МИНИСТЕРСТВО ОБЩЕГО И ПРОФЕССИОНАЛЬНОГО
ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
САМАРСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

Кафедра информатики и вычислительной математики

ЛАБОРАТОРНЫЙ ПРАКТИКУМ ПО МЕТОДАМ ОПТИМИЗАЦИИ

Издательство "Самарский университет"
1998

Печатается по решению Редакционно-издательского совета Самарского государственного университета

В лабораторный практикум собраны основные методы, используемые при решении задач оптимизации. По каждому методу дается его краткое описание и приводится вычислительный алгоритм.

В лабораторных работах предусматривается практическое освоение изложенных в них методов с использованием средств вычислительной техники.

Лабораторный практикум предназначен студентам дневного и вечернего отделений вузов при изучении курса "Методы оптимизации".

Составители: канд. физ.-мат. наук, доц. *А.Г.Коваленко*,
канд. физ.-мат. наук , доц. *И.А.Власова*,
канд. физ.-мат. наук, доц. *А.Ф.Федечев*.

Рецензент канд. техн. наук, доц. *Н.Л.Меньших*

ЛАБОРАТОРНАЯ РАБОТА № 1

Определение промежутка локализации точки минимума

Большинство методов поиска минимума функции $\varphi(x)$ одной переменной основаны на построении последовательности стягивающихся отрезков (промежутков), содержащих точку минимума. Начальный промежуток, как правило, неизвестен. Алгоритм поиска начального промежутка для унимодальной функции основан на поиске точек $a<x<b$, в которых $\varphi(x)\le min(\varphi(a),\varphi(b))$. Тогда точка x^* минимума $\varphi(x)$ содержится в промежутке $[a,b]$. Этот поиск реализует следующий алгоритм.

Алгоритм поиска промежутка локализации минимума

Задаются произвольные значения x и Δx, где x - начальная точка, Δx - начальный шаг, m - коэффициент увеличения шага.

Начало 1 $Fx:=\varphi(x)$; $y:=x+\Delta x$; $Fy:=\varphi(y)$;
Если $Fy > Fx$, *то*
Начало 2. $z:=x$; $Fz:=Fx$; $x:=y$; $Fx:=Fy$;
$y:=z$; $Fy:=Fz$; $\Delta x:=-\Delta x$;
Конец 2;
Пока $Fy < Fx$ *выполнить*
Начало 3
$x:=y$; $Fx:=\varphi(x)$; $\Delta x:=m\Delta x$; $y:=x+\Delta x$;
$Fy:=\varphi(y)$
Конец 3;
Если $\Delta x>0$, *то*
Начало 4
$a=x-\Delta x/m$; $b=y$;
Конец 4
Иначе
Начало 5
$a=y$; $b=x-\Delta x/m$;
Конец 5
Конец 1.

В результате работы алгоритма получаем интервал $[a,b]$, содержащий точку оптимума и точку $x\in[a,b]$, в которой известно значение $Fx = \varphi(x)$.

Задание

Определить промежуток локализации минимума функции $\varphi(x)=(x-d)(x-l)$. В качестве начальных значений взять $x=0$, $\Delta x=1$, $m=2$.

Варианты заданий

№ варианта	1	2	3	4	5	6	7	8	9	10	11	12	13
d	5	7	8	8	1	3	3	1	2	-5	-7	-8	-8
l	6	8	9	10	4	5	4	3	4	-6	-8	-9	-10
№ варианта	14	15	16	17	18	19	20	21	22	23	24	25	
d	-1	-3	-3	-1	-2	0	0	2	-2	-4	3	4	
l	-4	-5	-4	-3	-4	-5	5	6	-6	0	6	9	

Результаты расчетов поместить в таблицу имеющую вид:

№ шага *k*	*x*	*Fx*	Δx	*y*	*Fy*

Под номером *k* здесь понимается порядковый номер входа в блок Начало 3 Конец 3. Значения *a* и *b* полученные в результате работы алгоритма, выписать ниже этой таблицы.

ЛАБОРАТОРНАЯ РАБОТА № 2

Метод симметричного поиска минимума унимодальной функции, заданной на отрезке. Метод симметричного поиска с восстановлением

Пусть функция $\varphi(x)$, $x \in [a,b] \subset R^1$ унимодальная, а $x,y \in (a,b)$- пробные точки, причем $x<y$. Если $\varphi(x) < \varphi(y)$, то на интервале $(y,b]$ точки минимума нет, т.е. можно положить $b:=y$. Аналогично, если $\varphi(x) > \varphi(y)$, то $a:=x$. В методах симметричного поиска с однократным вычислением значения функции на шаге точки x и y располагаются симметрично относительно середины отрезка $[a,b]$. Обозначим через $\Delta_0=b-a$, $\Delta_1=\lambda\Delta_0(\lambda\in(0.5;1))$- расстояние от пробной точки до дальнего конца отрезка, т.е. $x=b-\Delta_1$, $y=a+\Delta_1$, $\Delta_2=\Delta_0-\Delta_1$, Δ_2 - расстояние от пробной точки до ближнего конца отрезка; $\Delta_3=\Delta_1-\Delta_2$, - расстояние между пробными точками, т.е. $y-x=\Delta_3$. Из условия $x<y$ вытекает $\Delta_3>0$ или $\Delta_2<\Delta_1$. В процессе поиска минимума функции $\varphi(x)$ на каждой итерации происходит отбраковка интервала, не содержащего точки оптимума. При переходе к следующей итерации производится переобозначение $\Delta_0:=\Delta_1$, $\Delta_1:=\Delta_2$. Если величина λ осуществляет золотое сечение отрезка $[0, 1]$, т.е. $\lambda=\bar{\lambda}=(\sqrt{5}-1)/2$, то метод поиска точки оптимума называется методом золотого сечения. Если $\lambda=\lambda_k=F_{k-1}/F_k$, где F_k числа Фибоначчи ($F_0=0$, $F_1=1$, $F_k=F_{k-1}+F_{k-2}$, k > 2), то получается метод Фибоначчи.

Если $\varepsilon > 0$ есть требуемая точность приближенного значения точки оптимума, то алгоритм симметричного поиска заканчивает работу при выполнении условия $\Delta_1 \leq \varepsilon$.

Алгоритм симметричного поиска

Задаются значения a , b , l , ε .

M1:
Начало 1
$\Delta_0:=b-a$; $\Delta_1:=\lambda\times\Delta_0$; $x:=b-\Delta_1$;
$y:=a+\Delta_1$; $Fx:=\varphi(x)$; $Fy:=\varphi(y)$; $\Delta_2=\Delta_0-\Delta_1$;
Пока $\Delta_2<\Delta_1$ *и* $\Delta_1>\varepsilon$ *выполнить:*
Начало 2
Если $Fx<Fy$, *то*
Начало 3
$b:=y$; $y:=x$; $Fy:=Fx$; $x:=b-\Delta_2$, $Fx:=\varphi(x)$;
Конец 3
Иначе
Начало 4
$a:=x$; $x:=y$; $Fx:=Fy$; $y:=a+\Delta_2$; $Fy:=\varphi(y)$
Конец 4;
$\Delta_0:=\Delta_1$; $\Delta_1:=\Delta_2$, $\Delta_2=\Delta_0-\Delta_1$;
Конец 2;
Если $Fy<Fx$, *то*
Начало 5 $x:=y$; $Fx:=Fy$ *Конец 5;*
Конец 1.

Этот алгоритм при $\lambda \neq \lambda$ после ряда шагов начинает расходиться, т.е. величина $\Delta_0=(b-a)$ вместо убывания начинает принимать различные по знаку и возрастающие по величине значения. Признаком расходимости алгоритма является выполнение неравенства $\Delta_2 > \Delta_1$. В этом случае применяем алгоритм с восстановлением сходимости. После выполнения цикла, телом которого является блок *Начало 2 - Конец 2*, выполнить следующий оператор: *если* $\Delta_2 \geq \Delta_1$, *то иди M1;*

За решение задачи принимаем значения x и Fx, полученные в результате работы алгоритма.

Задание

Определить минимум функции $\varphi(x)=(x-d)(x-l)$. Значения e=0,05; $l=l_6$. Варианты заданий взять из лабораторной работы N1. Начальный интервал вычислить с помощью лабораторной работы № 1.

Дополнительное задание

(Выполняется вместо основного задания в случае, если при выполнении лабораторной работы не используются средства вычислительной техники)

Определить минимум функции $\varphi(x)=|x-d|$, $x \in [a,b]$, $\varepsilon=0,05$; $\lambda=\lambda_6$.

Варианты заданий взять из следующей таблицы:

№ варианта	d	a	b
1	0,5	0	1
2	2	-1	2,5
3	0,2	0	2
4	5	-1,2	7
5	0,1	-0,5	0,5
6	1	-3	3
7	0,1	-3	3,5
8	2	0	4
9	0,3	0	2
10	3	-1	5
11	0,25	-1	1
13	1,5	0	2
14	0,12	-1	1
15	1	0	3
16	1,2	-1,5	1,5
17	3	0,2	2,8
18	2,2	-1	3
19	1,3	-2	2
20	0,8	-1	2
21	0,6	-1,3	1,3
22	0,55	-1,5	1,5
23	0,15	-1	1
24	0,12	0	3
25	0,45	-1	2

Результаты вычислений представить в виде таблицы, где k - порядковый номер входа в блок *Начало 2 ... Конец 2.*

N шага k	a	b	Δ	Δ	Δ	x	Fx	y	Fy

ЛАБОРАТОРНАЯ РАБОТА № 3

Алгоритм несимметричного поиска минимума унимодальной функции, заданной на отрезке

Алгоритм устойчив и отличается от алгоритма симметричного поиска правилом формирования пробных точек x и y. За основу расстановки пробных точек берется равенство $y-x=\Delta$. Если на отрезке $[a,b]$ имеется пробная точка x, то $y = x + \Delta$, и наоборот, если имеется пробная y, то $x=y-\Delta$. На i-ом шаге алгоритма несимметричного поиска величина $\Delta = \lambda^{i+2}\Delta_0$, где $\Delta_0=b-a$.

При $\lambda = 2$ алгоритм несимметричного поиска эквивалентен алгоритму золотого сечения.

Алгоритм несимметричного поиска

Задаются значения a, b, λ, $\varepsilon>0$.

Начало 1.
- $\Delta:=b-a$; $\Delta:=\lambda\Delta$; $x:=b-\Delta$; $\Delta:=\lambda\Delta$; $y:=x+\Delta$;
- $Fx:=\varphi(x)$; $Fy:=\varphi(y)$
- *Пока* $b-a>\varepsilon$ *выполнить.*
- *Начало 2*
  - $\Delta:=\lambda\Delta$;
  - *Если* $Fx \le Fy$, *то*
  - *Начало 3*
    - $b:=y$; $y:=x$; $Fy:=Fx$; $x:=y-\Delta$; $Fx:=\varphi(x)$
  - *Конец 3.*
  - *Иначе*
  - *Начало 4*
    - $a:=x$; $x:=y$; $Fx:=Fy$; $y:=x+\Delta$; $Fd:=\varphi(y)$
  - *Конец 4.*
- *Конец 2*

Конец 1.

