МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное бюджетное образовательное учреждение
высшего образования

«Воронежский государственный технический университет»

С. А. ОЛЕЙНИКОВА, Т. И. СЕРГЕЕВА, М. Ю. СЕРГЕЕВ

ЧИСЛЕННЫЕ МЕТОДЫ ОПТИМИЗАЦИИ

Практикум

Воронеж 2021

УДК 519.6(075.8)
ББК 22.193я7
О 532

Рецензенты:
кафедра математических методов исследования операций Воронежского государственного университета (зав. кафедрой д-р техн. наук, профессор Т. В, Азарнова);

д-р техн. наук, профессор В. Ф. Барабанов

Олейникова С. А., Сергеева Т. И., Сергеев М. Ю.
О532 **Численные методы оптимизации: практикум** / С. А. Олейникова, Т. И. Сергеева, М. Ю. Сергеев. - ФГБОУ ВО «Воронежский государственный технический университет». - Воронеж: Изд-во ВГТУ, 2021. - 90 с.

ISBN 978-5

Практикум содержит теоретические сведения и варианты заданий для лабораторных работ по курсу «Методы оптимизации». Основное внимание уделено численным методам оптимизации.
Предназначено для студентов направления 09.03.01 «Информатика и вычислительная техника» (профиль «Вычислительные машины, комплексы, системы и сети») очной и заочной форм обучения.

Ил. 22. Табл. 55. Библиогр.: 8 назв.

УДК 519.6(075.8
ББК 22.193я7

Печатается по решению редакционно-издательского совета Воронежского государственного технического университета

ISBN 978-

ВВЕДЕНИЕ

Функционирование любых производственных или обслуживающих систем требует решения оптимизационных задач. Это могут быть задачи оптимизации использования ресурсов или планирования работы предприятия, оптимизации транспортных потоков или определения наилучшего маршрута; повышения эффективности работы склада, оценка наилучшего объема заказов и т.д. Все эти задачи являются оптимизационными, поскольку должны определить наилучшее значение некоторого показателя, который можно назвать целевой функцией. В связи с этим необходимо использовать методы, которые позволят решать такие задачи, то есть методы оптимизации.

Данные методы можно классифицировать по многим критериям. Это задачи дискретной и непрерывной оптимизации, задачи на условный или безусловный экстремум, одномерные и многомерные задачи, задачи линейного и нелинейного программирования и т.д. Для каждой категории таких задач будут использоваться свои методы решения, ориентированные на ее специфику.

В данном пособии рассматриваются задачи разных типов, и для каждой категории описываются свои методы решения. Первая лабораторная работа посвящена изучению методов одномерной оптимизации: половинного деления и золотого сечения. Задачи линейного программирования являются предметом исследования во второй лабораторной работе. Третья лабораторная работа посвящена изучению методов нулевого порядка (использующих лишь значения целевой функции) для решения задачи на безусловный экстремум. Методы решения задач на безусловный экстремум с использованием производной (методы первого порядка) будут изучены в четвертой лабораторной работе. Один из методов решения задачи на условный экстремум (метод штрафных функций) описан в пятой лабораторной работе. Остальные лабораторные работы посвящены исследованию методов дискретной оптимизации. В частности, шестая лабораторная работа посвящена исследованию метода динамического программирования для решения разнообразных дискретных задач (задача о рюкзаке, о конвейере, о ресурсах и т.д.). В седьмой лабораторной работе описаны наиболее простые методы дискретной оптимизации: жадные алгоритмы. И наконец, последняя лабораторная работа посвящена изучению метаэвристических алгоритмов: генетических и муравьиных алгоритмов.

Пособие снабжено большим количеством примеров, облегчающих изучение каждого из методов. Оно может быть полезно всем студентам и магистрантам, занимающимся как дискретными, так и непрерывными оптимизационными задачами.

ЛАБОРАТОРНАЯ РАБОТА № 1

МЕТОДЫ ОДНОМЕРНОЙ ОПТИМИЗАЦИИ

Цель лабораторной работы:

Освоение теоретических сведений и приобретение практических навыков решения задач одномерной оптимизации путем программной реализации методов половинного деления и золотого сечения.

1.1. Краткие теоретические сведения

В общем случае задача принятия решений в условиях определённости может быть записана в следующем виде:

$$f(x) \to \max_{x \in D}(\min) \,. \tag{1.1}$$

Здесь x может быть как числом (одномерная задача), так и вектором (многомерная задача).

Рассмотрим случай, когда целевая функция является нелинейной и зависит от единственного аргумента x. Пусть решается задача на минимум:

$$f(x) \to min_{a \le x \le b} \,. \tag{1.2}$$

Существует целый ряд методов решения подобных задач. Большинство численных методов основано на предположении, что функция на всем отрезке сохраняет выпуклость (т.е. вторая производная на этом отрезке сохраняет знак). Это обеспечивает единственный глобальный минимум на данном отрезке. В противном случае весь отрезок разбивается на такие интервалы, каждый из которых содержит единственную глобальную точку минимума.

Аналитически разделить отрезок [a,b] на промежутки, внутри каждого из которых существует единственная точка минимума, можно, взяв вторую производную от искомой функции. Известно, что если вторая производная сохраняет знак, то функция сохраняет свою выпуклость (вогнутость). Поэтому, если удастся разбить весь отрезок на интервалы, внутри которых вторая производная знак не меняет, то можно утверждать, что на данном интервале будет единственный максимум (минимум) в точке максимума (минимума) или на концах отрезка, если функция монотонно возрастает или убывает.

Будем считать, что внутри рассматриваемого интервала функция сохраняет свою выпуклость или вогнутость.

1.1.1 Метод половинного деления

Метод возвращает приближенное решение задачи (1.2) с заданной точностью ε. Рассмотрим задачу на минимум. Суть метода заключается в следующем. Отрезок [a,b] разбивается напополам точкой (*рис. 1.1*).

$$c_0 = \frac{a+b}{2}. \tag{1.3}$$

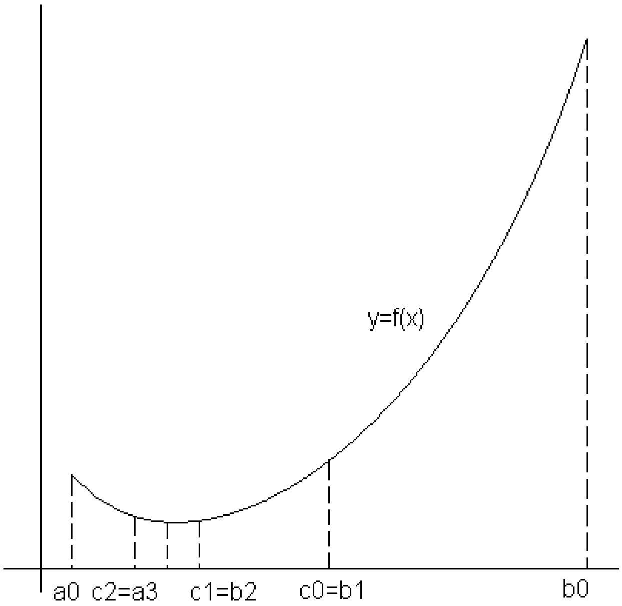

Рис. 1.1. Иллюстрация к методу половинного деления

Далее из двух отрезков [a,c_0] [c_0,b] выбирается тот, который содержит точку минимума. Он выбирается в качестве исходного отрезка [a,b] для следующего этапа. На следующем этапе новый отрезок [a,b], полученный на предыдущем этапе, делится пополам и вновь находится та его половина, которая содержит точку минимума. Данная процедура продолжается до тех пор, пока минимум не будет найден или пока не будет достигнута требуемая точность ε, т.е. пока не будет выполнено условие

$$|b - a| < \varepsilon. \tag{1.4}$$

Для поиска отрезка, который содержит точку максимума или минимума, к середине отрезка c добавляется и вычитается малое значение δ, зависящее от длины отрезка, но много меньшее этой длины. Это значение определяется формулой

$$\delta = k \cdot (b - a), \tag{1.5}$$

где k – произвольное число от 0 до 1.

Пусть $c_1 = c - \delta$; $c_2 = c + \delta$. Тогда если $f(c_1) < f(c_2)$, то можно предположить, что минимум находится в интервале [a,c_2], в противном случае – в интервале [c_1,b] (*рис. 1.2*).

Поэтому если $f(c_1) < f(c_2)$, то вместо точки b на следующем этапе будет рассматриваться точка c_2, в противном случае – вместо точки a – точка c_1.

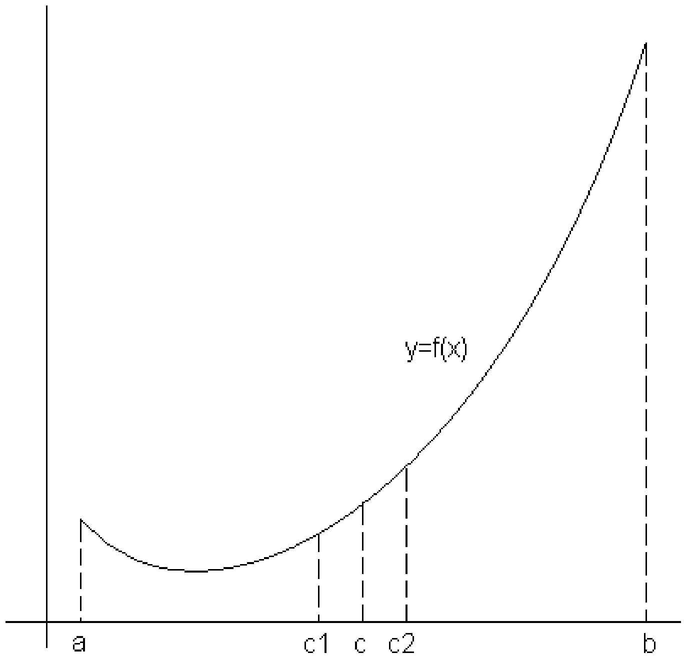

Рис. 1.2. Выбор интервала

1.1.2. Метод золотого сечения

Точка u является точкой золотого сечения отрезка [a,b], если длина всего отрезка так относится к длине большей его части, как большая часть относится к меньшей.

$$\frac{b-a}{u-a} = \frac{u-a}{b-u}. \tag{1.6}$$

Идея золотого сечения метода аналогична методу половинного деления. Однако точки c1 и c2 ищутся другим способом.

В качестве точки c1 в методе золотого сечения берется точка золотого сечения отрезка [a,b], а в качестве c2 – точка золотого сечения отрезка [a,c1].

Поиск точки золотого сечения можно упростить следующим образом. Найдем точку золотого сечения v отрезка [0,1]. Из формулы (1.6) следует, что

$$\frac{1-0}{v-0} = \frac{v-0}{1-v}. \tag{1.7}$$

Упростив, получим квадратное уравнение

$v^2+v-1=0$.

Решив, получим два ответа:

$v_1 = \frac{-1-\sqrt{5}}{2}$;

$v_1 = \frac{-1+\sqrt{5}}{2}$.

Поскольку точка должна принадлежать отрезку [0,1], остается одно решение $v_1 = \frac{-1+\sqrt{5}}{2}$.

Тогда точку с золотого сечения любого отрезка [a,b] можно получить по формуле:

$$c = a + v(b - a). \qquad (1.8)$$

Таким образом, общие схемы вышеприведенных алгоритмов поиска минимума на отрезке [a,b] можно кратко описать следующим образом:

пока(b-a)≥ε
Нц
  Определить c_1 и c_2;
  Если $f(c_1)<f(c_2)$
  То искать минимум на отрезке $[a,c_2]$
  Иначе искать минимум на отрезке $[c_1,b]$
Кц
Выдать середину полученного отрезка в качестве решения.

Решение задачи поиска максимума функции осуществляется аналогично. Разница заключается лишь в том, что в качестве отрезка для поиска минимума на следующем шаге берется такой из двух отрезков, который содержит точку максимума.

1.2. Задание к работе

Найти методом половинного деления и методом золотого сечения максимум (минимум) функции. *Исходными данными* являются:

- границы отрезка a и b;
- заданная точность ε.

В качестве *результатов* получить:

- максимальное (минимальное) значение функции на заданном интервале, полученное каждым методом;
- значение аргумента, соответствующее этому результату.

В качестве промежуточных результатов отображать:

- отрезок [a,b] на каждом шаге;
- значения c_1 и c_2;
- $f(c_1)$ и $f(c_2)$.

Все промежуточные значения поместить в таблицу:

Таблица 1.1

Промежуточные значения

№п/п	a	b	c_1	c_2	$f(c_1)$	$f(c_2)$

Предусмотреть возможность выбора метода решения задачи.

Отчет должен содержать:

- постановку задачи;
- краткое описание метода;
- краткую реализацию метода;
- результаты реализации (программную форму с входными данными и результатами).

1.3. Варианты заданий

Таблица 1.2

Варианты задания на лабораторную работу №1

№ варианта	Функция	A	B	E	Решение
1	$F(x)=-e^{-x}\cdot\ln(x)$	0.1	3	0.1 0.01 0.001	(1.763;-0.097)
2	$F(x)=-e^{-x}\cdot x$	0.1	3	0.1 0.01 0.001	(1.0;-0.368)
3	$F(x)=2\cdot x^2 + 3\cdot e^{-x}$	0	1	0.1 0.01 0.001	(0.469; 2.317)
4	$F(x)=x^2 + x + 5\cdot e^{-x}$	0	2	0.1 0.01 0.001	(0.719; 3.672)
5	$F(x)=2\cdot x^2 - x + e^{-x}$	-1	1	0.1 0.01 0.001	(0.415; 0.59)
6	$F(x)=-3\cdot e^{-x}\cdot\ln(2\cdot x)$	0.5	2.5	0.1 0.01 0.001	(1.173; -0.792)
7	$F(x)=x^4 - 10\cdot x^3+20\cdot x^2$	4	8	0.1 0.01 0.001	(5.765; -146.726)
8	$F(x)=x^4 - 12\cdot x^3+7\cdot x^2 - 5\cdot x$	8	9.5	0.1 0.01 0.001	(8.61; -1687.886)
9	$F(x)=x^4 -10\cdot x^2 + 5\cdot x$	6.5	8.5	0.1 0.01 0.001	(-2.352; -36.477)
10	$F(x)=x^4 - 10\cdot x^3 + 20\cdot x$	6.5	8	0.1 0.01 0.001	(7.409; -905.591)
11	$F(x)=x^4 - 6\cdot x^3 - 5\cdot x^2 + 10\cdot x$	4	6	0.1 0.01 0.001	(4.906; -200.466)

Окончание табл. 1.2

12	$F(x)=x^4 - 10\cdot x^3 - 30\cdot x^2 - 15\cdot x$	8	10	0.1 0.01 0.001	(9.179; -3300.241)
13	$F(x)=2x^2+3e^{-x}$	-1	1,5	0.1 0.01 0.001	(0,469; 2.317)
14	$F(x)=X^2-2x+e^x$	-1	1	0.1 0.01 0.001	(0,315; 0.839)
15	$F(x)=10x*\ln(x)-x^2/2$	0,1	1,1	0.1 0.01 0.001	(0,382; -3.749)
16	$F(x)=e^x-x^3/3+2x$	-4	4	0.1 0.01 0.001	(-1,492; -1.562)
17	$F(x)=X^2-2x-2\cos(x)$	0	2	0.1 0.01 0.001	(0,511;-2.505)
18	$F(x)=X^4-14x^3+60x^2-70x$	-1	2	0.1 0.01 0.001	(0.781;-24.37)
19	F(x)=x-ln(abs(ln(x)))	1	3	0.1 0.01 0.001	1,763; 2.33)
20	$F(x)=X/(x^2-2x+5)$	-4	2	0.1 0.01 0.001	(-2,23;-0.15)

ЛАБОРАТОРНАЯ РАБОТА № 2

МЕТОДЫ РЕШЕНИЯ ЗАДАЧ ЛИНЕЙНОГО ПРОГРАММИРОВАНИЯ

Цель лабораторной работы:

Освоение теоретических сведений и приобретение практических навыков решения задач линейного программирования.

2.1. Краткие теоретические сведения

Задачу линейного программирования можно сформулировать следующим образом. Необходимо максимизировать или минимизировать некоторую функцию f, линейно зависящую от своих аргументов $x_1,\ldots x_n$ при наличии линейных ограничений.

Универсальным методом решения задач линейного программирования является симплекс-метод. Однако его можно применять лишь в случае, если задача записана в так называемой канонической форме.

Канонический вид записи задачи линейного программирования следующий:

$$\begin{cases} \sum_{j=1}^{n} c_j x_j \to max, \\ \sum_{j=1}^{n} a_{ij} x_j = b_i, i = 1, \dots, m, \\ x_j \geq 0, j = 1, \dots, n. \end{cases} \quad (2.1)$$

Перевод задачи к каноническому виду осуществляется следующим образом:

- если целевая функция F(x) должна быть минимизирована, то к максимуму стремится функция –F(x);

- если имеет место неравенство

$$a_{i1}x_1 + a_{i2}x_2 + \cdots + a_{in}x_n \leq b_i, \quad (2.2)$$

то для того, чтобы привести неравенство к виде (2.1), необходимо добавить некоторую переменную x_{n+1}:

$$a_{i1}x_1 + a_{i2}x_2 + \cdots + a_{in}x_n + x_{n+1} = b_i\ ; \quad (2.3)$$

- если ограничение имеет вид

$$a_{j1}x_1 + a_{j2}x_2 + \cdots + a_{jn}x_n \geq b_j, \quad (2.4)$$

то для того, чтобы получить равенство, необходимо вычесть некоторую новую переменную

$$a_{j1}x_1 + a_{j2}x_2 + \cdots + a_{jn}x_n - x_{n+2} = b_j\ . \quad (2.5)$$

Предположим, что задача приведена к канонической форме. Рассмотрим более подробно симплекс-метод, позволяющий получить ее решение. Для этого введем следующее определение.

Определение. Пусть имеется система, состоящая из m уравнений и n неизвестных. Разделим переменные на два множества: (n-m) переменные положим равными нулю, а остальные m переменных определяются решением системы исходных уравнений. Если это решение единственно, то тогда ненулевые m переменных называют базисными; нулевые (n-m) переменных – свободными (небазисными), а соответствующие результирующие значения переменных называют базисным решением.

Основная идея симплекс-метода основывается на следующем.

1. Осуществляется поиск некоторого базисного решения. Если оно допустимое, то осуществляется переход к п.2; в противном случае необходимо найти другое базисное решение (процедура повторяется до тех пор, пока решение не будет допустимым).

2. Осуществляется проверка найденного решения на оптимальность. Если оно не оптимально, то предпринимается попытка поиска другого базисного решения.

Данные два шага повторяются до тех пор, пока очередное базисное решение не станет оптимальным.

Рассмотрим более подробно данные этапы. Для решения данной задачи строится симплекс-таблица. Ее структура в разных источниках может несущественно отличаться. Главные элементы, которые обязательно присутствуют в симплекс-таблице:
- первый столбец базисных переменных;
- столбцы небазисных переменных;
- столбец свободных членов;
- последняя строка таблицы – оценка целевой функции f.

Например, структура симплекс-таблицы может быть следующей:

Таблица 2.1

Пример 1 симплексной таблицы

Базисные переменные	Небазисные переменные		Свободный член
	X_1	X_2	
X_3	a_{11}	a_{12}	b_1
X_4	a_{21}	a_{22}	b_2
X_5	a_{31}	a_{32}	b_3
X_6	a_{41}	a_{42}	b_4
f			

Кроме того, в таблице могут присутствовать столбцы с базисными переменными и вспомогательный столбец с расчетами (данные столбцы обязательными не являются). В этом случае отчетливо виден единичный базис (табл. 2.2).

Таблица 2.2

Пример 2 симплексной таблицы

Базисные переменные	Небазисные переменные		Базисные переменные				Свободный член
	X_1	X_2	X_3	X_4	X_5	X_6	
X_3	a_{11}	a_{12}	1	0	0	0	b_1
X_4	a_{21}	a_{22}	0	1	0	0	b_2
X_5	a_{31}	a_{32}	0	0	1	0	b_3
X_6	a_{41}	a_{42}	0	0	0	1	b_4
f							

После построения начальной симплекс-таблицы выполняются следующие этапы.

1. Проверить базисное решение на оптимальность. Решение оптимально, если в строке f присутствуют лишь неотрицательные элементы (в задаче на максимум) или неположительные элементы (в задаче на минимум). Если решение оптимально, то необходимо выписать ответ, иначе перейти к пункту 2.

2. Определить ведущий элемент. Для этого необходимо выбрать небазисный столбец, в котором находится наибольшая «неправильного» знака оценка функции f (отрицательная (на максимум) / положительная (на минимум)). Для определения нужной строки (т.е. переменной, которую будем выводить из базиса) необходимо разделить столбцы свободных членов на элементы найденного столбца. Если в числителе положительное число, а в знаменателе отрицательное, отношение считается равным бесконечности. Элемент, стоящий на пересечении найденного столбца и строки, содержащей минимальное значение вычисленных отношений, называется **ведущим.**

Если в числителе отрицательное число, а в знаменателе – положительное, то система не имеет решений.

Таким образом, фактически деление происходит только на положительные элементы ведущего столбца.

Таблица 2.3

Определение ведущего элемента

Базисные переменные	Небазисные переменные		Базисные переменные				Свободный член	
	X_1	**X_2**	X_3	X_4	X_5	X_6		
X_3	a_{11}	**a_{12}**	1	0	0	0	b_1	$\frac{b_1}{a_{12}}$
X_4	a_{21}	**a_{22}**	0	1	0	0	b_2	$\frac{b_2}{a_{22}}$
X_5	a_{31}	**a_{32}**	0	0	1	0	b_3	$\frac{b_3}{a_{32}}$
X_6	a_{41}	**a_{42}**	0	0	0	1	b_4	$\frac{b_4}{a_{42}}$
f	F_1	**F_2**						

В таблице 2.3 показано, что F_2 – наибольший по модулю отрицательный (для задачи на максимум) или положительный (для задачи на минимум) элемент. Следовательно, в базис вводим переменную x_2. Чтобы определить, какую переменную вывести из базиса, необходимо разделить столбец свободных членов на ведущий столбец (в данном случае, на столбец с x_2). Пусть минимальное значение будет у отношения b_3/a_{33}. Тогда из базиса будем удалять элемент x_3.

3. <u>Провести симплекс-преобразование</u>. Для этого необходимо пересчитать все элементы в симплекс-таблице по формуле:

$$НЭ = СЭ - \frac{A \cdot B}{ВедЭ}\,. \tag{2.6}$$

Здесь НЭ – новый элемент симплекс-таблицы; СЭ – старый элемент; ВедЭ – ведущий элемент; А и В - это элементы, образующие прямоугольник с элементами СЭ и ВедЭ.

Строка, содержащая базисный элемент, просто делится на данный элемент. Приведем фрагмент симплекс-таблицы.

Таблица 2.4

Фрагмент симплекс – таблицы

Базисные переменные	X_3	X_4	Свободный член
X_1	2	1	4
X_2	1	**2**	3
f	2	-1	-5

Пусть найден базисный элемент. Тогда в результате симплекс преобразования по формуле (2.1) получим:

Таблица 2.5

Результат симплекс-преобразования

Базисные переменные	X_3	X_4	Свободный член
X_1	$2-\frac{1\cdot 1}{2}$	$1-\frac{2\cdot 1}{2}$	$4-\frac{3\cdot 1}{2}$
X_2	$\frac{1}{2}$	$\frac{2}{2}$	$\frac{3}{2}$
f	$2-\frac{3\cdot(-1)}{2}$	$-1-\frac{3\cdot(-1)}{2}$	$-5-\frac{3\cdot(-1)}{2}$

Шаги 1, 2 и 3 повторяем до тех пор, пока последняя строка не будет удовлетворять условию оптимальности.

Рассмотрим работу приведенного выше алгоритма на следующем примере.

$$3x_1 + 2x_2 \to max$$
$$x_1 + 3x_2 \leq 15$$
$$x_1 + x_2 \leq 7$$
$$2x_1 + 1x_2 \leq 12$$
$$x_1 \geq 0, x_2 \geq 0.$$

1. Приведем систему к каноническому виду. Для этого добавим вспомогательные переменные x_3, x_4, x_5, x_6. Получим следующую задачу:

$$3x_1 + 2x_2 \to max$$
$$x_1 + 3x_2 + x_3 = 15$$
$$x_1 + x_2 + x_4 = 7$$
$$2x_1 + x_2 + x_5 = 12$$
$$x_1 \geq 0, x_2 \geq 0.$$

<u>Итерация 1</u>

Выберем произвольным образом базисное решение (т.е. выделим базисные и небазисные переменные). Поскольку 4 уравнения, необходимо 4 базисных переменных.

