

Функция цели имеет вид:

$$F(x) = 3x_1 + 2x_2 + 4x_3 + 1x_4 + 5x_5 \tag{3}$$

1. Построить двойственную задачу
2. Автоматизировать приведение задачи в двойственную форму.
3. Привести прямую задачу к каноническому виду
4. Автоматизировать приведение задачи в каноническую форму.
5. Привести двойственную задачу к каноническому виду
6. Решить исходную задачу методом перебора крайних точек.
7. Решить двойственную задачу методом перебора крайних точек.
8. Решить прямую задачу симплекс-методом в таблицах.
9. Решить двойственную задачу симплекс-методом в таблицах.
10. Предусмотреть защиту от зацикливания. Показать работу алгоритма на отдельном примере.
11. Оформить интерфейс для взаимодействия с пользователем.
12. Привести сравнительный анализ соответствия симплекс-метода в таблицах и метода перебора крайних точек.

2 Условие применимости метода

2.1 Метод перебора крайних точек

Данный метод может быть применен к задачам с любым количеством переменных решения и ограничений. Если целевая функция линейна, а допустимая область выпукла, оптимальное решение всегда будет находиться в одной из крайних точек. Если целевая функция нелинейна или допустимая область не является выпуклой, оптимальное решение может находиться не в одной из крайних точек. В этом случае для поиска оптимального решения могут потребоваться дополнительные методы.

Заданная функция цели линейна, допустимая область выпукла, а значит, метод применим.

2.2 Симплекс-метод

Для применения симплекс-метода задача формулируется в канонической форме. Метод применим при любых вещественных значениях коэффициентов:

$$A \in R^{m \times n},\ b \in R^{m},\ c \in R^{n} \tag{4}$$

При этом ранг матрицы Aдолжен иметь значение не меньше m для существования базисного решения:

$$rang(A) \geq m \tag{5}$$

$$\begin{pmatrix} 1 & 2 & 5 & 2 & -2 & 4 & -4 & 1 & 0 & 0 \\ 1 & 12 & 5 & -1 & 1 & 3 & -3 & 0 & 1 & 0 \\ 9 & 15 & -5 & 3 & -3 & -2 & 2 & 2 & 0 & -1 \\ 12 & -1 & 4 & 4 & -4 & 4 & -4 & 0 & 0 & 0 \\ -2 & 1 & 4 & 2 & -2 & 2 & -2 & 0 & 0 & 0 \\ 4 & 4 & 4 & -1 & 1 & 2 & -2 & 0 & 0 & 0 \end{pmatrix} \to$$

$$\begin{pmatrix} 1 & 2 & 5 & 2 & -2 & 4 & -4 & 1 & 0 & 0 \\ 0 & 10 & 0 & -3 & 3 & -1 & 1 & -1 & 1 & 0 \\ 0 & 0 & -50 & -15.9 & 15.9 & -38.3 & 38.3 & -7.3 & 0.3 & -1 \\ 0 & 0 & -56 & -27.5 & 27.5 & -46.5 & 46.5 & -14.5 & 2.5 & 0 \\ 0 & 0 & 14 & 7.5 & -7.5 & 10.5 & -10.5 & 2.5 & -0.5 & 0 \\ 0 & 0 & -16 & -10.2 & 10.2 & -14.4 & 14.4 & -4.4 & 0.4 & 0 \end{pmatrix} \to$$

$$\begin{pmatrix} 1 & 2 & 5 & 2 & -2 & 4 & -4 & 1 & 0 & 0 \\ 0 & 10 & 0 & -3 & 3 & -1 & 1 & -1 & 1 & 0 \\ 0 & 0 & -50 & -15.9 & 15.9 & -38.3 & 38.3 & -7.3 & 0.3 & -1 \\ 0 & 0 & 0 & -9.692 & 9.692 & -3.604 & 3.604 & -6.324 & 2.164 & 1.12 \\ 0 & 0 & 0 & 3.048 & -3.048 & -0.224 & 0.224 & 0.456 & -0.416 & -0.28 \\ 0 & 0 & 0 & -5.112 & 5.112 & -2.144 & 2.144 & -2.064 & 0.304 & 0.32 \end{pmatrix} \to$$