В результате работы алгоритма получаем точки x и y и соответствующие им значения функций Fx и Fy. За оптимальное решение принимают точку, которой соответствует минимальное значение функции.

Задание

Определить минимальное значение функции $\varphi(x)$. Вид функции, варианты, значения a, b, λ, ε взять из дополнительного задания лабораторной работы № 2.

Результаты вычислений представить в виде таблицы

№ шага k	Δ	x	Fx	y	Fy

где k - порядковый номер входа в блок *Начало 2 ... Конец 2.*

ЛАБОРАТОРНАЯ РАБОТА № 4

Методы оптимизации первого порядка. Методы деления отрезка пополам и касательных

Рассматриваются унимодальные дифференцируемые функции $\varphi(x)$, $x \in [a,b]$.

Пусть в точке $y \in [a,b]$ найдено значение $\varphi'(y)$. Если $\varphi'(y)<0$, то на отрезке $[a,y]$ минимума $\varphi(x)$ нет, и наоборот, если $\varphi'(y)>0$, то на отрезке $[y,b]$ минимума нет. В первом случае следует $a:=y$, во втором $b:=y$. Метод деления пополам и метод касательных различаются между собой способом выбора пробной точки y. В методе деления отрезка пополам y находится в середине отрезка $[a,b]$, в методе касательных y является абсциссой пересечения касательных, проведенных к $\varphi(x)$ в точках a и b. В начале работы алгоритма целесообразно произвести проверку знака производной в концевых точках. Это позволит в ряде случаев сразу выявить оптимальные значения $\varphi(x)$.

Алгоритм метода деления отрезка пополам

Задаются значения a, b, ε.

Начало 1. $y:=a$; $PRF = \varphi'(y)$;
  Если $PRF>0$ *то идти* M;
  $y:=b$; $PRF:=\varphi'(y)$;
  Если $PRF\leq 0$, *то идти* M;
  Пока $b - a>\varepsilon$ *выполнить:*
  Начало 2 $y:=(b+a)/2$;
    $PRF:=\varphi'(y)$;
    Если $PRF=0$, *то иди* M;
    Если $PRF<0$, *то иди* $a:=y$;
    Если $PRF>0$, *то* $b:=y$
  Конец 2:
  M: $Fy:=\varphi(y)$;
Конец 1.

В результате работы алгоритма получим значения y и Fy, являющиеся приближенными решением задачи.

Для того, чтобы получить алгоритм метода касательных, в предыдущем алгоритме следует оператор $y:=(b+a)/2$ следует заменить на

$$y:=((\varphi'(b)b-\varphi(b)) - (\varphi'(a)a-\varphi(a)))/(\varphi'(b)-\varphi'(a))$$

Задание

1. Определить минимум функции $\varphi(x)=(x-d)(x-l), \varepsilon=0,05$ методом деления отрезка пополам. Варианты заданий и начальный интервал взять из лабораторной работы № 2.

2. Сравнить число шагов, выполненных алгоритмом лабораторной работы № 2 с числом шагов, выполненных алгоритмом лабораторной работы № 4.

3. Сделать три шага алгоритмом метода касательных. Сравнить трудоемкость одного шага метода касательных и метода деления отрезка пополам.

Результаты вычислений записать в таблицу.

№ шага	a	b	y	PRF

ЛАБОРАТОРНАЯ РАБОТА № 5

Методы второго порядка. Методы Ньютона, ДСК, Пауэла

Пусть найден некоторый промежуток $[a,b]$ локализации точки минимума функции $\varphi(x)$. Предполагаем, что на $[a,b]$ $0 < m \le \varphi''(x) \le M$. Пусть также y -пробная точка этого отрезка. В точке y разложим $\varphi(x)$ в ряд Тейлора и в качестве квадратичной аппроксимации возьмем 3 члена этого ряда

$$\varphi(x) \cong \varphi(y) + \varphi'(y)(x - y) + 1/2\times\varphi''(y)(x - y)^2.$$

Минимум этого полинома достигается в точке

$$x = y - \varphi'(y)/\varphi''(y)$$

Ее и принимаем в качестве нового приближения к экстремуму x^* функции $\varphi(x)$. Для получения более точного решения полагают $y := x$ и определяют новое приближение. В этом и состоит метод Ньютона (метод квадратичной аппроксимации). Заметим, что если отрезок $[a,b]$ локализует точку максимума, то значение $x = y - \varphi'(y)/\varphi''(y)$ будет приближенным значением максимума функции $\varphi(x)$. Этот метод быстро сходится, однако существуют случаи, когда метод расходится. Признаком расходимости является выполнение условия $x \notin [a,b]$

Алгоритм метода Ньютона

Определить начальный промежуток $[a,b]$, локализующий точку минимума и пробную точку y алгоритмом лабораторной работы № 1.

Выполнить $F_y := \varphi(y)$;

M:

Начало 1

$x = y - \varphi'(y) / \varphi''(y)$,

Если $x \notin [a,b]$, *то идти на M1;*

$F_x := \varphi(x)$;

Если $|F_x - F_y| > \varepsilon$, *то*

Начало 2

$y := x$, $F_y := F_x$ *идти на M;*

Конец 2

M1:

Конец 1

Для вычисления производных функций применяют формулы численного дифференцирования. Так если $y \in [a,b]$, то

$$\varphi'(y) = (\varphi(b) - \varphi(a))/(b-a);$$

$$\varphi''(y) = ((b-y)\varphi(a) - (b-a)\varphi(y) + (y-a)\varphi(b))/((y-a)(b-y)(b-a)).$$

В этом случае новое приближение будет иметь вид

$$x = y - ((y-a)(b-y)[\varphi(b) - \varphi(a)])/((b-y)\varphi(a) - (b-a)\varphi(y) + (y-a)\varphi(b)).$$

Заметим ,что для рассматриваемых функций $\varphi(x)$, если $x < y$ и $\varphi(x) < \varphi(y)$, то интервал $(y, b]$ не содержит точки оптимума, то есть можно $b := y$. Аналогично для $x < y$ и $\varphi(x) \geq \varphi(y)$, можно $a := x$. Авторами этого подхода являются Девис, Свен Кемпи (ДСК).

Алгоритм метода ДСК

Определить начальный промежуток $[a,b]$ локализации точки минимума и пробную точку y алгоритмом лабораторной работы № 1.

Выполнить

$Fa := \varphi(a)$; $Fb := \varphi(b)$; $Fy := \varphi(y)$;

$x := y - ((b-y)(y-a)(Fb - Fa))/((b-y)Fa - (b-a)Fy + (y-a)Fb))$ (*)

$Fx := \varphi(x)$;

Пока $(|x-y| > \varepsilon) \wedge (x \in [a,b])$ *выполнить*

Начало 1

Если $x > y$ *то*

Начало 2.

$z := y$; $Fz := Fy$; $y := x$; $Fy := Fx$;

$x := z$, $Fx := Fz$

Конец 2;

Если $Fx \leq Fy$ *то*

<u>Начало 3</u>
b:=y; Fb :=Fy ; y:=x; Fy :=Fx
<u>Конец 3</u>
<u>Иначе</u>
<u>Начало 4</u>
a:=x; Fa :=Fx ;
<u>Конец4</u>
x:= y - ((b-y)(y-a)(Fb - Fa))/((b-y)Fa -(b-a)Fy +(y-a)Fb)) (*)
Fx:=φ (x);
<u>Конец 1.</u>

Если в этом алгоритме окончание произошло из-за условия $x \notin [a,b]$, то алгоритм расходится, иначе x принимается за решение задачи.

Опишем кратко метод Пауэла. Пусть отрезок $[a,b]$ содержит оптимальную точку x^* и некоторую пробную точку $y \in [a,b]$. Значения $\varphi(x)$ в точках a, b, y обозначим F_a, F_b, F_y. Интерполяционный многочлен Лагранжа $L(x)$ с узлами интерполяции в точках a, y, b будет иметь вид

$$L(x) = (((x-y)(x-b))/((a-y)(a-b)))Fa + (((x-a)(x-b))/((y-a)(y-b)))Fy + \\ + (((x-a)(x-y))/((b-a)(b-y)))Fb$$

Из условия $L'(x) = 0$ находим точку x:

$$x=1/2\times((b^2-y^2)Fa-(b^2-a^2)Fy+(y^2-a^2)Fb)/((b-y)Fa-(b-a)Fy+(y-a)Fb)), \quad (**)$$

которую принимаем за приближенное решение поставленной задачи. Формула (**) является расчетным соотношением метода Пауэла нахождения приближенного значения x^*

Для построения алгоритма по методу Пауэла достаточно в алгоритме метода ДСК заменить операторы помеченные (*) формулой (**).

Задание

1. Для функции $\varphi(x)=(x-d)(x-l)^3$ осуществить поиск начального промежутка локализации точки минимума.

2. Для найденного промежутка и $\varepsilon = 0{,}5$ выполнить поиск минимума функции $\varphi(x)$ методом Пауэла.

3. Используя получившийся промежуток $[a,b]$, продолжить поиск минимума функции $\varphi(x)$ методом Ньютона для $\varepsilon = 0{,}025$.

Варианты значений d и l взять в лабораторной работе № 1. Результаты вычислений представить в виде таблицы:

№ шага	a	Fa	b	Fb	x	F x	y	F y

Поиск глобального экстремума в многоэкстремальной задаче

Рассмотрим задачу определения глобального минимума функции $\varphi(x)$ на ограниченном промежутке $[a,b]$. В основе метода решения лежит метод ветвей и границ, отбраковывающий интервалы, о которых заведомо известно, что они не содержат глобального экстремума. Для применимости этого метода предполагается, что функция $\varphi(x)$ удовлетворяет условию Липшица с константой L:

$$|\varphi(x)-\varphi(x')| \le L\,|x-x'|.$$

В случае дифференцируемости $\varphi(x)$:

$$|\varphi'(x)| \le L$$

Для работы метода требуется знание величины $\alpha \ge L$. В качестве нижней оценки u_i значения функционала на интервале Δi принимается ордината точки пересечения прямых, проходящих через точки $(x_i, \varphi(x_i))$, $(x_j, \varphi(x_j))$ с угловыми коэффициентами, равными соответственно α и $(-\alpha)$, где x_i, x_j - соответственно левый и правый конец интервала Δi. Абсцисса x_i точки пересечения принимается за точку разбиения Δi. Разбиению подвергается интервал Δk, для которого $u_k = v = \min u_i$, где минимум берется по всем интервалам Δi, сформированным к данному шагу. Рекордом считается точка z', в которой значение $W=\varphi(z')$ наименьшее среди всех значений $\varphi(z_i)$, вычисленных к данному шагу. Отбраковке подлежат интервалы Δi, у которых оценки $u_i \ge W$. На начальном шаге $\Delta_0 = [x_0, x_1]$, где $x_0 = a$, $x_1 = b$. Алгоритм прекращает работу в том случае, когда $W - v < \varepsilon$, где ε - заданная допустимая погрешность приближенного решения, значение z и будет искомым приближенным решением.