Перепишем систему в виде

$$3x_1 + 2x_2 \to max$$
$$x_1 + 3x_2 + \boldsymbol{x_3} = 15$$
$$x_1 + x_2 + \quad \boldsymbol{x_4} = 7$$
$$2x_1 + x_2 + \quad \boldsymbol{x_5} = 12$$
$$x_1 \geq 0, x_2 \geq 0.$$

Из данной системы видно, что, x_3, x_4 и x_5 будут базисными переменными, а x_1 и x_2 – небазисными (свободными).

Определим симплексную таблицу. Для этого:

- определим вектор коэффициентов (u_1, u_2 u_3) базисных переменных в целевой функции (в данном примере это вектор (0,0,0));

- определим вектор (v_1, v_2) небазисных переменных в целевой функции (в данном случае это вектор (1,2));

- каждая ячейка для оценки f представляет собой разность скалярного произведения вектора U на столбец, для которого вычисляется оценка и соответствующей компоненты вектора v.

Поскольку вектор U в данном случае нулевой, для первого столбца будем иметь значение -3, для второго -2 .

Таким образом, для рассматриваемого примера получится следующая симплекс-таблица.

Далее следуют столбцы с небазисными переменными.

Таблица 2.6

Исходная симплекс-таблица

Базисные переменные	Небазисные переменные		Базисные переменные			Свободный член
	X_1	X_2	X_3	X_4	X_5	
X_3	1	3	1	0	0	15
X_4	1	1	0	1	0	7
X_5	2	1	0	0	1	12
f	-3	-2				0

<u>Итерация 2</u>

Каждая итерация предназначена для вывода из базиса некоторой переменной и внесения некоторой небазисной переменной в базис. Это необходимо в силу того, что найденное решение не является оптимальным, следовательно, необходимо осуществлять поиск другого решения. При этом выбирается столбец с максимальным по модулю значением целевой функции, поскольку он в большей степени влияет на неоптимальность задачи.

Для перехода к следующей таблице найдём наибольшее (по модулю) из чисел оценок f: |-3| и |-2|. Это число 2. Поэтому ведущий столбец – тот столбец, в котором записано x_1.

Для определения ведущей строки находим минимум отношений свободных членов к элементам ведущего столбца, причём, если в числителе положительное число, а в знаменателе отрицательное, отношение считается равным бесконечности.

Будем иметь: $min(15,7,6) = 6$.

Следовательно, ведущей будет та строка, на которой этот минимум будет достигаться, т.е. строка со значением x_5. Смысл данной итерации заключается в том, что значение x_5 необходимо вывести из базиса, а значение x_2, наоборот, добавить в базис.

Ведущим элементом является элемент 2 (т.е. тот элемент, который стоит на пересечении ведущего столбца и ведущей строки).

Выполним симплекс-преобразование, как это показано в таблице 2.5. Строку с базисным элементом x_5 просто разделим на ведущий элемент (2), а к остальным применим формулу 2.6.

Прокомментируем данный процесс. Для получения значения, стоящего на пересечении строки x_3 и столбца x_2, будем строить следующий прямоугольник (табл. 2.7).

Таблица 2.7

Прямоугольник для получения результатов симплекс-преобразования

Базисные переменные	Небазисные переменные		Базисные переменные			Свободный член
	X_1	X_2	X_3	X_4	X_5	
X_3	1	3	1	0	0	15
X_4	1	1	0	1	0	7
X_5	2	1	0	0	1	12
f	-3	-2				0

$$a_{32}^{\text{нов}} = 3 - \frac{1 \cdot 1}{2} = 2\frac{1}{2}.$$

Значения элементов a_{33} и a_{34} не изменятся, поскольку в прямоугольнике, который получается соединением ведущего и данных элементов присутствует нулевое значение. Это очевидно еще и потому, что соответствующие элементы на данном этапе являются базисными. В последнем столбце (который будет выведен из базиса) значение изменится:

$$a_{35}^{\text{нов}} = 0 - \frac{1 \cdot 1}{2} = -\frac{1}{2}.$$

Аналогично вычисляем остальные поля, каждый раз образовывая прямоугольник, как это показано в табл. 2.7. В результате получим следующие значения новой симплекс-таблицы (табл. 2.8).

Таблица 2.8

Результат симплекс-преобразования

Базисные переменные	Небазисные переменные		Базисные переменные			Свободный член
	X_1	X_2	X_3	X_4	X_5	
X_3	0	$2\frac{1}{2}$	1	0	$-\frac{1}{2}$	9
X_4	0	$\frac{1}{2}$	0	1	$-\frac{1}{2}$	1
X_1	1	$\frac{1}{2}$	0	0	$\frac{1}{2}$	6
f	0	$-\frac{1}{2}$	0	0	$1\frac{1}{2}$	18

Проверим полученный план на оптимальность. Он не является оптимальным, поскольку среди индексных значений (в последнем столбце) присутствуют отрицательные значения. Следовательно, необходимо выполнить аналогичные действия: найти ведущий элемент и выполнить симплекс-преобразование.

<u>Итерация 3</u>

Ведущим столбцом является x_2, поскольку среди f у этого столбца единственное отрицательное значение. Разделим свободные члены на элементы ведущего столбца. Получим следующие значения:

$$9 : 2\frac{1}{2} = 3\frac{3}{5},$$
$$1 : \frac{1}{2} = 2,$$
$$6 : \frac{1}{2} = 12.$$

Выводим из базиса строку x_4; ведущий элемент равен ½.

Выполняем аналогичным образом симплекс-преобразование. Получаем следующую симплекс-таблицу (табл. 2.9).

Таблица 2.9

Результат симплекс-преобразования

Базисные переменные	Небазисные переменные		Базисные переменные			Свободный член
	X_1	X_2	X_3	X_4	X_5	
X_3	0	0	1	-5	2	4
X_2	0	1	0	2	-1	2
X_1	1	0	0	-1	1	5
f	0	0	0	1	1	19

Проверяем текущий план на оптимальность. Поскольку среди элементов индексной строки нет отрицательных элементов, план оптимален. Выпишем решение. Для этого смотрим на столбец свободных членов: $x_3=4$; $x_2=2$; $x_1=5$. Итоговое значение целевой функции равно 19. Проверим это:

$F(x)=5\cdot 3+2\cdot 2=19$.

Можно также проверить, что все ограничения имеют место.

Рассмотрим процесс решения задачи средствами Excel. Зарезервируем ячейки B_2 и C_2 под искомые переменные x_1 и x_2 (на начальном этапе можно ввести в соответствующие ячейки любые значения, например, нулевые). В ячейке B_4 будет располагаться целевая функция, которую необходимо задать через значения x_1 и x_2 (см. *рис. 2.1*).

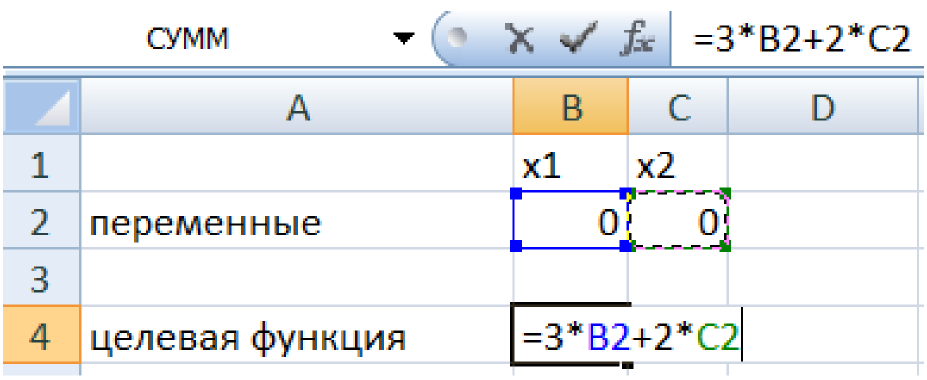

Рис. 2.1. Окно Excel с заготовкой решения

В столбец B, начиная с шестой строки будем вводить ограничения, а в столбец B введем ограничивающие значения (*рис. 2.2*).

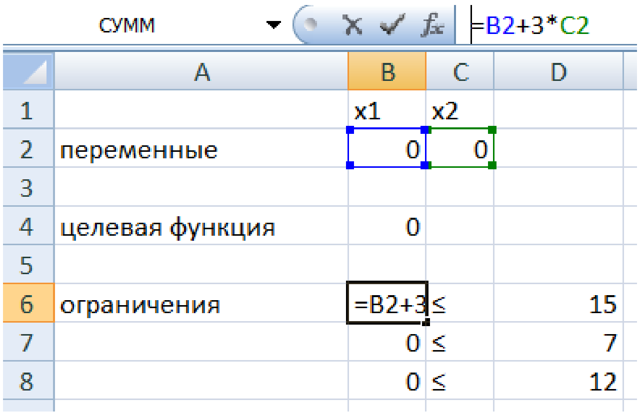

Рис. 2.2. Ограничения задачи

Встанем на ячейку целевой функции и запустим команду меню «Данные/ Поиск решения» (если этой команды нет, необходимо ее установить с помощью соответствующей надстройки Office| Параметры Excel| Надстройки). На *рис. 2.3* показаны надстройки с уже активным сервисом «Поиск решения». Если его нет, то он будет располагаться в неактивных надстройках приложения, ориентировочно там, где указывает стрелка.

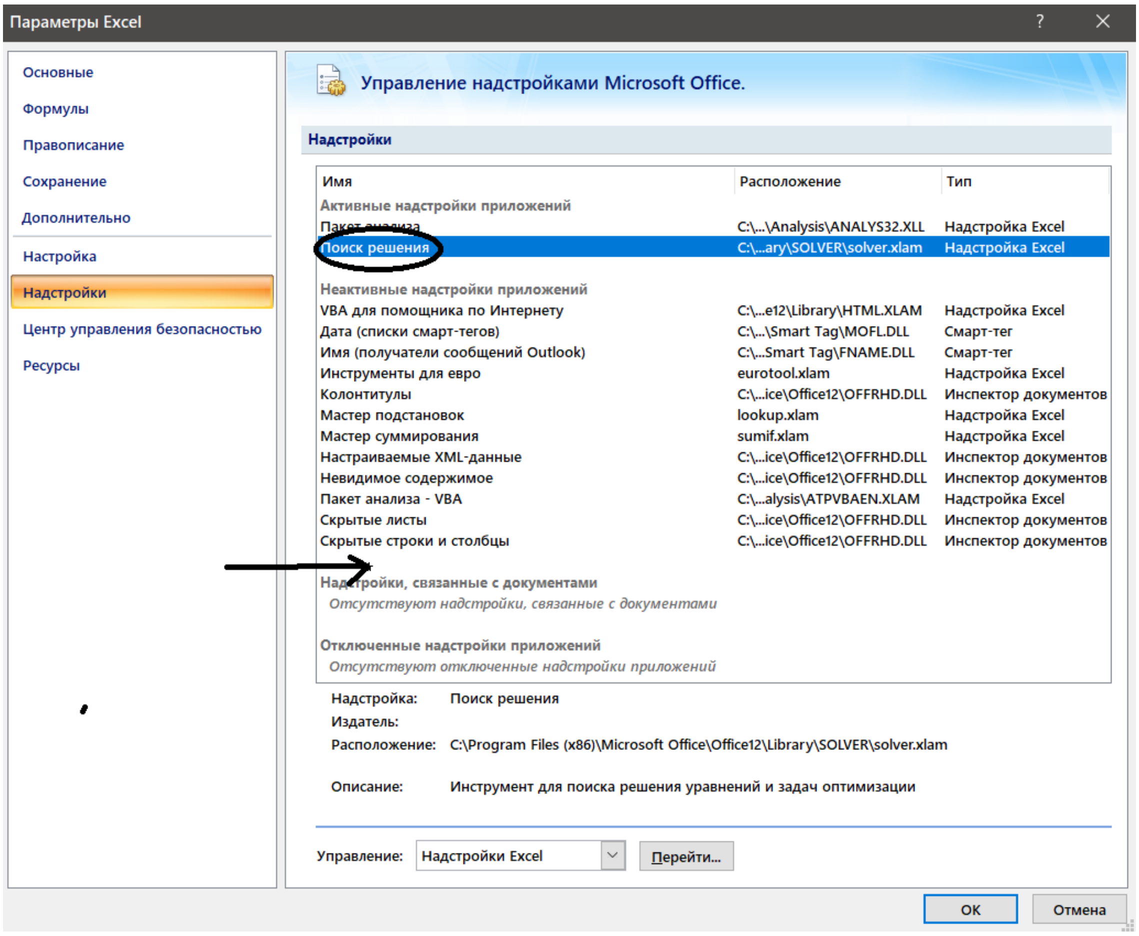

Рис. 2.3. Надстройки Excel

В открывшемся диалоговом окне необходимо указать:

- ячейку с целевой функцией (по умолчанию содержит ячейку, из которой был вызван поиск решения);
- аргументы целевой функции (указываются в поле «Изменяя ячейки»);
- ограничения.

Для рассматриваемого примера получим следующий вид окна поиска решения (*рис. 2.4*).

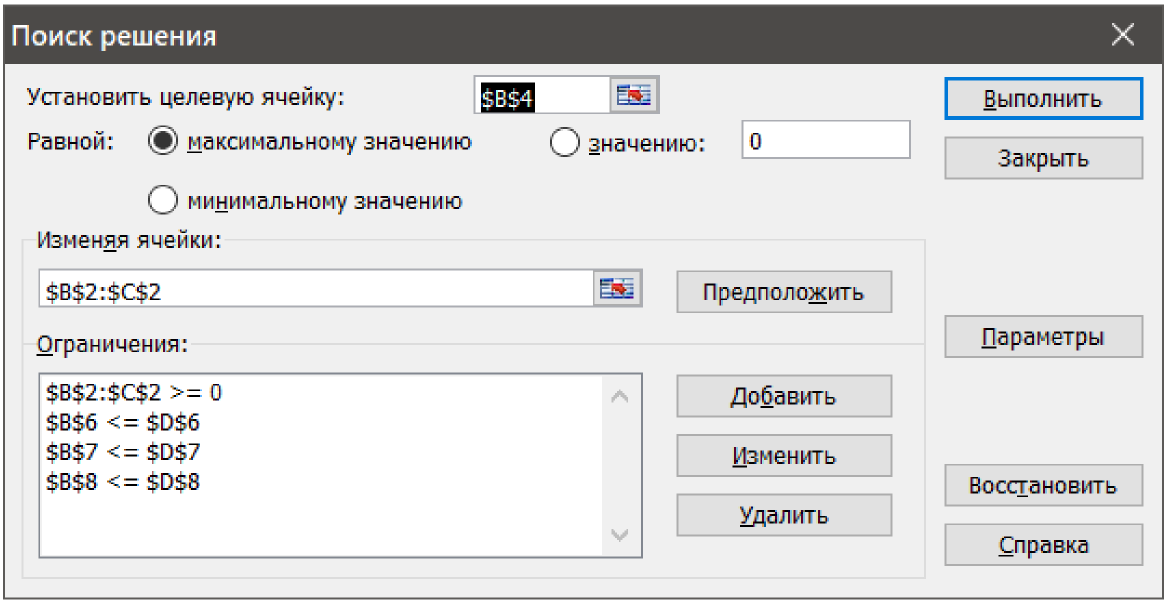

Рис. 2.4. Поиск решения

После этого необходимо нажать на кнопку «Выполнить». Получим результаты, представленные на *рис. 2.5*.

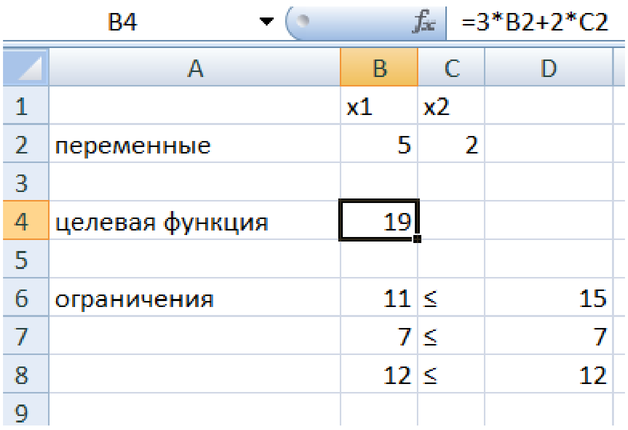

Рис. 2.5. Результаты решения задачи

Как можно видеть из данного рисунка, результаты расчета Excel совпадают с решением задачи симплекс-методом.

2.2. Задание к работе

Найти симплекс методом оптимального решения и проверить результат с помощью среды Excel.

Отчет должен содержать:

- вариант задания;
- решение задачи симплекс-методом;
- решение задачи средствами Excel;
- выводы.

2.3. Варианты заданий

Таблица 2.10

Варианты задания на лабораторную работу № 2

1. $x_1 + 4x_2 \to max$ $x_1 + 2x_2 \geq 2$ $-x_1 + x_2 \leq 3$ $x_1 + x_2 \leq 7$ $2x_1 + x_2 \leq 10$ $x_1 \geq 0, x_2 \geq 0$	2. $x_1 + x_2 \to max$ $x_1 + 3x_2 \geq 4$ $3x_1 + x_2 \geq 4$ $4x_1 + x_2 \leq 24$ $x_1 + 4x_2 \leq 24$ $x_1 \geq 0, x_2 \geq 0$	3. $2x_1 - x_2 \to max$ $x_1 + 3x_2 \geq 7$ $2x_1 - x_2 \geq 0$ $6x_1 + 10x_2 \leq 78$ $x_1 - 2x_2 \leq 2$ $x_1 \geq 0, x_2 \geq 0$
4. $x_1 - 4x_2 \to max$ $x_1 + 2x_2 \geq 4$ $-x_1 + 2x_2 \leq 10$ $x_2 \leq 6$ $3x_1 + 2x_2 \leq 24$ $x_1 \geq 0, x_2 \geq 0$	5. $5x_1 + x_2 \to max$ $x_1 - 3x_2 \leq 1$ $x_1 + x_2 \geq 5$ $x_2 \leq 8$ $2x_1 + x_2 \leq 16$ $x_1 \geq 0, x_2 \geq 0$	6. $2x_1 + x_2 \to max$ $-x_1 + 2x_2 \geq 0$ $x_1 + 2x_2 \geq 4$ $-x_1 + x_2 \leq 5$ $x_1 + x_2 \leq 9$ $x_1 \geq 0, x_2 \geq 0$
7. $-x_1 + x_2 \to min$ $x_1 + x_2 \geq 2$ $x_1 - 2x_2 \leq 3$ $-2x_1 + x_2 \leq 4$ $2x_1 + 3x_2 \leq 20$ $x_1 \geq 0, x_2 \geq 0$	8. $x_1 + 3x_2 \to max$ $x_1 + 4x_2 \geq 4$ $-2x_1 + x_2 \leq 6$ $x_2 \leq 10$ $x_1 + 3x_2 \leq 100$ $x_1 \geq 0, x_2 \geq 0$	9. $-x_1 - 4x_2 \to min$ $x_1 + 2x_2 \geq 4$ $5x_1 + x_2 \geq 11$ $-x_1 + 4x_2 \leq 23$ $x_1 - 2x_2 \leq 4$ $x_1 \geq 0, x_2 \geq 0$
10. $x_1 - 8x_2 \to min$ $2x_1 + x_2 \geq 5$ $x_1 + 4x_2 \geq 6$ $2x_1 - x_2 \leq 16$ $2x_1 + 3x_2 \leq 32$ $x_1 \geq 0, x_2 \geq 0$	11. $4x_1 + x_2 \to max$ $x_1 + 4x_2 \geq 6$ $-x_1 + 4x_2 \leq 28$ $2x_1 + 3x_2 \leq 32$ $-2x_1 + 5x_2 \leq 10$ $x_1 \geq 0, x_2 \geq 0$	12. $5x_1 + 3x_2 \to max$ $-3x_1 + 2x_2 \geq 0$ $-x_1 + 6x_2 \leq 3$ $2x_1 + 3x_2 \leq 7$ $4x_1 + x_2 \leq 10$ $x_1 \geq 0, x_2 \geq 0$

ЛАБОРАТОРНАЯ РАБОТА № 3

ЧИСЛЕННЫЕ МЕТОДЫ НУЛЕВОГО ПОРЯДКА РЕШЕНИЯ ЗАДАЧ НА БЕЗУСЛОВНЫЙ ЭКСТРЕМУМ

Цель работы: изучение теоретических основ и получение навыков практической реализации методов Хука-Дживса и Нелдера-Мида решения задач на безусловный экстремум.

3.1. Краткие теоретические сведения

3.1.1. Метод Хука-Дживса

Рассматривается задача на безусловный экстремум функции f, зависящей от нескольких переменных $x_1,\ldots,x_n$:

$$f(x_1, x_2, \ldots, x_n) \to min. \tag{3.1}$$

Метод Хука-Дживса относится к методам прямого поиска для определения минимума функции n переменных (метод нулевого порядка). Методы прямого поиска являются методами, в которых используются только значения функции.

Метод конфигураций (метод Хука-Дживса) представляет собой комбинацию исследующего поиска с циклическим изменением переменных и ускоряющего поиска по образцу. Исследующий поиск ориентирован на выявление локального поведения целевой функции и определение направления ее убывания вдоль «оврагов». Полученная информация используется при поиске по образцу при движении вдоль «оврагов».

Поиск методом Хука-Дживса состоит из последовательности шагов исследующего поиска вокруг базисной точки, за которой в случае успеха следует поиск по образцу.

Метод состоит из следующих шагов:

А. Выбрать начальную базисную точку $x^0 = (x_1^0, x_2^0, \ldots, x_n^0)$ и шаг $(\Delta x_1, \ldots, \Delta x_n)$ для каждого направления (в ряде случаев $\Delta x_1 = \cdots = \Delta x_n$).

Б. Вычислить $\boldsymbol{f(x)}$ в базисной точке $\mathbf{x}^{\mathbf{0}}$ и исследовать локальное поведение функции $\boldsymbol{f(x)}$с целью получения новой базисной точки. Новая базисная точка должна обеспечивать уменьшение функции $\boldsymbol{f(x)}$. Функция $\boldsymbol{f(x)}$ в базисной точке $\mathbf{x}^{\mathbf{0}}$ исследуется следующим образом.

1. Вычисляют значение функции $\boldsymbol{f(\mathbf{x}^{0})}$ в старой базисной точке $\mathbf{x}^{\mathbf{0}}$.

2. На каждой итерации i каждая переменная j (по очереди) изменяется прибавлением длины шага.

<u>Если</u>
$f\left(x_1^i, x_2^i, \ldots, x_{j-1}^i, x_j^i + \Delta x_j, x_{j+1}^i, \ldots, x_n^i\right) < f\left(x_1^i, x_2^i, \ldots, x_{j-1}^i, x_j^i, x_{j+1}^i, \ldots, x_n^i\right)$,
<u>то</u>

$$x^i_{\text{нов}} = \left(x^i_1, x^i_2, \dots, x^i_{j-1}, x^i_j + \Delta x_j, x^i_{j+1}, \dots, x^i_n\right).$$

Иначе если

$f\left(x^i_1, x^i_2, \dots, x^i_{j-1}, x^i_j - \Delta x_j, x^i_{j+1}, \dots, x^i_n\right) < f\left(x^i_1, x^i_2, \dots, x^i_{j-1}, x^i_j, x^i_{j+1}, \dots, x^i_n\right),$

то

$$x^i_{\text{нов}} = \left(x^i_1, x^i_2, \dots, x^i_{j-1}, x^i_j - \Delta x_j, x^i_{j+1}, \dots, x^i_n\right).$$

Перейти к следующей переменной j.

Графически проиллюстрировать данный поиск можно с помощью следующего рисунка (*рис. 3.1*).

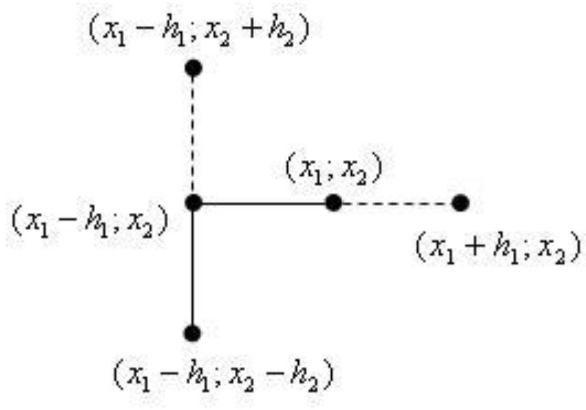

Рис. 3.1. Иллюстрация работы метода Хука-Дживса. Иследующий поиск

3. Проверка на соответствие новой и старой точки.

Если

$$x^i_{\text{нов}} = x^i,$$

т. е. уменьшение функции не было достигнуто, то исследование повторяется вокруг той же базисной точки x^i, но с уменьшенной длиной шага. На практике удовлетворительным является уменьшение шага (шагов) в десять раз от начальной длины.

Если

$$x^i_{\text{нов}} \neq x^i,$$

то производится *поиск по образцу*. Поиск по образцу заключается в движении по направлению от старого базиса к новому.

