Лекция **3**

4. ЧИСЛЕННЫЕ МЕТОДЫ ПОИСКА БЕЗУСЛОВНОГО ЭКСТРЕМУМА

Принципы построения численных методов. Применение необходимых и достаточных условий безусловного экстремума эффективно для решения ограниченного числа примеров, в которых вытекающие из условий соотношения имеют аналитическое решение. Для решения большинства практических задач они не могут быть рекомендованы по следующим причинам:

- целевая функция $f(x)$ может не иметь непрерывных производных до второго порядка включительно;
- использование необходимого условия первого порядка связано с решением системы n в общем случае нелинейных алгебраических уравнений, что представляет собой самостоятельную задачу, трудоемкость решения которой сравнима с трудоемкостью численного решения поставленной задачи поиска экстремума;
- возможны случаи, когда о целевой функции известно лишь то, что ее значение может быть вычислено с нужной точностью, а сама функция задана неявно.

Подавляющее большинство численных методов оптимизации относится к классу *итерационных*, т.е. порождающих последовательность точек в соответствии с предписанным набором правил, включающим критерий окончания. При заданной начальной точке x^0 методы генерируют последовательность $x^0, x^1, x^2, \dots$ Преобразование точки x^k в x^{k+1} представляет собой *итерацию*.

Для определенности рассмотрим задачу поиска безусловного локального минимума:

$$f(x^*) = \min_{x \in R^n} f(x).$$

Численное решение задачи безусловной оптимизации, как правило, связано с построением последовательности $\{x^k\}$ точек, обладающих свойством $f(x^{k+1}) < f(x^k)$, $k = 0,1,\dots$.

Общее правило построения последовательности $\{x^k\}$ имеет вид

$$x^{k+1} = x^k + t_k\, d^k, \quad k = 0,1,\dots,$$

где точка x^0 – *начальная точка* поиска; d^k – приемлемое направление перехода из точки x^k в точку x^{k+1}, обеспечивающее выполнение условия убывания функции и называемое *направлением спуска*; t_k – *величина шага*.

Начальная точка поиска x^0 задается, исходя из физического содержания решаемой задачи и наличия априорной информации о положении точек экстремума.

Классификация численных методов поиска безусловного экстремума. В зависимости от наивысшего порядка частных производных функции $f(x)$, используемых для формирования d^k и t_k, численные методы решения задачи безусловной минимизации принято делить на три группы:

- *методы нулевого порядка*, использующие только информацию о значении функции $f(x)$;
- *методы первого порядка*, использующие информацию о первых производных функции $f(x)$;
- *методы второго порядка*, требующие для своей реализации знания вторых производных функции $f(x)$.

МЕТОДЫ НУЛЕВОГО ПОРЯДКА

А. МЕТОДЫ ОДНОМЕРНОЙ МИНИМИЗАЦИИ

Постановка задачи. Требуется найти безусловный минимум функции $f(x)$ одной переменной, т.е. такую точку $x^* \in R$, что

$$f(x^*) = \min_{x \in R} f(x).$$

Поставленная *задача одномерной минимизации* может быть решена с помощью необходимых и достаточных условий безусловного экстремума. Однако проблема получения решения уравнения $\frac{df(x)}{dx} = 0$ может оказаться весьма сложной. Более того, в практических задачах функция $f(x)$ может быть не задана в аналитическом виде или часто неизвестно, является ли она дифференцируемой. Поэтому получение численного решения поставленной задачи является важным для приложений.

З а м е ч а н и я.

1. Для методов одномерной минимизации типично задание априорной информации о положении точки минимума с помощью начального *интервала неопределенности* $L_0 = [a_0, b_0]$. Предполагается, что точка минимума x^* принадлежит интервалу L_0, но ее точное значение неизвестно.

2. Большинство известных методов одномерной минимизации применяется для класса унимодальных функций.

Определение. Функция $f(x)$ называется *унимодальной на интервале* $L_0 = [a_0, b_0]$, если она достигает глобального минимума на $[a_0, b_0]$ в единственной точке x^*, причем слева от x^* эта функция строго убывает, а справа от x^* – строго возрастает.

3. Методы одномерной минимизации широко применяются в методах первого и второго порядков для нахождения оптимальной величины шага. При этом левая граница начального интервала неопределенности, как правило, совпадает с началом координат, т.е. $a_0 = 0$.

Стратегия поиска включает в себя три этапа:

1. Выбор начального интервала неопределенности. Границы a_0, b_0 интервала должны быть такими, чтобы функция $f(x)$ была унимодальной.

2. Уменьшение интервала неопределенности.

3. Проверку условия окончания. Поиск заканчивается, когда длина текущего интервала неопределенности $[a_k, b_k]$ оказывается меньше установленной величины.