Алгоритм глобального поиска

Начало
  $x_0 := a$; $x_1 := b$; $Fx_0 := \varphi(x_0)$; $Fx_1 := \varphi(x_1)$;
  Если $Fx_0 \le Fx_1$ *то*
  Начало 1 $W := Fx_0$; $x' := x_0$, *Конец 1*
  Иначе
  Начало 2 $W := Fx_1$; $x' := x_1$, *Конец 2;*
  $u_0 := 1/2 \times [\alpha(x_0 - x_1) + (Fx_0 + Fx_1)]$;
  $\xi_0 := (x_0 + x_1)/2 + (Fx_0 - Fx_1)/2\alpha$;
  $v := u_0$; $k := 1$;
  Пока $W - v < \varepsilon$ *выполнить*
  Начало 3 найти p, *для которого* $u_p = v$;
    $x := \xi_p$; $Fx := \varphi(x)$;
    Если $Fx > 2W - v$, *то*
      Начало 4 удалить пару (ξ_p, u_p)
        (т.е. исключаем интервал Δ_p *); иои М*
      Конси 4:
      Если $Fx < W$ *то*
      Начало 5 $W := Fx$; $x' := x$; *исключить*

все пары (ξ_i, u_i), *для которых* $u_i \geq W$

<u>*Конец 5.*</u>

$u_{k+1} := (Fx + v)/2$,

$\xi_{k+1} := \xi_p + (Fx - v)/(2\alpha)$;

$\xi_p := 2\xi_p - \xi_{k+1}$, $u_p := u_{k+1}$;

$k := k+1$;

M: $v := \min u_i$; *где i пробегает все индексы пар* (x_i, u_i), *сформированных и неотбракованных к данному шагу.*

<u>*Конец 3;*</u>

<u>*Конец*</u>

Значения x' и $W = \varphi(x')$, полученные в результате работы алгоритма, принимаем за приближенное решение задачи.

Задание

Минимизировать функцию $\varphi(x)$ вида:

$$\varphi(x) = \begin{cases} |x-d_1| - d_1, & \text{если } 0 \leq x < 2d_1 \\ |x-2d_1-d_2| - d_2, & \text{если } 2d_1 \leq x < 2d_1+2d_2 \\ |x-2d_1-2d_2-d_3| - d_3, & \text{если } 2d_1+2d_2 \leq x \leq 2d_1+2d_2+2d_3 \end{cases}$$

Значение $\alpha=2$, $\varepsilon=0{,}05$. Варианты значений d_1, d_2, d_3 взять из таблицы.

Таблица значений d_1, d_2, d_2

№ варианта	d_1	d_2	d_3
1	1	5	1
2	1	2	1
3	1	2	2
4	2	2	1
5	2	4	2
6	2	1	2
7	1	2	3
8	2	1	3
9	1	3	2
10	2	3	2
11	3	2	1
12	2	3	1
13	3	1	3
14	1	3	3
15	3	3	1
16	2	3	4
17	3	2	3
18	3	3	2
19	2	7	3
20	2	3	3
21	1	4	1
22	4	1	1
23	1	1	4
24	1	4	4
25	4	1	4

ЛАБОРАТОРНАЯ РАБОТА № 7

Циклический покоординатный поиск в задачах безусловной оптимизации

Рассматривается задача $min\ \varphi(x)$ при $x \in E_n$. Идея метода простого покоординатного поиска заключается в следующем. Берется начальная точка x_0 и вектор λ величин шагов по каждому направлению, и полагаем $x^*=x_0$. Из точки x^* пытаемся перейти в точку x' изменением только первой координаты на величину $\lambda[1]$. Если эта попытка удачная, т.е. $\varphi(x')<\varphi(x^*)$, то увеличиваем величину шага $\lambda[1]$, $\lambda[1]:=\lambda[1]*\alpha$ и переходим в точку x', полагая $x^*=x'$. Если попытка неудачная, т.е. $\varphi(x') > \varphi(x^*)$, то изменяем величину и направление шага $\lambda[1]$, полагая $\lambda[1]:=\lambda[1]*\beta$, где $-1<\beta< 0$. Затем переходим к следующему координатному направлению. По окончании спуска по всем n координатам оцениваем $|x_0 - x^*|$ и $|\lambda|$. Если эти величины меньше заданного числа ε, то процесс прекращаем, иначе полагаем $x_0 = x^*$ и процесс начинаем заново. Заметим, что возможен такой случай, что $x_0 = x^*$, а $|\lambda| > \varepsilon$. Это может произойти тогда, когда величины $\lambda[i]$, $i=1,\dots,n$ завышены и все шаги неудачные.

Алгоритм простого покоординатного поиска

<u>*Начало 1*</u>   <u>*Массивы*</u> $s, x_0, x^*, \lambda[1:n]$;
  $Fx_0:=\varphi(x_0)$; $x^*:=x_0$, $Fx^* := Fx_0$;
  M: <u>*Для*</u> *k:=1* <u>*до*</u> *n* <u>*выполнить*</u>
  <u>*Начало2*</u>
    $s:= e_k$;(*здесь и далее* e_k - *k-ый орт*) $x':=x^*+\lambda[k]*s$; $Fx':=\varphi(x')$;
      Если $Fx' < Fx^*$ *то*
      <u>*Начало3*</u> $x^*:=x'$;
        $Fx^*:=Fx'$; $\lambda[k]:=\lambda[k]*\alpha$
      <u>*Конец 3*</u>
      <u>*Иначе*</u> $\lambda[k]:=\lambda[k]*\beta$
      <u>*Конец 2;*</u>
      <u>*Если*</u> $(|x_0-x^*| > \varepsilon) \vee (|\lambda| > \varepsilon)$ <u>*то*</u>
      <u>*Начало 4*</u> $x_0 = x^*$; $Fx_0:= Fx^*$; <u>*иди*</u> *M* ; <u>*Конец 4 ;*</u>
<u>*Конец 1.*</u>

Алгоритм исчерпывающего покоординатного поиска

В отличие от простого покоординатного поиска, в исчерпывающем покоординатном поиске при удачном переходе из x^* в x' предлагается дальнейший спуск по этому направлению до тех пор, пока не получим неудачный переход. В этом случае осуществляем переход к следующему направлению спуска. Отличие алгоритма исчерпывающего покоординатного поиска от предыдущего алгоритма заключается лишь в блоке *Начало 2 ... Конец 2.*

Этот блок должен принять следующий вид:

Начало 2.

$s := e_k$; $x' := x^* + \lambda[k]*s$; $Fx' := \varphi(x')$;

Пока $Fx' < Fx^*$ *выполнить*

Начало 3

$x^* := x'$; $Fx^* := Fx'$; $\lambda[k] := \lambda[k]*\alpha$;

$x' := x^* + \lambda[k]*s$; $Fx' := \varphi(x')$;

Конец 3:

$\lambda[k] := \lambda[k]*\beta$

Конец 2:

Алгоритм экстремального покоординатного поиска

Этот алгоритм заключается в последовательном отыскании минимума по всем координатным направлениям пространства E_n. По каждому из этих направлений функция $\varphi(x)$ является функцией одной переменной, и методы минимизации известны. Если текущее значение переменной x равно x^*, то точка x' находится из условия:

$$\varphi(x') = \min \varphi(x^* + \lambda[k]*s), \text{ где } s := e_k$$

В этом случае блок *Начало 2...Конец2* будет иметь вид:

Начало 2

$s := e_k$.

$\lambda'[k] := \operatorname{argmin} \varphi(x^* + \lambda * s)$;

$x' := x^* + \lambda'[k]*s$; $x^* = x'$

Конец 2:

При этом условие $|\lambda| > \varepsilon$ можно не проверять, так как оно эквивалентно условию $|x_0 - x^*| > \varepsilon$.

Задание

Определить минимум функции

$$\varphi(x) = \frac{(\xi_1 - d_1)^2}{4} + \frac{(\xi_2 - d_2)^2}{4} + d_3$$

За начальное приближение взять точку $x_0 = (0,0)$, $\lambda = (2,2)$.

Варианты значений d_1, d_2, d_3 представлены в следующей таблице.

Таблица значений d_1, d_2, d_3

№ варианта	d_1	d_2	d_3
1	1	1	1
2	3	1	2
3	1	3	1
4	1	1	2
5	5	2	1
6	2	1	1
7	1	2,5	2
8	1	2,5	3
9	2,5	1	3
10	3	2	1
11	1	1	3
12	3	3	1
13	3	2	2
14	3	4	3
15	3	3	2
16	2,5	3,5	3
17	2,7	1	4
18	4	1	2
19	1	3	4
20	0,5	4	2
21	1	3	4
22	3	4	1
23	4	1	4
24	1	4	1
25	4	4	1

Для $\varepsilon =0,5$ выполнить минимизацию функции $\varphi(x)$ алгоритмом простого покоординатного поиска для $\varepsilon=0,2$. Значение $\alpha =2$, $\beta= -0,5$.

Закончить работу алгоритмом экстремального покоординатного поиска для $\varepsilon =0,05$. Таблицу представления данных разработать самостоятельно.

ЛАБОРАТОРНАЯ РАБОТА № 8

Метод деформированного многогранника (Нелдера-Мида)

Рассматривается задача $\min \varphi(x)$, $x \in E_n$. В пространстве E_n берется многогранник, вершины которого обозначим через $x[i]$, $i=1, ..,n+1$. Каждая переменная $x[i]$ представляет собой *n*- мерный вектор, $x[i] \in E_n$. В каждой из этих точек вычисляется значение $\varphi(x[i])$. Через *h* обозначим точку с наибольшим значением $\varphi(x[i])$,через *l*- с наименьшим. В данном методе точка, с наибольшим значением $\varphi(x)$ заменяется на точку с меньшим значением, отличную от точек многогранника. Через $x[n+2]$ обозначим точку, являющуюся "центром тяжести" вершин, отличных от $x[h]$. Точка $x[n+3]=x[n+2]+\alpha\ (x[n+2]-x[h])$ является отражением $x[h]$ относительно $x[n+2]$, где α–коэффициент отражения. Если $\varphi(x[n+3])<\varphi(x[l])$, то определяем точку растяжения $x[n+4]=x[n+2]+\gamma\ (x[n+3]-x[n+2])$, где γ коэффициент отражения. Если $\varphi(x[n+4])< \varphi(x[l])$, то $x[h]$ заменяем на $x[n+4]$, иначе на $x[n+3]$. В том случае, когда $\varphi(x[l])\le\varphi(x[n+3])<\varphi(x[h])$, то точку $x[h]$ заменяем на точку сжатия $x[n+5]=x[n+2]+\ \beta\ (x[h]-x[n+2])$; где β -коэффициент сжатия. Если $\varphi(x[n+3])\ \ge \varphi(x[h])$, то осуществляем сжатие (редукцию) многогранника. Все точки $x[k]$, $k=1,...,n+1$ полагаем равными $x[l]+\delta\ (x[k]-x[l])$. Алгоритм заканчивает работу, если

$$\sqrt{\left(\frac{1}{n+1}\sum_{k=1}^{n+1}(\varphi(x[k])-\varphi(x[n+2]))^2\right)} \le \varepsilon$$

Алгоритм деформированного многогранника

Задаются значения $x[1:n+1],\alpha\ ,\ \beta\ ,\ \gamma\ ,\delta$ *- коэффициенты соответственно отражения, сжатия, растяжения, редукции,* $\alpha > 0\ ,0 < \beta < 1,\ \gamma>1\ 0 < \delta <1$.

