**Численные методы поиска минимума функции нескольких переменных. Методы спуска**

# 1. Основные понятия и определения

Одной из наиболее часто встречающихся в инженерных расчетах и научных исследованиях вычислительных задач является задача минимизации функции n действительных переменных f (x1, x2,…, xn). Функция f (целевая функция) минимизируется на некотором множестве X ⊂ Rn. В случае, когда X = Rn (т.е. ограничения на переменные x1, x2,…, xn отсутствуют), принято говорить о задаче **безусловной минимизации**. В противном случае (т.е. когда X ≠ Rn) говорят о задаче **условной минимизации**.

## 1.1 Постановка задачи

Пусть требуется решить задачу:

.

В двумерном пространстве R2 решению этой задачи можно дать геометрическую интерпретацию. Пусть точка X = (x1, x2) лежит на плоскости Ox1x2. Введем третью координату X3 так, чтобы ось координат Ox3 была перпендикулярна плоскости Ox1x2 (рис. 1). Уравнению x3 = f (x1, x2) соответствует поверхность в трехмерном пространстве.

Если функция f (X) достигает локального минимума в точке x\* ∈ R2, то поверхность x3 = f (x1, x2) в некоторой окрестности точки x\* имеет форму чаши (рис.1).

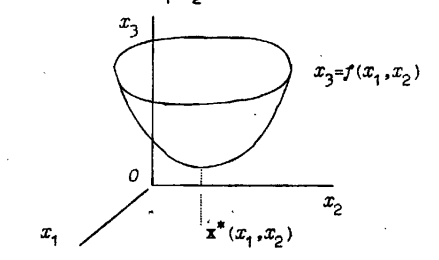


Рис. 1 Поверхность, соответствующая функции двух переменных

**Линиями уровня** функции f (x1, x2) называют семейство линий плоскости R2, на которых функция принимает постоянное значение. Неявным уравнением линии уровня является уравнение f (x1, x2) = C. Если функция f (X) имеет в R2 единственную точку локального минимума X\* (x1, x2), то такая функция называется **мономодальной**. Взаимное расположение ее линий уровня имеет вид, изображенный на рис. 2.

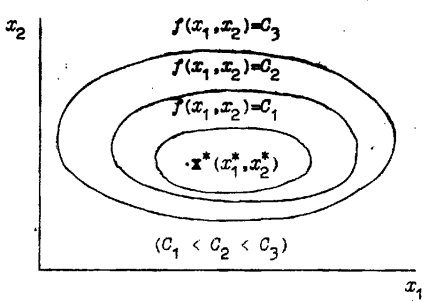


Рис. 2 Линии уровня мономодальной функции

**Мультимодальными** называются функции, которые имеют более одного экстремума. Такова, например, функция Химмельблау:

,

имеющая четыре изолированные точки минимума. На рис. 3 схематично изображены линии уровня этой функции.

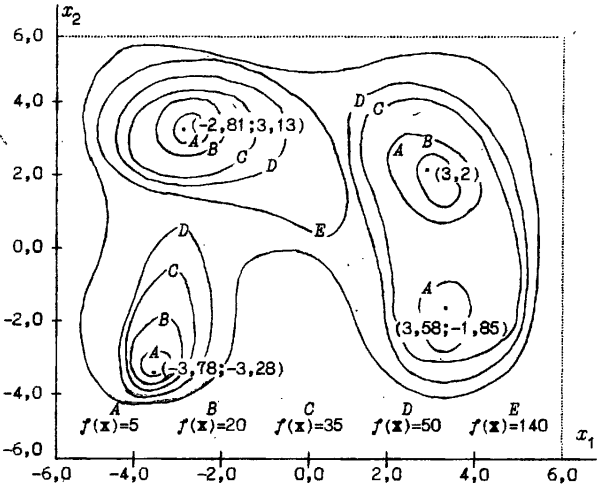


Рис. 3 Линии уровня функции Химмельблау

Чтобы найти точку X\* локального минимума функции f (X), составляют последовательность точек (приближений к решению) {X(k)} (k=0,1, …), сходящуюся к точке X\*.