Ответом является множество точек, принадлежащих последнему интервалу неопределенности, среди которых каким-либо образом выбирается решение задачи x^*.

В некоторых методах заранее задается или находится количество N вычислений функции. В этом случае продолжительность поиска ограничена.

А1. Метод дихотомии

Стратегия поиска

Задаются начальный интервал неопределенности и требуемая точность. Алгоритм опирается на анализ значений функции в двух точках (см. рис. 1). Для их нахождения текущий интервал неопределенности делится пополам и в обе стороны от середины откладывается по $\frac{\varepsilon}{2}$, где ε – малое положительное число. Условия окончания процесса поиска стандартные: поиск заканчивается, когда длина текущего интервала неопределенности оказывается меньше установленной величины.

Алгоритм

Шаг 1. Задать начальный интервал неопределенности $L_0 = [a_0, b_0]$, $\varepsilon > 0$ – малое число, $l > 0$ – точность.

Шаг 2. Положить $k = 0$.

Шаг 3. Вычислить $y_k = \frac{a_k + b_k - \varepsilon}{2}$, $f(y_k)$, $z_k = \frac{a_k + b_k + \varepsilon}{2}$, $f(z_k)$.

Шаг 4. Сравнить $f(y_k)$ с $f(z_k)$:

а) если $f(y_k) \le f(z_k)$, положить $a_{k+1} = a_k$, $b_{k+1} = z_k$ (рис. 1, *а*) и перейти к шагу 5;

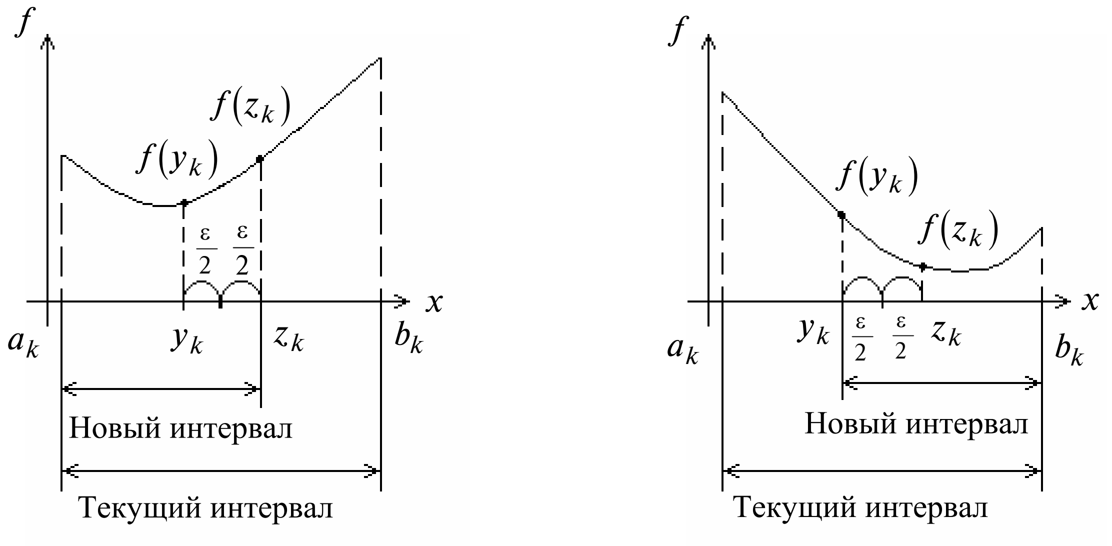
б) если $f(y_k) > f(z_k)$, положить $a_{k+1} = y_k$, $b_{k+1} = b_k$ (рис. 1, *б*).

Шаг 5. Вычислить $|L_{2(k+1)}| = |b_{k+1} - a_{k+1}|$ и проверить условие окончания:

а) если $|L_{2(k+1)}| \le l$, процесс поиска завершить. Точка минимума принадлежит интервалу: $x^* \in L_{2(k+1)} = [a_{k+1}, b_{k+1}]$. В качестве приближенного решения можно взять середину последнего интервала: $x^* \cong \frac{a_{k+1} + b_{k+1}}{2}$;

б) если $|L_{2(k+1)}| > l$, положить $k = k + 1$ и перейти к шагу 3.

З а м е ч а н и е. Текущие интервалы неопределенности $L_0, L_2, L_4, \ldots$ имеют четные номера, указывающие на количество сделанных вычислений функции.



а *б*

Рис. 1

А2. Метод золотого сечения

В методе золотого сечения в качестве точек вычисления функции выбираются точки золотого сечения.