M:

$h:=\arg\max\varphi\,(x[i]);\ i=1,...,n+1;$

$l:=\arg\min\varphi\,(x[i]);\ i=1,...,n+1;$

$x[n+3]:=x[n+2]+\ \alpha\ (\ x[n+2]-x[h]);$

$x[n+2]:=\frac{1}{n}\left(\sum_{i=1}^{n+1}(x[i]-x[h])\right)$

<u>Если</u> $\varphi\,(x[n+3])\le\varphi\,(x[l])$ *<u>то</u>*

<u>Начало 1</u>

$x[n+4]:=x[n+2]+\ \gamma\ (\ x[n+3]-x[n+2]);$

<u>Если</u> $\varphi\,(x[n+4])<\varphi\,(x[l])$ *<u>то</u>*

x[h]:=x[n+4]

Иначе

x[h]:=x[n+3];

иди М

Конец 1:

Если $\varphi(x[n+3]) < \varphi(x[h])$ *, то*

Начало 2.

x[h]:=x[n+2]+ β *(x[h]-x[n+2]);*

Иди М

Конец 2:

Для k:=1 до n+1 выполнить

x[k]:=x[l]+ δ *(x[k]-x[l]);*

Если $\sqrt{\left(\frac{1}{n+1}\sum_{k=1}^{n+1}(\varphi(x[k]) - \varphi(x[n+2]))^2\right)} > \varepsilon$ *, то*

Иди М

иначе

алгоритм заканчивает работу;

Задание

Определить минимум функции

$$\varphi(x) = \frac{(\xi_1 - d_1)^2}{4} + \frac{(\xi_2 - d_2)^2}{9} + d_3$$

Варианты значений d_1, d_2, d_2 представлены в таблице лабораторной работы N8. За начальное приближение взять треугольник с вершинами: *x[1]=(0,0), x[2]=(1,0), x[3]=(0,1).*

Коэффициенты принять равными:

$$\alpha = 1,\ \beta = 0{,}5,\ \gamma = 2,\ \delta = 0{,}5,\ \varepsilon = 0{,}5.$$

Результаты вычислений представить в виде таблицы:

x[1]	φ(x[1])	x[2]	φ(x[2])	x[3]	φ(x[3])	x[4]	φ(x[4])	x[5]	φ(x[5])	x[6]	φ(x[6])	x[7]	φ(x[7])	h	l

ЛАБОРАТОРНАЯ РАБОТА № 9

Метод наискорейшего спуска

Рассматривается метод минимизации функции $\varphi(x)$, $x \in E_n$, из заданного начального приближения x_0 по итерационной формуле

$$x_{k+1} = x_k + \lambda_k S_k, \; i=0,1,\ldots,$$

где x_k - приближение к решению задачи на i-ой итерации S_k - направление убывания функции $\varphi(x)$, λ_k - длина шага вдоль направления S_k.

Известно, что направление $S - \varphi'(x) = \left(\frac{\partial\varphi}{\partial x_1}, \frac{\partial\varphi}{\partial x_2}, \ldots, \frac{\partial\varphi}{\partial x_n}\right)^T$, вычисленное в точке x, дает направление наискорейшего возрастания функции, направление $S = -\varphi'(x)$ дает направление наискорейшего убывания функции. В методе наискорейшего спуска на каждом шаге $S_k = -\varphi_k(x)$. Направление $S_k \neq 0$ гарантирует, что среди точек $x = x_k + \lambda S_k$, $\lambda > 0$ существуют точки, для которых $\varphi(x) < \varphi(x_k)$. В методе наискорейшего спуска величина λ определяется из условия: $\lambda_k = \arg\min \varphi(x_k + \lambda S_k)$ $\lambda > 0$. Для отыскания λ_k применяются алгоритмы отыскания начального промежутка локализации точки минимума с последующим применением методов одномерной оптимизации.

Критерии останова процесса спуска могут быть различными. Наиболее распространенными критериями являются условия:

$$|x_{k+1} - x_k| \le \varepsilon, \quad |\varphi(x_{k+1}) - \varphi(x_k)| < \varepsilon, \quad |\varphi'(x_k)| < \varepsilon$$

В предлагаемом далее алгоритме используется последний критерий

Алгоритм метода наискорейшего спуска

<u>*Начало 1*</u>
  $x := x_0$; $PRFx := \varphi'(x)$;
  <u>*Пока*</u> $\|\varphi'(x)\| > \varepsilon$ <u>*выполнить*</u>
  <u>*Начало 2.*</u>
    $s := -\varphi'(x)$;
    $\lambda := \arg\min\limits_{\lambda \ge 0} \varphi(x + \lambda s)$.
    $x := x + \lambda s$;
  <u>*Конец 2;*</u>
<u>*Конец 1.*</u>

Задание

Определить минимум функции

$$\varphi(x) = \frac{(\xi_1 - d_1)^2}{4} + \frac{(\xi_2 - d_2)^2}{9} + d_3$$

Варианты значений d_1, d_2, d_3 взять из лабораторной работы № 8. За начальное приближение выбрать одну из вершин многогранника, полученного на последней итерации лабораторной работы № 9. Значение ε принять равным 0.01.

Замечание. Так как методы одномерной оптимизации в предыдущих лабораторных работах достаточно трудоемки, а минимизируемая функция $\varphi(x)$ достаточно проста, то величину λ целесообразно определять из уравнения:

$$\partial / \partial\lambda(\varphi(x + \lambda s)) = 0.$$

ЛАБОРАТОРНАЯ РАБОТА № 10

Методы второго порядка. Методы Ньютона и Ньютона-Рафсона

В методе Ньютона минимизируемая функция $\varphi(x)$ в точке x_k, $i=0,1,\ldots$ аппроксимируется полиномом

$$P(x) = \varphi(x_k) + \langle \varphi'(x_k), x - x_k \rangle + 1/2 \langle x - x_k, \varphi''(x_k)(x - x_k) \rangle,$$

где $\varphi''(x_k) = \left(\partial^2 \varphi / \partial\xi_i \partial\xi_j \right)$ - матрица, составленная из производных второго порядка, $i = 1,n$, $j = 1,n$. Минимизация функции $\varphi(x)$ заменяется на минимизацию $P(x)$:

$$P'(x) = 0, \text{ т.е. } \varphi'(x_k) + \varphi''(x_k)(x - x_k) = 0,$$

отсюда $x = x_k - (\varphi''(x_k))^{-1} \varphi'(x_k)$.

Точка x принимается за очередное приближение x_{k+1} минимума $\varphi(x)$.

Алгоритм метода Ньютона

<u>*Начало 1.*</u>
  $x := x_0$; $PRFx := \varphi'(x)$; $H := \varphi''(x)$;
  <u>*Пока*</u> $/PRFx/ > \varepsilon$ <u>*выполнить*</u>
  <u>*Начало 2.*</u>
    $x := x - H^{-1} * PRFx$;
    $PRFx := \varphi'(x)$;
    $H := \varphi''(x)$;
  <u>*Конец 2.*</u>
<u>*Конец 1.*</u>

Нетрудно видеть, что если $\varphi'(x)$ - квадратичная функция, то алгоритм получит значения минимума за одну итерацию. Критерием сходимости этого метода является положительная определенность матрицы $\varphi''(x)$.

Для восстановления сходимости метода Ньютона и ускорения его сходимости может быть предложена следующая его модификация (метод Ньютона-Рафсона). В нем итерация метода осуществляется по формуле:

$$x_{k+1} = x_k + \lambda_k S_k,$$

где $S_k = (\varphi''(x_k))^{-1}\varphi'(x_k)$, величина λ_k определяется из условия:

$$\varphi(x_k + \lambda_k S_k) = \min_{\lambda} \varphi(x_k + \lambda S_k)$$

Алгоритм метода Ньютона-Рафсона

<u>Начало 1</u>

$x := x_0$; $PRFx := \varphi'(x)$; $H := \varphi''(x)$;

Пока $|PRFx| > \varepsilon$ <u>выполнить</u>

<u>Начало 2</u>

$s := -H^{-1} * PRFx$; $\lambda := \underset{\lambda}{argmin}\,\varphi(x + \lambda s)$;

$x := x + \lambda s$; $H := \varphi''(x)$;

<u>Конец 2</u>

<u>Конец 1.</u>

Задание

Определить минимум функции

$$\varphi(x) = \left(1 + e^{[illegible]}\right)\left(\frac{(\xi_1 - d_1)^2}{2} + \frac{(\xi_2 - d_2)^2}{9}\right) + d_3$$

методом Ньютона $\varepsilon = 0,001$.

Варианты заданий d_1, d_2, d_3 взять из лабораторной работы № 8

В качестве начального приближения взять приближенное значение минимума из лабораторной работы № 9

ЛАБОРАТОРНАЯ РАБОТА № 11

Метод сопряженных направлений

Пусть H - симметричная матрица $n \times n$. Систему векторов $s_0, s_1, \ldots, s_{n-1}$ будем называть H - сопряженной (ортогональной), если для любых

$i,j = 0,\ldots,n-1$, $i \neq j$, $\langle s_i, Hs_j \rangle = 0$ и $\langle s_i, Hs_i \rangle \neq 0$.

Рассмотрим задачу минимизации квадратичной функции $\varphi(x) = a + \langle b, x \rangle + 1/2 \langle x, Hx \rangle$, где H- положительно определенная симметричная матрица. Итерационный процесс спуска будем осуществлять с помощью соотношения

$$x_{k+1} = x_k + \lambda_k s_k, \; k = 0,...,n,$$

где s_k - вектор из системы H- сопряженных векторов,

$$\lambda_k = \arg\min_{\lambda} \varphi(x_k + \lambda s_k)$$

и его значение равно $\lambda_k = -\frac{<\varphi'(x_k), s_k>}{<s_k, Hs_k>}$

Известно, что за n итераций такой спуск дает минимум квадратичной функции с положительно определенной матрицей.