$$\begin{pmatrix} 1 & 2 & 5 & 2 & -2 & 4 & -4 & 1 & 0 & 0 \\ 0 & 10 & 0 & -3 & 3 & -1 & 1 & -1 & 1 & 0 \\ 0 & 0 & -50 & -15.9 & 15.9 & -38.3 & 38.3 & -7.3 & 0.3 & -1 \\ 0 & 0 & 0 & -9.692 & 9.692 & -3.604 & 3.604 & -6.324 & 2.164 & 1.12 \\ 0 & 0 & 0 & 0 & 0 & -1.357 & 1.357 & -1.533 & 0.265 & 0.072 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1.546 & -0.885 & -0.284 \end{pmatrix}$$

После приведения матрицы к верхне-треугольному виду, можем сделать вывод, что ранг матрицы A в нашей задаче равен $r = 6\ (> m)$,а значит, симплекс-метод применим.

3 Описание алгоритма

3.1 Построение двойственной задачи

]A - матрица коэффициентов неравенства прямой задачи

]A' - матрица коэффициентов неравенства прямой задачи

]b - вектор свободных коэффициентов неравенства прямой задачи

]b' - вектор свободных коэффициентов неравенства двойственной задачи

]c - вектор коэффициентов целевой функции прямой задачи

]c' - вектор коэффициентов целевой функции двойственной задачи

Алгоритм:

1. Смена направления функции цели. Эквивалентность двух задач совпадении множества решений задач и выполнения равенства

$$\underset{x \in \Omega}{max} f(x) = -\underset{x \in \Omega}{min}\{-f(x)\} \tag{6}$$

2. Транспонируем матрицу A : $A^T = A'$

3. Смена вектора b (свободных коэффициентов) и вектора c (коэффициентов целевой функции) : $b' =$ c, $c' =$ b

4. Если $x_i \geq 0$, то i-ая строка новой системы будет иметь знать $\leq$. Если x_i не имеет ограничений на знак, то i-ая строка новой системы будет иметь знак $=$

5. Если в исходной системе ограничений i-ая строка ограничена знаком $\geq$, то новая переменная будет иметь ограничение на знак: $y_i \geq 0$.

6. Если в исходной системе ограничением i-ой строки служит знак $=$, то новая переменная не будет иметь ограничения на знак

3.2 Перевод в каноническую форму

Задачу линейного программирования всегда можно свести к канонической форме. Все ограничения имеют вид равенств и все переменные имеют ограничения на знак.

$$min(c^T[N] \cdot x[N]) \tag{7}$$

$$S = \{x[N] \mid A[M, N] \cdot x[N] = b[M],\ x[N] \geq 0\} \tag{8}$$

Будем считать, что количество строк матрицы A равно $m = |M|$, а количество столбцов $n = |N|$, причем $m \leq n$ и ранг матрицы A равен m.
Возьмем любой допустимый вектор $x_0[N] \in S$. Его компоненты или равны нулю или положительны. Введем множества

$$N_0^+ = \{i \in N | x_0[i] > 0\}, \quad N_0^0 = \{i \in N | x_0[i] = 0\} \tag{9}$$

Тогда ограничение типа равенства можно записать в виде

$$A[M, N_0^+] \cdot x[N_0^+] = b[M] \tag{10}$$

Так как $x_0[N_0^0] = 0$
Равенство (10) можно представить в виде

$$\sum_{j \in N_0^+} x_0[j] \cdot \overline{a_j}[M] = b[M] \tag{11}$$

Где $\overline{a_j}[M]$ - столбцы матрицы $A[M, N]$.

Алгоритм:

1. 1. Умножить на (-1) коэффициенты функции цели, если стоит задача минимизации и сделать задачу задачей максимизации.

2. Ввести новые переменны для всех переменных без ограничений знаков

   $$x_i = x_{i+n} - x_{i+n+1}$$

   Где n –количество переменных

3. Все неравенства преобразовать в путем умножения на (-1)

4. Сделать из неравенств равенства с помощью добавления новых переменных.