Определение. Точка производит *золотое сечение отрезка*, если отношение длины всего отрезка к большей части равно отношению большей части к меньшей.

На отрезке $[a_0, b_0]$ имеются две симметричные относительно его концов точки y_0 и z_0:

$$\frac{b_0 - a_0}{b_0 - y_0} = \frac{b_0 - y_0}{y_0 - a_0} = \frac{b_0 - a_0}{z_0 - a_0} = \frac{z_0 - a_0}{b_0 - z_0} = \frac{1 + \sqrt{5}}{2} \cong 1{,}618\,.$$

При этом точка y_0 производит золотое сечение отрезка $[a_0, z_0]$, а точка z_0 – отрезка $[y_0, b_0]$ (рис. 2).

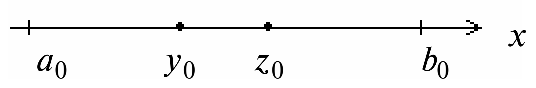


Рис. 2

Стратегия поиска

Задаются начальный интервал неопределенности и требуемая точность. Алгоритм уменьшения интервала опирается на анализ значений функции в двух точках (см. рис. 2). В качестве точек вычисления функции выбираются точки золотого сечения. Тогда с учетом свойств золотого сечения на каждой итерации, кроме первой, требуется произвести только одно новое вычисление функции. Условия окончания процесса поиска стандартные: поиск заканчивается, когда длина текущего интервала неопределенности оказывается меньше установленной величины.

Алгоритм

Шаг 1. Задать начальный интервал неопределенности $L_0 = [a_0, b_0]$, точность $l > 0$.

Шаг 2. Положить $k = 0$.

Шаг 3. Вычислить

$$y_0 = a_0 + \frac{3-\sqrt{5}}{2}(b_0 - a_0); \quad z_0 = a_0 + b_0 - y_0, \quad \frac{3-\sqrt{5}}{2} = 0{,}38196 .$$

Шаг 4. Вычислить $f(y_k)$, $f(z_k)$.

Шаг 5. Сравнить $f(y_k)$ и $f(z_k)$:

а) если $f(y_k) \le f(z_k)$, то положить $a_{k+1} = a_k$, $b_{k+1} = z_k$

и $y_{k+1} = a_{k+1} + b_{k+1} - y_k$, $z_{k+1} = y_k$ (рис. 3, *а*) и перейти к шагу 6;

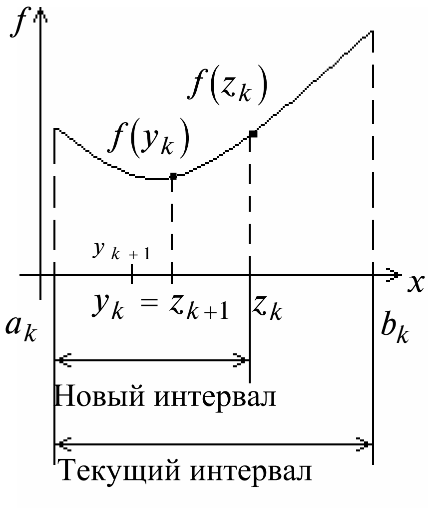
б) если $f(y_k) > f(z_k)$, то положить $a_{k+1} = y_k$, $b_{k+1} = b_k$

и $y_{k+1} = z_k$, $z_{k+1} = a_{k+1} + b_{k+1} - z_k$ (рис. 3, *б*).

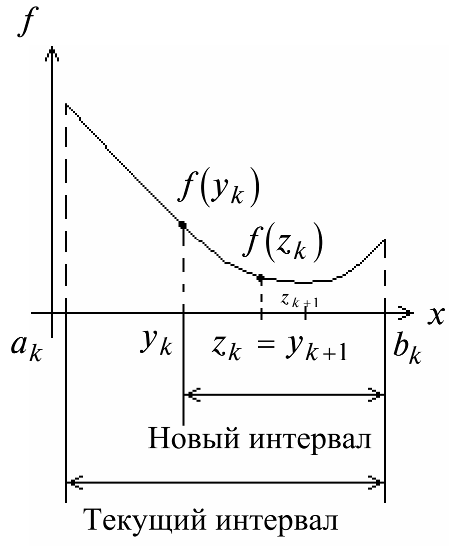
Шаг 6. Вычислить $\Delta = | a_{k+1} - b_{k+1} |$ и проверить условие окончания:

а) если $\Delta \le l$, процесс поиска завершить. Точка минимума принадлежит интервалу: $x^* \in [a_{k+1}, b_{k+1}]$. В качестве приближенного решения можно взять середину последнего интервала: $x^* \cong \dfrac{a_{k+1} + b_{k+1}}{2}$;

б) если $\Delta > l$, положить $k = k + 1$ и перейти к шагу 4.