Алгоритм метода сопряженных направлений

<u>*Начало 1*</u>

x:=x₀; PRFx:=φ'(x);

<u>для</u> k:=0 <u>до</u> n-1 <u>выполнить</u>

<u>*Начало 2*</u>

$\lambda = \lambda_k = -\frac{<\varphi'(x_k), s_k>}{<s_k, Hs_k>}$, $x := x + \lambda s_k$;

<u>*Конец 2*</u>

<u>*Конец 1.*</u>

Задание

Определить минимум функции

$$\varphi(x) = a + <b, x> + 1/2 <x, Hx>,$$

где $b = (b_1, b_2)^T$, $H = \begin{pmatrix} d_1 & 1 \\ 1 & d_2 \end{pmatrix}$, $x = (\xi_1, \xi_2)^T$

Значения a, b_1, b_2, d_1, d_2 взять из следующей таблицы вариантов:

№ варианта	a	b_1	b_2	d_1	d_2
1	1	1	1	3	4
2	1	1	2	2	3
3	1	1	3	2	2,5
4	1	2	3	2	1
5	1	2	1	1,5	3
6	1	2	2	2	4
7	1	2	3	3	1
8	1	3	1	3	2
9	1	3	2	3	3
10	1	3	3	1,5	2
11	2	1	1	1	4
12	2	1	2	2	[illegible]

13	2	1	3	2	4
14	2	2	1	1	3
15	2	2	2	1	2
16	2	2	3	2	2
17	2	3	1	3	1,5
18	2	3	2	1,5	4
19	2	3	3	3	2,5
20	3	1	1	1,5	1,5
21	3	1	2	1,5	3
22	3	1	3	2	4
23	3	2	1	2,5	4
24	3	2	2	2,5	1
25	3	2	3	3	3,5

В качестве x_0 взять точку $(0,0)$, направление $s_0 = \varphi'(x_0)$. Направление s_1 определить из условия $\langle s_0, Hs_1\rangle = 0$.

ЛАБОРАТОРНАЯ РАБОТА № 12

Отыскание базисных решений и опорных планов задачи линейного программирования

Рассмотрим задачу линейного программирования (ЛП) в виде:

$\langle c,x\rangle \longrightarrow max$

при ограничениях $\langle a_i, x\rangle \le b_i,\ i \in I_1$

$\langle a_i, x\rangle \le b_i,\ i \in I_2$

$\langle a_i, x\rangle \le b_i,\ i \in I_3$

$x_i \ge 0,\ i\in I_1;\ x_i \le 0,\ i\in I_2,\ x_i$ любое действительное число (л.д.ч.) для остальных $i \in I_3$.

Для $i \in I_1$ введем переменную $y_i = b_i$ $\langle a_i, x\rangle$, для $i \in I_2$ $y_i = \langle a_i, x\rangle$ b_i.

Тогда получим задачу:

$$\langle c,x\rangle + \sum 0 * y_i \longrightarrow max.$$

$$\langle a_i, x\rangle + y_i = b_i,\ i \in I_1$$

$$\langle a_i, x\rangle - y_i = b_i,\ i \in I_2$$

$$\langle a_i, x\rangle = b_i,\ i \in I_3$$

$$x_i \ge 0,\ i\in I_1;\ x_i \le 0,\ i\in I_2,\ x_i \text{ - л.д.ч.},\ i \in I_3.$$

Для $i\in I_1$ произведем замену переменных $x_i' = -x_i$, для $i \in I_3$ $x_i = x_i' - x_i''$. Все получившиеся переменные переобозначим через $x_1, x_2,\ldots,x_n$. Предыдущая задача будет иметь вид:

$$<c,x> \to max.$$
$$Ax=b$$
$$x \geq 0_n,$$

где $x\in E_n$, $b\in E_m$, $c\in E_m$, $0_n = (0,0,\ldots,0)^T$, A - матрица $(m \times n)$.

Задача такого вида называется канонической задачей ЛП, а описанная процедура называется приведением задачи ЛП к каноническому виду. В дальнейшем будем рассматривать задачи только такого вида.

Перепишем задачу в виде:

$$c_1 x_1 + c_2 x_2 + \ldots + c_n x_n \to max$$
$$a_1 x_1 + a_2 x_2 + \ldots + a_n x_n = b$$
$$x_i \geq 0,\ i=1,\ldots,n.$$

Здесь a_i - вектор-столбец матрицы A.

Выберем из a_i, $i=1,\ldots,n$ m линейно-независимых векторов $a_{i1},a_{i2},\ldots a_{im}$. Эти векторы будем называть базисными, также будем называть базисными переменные $x_B = (x_{i1},x_{i2},\ldots x_{im})^T$ и квадратную матрицу $B=(a_{i1},a_{i2},\ldots a_{im})$.

Задача *ЛП* может быть записана в виде:

$$<c_B, x_B> + <c',x'> \to max.$$
$$B x_B + A'x' = b$$
$$x_B \geq 0_m,\ x' \geq 0_{n-m},$$

где x', c', A' - внебазисные части.

Решение вида $\begin{pmatrix} x_B \\ x' \end{pmatrix} = \begin{pmatrix} B^{-1}b \\ 0_{n-m} \end{pmatrix}$ называются базисными, если $B^{-1}b \geq 0_m$, то опорным планом, если $B^{-1}\ b > 0_m$, то опорный план невырожденный. Наиболее простым является случай, когда $B=B^{-1}=E$. Рассмотрим способы приведения к нему.

Ограничения задачи $Ax=b$ запишем в виде таблицы (симплекс-таблицы):

x_B	b	x_1	x_2	$x3$	…	x_n
x_{i1}	b_1					
x_{i2}	b_2	a_1	a_2	a_3	…	a_n
x_{im}	b_m					

Будем говорить, что задача *ЛП* приведена к базису $x_{i1},x_{i2},\ldots x_{im}$, если на пересечении каждой строки x_{ik} со столбцом x_{ik} $a_{ik\,ik} = 1$, все остальные элементы столбца i_k равны 0. Легко видеть, что в такой задаче после соответствующей перестановки столбцов $B=E$. Для приведения задачи к базису надо для каждой строки x_{ik} симплекс-таблицы выполнить преобразования Гаусса-Жордано по формулам:

$a_{kj} = a_{kj} / (a_{k i_k})$ для k-ой строки

$a_{ij} = a_{ij} - (a_{kj} a_{i i_k})/(a_{k i_k})$

В результате этих преобразований в новой симплекс-таблице на пересечение строки k со столбцом i_k будет стоять 1, все остальные элементы будут равны 0. Заметим, что эти преобразования таблицы, перенесенные к системе $Ax=b$, дают ее эквивалентные преобразования.

Задание

1. Привести задачу к каноническому виду.
2. Выбрать не менее трех базисов.
3. Привести задачу к выбранным базисам:
    а) определить базисное решение;
    б) определить, является ли оно планом и является ли этот план вырожденным;
    в) вычислить значение функционала для опорного плана.
4. Среди всех опорных планов выбрать экстремальные (*min* и *max*).

Пример:

$$x_1 + x_2 \dashrightarrow max$$

$$2 x_1 + x_2 \leq 3$$
$$x_1 + 2 x_2 \geq 3$$
$$x_1 \text{ -л.о.ч.}$$
$$x_2 \geq 0$$

Приведем к каноническому виду:

$$x_1 + x_2 \dashrightarrow max$$
$$2 x_1^+ \;\; 2 x_1^- + x_2 + y_1 = 3$$
$$x_1^+ - x_1^- + 2 x_2 - y_2 = 3$$
$$x_1^+ \geq 0; x_1^- > 0; x_2 \geq 0; y_1 \geq 0; y_2 \geq 0;$$

Запишем симплекс-таблицу:

$x_Б$	b	x_1^+	x_1^-	x_2	y_1	y_2
	3	2	-2	1	1	0
	3	1	-1	2	0	-1

Выберем в качестве базисных переменных y_1 и y_2.

$x_Б$	b	x_1^+	x_1^-	x_2	y_1	y_2
y_1	3	2	-2	1	1	0
y_2	3	1	-1	2	0	-1

Выполним преобразования таким образом, чтобы на пересечении строки y_1 и столбца y_1 стояла 1, остальные элементы столбца y_1 равнялись 0, аналогично строка y_2 и столбец y_2.

x_B	b	x_1^+	x_1^-	x_2	y_1	y_2
y_1	3	2	-2	<u>1</u>	1	0
y_2	-3	-1	1	-2	0	1

Базисное решение имеет вид $x_1^+ = 0$; $x_1^- = 0$; $x_2 = 0$; $y_1 = 3$; $y_2 = -3$ Это базисное решение не является опорным планом, так как $y_2 < 0$.

Заменим базисную переменную y_1 на x_2. Разрешающий элемент в столбце подчеркнут. На месте разрешающего элемента преобразованием Гаусса-Жордана получим 1, все остальные элементы столбца - нули.

x_B	b	x_1^+	x_1^-	x_2	y_1	y_2
y_1	3	2	-2	1	1	0
x_1^+	3	3	-3	0	2	1

Базисное решение имеет вид (3,0,3,0,0). Оно является невырожденным опорным планом

Задание

Найти опорные планы и базисные решения для задачи линейного программирования. Рассмотреть не менее трех базисов. Выбрать среди них опорный план. Среди всех опорных планов выбрать экстремальные. Варианты заданий приведены ниже.