3.3 Метод перебора крайних точек

Крайняя точка - точка $x_0 \in S$ (выпуклому множеству) называется крайней точкой S, если не существует $x_1, x_2 \in S,\ x_1 \neq x_2$ таких, что $x_0 = (1 - \lambda)x_1 + \lambda x_2$ для некоторого $0 < \lambda < 1$.
Опорный вектор - вектор $x_0[N] \in S$ называется опорным, если векторы $\overline{a_j}[M]$, $j \in N_0^+$ линейно независимы.
Для применения алгоритма перебора крайних точек приводим задачу к канонической форме. Решение этой задачи связано с системой линейных уравнений (СЛАУ), возникающей после преобразования исходной задачи в каноническую форму. Основной трудностью здесь является поиск эффективного метода решения для неквадратных матриц. Метод реализации включает в себя два этапа. На первом этапе определяются крайние точки. Функция проходит через все возможные комбинации столбцов. Если на входе квадратная матрица, то решение системы линейных уравнений осуществляется непосредственным образом методом Гаусса. Отсекаются решения с отрицательными значениями, согласно ограничениям задачи в канонической форме.

1. Начинаем с получения входного объекта задачи, содержащего матрицу ограничений и вектор правой части.

2. Проверяем, является ли матрица ограничений квадратной. Если да, то система ограничений имеет единственное решение.

3. Если матрица не квадратная, генерируем все возможные комбинации столбцов размерности m, где m - количество строк матрицы ограничений.

4. Для каждой комбинации столбцов создаем соответствующую подматрицу и проверяем ее невырожденность (определитель не равен нулю). Если условие выполняется, решаем систему уравнений методом Гаусса.

5. Проверяем решение : если хотя бы одна компонента отрицательна, решение отбрасываем.

6. Допустимые решения добавляем в список решений.

7. Возвращаем список всех допустимых крайних точек, найденных при переборе комбинаций столбцов.

Вторая часть включает в себя оценку целевой функции. Данная функция перебирает все крайние точки, полученные из предыдущей части алгоритма. Затем происходит поиск точки, дающей наибольшее значение целевой функции.

1. В функции поиска оптимального решения из ранее вычисленных крайних точек находим максимальное значение целевой функции.

2. Начальное оптимальное решение устанавливаем как первая критическая точка, а начальное оптимальное значение - значение целевой функции в первой критической точке.

3. Вычисляем максимум знакчения целевой функции по всем критическим точкам.

4. После завершения перебора возвращаем найденную оптимальную точку и ее значение целевой функции.

3.4 Симплекс-метод

Алгоритм симплекс-метода в таблицах по большей части строится за счет таблиц. Для более удобной формулировки переформулируем задачу в канонической форме еще раз:

$$F = c^T x \to \max \tag{12}$$

$$Ax \geq b \tag{13}$$

$$x \geq 0 \tag{14}$$

1. Инициализация. Создание начальной симплекс-таблицы из матрицы коэффициентов A и вектора b, а также добавления коэффициентов c в качестве последний строки матрицы. Очевидно, что в этом случае необходимо также добавить к вектору свободных членов нулевой элемент для того, чтобы выровнять размерности. В этом случае будем такую матрицу называть таблицей T, которая определяется следующим образом:
$$T = \begin{pmatrix} T & b \\ c^T & 0 \end{pmatrix} \tag{15}$$

2. Проверка оптимальности. Если все элементы строки с целевой функций с неотрицательны, то текущее решение оптимально и алгоритм завершает работу:
$$\forall i : c_i \geq 0 \tag{16}$$

3. Выбор ведущего столбца: Выбирается как столбец с максимальным значением коэффициента целевой функции.
$$j = argmax(\mathbf{c}) \tag{17}$$

4. Выбор ведущей строки
   Выбирается как строка с самым минимальным положительным отношением элементов столбца свободных членов к соответствующим элементам выбранного столбца j:
$$i = argmin\left(\frac{T[:,-1]}{T[:,j]}\right) \tag{18}$$
   Среди всех строк выбираем ту, для которой достигается минимум отношения правой части и коэффициента вводимого столбца в таблице при условии, что этот коэффициент больше нуля. Если такой минимум достигается на нескольких строках, выбираем строку, соответствующую столбцу с наименьшим индексом. Этот порядок выбора называется правилом Блэнда, позволяющем избегать зацикливания при наличии нескольких возможных оптимальных путей.