В. При поиске по образцу используется информация, полученная в процессе исследования, и минимизация функции завершается поиском в направлении, заданном образцом. Поиск по образцу производится следующим образом:

В.1. Разумно двигаться из новой базисной точки $x^i_{\text{нов}}$ в направлении $x^i_{\text{нов}}$ – x^i, поскольку поиск в этом направлении уже привел к уменьшению значения функции. Вычисляют координаты новой точки в направлении $x^i_{\text{нов}} - x^i$ (точку образца, ускоряющий множитель выбран равным 2). В общем случае

$$x^i_{\text{нов\_обр}} = x^i + 2(x^i_{\text{нов}} - x^i)\ .$$

В.2. Заменяют значения в базисной точке x^i на значения в $x^i_{\text{нов}}$, значения во второй базисной точке заменяют на $x^i_{\text{нов\_обр}}$, то есть:

$b_i = b_{i+1}$, а $b_{i+1} = P_i$ для всех i. Продолжают исследования новой базисной точки (переход на этап Б2).

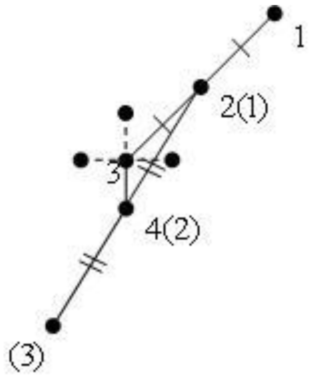

Рис. 3.2. Иллюстрация работы метода Хука-Дживса.
Поиск по образцу

Если исследующий поиск привел к тому, что

$$\boldsymbol{f}(x^i_{\text{нов}}) > f(x^i),$$

то следует возвратиться в старую базисную точку и уменьшить шаги $\boldsymbol{\Delta x_i}$, затем перейти на этап Б2. Другими словами $x^i_{\text{нов}} = x^i$для всех i; уменьшить шаг, например, в 10 раз; перейти на этап Б2.

Г. Процесс поиска решения заканчивают, когда длина шага (длины шагов) будет меньше заданного малого значения.

Недостаток метода Хука-Дживса состоит в том, что в случае сильно вытянутых, изогнутых или обладающих острыми углами линий уровня целевой функции он может оказаться неспособным обеспечить продвижение к точке минимума. Представим это на следующем примере.

Пример.

$F(x1,x2) = 100 \cdot (x2 - x1^2)^2 + (1 - x1)^2$

Точка минимума – (1,1).

Шаг 1. Задаем начальную точку x^0=(-1,-2); приращения (шаги): Δx=(1,1); коэффициент уменьшения шага: α = 2; ε=0.1.

Вычислим значение функции в т. $x^0=(-1,-2)^T$: $f(x_0) = 904$.

Итерация №0.

Шаг №2. Исследующий поиск. Фиксируя переменную x_2 = -2, дадим приращение x_1:

x_1=-1 + 1 = 0,

f(0;-2) = 401 < 904.

Фиксируя переменную $x_1 = 0$, дадим приращение x_2:

x_2=-2 + 1 = -1,

f(0;-1) = 101 < 401.

Значение ЦФ в пробной точке меньше значения ЦФ в исходной точке, поэтому шаг поиска успешный. **Базовая точка $x^1=(0;-1)^T$.** Переходим к поиску по образцу.

Шаг 3. Поиск по образцу. Осуществляется шаг из полученной базовой точки вдоль прямой, соединяющей эту точку с предыдущей базовой. Новая точка образца определяется по формуле: $x_p^k=x^k+(x^k-x^{k-1})$ $x^2=x^1+(x^1-x^0) = (1;0)$

$f(x^2)$=100.00.

Оставляем точку (1,0) (**временная базовая точка –** точка (1,0)).

Шаг №4 Исследующий поиск (после поиска по образцу).

Фиксируя переменную $x_2 = 0$, дадим приращение x_1:

x_1=1 + 1 = 2

f(2;0) = 1601 > 100

x_1=1 – 1 = 0

f(0;0) = 1.

Фиксируем $x_1 = 0$, даем приращение на x_2:

x_2=0 + 1 = 1

f(0;1) =101.

X_2=0 – 1 = -1

f(0;-1) =101.

По x_2 приращения нет. Получаем точку $x^k = (0;0)$.

Значение целевой функции в пробной точке меньше значения целевой функции в исходной точке, поэтому шаг поиска успешный. Базовая точка $x^2=(0;0)^T$. Переходим к поиску по образцу. Предыдущая базовая точка (0,-1), текущая (0,0).

Шаг 5. Поиск по образцу.

Новая точка образца определяется по формуле: $x_p^k=x^k+(x^k-x^{k-1})$

x_1=2*0-0 =0

x_2=2*0-(-1) = 1

$f(x^3)=101 \geq 1$, поэтому продолжаем исследовательский поиск. Точка (0,0).

Итерация №2. **Базовая Точка (0,0)**

Исследующий поиск.

Фиксируя переменную $x_2 = 0$, дадим приращение x_1:

x_1=0+ 1 = 1

f(1;0) = 100 > 1

x_1=0- 1 =- 1

f(-1;0) = 104 > 1

Фиксируя переменную $x_1 = 0$, дадим приращение x_2:

x_2=0 + 1 = 1
f(0;1) = 101 > 1
X_2=0 – 1 = -1
f(0;-1) = 101 > 1
Исследующий поиск не был удачным.
Проверка на окончание поиска.

$$|x| = \sqrt{1^2 + 1^2} = 1.4142,$$

$$|\Delta x| = 1.4142 > 0.1.$$

Неравенство не выполняется, поэтому уменьшаем приращение.
Δx_1 = 1 / 2 = 0.5,
Δx_2 = 1 / 2 = 0.5.

Итерация №3. **Точка (0,0)**

Исследующий поиск.
Фиксируя переменную x_2 = 1, дадим приращение x_1:
x_1=0 + 0.5 = 0.5
f(0.5;0) = 6.5 > 1
x_1=0 – 0.5 = -0.5
f(-0.5;0) = 8.5 > 1.
Фиксируя переменную x_1 = 0, дадим приращение x_2:
x_2=0 + 0.5 = 0.5
f(0;0.5) = 26 > 1
x_2=0 – 0.5 = -0.5
f(0;-0.5) = 26 > 1.
Исследующий поиск не был удачным.
Проверка на окончание поиска.
$|\Delta x|$ = 0.7071 > 0.1
Неравенство не выполняется, поэтому уменьшаем приращение.
Δx_1 = 0.5 / 2 = 0.25
Δx_2 = 0.5 / 2 = 0.25
и т.д.

3.1.2.Метод Нелдера-Мида

Метод Нелдера-Мида, также известный как метод деформируемого многогранника и симплекс-метод, – метод безусловной оптимизации функции от нескольких переменных, не использующий производной функции.

Суть метода заключается в последовательном перемещении и деформировании симплекса вокруг точки экстремума.

Метод находит как глобальные, так и локальные экстремумы и может «застрять» в одном из них. Если всё же требуется найти глобальный экстремум, можно пробовать выбирать другой начальный симплекс.

Пусть требуется найти безусловный минимум функции n переменных f(x_1, x_2,…, x_n). Предполагается, что серьёзных ограничений на область определения функции нет, то есть функция определена во всех встречающихся точках.

Параметрами метода являются:

- коэффициент отражения $\alpha > 0$, обычно выбирается равным 1;
- коэффициент сжатия $\beta > 0$, обычно выбирается равным 0,5;
- коэффициент растяжения $\gamma > 0$, обычно выбирается равным 2;
- точность алгоритма ε.

Подготовка. Вначале задают n + 1 точки $x = (x_1, x_2,\ldots, x_n, x_{n+1})$, образующие симплекс n-мерного пространства. В этих точках вычисляются значения функции: $f_1=f(x_1)$, $f_2=f(x_2)$, …, $f_{n+1}=f(x_{n+1})$.

Дальнейший план действий.

1. Сортировка. Из вершин симплекса выбирают три точки: x_h с наибольшим (из выбранных) значением функции f_h, x_g со следующим по величине значением f_g и x_l с наименьшим значением функции f_l. Целью дальнейших манипуляций будет уменьшение по крайней мере f_h.

2. Определяют центр тяжести всех точек за исключением x_h:

$$x_c = \frac{1}{n}\sum_{\substack{i=1 \\ i \neq h}}^{n+1} x_i \ . \qquad (3.2)$$

Вычисляют $f_c = f(x_c)$.

Отражение. Отражают точку x_h относительно x_c с коэффициентом α (при $\alpha = 1$ это будет центральная симметрия, в общем случае — гомотетия (преобразование подобия)); получают точку x_r и вычисляют в ней функцию: $f_r = f(x_r)$. Координаты новой точки вычисляются по формуле:

$$x_r = (1 + \alpha)x_c - \alpha x_h. \qquad (3.3)$$

Операция отражения на примере треугольного симплекса приведена на рис. 3.3.

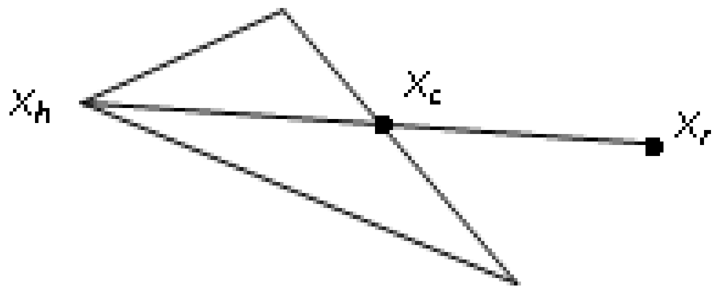

Рис. 3.3. Операция отражения на примере треугольного симплекса

4. Далее определяют, насколько удалось уменьшить функцию, определяют место f_r в ряду f_h ,f_g, f_l:

а) если $f_r < f_l$, то направление выбрано удачное и можно попробовать увеличить шаг — производят растяжение. Новая точка

$$x_b = (1 - \gamma)x_c + \gamma x_r \qquad (3.4)$$

и значение функции $f_b = f(x_b)$ (пример приведен на рис. 3.4).

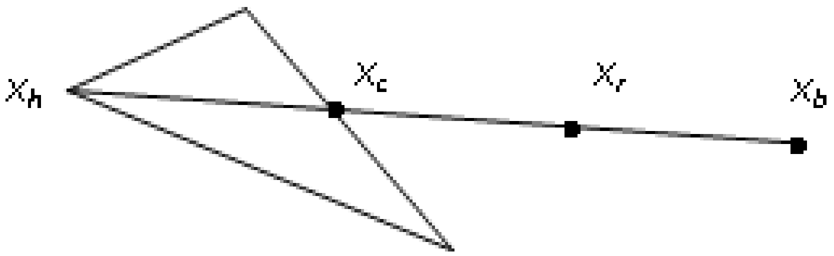

Рис. 3.4. Операция растяжения на примере треугольного симплекса

– если $f_b < f_l$, то можно расширить симплекс до этой точки: заменяют точку x_h на x_b и заканчивают итерацию (на шаг 8);

– если $f_b > f_l$, то переместились слишком далеко: в набор выбирают x_r (опять вместо x_h) и заканчивают итерацию (на шаг 8);

б) если $f_l < f_r < f_g$, то выбор точки уже неплохой (новая лучше двух прежних). Заменяют точку x_h на x_r и переходят на шаг 8;

в) если $f_h > f_r > f_g$, то меняют обозначения x_r, x_h (и соответствующие значения функции) местами и переходят на шаг 5.

г) если $f_r > f_h$, то просто переходят на следующий шаг 5.

В результате (возможно, после переобозначения) $f_r > f_h > f_g > f_l$.

Сжатие. Строят точку

$$x_s = \beta x_h + (1 - \beta)x_c. \qquad (3.5)$$

Вычисляют в ней значение f_s (пример приведен на рис. 3.5 и 3.6).

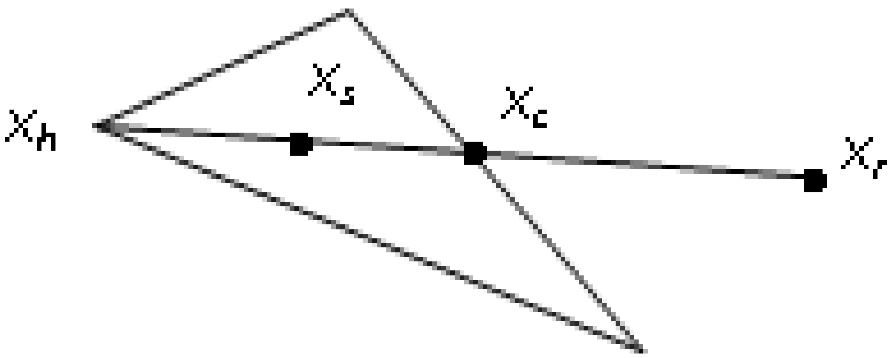

Рис. 3.5. Операция сжатия на примере треугольного симплекса (переход с шага 4г)

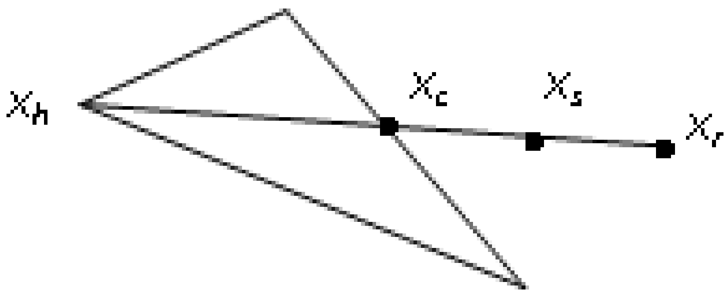

Рис. 3.6. Операция сжатия на примере треугольного симплекса (переход с шага 4в)

6. Если $f_s < f_h$, то заменяют точку x_h на x_s и переходят на шаг 8.

7. Если $f_s > f_h$, то выполняют <u>глобальное сжатие</u> симплекса — гомотетию к точке с наименьшим значением x_l: $x_i = x_l + (x_i - x_l)/2$ для всех требуемых точек x_i.

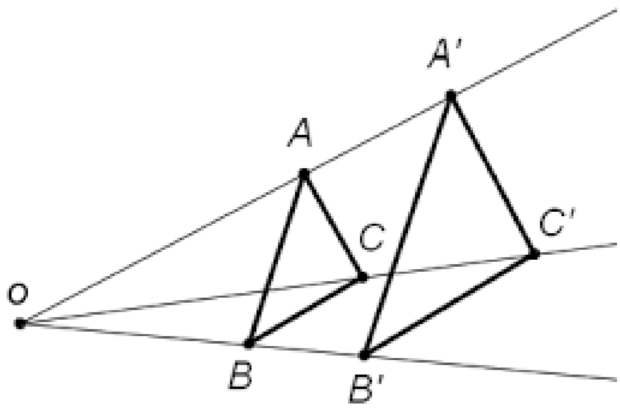

Рис. 3.7. Глобальное сжатие

8. Последний шаг — проверка сходимости. Может выполняться по-разному, например, оценкой дисперсии набора точек. Суть проверки заключается в том, чтобы проверить взаимную близость полученных вершин симплекса, что предполагает и близость их к искомому минимуму. Если требуемая точность ещё не достигнута, можно продолжить итерации с шага 1.

Пример работы алгоритма Нелдера-Мида

Дано:

$$f(x,y) = x^2 + xy + y^2 - 6x - 9y$$

3 начальные точки V1=(0,0); V2=(1,0); V3=(0,1).

<u>Шаг 1</u>

Вычисляем значения функции в точках:

f(V1)=0; f(V2)=-5; f(V3)=-8.

1.Сортировка.

Переобозначим точки следующим образом:

x_h =V1; x_g =V2; x_l =V3.

2.Определение центра тяжести.

$$x_c = \frac{x_g + x_1}{2} = \left(\frac{1+0}{2}, \frac{0+1}{2}\right) = \left(\frac{1}{2}, \frac{1}{2}\right).$$

3.Отражение

$x_r = (1+\alpha)\, x_c - \alpha x_h$

$$\alpha = 1$$

$x_r = 2\, x_c - x_h = 2\left(\frac{1}{2}, \frac{1}{2}\right) - (0{,}0) = (1{,}1)$

$$f(x_r) = -12.$$

Проверка: $f(x_r)$<$f(x_1)$, следовательно, направление выбрано удачно, увеличиваем отрезок (операция растяжения):

$x_b = (1-\gamma)\, x_c + \gamma x_r$

$\gamma = 2.$

$x_b = -x_c + 2x_r$=- $\left(\frac{1}{2}, \frac{1}{2}\right)$+2(1,1)=$\left(\frac{3}{2}, \frac{3}{2}\right)$,

$$f(x_b) = -15.75.$$

Получили новые вершины V1=$\left(\frac{3}{2}, \frac{3}{2}\right)$ V2=(1,0); V3=(0,1).

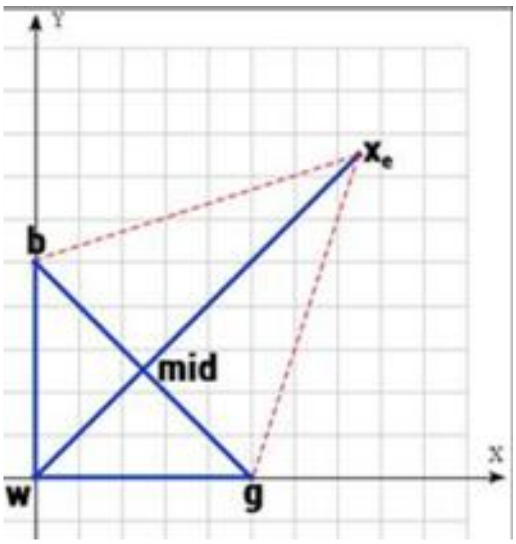

Рис. 3.8. Иллюстрация работа шага 1

Таблица 3.1

Результаты шага 1

Значение функции в лучшей точке	Значение функции в предпоследней точке	Значение функции в худшей точке
f($\frac{3}{2}$,$\frac{3}{2}$) = -15.75	f(0,1)=-8	f(1,0)=-5

Шаг 2

1. Сортировка. Переобозначим вершины: x_h=(1,0); x_g=(0,1); x_1=(1.5, 1.5)
2. Определение центра тяжести:
3.

$$x_c = \frac{x_g + x_1}{2} = \left(\frac{0+1.5}{2}, \frac{1+1.5}{2}\right) = (0.75, 1.25).$$

4. Отражение:

$$x_r = 2x_c - x_h = 2(0.75, 1.25) - (1,0) = (0.5, 2.5)$$
$$f(x_r) = -17.75.$$

5. Проверка: $f(x_r)$<$f(x_1)$, следовательно, направление выбрано удачно, увеличиваем отрезок (операция растяжения):

$$x_b = -x_c + 2x_r = 2(0.5, 2.5) - (0.75, 1.25) = (0.25, 3.75)$$
$$f(x_b) = -20.187.$$

6. Проверка: $f(x_b)$<$f(x_1)$, следовательно, точка (0.25,3,75) наилучшая, она будет являться заменой точке x_h.
7. Получили новые вершины V1=$\left(\frac{3}{2}, \frac{3}{2}\right)$ V2=(1,0); V3=(0.25, 0.75).

Таблица 3.2

Результаты шага 2

Значение функции в лучшей точке	Значение функции в предпоследней точке	Значение функции в худшей точке
F(0.25, 3,75)=-20.1875	f($\frac{3}{2}, \frac{3}{2}$) = -15.75	f(0,1)=-8

Шаг 3

1. Сортировка. Переобозначим вершины: x_h=(0,1); x_g=(1.5,1.5); x_1=(0.25, 3.75)
2. Определение центра тяжести:

$$x_c = \frac{x_g + x_1}{2} = \left(\frac{0.25+1.5}{2}, \frac{3.75+1.5}{2}\right) = (0.875, 2.625).$$

3. Отражение:

$$x_r = 2x_c - x_h = 2(0.875, 2.625) - (0,1) = (1.75, 4.25)$$
$$f(x_r) = -20.1875.$$

4. Проверка: $f(x_r)$<$f(x_1)$. Данное условие не выполняется ($f(x_r)$=$f(x_1) = -20.1875$).
5. Проверяем, лучше ли точка x_r, чем точки x_g и x_h. Исходя из табл. 3.2,

$$f(x_r) > f(x_g) > f(x_\mathrm{h})$$

Следовательно, меняем худшую точку на x_r. Результаты данного шага представлены в табл.3.3.

Таблица 3.3

Результаты шага 3

Значение функции в лучшей точке	Значение функции в предпоследней точке	Значение функции в худшей точке
F(0.25, 3,75)=-20.1875	F(1.75,4.25)=-20.1875	$f(\frac{3}{2},\frac{3}{2})$ = -15.75

Шаг 4

1. Сортировка. Переобозначим вершины: x_h=(1.5,1.5); x_g=(1.75, 4.25); x_1=(0.25,3.75).
2. Определение центра тяжести:

$$x_c = \frac{x_g+x_1}{2} = (1,4).$$

3. Отражение:

$$x_r = 2x_c - x_h = 2(1,4) - (1.5,1.5) = (0.5,6.5)$$
$$f(x_r) = -15.75.$$

4. Проверка: $f(x_r)<f(x_1)$. Условие не выполняется.
5. Проверка: $f_l < f_r < f_g$. Условие не выполняется.
6. Проверка: $f_h > f_r > f_g$. Условие не выполняется.
7. Проверка: $f_r > f_h$. Условие выполняется, переходим к операции сжатия. Для этого находим точку

$$x_s = \beta x_h + (1-\beta)x_c$$
$$\beta = 0.5$$
$$x_s = 0.5x_h + 0.5x_c = (1.25,2.75).$$

8. Вычисляем $f(x_s)$=-19.68.
9. Проверка $f_s < f_h$, следовательно, меняем x_h на x_s.

Таблица 3.4

Результаты шага 4

Значение функции в лучшей точке	Значение функции в предпоследней точке	Значение функции в худшей точке
F(1.75,4.25)=-20.1875	F(0.25, 3,75)=-20.1875	f(1.25,2.75)=-19.68

Шаг 5

1. Сортировка. Переобозначим вершины: x_h=(1.25,2.75); x_g=(0.25,3.75); x_1=(1.75, 4.25).
2. Определение центра тяжести:

$$x_c = \frac{x_g+x_1}{2} = (1,4).$$

3. Отражение:

$$x_r = 2x_c - x_h = 2(1,4) - (1.25,2.75) = (0.75,5.25)$$

$f(x_r) = -19.6875.$

4. Проверка: $f(x_r)<f(x_1)$. Условие не выполняется.

5. Проверка $f(x_r)<f(x_g)$. Условие не выполняется. Выполняем операцию сжатия относительно наихудшей точки:

$x_s = 0.5x_h + 0.5x_c=(0.625,1.375)+(0.5,2)=(1.125,3.375)$

$$x_s = 0.5x_h + 0.5x_c = (0.625,1.375) + (0.5,2) = (1.125,3.375)$$

f(1.125,3.375)=-20.6719.

6. Проверка $f_s < f_h$. Условие выполняется, следовательно, меняем x_h на x_s. Результаты пятого шага представлены в табл. 3.5.

Таблица 3.5

Результаты шага 5

Значение функции в лучшей точке	Значение функции в предпоследней точке	Значение функции в худшей точке
f(1.125,3.375)=-20.6719	F(1.75,4.25)=-20.1875	F(0.25,3.75)=-20.1875

Шаг 6

1. Сортировка. Переобозначим вершины: x_h=(0.25,3.75); x_g=(1.75,4.25); x_1=(1.125, 3.375).

2. Определение центра тяжести:

$$x_c = \frac{x_g+x_1}{2} = (1.4375,3.8125).$$

3. Отражение:

$$x_r = 2x_c - x_h = (2.625,3.875),$$
$$f(x_r) = -18.5469.$$

4. Проверка: $f(x_r)<f(x_1)$. Условие не выполняется.

5. Проверка: $f_h > f_r > f_g$. Условие не выполняется.

6. Проверка: $f_r > f_h$. Условие выполняется. Переходим к операции сжатия. Для этого формируем точку

$$x_s = 0.5x_h + 0.5x_c = (2.03125,3.84375),$$
$$f(x_s) = -20.0732.$$

Это наихудшее значение. Следовательно, выполняем глобальное сжатие в направлении значения x_1. В частности, приближаем в два раза точки (1.75,4.25) и (0.25,3.75) к точке (1.125,3.375).

Точка (1.75,4.25). Первая координата:

(1.75+1.125)/2=1.4375.