Последовательность значений функции f (X(k)) должна быть монотонно убывающей и ограниченной снизу:

.

Геометрический образ решения задачи минимизации для функции двух переменных напоминает спуск на дно чаши. Этим объясняется название методов решения задачи – «методы спуска».

Для различных методов спуска сначала выбирают начальную точку последовательности X(0). Дальнейшие приближения X(k) определяются соотношениями

(k = 0, 1, 2, … ),

где S(k) – вектор направления спуска; скалярная величина t(k) является решением задачи одномерной минимизации

*.*

Таким образом, задача поиска минимума функции нескольких переменных сводится к последовательности задач одномерной минимизации по переменной t.

Методы спуска отличаются выбором вектора спуска и способом решения задачи одномерной минимизации. Далее будут рассмотрены некоторые из методов спуска.

## 1.2 Исследование функции на экстремумы

**Теорема 1 (о необходимых условиях локального минимума)**

Пусть X\* - точка локального минимума функции f(X), которая имеет в этой точке непрерывные частные производные ; тогда частные производные функции f(X) в этой точке равны нулю, т.е.:

*.*

Иначе говоря, в этой точке градиент функции

равен нулевому вектору, т.е. grad f(X\*) = 0.

**Определение**

Точка X\*, удовлетворяющая условию grad f(X\*) = 0, называется **стационарной точкой** функции f(X).

Не всякая стационарная точка является решением задачи минимизации.

**Определение**

Матрица вида

называется **матрицей Гессе** функции f(X).

**Теорема 2 (о достаточных условиях локального экстремума)**

Пусть точка X\* является стационарной точкой функции f(X), т.е.

*,*

имеет непрерывные частные производные до второго порядка включительно, тогда

1) если все угловые миноры матрицы Гессе в точке X\* положительны:

то X\* - точка локального минимума функции f(X).

2) если знаки угловых миноров матрицы Гессе в точке X\* последовательно чередуются, начиная с отрицательного значения a11:

то X\* - точка локального максимума функции f(X).

Если определитель матрицы Гессе функции f(X) в точке X\* равен нулю, то требуется дополнительное исследование.

**Пример 1**

Решить задачу

.

Решение. Найдем стационарные точки функции f(X):

Решением этой системы уравнений являются две точки:

При этом

В стационарной точке (X\*)2 выполнены достаточные условия теоремы 2:

*,*

поэтому в точке (X\*)2 данная целевая функция достигает локального минимума.

**Пример 2**

Исследовать на экстремум функцию

Решение. Найдем стационарные точки:

или

Решением этой системы является точка

При этом определитель ∆ матрицы Гессе

положителен: и

Следовательно, в точке X\* имеет место строгий локальный максимум.

# 2. Метод покоординатного спуска

В методе покоординатного спуска в качестве очередного направления спуска выбирают направление одной из координатных осей. Наиболее известным является **метод циклического покоординатного спуска**. Опишем очередной (k + 1)-й цикл одного из вариантов этого метода, считая приближение X(k) уже найденным.

Цикл с номером k + 1 состоит из n шагов (по числу переменных функции). На первом шаге производят спуск по координате x1. Значения x2 = x2(k) , …, xn = xn(k) остальных координат фиксируют, а x1(k+1) выбирают из условия

*.*

То есть решается задача минимизации функции одной переменной

На втором шаге производят спуск по координате x2. Значения x1 = x1(k+1) , x3 = x3(k) ,…, xn = xn(k) остальных координат фиксируют и x2(k+1) выбирают как решение задачи одномерной оптимизации

.

Аналогично осуществляют остальные шаги. На последнем n-м шаге координату xn(k+1) определяют из условия

.

В результате получается очередное приближение X(k+1) к точке минимума. Далее цикл метода снова повторяют. На рис. 4 изображена геометрическая иллюстрация циклического покоординатного спуска.