5. Перерасчет таблицы

$$T[i,:] = \frac{T[i,:]}{T[i,j]}$$

$$T[k,i] = T[k,i] - T[k,j]T[i,:],\ \forall k \neq i$$

6. Повторение шагов с 1 по 5, пока не будет достигнут критерий остановки

Дополнительно к правилу Блэнда, направленному на предотвращение зацикливания и повторного посещения состояний, текущее состояние таблицы подвергается хешированию и затем включается в список посещённых состояний.

4 Результаты

4.1 Результат построения двойственной задачи к данной

$$\begin{cases} 2x_1 - 1x_2 - 3x_3 + 4x_4 + 2x_5 + -1x_6 = 3 \\ 1x_1 + 1x_2 - 9x_3 + 12x_4 - 2x_5 + 4x_6 \geq 2 \\ 4x_1 + 3x_2 + 2x_3 + 4x_4 + 2x_5 + 2x_6 = 4 \\ 2x_1 + 12x_2 - 15x_3 - 1x_4 + 1x_5 + 4x_6 = \geq 1 \\ 5x_1 + 5x_2 + 5x_3 + 4x_4 + 4x_5 + 4x_6 = \geq 5 \end{cases}$$

$$x_i \geq 0, \ i = \overline{1,3}$$

$$F(x) = 23x_1 + 12x_2 - 10x_3 + 439x_4 + 11x_5 + 13x_6 \to min$$

4.2 Результат перевода задачи в каноническую форму

$$\begin{cases} 1x_1 + 2x_2 + 5x_3 + 2x_4 - 2x_5 + 4x_6 - 4x_7 + 1x_8 = 23 \\ 1x_1 + 12x_2 + 5x_3 - 1x_4 + 1x_5 + 3x_6 - 3x_7 + 1x_9 = 12 \\ 9x_1 + 15x_2 - 5x_3 + 3x_4 - 3x_5 - 2x_6 + 2x_7 - 1x_{10} = 10 \\ 12x_1 - 1x_2 + 4x_3 + 4x_4 - 4x_5 + 4x_6 - 4x_7 = 439 \\ -2x_1 + 1x_2 + 4x_3 + 2x_4 - 2x_5 + 2x_6 - 2x_7 = 11 \\ 4x_1 + 4x_2 + 4x_3 - 1x_4 + 1x_5 + 2x_6 - 2x_7 = 13 \end{cases}$$

$$x_i \geq 0, \ i = \overline{1,10}$$

Функция цели:

$$F(x) = 2x_1 + 1x_2 + 5x_3 + 3x_4 - 3x_5 - 4x_6 - 4x_7 \to max$$

4.3 Результат перевода двойственной задачи в каноническую форму

$$\begin{cases} 2x_1 - 1x_2 - 3x_3 + 4x_4 - 4x_5 + 2x_6 - 2x_7 - 1x_8 + 1x_9 = 3 \\ 1x_1 + 1x_2 - 9x_3 + 12x_4 - 12x_5 - 2x_6 + 2x_7 + 4x_8 - 4x_9 - 1x_{11} = 2 \\ 4x_1 + 3x_2 + 2x_3 + 4x_4 - 4x_5 + 2x_6 - 2x_7 - 4x_8 + 4x_9 = 4 \\ +2x_1 + 12x_2 - 15x_3 - 1x_4 + 1x_5 + 1x_6 - 1x_7 + 4x_8 - 4x_9 - 1x_{11} = 1 \\ 5x_1 + 5x_2 + 5x_3 + 4x_4 - 4x_5 + 4x_6 - 4x_7 + 4x_8 - 4x_9 - 1x_{12} = 5 \end{cases}$$

$$x_i \geq 0, \ i = \overline{1,12}$$

Функция цели:

$$F(x) = -23x_1 - 12x_2 + 10x_3 - 439x_4 + 439x_5 - 11x_6 + 11x_7 - 13x_8 + 13x_9 \to max$$