Вторая координата:

(4.25+3.375)/2=3.8125.
Точка (0.25,3.75). Первая координата:
(0.25+1.125)/2=0.6875.
Вторая координата:
(3.75+3.375)/2=3.5625.

$$f(1.4375{,}3.8125) = -20.8555,$$
$$f(0.6875{,}3.5625) = -20.5742.$$

Таблица 3.6

Результаты шага 6

Значение функции в лучшей точке	Значение функции в предпоследней точке	Значение функции в худшей точке
f(1.4375,3.8125)=-20.8555	f(1.125,3.375)=-20.6719	F(0.6875,3.5625)=-20.5742

Шаг 7

1. Сортировка. Переобозначим вершины: x_h=(0.6875,3.5625); x_g=(1.125,3.375); x_1=(1.4375,3.8125).
2. Определение центра тяжести:

$$x_c = \frac{x_g + x_1}{2} = (1.28125{,}3.59375).$$

3. Отражение:

$$x_r = 2x_c - x_h = (1.875{,}3.625),$$
$$f(x_r) = -20.4219.$$

4. Проверка: $f(x_r) < f(x_1)$. Условие не выполняется.
5. Проверка: $f_h > f_r > f_g$. Условие не выполняется.
6. Проверка: $f_r > f_h$. Условие выполняется. Переходим к операции сжатия. Для этого строим точку

$$x_s = 0.5x_h + 0.5x_c = (1.5781{,}3.6094),$$
$$f(x_s)=-20.739.$$

7. Проверка $f_s < f_h$, следовательно, меняем x_h на x_s.

Таблица 3.7

Результаты шага 7

Значение функции в лучшей точке	Значение функции в предпоследней точке	Значение функции в худшей точке
f(1.4375,3.8125)=-20.8555	f(1.125,3.375)=-20.6719	F(1.578125,3.609375)=-20.739

Шаг 8

1. Сортировка. Переобозначим вершины: x_h=(1.125,3.375); x_g=(1.578125,3.609375); x_1=(1.4375,3.8125).
2. Определение центра тяжести:

$$x_c = \frac{x_g + x_1}{2} = (1.5078{,}3.7109).$$

3. Отражение:

$$x_r = 2x_c - x_h = (1.8906{,}4.0488),$$
$$f(x_r) = -20.1628.$$

4. Проверка: $f(x_r) < f(x_1)$. Условие не выполняется.
5. Проверка: $f_h > f_r > f_g$. Условие не выполняется.
6. Проверка: $f_r > f_h$. Условие выполняется. Переходим к операции сжатия. Для этого находим точку

$$x_s = 0.5x_h + 0.5x_c = (1.6992{,}3.8789),$$
$$f(x_s)=-20.5811.$$

Это наихудшее значение. Следовательно, выполняем глобальное сжатие в направлении значения x_1.
Точка (1.125,3.375). Первая координата:
(1.4375+1.125)/2=1.28125.
Вторая координата:
(3.8125+3.375)/2=3.59375.

Точка (1.578125,3.609375). Первая координата:
(1.578125+1.4375)/2=1.507813.
Вторая координата:
(3.609375+3.8125)/2=3.710938,
F(1.28125,3.59375)=-20.8701,
F(1.507813,3.710938)=-20.8054.

Таблица 3.8

Результаты шага 8

Значение функции в лучшей точке	Значение функции в предпоследней точке	Значение функции в худшей точке
F(1.28125,3.59375)=-20.8701	f(1.4375,3.8125)=-20.8555	F(1.507813,3.710938)=-20.8054

Шаг 9

1. Сортировка. Переобозначим вершины: x_h=(1.507813,3.710938); x_g=(1.4375,3.8125); x_1=(1.28125,3.59375).
2. Определение центра тяжести:

$$x_c = \frac{x_g + x_1}{2} = (1.3594{,}3.7031).$$

3. Отражение:

$$x_r = 2x_c - x_h = (1.210938{,}3.695313),$$
$$f(x_r) = -20.9269.$$

4. Проверка: $f(x_r)<f(x_1)$. Условие выполняется. Следовательно, направление выбрано удачно, увеличиваем отрезок (операция растяжения):
$x_b = (1-\gamma)\ x_c + \gamma x_r$

$$\gamma = 2$$

$x_b = -x_c + 2x_r$=(1.0625,3.6875)
$f(x_b)$=-20.918

5. Проверка $f(x_b)<f(x_r)$. Условие не выполняется. Следовательно, меняем x_h на x_r.

Таблица 3.9

Результаты шага 9

Значение функции в лучшей точке	Значение функции в предпоследней точке	Значение функции в худшей точке
f(1.210938,3.695313)=-20.9269	F(1.28125,3.59375)=-20.8701	f(1.4375,3.8125)=-20.8555

Данный процесс продолжается до тех пор, пока не будет выполнено условие остановки.

Условие остановки

Метод завершает свою работу в случае, если полученный симплекс имеет величину меньше заданной точности.

Найдем следующие значения (в общем случае для n+1 точки):

Для координаты i:

1. Рассчитаем среднее значение по всем точкам:

$$\overline{x_i} = \frac{1}{n+1}\sum_{j=1}^{n+1} x_i^j. \tag{3.6}$$

2. Определим дисперсию:

$$\sigma_i = \frac{1}{(n+1)}\sum_{j=1}^{n+1}\left(x_j^i - \overline{x_i}\right)^2. \tag{3.7}$$

Приведем пример расчета для точек из последней таблицы. Имеем три точки (1.210938,3.695313), (1.28125,3.59375) и (1.4375,3.8125). Ищем среднее по первой координате:

$$\overline{x_1}=\frac{1}{3}(1.210938 + 1.28125 + 1.4374) = 1.309896,$$

$$\sigma_1 = \frac{1}{3}((1.210938 - 1.309896)^2 + (1.28125 - 1.309896)^2 + (1.4374 - 1.309896)^2) = 0.008965.$$

Аналогичным образом рассчитываются значения σ по остальным координатам (в данном случае, по второй координате).

Если все эти значения меньше заданной точности ε, то метод завершает свою работу.

3.2.Задание к работе

Методом Хука-Дживса или методом Недлера-Мида осуществить поиск оптимального решения экстремальной задачи. Кроме окончательного ответа вывести таблицу, каждая строка которой содержит промежуточные результаты на данном шаге.

Для метода Хука-Дживса таблица будет следующей

Таблица 3.10

Промежуточные результаты для метода Хука-Дживса

№ ит.	Старая точка x_s	$F(x_s)$	новая точка x_n	$F(x_n)$

Для метода Нелдера-Мида необходимо вывести следующую таблицу.

Таблица 3.11

Промежуточные результаты для метода Нелдера-Мида

№ ит.	Худшая точка w	F(w)	“средняя точка” g	F(g)	Лучшая точка b	F(b)

Отчет должен содержать:

- вариант задания;
- краткие описание метода;
- результат реализации метода (программную форму с входными данными и результатами);
- выводы.

3.3. Варианты заданий

Таблица 3.12

Варианты задания на лабораторную работу № 3

	Целевая функция	Метод	Нач. т.	Ответ
1	$f(x_1,x_2)=4x_1^2+5x_2^2-3x_1x_2+9x_1-2x_2+5$	Х.-Д.	(2,3)	$\left(-\frac{84}{71},-\frac{11}{71}\right)$
2	$f(x_1,x_2)=4x_1^2+5x_2^2-3x_1x_2+9x_1-2x_2+5$	Н.-М.	(2,3) (-1,-1) (0,0)	$\left(-\frac{84}{71},-\frac{11}{71}\right)$
3	$f(x_1,x_2)=x_1^2+3x_2^2+3x_1x_2-x_1-2x_2-1$	Х.-Д.	(3,3)	$\left(0,\frac{1}{3}\right)$

Окончание табл. 3.12

4	$f(x_1,x_2) = x_1^2 + 3x_2^2 + 3x_1x_2 - x_1 - 2x_2 - 1$	Н.-М.	(3,3) (1,0) (0,-1)	$\left(0,\frac{1}{3}\right)$
5	$f(x_1,x_2) = 6x_1^2 + x_2^2 - x_1x_2 + 4x_1 - 8x_2 + 1$	Х.-Д.	(2,2)	(0,4)
6	$f(x_1,x_2) = 6x_1^2 + x_2^2 - x_1x_2 + 4x_1 - 8x_2 + 1$	Н.-М.	(2,2) (-2,-2) (1,0)	(0,4)
7	$f(x_1,x_2) = 2x_1^2 + 4x_2^2 - 5x_1x_2 + 11x_1 + 8x_2 - 3$	Х.-Д.	(1,1)	$\left(-\frac{128}{7},-\frac{87}{7}\right)$
8	$f(x_1,x_2) = 2x_1^2 + 4x_2^2 - 5x_1x_2 + 11x_1 + 8x_2 - 3$	Н.-М.	(1,1) (0,2) (2,0)	$\left(-\frac{128}{7},-\frac{87}{7}\right)$
9	$f(x_1,x_2) = 7x_1^2 + 3x_2^2 + 0.5x_1x_2 - 3x_1 - 5x_2 + 2$	Х.-Д.	(2,-2)	(0.185,0.817)
10	$f(x_1,x_2) = 7x_1^2 + 3x_2^2 + 0.5x_1x_2 - 3x_1 - 5x_2 + 2$	Н.-М.	(2,-2) (0,2) (-2,0)	(0.185,0.817)
11	$f(x_1,x_2) = 3x_1^2 + 5x_2^2 + 2x_1x_2 - 7x_1 + x_2 - 4$	Х.-Д.	(2,-2)	$\left(\frac{9}{7},-\frac{5}{14}\right)$
12	$f(x_1,x_2) = 3x_1^2 + 5x_2^2 + 2x_1x_2 - 7x_1 + x_2 - 4$	Н.-М.	(2,-2) (-2,2) (0,0)	$\left(\frac{9}{7},-\frac{5}{14}\right)$

ЛАБОРАТОРНАЯ РАБОТА 4

ЧИСЛЕННЫЕ МЕТОДЫ ПЕРВОГО ПОРЯДКА РЕШЕНИЯ ЗАДАЧ НА БЕЗУСЛОВНЫЙ ЭКСТРЕМУМ

Цель работы: изучение и практическая реализация градиентных методов.

4.1. Краткие теоретические сведения

Рассматривается задача безусловной оптимизации:

$$f(x_1, x_2, \dots, x_n) \to min. \quad (4.1)$$

Градиентный метод относится к классическим численным методам безусловной оптимизации. Идея метода заключается в замене минимизируемой функции в окрестности очередной точки x^k первыми членами ее разложения в ряд Тейлора. В градиентном методе используется линейная часть разложения.

Общая схема метода:

$$x^{k+1} = x^k + \alpha_k h^k. \tag{4.2}$$

Вектор h^k определяет направление (k+1)-го шага оптимизации, а коэффициент α_k - длину этого шага.

В градиентном методе используются разные способы выбора длины шага. Существуют три разновидности метода:

-с постоянным шагом;

- с изменяющимся шагом (дробление шага);

- метод скорейшего спуска.

Для методов с постоянным шагом в качестве значения α_k берется некоторая константа. Однако такой подход имеет следующие недостатки:

- в случае, если шаг очень маленький, сходимость будет крайне медленной;

- в случае, если шаг достаточно большой, возможна ситуация, когда сходимость будет вообще недостижима (большой шаг не позволит приблизиться к оптимуму, каждый раз переходя через данную точку).

В следствие этого, такой метод на практике используется крайне редко.

Так, если длина шага выбирается из условия минимизации функции вдоль направления антиградиента, то получаем вариант метода, который называется методом наискорейшего спуска.

Дробление шага

Если h^k – направление убывания, то дробление шага можно осуществлять следующим образом.

Выбираются некоторые константы $\beta>0$; $0< \lambda<1$. Для коэффициента $\alpha=\beta$ проверяется выполнение условия:

$$f(x^k + \alpha_k h^k) < f(x^k). \tag{4.3}$$

Если оно выполнено, то предполагают $\alpha_k=\alpha$; если нет, то производится дробление шага:

$$\alpha=\lambda\beta \tag{4.4}$$

и вновь проверяется выполнение условия (4.3). Процесс дробления продолжается до тех пор, пока условие (4.3) не станет истинным.

В методе скорейшего спуска h^k берется равным антиградиенту функции f в точке x^k, т.е.

$$h^k = -f'(x^k), \tag{4.5}$$
$$x^{k+1} = x^k - \alpha_k f'(x^k). \tag{4.6}$$

Сходимость градиентного метода

1) Случай невыпуклой функции

В случае, когда минимизируемая функция f не предполагается выпуклой, градиентный метод может обеспечить сходимость к множеству стационарных точек функции f.

Теорема. Пусть функция f дифференцируема и ограничена снизу на R^n, а ее градиент удовлетворяет условию Липшица:

$$\left\| f'(x_2) - f'(x_1) \right\| \le M \| x_2 - x_1 \|. \tag{4.7}$$

Тогда если выполняются условия

$$f\left(x^k + \alpha_k f'(x^k)\right) - f(x^k) \le \varepsilon \alpha_k \left\| f'(x^k) \right\|, \tag{4.8}$$
$$\alpha_k \ge min\left(\beta, \frac{\lambda(1-\varepsilon)}{M}\right), \tag{4.9}$$

то при произвольной начальной точке x^0 для метода (4.6) имеем

$$\lim_{k\to\infty} \left\| f'(x^k) \right\| = 0. \tag{4.10}$$

Доказательство приведено в [4].

Условие (4.8) означает, что любая предельная точка последовательности $\{x^k\}$, генерируемая градиентным методом, является стационарной точкой функции f. Однако возможны случаи, когда эта точка не является точкой минимума.

Пример. С помощью градиентного метода решить задачу безусловной оптимизации:

$$f(x_1, x_2) = 3x_1^2 - 4x_1 + x_2^2 - x_1 x_2 \to min$$

В качестве начального приближения взять точку (-2,3).

Решим сначала данную задачу аналитически.

$$\frac{\partial F}{\partial x_1} = 6x_1 - x_2 - 4,$$
$$\frac{\partial F}{\partial x_2} = -x_1 + 2x_2.$$

Приравняв к нулю, получим:

$$\begin{cases} 6x_1 - x_2 - 4 = 0 \\ -x_1 + 2x_2 = 0 \end{cases}.$$

Получена система линейных уравнений, которую можно решить симплекс-методом или методом Гаусса. Однако поскольку количество уравнений равно двум, и система достаточно проста, выразим из второго уравнения x_1 и подставим в первое:

$$\begin{cases} x_1 = 2x_2 \\ 12x_2 - x_2 - 4 = 0 \end{cases}.$$

Отсюда получим ответ:

$$\begin{cases} x_2 = \dfrac{4}{11} \\ x_1 = \dfrac{8}{11} \end{cases}.$$

Таким образом, точное решение задачи (8/11, 4/11)=(0,36, 0,72).
Вычислим теперь данное решение приближенно.
Расчетные формулы имеют следующий вид:

$$x_1^k = x_1^{k-1} - \lambda_k \frac{\partial F}{\partial x_1},$$
$$x_2^k = x_2^{k-1} - \lambda_k \frac{\partial F}{\partial x_2}.$$

А). Решим задачу с **постоянным** шагом.
Здесь важной задачей является выбор шага.
Пусть, например, выбран шаг λ=0.5.
Рассчитаем значения производных в начальной точке (-2,3):

$$\left.\frac{\partial F}{\partial x_1}\right|_{x_1=-2;\, x_2=3} = 6 \cdot (-2) - 3 - 4 = -19,$$
$$\left.\frac{\partial F}{\partial x_2}\right|_{x_1=-2;\, x_2=3} = -(-2) + 2 \cdot 3 = 8.$$

Тогда:

$$x_1^1 = -2 - 0.5 \cdot (-19) = 7.5,$$
$$x_1^2 = 3 - 0.5 \cdot 8 = -1.$$

Получили новую точку (7.5, -1). Вычислим значение функции в данной точке:

$$f(x_1, x_2) = 3x_1^2 - 4x_1 + x_2^2 - x_1x_2 = 147.25.$$

Далее опять вычисляем значение производных в данной точке: подставляем λ=0.5 и переходим к следующему шагу. Итерационный процесс можно остановить в случае, если разница между текущим и предыдущим значениями не превышает заданной точности.

Расчеты приведены в следующей таблице.

Таблица 4.1

Расчеты градиентного метода с шагом 0.5

x1	x2	f(x1,x2)	df/dx1	df/dx2
-2	3	35	-19	8
7,5	-1	147,25	42	-9,5
-13,5	3,75	665,4375	-88,75	21
30,875	-6,75	2990,266	188	-44,375
-63,125	15,4375	13419,61	-398,188	94
135,9688	-31,5625	60206,33	843,375	-199,094
-285,719	67,98438	270094,8	-1786,3	421,6875
607,4297	-142,859	1211669	3783,438	-893,148
-1284,29	303,7148	5435633	-8013,45	1891,719
2722,436	-642,145	24384623	16972,76	-4006,72

Очевидно, что итерационный процесс будет расходиться.

Изменим шаг до 0.1. Аналогичным образом будем считать производные в полученных точках и по формуле (4.2) искать значение в следующей точке.

Результаты будут следующие:

Таблица 4.2

Расчеты градиентного метода с шагом 0.1

x1	x2	f(x1,x2)	df/dx1	df/dx2
-2	3	35	-19	8
-0,1	2,2	5,49	-6,8	4,5
0,58	1,75	0,7367	-2,27	2,92
0,807	1,458	-0,3251	-0,616	2,109
0,8686	1,2471	-0,73897	-0,0355	1,6256
0,87215	1,08454	-0,97632	0,14836	1,29693
0,857314	0,954847	-1,13117	0,189037	1,05238
0,83841	0,849609	-1,23533	0,180853	0,860808
0,820325	0,763528	-1,30587	0,158422	0,706731
0,804483	0,692855	-1,3537	0,134042	0,581227
0,791079	0,634732	-1,38614	0,111739	0,478386
0,779905	0,586894	-1,40814	0,092534	0,393883
0,770651	0,547505	-1,42307	0,076402	0,32436
0,763011	0,515069	-1,43319	0,062997	0,267128
0,756711	0,488357	-1,44006	0,051912	0,220002
0,75152	0,466356	-1,44472	0,042765	0,181193
0,747244	0,448237	-1,44788	0,035225	0,149231
0,743721	0,433314	-1,45002	0,029013	0,122907
0,74082	0,421023	-1,45148	0,023896	0,101227
0,73843	0,410901	-1,45247	0,019681	0,083371

Окончание табл. 4.2

0,736462	0,402564	-1,45313	0,01621	0,068665
0,734841	0,395697	-1,45359	0,01335	0,056553
0,733506	0,390042	-1,4539	0,010995	0,046577
0,732407	0,385384	-1,45411	0,009056	0,038361
0,731501	0,381548	-1,45425	0,007459	0,031595
0,730755	0,378388	-1,45434	0,006143	0,026022
0,730141	0,375786	-1,45441	0,005059	0,021432
0,729635	0,373643	-1,45445	0,004167	0,017651
0,729218	0,371878	-1,45448	0,003432	0,014538
0,728875	0,370424	-1,4545	0,002827	0,011973
0,728592	0,369227	-1,45452	0,002328	0,009861
0,72836	0,368241	-1,45453	0,001917	0,008122
0,728168	0,367429	-1,45453	0,001579	0,006689

Получили решение с заданной точностью за 34 итерации.

Б) Решение задачи с **дроблением шага**.

Главным недостатком градиентного метода с постоянным шагом является невозможность гарантировать сходимость. В частности, при больших значениях λ метод может расходиться, а при маленьких заведомо будет тратиться очень большое количество операций для вычислительного поиска. Решить данную проблему можно, используя дробление шага.

Рассмотрим значения, полученные в результате выполнения первого шага:

$$x_1^1 = -2 - 1 \cdot (-19) = 17,$$
$$x_1^2 = 3 - 1 \cdot 8 = -5.$$

Получили новую точку (17, -5). Вычислим значение функции в данной точке:

$$f(x_1, x_2) = 3x_1^2 - 4x_1 + x_2^2 - x_1 x_2 = 909.$$

Поскольку значение функции в этой точке существенно превосходит значение в исходной точке (-2,3), уменьшим λ вдвое.

$$x_1^1 = -2 - 0.5 \cdot (-19) = 7.5,$$
$$x_1^2 = 3 - 0.5 \cdot 8 = -1,$$
$$f(x_1, x_2) = 147.25.$$

Это значение также больше, чем значение 35, полученное в начальной точке. Следовательно, необходимо еще уменьшить значение λ:

$$x_1^1 = -2 - 0.25 \cdot (-19) = 2.75,$$

$$x_1^2 = 3 - 0.25 \cdot 8 = 1,$$
$$f(x_1, x_2) = 9.9375.$$

Поскольку это значение меньше, чем 35, оставляем точку (2.75,1) как результат первого шага. Вычисляем производную в данной точке:

$$\left.\frac{\partial F}{\partial x_1}\right|_{x_1=-2.75;\, x_2=1} = 6 \cdot (2.75) - 1 - 4 = 11.5,$$
$$\left.\frac{\partial F}{\partial x_2}\right|_{x_1=-2.75;\, x_2=1} = -2.75 + 2 \cdot 1 = -0.75.$$

Определяем новую точку по формуле:

$$x_1^1 = 2.75 - 0.25 \cdot (11.5) = -0.125,$$
$$x_1^2 = 3 - 0.25 \cdot (-0.75) = 1.1875.$$

Вычисляем значение функции в данной точке:

$$f(x_1, x_2) = 2{,}105469.$$

Так как это значение меньше предыдущего (9.9375), то шаг λ=0.25 оставляем и переходим к следующему шагу.

Результаты сведены в следующую таблицу.

Таблица 4.3

Расчеты градиентного метода с дроблением шага

	x1	x2	f(x1,x2)	df/dx1	df/dx2	alpha
нач усл	-2	3	35	-19	8	1
шаг 1	17	-5	909			0,5
	7,5	-1	147,25			0,25
	2,75	1	9,9375	11,5	-0,75	0,25
шаг 2	-0,125	1,1875	2,105469	-5,9375	2,5	0,25
шаг 3	1,359375	0,5625	-0,34204	3,59375	-0,23438	0,25
шаг 4	0,460938	0,621094	-1,10689	-1,85547	0,78125	0,25
шаг 5	0,924805	0,425781	-1,3459	1,123047	-0,07324	0,25
шаг 6	0,644043	0,444092	-1,42059	-0,57983	0,244141	0,25
шаг 7	0,789001	0,383057	-1,44394	0,350952	-0,02289	0,25
шаг 8	0,701263	0,388779	-1,45123	-0,1812	0,076294	0,25
шаг 9	0,746563	0,369705	-1,45351	0,109673	-0,00715	0,25
шаг 10	0,719145	0,371493	-1,45422	-0,05662	0,023842	0,25
шаг 11	0,733301	0,365533	-1,45444	0,034273	-0,00224	0,25
шаг 12	0,724733	0,366092	-1,45451	-0,0177	0,007451	0,25
шаг 13	0,729157	0,364229	-1,45454	0,01071	-0,0007	0,25

Как можно видеть из данной таблицы, для нахождения решения потребовалось гораздо меньше времени, чем в предыдущем случае.

В). Рассмотрим решение задачи методом **скорейшего спуска**

Пусть x^0=(-2,3) .

ШАГ 1

$$\frac{\partial F}{\partial x_1} = -19,$$
$$\frac{\partial F}{\partial x_2} = 8.$$
$$x_1^1 = -2 + \lambda \cdot 19,$$
$$x_2^1 = 3 - \lambda \cdot 8.$$

Найдем λ, исходя из требования минимизации функции F:

$F(x_1, x_2) = 3(-2 + \lambda \cdot 19)^2 - 4(-2 + \lambda \cdot 19) + (3 - \lambda \cdot 8)^2 + (-2 + \lambda \cdot 19) * (3 - \lambda \cdot 8),$

$\frac{\partial F}{\partial \lambda} = 6(-2 + \lambda \cdot 19) \cdot 19 - 4 \cdot 19 + 2(3 - \lambda \cdot 8) \cdot (-8) + 19 \cdot (3 - \lambda \cdot 8) - 8 * (-2 + \lambda \cdot 19) = 0.$

2598λ-425=0,

λ=0.164.

$x_1^1 = -2 + 0.164 \cdot 19 = 1.116,$

$x_2^1 = 3 - 0.164 \cdot 8 = 1.688,$

F=0.23.

ШАГ 2

$$\frac{\partial F}{\partial x_1} = 6x_1 - x_2 - 4 = 1.008,$$

$$\frac{\partial F}{\partial x_2} = -x_1 + 2x_2 = 2.26.$$
$$x_1^2 = 1.116 - \lambda \cdot 1.008,$$
$$x_2^2 = 1.688 - \lambda \cdot 2.26.$$

$F(x_1, x_2) = 3 \cdot (1.116 - \lambda \cdot 1.008)^2 - 4(1.116 - \lambda \cdot 1.008) + (1.688 - \lambda \cdot 2.26)^2 + (1.116 - \lambda \cdot 1.008) \cdot (1.688 - \lambda \cdot 2.26),$

$\frac{\partial F}{\partial \lambda} = 6(1.116 - \lambda \cdot 1.008) \cdot (-1.008) + 4 \cdot 1.008 + 2(1.688 - \lambda \cdot 2.26) * (-2.26) + 1.008 \cdot (1.688 - \lambda \cdot 2.26) + 2.26 \cdot (1.116 - \lambda \cdot 1.008)$ =0.