а



б

Рис. 3

З а м е ч а н и я .

1. Текущие интервалы неопределенности имеют следующий вид: $L_0, L_2, L_3, L_4, \ldots$. Они отражают тот факт, что на первой итерации производится два вычисления функции, а на последующих – по одному.

2. Сокращение длины интервала неопределенности постоянно:

$$\frac{|L_0|}{|L_2|} = \frac{|L_2|}{|L_3|} = \frac{|L_3|}{|L_4|} = \ldots = \frac{1+\sqrt{5}}{2} \cong 1{,}618 .$$

А3. Метод квадратичной интерполяции

Стратегия поиска

Задается начальная точка и с помощью пробного шага находятся три опорные точки таким образом, чтобы они располагались как можно ближе к искомой точке минимума. В полученных точках вычисляются значения функции. Затем строится интерполяционный полином второй степени, проходящий через имеющиеся три точки. В качестве приближения точки минимума берется точка минимума полинома. Процесс поиска заканчивается, когда полученная точка отличается от наилучшей из трех опорных точек не более чем на заданную величину.

Алгоритм

Шаг 1. Задать начальную точку x_1, величину шага $\Delta x > 0$, ε_1 и ε_2 – малые положительные числа, характеризующие точность.

Шаг 2. Вычислить $x_2 = x_1 + \Delta x$.

Шаг 3. Вычислить $f(x_1) = f_1$ и $f(x_2) = f_2$.

Шаг 4. Сравнить $f(x_1)$ с $f(x_2)$:

а) если $f(x_1) > f(x_2)$, положить $x_3 = x_1 + 2\Delta x$ (рис. 4, *а*);

б) если $f(x_1) \le f(x_2)$, положить $x_3 = x_1 - \Delta x$ (рис. 4, *б*).

Шаг 5. Вычислить $f(x_3) = f_3$.

Шаг 6. Найти $F_{\min} = \min\{f_1, f_2, f_3\}$, $x_{\min} = x_i : f(x_i) = F_{\min}$.

Шаг 7. Вычислить точку минимума интерполяционного полинома, построенного по трем точкам:

$$\bar{x} = \frac{1}{2}\frac{(x_2^2 - x_3^2)f_1 + (x_3^2 - x_1^2)f_2 + (x_1^2 - x_2^2)f_3}{(x_2 - x_3)f_1 + (x_3 - x_1)f_2 + (x_1 - x_2)f_3},$$

и величину функции $f(\bar{x})$(рис. 4).

Если знаменатель в формуле для $\bar{x}$ на некоторой итерации обращается в нуль, то результатом интерполяции является прямая. В этом случае рекомендуется обозначить $x_1 = x_{\min}$ и перейти к шагу 2.

Шаг 8. Проверить выполнение условий окончания:

$$\left| \frac{F_{\min} - f(\bar{x})}{f(\bar{x})} \right| < \varepsilon_1 , \quad \left| \frac{x_{\min} - \bar{x}}{\bar{x}} \right| < \varepsilon_2 .$$

Тогда:

а) если оба условия выполнены, процедуру закончить и положить $x^* \cong \bar{x}$;

б) если хотя бы одно из условий не выполнено и $\bar{x} \in [x_1, x_3]$, выбрать наилучшую точку ($x_{\min}$ или $\bar{x}$) и две точки по обе стороны от нее. Обозначить эти точки в естественном порядке и перейти к шагу 6;

в) если хотя бы одно из условий не выполнено и $\bar{x} \notin [x_1, x_3]$, то положить $x_1 = \bar{x}$ и перейти к шагу 2.