№ варианта	Задание
1	$-3x_1 - 2x_2 - x_3 \text{-->} min$ $x_1 + x_2 + 2x_3 = 1$ $x_1 - x_2 + x_3 = 1$ $x_i \geq 0,\ i = 1,2,3$
2	$2x_1 + 3x_2 + 5x_3 \text{-->} max$ $x_1 + x_2 + x_3 \leq 1$ $x_1 + x_2 + x_3 = 1$ $x_1 - x_2 + x_3 = 1$ $x_i \geq 0,\ i = 1,2,3$
3	$x_1 - 4x_2 + 5x_3 \text{-->} max$ $2x_1 + x_2 + 2x_3 = 4$ $x_1 - x_2 - x_3 \leq 2$ $x_i \geq 0,\ i = 1,2,3$

4	$x_1 - 4x_2 + 5x_3 \longrightarrow max$ $2x_1 + x_2 + x_3 \leq 4$ $x_1 - x_2 - x_3 = 2$ $x_i \geq 0,\ i=1,2,3$
5	$-x_1 + 4x_2 - 5x_3 \longrightarrow min$ $2x_1 + x_2 + x_3 \leq 4$ $x_1 - x_2 - x_3 \geq 2$ $x_i \geq 0,\ i=1,2,3$
6	$3x_1 - 2x_2 - 2x_3 - 3x_4 \dashrightarrow$ max $x_1 - x_2 + x_3 + x_4 = 1$ $x_1 - x_2 - x_3 - x_4 = 1$ $x_i \geq 0,\ i=1,2,3$
7	$x_1 + x_2 + x_3 \dashrightarrow min$ $x_1 + x_2 + x_3 \leq 1$ $x_1 + x_2 + x_3 \geq 1$ $x_1 - x_2 + x_3 \leq 1$ $x_i \geq 0,\ i=1,2,3$
8	$x_1 - x_2 - x_3 \longrightarrow max$ $10x_1 + x_3 \leq 10$ $10x_2 + x_3 \geq 10$ $x_i \geq 0,\ i=1,2,3$
9	$x_1 + x_2 + 2x_3 \longrightarrow min$ $10x_1 + x_3 \geq 10$ $10x_2 + x_3 \leq 10$ $x_i \geq 10,\ i=1,2,3$
10	$-x_1 - x_2 - 2x_3 \dashrightarrow max$ $10x_1 + x_3 \geq 1$ $10x_2 + x_3 \leq 1$ $x_i \geq 0,\ i=1,2,3$
11	$x_1 + x_2 + x_3 + x_4 \longrightarrow max$ $x_1 + x_2 + 3x_3 + 4x_4 = 12$ $x_1 - x_2 + x_3 - x_4 = 2$ $x_i \geq 0,\ i=1,2,3,4$
12	$x_1 + 2x_2 - x_3 - x_4 - x5 \longrightarrow min$ $x_1 + x_2 + 2x_3 - x_4 \leq 2$ $x_1 + x_2 + 3x_3 + 4x_4 = 12$ $x_i \geq 0,\ i=1,2,3,4,5$

11	$x_3 + x_4 \rightarrow max$ $x_1 + x_2 + 3x_3 + 4x_4 = 12$ $x_1 + x_2 + x_3 - x_4 \le 2$ $x_i \ge 0,\ i=1,2,3,4$
14	$-x_1 - x_2 + x_3 \rightarrow min$ $x_1 + x_2 + x_3 + x_4 = 4$ $x_1 - 3x_2 + x_3 - x_4 = -2$ $x_i \ge 0,\ i=1,2,3,4$
15	$-x_1 + 2x_2 - x_3 \rightarrow max$ $x_1 - x_2 + 2x_3 \le 0$ $x_1 + x_2 + 5x_3 \ge 2$ $x_i \ge 0,\ i=1,2,3$
16	$x_1 - x_2 + 4x_3 \rightarrow max$ $x_1 + 2x_2 - 3x_3 = 3$ $2x_1 - x_2 + 4x_3 = 1$ $x_i \ge 0,\ i=1,2,3$
17	$-x_1 + 4x_2 - x_3 \rightarrow max$ $x_1 + 2x_2 + x_3 = 3$ $2x_1 + x_2 - x_3 = 0$ $x_i \ge 0,\ i=1,2,3$
18	$x_1 - 2x_2 - 4x_3 \rightarrow min$ $x_1 - x_2 - 2x_3 \ge 1$ $x_1 + x_2 + x_3 \le 3$ $x_i \ge 0,\ i=1,2,3$
19	$4x_1 + x_2 + 5x_3 \rightarrow max$ $x_2 + x_3 \le 2$ $3x_1 + 2x_2 - x_3 \le 1$ $x_i \ge 0,\ i=1,2,3$
20	$x_1 + x_3 + x_4 \rightarrow max$ $x_1 - x_2 + x_3 = 1$ $3x_1 + 2x_2 + 2x_3 = 17$ $x_i \ge 0,\ i=1,3$
21	$x_1 + 4x_2 - 7x_3 \rightarrow max$ $2x_1 - 2x_2 + 14x_3 \ge 2$ $x_1 - 2x_2 + 10x_3 = 0$ $x_i \ge 0,\ i=1,2,3$
22	$8x_1 + 2x_2 - 3x_3 \rightarrow max$ $x_1 + x_2 + x_3 = 5$ $3x_1 + x_2 - x_3 = -3$ $x_i \ge 0,\ i=1,2,3$

23	$3x_1 + 2x_2 + 10x_3 \to min$ $x_1 + 10x_2 + 11x_3 = 31$ $x_1 - x_2 = -2$ $x_i \geq 0,\ i=1,2,3$
24	$2x_1 + x_2 + x_3 + x_4 \to max$ $x_1 - x_2 + x_3 - x_4 \leq 2$ $x_1 + x_3 - x_4 \geq 1$ $x_1 \geq 0,\ x_2 \geq 0,\ x_3 \geq 0,\ x_4 \geq 0$
25	$-x_1 - x_2 + 4x_3 + x_4 = 2$ $x_1 + x_2 + 4x_3 + 2x_4 = 2$ $x_1 - x_2 - 2x_3$ $x_i \geq 0,\ i=1,2,3,4$
26	$x_1 + x_2 + 2x_4 \to max$ $x_1 + x_3 + x_4 = 4$ $x_1 - 2x_2 - 3x_3 + x_4 = 0$ $x_i \geq 0,\ i=1,2,3,4$
27	$x_1 - x_2 + x_3 + 2x_4 \to max$ $x_1 + x_2 + x_3 + 2x_4 = 7$ $x_2 + x_3 + x_4 = 5$ $x_3 - x_4 = 3$ $x_i \geq 0,\ i=1,2,3,4$
28	$-x_1 - x_2 - x_3 - 5x_4 \to min$ $x_1 - x_2 + 3x_4 = -1$ $x_1 - x_2 - x_4 = 3$ $2x_1 + x_2 + x_3 - x_4 = 5$ $x_i \geq 0,\ i=1,2,3,4$
29	$x_1 + x_2 - x_3 \to max$ $x_1 + x_3 = 2$ $x_1 + 0.5x_2 - x_3 = 0$ $x_i \geq 0,\ i=1,2,3$
30	$x_1 + x_2 + x_3 \to max$ $2x_1 + x_2 + x_3 = 3$ $x_1 + 2x_2 - x_3 = 3$ $2x_1 + x_2 + x_3 - x_4 = 5$ $x_i \geq 0,\ i=1,2,3$

ЛАБОРАТОРНАЯ РАБОТА № 13

Симплекс-метод. Метод искусственного базиса

Рассмотрим задачу линейного программирования в виде:

$$\langle c,x\rangle \longrightarrow max \qquad (1)$$

$$Ax=b,\ x\geq 0_n.$$

Будем считать, что $b\geq 0_m$ и в матрице A можно выделить матрицу B (базисную матрицу), которая перестановкой столбцов может быть приведена к единичной. Всю информацию занесем в симплекс-таблицу:

x_B	B	x_1	x_2	x_3	...	x_n
x_{i1} x_{i2} ... x_{im}						
d	d_0	d_1	d_2	d_3	...	dn

Столбцы x_i, $i=1,...,n$ содержат элементы векторов-столбцов a_i матрицы A. В строку d записываем значения d_j, которые вычисляются по формуле:

$$d_j=\langle c_B,\ a_j\rangle-c_j,\ j=1,...,n,$$

$$d_o=\langle c_B,\ b\rangle.$$

В столбце x_B содержатся базисные переменные, при этом должно выполняться требование: на пересечении строки x_{ik} и столбца x_{ik} должна стоять 1, остальные элементы этого столбца равны 0.

Алгоритм симплекс-метода

<u>*Начало.*</u>

M_1: $k:=argmin(d_j)$, $j=1,...,n$

Если $d_k\geq 0$, то получен оптимальный опорный план, <u>иначе</u>

<u>Если</u> $a_{ik}\leq 0\ \forall i=1,...,m$, то максимум в задаче равен ∞,

<u>Иначе</u> $l:=argmin(b_i/a_{ik})$, $i=1,...,m$, $a_{ik}>0$.

Переменная x_l из базиса выходит, в базис входит переменная x_k.

Выполним преобразования симплекс-таблицы по формулам:

$$a_{ij}=\begin{cases} a_{ij}-\dfrac{a_{lj}a_{ik}}{a_{lk}} & i\neq l,\ j=\overline{0,n} \\ \dfrac{a_{lj}}{a_{lk}} & i=l,\ j=\overline{0,n} \end{cases}, \qquad d_j=d_j-\frac{a_{lj}d_k}{a_{lk}}\ \ j=\overline{0,n}, \quad a_{ij}=a_{ij},\ \ i=\overline{1,m},\ j=\overline{1,n}$$

<u>идти на M_1</u>

<u>*Конец*</u>

Если в задаче (1) $b \geq 0_m$, но нет единичной базисной матрицы B, применяем метод искусственного базиса.
Строим новую задачу:

$$<-I_m, z> \longrightarrow max$$
$$Ax+Ez=b$$
$$x \geq 0_n, \; z \geq 0_m$$

и решаем ее алгоритмом симплекс-метода. Если в ней значения функционала равно 0, то, используя получившийся базис, решаем исходную задачу.

Замечание. В вырожденных задачах при значении функционала равном 0 не все переменные z_i могут выйти из базиса. В этом случае нужно сделать еще несколько итераций для вывода их из базиса.

Пример

Рассмотрим задачу:

$$-2x_1 - x_2 \dashrightarrow max$$

$$x_1 + x_2 \leq 1$$
$$2x_1 + 3x_2 \geq 6$$
$$x_1 \geq 0, x_2 \geq 0.$$

На рисунке видно, что множество планов x - пусто. Приведем задачу к каноническому виду:

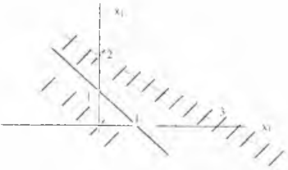

Рис. 1. Геометрическая интерпретация задачи 1.

$$\begin{array}{ll} -2x_1 - x_2 & \dashrightarrow max \\ x_1 + x_2 + y_1 & = 1 \\ 2x_1 + 3x_2 & - y_2 = 6 \end{array}$$

В первом равенстве в качестве базисной переменной можно взять y_1.

во втором равенстве такой переменной нет, поэтому вводим искусственную переменную z_1, и задачу по поиску исходного опорного плана можно записать:

$$-2x_1 - x_2 - z_1 \longrightarrow max$$
$$x_1 + x_2 + y_1 = 1$$
$$2x_1 + 3x_2 - y_2 + z_1 = 6$$
$$x_i \ge 0, y_i \ge 0, z_1 \ge 0$$

Составим симплекс-таблицу:

x_B	B	x_1	x_2	y_1	y_2	z_1
y_1	1	1	$\boxed{1}$	1	0	0
z_1	6	2	3	0	-1	1
d	-6	-2	-3	0	1	0

Находим $min\ d_j = -3,\ j=1,...,5,\ argmin(d_j)=2$, следовательно, переменную x_2 вводим в базис. Находим $argmin(\ b_i / a_{i2}) = 1, a_{i2} > 0,\ i = 1,2$, следовательно, переменную y_1 из базиса выводим. Элемент $a_{lk} = a_{12} = 1$ обведем квадратом. Преобразуем симплекс-таблицу по формулам, выписанным в алгоритме, получим:

x_B	b	x_1	x_2	y_1	y_2	z_1
x_2	1	1	1	1	0	0
z_1	3	-1	0	-3	-1	-1
d	-3	1	0	3	1	0

Все значения $d_j \ge 0,\ j=1,...,n$, поэтому в искусственной задаче максимум равен 3, следовательно, в исходной задаче решения нет.