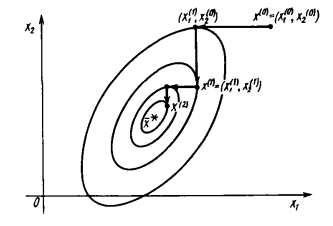


Рис. 4 Геометрическая иллюстрация циклического покоординатного спуска

# 3. Градиентные методы

## 3.1 Общая характеристика градиентных методов

В методе покоординатного спуска осуществляется поиск из заданной точки в направлении, параллельном одной из осей, до точки минимума в данном направлении. Затем поиск производится в направлении другой оси, и т.д. Кажется разумным модифицировать этот метод таким образом, чтобы на каждом этапе поиск точки минимума производился вдоль «наилучшего» направления. Известно, что направление градиента является направлением наискорейшего возрастания функции. Следовательно, противоположное направление (антиградиент) является направлением наискорейшего убывания функции.

Напомним, что градиент функции

.

В точке минимума grad f (X) = 0.

Метод градиентного спуска основан на движении в направлении максимального убывания функции, т.е. в направлении –grad f (X). Очередное приближение выбирается следующим образом:

*.*

С учетом нормировки вектора градиента данная формула примет вид:

,

где

λ(K) определяет **величину шага** и может быть:

* **постоянным**, в этом случае метод может расходиться;
* **дробным шагом**, т.е. длина шага в процессе спуска делится на некоторое число;
* **наискорейшим спуском**: .

**Критерии остановки процесса приближенного нахождения минимума** могут быть различными:

* ;
* ;
* сочетание первого и второго критериев.

## 3.2 Градиентный метод с постоянным шагом

В данном методе величина шага λ задана заранее.

Очередное приближение в ней получается по следующей формуле:

*.*

То есть после совершения очередного шага необходимо проверить, не выполнен ли критерий остановки процесса поиска минимума (рассмотрен выше). Если не выполнен, то следует пересчитать вектор градиента и сделать очередной шаг. Данная процедура повторяется до тех пор, пока не выполнено условие остановки.

При больших значениях λ процесс может не сходиться к решению.

## 3.3 Метод наискорейшего спуска

Точку X(k+1) в методе наискорейшего спуска определяют из решения задачи одномерной минимизации функции

.

Данную задачу можно численно решить методом сканирования или любым другим методом одномерной минимизации.

То есть отличием данного метода от градиентного метода с постоянным шагом является то, что движение в направлении антиградиента на текущей итерации выполняется до тех пор, пока происходит уменьшение значения целевой функции. После этого градиент пересчитывается.

Данная процедура повторяется до тех пор, пока не выполняется выбранный критерий остановки (описаны выше).

# Практическая часть

Примечания:

* в пунктах 2-4 для решения задачи одномерной минимизации считать функцию одной переменной монотонной, т.е. от текущей точки производить шаги в сторону уменьшения значения функции до тех пор, пока не будет достигнут минимум функции одной переменной в данном направлении. Величину шага принять равной 0,001.
* в методах градиентного спуска (задания 3 и 4) вручную рассчитать частные производные для заданных функций.
* критерий остановки процесса использовать . Принять ε = 0,00001.

Задания:

1. Исследовать заданную функцию на экстремум (табл. 1) и построить ее график. Информация о построении графиков с использованием python приведена в приложении.

|  |  |
| --- | --- |
| Номер варианта | Функция |
| 1 |  |
| 2 |  |
| 3 |  |
| 4 |  |
| 5 |  |
| 6 |  |
| 7 |  |
| 8 |  |
| 9 |  |
| 10 |  |

2. Написать программу нахождения минимума функции по методу покоординатного спуска для функции из п.1. Схема алгоритма метода покоординатного спуска изображена на рис. 5 и 6. Начальную точку взять (x0, y0) = (1; 1).

