МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ

Нижегородский государственный университет им. Н.И. Лобачевского
Национальный исследовательский университет

АЛГОРИТМ ФУРЬЕ-МОЦКИНА ДЛЯ ПОСТРОЕНИЯ ДВОЙСТВЕННОГО ОПИСАНИЯ ПОЛИЭДРА

Учебно-методическое пособие

Рекомендовано методической комиссией факультета ВМК
для студентов ННГУ, обучающихся по направлениям подготовки
02. 03. 02 “Фундаментальная информатика и информационные технологии”,
01. 03. 02 “Прикладная математика и информатика”,
01. 03. 01 “Математика”,
02. 03. 01 “Математика и компьютерные науки”.

Нижний Новгород
2015

УДК 512.14+512.622(077)
ББК В142(Я73)

АЛГОРИТМ ФУРЬЕ-МОЦКИНА ДЛЯ ПОСТРОЕНИЯ ДВОЙСТВЕННОГО ОПИСАНИЯ ПОЛИЭДРА. Авторы: Грибанов Д.В., Шевчук Е.А. Учебно-методическое пособие. – Нижний Новгород: Нижегородский госуниверситет, 2015.

Рецензент: к.т.н., доцент Васин Д.Ю.

В данном учебно-методическом пособии приведен алгоритм Фурье-Моцкина для построения двойственного описания полиэдра. Алгоритм позволяет переходить от конечно определенного описания полиэдра к конечно порожденному и наоборот. Приведена часть теории полиэдров необходимая для понимания алгоритма. Все необходимые утверждения, леммы и теоремы приведены с доказательствами. Доказана корректность алгоритма. Также разобрано несколько примеров и приведены задачи связанные с задачей двойственного описания полиэдра.

Учебно-методическое пособие предназначено для студентов третьего и четвертого курсов, обучающихся по направлениям: “Фундаментальная информатика и информационные технологии”, “Прикладная математика и информатика”, “Математика”, “Математика и компьютерные науки”, а также может быть использовано школьниками старших классов, занимающихся научной работой в рамках НОУ.

Ответственный за выпуск:
заместитель председателя методической комиссии факультета ВМК ННГУ,
к.т.н., доцент **В.М. Сморкалова**

УДК 512.14+512.622(077)
ББК В142(Я73)

Условные обозначения

$\mathbb{N}$ - множество натуральных чисел.

$\mathbb{Z}$ - множество целых чисел.

$\mathbb{Q}$ - множество рациональных чисел.

$\mathbb{R}$ - множество действительных чисел.

$\mathbb{F}_+$ - множество неотрицательных чисел из числового множества $\mathbb{F}$ (в качестве $\mathbb{F}$ может стоять любое числовое множество, на котором определена отношение сравнения чисел, например $\mathbb{R}$).

$\mathbb{F}^{m\times n}$ - множество m на n матриц с элементами из числового множества F (в качестве в качестве $\mathbb{F}$ может стоять любое числовое множество, например $\mathbb{Z}$ или $\mathbb{R}_+$).

$\mathbb{F}^n$ - множество $F^{n\times 1}$ матриц, или попросту множество вектор столбцов с элементами из числового множества $\mathbb{F}$.

Пусть A есть некоторая матрица, тогда $A_{i\,j}$ обозначает элемент матрицы A находящийся в i-ой строке и j-ом столбце.

Пусть A есть некоторая матрица, тогда $A_{i\,*}$ обозначает вектор строку, которая является i-ой строкой матрицы A.

Пусть A есть некоторая матрица, тогда $A_{*\,i}$ обозначает вектор столбец, который является i-ым столбцом матрицы A.

Пусть A есть некоторая матрица, тогда за $lin.hull(A)$ обозначим множество всех линейных комбинаций столбцов матрицы A с коэффициентами из $\mathbb{R}$. То есть $lin.hull(A) = \{At : t \in \mathbb{R}^n\}$.

Пусть A есть некоторая матрица, тогда за $cone.hull(A)$ обозначим множество всех линейных комбинаций столбцов матрицы A с коэффициентами из $\mathbb{R}_+$. То есть $cone.hull(A) = \{At : t \in \mathbb{R}^n_+\}$.

Некоторые факты о конусах

Определение 1 *Множество $M \subseteq \mathbb{R}^n$ называется выпуклым множеством, если для любых точек $u, v \in M$ выполнено, что отрезок соединяющий точки u, v включен в M. То есть $\{tu + (1-t)v : 0 \le t \le 1, t \in \mathbb{R}\} \subseteq M$.*

Определение 2 *Множество $M \subseteq \mathbb{R}^n$ есть конус, если для любых $x, y \in M$ и $\alpha, \beta \in \mathbb{R}_+$ верно, что $\alpha x + \beta y \in M$.*

Очевидно конус является выпуклым множеством. Также очевидно пересечение выпуклых множеств является выпуклым множеством, а пересечение конусов является конусом. Попробуйте это доказать.

Далее нас будут интересовать только конуса определенные конечными системами однородных неравенств или заданные конечными системами векторов.

Определение 3 *Пусть $A \in \mathbb{R}^{m\times n}$, тогда множество $C(A) = \{x \in \mathbb{R}^n : Ax \le 0\}$ будем называть конусом определенным строчками матрицы A. Очевидно данное множество является конусом в смысле предыдущего определения.*

Определение 4 *Пусть $M \subset \mathbb{R}^n$ есть конус. Тогда линейной частью конуса M будем называть множество $lineal(M) = \{x \in \mathbb{R}^n : tx \in M$ для всех $t \in \mathbb{R}\}$.*

Определение 5 *Также $lineal(M)$ можно определить как максимальное по включению подмножество M, являющееся линейным пространством.*

Утверждение 1 *Данные определения эквивалентны.*

Доказательство. В первую очередь покажем, что второе определение корректно. То есть оно определяет множество $lineal(M)$ единственным образом. Действительно пусть существуют два несовпадающих множества V_1 и V_2, являющихся максимальными по включению подмножествами M и одновременно линейными пространствами. Пусть столбцы матрицы B формируют базис пространства V_1, так как $V_1 \neq V_2$, то базис B пространства V_1 можно дополнить вектором $b \in V_2 \setminus V_1$. Столбцы матрицы $B' = (B\, b)$ будут формировать новый базис. Осталось показать, что $lin.hull(B') \subseteq M$. Действительно, так как любой вектор из $V_1 = lin.hull(B)$ принадлежит M, а также b принадлежит M, то по определению конуса любая их линейная комбинация принадлежит M. Полученное противоречие доказывает корректность второго определения.

Теперь непосредственно докажем корректность первого и второго определений. Очевидно, что по первому определению $lineal(M)$ является линейным пространством и подмножеством M. Максимальность следует из того, что в противном случае в $lineal(M)$ можно было бы добавить новый вектор.