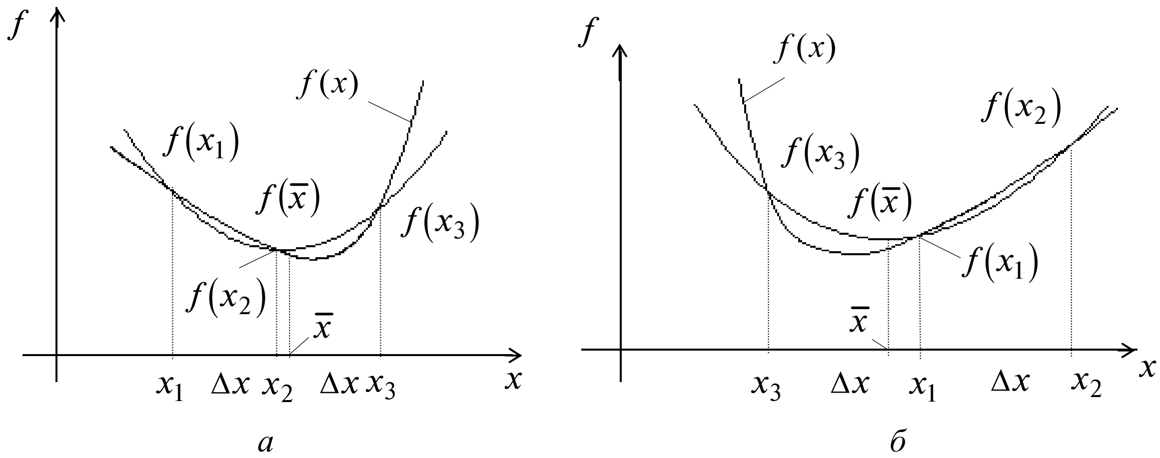


Рис. 4

З а м е ч а н и е. Для решения задачи одномерной минимизации также применяются: метод равномерного поиска, метод деления интервала пополам, метод Фибоначчи, кубической интерполяции.

Б. МЕТОД КОНФИГУРАЦИЙ

Постановка задачи

Требуется найти безусловный минимум функции $f(x)$ многих переменных, т.е. найти такую точку $x^* \in R^n$, что $f(x^*) = \min\limits_{x \in R^n} f(x)$.

Стратегия поиска

Метод конфигураций, или метод Хука–Дживса (Hooke–Jeeves), 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».

Исследующий поиск начинается в некоторой начальной точке x^0, называемой *старым базисом*. В качестве множества направлений поиска выбирается множество координатных направлений. Задается величина шага, которая может быть различной для разных координатных направлений и переменной в процессе поиска. Фиксируется первое координатное направление и делается шаг в сторону увеличения соответствующей переменной. Если значение функции в пробной точке меньше значения функции в исходной точке, шаг считается удачным. В противном случае необходимо вернуться в предыдущую точку и сделать шаг в противоположном направлении с последующей проверкой поведения функции. После перебора всех координат исследующий поиск завершается. Полученная точка называется *новым базисом* (на рис. 5 в точке x^0 произведен исследующий поиск и получена точка x^1 − новый базис). Если исследующий поиск с данной величиной шага неудачен, то она уменьшается и процедура продолжается. Поиск заканчивается, когда текущая величина шага станет меньше некоторой величины.

Поиск по образцу заключается в движении по направлению от старого базиса к новому (от точки x^0 через точку x^1, из точки x^1 через точку x^2, из x^2 через x^3 на рис. 5). Величина ускоряющего шага задается ускоряющим множителем λ. Успех поиска по образцу определяется с помощью исследующего поиска из полученной точки (например из точек 6, 11, 15 на рис. 5). Если при этом значение в наилучшей точке меньше, чем в точке предыдущего базиса, то поиск по образцу удачен (точки 6, 11 − результат удачного поиска по образцу, а точка 15 − неудачного). Если поиск по образцу неудачен, происходит возврат в новый базис, где продолжается исследующий поиск с уменьшенным шагом. На рис. 5 удачный поиск отображается сплошными линиями, а неудачный − штриховыми, числа соответствуют порождаемым алгоритмом точкам.

Обозначим через $d_1,\dots,d_n$ − координатные направления:

$$d_1 = \begin{pmatrix} 1 \\ 0 \\ \vdots \\ 0 \end{pmatrix}, \quad d_2 = \begin{pmatrix} 0 \\ 1 \\ \vdots \\ 0 \end{pmatrix}, \dots, \quad d_n = \begin{pmatrix} 0 \\ 0 \\ \vdots \\ 1 \end{pmatrix}.$$

При поиске по направлению d_i меняется только переменная x_i, а остальные переменные остаются зафиксированными.