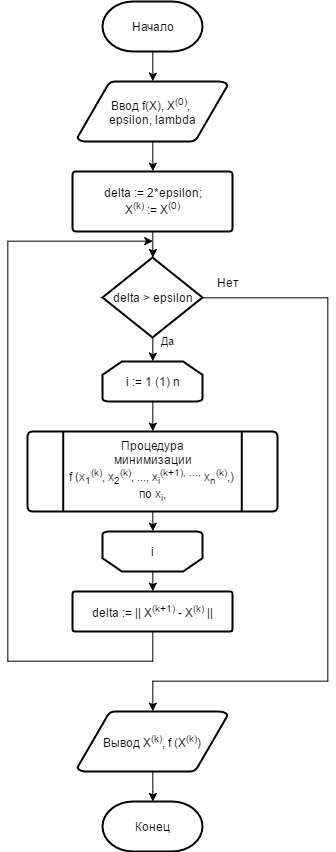


Рис. 5 Схема алгоритма метода покоординатного спуска

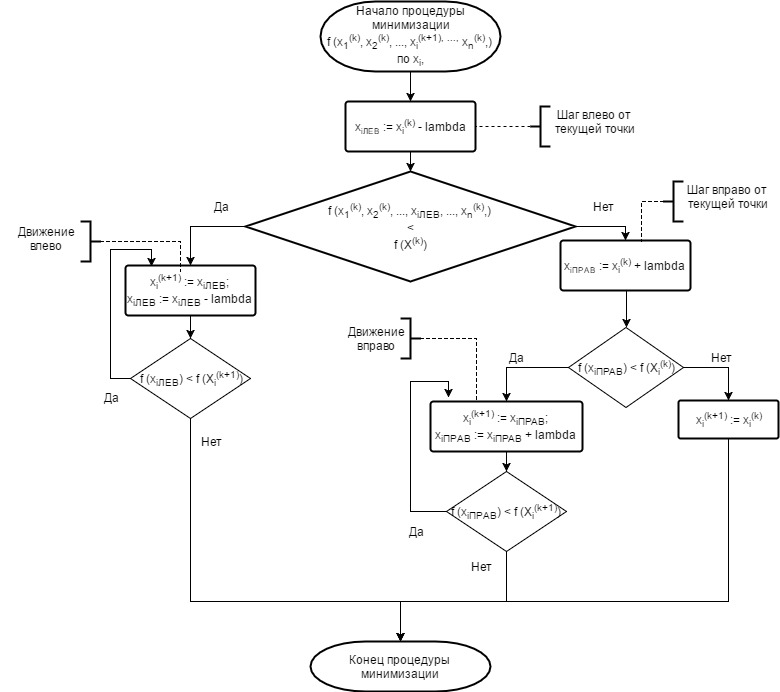


Рис. 6 Процедура минимизации функции по одной координате

3. Написать программу нахождения минимума функции по методу градиентного спуска с постоянным шагом для функции из п.1. Схема алгоритма приведена на рис. 7. Сравнить результаты для величины шага λ = 0,1; 0,001; 0,0001. Начальную точку взять (x0, y0) = (1; 1).

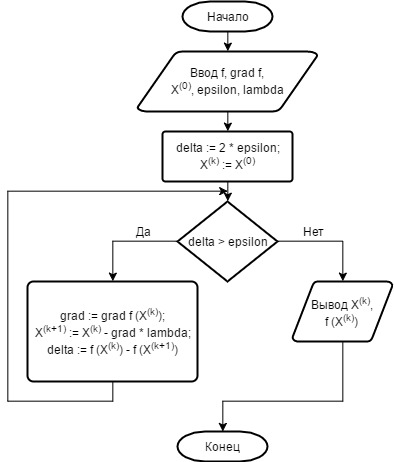


Рис. 7 Схема алгоритма метода градиентного спуска с постоянным шагом

4. Написать программу алгоритма нахождения минимума функции по методу наискорейшего градиентного спуска для функции из п.1. Схема алгоритма приведена на рис. 8 и 9. Начальную точку взять (x0, y0) = (1; 1).