11,45λ-6,58546=0.

λ=0.57513.

$x_1^2 = 1.116 - 0.57513 \cdot 1.008 = 0{,}536269,$

$x_2^2 = 1.688 - 0.67499 \cdot 2.26 = 0{,}3882.$

и т.д.

4.2. Задание к работе

Заданным градиентным методом решить заданную задачу не безусловный экстремум (согласно варианту). Проверить полученный результат с аналитическими расчетами.

Отчет должен содержать:

- вариант задания;
- краткие описание метода;
- результат реализации метода (программную форму с входными данными и результатами);
- выводы.

4.3. Варианты заданий

Таблица 4.4

Варианты задания на лабораторную работу № 4

	Целевая функция	Нач. т.	Ответ
1	$f(x_1,x_2) = 4x_1^2 + 5x_2^2 - 3x_1x_2 + 9x_1 - 2x_2 + 5$	(2,3)	$\left(-\frac{84}{71}, -\frac{11}{71}\right)$
2	$f(x_1,x_2) = 4x_1^2 + 5x_2^2 - 3x_1x_2 + 9x_1 - 2x_2 + 5$	(2,3)	$\left(-\frac{84}{71}, -\frac{11}{71}\right)$
3	$f(x_1,x_2) = x_1^2 + 3x_2^2 + 3x_1x_2 - x_1 - 2x_2 - 1$	(3,3)	$\left(0, \frac{1}{3}\right)$
4	$f(x_1,x_2) = x_1^2 + 3x_2^2 + 3x_1x_2 - x_1 - 2x_2 - 1$	(3,3)	$\left(0, \frac{1}{3}\right)$
5	$f(x_1,x_2) = 6x_1^2 + x_2^2 - x_1x_2 + 4x_1 - 8x_2 + 1$	(2,2)	(0,4)
6	$f(x_1,x_2) = 6x_1^2 + x_2^2 - x_1x_2 + 4x_1 - 8x_2 + 1$	(2,2)	(0,4)
7	$f(x_1,x_2) = 2x_1^2 + 4x_2^2 - 5x_1x_2 + 11x_1 + 8x_2 - 3$	(1,1)	$\left(-\frac{128}{7}, -\frac{87}{7}\right)$
8	$f(x_1,x_2) = 2x_1^2 + 4x_2^2 - 5x_1x_2 + 11x_1 + 8x_2 - 3$	(1,1)	$\left(-\frac{128}{7}, -\frac{87}{7}\right)$
9	$f(x_1,x_2) = 7x_1^2 + 3x_2^2 + 0.5x_1x_2 - 3x_1 - 5x_2 + 2$	(2,-2)	(0.185,0.817)
10	$f(x_1,x_2) = 7x_1^2 + 3x_2^2 + 0.5x_1x_2 - 3x_1 - 5x_2 + 2$	(2,-2)	(0.185,0.817)
11	$f(x_1,x_2) = 3x_1^2 + 5x_2^2 + 2x_1x_2 - 7x_1 + x_2 - 4$	(2,-2)	$\left(\frac{9}{7}, -\frac{5}{14}\right)$
12	$f(x_1,x_2) = 3x_1^2 + 5x_2^2 + 2x_1x_2 - 7x_1 + x_2 - 4$	(2,-2)	$\left(\frac{9}{7}, -\frac{5}{14}\right)$

ЛАБОРАТОРНАЯ РАБОТА 5

ЧИСЛЕННЫЕ МЕТОДЫ РЕШЕНИЯ ЭКСТРЕМАЛЬНЫХ ЗАДАЧ С ОГРАНИЧЕНИЯМИ

Цель работы: изучение теоретических основ и навыков практической реализации методов решения задач на условный экстремум.

Краткие теоретические сведения

В общем случае задача на условный экстремум имеет вид:

$$\begin{cases} f(x) \to min, \\ g_i(x) \le 0, i = 1, \dots, n \end{cases}. \tag{5.1}$$

Здесь f – целевая функция, множество функций $g_1,\ldots,g_n$ образуют ограничения задачи.

Основная идея метода штрафных функций заключается в переходе от задачи условной оптимизации к задаче безусловной оптимизации путем изменения исходной целевой функции за счет ограничений. В частности, к целевой функции добавляется штраф, который мал (или равен нулю), если ограничения выполнены (или выполнены с точностью до ε). Вид новой функции можно описать следующим образом:

$$F(x,r) = f(\mathrm{x}) + P(x,r). \tag{5.2}$$

Здесь P(x,r) – штраф, r – положительная величина, регулирующая «вклад» штрафа в целевую функцию.

В случае невыполнения ограничений штраф возрастает, причем, чем дальше найденная точка от допустимой области, тем больше величина штрафа.

Виды штрафов:

1.Квадратичный штраф используется для учета ограничений равенств:

$$P(x,r^k) = r^k \sum_{i=1}^{K} \left(h_i(x)\right)^2 . \tag{5.3}$$

2. Штраф типа квадрата срезки:

$$P(x,r^k) = r^k \sum_{j=1}^{J} g_j^+(x)^2 , \tag{5.4}$$

где $g_j^+(x)$ - срезка функции:

$$g_j^+(x) = \max\{0, g_j(x)\}^2 = \begin{cases} g_j(x), & если\ g_j(x) > 0 \\ 0, & если\ g(x) \le 0 \end{cases} . \tag{5.5}$$

Если в задаче (1) имеется n ограничений, то функция P должна содержать n слагаемых: по одному на каждое ограничение.

Алгоритм метода штрафных функций

Предварительный этап: составить штрафную функцию и новую целевую функцию F.

Задать начальную точку расчета x^0, рассчитать частные производные по каждому аргументу в этой точке.

Задать начальное значение r (например, 0.01).

Этап i (i=1,2,…). Каким-либо численным методом решения задач безусловной оптимизации найти минимум x^i функции F.

Если величина штрафа меньше заданной точности,

То выдать x^i в качестве ответа

Иначе увеличить r в «C» раз и перейти к этапу i+1.

Замечания.

1. Начальная точка, как правило, берется такой, чтобы ограничения нарушались (поэтому данный метод иногда называют методом внешних точек).
2. Значение C целесообразно брать от 4 до 10, постепенно, тем самым, наращивая r, и, как следствие, вклад штрафа в результирующую функцию.
3. Если x* - минимум некоторой функции f(x), то эта же точка является максимумом для функции –f(x).

Рассмотрим метод штрафных функций на следующем примере. Пусть необходимо решить следующую оптимизационную задачу.

$$f(x_1, x_2) = 4x_1^2 + 4x_1 + x_2^2 - 8x_2 + 5 \to min,$$

$$2x_1 - x_2 = 6.$$

В качестве начальной точки выберем точку (0,0).

Решив задачу аналитически, получим: x_1=9/4; x_2=-3/2. При этом, если решать задачу без ограничения, то получим ответ x_1=-0.5; x_2=4.

Таким образом, видно, что добавление дополнительного ограничения существенно меняет точку минимума.

Выберем в качестве метода безусловной оптимизации градиентный метод с изменяющимся шагом.

Этап 1 R=0.01 (вклад штрафа в целевую функцию крайне мал).

$$F(x_1, x_2) = 4x_1^2 + 4x_1 + x_2^2 - 8x_2 + 5 + 0{,}01(2x_1 - x_2 - 6)^2.$$

Выберем начальную точку (0,0) и будем решать задачу градиентным методом:

$$F(0,0) = 0.36,$$

$$\frac{\partial F}{\partial x_1} = 8x_1 + 4 + 0{,}02(2x_1 - x_2 - 6)2 = 3{,}76,$$

$$\frac{\partial F}{\partial x_2} = 2x_2 - 8 + 0{,}02(2x_1 - x_2 - \ 6)(-1) = -7{,}88.$$

$$x_1^1 = 0 - \lambda * 3.76,$$
$$x_2^1 = 0 + \lambda * 7.88,$$

$F(x_1, x_2) = 4(-\lambda * 3.76)^2 + 4(-\lambda * 3.76) + (\lambda * 7.88)^2 - 8(\lambda * 7.88) + 5 + 0{,}01(2(-\lambda * 3.76) - (\lambda * 7.88) - \ 6)^2.$

Возьмем начальное значение λ=1. Если значение новой функции будет меньше, чем значение исходной функции (0.36), значит, данный шаг приемлем – оставляем λ, в противном случае – уменьшаем его вдвое.

$$F(-3.76, 7.88) = 45.1444.$$

Так как $45.1444 > 0.36$, то уменьшаем λ вдвое. Получаем точку:

$$x_1^1 = -0{,}5 * 3.76 = -1.88,$$
$$x_2^1 = 0.5 * 7.88 = 3.94,$$
$$F(-1.88, 3.94) = -7.5019.$$

Так как $-7.5019 < 0.36$, то оставляем данную точку в качестве результата первой итерации. Итак:

$$x_1^1 = -1.88,$$
$$x_2^1 = 3.94.$$
$$\left.\frac{\partial F}{\partial x_1}\right|_{x_1=-1.88;\, x_2=3.94} = -11.588,$$
$$\left.\frac{\partial F}{\partial x_2}\right|_{x_1=-1.88;\, x_2=3.94} = 0.154.$$

Следующая точка будет определяться по формуле:

$$x_1^2 = -1.88 - \lambda * (-11.88) = -1.88 - 0.5 * (-11.88) = 3.914,$$
$$x_2^2 = 3.94 - \lambda * 0.154 = 3.94 - 0.5 * 0.154 = 3.863.$$

Вычисляем значение функции в данной точке:

$$F(3.914, 3.863) = 60.99377.$$

Так как $60.99377 > -7.5019$, то опять уменьшаем λ вдвое:

$$x_1^2 = -1.88 - 0.25 * (-11.88) = 1.017,$$
$$x_2^2 = 3.94 - 0.25 * 0.154 = 3.9015.$$

Вычисляем значение функции в данной точке и т.д.
После этого делаем аналогичные операции.

Результаты представлены в следующей таблице.

Таблица 5.1

Результаты первого этапа метода штрафных функций

шаг	X_1	X_2	$F(x_1, x_2)$	$\frac{\partial F}{\partial x_1}$	$\frac{\partial F}{\partial x_2}$	λ
0	0	0	0,36	3,76	-7,88	1
1.1	-3,76	7,88	45,1444			0.5
1.2	-1,88	3,94	-7,5019	-11,588	0,154	0.5
2.1	3,914	3,863	60,99377			0,25
2.2	1,017	3,9015	-7,1662			0,125
2.3	-0,4315	3,92075	-15,812	0,11665	0,0572	0,125
3.1	-0,4461	3,9136	-15,8133	-0,0009	0,0433	0,125
4.1	-0,446	3,90819	-15,8135	0,00023	0,0324	0,125

Завершен первый этап данного метода (значения x на шаге 4.1 отличаются от значений на шаге 3.1 на величину меньше 0.01). Получили ответ на первой итерации:

$$X_1=-0,446;\ x_2=3,90819.$$

Данный ответ крайне близок к аналитическому значению, полученному без учета ограничений задачи (**x_1=-0.5; x_2=4**). Проверим, чему равны ограничения:

$$2x_1 - x_2 - 6 = -10.8.$$

Объясним данный результат. В сформированной функции величина штрафа пока незначительна. Поэтому был найден глобальный минимум функции безотносительно ограничений, поскольку, значение в данной точке существенно меньше минимума, который достигается с учетом ограничений.

<u>Этап 2</u>

Поскольку этот результат крайне далек от требуемого (ограничения нарушены), поэтому увеличиваем r в 10 раз (что увеличит штраф за отклонение ограничений в целевой функции) и, начиная с найденной точки повторяем итерации метода.

r=0.1.

$$F(x_1,x_2) = 4x_1^2 + 4x_1 + x_2^2 - 8x_2 + 5 + 0,1(2x_1 - x_2 - 6)^2,$$

$$F(-0.446{,}3.90819) = -5.3156.$$

$$\frac{\partial F}{\partial x_1} = 8x_1 + 4 + 0,1 * 2 * (2x_1 - x_2 - 6) * 2,$$

$$\frac{\partial F}{\partial x_2} = 2x_2 - 8 + 0,1 * 2 * (2x_1 - x_2 - 6)(-1);$$

$$\left.\frac{\partial F}{\partial x_1}\right|_{x_1=-0.446;\ x_2=3.90819} = -3.8878,$$
$$\left.\frac{\partial F}{\partial x_2}\right|_{x_1=-0.446;\ x_2=3.90819} = 1.9764;$$
$$F(-0.446, 3.90819) = -5.3156.$$
$$x_1^1 = -0.446 + \lambda * 3.8878,$$
$$x_2^1 = 3.90819 - \lambda * 1.9764.$$

$$F(x_1, x_2) = 4(-0.446 + \lambda * 3.8878)^2 + 4(-0.446 + \lambda * 3.8878) + (3.90819 - \lambda * 1.9764)^2 - 8(3.90819 - \lambda * 1.9764) + 5 + 0{,}1(2(-0.446 + \lambda * 3.8878) - (3.90819 - \lambda * 1.9764) - 6)^2.$$

Возьмем начальное значение λ=1.

$$F(-0.446 + 3.8878, 3.90819 - 1.9764) = F(3{,}44185, 1{,}93179) = 49.54.$$

Поскольку 49.54>-5.3156, то уменьшаем λ вдвое. Снова вычисляем значение в предполагаемой новой точке; сравниваем значения функций в новой и старой точках. Если значение в новой точке будет меньше, то запоминаем ее как точку на данном шаге и следующую итерацию начинаем с нее. В противном случае опять уменьшаем шаг, и в качестве основы берем точку с предыдущей итерации.

Сведем результаты в таблицу.

Таблица 5.2

Результаты второго этапа метода штрафных функций

Шаг	X_1	X_2	$F(x_1, x_2)$	$\frac{\partial F}{\partial x_1}$	$\frac{\partial F}{\partial x_2}$	λ
II	-0,446	3,90819	-5,3156	-3,8878	1,97640	1
1.1	3,44185	1,93179	49,540			0,5
1.2	1,49794	2,91999	3,643			0,25
2.1	0,52598	3,41409	-5,454	4,86303	0,5006	0,25
2.2	-0,68977	3,288938	-4,9687			0,125
3	-0,08189	3,351513	-6,8261	-0,46127	0,6061	0,125
4	-0,02424	3,275752	-6,8759	0,076432	0,4163	0,125
5	-0,03379	3,223708	-6,8952	0,013174	0,3057	0,125
6	-0,03544	3,185499	-6,9053	0,013966	0,2223	0,125
7	-0,03718	3,157715	-6,9106	0,009717	0,1618	0,125
8	-0,0384	3,137484	-6,9135	0,007121	0,1178	0,125
9	-0,03929	3,122756	-6,915	0,005179	0,0858	0,125

Окончание табл. 5.2

10	-0,03993	3,112034	-6,9158	0,003771	0,0624	0,125
11	-0,04041	3,104228	-6,9162	0,002745	0,0455	0,125
12	-0,04075	3,098545	-6,9164			

Завершили второй этап метода. Ответ на втором этапе

$$X_1=-0,04075;\ x_2=3,098545.$$

Штраф равен -9,5626.

По сравнению с этапом 1, штраф не сильно уменьшился. Поэтому опять увеличиваем r в 10 раз и повторяем итерационную процедуру:

<u>Этап 3</u> r=1

$$F(x_1,x_2) = 4x_1^2 + 4x_1 + x_2^2 - 8x_2 + 5 + (2x_1 - x_2 - 6)^2,$$
$$F(-0.04075{,}3.098545) = -5.3156.$$
$$\frac{\partial F}{\partial x_1} = 8x_1 + 4 + 2*(2x_1 - x_2 - 6)*2,$$
$$\frac{\partial F}{\partial x_2} = 2x_2 - 8 + 2*(2x_1 - x_2 - 6)(-1).$$

Начиная с точки x_1=-0,04075; x_2=3,098545, ищем безусловный экстремум новой функции:

$$\left.\frac{\partial F}{\partial x_1}\right|_{x_1=-0.04075;\, x_2=3.0985} = -33.0461,$$
$$\left.\frac{\partial F}{\partial x_2}\right|_{x_1=-0.04075;\, x_2=3.0985} = 16.55717.$$
$$x_1^1 = -0.04075 - \lambda*(-33.0461) = -0.04075 + 33.0461 = 33.0054,$$
$$x_2^1 = 3.0985 - \lambda*16.55717 = 3.0985 - 16.55717 = -13{,}4586.$$

Таблица 5.3

Результаты третьего этапа метода штрафных функций

Шаг	X_1	X_2	$F(x_1, x_2)$	$\frac{\partial F}{\partial x_1}$	$\frac{\partial F}{\partial x_2}$	λ
	-0,040748	3,098545	68,92943	-33,0461	16,55717	1
1.1	33,005402	-13,4586	10176,01			0,5
1.2	16,482327	-5,18004	2254,152			0,25
1.3	8,2207894	-1,04075	444,4616			0,125
1.4	4,0900206	1,028898	77,42573			0,0625

Окончание табл. 5.3

2.1	2,0246363	2,063722	28,36013	4,139293	4,156342	
3.1	1,7659304	1,80395	26,61115	1,039086	4,15208	0,0625
4.1	1,7009876	1,544445	25,56719	1,03802	3,373831	0,0625
5.1	1,6361114	1,333581	24,85631	0,843458	2,789878	0,0625
6.1	1,5833952	1,159214	24,37165	0,69747	2,303273	0,0625
7.1	1,5398034	1,015259	24,04123	0,575818	1,901822	0,0625
8.1	1,5038147	0,896395	23,81596	0,475456	1,570321	0,0625
9.1	1,4740988	0,79825	23,66237	0,39258	1,296605	0,0625
10.1	1,4495625	0,717212	23,55766	0,324151	1,070599	0,0625
11.1	1,429303	0,6503	23,48627	0,26765	0,883987	0,0625
12.1	1,4125749	0,595051	23,4376	0,220997	0,729903	0,0625
13.1	1,3987626	0,549432	23,40442	0,182476	0,602676	0,0625
14.1	1,3873579	0,511764	23,3818	0,150669	0,497626	0,0625
15.1	1,3779411	0,480663	23,36638	0,124406	0,410887	0,0625
16.1	1,3701657	0,454982	23,35586	0,102722	0,339267	0,0625
17.1	1,3637456	0,433778	23,34869	0,084817	0,28013	0,0625
18.1	1,3584446	0,41627	23,3438	0,070033	0,231302	0,0625
19.1	1,3540675	0,401814	23,34047	0,057825	0,190985	0,0625
20.1	1,3504534	0,389877	23,3382	0,047746	0,157695	0,0625
21.1	1,3474693	0,380021	23,33665	0,039424	0,130208	0,0625
22.1	1,3450053	0,371883	23,3356	0,032552	0,107512	0,0625
23.1	1,3429708	0,365164	23,33488	0,026878	0,088772	0,0625
24.1	1,3412909	0,359616	23,33438	0,022193	0,073298	0,0625
25.1	1,3399039	0,355034	23,33405	0,018325	0,060522	0,0625

Определяем значение ограничений: $2x_1 - x_2 - 6 =$-3,67523

Этот результат все еще далек от необходимого значения, поэтому опять увеличиваем r в 10 раз и переходим к следующему этапу.

Этап 4 r=10

Таблица 5.4

Результаты четвертого этапа метода штрафных функций

Шаг	X_1	X_2	$F(x_1, x_2)$	$\frac{\partial F}{\partial x_1}$	$\frac{\partial F}{\partial x_2}$	λ
	1,3399039	0,35503	144,8997	-132,29	66,2146	1
1.1	133,62974	-65,86	1146895			0,5

Окончание табл. 5.4

1.2	67,484821	-32,752	281361,2			0,25
1.3	34,412362	-16,199	67713,36			0,125
1.4	17,876133	-7,9218	15669,2			0,063
1.5	9,6080185	-3,7834	3342,07			0,031
1.6	5,4739612	-1,7142	602,2396			0,016
1.7	3,4069325	-0,6796	88,2583	90,99281	-39,2278	0,016
2.1	1,9851698	-0,0666	62,77651	-58,6397	31,12726	0,016
3.1	2,901415	-0,553	51,27441	41,44442	-16,2225	0,016
………………						
43.1	2,1732286	-1,12316	40,64443	0,170373	0,361416	0,016
44.1	2,1705666	-1,1288	40,642	0,161995	0,343662	0,016

Завершили четвертый этап метода. Ответ на четвертом этапе:

$$x_1=2{,}1705;\ x_2=-1{,}1288.$$

Штраф равен -0,53.

Этап 5 r=100

Таблица 5.5

Результаты пятого этапа метода штрафных функций

Шаг	X_1	X_2	$F(x_1, x_2)$	$\frac{\partial F}{\partial x_1}$	$\frac{\partial F}{\partial x_2}$	λ
	2,170567	-1,1288	51,88069	-84,6482	42,74874	1
1.1	86,81872	-43,8775	2269703			0,5
1.2	44,49464	-22,5032	565216,5			0,25
1.3	23,33261	-11,816	140218,9			0,125
1.4	12,75159	-6,4724	34531,59			0,0625
1.5	7,461076	-3,8006	8390,785			0,03125
1.6	4,815821	-2,4647	1996,095			0,015625
1.7	3,493194	-1,79675	467,6783			0,007813
1.8	2,831880	-1,46278	120,7021			0,003906
2.1	2,501223	-1,29579	51,52205	83,65729	-40,4153	0,001953
3.1	2,174437	-1,13792	51,17776	-81,246	41,04491	0,000977
2.1	2,491804	-1,29825	50,84724	80,30615	-38,7824	0,000977
3.1	2,178108	-1,14676	50,52994	-77,9805	39,40917	0,000977

На этом этапе получили решение $x_1=2{,}35956$; $x_2=-1{,}22453$.

Штраф равен -0,056.

Данные значения показывают, что на данной итерации уже найдена точка, которая располагается достаточно близко к искомому значению (2.25, -1.5). Поскольку производные в отличие от предыдущих этапов далеки от 0, можно продолжить итерационную процедуру. Однако, сходимость к истинному значению будет осуществляться крайне медленно.

В частности, продолжая, получим:

Таблица 5.6

Завершение пятого этапа метода штрафных функций

Шаг	X_1	X_2	$F(x_1, x_2)$	$\frac{\partial F}{\partial x_1}$	$\frac{\partial F}{\partial x_2}$	λ
	…					
237	2,2273406	-1,43307	42,90179	-0,63156	0,359007	0,003906
238	2,2298076	-1,43447	42,90176	0,655463	-0,27744	0,003906
	…					

Таким образом, видно, что итерационный процесс медленно сходится к решению (2.25, -1.5).

При необходимости, можно увеличить r и продолжить.

Сведем результаты всех этапов в следующую таблицу.

Таблица 5.7

Общие результаты решения задачи

Этап	r	X_1	X_2	Расхождение с ограничением
1	0.01	-0,44597	3,908188	-10,8
2	0.1	-0,040748	3,098545	-9,5626
3	1	1,3402598	0,35621	-3,67569
4	10	2,170623	-1,12868	-0,53
5	100	2,3595646	-1,22453	-0,056

5.2. Задание к работе

Методом штрафных функций решить задачу на условный экстремум с заданной точностью. В качестве исходных данных вводятся:

- начальная точка;
- точность решения задачи.

На выходе необходимо получить:

- точку максимума или минимума;
- значение функции в данной точке.

В качестве промежуточных результатов вывести решение задачи на безусловный экстремум на каждом шаге. Для этого сформировать следующую таблицу.

Таблица 5.8

Промежуточные результаты для метода штрафных функций

Этап	r	X_1	X_2	$F(x_1,x_2)$	Расхождение с ограничением (штраф)
1	0.01				
2	0.1				
		…			

В качестве метода решения задачи на безусловный экстремум использовать ЛЮБОЙ метод решения задач безусловной оптимизации.