4.4 Результаты решения задач

4.4.1 Прямая задача

Методом перебора крайних точек было получено следующее решение задачи:

1. В базисе канонической формы:
   $x = (28.7931\ \ 14.5632\ \ 0\ \ 71.4828\ \ 0\ \ 0\ \ 44.4713\ \ 0\ \ 13.3448\ \ 770.9770)$
   $F(x) = 108.7126$

2. В базисе исходной задачи:
   $x = (71.4828\ \ 28.7931\ \ -44.4713\ \ 14.5632\ \ 0)$
   $F(x) = 108.7126$

Симплекс-методом было получено следующее решение задачи:

1. В базисе канонической формы:
   $x = (28.7931\ \ 14.5632\ \ 0\ \ 71.4828\ \ 0\ \ 0\ \ 44.4713\ \ 0\ \ 13.3448\ \ 770.9770)$
   $F(x) = 108.7126$

2. В базисе исходной задачи:
   $x = (71.4828\ \ 28.7931\ \ -44.4713\ \ 14.5632\ \ 0)$
   $F(x) = 108.7126$

4.4.2 Двойственная задача

Методом перебора крайних точек было получено следующее решение задачи:

1. В базисе канонической формы:
   $y = (0.6897\ \ 0\ \ 0\ \ 0.2069\ \ 0\ \ 0.3333\ \ 0\ \ 0\ \ 0.1264\ \ 0\ \ 0\ \ 0.1034)$
   $F(y) = -108.7126$

2. В базисе исходной задачи:
   $y = (0.6897\ \ 0\ \ 0\ \ 0.2069\ \ 0.3333\ \ -0.1264)$
   $F(y) = 108.7126$

Симплекс-методом было получено следующее решение задачи:

1. В базисе канонической формы:
   $y = (0.6897\ \ 0\ \ 0\ \ 0.2069\ \ 0\ \ 0.3333\ \ 0\ \ 0\ \ 0.1264\ \ 0\ \ 0\ \ 0.1034)$
   $F(y) = -108.7126$

2. В базисе исходной задачи:
   $y = (0.6897\ \ 0\ \ 0\ \ 0.2069\ \ 0.3333\ \ -0.1264)$
   $F(y) = 108.7126$

5 Обоснование достоверности полученных результатов

Симплекс-метод и метод перебора опорных точек представляют собой методы решения задач линейного программирования.

Симплекс-метод, как итеративный метод, обладает преимуществом постепенного приближения к точке оптимума. На каждой итерации он эффективно изменяет базис, стремясь к оптимальному решению задачи.

В отличие от него, метод перебора опорных точек является неитерационным. Его суть заключается в исчерпывающем переборе всех возможных комбинаций опорных векторов для выбора оптимального решения.

Хотя симплекс-метод обычно более эффективен, особенно в задачах с большой размерностью, благодаря постоянному движению в сторону оптимума, метод перебора опорных точек предоставляет более стабильный и надежный подход. Симплекс-метод, как итеративный метод, подвержен зацикливанию и численным ошибкам. В лабораторной работе не удалось найти решение двойственной задачи с помощью симплекс-метода. Это связано с тем, что не удалось найти положительное отношение свободных членов к выбранному столбцу с максимальным по модулю значением. коэффициента функции цели в симплекс-таблице.

