# Тема 12. Минимизация функций нескольких переменных

Забросить ключ от дома и уйти,
Не как изгнанник, что бредёт без смысла,
Но выбирая свой маршрут разумно,
Меняя скорость, если склон сменился.

У. Х. Оден «*The Jorney*» (1928)

## 12.1. Постановка задачи. Проблемы минимизации.

На практике часто встречаются задачи, в которых требуется максимизировать или минимизировать некоторую функцию от $n$ переменных $f(\boldsymbol{x})$, где $\boldsymbol{x} = (x_1, \ldots, x_n)$.

Очевидно, функция $-f(\boldsymbol{x})$ имеет минимум там, где $f(\boldsymbol{x})$ имеет максимум. Поэтому достаточно уметь находить минимум функции.

Функцию двух переменных $f(x, y)$ удобно представлять себе в виде поверхности, заданной уравнением $z = f(x, y)$: над точкой плоскости с координатами $(x, y)$ поверхность располагается на высоте $z$ (рис. 1).

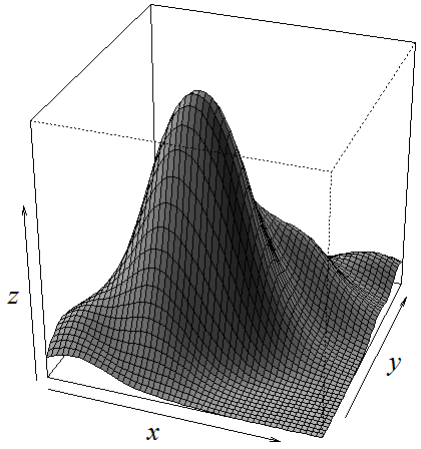


**Рис. 1**

Функцию трёх переменных $f(x, y, z)$ можно интерпретировать как температуру в точке пространства с координатами $(x, y, z)$. Функцию четырёх и большего числа переменных представить невозможно. Однако с такими функциями приходится иметь дело, потому что интересующие исследователя объекты зачастую описывается $n$ характеристиками, где $n > 3$.

Главная проблема при поиске глобального минимума функции $n$ переменных даже при умеренных $n$ заключается в невозможности прямого перебора значений функции во всех точках $n$-мерной сетки, имеющей достаточно малую длину стороны ячейки. Например, для сетки с шагом 0,01 на $n$-мерном кубе со сторонами длины 1 при $n$ = 10 придётся вычислить $100^{10} = 10^{20}$ значений функции. Даже если отдельное значение рассчитывается очень быстро, вычисление функции во всех $10^{20}$ точках сетки займёт нереально большое время на любом компьютере.

Поэтому на практике для поиска минимума функции многих переменных применяются разные численные методы, например, рассматриваемые ниже *методы спуска*. Эти алгоритмы не гарантируют нахождения глобального минимума, а лишь позволяют найти ближайший к стартовой точке локальный минимум (в частности, в двумерном случае происходит скатывание по поверхности, задаваемой функцией, на дно соседней «ямки»).

Обычно с этой проблемой пытаются справиться, многократно стартуя из случайно выбранных начальных точек. Однако в пространствах большого числа измерений нелинейные функции нередко имеют очень много локальных минимумов (подобных перевёрнутой вниз поверхности, изображённой на рис. 2). При этом «область притяжения» к точке глобального минимума может иметь ничтожный $n$-мерный объём (см. материал в конце темы 7 про объём $n$-мерного шара). Иначе говоря, для таких функций абсолютно нереально выбрать стартовую точку так, чтобы спуститься из неё именно в глобальный минимум, минуя локальные.

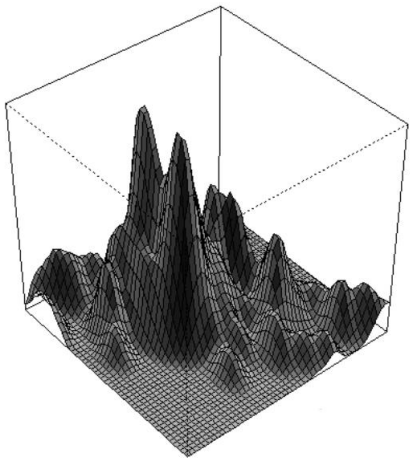

**Рис. 2**

## 12.2. Одномерная минимизация методом «золотого сечения»

Как правило, при минимизации функции нескольких переменных методами спуска в качестве вспомогательного алгоритма используются методы одномерной минимизации, т. е. методы быстрого поиска минимума функции одной переменной. Одним из самых простых таких алгоритмов является метод «золотого сечения».

*Золотым сечением* называют пропорцию, почитавшуюся в древнегреческом искусстве и архитектуре, при которой «меньшее» относится к «большему» как «большее» к «целому»: $(1-\kappa):\kappa=\kappa:1$ (рис. 3). Из квадратного уравнения $1-\kappa=\kappa^2$, находим, что $\kappa=(\sqrt{5}-1)/2\approx 0{,}618$.