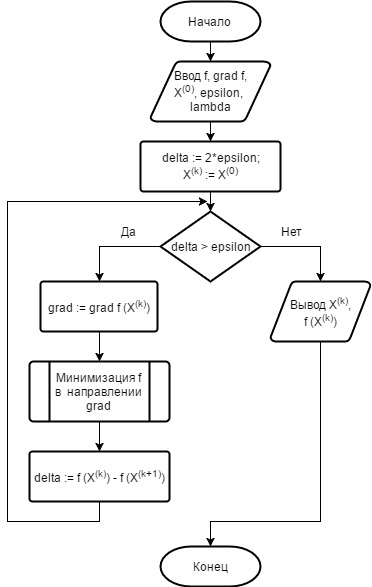


Рис. 8 Схема алгоритма метода наискорейшего спуска

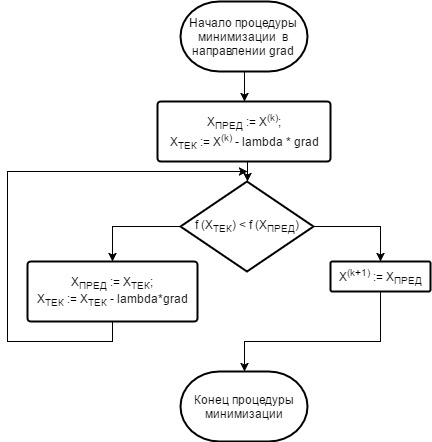


Рис. 9 Схема алгоритма процедуры минимизации в методе наискорейшего спуска

5. Составить отчет, содержащий расчеты из пункта 1 и график, результаты работы программ из пунктов 2-4.

# Приложения

## Приложение 1 Построение графиков с использованием Python

На компьютер должен быть установлен Python версии 3.5 или новее.

Для построения графика с помощью python необходимо установить библиотеки numpy и matplotlib. Для этого необходимо в командной строке Windows (нажать Win+R, ввести cmd, нажать Enter) выполнить следующие команды:

* pip install numpy
* pip install matplotlib

После этого запустите idle и создайте новый файл.

Пример скрипта, производящего построение графика приведен ниже.

import numpy as np #1

from mpl\_toolkits.mplot3d import Axes3D #2

import matplotlib.pyplot as plt #3

import random #4

#5

def fun(x, y): #6

return x\*\*2 + y #7

#8

fig = plt.figure() #9

ax = fig.add\_subplot(111, projection='3d') #10

x = y = np.arange(-10.0, 10.0, 0.5) #11

X, Y = np.meshgrid(x, y) #12

zs = np.array([fun(x,y) for x,y in zip(np.ravel(X), np.ravel(Y))]) #13

Z = zs.reshape(X.shape) #14

#15

ax.plot\_surface(X, Y, Z) #16

#17

ax.set\_xlabel('X Label') #18

ax.set\_ylabel('Y Label') #19

ax.set\_zlabel('Z Label') #20

#21

plt.show() #22

Если ваша функция содержит экспоненту, добавьте в начало скрипта строку

import math

Для возведения экспоненты в степень можно использовать функцию math.exp(x), где x-степень, в которую требуется возвести экспоненту.

Для возведения произвольного числа в степень используется оператор \*\*. Например, 2\*\*3 = 23.

В python символ # означает начало однострочного комментария.

В приведенном выше скрипте строка 6 – объявление функции с именем fun, принимающей два аргумента – x и y. Следующая строчка c №7– это значение, возвращаемое данной функцией. Замените его на свою функцию для построения графика.

Обратите внимание, что текст строки 7 (тело функции fun) должен быть отделен от начала строки 4 пробелами или одним символом табуляции – правило выделения вложенных блоков кода в python.

Строка 11 определяет диапазоны изменения переменных x и y. Первое значение – начальное значение, второе - конечное, третье – шаг изменения.

Сохраните скрипт и запустите на выполнение нажатием клавиши F5.