Что касается результатов, то для прямой задачи полученные значения вектора решения и целевой функции совпадают для симплекс-метода и метода перебора крайних точек. Это свидетельствует о достоверности результатов. Также проведем дополнительные вычисления по проверке условий оптимальности для найденного решения.
Известно, что для того, чтобы $x_*[N]$ была оптимальной точкой в задаче линейного программирования, необходимо и достаточно, чтобы $\exists y_*[M]$ такой, что

$$c^T[N] - y_*^T[M] \cdot A[M,N] \geq 0$$

$$(c^T[N] - y_*^T[M] \cdot A[M,N]) \cdot x_*[N] = 0$$

Имеем вектор:

$$x^* = (28.7931 \;\; 14.5632 \;\; 0 \;\; 71.4828 \;\; 0 \;\; 0 \;\; 44.4713 \;\; 0 \;\; 13.3448 \;\; 770.9770)$$

$$A = \begin{pmatrix} 1 & 2 & 5 & 2 & -2 & 4 & -4 & 1 & 0 & 0 \\ 1 & 12 & 5 & -1 & 1 & 3 & -3 & 0 & 1 & 0 \\ 9 & 15 & -5 & 3 & -3 & -2 & 2 & 0 & 0 & -1 \\ 12 & -1 & 4 & 4 & -4 & 4 & -4 & 0 & 0 & 0 \\ -2 & 1 & 4 & 2 & -2 & 2 & -2 & 0 & 0 & 0 \\ 4 & 4 & 4 & -1 & 1 & 2 & -2 & 0 & 0 & 0 \end{pmatrix}$$

$$c = (2;\; 1;\; 5;\; 3;\; -3;\; 4;\; -4;\; 0;\; 0;\; 0)$$

$]y = (y1 \;\; y2 \;\; y3 \;\; y4 \;\; y5 \;\; y6)$

$$y_*^T[M] \cdot A[M,N] = \begin{pmatrix} y_1 + y_2 + 9y_3 + 12y_4 - 2y_5 + 4y_6 \\ 2y_1 + 12y_2 + 15y_3 - y_4 + y_5 + 4y_6 \\ 5y_1 + 5y_2 - 5y_3 + 4y_4 + 4y_5 + 4y_6 \\ 2y_1 - y_2 + 3y_3 + 4y_4 + 2y_5 - y_6 \\ -2y_1 + y_2 - 3y_3 - 4y_4 - 2y_5 + y_6 \\ 4y_1 + 3y_2 - 2y_3 + 4y_4 + 2y_5 + 2y_6 \\ -4y_1 - 3y_2 + 2y_3 - 4y_4 - 2y_5 - 2y_6 \\ y_1 \\ y_2 \\ -y_3 \end{pmatrix}$$

$$c^T[N] - y_*^T[M] \cdot A[M,N] = \begin{pmatrix} 2 - y_1 - y_2 - 9y_3 - 12y_4 + 2y_5 - 4y_6 \\ 1 - 2y_1 - 12y_2 - 15y_3 + y_4 - y_5 - 4y_6 \\ 5 - 5y_1 - 5y_2 + 5y_3 - 4y_4 - 4y_5 - 4y_6 \\ 3 - 2y_1 + y_2 - 3y_3 - 4y_4 - 2y_5 + y_6 \\ -3 + 2y_1 - y_2 + 3y_3 + 4y_4 + 2y_5 - y_6 \\ 4 - 4y_1 - 3y_2 + 2y_3 - 4y_4 - 2y_5 - 2y_6 \\ -4 + 4y_1 + 3y_2 - 2y_3 + 4y_4 + 2y_5 + 2y_6 \\ -y_1 \\ -y_2 \\ y_3 \end{pmatrix}$$

$$(c^T[N] - y_*^T[M] \cdot A[M,N]) \cdot x^*[N] = \begin{pmatrix} 2 - y_1 - y_2 - 9y_3 - 12y_4 + 2y_5 - 4y_6 \\ 1 - 2y_1 - 12y_2 - 15y_3 + y_4 - y_5 - 4y_6 \\ 5 - 5y_1 - 5y_2 + 5y_3 - 4y_4 - 4y_5 - 4y_6 \\ 3 - 2y_1 + y_2 - 3y_3 - 4y_4 - 2y_5 + y_6 \\ -3 + 2y_1 - y_2 + 3y_3 + 4y_4 + 2y_5 - y_6 \\ 4 - 4y_1 - 3y_2 + 2y_3 - 4y_4 - 2y_5 - 2y_6 \\ -4 + 4y_1 + 3y_2 - 2y_3 + 4y_4 + 2y_5 + 2y_6 \\ -y_1 \\ -y_2 \\ y_3 \end{pmatrix} \cdot \begin{pmatrix} 28.7931 \\ 14.5632 \\ 0 \\ 71.4828 \\ 0 \\ 0 \\ 44.4713 \\ 0 \\ 13.3448 \\ 770.9770 \end{pmatrix} =$$