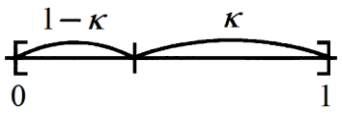


**Рис. 3**

Будем минимизировать функцию $\varphi(t)$. Предположим, что $\varphi(t)$ имеет единственный минимум на отрезке $[a, b]$ в некоторой неизвестной точке $\tilde{t}\in[a, b]$ (в противном случае будет найден один из локальных минимумов).

### *Алгоритм «золотого сечения»*

***Шаг* 1**. На отрезке $[a, b]$ возьмём две пробные точки: $r=a+(1-\kappa)(b-a)$ и $s=a+\kappa(b-a)$, вычислим значения $\varphi(r)$ и $\varphi(s)$. Заметим, что точки $r$ и $s$ осуществляют золотое сечение не только отрезка $[a, b]$, но и отрезков $[a, s]$ и $[r, b]$ соответственно (рис. 4). Это позволяет вычислять всего одно значение функции при следующем шаге.

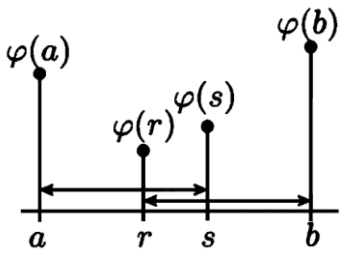


**Рис. 4**

***Шаг* 2.** Если $\varphi(r) < \varphi(s)$, то полагаем $\tilde{t} = r,\ a' = a,\ r' = a + (1-\kappa)(s-a),\ s' = r,\ b' = s$ и вычисляем значение $\varphi(r')$. В противном случае полагаем $\tilde{t} = s,\ a' = r,\ r' = s,\ s' = a + \kappa(b-r),\ b' = b$ и вычисляем значение $\varphi(s')$.

***Шаг* 3.** Если $b' - a' \le \varepsilon$ (скажем, $\varepsilon = 0{,}01$), то заканчиваем поиск, иначе убираем у всех переменных штрихи и возвращаемся к шагу 2.

Отметим, что при каждом выполнении шага 2 длина отрезка локализации точки минимума уменьшается в $1/\kappa \approx 1{,}618$ раз. Всего за 10 итераций первоначальный отрезок $[a, b]$ сожмётся в $1{,}618^{10} \approx 123$ раза.

## 12.3. Методы наискорейшего спуска и сопряжённых градиентов

Методы многомерной минимизации, называемые ***методами спуска***, заключаются в чередовании следующих двух процедур:

а) выбор на $k$-м шаге направления движения ($n$-мерного вектора) $\boldsymbol{p}_k$;

б) минимизация функции $f(\boldsymbol{x})$ по направлению $\boldsymbol{p}_k$, т. е. для текущего положения $\boldsymbol{x}_k$ на $k$-м шаге поиска и выбранного направления движения $\boldsymbol{p}_k$ осуществляется поиск минимума функции $\varphi(t) = f(\boldsymbol{x}_k + t\,\boldsymbol{p}_k)$ по единственной действительной переменной $t$.

Пожалуй, самым известным таким методом является ***метод наискорейшего спуска***, у которого направлением спуска служит антиградиент функции $f(\boldsymbol{x})$ в текущей точке $\boldsymbol{x}_k$, т. е.

$$\boldsymbol{p}_k = -\boldsymbol{g}(\boldsymbol{x}_k) = \left( -\frac{\partial f(\boldsymbol{x}_k)}{\partial x_1}, \ldots, -\frac{\partial f(\boldsymbol{x}_k)}{\partial x_n} \right).$$

Метод наискорейшего спуска базируется на известном из математического анализа утверждении, что гладкая функция многих переменных убывает быстрее всего в направлении *антиградиента* $-\boldsymbol{g}(\boldsymbol{x})$. Действительно, из разложения по формуле Тейлора (см. формулу (6) в теме 11) и *неравенства Коши — Буняковского — Шварца*

$$|\boldsymbol{a}^T \boldsymbol{b}| \le |\boldsymbol{a}| \cdot |\boldsymbol{b}|$$

при $\boldsymbol{a} = \boldsymbol{g}(\boldsymbol{x}_0)$ и $\boldsymbol{b} = \boldsymbol{x} - \boldsymbol{x}_0$ следует, что с точностью до пренебрежимо малой нелинейности функция $f(\boldsymbol{x})$ быстрее всего локально растёт в направлении градиента $\boldsymbol{g}(\boldsymbol{x})$ и, соответственно, быстрее всего убывает в направлении антиградиента $-\boldsymbol{g}(\boldsymbol{x})$.

***Теорема.*** *Необходимым условием* минимума гладкой функции является равенство $\boldsymbol{g}(\tilde{\boldsymbol{x}}) = \boldsymbol{0}$, где $\tilde{\boldsymbol{x}}$ — точка локального минимума, $\boldsymbol{0}$ — нулевой вектор.

Поэтому одним из индикаторов необходимости остановки спуска служит условие $|\boldsymbol{g}(\boldsymbol{x}_k)| < \varepsilon$, где $\varepsilon$ — достаточно малое число (скажем, 0,001).