5.3. Варианты заданий

Таблица 5.9

Варианты заданий на лабораторную работу № 5

	Функция и ограничения	Нач. т.	Ответ
1	$f(\bar{x}) = -4x_1^2 - 8x_1 + x_2 + 3 \to \max$ $-x_1 - x_2 = 2$	(-1,-1)	(-1.125; -0.875)
2	$f(\bar{x}) = x_1 - 2x_2^2 + 4x_2 \to \max$ $-3x_1 - 2x_2 = 6$	(1,-1)	(-2.5556,0.8333)
3	$f(\bar{x}) = -8x_1^2 + 4x_1 - x_2^2 + 12x_2 - 7 \to \max$ $2x_1 + 3x_2 = -6$	(-2,3)	(-0.3947;-1.73685)
4	$2x_1^2 - 4x_1x_2 + 3x_2^2 + 1 \to min$ $-7x_1 + x_2 = 3$	(4,4)	(-0.4711; -0.2975)
5	$4x_1^2 - 4x_1 + x_2^2 + 3 \to min$ $x_1 + 3x_2 = 5$	(-1,2)	(0.6216;1.459)
6	$3x_1^2 - x_1x_2 + 5x_2^2 - 8 \to min$ $-x_1 + 2x_2 = 1$	(2,-2)	(-0.2667; 0.3667)
7	$x_1^2 + x_1x_2 + x_2^2 - 10 \to min$ $x_1 + x_2 = 4$	(-1,1)	(2,2)
8	$4x_1^2 - x_1x_2 + 5x_2^2 + 3 \to min$ $-4x_1 + x_2 = -2$	(2,-1)	(0.4875; -0.05)
9	$x_1^2 - 4x_1 + 3x_2^2 - x_2 + 3 \to min$ $x_1 + 3x_2 = 0$	(0,0)	(1.375; -0.4583)

Окончание табл. 5.9

10	$2x_1^2 - 3x_1 + 3x_2^2 + x_2 - 8 \rightarrow min$ $-2x_1 + 5x_2 = 1$	(-1,-1)	(0.4274, 0,371)

ЛАБОРАТОРНАЯ РАБОТА 6

МЕТОД ДИНАМИЧЕСКОГО ПРОГРАММИРОВАНИЯ

Цель работы: Получение навыков решения практических задач методом динамического программирования.

6.1. Краткие теоретические сведения

Метод динамического программирования является одним из наиболее распространенных способов решения сложных оптимизационных задач путем разбиения их на более простые подзадачи.

Ключевая идея в методе динамического программирования заключается в разбиении (как правило, рекурсивном) данной задачи на более мелкие подзадачи, а затем, после решения этих частных подзадач, - в объединении полученных решений в одно целое. В связи с этим возникают ограничения на задачи, для решения которых может применяться метод динамического программирования. Они должны обладать так называемой оптимальной подструктурой. Это означает, что в оптимальном решении задачи содержится оптимальное решение ее частных подзадач.

Исходя из этого, в общем виде метод динамического программирования состоит из следующих этапов:

- разбиение задачи на подзадачи меньшего размера;
- нахождение оптимального решения подзадач (данная процедура осуществляется рекурсивно по данным трем этапам);
- использование полученного решения всех частных подзадач для формирования общего решения задачи.

Метод динамического программирования представляет собой рекурсивный процесс, позволяющий на каждом шаге с помощью некоторой рекуррентной формулы перейти от задачи некоторой размерности N к одной или нескольким задачам меньшей размерности. Данная формула должна быть выведена для каждой конкретной задачи, поскольку при формировании рекуррентного выражения должна учитываться ее специфика.

Рассмотрим решение задачи для некоторых классов задач.

6.1.1 Метод динамического программирования для решения задачи о ранце.

Задача о ранце (о рюкзаке) заключается в выборе из некоторого множества вещей, которые характеризуются весом и ценностью, такого подмножества, которое при выполнении ограничений на размер ранца (рюкзака) обладало бы максимальной ценностью.

Выведем формулу, позволяющую методом динамического программирования решать задачу о ранце. Пусть получено решение об i-1 вещи и решается задача о вещи i. Пусть к настоящему моменту рюкзак имеет вместимость r. На данном этапе есть два варианта:

- вещь i положим в рюкзак;
- вещь i брать не будем.

Рассмотрим суммарную ценность P(i,r) на данном этапе. Она зависит от двух переменных:

- номера вещи i, которая в данный момент рассматривается;
- суммарного объема всех вещей r, которые к настоящему моменту положены в рюкзак.

Если вещь i не брать, то ценность не изменится. Вес вещей также не изменится. Поэтому ценность от вещей 1,2,…, i будет равна ценности вещей 1,2,…, i-1 при сохранении объема r, то есть:

$$P(i,r) = P(i-1,r). \qquad (6.1)$$

Если вещь i в рюкзак положить, то суммарная ценность увеличится на ценность i-й вещи – p_i . При этом, если на шаге i суммарный вес вещей равен r, то на предыдущем шаге он должен равняться r-c_k, где c_k – вес вещи i. Поскольку ценность должна быть максимальной, получим рекурсивную формулу:

$$P(i,r) = max\{P(i-1,r); p_i + P(i-1,r-c_i)\}. \qquad (6.2)$$

Приведем пример решения задачи о ранце для следующих исходных данных. Пусть вес и ценность вещей задана таблицей.

Таблица 6.1

Вес и ценность вещей

Номер	1	2	3	4
Вес	2	1	3	4
Ценность	3	2	4	5

Пусть рюкзак ограничен весом 5 единиц.

Опишем следующую таблицу. По столбцам будем откладывать возможный вес рюкзака, а по строкам – количество предметов.

Таблица 6.2

Основа для решения задачи о ранце методом динамического программирования

Кол. Предм. /R	0	1	2	3	4	5
P(0,r)						
1 P(1,r)						
2 P(2,r)						
3 P(3,r)						
4 P(4,r)						

Очевидно, что первая строка будет нулевой (если ни одну вещь не положили, то и стоимости никакой нет). Она в дальнейшем понадобится для определения ценности на последующих этапах с помощью формулы (1).

Таблица 6.3

Нулевая строка таблицы динамического программирования

Кол. Предм. /R	0	1	2	3	4	5
P(0,r)	0	0	0	0	0	0

Рассмотрим формирование первой ненулевой строки (1 P(1,r)). Она означает, что у нас есть только первый предмет. С учетом того, что вес ранца не может быть отрицательным, при r=0 и r=1 выбора нет – ничего не кладем. В остальных случаях кладем первый предмет и получаем суммарную ценность равную 3.

Таблица 6.4

Нулевая и первая строки решения задачи

Кол. Предм. /R	0	1	2	3	4	5
P(0,r)	0	0	0	0	0	0
1 P(1,r)	0	0	3+P(0,0)=3	3	3	3

Кроме данной информации, необходимо еще запомнить два момента. Первый: кладем или не кладем вещь при данном размере рюкзака, второй – каким образом было получено данное решение.

Сведения о помещении или «не помещении» вещи в рюкзак отмечаем знаками «+» или «-». На данном этапе (поскольку вещь под номером 0 нас не интересует, можно не отмечать, каким образом получено данное решение). Поэтому перепишем фрагмент следующим образом.

Таблица 6.5

Определение пути принятия решения

Кол. Предм. /R	0	1	2	3	4	5
P(0,r)	0	0	0	0	0	0
1 P(1,r)	0 («-»)	0 («-»)	3 («+»)	3 («+»)	3 («+»)	3 («+»)

Рассмотрим формирование ячеек во второй строке. Она (после завершения прямого хода) должна ответить на вопрос о том, класть ли вещь под номером 2 в рюкзак или нет. Столбец 1 означает, что размер рюкзака равен 1. Если ничего не положим, то в результате получим 0. Если положим вещь 2 (ее вес равен 1, следовательно, мы можем это сделать), получим ценность 2. Следовательно, выбирая наибольший из этих элементов, получим число 2 (оно записано в строке 2, в столбце 1). Столбец 2 (объем рюкзака равен 2). Рассмотрим формулу (1). На данном этапе можем либо не положить вещь 2, либо нет. Если вещь 2 не кладем, тогда ценность P(2,r) будет совпадать с ценностью P(1,r), т.е. ценностью, когда положили вещь 1 и объем не увеличился. Это значение равно 3. Во втором случае вещь 2 можно положить. Тогда с учетом того, что вес вещи 2 составляет 1 единицу, общая стоимость рассчитывается как стоимость вещи 2 плюс стоимость ценности для одной вещи и объема рюкзака, равным 2-1=1. Из фрагмента таблицы, полученной на втором этапе, видим, что на пересечении строки 1 и столбца 1 стоит значение 0. Значит:

$$P(2,2) = max\{P(1,2); 2 + P(1,1)\} = \max\{3, 2 + 0\} = 3.$$

Делаем выводы, что:

1) Вещь два не берем (тогда, взяв вещь 1, получим ценность выше);
2) Данное решение получено с помощью решения P(1,2) (т.е. вещь не взяли, следовательно, размер не изменился).

Рассмотрим столбец 3 (емкость равна 3).

Если вещь 2 не положим, то ценность будет определяться ценностью помещения в рюкзак вещи 1 (т.е. значением 3) и не изменившейся емкостью рюкзака r. Вещь 2 можно положить в рюкзак (поскольку емкость 3 позволяет положить обе вещи). В этом случае согласно формуле (1) ценность P(2,3) будет определяться как ценность вещи 2 (т.е. 2) плюс значение P(1,3-1)=P(1,2). Это значение равно 2. Иными словами, если вещь 2 будет положена в рюкзак, то с учетом того, емкость позволяет положить и первую вещь; суммарная ценность равна 5. Итак:

$$P(2,3) = max\{P(1,3); 2 + P(1,2)\} = \max\{3, 2 + 3\} = 5.$$

Делаем выводы, что:

1) вещь под номером 2 берем;
2) данный результат получен с помощью решения P(1,2).

Аналогичным образом получаются значения для остальных ячеек (поскольку новых вещей для данной строки нет). В итоге получим таблицу.

Таблица 6.6

Принятие решения при двух предметах

Кол.-во предм. /R	0	1	2	3	4	5
P(0,r)	0	0	0	0	0	0
1 P(1,r)	0 («-»)	0 («-»)	3 («+»)	3 («+»)	3 («+»)	3 («+»)
2	0(«-»)	2(«+»)	3(«-»)	5(«+»)	5(«+»)	5(«+»)

Здесь стрелками показано, на основании какой ветки в формуле (1) получено данное решение.

Формирование третьей строки (делаем вывод о том, кладем или нет третью вещь).

Столбец 0 – 0.

Столбец 1 (емкость рюкзака равна 1, следовательно, вещь 3 положить не можем). Поэтому

P(3,1)=P(2,1)=3.

Выводы:

1) вещь 3 не кладем;
2) пришли из ячейки P(2,1).

Столбец 2.

Вещь 3 не кладем (как и в предыдущем случае, не позволяет емкость). Следовательно,

P(3,2)=P(2,2)=3.

Выводы:

1) вещь 3 не кладем;
2) пришли из ячейки P(2,2).

Столбец 3. Вещь 3 положить можем. Ищем максимум.

Если вещь 3 не кладем, то P(3,3)=P(2,3)=5 (т.е. оставим вещи 1 и 2 и получим ценность 5).

Если вещь 3 кладем, то

P(3,3)=4+P(2,0)=4.

Вещь 3 выгоднее в данном случае не класть.

Выводы:

1) вещь 3 не кладем;
2) пришли из ячейки P(2,3).

Столбец 4. Ищем максимум.

Если вещь 3 не кладем, то P(3,4)=P(2,4)=5.

Если вещь 3 кладем, то с учетом веса можно положить еще и вещь 1, т.е. P(3,4)=4+P(2,1)=6.

В денном случае вещь выгоднее положить.

Выводы:

1) вещь 3 кладем;
2) Пришли из ячейки P(2,1).

Столбец 5.

В данном случае, если вещь 3 кладем, то в оставшуюся емкость (5-3=2) можно положить вещь 1, которая более ценна. То же самое показывает формула (1):

P(3,5)=4+P(2,2)=4+3=7.

Выводы:

1) вещь 3 кладем;
2) пришли из ячейки P(2,2).

Таблица к данному этапу (табл. 6.7).

Таблица 6.7

Принятие решения при трех предметах

Кол-во предм. /R	0	1	2	3	4	5
P(0,r)	**0**	**0**	**0**	**0**	**0**	**0**
1 P(1,r)	**0 («-»)**	**0 («-»)**	**3 («+»)**	**3 («+»)**	**3 («+»)**	**3 («+»)**
2	**0(«-»)**	**2(«+»)**	**3(«-»)**	**5(«+»)**	**5(«+»)**	**5(«+»)**
3	**0(«-»)**	**2(«-»)**	**3 («-»)**	**5 («-»)**	**6(«+»)**	**7 («+»)**

Последняя четвертая строка. Вещь 4 имеет вес 4, поэтому в столбцы 0,1,2,3 мы ее не кладем из за отсутствия необходимой емкости. В этом случае ценность P(k,r) будет определяться ценностью P(k-1,r).

Последний столбец 5.

Рассмотрим два варианта:

1) Вещь 4 кладем. Тогда P(4,5) будет определяться как 5+P(3,1) (ценность вещи 5, из текущей емкости 5 вычли вес вещи 4, получили 1). Итого, 5+ P(3,1)=5+2=7.

2) Вещь 4 не кладем. Тогда P(4,5)=P(3,5) -7.

Таблица 6.8

Принятие решения при четырех предметах

Кол-во предм. /R	0	1	2	3	4	5
P(0,r)	**0**	**0**	**0**	**0**	**0**	**0**
1 P(1,r)	**0 («-»)**	**0 («-»)**	**3 («+»)**	**3 («+»)**	**3 («+»)**	**3 («+»)**
2	**0(«-»)**	**2(«+»)**	**3(«-»)**	**5(«+»)**	**5(«+»)**	**5(«+»)**
3	**0(«-»)**	**2(«-»)**	**3 («-»)**	**5 («-»)**	**6(«+»)**	**7 («+»)**
4	**0(«-»)**	**2(«-»)**	**3 («-»)**	**5 («-»)**	**6(«+»)**	**7 («+»)**

Итак, суммарная ценность всех вещей, которую можно положить в рюкзак, равна 7. Рассмотрим, какие вещи стоит положить в него. Здесь есть 2 варианта.

Вариант 1. Вещь 4 не кладем. Тогда, двигаясь по строкам, получаем следующие ответы:

Вещь 4 – «-» (перейти к ячейке (3,5) и, исходя из этого, выписать ответ для вещи 3),

Вещь 3 – «+» (по стрелке перейти к ячейке (2,2) и записать ответ),

Вещь 2 – «-» (переходим по стрелке к ячейке (1,2)),

Вещь 1 – «+».

Вариант 2.

Вещь 4 кладем.

Вещь 4 – «+» (перейти к ячейке (3,1)),

Вещь 3 – «-» (перейти к ячейке (2,1)),

Вещь 2 – «+» (перейти к ячейке (1,0)),

Вещь 1 – «-».

Замечание.

Решение данной задачи можно было начать с ячейки (4,5), последовательно находя значения только в тех ячейках матрицы, которые необходимы для расчета. В данном случае на каждом шаге получали бы рекуррентное соотношение, которое можно было бы окончательно посчитать, дойдя до строки 1.

6.1.2. Метод динамического программирования для решения задачи о конвейере [8].

Рассмотрим применение метода динамического программирования для решения задачи о конвейере. Цех оснащен двумя конвейерами. На оба конвейера поступают для сборки детали, после чего на каждом рабочем месте выполняется некоторое обслуживание (например, частичная сборка). На каждом конвейере имеется n рабочих мест, пронумерованных от 1 до N.

На обоих конвейерах на рабочих местах с одинаковыми номерами выполняются одни и те же операции. Однако время выполнения одних и тех же операций на разных конвейерах отличается.

- S_{ij} – рабочее место j на конвейере i;
- a_{ij} –время обслуживания на рабочем месте S_{ij};
- e_i – постановка детали на конвейер i;
- x_i – снятие готовой детали с конвейера i;
- t_{ij} - время, требуемое на перемещение детали с одного конвейера на другой после выполнения операции S_{ij}.

Без ограничения общности, иллюстрация работы такого конвейера с числовыми значениями приведена на *рис. 6.1*.

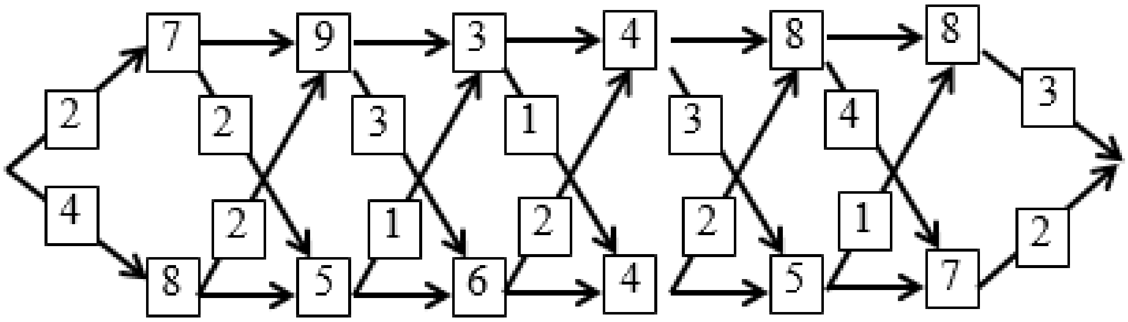

Рис. 6.1. Иллюстрация задачи о конвейере

Необходимо вывести рекуррентную формулу, Выведем рекуррентное соотношение, позволяющее оценивать время для любой стадии.

Введем в рассмотрение функцию f(n) –

Найдем оптимальное время завершения обслуживания детали. Обозначим через $f_i(j)$ – минимально возможное время, которое деталь проходит от начала до рабочего места s_{ij}. Конечная цель заключается в определении кратчайшего времени f^*, в течение которого происходит обслуживание детали на всех этапах. Оно будет определяться следующим образом:

$$f^* = \min\left(f_1(n) + x_1, f_2(n) + x_2\right). \tag{6.3}$$

Здесь $f_1(n)$ – минимально возможное время, которое деталь проходит от начала до последнего рабочего места первого конвейере, а $f_2(n)$ – второго. Таким образом, необходимо определить величины $f_1(n)$ и $f_2(n)$. Выведем рекурсивную и нерекурсивную формулу их определения.

Для первого рабочего места первого и второго конвейеров будут справедливы следующие формулы:

$$f_1(1) = e_1 + a_{11} \tag{6.4}$$

и

$$f_2(1) = e_2 + a_{21}, \tag{6.5}$$

соответственно.

Для остальных рабочих мест будут справедливы следующие формулы:

$$f_1(j) = \min\left(f_1(j-1) + a_{1j}, f_2(j-1) + t_{2j-1} + a_{1j}\right) \tag{6.6}$$

и

$$f_2(j) = \min\left(f_2(j-1) + a_{2j}, f_1(j-1) + t_{1j-1} + a_{2j}\right). \tag{6.7}$$

Проиллюстрируем работу данных формул для задачи, иллюстрация которой представлена на рис.6.1.

Прямой ход будет заключаться в определении минимального времени для каждого из конвейеров.

Для первого рабочего места для каждого конвейера будут использоваться формулы (6.4) и (6.5). Получим:

$$f_1(1) = 2 + 7 = 9,$$
$$f_2(1) = 4 + 8 = 12.$$

Определим время для обработки деталей на каждом из конвейеров на втором этапе (второе рабочее место). Кроме того, будем определять, на каком конвейере деталь была обработана на предыдущей стадии.

$F_1(2)$=min(9+9,12+2+9)=18 (на предыдущей стадии деталь была обработана на конвейере 1);

$f_2(2)$=min(9+9,12+2+9)=18 (на предыдущей стадии деталь была обработана на конвейере 1).

На третьем этапе на каждом из конвейеров время обработки будет следующим:

$f_1(3)$=min(18+3,16+1+3)=20 (на предыдущей стадии деталь была обработана на конвейере 2);

$f_2(3)$=min(16+6,18+3+6)=22 на предыдущей стадии деталь была обработана на конвейере 2).

На четвертом этапе на каждом из конвейеров время обработки будет следующим:

$f_1(4)$=min(20+4,22+2+4)=24 (на предыдущей стадии деталь была обработана на конвейере 1);

$f_2(4)$=min(22+4,20+1+4)=25 (на предыдущей стадии деталь была обработана на конвейере 1).

На пятом этапе получим следующее время обработки на каждом из конвейеров:

f1(5)=min(24+8,25+2+8)=32 (на предыдущей стадии деталь была обработана на конвейере 1);

f2(5)=min(25+5,24+3+5)=30 (на предыдущей стадии деталь была обработана на конвейере 2).

На пятом этапе получим следующее время обработки на каждом из конвейеров:

f1(6)=min(32+4,30+1+4)=35 (на предыдущей стадии деталь была обработана на конвейере 2);

f1(6)=min(30+7,32+4+7)=37 (на предыдущей стадии деталь была обработана на конвейере 2).

С помощью формулы (6.3) определим наименьшее время обслуживания:

Сведем результаты в следующую таблицу.

Таблица 6.9

Решение задачи о конвейере

f1(1)=2+7=9	f2(1)=4+8=12
f1(2)=min(9+9,12+2+9)=18 (к.1)	f2(2)=min(12+5,9+2+5)=16 (к.1)
f1(3)=min(18+3,16+1+3)=20 (к.2)	f2(3)=min(16+6,18+3+6)=22 (к.2)
f1(4)=min(20+4,22+2+4)=24 (к.1)	f2(4)=min(22+4,20+1+4)=25 (к.1)
f1(5)=min(24+8,25+2+8)=32 (к.1)	f2(5)=min(25+5,24+3+5)=30 (к.2)
f1(6)=min(32+4,30+1+4)=35 (к.2)	f1(3)=min(30+7,32+4+7)=37 (к.2)

f*=min(35+3,37+2)=38 (минимум достигнут в случае, если на последней стадии деталь была обработана на конвейере 1).

Обратный ход.

Определим путь прохождения детали, исходя из того, на каком этапе на каком из конвейеров она была обработана. Поскольку минимум был достигнут на конвейере 1, то именно на этом этапе будет использован конвейер 1. Далее переходим к соответствующей ячейке таблицы (первый столбец, шестая строка). На данное рабочее место деталь попала с конвейера 2. Переходим к пятому этапу (второй столбец, пятая строка). Видим, что на четвертом этапе деталь также должна быть обработана на конвейере 2. Переходим к четвертому этапу (второй столбец, четвертая строка). Из таблицы следует, что на третьем этапе деталь была обработана на конвейере 1 (первый столбец, третья строка). Здесь видно, что на втором этапе деталь необходимо обработать на конвейере 2 (переходим ко второму столбцу второй ячейки). Исследуя соответствующую ячейку видно, что на первой стадии деталь должна быть обработана на первом конвейере. Подытоживая полученные результаты, получим:

Таблица 6.10

Выбор рабочих мест для задачи о конвейере

Рабочее место (этап)	Конвейер
1	К.1
2	К.2
3	К.1
4	К.2
5	К.2
6	К.1

6.1.3. Метод динамического программирования для оптимального разбиения отрезка

Пример. Рассмотрим проблему оптимального разбиения некоторого отрезка [A,B] на участки, которая называется задачей о ближайшем соседе. Она может быть сформулирована следующим образом. Дано целое положительное число M и неотрицательная функция f(x,y), которая отражает затраты, связанные с «обслуживанием» отрезка [x,y]⊆[0,M]. Требуется разбить отрезок [0,M] на n частей таким образом, чтобы суммарные затраты, соответствующие этому разбиению, были минимальны. Математическую постановку задачи можно описать следующим образом:

$$\begin{cases} \sum_{k=1}^{n} f(x_{k-1}, x_k) \to min; \\ 0 = x_0 \le x_1 \le \cdots \le x_n = M \end{cases} \quad (6.8)$$

Пусть Sn(M) – оптимальное значение целевой функции задачи (6.8). Обозначим эту задачу через <n,M>. Пусть $S_k(y)$ – оптимальное значение целевой функции задачи <k,y>. Справедливы рекуррентные соотношения:

$$S_k(y) = \begin{cases} f(0, y), k = 1, y = 1, \dots, M. \\ \min_{0 \le x \le y}\left(S_{k-1}(x) + f(x, y)\right), k = 2, \dots, n, y = 1, \dots, M. \end{cases} \quad (6.9)$$

Пример. Пусть требуется разбить отрезок [0,8] на 4 части таким образом, чтобы минимизировать суммарные затраты. Пусть целевая функция f(x,y) задана следующей таблицей (табл. 6.11).

Таблица 6.11

Целевая функция для задачи о разбиении

	1	2	3	4	5	6	7	8
0	3	19	24	41	42	63	66	83
1	0	6	18	26	39	48	56	77
2	-	0	11	19	35	44	55	56
3	-	-	0	13	25	27	45	53
4	-	-	-	0	3	15	24	37
5	-	-	-	-	0	3	16	27
6	-	-	-	-	-	0	12	21
7	-	-	-	-	-	-	0	16
8	-	-	-	-	-	-	-	0

Найдем сначала $S_1(y)$, пользуясь прямой формулой (6.9) (без рекурсии).
$S_1(0)=0$; $S_1(1)=3$; $S_1(2)=19$; $S_1(3)=24$; $S_1(4)=41$; $S_1(5)=42$; $S_1(6)=63$; $S_1(7)=66$; $S_1(8)=83$.
Выпишем значения S_1 в следующую таблицу.