$= 28.7931 \cdot (2 - y_1 - y_2 - 9y_3 - 12y_4 + 2y_5 - 4y_6) + 14.5632 \cdot (1 - 2y_1 - 12y_2 - 15y_3 + y_4 - y_5 - 4y_6) + 71.4828 \cdot (3 - 2y_1 + y_2 - 3y_3 - 4y_4 - 2y_5 + y_6) + 44.4713 \cdot (-4 + 4y_1 + 3y_2 - 2y_3 + 4y_4 + 2y_5 + 2y_6) - 13.3448 \cdot y_2 + 770.9770 \cdot y_3$

Получили СЛАУ:

$$\begin{cases} -y_1 - y_2 - 9y_3 - 12y_4 + 2y_5 - 4y_6 = -2 \\ -2y_1 - 12y_2 - 15y_3 + y_4 - y_5 - 4y_6 = -1 \\ -2y_1 + y_2 - 3y_3 - 4y_4 - 2y_5 + y_6 = -3 \\ 4y_1 + 3y_2 - 2y_3 + 4y_4 + 2y_5 + 2y_6 = 4 \\ y_2 = 0 \\ y_3 = 0 \end{cases}$$

Решение СЛАУ: $y = (\frac{20}{29} \;\; 0 \;\; 0 \;\; \frac{6}{29} \;\; \frac{1}{3} \;\; -\frac{11}{87})$
В десятичной записи: $y = (0.6897 \;\; 0 \;\; 0 \;\; 0.2069 \;\; 0.3333 \;\; -0.1264)$
Решение СЛАУ совпадает с решением двойственной задачи.

6 Дополнительные исследования

В дополнение к основному анализу, было проведено сравнение табличного симплекс-метода с примером, представленным на лекции. В последнем случае следующее приближение к решению получалось в соответствии с правилом:

$$x_{k+1}[N] = x_k[N] - \theta u_k[N]$$

где

$$\theta = \min_{i \in N_k,\ u_k[i] > 0} \frac{x_k[i]}{u_k[i]}$$

$$u_k[N_k] = B[N_k, M]A[M, j_k]$$

где j_k - индекс такого столбца, что $d_k[j_k] < 0$, и в этом случае $u_k[j_k] = -1$.

Сравнив два метода, можно выделить следующие аспекты:

1. Оба метода требуют приведения задачи к каноническому виду.

2. Условия остановки для обоих методов эквивалентны: для табличного симплекс-метода - положительность всех компонент вектора целевой функции $c^T[L_k] > 0$, для лекционного симплекс-метода - $c^T[L_k] > 0$, где $y^T[M] = c^T[N_k] \cdot B[N_k, M] = 0$ и $c^T[L_k] - y^T[M] \cdot A[M, L_k] > 0$.

3. Замена базиса в обоих методах осуществляется схожим образом.

Хотя табличный симплекс-метод легок в реализации, на практике его применяют редко из-за низкой вычислительной эффективности. На каждой итерации требуется пересчитывать всю симплекс-таблицу, хотя для выбора базисной переменной нужен всего один столбец этой матрицы.

В отношении исследованных методов можно сказать, что метод перебора является простым и понятным, однако составление и решение C_n^m матриц может существенно замедлить работу в случае задач большой размерности. Симплекс-метод, в свою очередь, представляет собой универсальный метод, способный решить любую задачу линейного программирования. Решение гарантировано будет найдено за $O(2^n)$ операций, где n – количество переменных.