x_2

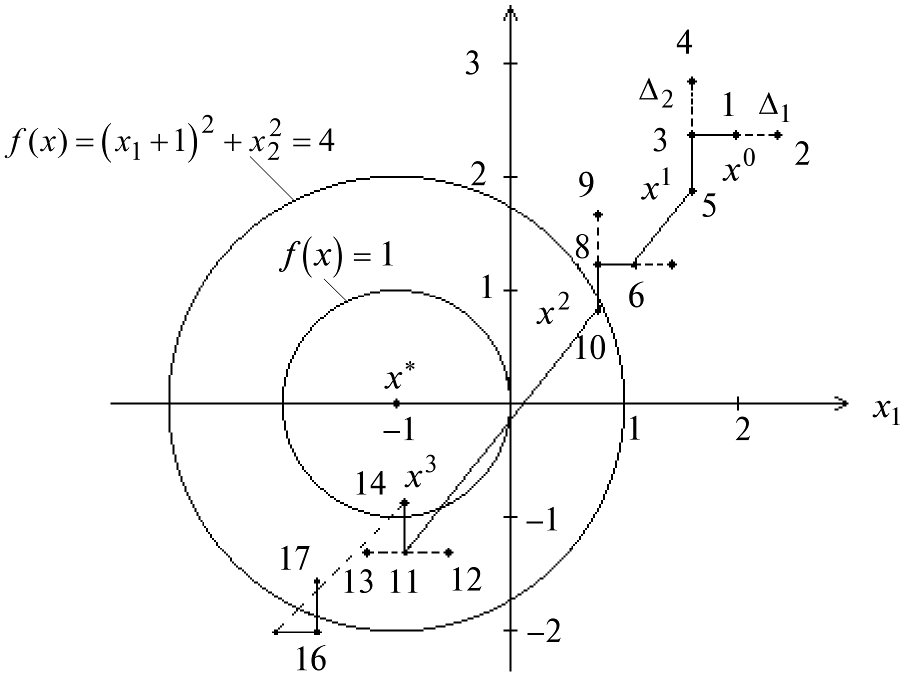


Рис. 5

Алгоритм

Шаг 1. Задать начальную точку x^0, число $\varepsilon > 0$ для остановки алгоритма, начальные величины шагов по координатным направлениям $\Delta_1, \ldots, \Delta_n \geq \varepsilon$, ускоряющий множитель $\lambda > 0$, коэффициент уменьшения шага $\alpha > 1$. Положить $y^1 = x^0$, $i = 1$, $k = 0$.

Шаг 2. Осуществить исследующий поиск по выбранному координатному направлению:

а) если $f(y^i + \Delta_i\, d_i) < f(y^i)$, т.е. $f(y_1^i, \ldots, y_i^i + \Delta_i, \ldots, y_n^i) < f(y_1^i, \ldots, y_i^i, \ldots, y_n^i)$, шаг считается удачным. В этом случае следует положить $y^{i+1} = y^i + \Delta_i\, d_i$ и перейти к шагу 3;

б) если в п. “а” шаг неудачен, то делается шаг в противоположном направлении. Если $f(y^i - \Delta_i\, d_i) < f(y^i)$, т.е. $f(y_1^i, \ldots, y_i^i - \Delta_i, \ldots, y_n^i) < f(y_1^i, \ldots, y_i^i, \ldots, y_n^i)$, шаг считается удачным. В этом случае следует положить $y^{i+1} = y^i - \Delta_i d_i$ и перейти к шагу 3;

в) если в пп. “а” и “б” шаги неудачны, положить $y^{i+1} = y^i$.

Шаг 3. Проверить условия:

а) если $i < n$, то положить $i = i + 1$ и перейти к шагу 2 (продолжить исследующий поиск по оставшимся направлениям);

б) если $i = n$, проверить успешность исследующего поиска:

- если $f(y^{n+1}) < f(x^k)$, перейти к шагу 4;
- если $f(y^{n+1}) \geq f(x^k)$, перейти к шагу 5.

Шаг 4. Провести поиск по образцу. Положить

$$x^{k+1} = y^{n+1}, \quad y^1 = x^{k+1} + \lambda(x^{k+1} - x^k), \quad i = 1, \quad k = k+1$$

и перейти к шагу 2.

Шаг 5. Проверить условие окончания:

а) если все $\Delta_i \le \varepsilon$, то поиск закончить: $x^* \cong x^k$;

б) для тех i, для которых $\Delta_i > \varepsilon$, уменьшить величину шага: $\Delta_i = \dfrac{\Delta_i}{\alpha}$. Положить $y^1 = x^k$, $x^{k+1} = x^k$, $k = k+1$, $i = 1$ и перейти к шагу 2.

З а м е ч а н и е. В алгоритме можно использовать одинаковую величину шага по координатным направлениям, т.е. вместо $\Delta_1, \Delta_2, \ldots, \Delta_n$ применять Δ.

B. МЕТОД ДЕФОРМИРУЕМОГО МНОГОГРАННИКА

Постановка задачи

Требуется найти безусловный минимум функции $f(x)$ многих переменных, т.е. найти такую точку $x^* \in R^n$, что $f(x^*) = \min\limits_{x \in R^n} f(x)$.