Обратно, по второму определению имеем, что если $x \in lineal(M)$, то $tx \in lineal(M)$ для $t \in \mathbb{R}$. Так как $lineal(M) \subseteq M$, то $tx \in M$. Таким образом $lineal(M) \subseteq \{x \in \mathbb{R}^n : \ tx \in M \text{ для всех } t \in \mathbb{R}\} \subseteq M$. Далее утверждение следует из максимальности $lineal(M)$. ■

Утверждение 2 *Пусть $C = C(A)$ для некоторой матрицы $A \in \mathbb{R}^{m\times n}$. Тогда $lineal(C) = \{x \in \mathbb{R}^n : \ Ax = 0\}$.*

Доказательство. Покажем, что $lineal(C) \subseteq \{x \in \mathbb{R}^n : \ Ax = 0\}$. Действительно, пусть $x \in lineal(C)$, тогда и $-x \in lineal(C)$. В силу того, что $lineal(C) \subseteq C$ имеем $Ax \leq 0$ и $A(-x) \leq 0$, откуда $Ax = 0$.

Обратное включение следует из того, что $\{x \in \mathbb{R}^n : \ Ax = 0\}$ является линейным пространством включенным в C. ■

Определение 6 *Конус C называется острым если $dim(lineal(C)) = 0$.*

Утверждение 3 *Пусть $C = C(A)$ для некоторой матрицы A. C является острым тогда и только тогда, когда существует вектор $c \in \mathbb{R}^n$, такой что для любой точки $x \in C \setminus \{0\}$ верно $c^\top x > 0$. Вектор c может быть эффективно найден.*

Доказательство. Докажем достаточность. Предположим от противного, что $dim(lineal(C)) > 0$. Тогда существует вектор $x \in lineal(C) \setminus \{0\}$. Рассмотрим точки $tx \in lineal(C) \subseteq C$ для $t \in \mathbb{R}$. Из условий следует, что $c^\top tx = tc^\top x \geq 0$ для всех $t \in \mathbb{R}$, что возможно лишь в случае $c^\top x = 0$. Так как $x \neq 0$, достигнуто противоречие.

Докажем необходимость. Рассмотрим вектор $c^\top = -(1,\dots,1)A$. Тогда для любого $x \in C \setminus \{0\}$ верно, что $c^\top x = -(1,\dots,1)Ax > 0$. Последнее неравенство следует из того, что $lineal(C) = \{0\}$ откуда $Ax \neq 0$, следовательно не все неравенства системы $Ax \leq 0$ обращаются в равенства на векторе x. Вектор c и является искомым вектором. Доказательство даёт эффективный способ поиска такого вектора. ■

Определение 7 *Пусть даны множества $M \subseteq \mathbb{R}^n$ и $N \subseteq \mathbb{R}^n$. Суммой Минковского множеств M и N называется множество $M + N = \{x \in \mathbb{R}^n : \ x = y + z,\, y \in M,\, z \in N\}$.*

Очевидно, что если два множества были конусами, то их сумма Минковского тоже является конусом.

Следующее утверждение по существу позволяет вести дальнейшую работу только с острыми конусами, так как оно показывает что любой конус можно представить в виде суммы Минковского его линейной части и какого то конечно определенного острого конуса.

Утверждение 4 *Пусть $C = C(A)$ для некоторой матрицы $A \in \mathbb{R}^{m\times n}$ ранга r. Тогда верно тождество $C = lineal(C) + C \cap U$, где U есть дополнительное подпространство к $lineal(C)$ (то есть $lineal(C) \oplus U = \mathbb{R}^n$). Конус $C \cap U$ является острым. Более того, существует $(m + 2(n-r)) \times n$ матрица B, что $C \cap U = C(B)$.*

Доказательство. Пусть $x \in C$, тогда $x = y + z$, где y есть проекция x на $lineal(C)$ параллельно U, а $z \in U$ есть ортогональная составляющая. Так как $y \in lineal(C)$, то $-y \in lineal(C) \subseteq C$. Из того, что $z = x + (-y)$ и того, что C является конусом следует, что $z \in C \cap U$. Таким образом $x \in lineal(C) + C \cap U$.

Обратное включение является очевидным.

Покажем, что $C \cap U$ есть острый конус. Если $lineal(C \cap U) \neq \{0\}$, то базис $lineal(C)$ может быть дополнен вектором из $lineal(C \cap U)$, так как $lineal(C) \cap U = \emptyset$. Это противоречит максимальности $lineal(C)$.

В дальнейшем, неоднозначность в выборе U приводит к неоднозначности в выборе матрицы B. Например в качестве U можно выбрать ортогональное дополнение к $lineal(C(A))$ считая скалярное произведение стандартным ($(x, y) = x^\top y$). Давайте так и поступим. Пусть столбцы матрицы $L \in \mathbb{R}^{n\times(n-r)}$ образуют базис $lineal(C)$. Тогда $U = \{x \in \mathbb{R}^n : \; L^\top x = 0\}$. Осталось лишь выписать систему неравенств для $C(A) \cap U$. Подходящей системой будет: $\begin{cases} Ax \leq 0 \\ L^\top x \leq 0 \\ -L^\top x \leq 0 \end{cases}$. Таким образом положив $B = \begin{pmatrix} A \\ L^\top \\ -L^\top \end{pmatrix}$ имеем $C \cap U = C(B)$. ■

Замечание 1 *На самом деле множество $C \cap U = C(B)$ из предыдущего утверждения можно представить используя меньшее количество переменных, то есть $C \cap U$ имеет размерность меньшую чем n. Об этом сказано в следующем утверждении.*

Утверждение 5 *Конус $C \cap U$ из предыдущего утверждения можно записать как $C \cap U = \{x = Ft : \; Gt \leq 0\}$, где столбцы $F \in \mathbb{R}^{n\times r}$ образуют базис U и $G \in \mathbb{R}^{m\times r}$. Таким образом множество $C \cap U$ имеет размерность r.*

Доказательство. В качестве базиса U можно выбрать линейно независимые вектор строки матрицы A, их будет в точности r. Запишем их столбцами в матрицу F. Тогда в качестве матрицы G подойдет $m \times r$ матрица AF. ■

Определение 8 *Пусть дан конус $C = C(A)$ для некоторой матрицы A. Размерностью конуса C будем называть размерность линейной оболочки его векторов, то есть $dim(C) = dim(lin.hull(C))$.*

Замечание 2 *Легко понять, что для конуса $C = C(A)$ множество $lin.hull(C)$ является единственным минимальным по включению линейным подпространством содержащим C. Таким образом предыдущее определение является осмысленным.*

Утверждение 6 *Пусть дан конус $C = C(A)$ для некоторой матрицы A. Тогда $lin.hull(C) = \{x : A^= x = 0\}$, где матрица $A^=$ составлена из тех строк A_{i*} матрицы A, что $A_{i*}x = 0$ для любых $x \in C(A)$. По этой причине строки матрицы $A^=$ называют неявными равенствами.*

Определение 9 *Пусть $A \in \mathbb{R}^{n\times m}$. Множество $cone.hull(A) = \{x \in \mathbb{R}^n : \; x = At, \, t \in \mathbb{R}^m_+\}$ будем называть конусом порожденным столбцами матрицы A.*

Двойственное описание для конусов и алгоритм Фурье-Моцкина

Нашей дальнейшей целью будет поиск другого способа описания для конуса $C(A)$.

