Курс: Технологии компьютерного проектирования и оптимизации комплексных систем (ТКПОКС)

Тема 2: Автоматизация системного проектирования

- Задачи и методы параметрической оптимизации
- Методы и алгоритмы решения задач линейного программирования
  - Табличный симплекс-метод
  - Венгерский метод

Постановка задачи параметрической оптимизации

Параметрическая оптимизация – процедура определения внутренних параметров проектируемого объекта заданной структуры, при которых достигается наилучшее сочетание его свойств.

Параметрическая оптимизация включает:

- переход от исходной неформальной постановки задачи, выраженной на качественном уровне назначения объекта и требований к его свойствам до математической постановки;
- решение задачи сформулированной постановки.

Постановка задачи параметрической оптимизации

Формализация задачи оптимизации сводится к её формулированию в виде задачи математического программирования:

$$\left.\begin{array}{l} \underset{\mathbf{X}\in\mathbf{XD}}{\mathrm{extr}}\ F(\mathbf{X}), \\ \mathbf{XD} = \{\mathbf{X} \mid \mathbf{\Phi}(\mathbf{X}) \geqslant 0,\ \mathbf{\Psi}(\mathbf{X}) = 0\}, \end{array}\right\}$$

где F(X) – целевая функция

X – вектор управляемых параметров

XD – область варьирования X

Выходной параметр – величина, характеризующая определённое свойство проектируемого объекта.

Функции Φ(X), F(X), Ψ(X) непосредственно или косвенно связаны с выходными параметрами $\mathbf{y}_j(\mathbf{X})$, $j = \overline{1, m}$.

Линейное программирование (ЛП)

Формы записи задач линейного программирования.

Задача линейного программирования **в общем виде** записывается следующим образом: найти минимум (максимум) целевой функции $f(x)=c_1x_1+c_2x_2+\ldots+c_nx_n$, где $(c_1,c_2,\ldots,c_n)=\overline{c}$ – вектор параметров целевой функции, и удовлетворяющий системе ограничений:

$$\begin{cases} \sum\limits_{j=1}^{n} a_{ij}x_j \le b_i, \quad i=1,\ldots m_1, \\ \sum\limits_{j=1}^{n} a_{ij}x_j = b_i, \quad i=m_1+1,\ldots m_2, \\ \sum\limits_{j=1}^{n} a_{ij}x_j \ge b_i, \quad i=m_2+1,\ldots m_3, \\ x_j \ge 0, \; j=1,\ldots n. \end{cases}$$

Т. е. система ограничений содержит как ограничения-равенства, так и ограничения-неравенства.

Задача линейного программирования имеет **каноническую форму**, если все ее ограничения записаны в форме равенств (кроме ограничений неотрицательности переменных $x_j \geq 0, \quad j = 1,...n$), т.е.

$$\begin{aligned} &\min f(x) = (c, x) = c^* x, \\ &Ax = b, \\ &x \geq 0, \end{aligned}$$

где A – действительная матрица ограничений; b – вектор правой части ограничений; c – вектор-строка коэффициентов целевой функции.

Задача линейного программирования имеет **стандартную (основную) форму**, если ограничения имеют вид

$$\begin{aligned} &\sum_{j=1}^{n} a_{ij} x_j \leq b_i, \quad i = 1,...m; \\ &x_j \geq 0, \quad j = 1,...n. \end{aligned}$$