Стратегия поиска

В основу метода деформируемого многогранника, или метода Нелдера–Мида (Nelder–Mead), положено построение последовательности систем $n+1$ точек $x^i(k),\ i = 1,\ldots,n+1$, которые являются вершинами выпуклого многогранника. Точки системы $x^i(k+1),\ i = 1,\ldots,n+1$, на $(k+1)$-й итерации совпадают с точками системы $x^i(k),\ i = 1,\ldots,n+1$, кроме $i = h$, где точка $x^h(k)$ – наихудшая в системе $x^i(k),\ i = 1,\ldots,n+1$, т.е. $f(x^h(k)) = \max\limits_{1 \le i \le n+1} f(x^i(k))$. Точка $x^h(k)$ заменяется на другую точку по специальным правилам, описанным ниже. В результате многогранники деформируются в зависимости от структуры линий уровня целевой функции, вытягиваясь вдоль длинных наклонных плоскостей, изменяя направление в изогнутых впадинах и сжимаясь в окрестности минимума. Построение последовательности многогранников заканчивается, когда значения функции в вершинах текущего многогранника отличаются от значения функции в центре тяжести системы $x^i(k),\ i = 1,\ldots,n+1;\ i \ne h$, не более чем на величину $\varepsilon > 0$.

Алгоритм

Шаг 1. Задать координаты вершин многогранника $x^1,\ldots,x^{n+1}$; параметры отражения α, сжатия β, растяжения γ; число $\varepsilon > 0$ для остановки алгоритма. Положить $k = 0$ (последующие шаги 2–6 соответствуют текущему номеру k системы точек).

Шаг 2. Среди вершин найти «наилучшую» x^l и «наихудшую» x^h, соответствующие минимальному и максимальному значениям функции:

$$f\left(x^{l}\right)=\min_{j=1,\ldots,n+1} f\left(x^{j}\right); \qquad f\left(x^{h}\right)=\max_{j=1,\ldots,n+1} f\left(x^{j}\right),$$

а также точку x^s, в которой достигается второе по величине после максимального значение функции $f\left(x^s\right)$.

Шаг 3. Найти «центр тяжести» всех вершин многогранника, за исключением «наихудшей» x^h:

$$x^{n+2}=\frac{1}{n}\left[\sum_{j=1}^{n+1} x^{j}-x^{h}\right]=\frac{1}{n}\sum_{\substack{j=1\\ j\neq h}}^{n+1} x^{j}.$$

Шаг 4. Проверить условие окончания:

а) если $\sigma=\left\{\frac{1}{n+1}\sum_{j=1}^{n+1}\left[f\left(x^{j}\right)-f\left(x^{n+2}\right)\right]^{2}\right\}^{\frac{1}{2}}\leq\varepsilon$, процесс поиска можно завершить и в качестве приближенного решения взять наилучшую точку текущего многогранника: $x^* \cong x^l$;

б) если $\sigma>\varepsilon$, продолжать процесс.

Шаг 5. Выполнить операцию *отражения* «наихудшей» вершины через центр тяжести x^{n+2} (рис. 6, *а*):

$$x^{n+3}=x^{n+2}+\alpha\left(x^{n+2}-x^{h}\right).$$

Шаг 6. Проверить выполнение условий:

а) если $f\left(x^{n+3}\right)\leq f\left(x^{l}\right)$, выполнить операцию *растяжения* (рис. 6, *б*):

$$x^{n+4}=x^{n+2}+\gamma\left(x^{n+3}-x^{n+2}\right).$$

Найти вершины нового многогранника:

- если $f\left(x^{n+4}\right)<f\left(x^{l}\right)$, то вершина x^h заменяется на x^{n+4} (при $n=2$ многогранник будет содержать вершины x^1, x^3, x^6). Затем следует положить $k=k+1$ и перейти к шагу 2;
- если $f\left(x^{n+4}\right)\geq f\left(x^{l}\right)$, то вершина x^h заменяется на x^{n+3} (при $n=2$ многогранник будет содержать вершины x^1, x^3, x^5). Далее следует положить $k=k+1$ и перейти к шагу 2;

б) если $f\left(x^{s}\right)<f\left(x^{n+3}\right)\leq f\left(x^{h}\right)$, то выполнить операцию *сжатия* (рис. 6, *в*):

$$x^{n+5}=x^{n+2}+\beta\left(x^{h}-x^{n+2}\right).$$

Заменить вершину x^h на x^{n+5}, положить $k=k+1$ и перейти к шагу 2 (при $n=2$ многогранник будет содержать вершины x^1, x^3, x^7);