Определение 10 *Дана матрица $A \in \mathbb{R}^{n\times n}$, причем $det(A) \neq 0$. Острый конус $C(A)$ называется элементарным.*

Лемма 1 *Для невырожденной матрицы $A \in \mathbb{R}^{n\times n}$ верно тождество $C(A) = cone.hull(-A^{-1})$.*

Доказательство. Пусть $x \in C(A)$, тогда $Ax \leq 0$, откуда $Ax = -y$ для некоторого $y \in \mathbb{R}^n_+$. Таким образом имеем $x = -A^{-1}y$, что эквивалентно тому, что $x \in cone.hull(-A^{-1})$.

Наоборот пусть $x \in cone.hull(-A^{-1})$, тогда $x = -A^{-1}y$ для некоторого $y \in \mathbb{R}^n_+$. Имеем $Ax = -y \leq 0$. ∎

Предыдущая лемма утверждает, что для элементарных существуют два эквивалентных способа описания. Наша цель доказать, что это верно для всех конусов вида $C(A)$ и для всех конусов $cone.hull(B)$. Эти два способа описания конусов называются двойственным описанием. Следующая лемма является важной частью классической теоремы Минковского-Фаркаша-Вейля.

Замечание 3 *Заметим, что ели для некоторого конуса C, верно $C = lineal(C) + cone.hull(B)$, то $C = cone.hull(B')$, для некоторой матрицы B' с большим числом столбцов. Действительно, пусть столбцы матрицы A образуют базис $lineal(C)$, тогда очевидно $lineal(C) = cone.hull(A\,b)$, где b есть сумма столбцов матрицы A взятая с противоположным знаком, то есть $b = -A(1,\ldots,1)^\top$. Таким образом лемма 4 позволяет вести поиск двойственного описания только для острых конусов.*

Лемма 2 *Пусть дан острый конус $C = C(A)$ для некоторой матрицы $A \in \mathbb{R}^{m\times n}$. Тогда существует матрица $B \in \mathbb{R}^{n\times s}$, такая что $C = cone.hull(B)$.*

Доказательство. Опишем алгоритм позволяющий найти матрицу B. Предположим что матрица составленная из n первых строк матрицы A является невырожденной, обозначим данную матрицу $A^{(0)}$. Такое предположение можно сделать без ограничения общности используя утверждение 2, так как конус является острым. Пусть $B^{(0)} = {A^{(0)}}^{-1}$. Мы знаем, что $C(A^{(0)}) = cone.hull(B^{(0)})$. Алгоритм будет построен следующим образом: по индукции будем считать, что для шага с номером k матрица $B^{(k)}$ уже построена, то есть $C(A^{(k)}) = cone.hull(B^{(k)})$. Добавим к $A^{(k)}$ следующую строку матрицы A и получим матрицу $A^{(k+1)}$. Дальше, способом описанным ниже найдем матрицу $B^{(k+1)}$, такую что $C(A^{(k+1)}) = cone.hull(B^{(k+1)})$. Продолжая это процедуру нужное количество раз все строки матрицы A будут использованы и будет получена финальная матрица $B = B^{(l)}$, такая что $C(A) = cone.hull(B)$, где за l обозначим количество шагов. Очевидно $l = m - n$, так как это в точности количество оставшихся строк матрицы A после удаления первых n независимых строк на нулевом шаге. Алгоритм действительно является on-line алгоритмом, так как неравенства могут поступать друг за другом, независимо.

Пусть c есть строка матрицы A добавленная на $k+1$ шаге. Тогда рассмотрим следующие множества: $I_0 = \{i : cB^{(k)}_{*i} = 0\}$, $I_- = \{i : cB^{(k)}_{*i} < 0\}$ и $I_+ = \{i : cB^{(k)}_{*i} > 0\}$.

Тогда в матрицу $B^{(k+1)}$ войдут следующие столбцы:

1) $B^{(k)}_{*i}$ для $i \in I_0 \cup I_-$.

2) $\alpha B^{(k)}_{*i} + \beta B^{(k)}_{*j}$ для $i \in I_-$ и $j \in I_+$. Строго положительные числа α, β выбираются так, чтобы $\alpha c B^{(k)}_{*i} + \beta c B^{(k)}_{*j} = 0$. Например можно взять $\alpha = cB^{(k)}_{*j}$ и $\beta = -cB^{(k)}_{*i}$.

Покажем, что $C(A^{(k+1)}) = cone.hull(B^{(k+1)})$. Так как все столбцы $B^{(k+1)}$ являются положительными комбинациями столбцов $B^{(k)}$, то в силу определения конуса имеем $cone.hull(B^{(k+1)}) \subseteq C(A^{(k)}) \subseteq C(A^{(k+1)})$.

Докажем обратное включение. Пусть $x \in C(A^{(k+1)})$. Точки $C(A^{(k+1)})$ есть в точности точки $x \in C(A^{(k)}) = cone.hull(B^{(k)})$ дополнительно удовлетворяющие неравенству $cx \le 0$. Пусть также $\delta = cx$. Так как $x \in cone.hull(B^{(k)})$, то верно, что $x = \sum_{i \in I_0} t_i B^{(k)}_{*i} + \sum_{i \in I_-} t_i B^{(k)}_{*i} + \sum_{j \in I_+} t_j B^{(k)}_{*j}$.

Возможно два случая:

1) $I_+ = \emptyset$. В данном случае очевидно $C(A^{(k+1)}) = cone.hull(B^{(k)})$ так как неравенство $cx \le 0$ ничего не отсекает.

2) $I_+ \ne \emptyset$.

Далее будем рассматривать только случай 2). Тогда $\delta = cx = \sum_{i \in I_+} t_i cB^{(k)}_{*i} + \sum_{i \in I_-} t_i cB^{(k)}_{*i}$. В силу того, что $I_+ \ne \emptyset$ имеем $\sum_{i \in I_+} t_i cB^{(k)}_{*i} > 0$. Пусть $M = \sum_{i \in I_+} t_i cB^{(k)}_{*i} - \delta$. Верно, что $M > 0$ в силу того, что $\delta \le 0$. Из за того, что $\delta = \sum_{i \in I_+} t_i cB^{(k)}_{*i} + \sum_{i \in I_-} t_i cB^{(k)}_{*i}$ также имеем $M = \sum_{i \in I_-} t_i(-cB^{(k)}_{*i})$.

Тогда можно переписать формулу для x следующим способом:

$$
\begin{aligned}
x = \sum_{i \in I_0} t_i B^{(k)}_{*i} + \sum_{i \in I_-} t_i B^{(k)}_{*i} + \sum_{i \in I_+} t_i B^{(k)}_{*i} \\
= \sum_{i \in I_0} t_i B^{(k)}_{*i} + 1/M \sum_{i \in I_-} (\sum_{j \in I_+} t_j cB^{(k)}_{*j} - \delta) t_i B^{(k)}_{*i} + 1/M \sum_{j \in I_+} (\sum_{i \in I_-} t_i(-cB^{(k)}_{*i})) t_j B^{(k)}_{*j} \\
= \sum_{i \in I_0} t_i B^{(k)}_{*i} + \sum_{i \in I_-} \frac{-\delta t_i}{M} B^{(k)}_{*i} + \sum_{i \in I_-} \sum_{j \in I_+} \frac{t_i t_j}{M} ((cB^{(k)}_{*j}) B^{(k)}_{*i} + (-cB^{(k)}_{*i}) B^{(k)}_{*j}).
\end{aligned}
$$

Получено явное выражение x через столбцы матрицы $B^{(k+1)}$. Таким образом доказано обратное включение.