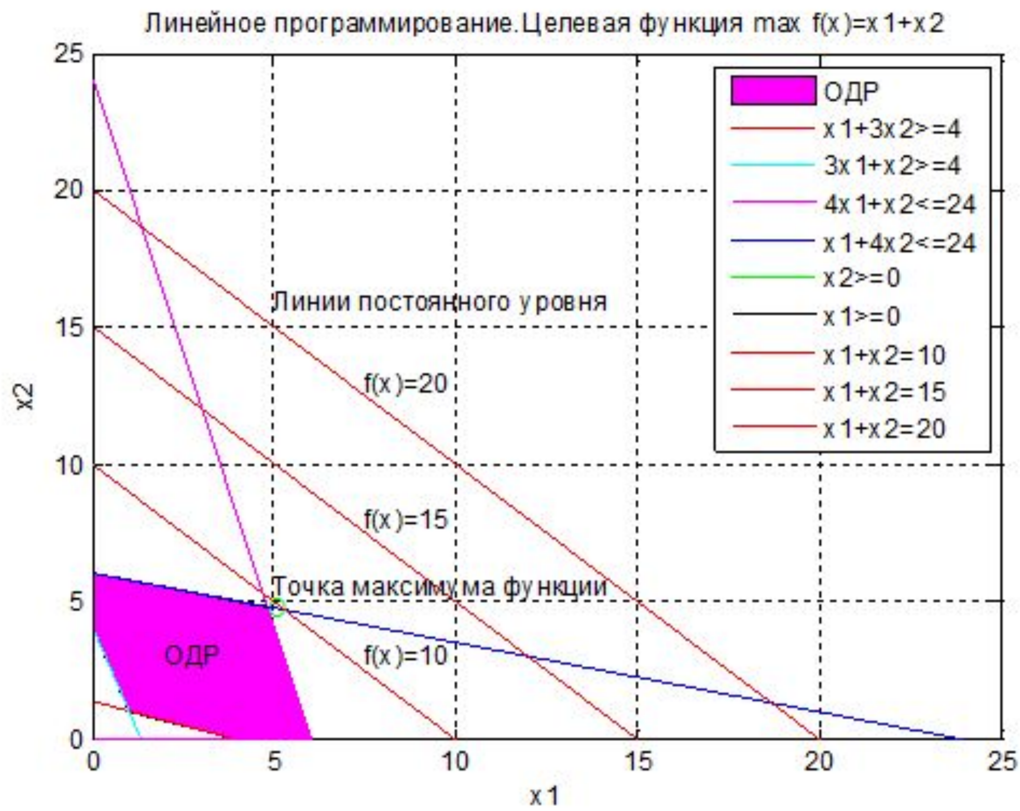
Упорядоченный набор $x = (x_1, x_2, ..., x_n)$, который удовлетворяет всем ограничениям, называется **допустимым планом** задачи линейного программирования.

Оптимальным планом задачи линейного программирования называется такой допустимый план x^*, для которого выполняется условие:

- для всех допустимых планов выполняются неравенства $f(x^*) \geq f(x)$.

Таким образом, ограничение-неравенство $\sum_{j=1}^{n} a_{ij} x_j \leq b_i$ преобразуется в ограничение-равенство $\sum_{j=1}^{n} a_{ij} x_j + x_{n+1} = b_i,\ x_{n+1} \geq 0$, а ограничение-неравенство $\sum_{j=1}^{n} a_{ij} x_j \geq b_i$ преобразуется в ограничение-равенство $\sum_{j=1}^{n} a_{ij} x_j - x_{n+1} = b_i,\ x_{n+1} \geq 0$.

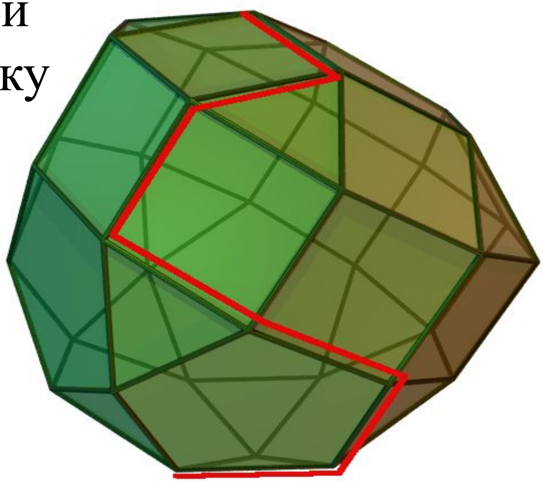
Если переменная x_k по условию задачи отрицательна, то ее следует заменить двумя неотрицательными переменными x_{n+1} и x_{n+2}, приняв $x_k = x_{n+1} - x_{n+2}$.


Линейное программирование.Целевая функция max f(x)=x1+x2
ОДР
x1+3x2>=4
3x1+x2>=4
4x1+x2<=24
x1+4x2<=24
x2>=0
x1>=0
x1+x2=10
x1+x2=15
x1+x2=20
Линии постоянного уровня
f(x)=20
f(x)=15
f(x)=10
Точка максимума функции
ОДР
x1
x2
0
5
10
15
20
25

Симплекс-метод

- предложен Данцигом (1951 г.)