Таблица 6.12

Результаты после первого этапа

y	0	1	2	3	4	5	6	7	8
S_1(y)	0	3	19	24	41	42	63	66	83

Далее будем искать $S_2(y)$ через $S_1(y)$.

$$S_2(y) = \min_{0 \le x \le y} \big(S_1(x) + f(x,y)\big), y = 1, \dots, M. \qquad (6.10)$$

$S_2(1)$=min($S_1(0)$+f(0,1))=3 (нельзя сформировать другие отрезки, заканчивающиеся 1);

$S_2(2)$=min($S_1(0)$+f(0,2); $S_1(1)$+f(1,2))=min(0+19; 3+6)=9 (запомним, что минимум достигается через точку 1);

$S_2(3)$=min($S_1(0)$+f(0,3); $S_1(1)$+f(1,3); $S_1(2)$+f(2,3))=min(0+24; 3+18; 19+11)=21 (запомним, что минимум достигается через точку 1);

$S_2(4)$=min($S_1(0)$+f(0,4); $S_1(1)$+f(1,4); $S_1(2)$+f(2,4); $S_1(3)$+f(3,4))=min(0+41; 3+26; 19+19; 24+13)=29 (запомним, что минимум достигается через точку 1);

$S_2(5)$=min($S_1(0)$+f(0,5); $S_1(1)$+f(1,5); $S_1(2)$+f(2,5); $S_1(3)$+f(3,5); $S_1(4)$+f(4,5))=min(0+42; 3+26; 19+19; 24+13; 41+3)=42 (запомним, что минимум достигается через точку 0);

$S_2(6)$=min($S_1(0)$+f(0,6); $S_1(1)$+f(1,6); $S_1(2)$+f(2,6); $S_1(3)$+f(3,6); $S_1(4)$+f(4,6); $S_1(5)$+f(5,6))=min(0+63; 3+48; 19+44; 24+27; 41+15; 42+3)=45 (запомним, что минимум достигается через точку 5);

$S_2(7)$=min($S_1(0)$+f(0,7); $S_1(1)$+f(1,7); $S_1(2)$+f(2,7); $S_1(3)$+f(3,7); $S_1(4)$+f(4,7); $S_1(5)$+f(5,7), $S_1(6)$+f(6,7))=min(0+66; 3+56; 19+55; 24+45; 41+24; 42+16; 66+12)=58 (запомним, что минимум достигается через точку 5);

$S_2(8)$=min($S_1(0)$+f(0,8); $S_1(1)$+f(1,8); $S_1(2)$+f(2,8); $S_1(3)$+f(3,8); $S_1(4)$+f(4,8); $S_1(5)$+f(5,8), $S_1(6)$+f(6,8), $S_1(7)$+f(7,8))=min(0+83; 3+77; 19+56; 24+53; 41+37; 42+27; 66+21; 83+16)=69 (запомним, что минимум достигается через точку 5).

Дополним предыдущую таблицу столбцом, содержащим значения S_2.

Таблица 6.13

Результаты после второго этапа

y	0	1	2	3	4	5	6	7	8
$S_1(y)$	0	3	19	24	41	42	63	66	83
$S_2(y)$	0	3(0)	9(1)	21(1)	29(1)	42(0)	45(5)	58(5)	69(5)

В скобках указано, через какое значение достигается данный минимум.

Перейдем к третьему отрезку.

$$S_3(y) = \min_{0 \le x \le y} \big(S_2(x) + f(x,y)\big), y = 1, \dots, M. \qquad (6.11)$$

$S_3(1)$=min($S_2(0)$+f(0,1))=3 (нельзя сформировать другие отрезки, заканчивающиеся 1);

$S_3(2)$=min($S_2(0)$+f(0,2); $S_2(1)$+f(1,2))=min(0+19; 3+6)=9 (запомним, что минимум достигается через точку 1);

$S_3(3)$=min($S_2(0)$+f(0,3); $S_2(1)$+f(1,3); $S_2(2)$+f(2,3))

и т.д.

Таблица 6.14

Результаты после третьего этапа

y	0	1	2	3	4	5	6	7	8
$S_1(y)$	0	3	19	24	41	42	63	66	83
$S_2(y)$		3(0)	9(1)	21(1)	29(1)	42(0)	45(5)	58(5)	69(5)
$S_3(y)$		3(0)	9(1)	20(2)	28(2)	32(4)	44(4)	53(4)	65(2)

Последний шаг прямого хода метода динамического программирования выполняется аналогично. Сведем результаты в следующую таблицу.

Таблица 6.15

Результаты после четвертого этапа

y	0	1	2	3	4	5	6	7	8
$S_1(y)$	0	**3**	19	24	41	42	63	66	83
$S_2(y)$		3(0)	9(1)	21(1)	**29(1)**	42(0)	45(5)	58(5)	69(5)
$S_3(y)$		3(0)	9(1)	20(2)	28(2)	**32(4)**	44(4)	53(4)	65(2)
$S_4(y)$		3(0)	9(1)	20(2)	28(2)	31(4)	35(5)	48(5)	**59**(**5**)

Обратный ход метода динамического программирования.

На последнем шаге прямого хода найдены минимальные затраты S* =S_4(8)=59, соответствующие оптимальному разбиению исходного отрезка [0,8], и оптимальное значение последней точки разбиения $x_3^* = 5$. Следовательно, теперь отрезок [0,5] необходимо оптимально разбить на три части. Значение x_2^* получим как условно оптимальное решение, соответствующее величине S_3(5) из таблицы. Это будет точка $x_2^* = 4$. Теперь необходимо найти последнюю точку, которая будет разбивать отрезок [0,4]. Согласно значению, находящемуся в строке S_2(y) и в строке 4, видим, что следующий столбец будет 1. Следовательно, отрезок [0,4] должен быть разбит точкой $x_1^* = 1$.

Таким образом, ответ у задачи будет следующий. Отрезок [0,8] сможет быть разбит на 4 отрезка следующим образом: [0,1], [1,4]; [4,5]; [5,8]. При этом суммарные затраты будут равны 59 единиц.

Задание к работе

Решить методом динамического программирования задачу, согласно варианту. Отчет должен содержать:
- постановку задачи;
- алгоритм ее решения;
- результат реализации метода (программную форму с входными данными и результатами);
- выводы.

6.3. Варианты заданий

1. Методом динамического программирования решить задачу о ранце. Имеется n вещей, каждая из которых характеризуется весом и ценностью, а также рюкзак, позволяющий положить вещи, суммарный вес которых не будет превышать R единиц. Выбрать вещи, обладающие наибольшей стоимостью при ограничении на вес рюкзака.

2. Методом динамического программирования решить задачу о конвейере (случай двух параллельных линий, см. п. 6.1.2) . Исходные данные:

- количество операций (этапов);
- временные характеристики (помещение/снятие с каждого конвейера, длительность обработки на данном конвейере данной операции, время для перехода на другой конвейер после данной операции с данного конвейера).

На выходе получить номер конвейера для каждой операции.

3. Методом динамического программирования решить задачу о конвейере (случай трех параллельных линий; расширить формулу, приведенную в п. 6.1.2). Исходные данные:

- количество операций (этапов);
-временные характеристики (помещение/снятие с каждого конвейера, длительность обработки на данном конвейере данной операции, время для перехода на другой конвейер после данной операции с данного конвейера).

На выходе получить номер конвейера для каждой операции.

4. Методом динамического программирования решить задачу о кратчайшем пути в графе.

5. Методом динамического программирования решить задачу о ближайшем соседе (задача оптимального разбиения отрезка на участки).

ЛАБОРАТОРНАЯ РАБОТА 7

ЖАДНЫЕ АЛГОРИТМЫ

Цель работы: изучение особенностей разработки жадных алгоритмов, а также получение практических навыков программной реализации данных алгоритмов.

7.1. Краткие теоретические сведения

Жадные алгоритмы – это разновидность эвристических алгоритмов, в которых на каждом шаге делается выбор, который является наилучшим в данный момент. Другими словами, производится локально оптимальный выбор в надежде, что он приведет к оптимальному решению глобальной задачи.

Жадные алгоритмы не всегда приводят к оптимальному решению, но во многих задачах они дают приемлемый результат. Основным преимуществом жадных алгоритмов является их низкая вычислительная сложность (по сравнению с точными алгоритмами). В качестве недостатка необходимо отметить возможность получения в ряде случаев решений, далеких от оптимальных.

Основное свойство жадного алгоритма: глобальное оптимальное решение можно получить, делая локальный оптимальный (жадный) выбор.

7.2. Задание к работе

Разработать жадный алгоритм для решения задачи, согласно варианту.

Отчет должен содержать:
- постановку задачи;
- алгоритм ее решения;
- результат реализации метода (программную форму с входными данными и результатами);
- выводы.

7.3. Варианты заданий

1. Задача о выборе заявок (мероприятий)

Имеется n мероприятий, ценность которых идентична для некоторого лица. Каждое мероприятие характеризуется началом, и длительностью. Необходимо составить график посещения мероприятий таким образом, чтобы посетить как можно больше мероприятий.

Указать общее решение задачи и продемонстрировать его на конкретных исходных данных (задать время T, количество мероприятий n, время начала i-го мероприятия, время окончания i-го мероприятия)

Исходные данные: см. выше

Выходные данные: график посещения мероприятий

2. Задача о сапожнике

В обязанности военного сапожника входит быстро чинить принесенную обувь. Начальство оценивает сапожника по количеству починенной обуви независимо от того, насколько трудоемок ремонт. Как сапожнику организовать свою работу, чтобы выглядеть в глазах начальства наиболее выигрышно?

Исходные данные – количество сапог, время, необходимое для починки i-й пары сапог, общее время T (период, за который оценивается работа сапожника).

Цель – починить максимальное количество пар.

3. Задача о минимальном покрытии

Реализовать жадный алгоритм для задачи о покрытии. На прямой расположены отрезки [ai, bi], 1,…,n. Все вместе они полностью (возможно с избытком) покрывают отрезок [1,100]. Необходимо выбрать из них наименьше количество таким образом, чтобы они продолжали покрывать отрезок [1,100].

4. Задача о выполнении заданий

Имеется множество заданий $S=\{S_1,\dots,S_n\}$, в котором для выполнения задания a_i требуется p_i единиц времени. Имеется устройство, которое в каждый момент времени может выполнять лишь одно задание. Пусть c_i – время завершения задания i. Необходимо определить время начала работ таким образом, чтобы минимизировать величину $\sum_{i=1}^{n} \frac{c_i}{n}$. Например, если имеются два задания a_1 с временем выполнения $p_1=3$ и a_2 с временем выполнения $p_2=5$, то если первым выполнить задание a_2, а вторым – задание a_1, то $c_1=5$; $c_2=8$; среднее время завершения равно (5+8)/2.

ЛАБОРАТОРНАЯ РАБОТА 8

ЭВРИСТИЧЕСКИЕ АЛГОРИТМЫ

Цель работы: освоение практических навыков решения задач дискретной оптимизации метаэвристисескими методами (генетическим методом и методом муравьиных колоний).

Освоение теоретических навыков разработки эвристических и метаэвристических алгоритмов, а также получение практических навыков программной реализации данных алгоритмов.

8.1. Краткие теоретические сведения

Под эвристическим алгоритмом понимают алгоритм решения задачи, включающий практический метод, не являющийся гарантированно точным или оптимальным, но достаточный для решения поставленной задачи. Использование такого алгоритма позволяет ускорить решение задачи в тех случаях, когда точное решение не может быть найдено.

Эвристические алгоритмы широко применяются для решения задач, обладающих высокой вычислительной сложностью. В таких случаях вместо полного перебора вариантов, занимающего существенное время, а иногда технически невозможного, применяется значительно более быстрый, но недостаточно обоснованный теоретически алгоритм, позволяющий получить решение, которое, как правило, близко к оптимальному.

Тем не менее, при использовании эвристических алгоритмов необходимо понимать, что он не гарантирует наилучшего решения и в некоторых случаях может дать результат, далекий от оптимального. Тем не менее, в большинстве случаев эвристические алгоритмы дают приемлемое и по времени и по «качеству» решение.

Метаэвристика – это метод оптимизации, многократно использующий простые правила или эвристики для достижения решения, близкого к оптимальному.

Метаэвристики пытаются объединить основные эвристические методы в рамках алгоритмических схем более высокого уровня, направленных на эффективное изучение пространства поиска.

Рассмотрим следующие метаэвристические алгоритмы.

8.1.1. Генетический алгоритм

Генетический алгоритм это метаэвристический алгоритм, используемый для решения оптимизационных задач путём случайного подбора, комбинирования и вариации искомых параметров с использованием механизмов, аналогичных естественному отбору в природе. Является разновидностью эволюционных вычислений, с помощью которых решаются оптимизационные задачи с использованием методов естественной эволюции, таких как наследование, мутации, отбор и скрещивание.

При инициализации алгоритма задается множество особей, представляющих текущую популяцию. Как правило, и количество особей и сами особи генерируются случайным образом. Каждая из них оценивается с помощью целевой функции, которая в данном случае называется функцией приспособленности.

Из текущей популяции выбираются особи, к которым применяются так называемые генетические операторы, такие как скрещивание и мутация. В результате этого получается новое поколение особей, которое также оценивается в помощью функции приспособленности. Наилучшее решение функции для данной популяции запоминается как эталонное, которое на следующих стадиях сравнивается с наилучшим. Алгоритм заканчивает свою работу в случае, если в течение нескольких стадий не удалось улучшить решение или если проведено заданное число итераций.

Таким образом, можно выделить следующие этапы генетического алгоритма:

1) задать целевую функцию (приспособленности) для особей популяции;
2) создать начальную популяцию;
3) Начало цикла:
   3.1) размножение (скрещивание);
   3.2) мутирование;
   3.3) вычислить значение целевой функции для всех особей;
   3.4) формирование нового поколения (селекция).

Если выполняются условия остановки, то конец цикла, иначе перейти к п.3.1.

Рассмотрим стадии генетического алгоритма более подробно. Содержание хромосомы (размер, тип и т.д.) будет зависеть от специфики задания. Например, для задач о ранце хромосома представляет собой бинарный вектор, каждый элемент которого равен 0, если вещь не кладем в рюкзак, и 1 в противном случае. Для задач на графах (поиск кратчайшего пути, задача о коммивояжере) каждая хромосома представляет собой один их путем (т.е. один из вариантов решения задачи).

После создания популяции осуществляется процедура скрещивания. Примером простейшего скрещивания является одноточечное скрещивание. В этом случае случайным (или заданным) образом определяется некоторая точка в массиве каждой хромосомы, и та часть, которая левее данной точки, берется, например, от родителя А, а та, которая правее – от родителя В. Графическая иллюстрация одноточечного скрещивания приведена на *рис. 8.1*.

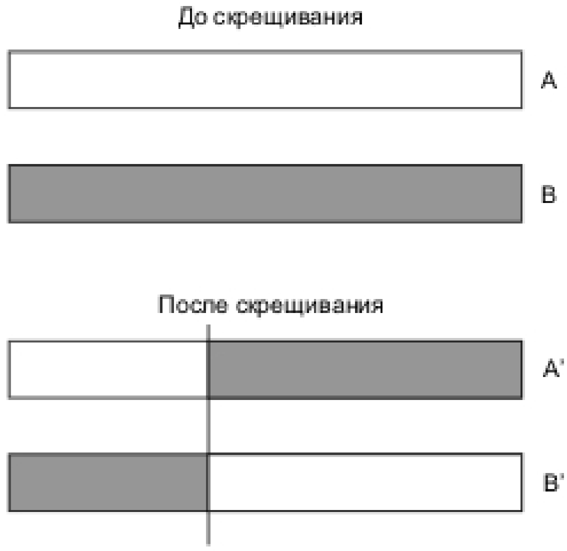

Рис. 8.1. Иллюстрация одноточечного скрещивания

Для двухточечного скрещивания в массиве потомка определяются уже две точки. В результате получим три части массива, две из которых будут наследоваться, например, от родителя а, а одна – от родителя В. Пример двухточечного скрещивания приведен на *рис. 8.2*.

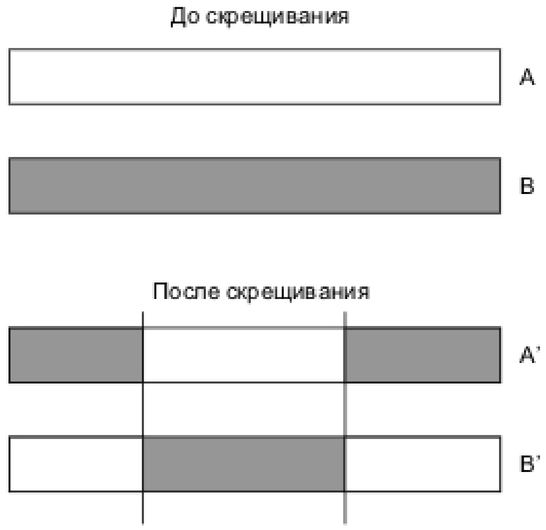

Рис. 8.2. Двухточечное скрещивание

Существует ещё два способа формирования родительской пары – инбридинг и аутбридинг. Оба эти метода основаны на понятии схожести особей. Под инбридингом понимается такой метод, когда первый родитель выбирается случайно, а вторым с большей вероятностью будет максимально близкая к нему особь. Аутбридинг же, наоборот, формирует брачные пары из максимально далеких особей. К преимуществам данных подходов можно отнести возможность их применения для решения многоэкстремальных задач. Однако их влияние на поведение генетического алгоритма не одинаково. Инбридинг характеризуется тем, что при данном способе поиск концентрируется в локальных узлах. Это приводит к разбиению популяции на отдельные локальные группы вокруг подозрительных на экстремум участков. Аутбридинг предупреждает сходимость алгоритма к уже найденным решениям, работая с новыми, неисследованными областями.

С помощью мутации создаются такие цепочки генов, которые не входили в первое поколение, но были необходимы для получения правильного решения. В тоже время она создает бесполезные и «вредные» цепочки, что затрудняет поиск решения. Необходимо правильно пообобрать настройки мутации, чтобы она и помогала и мало мешала поиску решения.

Существует множество разных алгоритмов мутаций. Среди них наиболее распространенными являются следующие:

- инверсия гена;
- замена гена на случайный;
- замена гена на соседний.

В начале, каждой из трёх процедур, реализующих перечисленные алгоритмы, хромосомам присваиваются произвольные вероятности мутации. И только если эта вероятность больше 80%, хромосома мутирует. Рассмотрим отдельно каждый алгоритм мутации.

Формирование нового поколения может осуществляться разными способами, но наиболее распространенными являются следующие два:

- элитный;
- отбор с вытеснением.

При элитном отборе в новое поколение попадают наилучшие с точки зрения целевой функции как предков, таки потомков. Для такого алгоритма характерна, как правило, достаточно быстрая сходимость.

Отбор с вытеснением подразумевает, что новую популяцию будут составлять лишь потомки (отобранные из множества потомков с точки зрения целевой функции).

8.1.2. Метод муравьиных колоний (муравьиные алгоритмы)

Метод муравьиных колоний представляет собой вероятностную жадную эвристику, где вероятности устанавливаются, исходя из информации о качестве решения, полученной из предыдущих решений.

Идея муравьиного алгоритма заключается в имитации поведения муравьёв, связанного с их способностью быстро находить кратчайший путь от муравейника к источнику пищи и адаптироваться к изменяющимся условиям, находя новый кратчайший путь. При своём движении муравей метит путь феромоном, и эта информация используется другими муравьями для выбора пути. Это элементарное правило поведения и определяет способность муравьёв находить новый путь, если старый оказывается недоступным. Схематично поведение муравьев можно представить на *рис. 8.3*.

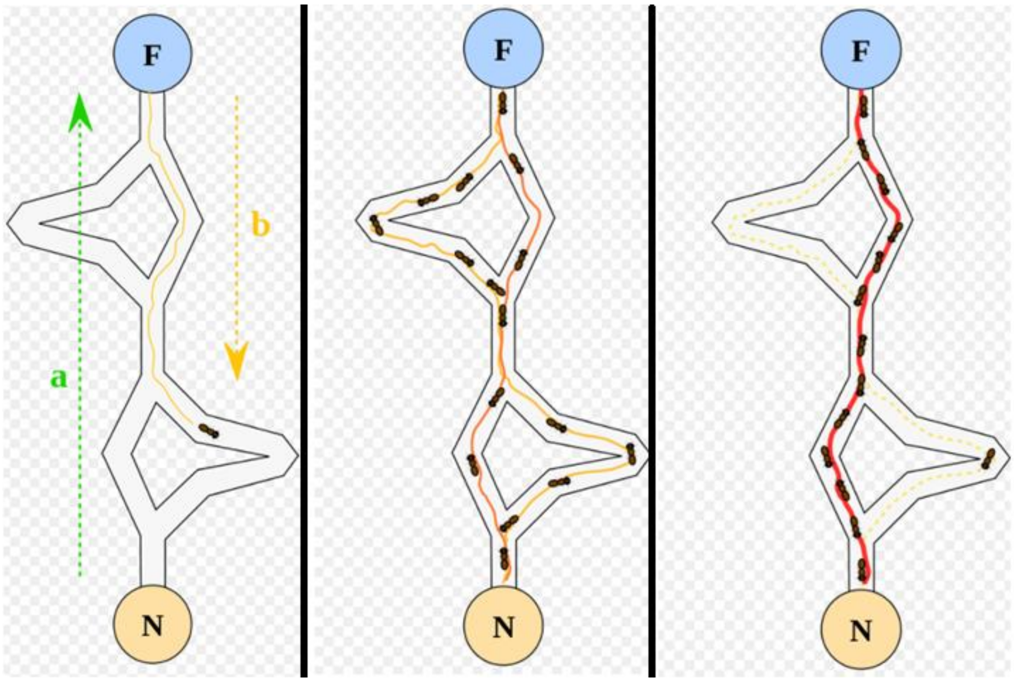

Рис. 8.3. Поведение муравьев

На этом рисунке видно, что сначала один муравей нашел некоторый путь к цели F и вернулся обратно в исходный узел N. Далее (средний рисунок) некоторые муравьи нашли более короткий путь, на котором они оставляют следы, и постепенно (правый рисунок) именно данный путь становится актуальным, и подавляющее большинство муравьев используют именно его.

Муравьиные алгоритмы в общем виде состоят из следующих этапов.

Первый этап – это, фактически инициализация алгоритма. Он заключается в создании множества муравьев, количество которых, как правило, случайное и зависит от специфики задачи. Кроме того, на данном этапе необходимо инициализировать все параметры, на основании которых в дальнейшем будут рассчитаны вероятности выбора каждого отрезка пути. На данном этапе эти параметры задаются случайным образом.

Второй этап – это поиск решения, заключающийся в определении наилучшего пути к цели, который на каждом этапе определяется вероятностью выбора того или иного отрезка пути. Данная вероятность рассчитывается по формуле:

$$P_{ij,k}(t) = \frac{(\tau_{ij})^{\alpha} \cdot (\eta_{ij})^{\beta}}{\sum_{l \in J_{i,k}} (\tau_{il})^{\alpha} \cdot (\eta_{il})^{\beta}}. \qquad (8.1)$$

где τ_{ij} – уровень феромона; η_{ij} – величина, обратная расстоянию между узлами i и j; α и β - параметры (константы).

При α=0 муравьиный алгоритм превращается в обычный жадный алгоритм, поскольку в формуле (8.1) будет учитываться лишь расстояние, и не будет учитываться опыт других муравьев. При β=0 получается другой крайний случай, когда во внимание берется лишь предыдущий опыт, а расстояние между узлами неважно.

Оба этих крайних случая, как правило, дают результаты, которые зачастую далеки от оптимальных.

Уровень феромона обновляется в соответствии с формулой:

$$\tau_{ij}(t+1) = (1-\rho)\tau_{ij}(t) + \frac{Q}{L_k(t)} \tag{8.2}$$

Здесь ρ - интенсивность испарения; $L_k(t)$ – цена текущего решения для муравья k; Q – параметр, представляющий собой значение порядка цены оптимального решения. Таким образом, $\frac{Q}{L_k(t)}$ – феромон, откладываемый муравьем k.

8.1.3. Задачи, для которых можно применить эволюционные алгоритмы

Транспортная задача.

Рассмотрим простейший вариант транспортной задачи с ограничением по времени. Имеются некоторое количество транспортных средств, один склад (депо) и некоторое количество клиентов [1]. Задача заключается в поиске эффективного маршрута для транспортных средств, обслуживающих определенное количество клиентов. При этом все маршруты должны начинаться и заканчиваться в депо. Рисунок 1 иллюстрирует пример такого маршрута. Точка 0 – депо, точки 1–10 – клиенты.