Заметим, что в алгоритме коэффициенты α и β предлагается выбрать так, чтобы $\alpha cB^{(k)}_{*i} + \beta cB^{(k)}_{*j} = 0$. В формуле для x же выбор ограничен лишь $\alpha = cB^{(k)}_{*j}$ и $\beta = -cB^{(k)}_{*i}$.

Однако формулу можно модифицировать. Пусть для конкретных i, j уже найдены такие α_{ij} и β_{ij}, что $\alpha_{ij} cB^{(k)}_{*i} + \beta_{ij} cB^{(k)}_{*j} = 0$. Тогда можно переписать формулу для x следующим образом:

$$
x = \sum_{i \in I_0} t_i B^{(k)}_{*i} + \sum_{i \in I_-} \frac{-\delta t_i}{M} B^{(k)}_{*i} + \sum_{i \in I_-} \sum_{j \in I_+} \frac{t_i t_j cB^{(k)}_{*j}}{\alpha_{ij} M} (\alpha_{ij} B^{(k)}_{*i} + \beta_{ij} B^{(k)}_{*j}).
$$

■

Замечание 4 *Вычисления в алгоритме Фурье-Моцкина можно существенно сократить, и мы настоятельно советуем воспользоваться этим замечанием при выполнении практических заданий. Разобранный пример работы алгоритма с этим замечанием приведен в конце данного параграфа.*

Замечание состоит в следующем: оказывается нужно перебирать не все пары (i, j), где $i \in I_-$ и $j \in I_+$. Введем множество $J(t) = \{i : A^{(k)}_{i*} B^{(k)}_{*t} = 0\}$, элементы данного множества суть номера неравенств обратившихся в равенство на столбце $B^{(k)}_{*t}$. Тогда пусть $J = \{(i, j) : |J(i) \cap J(j)| \ge n - 2\}$.

Выбирать нужно такие пары $(i, j) : i \in I_-$ и $j \in I_+$, что также $(i, j) \in J$. Словесно последнее условие можно описать следующим способом: выбираем i, j таким образом, чтобы *количество неравенств одновременно обратившихся в равенство на векторах $B_{*i}^{(k)}$ и $B_{*j}^{(k)}$ было не меньше $n-2$.*

Пока мы не можем доказать справедливость этого замечания.

Для доказательства того, что для любого конуса вида $cone.hull(B)$ найдется матрица A, такая что $cone.hull(B) = C(A)$ нам нужны дополнительные леммы. Эти леммы и сами по себе представляют большой интерес. Например лемма Фаркаша является критерием образования неравенств следствий.

Лемма 3 (Лемма Фаркаша) *Дана матрица $A \in \mathbb{R}^{m\times n}$ и вектор строка $b^\top \in \mathbb{R}^n$. Неравенство $b^\top x \leq 0$ является следствием системы $Ax \leq 0$ тогда и только тогда, когда $b^\top = t^\top A$ для некоторого $t \in \mathbb{R}^m_+$, или другими словами $b \in cone.hull(A^\top)$.*

Доказательство. Докажем сперва достаточность. Если $Ax \leq 0$, то очевидно $bx = tAx \leq 0$.

Докажем необходимость. Пусть неравенство $b^\top x \leq 0$ является следствием системы $Ax \leq 0$, и предположим от противного, что не найдется такого $t^\top \in \mathbb{R}^m_+$, для которого верно $b^\top = t^\top A$. Это значит, что $b \notin cone.hull(A^\top)$. Откуда по теореме отделимости должно найтись такое $z^\top \in \mathbb{R}^m_+$, что $z^\top b > 0$, но $z^\top A^\top \leq 0$. Откуда получаем что $Az \leq 0$, но $b^\top z > 0$. Таким образом $b^\top x \leq 0$ не является неравенством следствием, получено противоречие. ∎

Лемма 4 *Пусть для пары матриц A и B выполнено $C(A) = cone.hull(B)$. Тогда $C(B^\top) = cone.hull(A^\top)$.*

Доказательство. Пусть $x \in cone.hull(A^\top)$, тогда существует вектор $t \geq 0$, такой что $x = A^\top t$. Тогда верно $B^\top A^\top t = (t^\top AB)^\top \leq 0$. Таким образом $x \in C(B^\top)$.

Пусть наоборот $x \in C(B^\top)$, тогда верно $B^\top x \leq 0$. Более того, для каждого вектора $t \geq 0$ верно $t^\top B^\top x \leq 0$ или, что эквивалентно $x^\top Bt \leq 0$. Так как $cone.hull(B) = C(A)$, то для каждого $y \in \{y : \; Ay \leq 0\}$ верно $x^\top y \leq$. Таким образом $x^\top y \leq 0$ является следствием системы $Ay \leq 0$. По лемме Фаркаша получаем, что $x \in cone.hull(A^\top)$. ∎

Теперь мы готовы доказать основную теорему этого раздела.

Теорема 5 (Минковский-Фаркаш-Вейль) *Пусть множество C есть конус, тогда следующие утверждения эквивалентны.*

1) Существует матрица A такая, что $C = C(A)$.

2) Существует матрица B, такая, что $C = cone.hull(B)$.

Доказательство. То что из 1) следует 2) было доказано алгоритмом Фурье-Моцкина. Следование 2) из 1) получается применением алгоритма Фурье-Моцкина к матрице $B^\top$, тогда из леммы 4 следует, что $cone.hull(B) = C(A)$. ∎

Замечание 5 *Заметим, что из алгоритма Фурье-Моцкина следует, что матрицы A и B принадлежат одному полю и все вычисления можно проводить в этом поле. Например если $C(A)$ представлен рациональной матрицей A, то все вычисления можно провести на компьютере используя рациональную арифметику, результатом будет рациональная матрица B, такая что $C(A) = cone.hull(B)$.*

Список литературы

[1] Шевченко, В.Н., Золотых, Н.Ю.: Линейное и целочисленное линейное программирование. Нижний Новгород Издательство Нижегородского гос. университета (2005)

[2] Емеличев, В.А., Ковалев, М.М., Кравцов, М.К.: Многогранники Графы Оптимизация. Москва Наука (1981)

[3] Циглер, Г.М.: Теория многогранников. Пер. с англ. под ред. Долбилина Н.П. Москва МЦНМО (2014)

[4] Схрейвер, А. Теория линейного и целочисленного программирования. Пер с англ. под ред. Хачияна Л.П. Москва Мир (1991)

[5] Schrijver, A.: Theory of Linear and Integer Programming. WileyInterscience series in discrete mathematics. John Wiley & Sons (1998)

[6] Ziegler, G.: Lectures on polytopes. Springer-Verlag, GTM 152 (1996)

[7] Grünbaum, B.: Convex Polytopes. Springer-Verlag, New York (2003)

Вариант 1

Задача 1

По данным из таблицы поставить задачу линейного программирования (транспортную задачу), то есть написать целевую функцию и систему линейных ограничений.