Идея состоит в продвижении по выпуклому многограннику ограничений от вершины к вершине, при котором на каждом шаге значение целевой функции улучшается пока не будет достигнут оптимум.



Алгоритм симплекс-метода

Подготовительный этап

Приводим задачу ЛП к каноническому виду

$F=a_{0,1}x_1+a_{0,2}x_2+\ldots a_{0,n}x_n+b_0 \rightarrow \max$

$a_{1,1}x_1+a_{1,2}x_2+\ldots a_{1,n}x_n+x_{n+1}=b_1$

$a_{2,1}x_1+a_{2,2}x_2+\ldots a_{2,n}x_n+x_{n+2}=b_2$

....

$a_{m,1}x_1+a_{m,2}x_2+\ldots a_{m,n}x_n+x_{n+m}=b_m$

если в исходной задаче необходимо найти минимум - знаки коэффициентов целевой функции F меняются на противоположные $a_{0,n}= -a_{0,n}$. Знаки коэффициентов ограничивающих условий со знаком "≥" так же меняются на противоположные. В случае если условие содержит знак "≤" - коэффициенты запишутся без изменений.

Шаг 0. Составляем симплексную таблицу, соответствующую исходной задаче

	x_1	x_2	…	x_{n-1}	x_n	b
F	$-a_{0,1}$	$-a_{0,2}$	…	$-a_{0,n-1}$	$-a_{0,n}$	$-b_0$
x_{n+1}	$a_{1,1}$	$a_{1,2}$	…	$a_{1,n-1}$	$a_{1,n}$	b_1
x_{n+2}	$a_{2,1}$	$a_{2,2}$	…	$a_{2,n-1}$	$a_{2,n}$	b_2
…	…	…	…	…	…	…
x_{n+m}	$a_{m,1}$	$a_{m,2}$	…	$a_{m,n-1}$	$a_{m,n}$	b_m

x_1, x_2, x_n - исходные переменные,
x_{n+1}, x_{n+2}, x_{n+m} - дополнительные переменные (базисные).

Шаг 1. Проверка на допустимость

Проверяем на положительность элементы столбца **b**, если среди них нет отрицательных то найдено допустимое решение, переходим к **шагу 2.**

Если в столбце **b** имеются отрицательные элементы то выбираем среди них максимальный по модулю - он задает ведущую строку **k**. В этой строке находим максимальный по модулю отрицательный элемент $\mathbf{a}_{\mathbf{k,l}}$ - он задает ведущий столбец - **l** и является *ведущим элементом.*

Переменная, соответствующая ведущей строке исключается из базиса, переменная соответствующая ведущему столбцу включается в базис. Пересчитываем симплекс-таблицу согласно ***правилам***.

Если же среди свободных членов есть отрицательные элементы - а в соответствующей строке - нет то условия задачи несовместны и решений у нее нет.

Шаг 2. Проверка на оптимальность

На предыдущем этапе найдено допустимое решение. Проверим его на оптимальность.

Если в строке F (не беря в расчет элемент b_0 - текущее значение целевой функции) нет отрицательных, то найдено оптимальное решение.

Если в строке F есть отрицательные элементы то решение требует улучшения. Выбираем среди отрицательных элементов строки F максимальный по модулю (исключая значение функции b_0)

$\mathbf{a_{0,l}=min\{a_{0,i}\}}$ l – ведущий столбец

$\mathbf{b_k/a_{k,l}=min\ \{b_i/a_{i,l}\}}$ при $a_{i,l}>0$, $b_i>0$

k – ведущая строка.

Элемент $\mathbf{a_{k,l}}$ - ведущий (разрешающий).

*Пересчитываем симплекс-таблицу по **формулам**.*

Если в новой таблице после перерасчета в строке F остались отрицательные элементы переходим к **шагу 2**

Если невозможно найти ведущую строку, так как нет положительных элементов в ведущем столбце, то функция в области допустимых решений задачи не ограничена - алгоритм завершает работу.

Если в строке **F** и в столбце свободных **b** членов все элементы положительные, то найдено оптимальное решение.