Рассмотрим еще один пример. Изменим в первом неравенстве исходной задачи знак неравенства на противоположный. Задача примет вид:

$$-2x_1 - x_2 \longrightarrow max$$
$$x_1 + x_2 \ge 1$$
$$2x_1 + 3x_2 \ge 6$$
$$x_1 \ge 0, x_2 \ge 0.$$

Приведем задачу к каноническому виду:

$$-2x_1 - x_2 \longrightarrow max$$
$$x_1 + x_2 - y_1 = 1$$
$$2x_1 + 3x_2 - y_2 = 6$$
$$x_1 \ge 0, x_2 \ge 0, y_1 \ge 0, y_2 \ge 0$$

Ни в первом, ни во втором уравнении нет переменных, которые можно ввести в базис. Применим метод искусственного базиса. Задача преобразуется к виду:

$$-2x_1 - x_2 - z_1 - z_2 \longrightarrow max$$
$$x_1 + x_2 - y_1 + z_1 = 1$$
$$2x_1 + 3x_2 - y_2 + z_2 = 6$$

Составим симплекс-таблицу и решим задачу.

x_B	B	x_1	x_2	y_1	y_2	z_1	z_2
z_1	1	1	1	1	0	1	0
z_2	6	2	3	0	-1	0	1
d	-7	-3	-4	1	1	0	0
x_2	1	1	1	-1	0	1	0
z_2	3	-1	0	3	-1	-3	1
d	-3	1	0	-3	1	4	0
x_2	2	2/3	1	0	-1/3	0	1/3
y_1	1	-1/3	0	1	-1/3	-1	1/3
d	0	0	0	0	0	1	1

В этой задаче минимум равен 0, переменные z_i вышли из базиса, поэтому можно приступить к решению исходной задачи, используя полученный опорный план.

x_B	b	x_1	x_2	y_1	y_2
x_2	2	2/3	1	0	-1/3
y_1	1	-1/3	0	1	-1/3
d	-2	-1/3	0	0	2/3
x_1	3	1	3/2	0	-1/2
y_1	2	0	1/2	1	-1/2
d	-1	0	1	0	1/2

Все оценки $d_j \geq 0$, $j=\overline{1,4}$, поэтому получено оптимальное решение.

$$x_1 = 3,\ x_2 = 0\ ,\ y_1 = 2,\ y_2 = 0$$

Значение функционала $f = -1$.

Задание

Среди базисных решений, полученных в предыдущей лабораторной работе, выбрать решение, не являющееся опорным планом. Взяв этот базис за исходный, решить задачу симплексным методом искусственного базиса. Варианты задания взять из лабораторной работы № 12. Результаты вычислений представить в виде симплекс-таблицы.

ЛАБОРАТОРНАЯ РАБОТА № 14

Модифицированный симплексный метод

Задача линейного программирования записана в виде:

$$<c\,x> \longrightarrow max$$

$$Ax=b,\ x \geq 0_n\ .$$

Будем считать, что $b \geq 0_n$, и в матрице A сложно выделить базисную матрицу B, для которой $x_B \geq 0_m$,а также известна обратная матрица B^{-1}. Введем искусственные переменые, задача примет вид:

$$-\sum_{i=1}^{m} z_i \to \max$$

$$Ax+z=b,\ x \geq 0_n\ ,\ z \geq 0_m\ .$$

Составим симплекс-таблицу:

x_B	B^{-1}	$B^{-1}b$	$B^{-1}a_k$
z_i	1 0 . . . . . 0 0 1 0 . . . 0 . . . . . 0 . . . . . 0 1	b_i	a_j
λ	$\lambda_i\ \lambda_k$	d_0	d_k

Здесь $\lambda = c_B B^{-1}$ Вычислим оценки d_j по формулам $d_j =<\lambda, a_j> - c_j$. Переменную, для которой $d_k <0$, вводим в базис. Для этого заполняем столбец

$B^{-1}a_k$ и выполняем преобразование таблицы по формулам пересчета, приведенным в алгоритме симплекс-метода . Вычисления продолжаем до тех пор, пока все оценки d_j не станут положительными или равными нулю. Таким образом будет решена искуственная задача и тем самым найден исходный базис и B^{-1} исходной задачи.

В получившейся симплекс-таблице заново по исходной задаче заполняем строку λ и продолжаем решение до тех пор, пока существуют $d_k < 0$

Пример

$$-x_1 - 2x_2 \longrightarrow max$$

$$x_1 + x_2 \geq 1$$

$$2x_1 + 3x_2 \geq 6$$

$$x_i \geq 0$$

После приведения задачи к канонической форме и построения искусственной задачи, получим:

$$-z_1 \quad -z_2 \quad \longrightarrow max$$

$$x_1 + x_2 - y_1 \quad + z_1 \quad = 1$$

$$2x_1 + 3x_2 \quad -y_2 \quad + z_2 = 6$$

$$x_i \geq 0, \qquad y_i \geq 0, \qquad z_i \geq 0.$$

Составляем симплекс-таблицу модифицированного симплекс-метода:

x_B	B^{-1}	$B^{-1}b$	$B^{-1}a_i$
z_1	1 0	1	|1|
z_2	0 1	6	2
λ	-1 -1	-7	-3

Вычислим λ, $B^{-1}b$, $B^{-1}a_i$ и d_j по соответствующим формулам, $d_1 > 0$, следовательно, переменную x_1 вводим в базис, переменную z_1 выводим из базиса (правило поиска переменной, выходящей из базиса аналогично прямому симплекс-методу).

Вновь заполняем симплекс-таблицу:

x_B	B^{-1}	$B^{-1}b$	$B^{-1}a_i$
x_1	1 0	1	1
z_2	-2 1	4	1
λ	2 -1	-4	-1

Найдем $d_2 = -1$, так как $d_2 < 0$, то x_2 вводим в базис, x_1 - выводим из базиса. Выполним преобразования. Симплекс-таблица примет вид:

x_B	B^{-1}	$B^{-1}b$	$B^{-1}a_i$
x_2	1 0	1	-1
z_2	-3 1	3	3
λ	3 -1	-3	-3

Ищем оценки d_i:

$$d_1 = \begin{pmatrix} 3 \\ -1 \end{pmatrix}\begin{pmatrix} 1 \\ 2 \end{pmatrix} = 1, \quad d_1 \geq 0, \quad d_3 = \begin{pmatrix} 3 \\ -1 \end{pmatrix}\begin{pmatrix} -1 \\ 0 \end{pmatrix} = -3, \quad d_3 < 0$$

y_1 вводим в базис. Заполним столбец $B^{-1}a_3$:

$$\begin{pmatrix} 1 & 0 \\ -3 & 1 \end{pmatrix}\begin{pmatrix} -1 \\ 0 \end{pmatrix} = \begin{pmatrix} -1 \\ 3 \end{pmatrix}$$

Выберем из найденного столбца минимальное среди всех положительных значений. В данном случае имеем одно положительное значение 3, следовательно, z_2 выводим из базиса. Вновь строим симплекс-таблицу:

x_B	B^{-1}	$B^{-1}b$	$B^{-1}a_i$
x_2	0 1/3	2	
y_1	-1 1/3	1	
λ	0 0	0	

Искусственная задача решена, так как z_1 и z_2 из базиса вышли. Решаем исходную задачу, заново вычисляем λ и вносим его в таблицу:

$$\lambda = c_B B^{-1} = (-2,0)\begin{pmatrix} 0 & 1/3 \\ -1 & 1/3 \end{pmatrix} = (0, -2/3)$$

$x_Б$	B^{-1}	$B^{-1}b$	$B^{-1}a_s$
x_2	0 1/3	2	2/3
y_1	-1 1/3	1	-1/3
λ	0 -2/3	-4	-1/3

Найдем

$$d_1 = \begin{pmatrix} 1 \\ 2 \end{pmatrix}\begin{pmatrix} 0 \\ 2/3 \end{pmatrix} + 1 = -1/3$$

Переменная x_1 вводится в базис. x_2 - выводится из базиса. Преобразуем симплекс-таблицу:

$x_Б$	B^{-1}	$B^{-1}b$	$B^{-1}a_s$
x_1	0 1/3	3	
y_1	-1 1/2	2	
λ	0 -1/2	-3	

Вычислим оценки d_1, d_2 =<(0,-1/2)(1 3)> + 2=1/2: d_3 =1/2 Все оценки положительные, следовательно, получено оптимальное решение

x'=(3,0). y=(2,0).

Задание

Решить задачу линейного программирования модифицированным симплекс-методом. Результаты вычислений представить в виде симплекс-таблицы. Варианты заданий взять из лабораторной работы № 12

ЛАБОРАТОРНАЯ РАБОТА № 15

Двойственный симплекс-метод

Рассматривается задача линейного программирования в виде:

$\langle c\, x\rangle \longrightarrow max$

$Ax=b,\ x \geq 0_n$.

Предположим, что матрица A содержит единичную матрицу (это всегда можно сделать). Строим симплекс-таблицу:

$x_Б$	b	x_1	x_2	x_3	...	x_n
d	d_0	d_1	d_2	d_3	...	d_n

В двойственном симплекс-методе допустимо, чтобы в столбце были отрицательные элементы. Рассчитываем оценки (строку d) так же, как и в прямом симплекс- методе. Задача двойственно допустима, если $d_j \geq 0$. Предположим, что все $d_j \geq 0$.

Алгоритм двойственного симплекс-метода

1. В столбце B ищем $l = \arg\min b_i$, $i = 1, \ldots, m$. Если $b_l > 0$, то решение оптимально, иначе x_l - выводим из базиса.

2. В строке l находим отрицательные элементы. Если все $a_{lj} \geq 0$, то исходная задача решения не имеет. Найдем

$k = \arg\max(d_j / a_{lj})$, $j = 1, \ldots, n$, $a_{lj} < 0$.

Переменная x_k вводится в базис, a_{lk} - разрешающий элемент.

3. Делим строку l на $a_{lk} \geq 0$, затем выполним преобразование симплекс-таблицы аналогично прямому симплексному методу.