33	72	63	43	74	46	***48***
89	90	26	78	85	58	***72***
87	64	58	49	46	67	***90***
56	80	70	37	59	61	***30***
45	56	80	29	34	82	***50***
27	***38***	***55***	***57***	***33***	***80***	

Составить начальный опорный план двумя способами: методом северо-западного угла и методом минимального элемента. Для каждого начального опорного плана подсчитать стоимость перевозок. Найти оптимальный план для полученной транспортной задачи. Определить, единственен он или нет.

Задача 2

По данным из нижеприведенной таблицы поставить задачу линейного программирования на максимум и решить ее столбцовым симплекс-методом.

12	**13**	**15**	**9**	
4	5	8	5	***45***
8	7	10	7	***27***
12	8	6	8	***32***
3	9	10	6	***34***
7	3	8	3	***16***
2	5	4	8	***22***
10	6	7	9	18

Поставить двойственную задачу и решить ее методом Фурье.

Задача 3

По данным из нижеприведенной таблицы построить задачу линейного программирования на минимум и решить ее строчным симплекс-методом.

3	**8**	**5**	**6**	**4**	**12**	
3	9	7	2	3	7	***27***
9	7	1	3	9	8	***23***
7	5	2	9	7	4	***31***

Поставить двойственную задачу и решить ее методом Фурье.

Примечание: методом Фурье можно решить только одну из задач (2 или 3).

Вариант 2

Задача 1

По данным из таблицы поставить задачу линейного программирования (транспортную задачу), то есть написать целевую функцию и систему линейных ограничений.

33	72	63	43	74	46	***48***
89	90	26	78	85	58	***72***
87	64	58	29	46	67	***90***
56	80	70	37	59	61	***40***
45	56	80	29	54	82	***50***
27	***38***	***65***	***57***	***33***	***80***	

Составить начальный опорный план двумя способами: методом северо-западного угла и методом минимального элемента. Для каждого начального опорного плана подсчитать стоимость перевозок. Найти оптимальный план для полученной транспортной задачи. Определить, единственен он или нет.

Задача 2

По данным из нижеприведенной таблицы поставить задачу линейного программирования на максимум и решить ее столбцовым симплекс-методом.

12	**13**	**15**	**9**	
4	5	8	5	***45***
8	7	10	7	***24***
7	8	16	8	***32***
3	9	10	6	***35***
7	2	8	3	***16***
2	5	4	8	***22***
10	6	7	9	18

Поставить двойственную задачу и решить ее методом Фурье.

Задача 3

По данным из нижеприведенной таблицы построить задачу линейного программирования на минимум и решить ее строчным симплекс-методом.

3	**8**	**5**	**6**	**4**	**12**	
3	1	7	2	1	7	***27***
9	7	2	3	9	8	***23***
7	5	2	9	7	4	***31***

Поставить двойственную задачу и решить ее методом Фурье.

Примечание: методом Фурье можно решить только одну из задач (2 или 3).

Вариант 3

Задача 1

По данным из таблицы поставить задачу линейного программирования (транспортную задачу), то есть написать целевую функцию и систему линейных ограничений.

33	72	63	43	74	46	***48***
89	90	26	78	85	58	***72***
87	64	38	69	46	67	***90***
56	70	70	37	59	61	***42***
45	56	80	29	34	82	***38***
27	***38***	***35***	***77***	***33***	***80***	

Составить начальный опорный план двумя способами: методом северо-западного угла и методом минимального элемента. Для каждого начального опорного плана подсчитать стоимость перевозок. Найти оптимальный план для полученной транспортной задачи. Определить, единственен он или нет.

Задача 2

По данным из нижеприведенной таблицы поставить задачу линейного программирования на максимум и решить ее столбцовым симплекс-методом.

12	**12**	**16**	**9**	
4	5	8	5	***45***
8	7	10	8	***27***
12	8	6	8	***32***
3	9	8	6	***34***
7	4	8	3	***16***
2	5	4	8	***22***
10	6	7	9	***29***

Поставить двойственную задачу и решить ее методом Фурье.

Задача 3

По данным из нижеприведенной таблицы построить задачу линейного программирования на минимум и решить ее строчным симплекс-методом.

3	**8**	**5**	**6**	**4**	**12**	
3	9	7	2	4	7	***27***
9	8	1	3	9	8	***23***
7	5	2	9	7	4	***31***

Поставить двойственную задачу и решить ее методом Фурье.

Примечание: методом Фурье можно решить только одну из задач (2 или 3).

Вариант 4

Задача 1

По данным из таблицы поставить задачу линейного программирования (транспортную задачу), то есть написать целевую функцию и систему линейных ограничений.

33	72	63	43	74	46	***48***
89	90	26	78	85	59	***50***
87	64	58	49	46	67	***90***
56	80	44	37	59	61	***30***
45	56	80	29	34	82	***72***
27	***33***	***55***	***57***	***38***	***80***	

Составить начальный опорный план двумя способами: методом северо-западного угла и методом минимального элемента. Для каждого начального опорного плана подсчитать стоимость перевозок. Найти оптимальный план для полученной транспортной задачи. Определить, единственен он или нет.

Задача 2

По данным из нижеприведенной таблицы поставить задачу линейного программирования на максимум и решить ее столбцовым симплекс-методом.

12	**12**	**16**	**9**	
4	5	8	5	***45***
8	7	10	7	***27***
12	8	6	8	***32***
3	9	9	7	***34***
7	3	8	3	***16***
4	5	4	8	***22***
10	6	7	9	18

Поставить двойственную задачу и решить ее методом Фурье.

Задача 3

По данным из нижеприведенной таблицы построить задачу линейного программирования на минимум и решить ее строчным симплекс-методом.

3	**8**	**5**	**6**	**4**	**12**	
2	9	7	2	4	7	***27***
9	8	1	3	9	8	***23***
7	5	2	9	7	4	***31***

Поставить двойственную задачу и решить ее методом Фурье.

Примечание: методом Фурье можно решить только одну из задач (2 или 3).

Вариант 5

Задача 1

По данным из таблицы поставить задачу линейного программирования (транспортную задачу), то есть написать целевую функцию и систему линейных ограничений.

33	72	63	43	74	46	***48***
89	90	26	78	85	58	***72***
87	64	58	49	46	67	***100***
56	80	70	37	59	61	***30***
45	56	80	29	34	82	***50***
36	***38***	***55***	***57***	***34***	***80***	

Составить начальный опорный план двумя способами: методом северо-западного угла и методом минимального элемента. Для каждого начального опорного плана подсчитать стоимость перевозок. Найти оптимальный план для полученной транспортной задачи. Определить, единственен он или нет.

Задача 2

По данным из нижеприведенной таблицы поставить задачу линейного программирования на максимум и решить ее столбцовым симплекс-методом.

12	**13**	**15**	**9**	
4	5	8	5	***45***
8	7	9	7	***27***
12	8	6	8	***32***
3	9	10	5	***34***
7	3	8	3	***16***
2	5	4	8	***22***
10	6	7	9	18

Поставить двойственную задачу и решить ее методом Фурье.

Задача 3

По данным из нижеприведенной таблицы построить задачу линейного программирования на минимум и решить ее строчным симплекс-методом.

3	**8**	**5**	**6**	**4**	**12**	
3	8	7	2	3	7	***27***
9	7	1	3	9	8	***25***
7	5	2	8	7	4	***31***

Поставить двойственную задачу и решить ее методом Фурье.