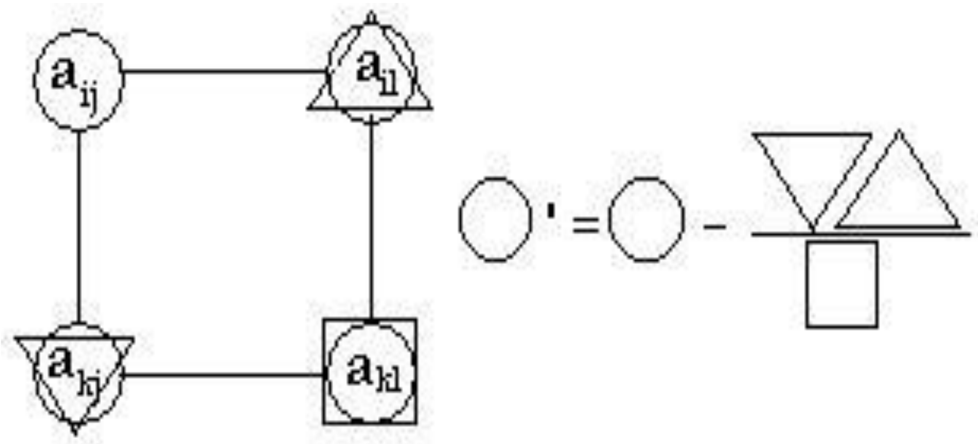
Правила преобразований симплексной таблицы

При составлении новой симплекс-таблицы в ней происходят следующие изменения:

- вместо базисной переменной x_k записываем x_l; вместо небазисной переменной x_l записываем x_k.
- ведущий элемент заменяется на обратную величину $a_{k,l}' = 1/a_{k,l}$
- все элементы ведущего столбца (кроме $a_{k,l}$) умножаются на $-1/a_{k,l}$
- все элементы ведущей строки (кроме $a_{k,l}$) умножаются на $1/a_{k,l}$
- оставшиеся элементы симплекс-таблицы преобразуются по формуле $a_{i,j}' = a_{i,j} - a_{i,l} \times a_{k,j} / a_{k,l}$

Правила преобразований симплексной таблицы

Схему преобразования элементов симплекс-таблицы (кроме ведущей строки и ведущего столбца) называют схемой ”прямоугольника”.



Пример решения задачи линейного программирования симплекс методом

Целевая функция:

$$2x_1+5x_2+3x_3+8x_4 \to \min$$

Ограничивающие условия:

$$3x_1+6x_2-4x_3+x_4\leq 12$$
$$4x_1-13x_2+10x_3+5x_4\geq 6$$
$$3x_1+7x_2+x_3\geq 1$$

Приведем систему ограничений к каноническому виду:

$$-2x_1-5x_2-3x_3-8x_4 \to \max$$
$$3x_1+6x_2-4x_3+x_4+x_5= 12$$
$$-4x_1+13x_2-10x_3-5x_4+x_6= -6$$
$$-3x_1-7x_2-x_3+x_7= -1$$

Формирование исходной симплекс таблицы

	x1	x2	x3	x4	b
F	2	5	3	8	0
x5	3	6	-4	1	12
x6	-4	13	**-10**	-5	-6
x7	-3	-7	-1	0	-1

Пересчитаем симплекс-таблицу:

	x1	x2	x6	x4	b
F	0.8	8.9	0.3	6.5	-1.8
x5	4.6	0.8	-0.4	3	14.4
x3	0.4	-1.3	-0.1	0.5	0.6
x7	-2.6	**-8.3**	-0.1	0.5	-0.4

Пересчитаем симплекс-таблицу:

	x1	**x7**	**x6**	**x4**	**b**
F	-1.988	1.072	0.193	7.036	-2.229
x5	4.349	0.096	-0.41	3.048	14.361
x3	0.807	-0.157	-0.084	0.422	0.663
x2	**0.313**	-0.12	0.012	-0.06	0.048

Ведущей строкой является X2, а ведущий элемент: 0.313.