Однако данная самая выгодная локально («жадная») стратегия спуска обычно оказывается совершенно непрактичной для «овражных» функций: движение в направлении антиградиента приводит к огромному числу смен направления движения при перемещении вдоль «дна оврага». Наглядный пример даёт поиск минимума *функции Розенброка*

$$f(x, y) = 100(y - x^2)^2 + (1 - x)^2,$$

которая определяет поверхность типа серповидного ущелья с точкой минимума $(1, 1)$. Слева на рис. 5, взятом из монографии Гилл Ф., Мюррей У., Райт М. «Практическая оптимизация», показан процесс минимизации функции Розенброка методом наискорейшего спуска из начальной точки $(-6/5, 1)$. Гладкие кривые — это линии уровня поверхности, отрезки ломаной соответствуют шагам спуска. Алгоритм мог бы надолго «застрять» вблизи начала координат, если бы не помогла счастливая случайность — на одном из шагов спуска в результате одномерного поиска был обнаружен второй минимум по направлению. Затем было выполнено ещё несколько сотен шагов, но ощутимого уменьшения функции это не дало. Поиск был прерван после 1000 шагов вдали от искомой точки минимума.

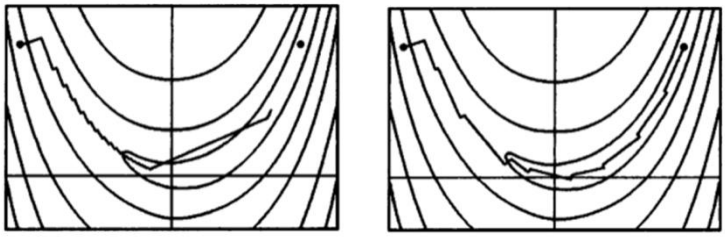

**Рис. 5**

Для минимизации «овражных» функций обычно применяются другие методы спуска, например, ***метод сопряжённых градиентов***. При этом методе направление спуска $\boldsymbol{p}_k$ на шаге с номером $k$ выбирается согласно формулам

$$\boldsymbol{p}_1 = -\boldsymbol{g}_1; \quad \boldsymbol{p}_k = -\boldsymbol{g}_k + \alpha_k \boldsymbol{p}_{k-1}, \quad k = 2, 3, \ldots, n, \tag{1}$$

где $\alpha_k = |\boldsymbol{g}_k|^2 / |\boldsymbol{g}_{k-1}|^2$, $\boldsymbol{g}_k$ — градиент на шаге с номером $k$. Если в результате $n$ шагов функция практически не уменьшилась, то поиск минимума прекращается. В противном случае выполняются дополнительные $n$ шагов поиска согласно формулам (1) и т. д.

На рис. 5 справа на карте линий уровня функции Розенброка приведена траектория спуска для метода сопряжённых градиентов. Алгоритм позволил дойти до точки минимума за два десятка шагов. Хорошо виден циклический характер метода в случае функции двух переменных: движение вдоль дна «ущелья» подправляется перемещением по антиградиенту.

Впервые этот метод был опубликован в качестве численного алгоритма для решения системы линейных уравнений $\mathbf{A}\boldsymbol{x} = \boldsymbol{b}$ с положительно определённой матрицей $\mathbf{A}$ (свойство положительной определённости было определено в разделе 10.2). Можно показать, что данная задача равносильна минимизации квадратичной функции

$$f(\boldsymbol{x}) = \frac{1}{2}\boldsymbol{x}^T \mathbf{A}\boldsymbol{x} - \boldsymbol{b}^T \boldsymbol{x}.$$

Оказывается, что (при отсутствии погрешностей округления) метод сопряжённых градиентов позволяет найти точку минимума $f(\boldsymbol{x})$ не более, чем за $n$ итераций, где $n$ — размерность $\boldsymbol{x}$.

При этом направления одномерной минимизации $\boldsymbol{p}_1, \ldots, \boldsymbol{p}_n$ являются *взаимно сопряжёнными* относительно матрицы $\mathbf{A}$:

$$\boldsymbol{p}_i^T \mathbf{A} \boldsymbol{p}_j = 0 \text{ при всех } i \neq j,$$

что объясняет название метода. Рис. 6 иллюстрирует ход процесса минимизации при $n = 2$.

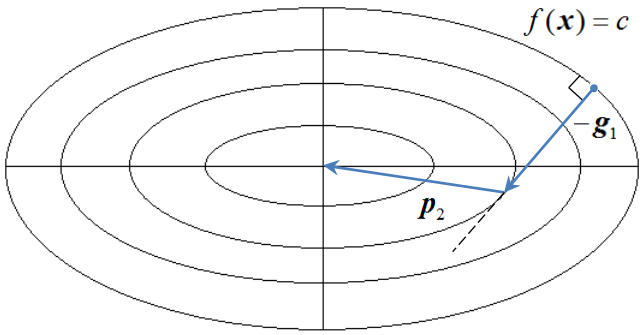


**Рис. 6**

Сначала выполняется минимизация в направлении антиградиента: $\boldsymbol{p}_1 = -\boldsymbol{g}_1$. Затем выбирается направление $\boldsymbol{p}_2$, приводящее в точку минимума — центр симметрии эллипса.