Примечание: методом Фурье можно решить только одну из задач (2 или 3).

Вариант 6

Задача 1

По данным из таблицы поставить задачу линейного программирования (транспортную задачу), то есть написать целевую функцию и систему линейных ограничений.

33	72	63	43	74	46	***48***
89	90	26	78	85	58	***72***
87	64	58	49	46	67	***90***
56	80	70	37	59	61	***30***
45	56	77	29	34	82	***50***
27	***38***	***55***	***33***	***57***	***80***	

Составить начальный опорный план двумя способами: методом северо-западного угла и методом минимального элемента. Для каждого начального опорного плана подсчитать стоимость перевозок. Найти оптимальный план для полученной транспортной задачи. Определить, единственен он или нет.

Задача 2

По данным из нижеприведенной таблицы поставить задачу линейного программирования на максимум и решить ее столбцовым симплекс-методом.

12	**13**	**15**	**9**	
4	5	8	5	***45***
8	7	10	7	***27***
12	8	6	8	***32***
3	9	10	6	***16***
7	3	8	3	***16***
2	5	4	8	***34***
10	6	7	9	18

Поставить двойственную задачу и решить ее методом Фурье.

Задача 3

По данным из нижеприведенной таблицы построить задачу линейного программирования на минимум и решить ее строчным симплекс-методом.

3	**8**	**5**	**6**	**4**	**12**	
3	9	7	2	3	7	***25***
9	7	1	4	9	8	***23***
7	5	2	9	7	4	***32***

Поставить двойственную задачу и решить ее методом Фурье.

Примечание: методом Фурье можно решить только одну из задач (2 или 3).

Вариант 8

Задача 1

По данным из таблицы поставить задачу линейного программирования (транспортную задачу), то есть написать целевую функцию и систему линейных ограничений.

33	72	63	43	74	46	***48***
89	90	26	78	85	58	***72***
87	64	58	48	46	67	***90***
54	80	70	37	59	61	***30***
45	56	80	29	34	82	***50***
27	***38***	***55***	***57***	***33***	***80***	

Составить начальный опорный план двумя способами: методом северо-западного угла и методом минимального элемента. Для каждого начального опорного плана подсчитать стоимость перевозок. Найти оптимальный план для полученной транспортной задачи. Определить, единственен он или нет.

Задача 2

По данным из нижеприведенной таблицы поставить задачу линейного программирования на максимум и решить ее столбцовым симплекс-методом.

12	**13**	**15**	**9**	
4	5	8	5	***45***
8	7	11	7	***27***
12	8	6	8	***32***
3	9	10	6	***34***
7	3	8	3	***16***
2	5	4	8	***22***
10	6	7	9	***18***

Поставить двойственную задачу и решить ее методом Фурье.

Задача 3

По данным из нижеприведенной таблицы построить задачу линейного программирования на минимум и решить ее строчным симплекс-методом.

3	**8**	**5**	**6**	**4**	**12**	
3	9	7	2	3	7	***27***
9	7	1	3	11	8	***23***
7	5	2	9	7	4	***31***

Поставить двойственную задачу и решить ее методом Фурье.

Примечание: методом Фурье можно решить только одну из задач (2 или 3).

Вариант 9

Задача 1

По данным из таблицы поставить задачу линейного программирования (транспортную задачу), то есть написать целевую функцию и систему линейных ограничений.

33	72	63	43	74	46	***48***
89	90	26	78	85	58	***72***
87	64	58	49	46	56	***90***
64	80	70	37	59	61	***30***
45	56	80	29	34	82	***60***
27	***38***	***55***	***57***	***43***	***80***	

Составить начальный опорный план двумя способами: методом северо-западного угла и методом минимального элемента. Для каждого начального опорного плана подсчитать стоимость перевозок. Найти оптимальный план для полученной транспортной задачи. Определить, единственен он или нет.

Задача 2

По данным из нижеприведенной таблицы поставить задачу линейного программирования на максимум и решить ее столбцовым симплекс-методом.

12	**13**	**15**	**9**	
4	5	8	5	***45***
8	7	10	7	***27***
12	8	6	8	***32***
3	9	10	6	***34***
7	3	9	3	***16***
2	5	4	8	***22***
10	6	7	9	***18***

Поставить двойственную задачу и решить ее методом Фурье.

Задача 3

По данным из нижеприведенной таблицы построить задачу линейного программирования на минимум и решить ее строчным симплекс-методом.

3	**8**	**5**	**6**	**4**	**12**	
3	9	7	2	3	7	***32***
9	7	1	4	9	8	***16***
7	5	2	9	7	4	***24***

Поставить двойственную задачу и решить ее методом Фурье.

Примечание: методом Фурье можно решить только одну из задач (2 или 3).

Вариант 10

Задача 1

По данным из таблицы поставить задачу линейного программирования (транспортную задачу), то есть написать целевую функцию и систему линейных ограничений.

33	72	63	43	74	46	***58***
89	90	26	78	85	58	***72***
87	64	58	49	46	67	***90***
56	44	70	37	59	61	***30***
45	56	80	29	34	72	***50***
27	***48***	***55***	***57***	***33***	***80***	

Составить начальный опорный план двумя способами: методом северо-западного угла и методом минимального элемента. Для каждого начального опорного плана подсчитать стоимость перевозок. Найти оптимальный план для полученной транспортной задачи. Определить, единственен он или нет.

Задача 2

По данным из нижеприведенной таблицы поставить задачу линейного программирования на максимум и решить ее столбцовым симплекс-методом.

12	**13**	**15**	**9**	
4	5	8	5	***45***
8	7	10	7	***27***
12	8	6	8	***32***
3	9	10	6	***34***
4	3	8	3	***16***
8	5	4	8	***22***
10	6	7	9	***18***

Поставить двойственную задачу и решить ее методом Фурье.

Задача 3

По данным из нижеприведенной таблицы построить задачу линейного программирования на минимум и решить ее строчным симплекс-методом.

3	**8**	**5**	**6**	**4**	**12**	
3	9	7	2	3	7	***26***
9	7	1	3	9	8	***23***
7	5	2	9	7	14	***32***

Поставить двойственную задачу и решить ее методом Фурье.

Примечание: методом Фурье можно решить только одну из задач (2 или 3).

Вариант 11

Задача 1

По данным из таблицы поставить задачу линейного программирования (транспортную задачу), то есть написать целевую функцию и систему линейных ограничений.

33	72	63	43	74	46	***48***
89	90	26	78	85	58	***72***
87	64	58	49	46	67	***95***
56	80	70	37	59	61	***35***
45	56	80	29	34	82	***50***
29	***38***	***55***	***57***	***39***	***84***	

Составить начальный опорный план двумя способами: методом северо-западного угла и методом минимального элемента. Для каждого начального опорного плана подсчитать стоимость перевозок. Найти оптимальный план для полученной транспортной задачи. Определить, единственен он или нет.

Задача 2

По данным из нижеприведенной таблицы поставить задачу линейного программирования на максимум и решить ее столбцовым симплекс-методом.