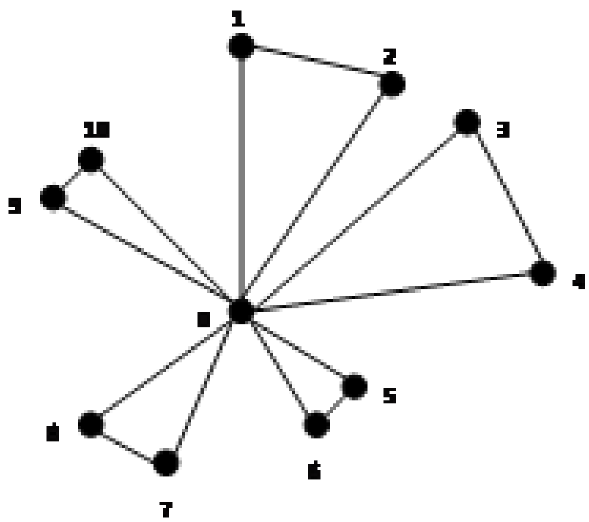

Рис. 8.4. Иллюстрация к транспортной задаче

Данную задачу можно решать с точки зрения следующих целевых функций:

- Минимизировать общее количество транспортных средств (V – количество идентичных автомобилей грузоподъемностью q), необходимых для удовлетворения всех потребностей клиентов;

- Минимизировать общее расстояние, пройденное всеми транспортными средствами.

На маршруты должны налагаться следующие ограничения:

- каждый клиент должен быть обслужен только одним транспортным средством;

- транспортное средство не может обслужить больше клиентов, чем позволяет его вместимость;

- возможны временные ограничения (прибытие автомобиля к клиенту должно быть в пределах заданного временного окна).

В частности, генетический алгоритм решения данной задачи будет основан на следующих операциях:

- инициализация (основной задачей оператора инициализации является генерация маршрутов таким образом, чтобы они были корректны);

- оператор скрещивания. Он работает таким образом, чтобы не только получить новые маршруты, но и проверить их на корректность;

- оператор мутации предназначен для вывода популяции из локального оптимума. Он получает на вход хромосому и с некоторой вероятностью инвертирует часть ее генов.

Рассмотрим данные операторы более подробно.

Целью оператора инициализации является получение множества маршрутов. Таким образом, каждая отдельная хромосома представляет собой отдельный маршрут. С математической точки зрения такой маршрут представляет собой связный граф, состоящий из связных подграфов, каждый из которых дважды соединен с центром (отрезок «выезда» со склада и «въезда в него»). В некоторых случаях обратно можно добраться по той же дороге, однако это, скорее всего, приведет к увеличению длины пути.

Оператор скрещивания позволяет получить новый маршрут на основании двух заданных маршрутов. Исходными для оператора скрещивания являются два маршрута A и B (родительские маршруты). В качестве скрещивания можно использовать двухточечное скрещивание, приведенное на рис. 8.2, дополнительно проверив, что в новых маршрутах-потомках будут присутствовать все точки, причем по одному разу.

Оператор мутации изменяет порядок пунктов в маршруте. В этом случае будет выполнено требование однократного посещения каждого из пунктов.

Задача о коммивояжере

Решим муравьиным алгоритмом задачу о коммивояжере. Без ограничения общности возьмем $\alpha=1$; $\beta=1$. Пусть исходная матрица расстояний между городами имеет вид, представленный в табл. 8.1.

Таблица 8.1

Матрица расстояний

	1	2	3	4	5
1	×	38	74	59	45
2	38	×	46	61	72
3	74	46	×	49	85
4	59	61	49	×	42
5	45	72	85	42	×

Пусть в начале работы алгоритма сгенерированы случайный числа, показывающие уровень феромона τ (в данном примере он для разных ребер будет различным, можно при инициализации взять все начальные значения τ одинаковыми). Результаты работы датчика случайных чисел занесены в следующую таблицу.

Таблица 8.2

Начальные значения τ

	1	2	3	4	5
1	×	3	2	2	2
2	3	×	1	1	1
3	2	1	×	2	2
4	2	1	2	×	1
5	2	1	2	1	×

Пусть некоторый муравей находится в первом городе. Определим вероятности его перемещения в другие города. Для этого воспользуемся формулой (8.1).

$$P_{12} = \frac{3 \cdot \left(\frac{1}{38}\right)}{3 \cdot \left(\frac{1}{38}\right) + 2 \cdot \left(\frac{1}{74}\right) + 2 \cdot \left(\frac{1}{59}\right) + 2 \cdot \left(\frac{1}{45}\right)} = 0.4283$$

$$P_{13} = \frac{2 \cdot \left(\frac{1}{74}\right)}{3 \cdot \left(\frac{1}{38}\right) + 2 \cdot \left(\frac{1}{74}\right) + 2 \cdot \left(\frac{1}{59}\right) + 2 \cdot \left(\frac{1}{45}\right)} = 0.1466$$

$$P_{14} = \frac{2 \cdot \left(\frac{1}{59}\right)}{3 \cdot \left(\frac{1}{38}\right) + 2 \cdot \left(\frac{1}{74}\right) + 2 \cdot \left(\frac{1}{59}\right) + 2 \cdot \left(\frac{1}{45}\right)} = 0.184$$

$$P_{15} = \frac{2 \cdot \left(\frac{1}{45}\right)}{3 \cdot \left(\frac{1}{38}\right) + 2 \cdot \left(\frac{1}{74}\right) + 2 \cdot \left(\frac{1}{59}\right) + 2 \cdot \left(\frac{1}{45}\right)} = 0.2411$$

Далее необходимо с полученными вероятностями выбрать тот или иной путь. Для этого:

- сгенерируем случайное число в пределах от 0 до 1;
- если оно меньше или равно 0.4283, то выбираем маршрут 1->2;
- если оно больше, чем 0.4283, но меньше, чем 0.4283+0.1466=0.5749, то выбираем маршрут 1->3;
- если значение больше, чем 0.5749, но меньше, чем 0.5749+0.184=0.7589, то выбираем маршрут 1->4;
- в противном случае выбираем маршрут 1->5.

Фактически отрезки полученных вероятностей уложили в отрезке [0,1] для того, чтобы определить вероятность выпадения каждого из значений.

Аналогичным образом рассчитаем вероятности выхода из остальных точек и определим вероятности попадания во все пункты из пункта 2,3,4 и 5.

Пусть без ограничения общности генератор случайных чисел выдал число 0.68. Это означает, что на данном этапе данный маршрут будет включать путь 1->4. Выделим данный путь на графе (рис. 8.5).

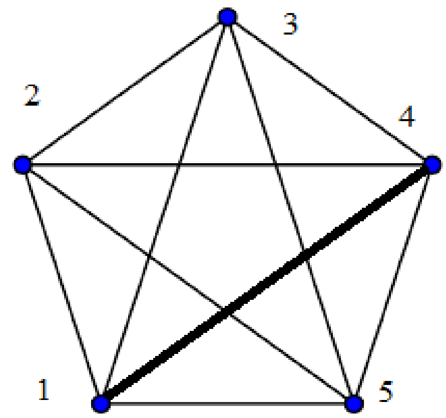

Рис. 8.5. Формирование маршрута. Итерация 1

Вычеркнем город 1 из списка городов, которые необходимо посетить. Рассчитаем вероятности переезда в любые города из четвертого (кроме первого).

$$P_{42} = \frac{1 \cdot \left(\frac{1}{61}\right)}{1 \cdot \left(\frac{1}{61}\right) + 2 \cdot \left(\frac{1}{49}\right) + 1 \cdot \left(\frac{1}{42}\right)} = 0.2023$$

$$P_{43} = \frac{2 \cdot \left(\frac{1}{49}\right)}{1 \cdot \left(\frac{1}{61}\right) + 2 \cdot \left(\frac{1}{49}\right) + 1 \cdot \left(\frac{1}{42}\right)} = 0.5038$$

$$P_{45} = \frac{1 \cdot \left(\frac{1}{42}\right)}{1 \cdot \left(\frac{1}{61}\right) + 2 \cdot \left(\frac{1}{49}\right) + 1 \cdot \left(\frac{1}{42}\right)} = 0.2938$$

Аналогичным образом осуществляем выбор следующего пути маршрута:

- сгенерируем случайное число в пределах от 0 до 1;
- если оно меньше или равно 0.2023, то выбираем маршрут 4->2;
- если оно больше, чем 0.2023, но меньше, чем 0.2023+0.5038=0.7061, то выбираем маршрут 4->3;
- если значение больше, чем 0.5749, но меньше, чем 0.5749+0.184=0.7589, то выбираем маршрут 4->5;
- в противном случае выбираем маршрут 1->5.

Пусть без ограничения общности выпало число 0.33. Это означает, что на данном этапе будет выбран путь 4->3 (рис. 8.6).

Вычеркиваем город 4 из списка городов, которые необходимо посетить, - следовательно, из города 3 можно переехать лишь в города 2 и 5. Рассчитаем вероятности согласно формуле 8.1.

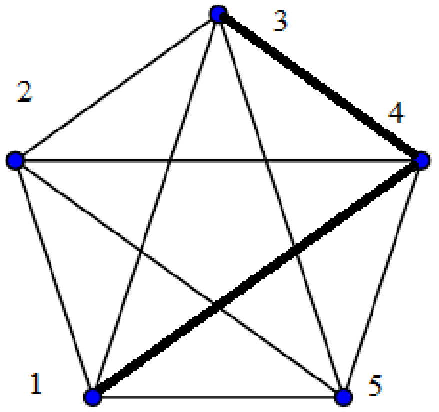

Рис. 8.6. Формирование маршрута. Итерация 2

$$P_{32} = \frac{1\cdot\left(\frac{1}{46}\right)}{1\cdot\left(\frac{1}{46}\right)+2\cdot\left(\frac{1}{85}\right)} = 0.4802,$$
$$P_{35} = \frac{2\cdot\left(\frac{1}{85}\right)}{1\cdot\left(\frac{1}{46}\right)+2\cdot\left(\frac{1}{85}\right)} = 0.5198.$$

Генерируем случайное число. Если оно меньше, чем 0.4802, то едем из третьего города во второй; если больше, то - в пятый.

Пусть сгенерировали число 0.24. Следовательно, переходим из города 3 в город 2. После этого остается единственный маршрут: из 2 в 5. После этого возвращаемся из пятого города в первый.

Таким образом, сгенерирован случайный маршрут: 1 ->4->3->2->5->1

Далее рассчитаем длину полученного маршрута. Она будет равна:

L_0=59+49+46+72+45=271.

Теперь рассчитаем новый феромон, который будем использовать на следующем шаге по формуле (8.2). Пусть ρ=0.1 (интенсивность испарения).

Таким образом, после данного этапа у ребер, по которым муравей прошел, уровень феромона должен увеличиться (это будет зависеть от ρ и Q), а у остальных, наоборот, уменьшиться.

Рассчитаем сначала τ_{14}, τ_{43}, τ_{32}, τ_{25}, τ_{51}.

Выберем для данного примера значение Q=100. Получим:

$$\tau_{14}(2) = (1-\rho)\tau_{14}(1) + \frac{100}{271} = 0.9 \cdot 2 + 0{,}369 = 2.169$$

$$\tau_{43}(2) = (1-\rho)\tau_{43}(1) + \frac{100}{271} = 0.9 \cdot 2 + 0.369 = 2.169$$

$$\tau_{32}(2) = (1-\rho)\tau_{32}(1) + \frac{100}{271} = 0.9 \cdot 1 + 0.369 = 1.269$$

$$\tau_{25}(2) = (1-\rho)\tau_{25}(1) + \frac{100}{271} = 0.9 \cdot 1 + 0.369 = 1.269$$

$$\tau_{51}(2) = (1-\rho)\tau_{51}(1) + \frac{100}{271} = 0.9 \cdot 2 + 0.369 = 2.169$$

Для остальных ребер значения уменьшатся следующим образом:

$$\tau_{ij}(2) = (1-\rho)\tau_{ij}(1) = 0.9 \cdot \tau_{ij}(1)$$

Поскольку изначально было лишь 3 значения, покажем данные изменения.

Если начальный уровень феромона был равен 1, то на данном этапе он станет 0.9.

Если начальный уровень феромона был равен 2, то на данном этапе он станет 1.8.

Если начальный уровень феромона был равен 3, то на данном этапе он станет 2.7.

Таким образом, получим следующую новую матрицу:

Таблица 8.3

Новые значения

τ

	1	2	3	4	5
1	×	2.7	1.8	2.169	1.8
2	2.7	×	0.9	0.9	1.269
3	1.8	1.269	×	1.8	1.8
4	1.8	0.9	2.169	×	0.9
5	2.169	0.9	1.8	0.9	×

На этом текущий шаг цикла завершен, можно перейти к следующему шагу.

8.2. Задание к работе

Решить заданным методом задачу, согласно варианту. Отчет должен содержать:

- постановку задачи;
- алгоритм ее решения;
- результат реализации метода (программную форму с входными данными и результатами);
- выводы.

8.3. Варианты заданий

1. Разработать генетический алгоритм для решения задачи о ранце (рюкзаке). Имеется n вещей, каждая из которых характеризуется весом и ценностью, а также рюкзак, позволяющий положить вещи, суммарный вес которых не будет превышать R единиц. Выбрать вещи, обладающие наибольшей стоимостью при ограничении на вес рюкзака.
2. С помощью генетического алгоритма решить задачу о кратчайшем пути.

Хромосома – некоторый путь между двумя заданными вершинами. Отбор родителей и операцию мутации осуществить по своему усмотрению ().

3. С помощью генетического алгоритма решить задачу о коммивояжере.

Хромосома – некоторый путь между двумя заданными вершинами. Отбор родителей и операцию мутации осуществить по своему усмотрению. Скрещивание производим следующим образом:

Первое ребро (например, А-В) берем от первого родителя. Затем ищем ребро, начинающееся на В у второго родителя. Если добавление данного ребра не образует цикла, то добавляем его; в противном случае ищем ближайшее допустимое ребро с вершиной в точке В. Пусть, например, это ребро В-С. Далее переходим к первому родителю и ищем ребро, начинающееся с С. Таким же образом определяем вторую точку ребра, начинающегося на С (если она не образует цикл, то оставляем ее, в противном случае выбираем ближайшую допустимую).

Например:

Родитель 1: 1->4->3->7->2->6->5->8->1

Родитель 2: 1->7->3->4->6->8->5->2->1

Скрещивание. Шаг 1. Взяли от родителя 1 ребро 1->4. Занесли точку 1 в «черный список» (в нее возвращаться нельзя).

Шаг 2. Прошли по ребрам родителя 2 и нашли ребро, начинающееся на 4. Это 4->6. Цикла нет. Дополнили «черный список» цифрой 4.

Шаг 3. У родителя 1 нашли ребро, начинающееся на 6. Это 6->5. Занесли в «черный список» 6.

Шаг 4. У родителя 2 нашли ребро, начинающееся на 5. Это 5->2. Добавляем к потомку данное ребро, запомнили, что в 5 попадать нельзя.

Шаг 5. У родители 1 нашли ребро, начинающееся с цифры 2. Это ребро 2->6. 6 – это недопустимая вершина (как и 1,4 и 5). Нашли ближайшую к 2 точку из множества 3,7,8. Пусть это будет 7. Запомнили ребро 2->7. Занесли 2 в список «запрещенных» городов. Теперь из разрешенных остались только 3 и 8.

Шаг 6. У родителя 2 ищем ребро, начинающееся с 7. Это 7-3. Запоминаем его для потомка.

Шаг 7. Остается единственное ребро: 7->8. И возвращаемся в исходный город 8->1.

Таким образом, потомок следующий: 1->4->6->5->2->7->3->8->1

Мутацию производить следующим образом:

- сгенерировать случайным образом значение i от 1 до n, где n – число городов;

- определить город t_i в текущем маршруте;

- определить ближайший город t_j к городу t_i;

- удалить город t_i из позиции I в маршруте и поместить его в позицию j-1 или j+1.

Например, исходный маршрут:

1->7->3->4->6->8->5->2->1

Мутация. Пусть случайным образом сгенерировано число 5. Город под номером 5 в маршруте – это город 6. Ищем ближайший город к городу 6. Пусть это будет город 2 (позиция города 2 в маршруте – 8). Удаляем город, стоящий на позиции 5 и вставляем его на позицию 8-1=7. Получаем:

1->7->3->4->8->5->6->2->1

Новое поколение формировать на основании элитного отбора.

4. С помощью генетического алгоритма решить задачу о коммивояжере.

Хромосома – некоторый путь между двумя заданными вершинами. Отбор родителей и операцию мутации осуществить по своему усмотрению. Скрещивание производим следующим образом:

- у двух потомков оставить значения 1 вначале и в конце маршрута;

- маршруты двух родителей T_1 и T_2 с помощью двух секущих разделить на три части;

- обменяться средними частями между родителями;

- заполнить города из исходных родителей в двух потомках таким образом, чтобы не было конфликтов (на месте города – конфликта поставить значение 0);

- начинаем заполнять нули «из середины» к концам маршрута: вместо нуля ставим ближайший из допустимых к данному городу город.

Например:

T_1=(1,2,3, | 4,5,6,7, |8,9,1)

T_2=(1,8,7, |6,9,3,4, |5,2,1)

Шаг 1. Обмениваемся «средними частями»:

P_1=(1,X,X, | 6,9,3,4, |X,X,1)

P_2=(1,X,X, |4,5,6,7,|X,X,1)

Шаг 2. Начиная с середины, заполняем потомки P_1 и P_2 значениями с соответствующих городов. Как только будет недопустимый вариант, остальные значения влево и вправо соответственно будут нулевые. Итак, для P_1 будем иметь:

P_1=(1,0,0, | 6,9,3,4, |8,0,1).

Для P_2 получим:

P_2=(1,0,0, |4,5,6,7,|0,0,1).

Шаг 3. Дописываем окончательно потомков. Рассмотрим P_1. Требуется заполнить три нулевых значения городами - 2,5 и 7. Пусть ближайший к 6 из этих городов – это 5:

P_1=(1,0,5, | 6,9,3,4, |8,0,1).

Ищем ближайший город к 5 из городов 2 и 7. Пусть это 2. Тогда потомок 1 будет иметь вид:

P_1=(1,2,5, | 6,9,3,4, |8,7,1).

Аналогичным образом поступаем для P_2.

Мутацию производить следующим образом:

- сгенерировать случайным образом значение i от 1 до N, где N – число городов;
- определить город t_i в текущем маршруте;
- определить ближайший город t_j к городу t_i;
- удалить город t_i из позиции i в маршруте и поместить его в позицию j-1 или j+1.

Например, исходный маршрут:

1->7->3->4->6->8->5->2->1.

Мутация. Пусть случайным образом сгенерировано число 5. Город под номером 5 в маршруте – это город 6. Ищем ближайший город к городу 6. Пусть это будет город 2 (позиция города 2 в маршруте – 8). Удаляем город, стоящий на позиции 5 и вставляем его на позицию 8-1=7. Получаем:

1->7->3->4->8->5->6->2->1.

Новое поколение формировать на основании элитного отбора.

5. Методом муравьиных колоний решить задачу о кратчайшем пути.
6. Методом муравьиных колоний решить задачу о коммивояжере.

ЗАКЛЮЧЕНИЕ

Данный лабораторный практикум предназначен для выполнения лабораторных работ по курсу «методы оптимизации». Следует отметить, что множество вопросов, связанных с оптимизационными задачами, не вошли в данный практикум в связи с ограниченным числом лабораторных работ в данном курсе. Так, в работе почти отсутствует материал, посвященный аналитическому аппарату, предназначенному для решения оптимизационных задач. Это обусловлено направленностью лабораторных работ на численное решение задач. Аналитические подходы к решению оптимизационных задач (метод множителей Лагранжа, теорема Куна-Таккера и т.д.) в полной мере отражены в источниках [1,2].

Данный практикум в силу ограниченного объема лабораторных работ и времени, предназначенного на выполнения каждой из них, к сожалению, не позволяет поместить все оптимизационные задачи, отличающиеся несомненным практическим интересом и методы их решения. Так, за рамками данного пособия остался метод ветвей и границ, являющийся одним из центральных методов дискретной оптимизации и его применение, например, к решению задачи о коммивояжере. Более подробно его можно изучить с помощью источника [4].

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Васильев Ф. П. Методы оптимизации. Книга 1/ Ф. П. Васильев. М.: МЦНМО, 2011. – 620 с.
2. Васильев Ф. П. Методы оптимизации. Книга 2/ Ф. П. Васильев. М.: МЦНМО, 2011. – 434 с.
3. Васильев Ф. П. Численные методы решения экстремальных задач. – М.: Наука, 1988. – 552 с.
4. Сухарев А. Г. Курс методом оптимизации/ А. Г. Сухарев, А. В. Тимохов, В. В. Федоров: учеб. пособие. – М.: ФИЗМАТЛИТ, 2005. – 368 с.
5. Таха Х. Введение в исследование операций, 7-е издание.: Пер. с англ. – М.: Издательский дом «Вильямс», 2005. – 912 с.
6. Прокопенко Н. Ю. Методы оптимизации [Текст]: учеб. пособие /Н. Ю. Прокопенко; Нижегор. гос. архитектур. – строит. ун-т. – Н. Новгород: ННГАСУ, 2018. – 118 с. ISBN 978-5-528-00287-3.
7. Сигал И Х. Задача о рюкзаке: теория и вычислительные алгоритмы: учебное пособие – М.: МИИТ. 1999 – 74 с.
8. Кормен Томас Х. Алгоритмы: построение и анализ, 2-е издание: Пер. с англ./ Т. Х. Кормен, Ч. И. Лейзерсон, Р. Л. Ривест, К. Штайн. – М.: Издательский дом «Вильямс», 2005. – 1296 с.

ПРИЛОЖЕНИЕ

ЭЛЕМЕНТЫ ПРОГРАММИРОВАНИЯ НА C#

Вещественные переменные описывают с помощью служебного слова double. Например:

double a, b, t, y1, y2;

Ввод данных из поля ввода, например из textBox1 в переменную a, реализуется так:

a = Convert.ToDouble(textBox1.Text);

При этом текстовая переменная преобразуется в вещественное число.

Вычисления с применением математических функций записывают следующим образом:

y1 = -Math.Exp(-x1) * Math.Log(x1);

Оператор цикла с предусловием записывают так:

while (Math.Abs(b – a) >= t)

{

Операторы, повторяющиеся в цикле

}

Оператор цикла for имеет следующую структуру:

for (инициализатор; условие; итератор)

{

Операторы, повторяющиеся в цикле

}

Например, с помощью данного цикла можно рассчитать факториал числа:

p=1;

for (int I = I; I < 5; i++)

{

p = p * i;

}

Вывод переменной (например x1) **в компонент listBox1** записывают так:

listBox1.Items.Add(Convert.ToString(x1));

При этом переменная x1 преобразуется в строковый формат представления.

Проверку условия записывают так:

if (y1 <= y2) {операторы, если условие выполняется}

else

{

операторы, если условие не выполняется

}

Вывод результата (функции y) **в компонент textBox5** записывают так:

textBox5.Text = Convert.ToString(y);

При этом реализуется преобразование переменной y в строковый формат.

Работа с массивами в C#

Для создания одномерного массива используется оператор new и указывается тип элементов массива и число элементов. Например:

```
int[] array = new int[5];
…
```

После этого программе доступны элементы массива array[0],…,array[4].

Элементы массива можно инициализировать при объявлении:

```
int[] array1 = new int[] { 1, 3, 5, 7, 9 };
…
```

Получили массив array1, у которого array[0]=1; array1[1]=3,…,array1[4]=9.

Далее с массивами работают, как с обычными переменными. Например, присвоить элементу некоторое значение можно с помощью оператора:

```
array[i] =  Convert.ToDouble(textBox1.Text);
…
```

Для работы с таблицами и представления данных в табличном формате в C# существует элемент управления dataGridView.

Основные свойства DataGridView

Свойство	Назначение
Width	Ширина таблицы
Height	Высота таблицы
Colcount	Количество столбцов
RowCount	Количество строк
Cells	Возвращает коллекцию ячеек, заполняющих строку

Доступ к любой ячейке DataGridView можно получить следующим образом:

```
dataGridView1.Rows[RowIndex].Cells[ColumnIndex].Value = Value;
…
```

Здесь RowIndex – номер строки элемента dataGridView1; ColumnIndex – номер столбца; Value – присваиваемое значение. Например:

```
dataGridView1.Rows[2].Cells[3].Value = 5;
```

присваивает значение 5 в ячейку таблицы, находящейся во второй строке третьего столбца. Следующие пример позволит заполнить матрицу элементами dataGridView:

```
for (int i = 0; i < dataGridView1.RowCount; i++)
          {
             for (int j = 1; j < dataGridView1.ColumnCount; j++)
             {
               a[I, j] = Convert.ToInt32(dataGridView3.Rows[i].Cells[j].Value);
             }
          }
```

ОГЛАВЛЕНИЕ

Учебное издание

Олейникова Светлана Александровна

Сергеева Татьяна Ивановна

Сергеев Михаил Юрьевич

ЧИСЛЕННЫЕ МЕТОДЫ ОПТИМИЗАЦИИ

Редактор Л. Г. Сотникова

Подписано в печать .2021.

Формат 60х84 1/16. Бумага для множительных аппаратов.

Уч.-изд. л. 5,2. Усл. печ. л. 5,6. Тираж 50 экз.

Заказ №

ФГБОУ ВО «Воронежский государственный технический университет»
394026 Воронеж, Московский проспект, 14

Участок оперативной полиграфии издательства ВГТУ
394026 Воронеж, Московский проспект, 14