Переход к двойственно допустимой форме

Рассмотрим случай, когда для исходного базиса среди d_j существуют строго отрицательные d_j. В этом случае к исходной добавляется фиктивное ограничение:

$$\Sigma x_j \leq M \qquad (1)$$

Суммирование по тем j, для которых $d_j < 0$, здесь M - параметр, который может быть сколь угодно большим числом. Приведем это неравенство к каноническому виду

$$\sum_{j: d_j < 0} x_j + x_{n+1} = M \qquad (2)$$

Тогда симплекс-таблица примет вид:

x_B	b^1	b^0	x_1	x_2	...	x_n
x_{n+1}	1	0				
d						

Столбец b разделен на 2 части. Значение b_i представляется в виде:

$$b_i = b_i^1 M + b_i^0, \qquad (3)$$

так как M может быть сколь угодно большим числом, то b_i совпадает по знаку с b_i^1, если $b_i^1 \neq 0$, и совпадает по знаку с b_i^0, если $b_i^1 = 0$.

На первом шаге в качестве разрешающей строки берется x_{n+1} строка. В качестве разрешающего столбца столбец с минимальной оценкой d_j. После первого шага симплекс-метода все оценки становятся больше либо равны нулю, и задача принимает двойственно допустимую форму.

Замечание. Если в оптимальном решении в значении функционала коэффициент при M (d_0')>0, то максимум задачи неограничен.

Пример 1

$$-2x_1 - x_2 \longrightarrow max$$

$$x_1 + x_2 \geq 1$$
$$2x_1 + 3x_2 \leq 6$$
$$x_i \geq 0$$

Приведем к каноническому виду:

$$-2x_1 - x_2 \longrightarrow max$$

$$-x_1 - x_2 + y_1 = -1$$

$$2x_1 + 3x_2 + y_2 = 6$$

$$x_i \geq 0,\ y_i \geq 0.$$

Составим симплекс-таблицу:

x_B	B	x_1	x_2	y_1	y_2
y_1	-1	-1	-1	1	0
y_2	6	2	3	0	1
d	0	2	1	0	0

Форма двойственно допустима, поэтому решаем по алгоритму двойственного симплексного метода. Переменную y_1 выводим из базиса, x_2- вводим в базис:

x_B	B	x_1	x_2	y_1	y_2
y_1	1	1	1	-1	0
y_2	3	-1	0	3	0
d	-1	1	0	1	0

Все оценки $d_j \geq 0$, $b_i > 0$; следовательно, получено оптимальное решение.

$x^*=(0,1)$, $f(x^*)=-1$

Пример 2

$$2x_1 + x_2 \longrightarrow max$$

$$-x_1 - x_2 + y_1 = -1$$

$$-2x_1 + 3x_2 + y_2 = 6$$

$$x_i \geq 0,\ y_i \geq 0.$$

Симплекс-таблица имеет вид:

x_B	B	x_1	x_2	y_1	y_2
y_1	-1	-1	-1	1	0
y_2	6	2	3	0	1
d	0	-2	-1	0	0

Из симплекс-таблицы видно, что она имеет двойственно недопустимую форму. Строим дополнительное ограничение по формуле (1):

$$x_1 + x_2 \leq M$$

Приводим его к каноническому виду и записываем в симплекс-таблицу:

$$x_1 + x_2 + y_3 = M$$

Симплекс-таблица примет вид:

x_B	B	x_1	x_2	y_1	y_2	y_3
y_1	-1	-1	-1	1	0	0
y_2	6	2	3	0	1	0
y_3	M	1	1	0	0	1
d	0	-2	-1	0	0	0

y_3 - выводим из базиса, x_1 - вводим в базис. Выполняем симплекс-преобразование, получим двойственно допустимую форму

X_B	k^T_i	b^0_i	x_1	x_2	y_1	y_2	y_3
y_1	1	-1	0	0	1	0	1
y_2	-2	6	0	1	0	1	-2
x_1	1	0	1	1	0	0	1
d	1	0	1	1	0	0	2

y_2 - выводим из базиса, x_2 - вводим в базиса. Выполним симплекс-преобразование, получим таблицу:

x_B	b'_i	b^0	x_1	x_2	y_1	y_2	y_3
y_1	0	1	0	-1/3	1	1/3	0
y_3	1	-2	0	1/3	0	-1/3	1
x_1	0	2	1	2/3	0	1/3	0
d	0	4	0	0	0	0	0

Получено оптимальное решение, причем $y_3=M-2$, $y_1=1$, $x_1=0$, $x_2=0$, значение функционала равно 4.

Задание

Решить задачу линейного программирования двойственным симплексным методом. Изменить вариант задания таким образом, чтобы можно было применить двойственный симплекс-метод. Результаты вычислений представить в виде симплекс-таблицы.

Варианты заданий взять в лабораторной работе № 12

ЛАБОРАТОРНАЯ РАБОТА № 16

Метод отсекающих гиперплоскостей для решения задач нелинейного программирования

Рассматривается задача нелинейного программирования в виде:

$<c\,x> \longrightarrow min$ (1)

$Ax=b,\ x\geq 0$.

$\varphi(x)\leq 0$.

То есть в этой задаче часть ограничений линейная, часть нелинейная (если нелинейных ограничений $\varphi_i(x),\ i\in J$, несколько, то $\varphi(x)=max\{\varphi_i(x)|\ i\in J\}$).

Метод решения

1 Отбрасываем нелинейное ограничение и решаем задачу линейного программирования. Пусть x' ее решение.

2. Полученное оптимальное решение проверяем. Если оно удовлетворяет всем нелинейным ограничениям, то задача решена, иначе строим дополнительное ограничение, отсекающее это решение, но не отсекающее решения исходной задачи. Для этого выполняем пункт 3.

3. Находим $a=\sum\alpha_i\, x\varphi_i'(x')$, $\sum\alpha_i=1,\ \alpha_i\geq 0$, где индекс i пробегает значения тех функций $\varphi_i(x')$, для которых $\varphi_i(x')=\varphi(x')$. Добавляем полученное ограничение к имевшимся ранее линейным ограничениям и переходим к шагу 1.

Работу алгоритма повторяется до тех пор, пока не будет найдено оптимальное решение, удовлетворяющее всем нелинейным и линейным ограничениям.

Рассмотрим пример:

$$x_1+x_2 \longrightarrow max$$

$$x_1 \leq 2$$

$$x_2 \leq 2$$

$$x_2\leq x_1^2+3,$$

$$x_1\leq 0$$

$$x_2\leq 0$$

Третье ограничение эквивалентно $\varphi_1(x)=x_2+x_1^2-3\leq 0$.

1. Отбрасываем нелинейное ограничение. Приводим к каноническому виду:

$$x_1 + x_2 \dashrightarrow max$$

$$x_1 \qquad + y_1 = 2$$

$$x_2 + y_2 = 2$$

$$x_1,\ x_2,\ y1,\ y2 > 0.$$

Строим симплекс-таблицу:

x_B	B	x_1	x_2	y_1	y_2
x_1	2	1	0	1	0
x_2	2	0	1	0	1
d	4	0	0	1	1

Все $d_j \geq 0$, следовательно, $x'=(2,2)$ оптимальное решение. Подставим x' в нелинейное ограничение. x' не удовлетворяет нелинейному ограничению, поэтому будем строить дополнительное ограничение по формуле:

$$\varphi_1(x') + <a, x - x'> \leq 0;$$

$$\varphi_1(x') = 3;\ \varphi_1'(x) = \begin{pmatrix} 2x_1 \\ 1 \end{pmatrix};\ \varphi_1'(x') = \begin{pmatrix} 4 \\ 1 \end{pmatrix};$$

тогда ограничение принимает вид:

$$3 + < \begin{pmatrix} 4 \\ 1 \end{pmatrix} (x_1 - 2, x_2 - 2) > \leq 0$$

откуда

$$4(x_1 - 2) + x_2 - 2 + 3 \leq 0.$$

Приведем это ограничение к каноническому виду. $4x_1 + x_2 + y_3 = 7$. Из исходной задачи следует, что $x_1 = 2 - y_1$, $x_2 = 2 - y_2$, тогда ограничение примет вид:

$$-4y_1 - y_2 + y_3 = -3$$

Добавим эту строку к последней симплекс-таблице, получим задачу, которую будем решать двойственным симплексным методом.

	λ_B	B	x_1	x_2	y_1	y_2	y_3
1	x_1	2	1	0	1	0	0
1	x_2	2	0	1	0	1	0
0	y_3	-3	0	0	-4	-1	1
	d	4	0	0	1	1	0
	x_1	5/4	1	0	0	-1/4	1/4
	x_2	2	0	1	0	1	0
	y_1	3/4	0	0	1	1/4	-1/4
	d	13/4	0	0	0	3/4	1/4

Таким образом, оптимальное решение имеет вид $x'=(5/4,2)$. Подставим полученное решение в нелинейное ограничение; x' не удовлетворяет нелинейному ограничению $\varphi_1(x)$, следовательно, строим дополнительное ограничение по формуле (2):

$$\varphi_1(x')=9/16;\ \varphi_1'(x')=(5/2,1),$$

следовательно,

$$9/16+<(5/2,1)^T(x_1-5/4,x_2-2)>\leq 0,$$

отсюда

$$9/16+5/2\,x_1-25/8+x_2-2\leq 0.$$

Подставим выражения для x_1 и x_2 из последней таблицы:

$$x_1=5/4+1/4y_2-1/4y_3,$$

$$x_2=2-y_2.$$

Тогда дополнительное ограничение примет вид:

$$-3/8\,y_2-5/8\,y_3+9/16\leq 0,$$

Приведем к каноническому виду:

$$-3/8\,y_2-5/8\,y_3+y_4=-9/16$$

Добавим это ограничение к последней симплекс-таблице и вновь будем искать оптимальное решение двойственным симплексным методом и т.д

Задание

Самостоятельно построим нелинейное ограничение, исходя из известного оптимального решения вашего варианта из предыдущей лабораторной работы. На начальном шаге известное оптимальное решение не должно удовлетворять построенному нелинейному ограничению. Варианты заданий взять из лабораторной работы N 12. Результаты вычислений представить в виде симплекс-таблицы. Выполнить не менее трех итераций

Литература

1. Карманов В.Г. Математическое программирование. М.: Наука, 1986.
2. Моисеев Н.Н., Иванов Ю.П.,Столяров Е.М. Методы оптимизации. М., 1978.
3. Васильев Ф.П. Численные методы решения экстремальных задач. М., 1980.
4. Химмельблау Д. Прикладное нелинейное программирование. М., 1975.
5. Пшеничный Б.Н., Данилин Ю.М. Численные методы в экстремальных задачах. М.: Наука, 1975.
6. Тетерев А.Г. Методы одномерной оптимизации. Куйбышев: КГУ, 1983.
7. Стронгин Р.Г. Численные методы в многоэкстремальных задачах. М., 1978.
8. Сухарев А.Г., Тимохов А.В., Федоров В.В. Курс методов оптимизации. М.: Наука, 1986.

СОДЕРЖАНИЕ