12	**13**	**15**	**9**	
4	5	8	5	***45***
8	7	11	7	***27***
12	8	6	8	***32***
3	9	10	6	***34***
7	3	8	3	***32***
2	5	4	8	***22***
10	6	7	9	***18***

Поставить двойственную задачу и решить ее методом Фурье.

Задача 3

По данным из нижеприведенной таблицы построить задачу линейного программирования на минимум и решить ее строчным симплекс-методом.

3	**8**	**5**	**6**	**4**	**12**	
3	9	7	2	3	7	***27***
9	7	1	4	9	8	***23***
7	5	2	9	7	4	***31***

Поставить двойственную задачу и решить ее методом Фурье.

Примечание: методом Фурье можно решить только одну из задач (2 или 3).

Вариант 12

Задача 1

По данным из таблицы поставить задачу линейного программирования (транспортную задачу), то есть написать целевую функцию и систему линейных ограничений.

33	72	63	43	74	46	***48***
89	90	32	78	85	58	***72***
87	64	58	49	46	67	***90***
56	80	70	37	59	61	***30***
45	56	80	29	34	82	***50***
27	***38***	***55***	***57***	***33***	***80***	

Составить начальный опорный план двумя способами: методом северо-западного угла и методом минимального элемента. Для каждого начального опорного плана подсчитать стоимость перевозок. Найти оптимальный план для полученной транспортной задачи. Определить, единственен он или нет.

Задача 2

По данным из нижеприведенной таблицы поставить задачу линейного программирования на максимум и решить ее столбцовым симплекс-методом.

12	**13**	**15**	**9**	
4	5	8	5	***45***
8	7	10	7	***27***
12	8	6	8	***32***
3	9	7	6	***34***
7	3	8	3	***16***
2	5	4	8	***22***
10	6	7	9	***18***

Поставить двойственную задачу и решить ее методом Фурье.

Задача 3

По данным из нижеприведенной таблицы построить задачу линейного программирования на минимум и решить ее строчным симплекс-методом.

3	**8**	**5**	**6**	**4**	**12**	
3	9	7	2	3	7	***32***
9	7	1	3	4	8	***25***
7	5	2	9	7	4	***31***

Поставить двойственную задачу и решить ее методом Фурье.

Примечание: методом Фурье можно решить только одну из задач (2 или 3).

Вариант 13

Задача 1

По данным из таблицы поставить задачу линейного программирования (транспортную задачу), то есть написать целевую функцию и систему линейных ограничений.

33	72	63	43	74	46	***48***
89	90	26	64	85	58	***72***
87	64	58	49	46	67	***90***
56	80	70	37	59	61	***35***
45	56	80	29	34	82	***55***
32	***38***	***55***	***57***	***38***	***80***	

Составить начальный опорный план двумя способами: методом северо-западного угла и методом минимального элемента. Для каждого начального опорного плана подсчитать стоимость перевозок. Найти оптимальный план для полученной транспортной задачи. Определить, единственен он или нет.

Задача 2

По данным из нижеприведенной таблицы поставить задачу линейного программирования на максимум и решить ее столбцовым симплекс-методом.

12	**13**	**15**	**9**	
4	5	8	5	***45***
8	7	10	7	***27***
12	8	6	8	***32***
3	9	10	6	***34***
7	4	8	3	***16***
2	5	4	8	***22***
10	6	7	9	***18***

Поставить двойственную задачу и решить ее методом Фурье.

Задача 3

По данным из нижеприведенной таблицы построить задачу линейного программирования на минимум и решить ее строчным симплекс-методом.

3	**8**	**5**	**6**	**4**	**12**	
3	9	7	2	3	7	***32***
9	7	1	3	9	8	***23***
7	5	2	9	7	4	***31***

Поставить двойственную задачу и решить ее методом Фурье.

Примечание: методом Фурье можно решить только одну из задач (2 или 3).

Вариант 14

Задача 1

По данным из таблицы поставить задачу линейного программирования (транспортную задачу), то есть написать целевую функцию и систему линейных ограничений.

33	72	63	43	74	46	***48***
89	90	26	78	85	58	***72***
87	64	58	52	46	67	***90***
56	80	70	37	59	61	***30***
45	56	80	29	34	82	***50***
27	***38***	***55***	***57***	***33***	***80***	

Составить начальный опорный план двумя способами: методом северо-западного угла и методом минимального элемента. Для каждого начального опорного плана подсчитать стоимость перевозок. Найти оптимальный план для полученной транспортной задачи. Определить, единственен он или нет.

Задача 2

По данным из нижеприведенной таблицы поставить задачу линейного программирования на максимум и решить ее столбцовым симплекс-методом.

12	**13**	**15**	**9**	
4	5	8	5	***45***
8	7	10	7	***27***
1	8	6	8	***32***
3	9	10	6	***35***
7	3	8	3	***16***
2	5	4	8	***25***
10	6	7	9	***18***

Поставить двойственную задачу и решить ее методом Фурье.

Задача 3

По данным из нижеприведенной таблицы построить задачу линейного программирования на минимум и решить ее строчным симплекс-методом.

3	**8**	**5**	**6**	**4**	**12**	
3	9	7	2	3	7	***27***
9	7	1	3	9	8	***23***
7	5	2	9	7	4	***25***

Поставить двойственную задачу и решить ее методом Фурье.

Примечание: методом Фурье можно решить только одну из задач (2 или 3).

Вариант 15

Задача 1

По данным из таблицы поставить задачу линейного программирования (транспортную задачу), то есть написать целевую функцию и систему линейных ограничений.

33	72	63	43	74	46	***48***
89	90	26	78	85	58	***72***
87	64	58	49	63	67	***90***
56	80	72	37	59	61	***30***
45	56	80	29	34	82	***50***
27	***38***	***55***	***57***	***33***	***80***	

Составить начальный опорный план двумя способами: методом северо-западного угла и методом минимального элемента. Для каждого начального опорного плана подсчитать стоимость перевозок. Найти оптимальный план для полученной транспортной задачи. Определить, единственен он или нет.

Задача 2

По данным из нижеприведенной таблицы поставить задачу линейного программирования на максимум и решить ее столбцовым симплекс-методом.

12	**13**	**15**	**9**	
4	5	8	5	***45***
8	7	10	7	***27***
12	5	6	8	***32***
3	9	10	6	***34***
7	3	8	3	***16***
2	5	4	8	***22***
10	6	7	9	***18***

Поставить двойственную задачу и решить ее методом Фурье.

Задача 3

По данным из нижеприведенной таблицы построить задачу линейного программирования на минимум и решить ее строчным симплекс-методом.

3	**8**	**5**	**6**	**4**	**12**	
3	9	7	2	3	7	***27***
9	7	1	3	9	8	***25***
7	5	2	9	7	4	***31***

Поставить двойственную задачу и решить ее методом Фурье.

Примечание: методом Фурье можно решить только одну из задач (2 или 3).

Вариант 16

Задача 1

По данным из таблицы поставить задачу линейного программирования (транспортную задачу), то есть написать целевую функцию и систему линейных ограничений.

33	72	63	43	74	46	***48***
89	90	26	78	85	58	***72***
87	64	58	49	46	67	***90***
56	80	70	38	59	61	***30***
45	56	80	29	34	82	***50***
27	***38***	***55***	***57***	***33***	***80***	

Составить начальный опорный план двумя способами: методом северо-западного угла и методом минимального элемента. Для каждого начального опорного плана подсчитать стоимость перевозок. Найти оптимальный план для полученной транспортной задачи. Определить, единственен он или нет.