Пересчитаем симплекс-таблицу:

	x2	**x7**	**x6**	**x4**	**b**
f	6.351	0.31	0.269	6.655	**-1.924**
x5	-13.895	1.763	-0.577	3.882	13.694
x3	-2.578	0.152	-0.115	0.577	0.539
x1	3.195	-0.383	0.038	-0.192	0.153

Решение задачи о назначении
(Венгерский метод)

Постановка задачи

$$\{C_{1j_1}, C_{2j_2}, \ldots, C_{nj_n}\}$$

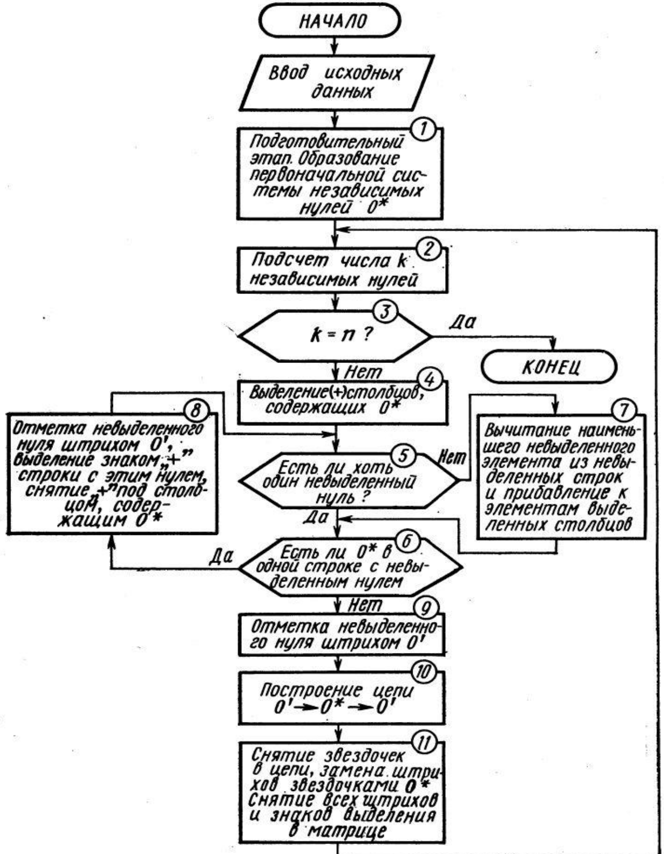
$$\mathbf{C} = \begin{bmatrix} C_{11} & C_{12} & \ldots & C_{1n} \\ C_{21} & C_{22} & \ldots & C_{2n} \\ \ldots & \ldots & \ldots & \ldots \\ C_{n1} & C_{n2} & \ldots & C_{nn} \end{bmatrix}$$

$$\sum_{k=1}^{n} c_{kj_k}$$

Вводимые понятия:

- Независимые нули
- Две прямоугольные матрицы называются эквивалентными (C ~ D), если Cij ~Dij для всех i,j .

Блок-схема алгоритма венгерского метода



Составим матрицу задания:

Операции / Оборудование	1	2	3	4
1	3	2	6	7
2	2	3	4	5
3	1	6	2	5
4	2	4	4	10

3	2	**6**	7
2	3	4	5
1	**6**	2	5
2	4	4	**10**

Предварительный этап

0	4	0	3
1	3	2	5
2	0	4	5
1	2	2	0

0*	4	0	3
0	2	1	4
2	0*	4	5
1	2	2	0*

Первая итерация. Первый этап

+	+		+	
0*	4	0	3	
0	2	1	4	
2	0*	4	5	
1	2	2	0*	

Вторая итерация. Первый этап

(+)	+		+	
0*	4	$0'$	3	+
0	2	1	4	
2	0*	4	5	
1	2	2	0*	

Первая итерация. Второй этап

(+)	+		+	
0*	4	$0'$	3	+
$0'$	2	1	4	
2	0*	4	5	
1	2	2	0*	

(+)	+		+	
0*	4	$0'$	3	+
$0'$	2	1	4	
2	0*	4	5	
1	2	2	0*	

Второй этап

0	4	0*	3	
0*	2	1	4	
2	0*	4	5	
1	2	2	0*	

Операции / Оборудование	1	2	3	4
1	3	2	6	7
2	2	3	4	5
3	1	6	2	5
4	2	4	4	10

Соответствующее значение целевой функции:

$$F = C_{21} + C_{32} + C_{13} + C_{44} = 2 + 6 + 6 + 10 = 24.$$