Задача 2

По данным из нижеприведенной таблицы поставить задачу линейного программирования на максимум и решить ее столбцовым симплекс-методом.

12	**13**	**15**	**9**	
4	5	8	5	***45***
8	7	10	7	***27***
12	8	6	8	***32***
3	9	10	6	***34***
7	5	8	3	***16***
2	5	4	8	***22***
10	6	7	9	***18***

Поставить двойственную задачу и решить ее методом Фурье.

Задача 3

По данным из нижеприведенной таблицы построить задачу линейного программирования на минимум и решить ее строчным симплекс-методом.

3	**8**	**5**	**6**	**4**	**12**	
5	9	7	2	3	7	***27***
9	7	1	3	9	8	***23***
7	5	2	9	7	4	***31***

Поставить двойственную задачу и решить ее методом Фурье.

Примечание: методом Фурье можно решить только одну из задач (2 или 3).

Вариант 17

Задача 1

По данным из таблицы поставить задачу линейного программирования (транспортную задачу), то есть написать целевую функцию и систему линейных ограничений.

42	72	63	43	74	46	***48***
89	90	26	78	85	58	***72***
87	64	58	49	46	67	***90***
56	80	70	37	59	61	***30***
45	56	80	29	34	82	***50***
27	***38***	***55***	***57***	***33***	***80***	

Составить начальный опорный план двумя способами: методом северо-западного угла и методом минимального элемента. Для каждого начального опорного плана подсчитать стоимость перевозок. Найти оптимальный план для полученной транспортной задачи. Определить, единственен он или нет.

Задача 2

По данным из нижеприведенной таблицы поставить задачу линейного программирования на максимум и решить ее столбцовым симплекс-методом.

12	**13**	**15**	**9**	
4	5	8	5	***45***
8	7	10	7	***27***
12	8	6	8	***32***
3	9	10	5	***34***
7	3	8	3	***16***
2	5	4	8	***22***
10	6	7	9	*18*

Поставить двойственную задачу и решить ее методом Фурье.

Задача 3

По данным из нижеприведенной таблицы построить задачу линейного программирования на минимум и решить ее строчным симплекс-методом.

3	**8**	**5**	**6**	**4**	**12**	
3	1	7	2	3	1	***27***
9	7	1	3	9	8	***23***
7	5	2	9	7	4	***31***

Поставить двойственную задачу и решить ее методом Фурье.

Примечание: методом Фурье можно решить только одну из задач (2 или 3).

Вариант 18

Задача 1

По данным из таблицы поставить задачу линейного программирования (транспортную задачу), то есть написать целевую функцию и систему линейных ограничений.

33	72	63	43	74	46	***48***
89	90	26	78	85	58	***82***
87	64	58	79	46	67	***90***
66	80	70	37	59	61	***30***
45	56	80	29	34	82	***50***
27	***38***	***65***	***57***	***33***	***80***	

Составить начальный опорный план двумя способами: методом северо-западного угла и методом минимального элемента. Для каждого начального опорного плана подсчитать стоимость перевозок. Найти оптимальный план для полученной транспортной задачи. Определить, единственен он или нет.

Задача 2

По данным из нижеприведенной таблицы поставить задачу линейного программирования на максимум и решить ее столбцовым симплекс-методом.

12	**13**	**15**	**9**	
4	5	8	5	***45***
8	17	10	7	***27***
12	18	6	8	***32***
3	9	10	6	***34***
7	3	8	13	***16***
2	5	4	8	***42***
10	6	17	9	***58***

Поставить двойственную задачу и решить ее методом Фурье.

Задача 3

По данным из нижеприведенной таблицы построить задачу линейного программирования на минимум и решить ее строчным симплекс-методом.

3	**8**	**5**	**6**	**4**	**12**	
3	9	7	2	3	7	***127***
9	17	1	13	9	8	***123***
7	5	12	9	7	14	***131***

Поставить двойственную задачу и решить ее методом Фурье.

Примечание: методом Фурье можно решить только одну из задач (2 или 3).

Вариант 19

Задача 1

По данным из таблицы поставить задачу линейного программирования (транспортную задачу), то есть написать целевую функцию и систему линейных ограничений.

33	72	63	43	74	46	***48***
89	90	26	78	85	58	***72***
87	64	58	29	46	67	***90***
56	80	70	37	59	61	***40***
45	56	80	29	54	82	***50***
27	***38***	***65***	***57***	***33***	***80***	

Составить начальный опорный план двумя способами: методом северо-западного угла и методом минимального элемента. Для каждого начального опорного плана подсчитать стоимость перевозок. Найти оптимальный план для полученной транспортной задачи. Определить, единственен он или нет.

Задача 2

По данным из нижеприведенной таблицы поставить задачу линейного программирования на максимум и решить ее столбцовым симплекс-методом.

12	**13**	**15**	**9**	
4	5	8	5	***45***
8	7	10	7	***24***
7	8	16	8	***32***
3	9	10	6	***35***
7	2	8	3	***16***
2	5	4	8	***22***
10	6	7	9	18

Поставить двойственную задачу и решить ее методом Фурье.

Задача 3

По данным из нижеприведенной таблицы построить задачу линейного программирования на минимум и решить ее строчным симплекс-методом.

3	**8**	**5**	**6**	**4**	**12**	
3	1	7	2	1	7	***27***
9	7	2	3	9	8	***23***
7	5	2	9	7	4	***31***

Поставить двойственную задачу и решить ее методом Фурье.

Примечание: методом Фурье можно решить только одну из задач (2 или 3).

Вариант 20

Задача 1

По данным из таблицы поставить задачу линейного программирования (транспортную задачу), то есть написать целевую функцию и систему линейных ограничений.

33	72	63	43	74	46	***48***
89	90	26	78	85	58	***82***
87	64	58	79	46	67	***90***
66	80	70	37	59	61	***30***
45	56	80	29	34	82	***50***
27	***38***	***65***	***57***	***33***	***80***	

Составить начальный опорный план двумя способами: методом северо-западного угла и методом минимального элемента. Для каждого начального опорного плана подсчитать стоимость перевозок. Найти оптимальный план для полученной транспортной задачи. Определить, единственен он или нет.

Задача 2

По данным из нижеприведенной таблицы поставить задачу линейного программирования на максимум и решить ее столбцовым симплекс-методом.

12	**13**	**15**	**9**	
4	5	8	5	***45***
8	17	10	17	***27***
12	18	26	8	***32***
3	9	10	6	***34***
7	3	8	13	***116***
2	5	4	8	***42***
10	6	17	9	***158***

Поставить двойственную задачу и решить ее методом Фурье.

Задача 3

По данным из нижеприведенной таблицы построить задачу линейного программирования на минимум и решить ее строчным симплекс-методом.

3	**8**	**5**	**6**	**4**	**12**	
3	9	27	42	3	27	***127***
9	17	12	13	9	8	***123***
7	5	12	9	7	14	***131***

Поставить двойственную задачу и решить ее методом Фурье.

Примечание: методом Фурье можно решить только одну из задач (2 или 3).