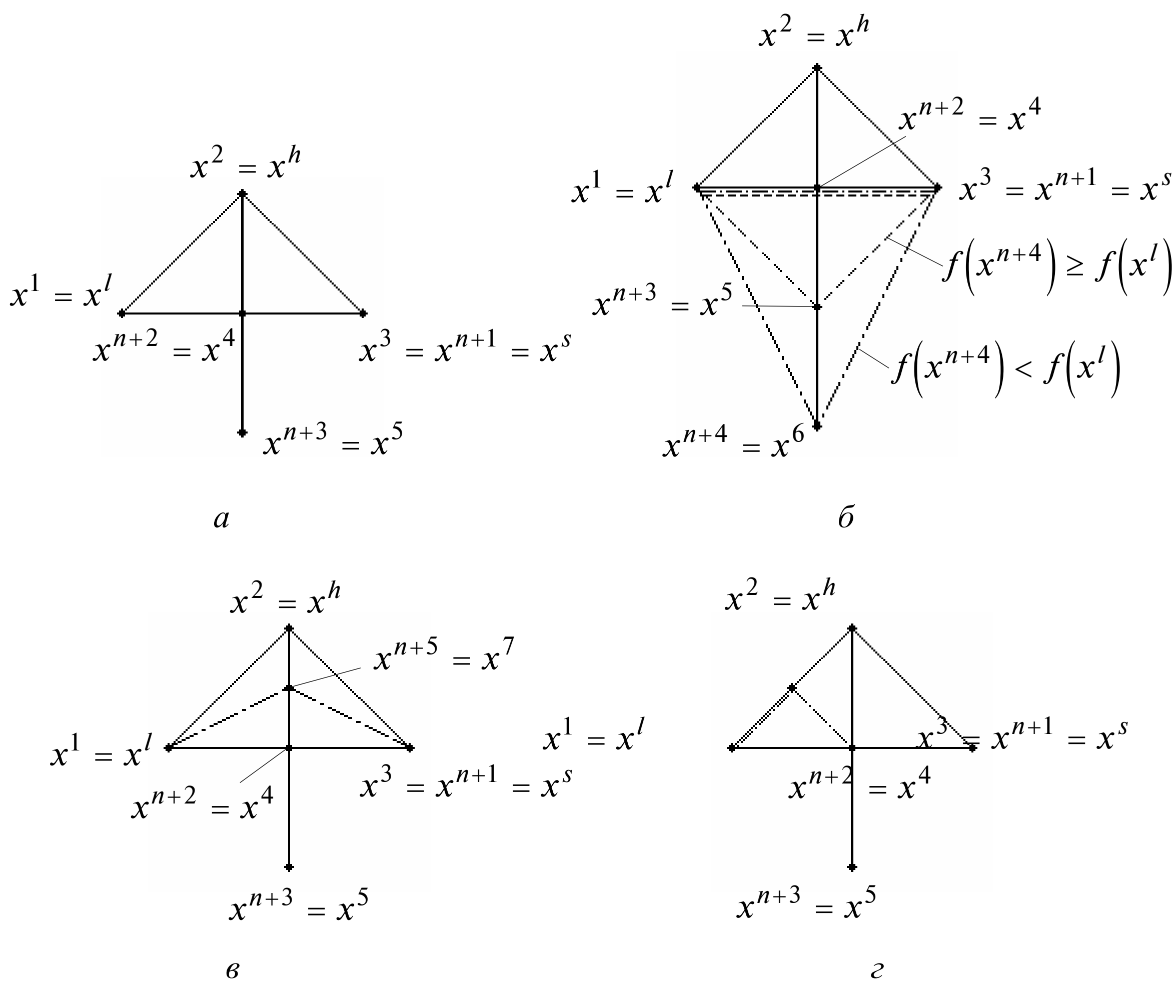


Рис. 6

в) если $f\left(x^l\right) < f\left(x^{n+3}\right) \le f\left(x^s\right)$, то вершину x^h заменить на x^{n+3}. При этом следует положить $k = k + 1$ и перейти к шагу 2;

г) если $f\left(x^{n+3}\right) > f\left(x^h\right)$, выполнить операцию *редукции* (рис. 6, *г*). Формируется новый многогранник с уменьшенными вдвое сторонами и вершиной x^l:

$$x^j = x^l + 0{,}5\left(x^j - x^l\right), \quad j = 1, \ldots, n + 1.$$

При этом следует положить $k = k + 1$ и перейти к шагу 2.

З а м е ч а н и е. Нелдер и Мид рекомендуют использовать параметры $\alpha = 1;\ \beta = 0{,}5;\ \gamma = 2$; Павиани (Paviani): $\alpha = 1$; $0{,}4 \le \beta \le 0{,}6$; $2{,}8 \le \gamma \le 3$; Паркинсон и Хатчинсон (Parkinson, Hutchinson): $\alpha = 2;\ \beta = 0{,}25;\ \gamma = 2{,}5$. В последнем случае в рамках операции отражения фактически выполняется растяжение.

Г. МЕТОДЫ СЛУЧАЙНОГО ПОИСКА

Постановка задачи

Требуется найти безусловный минимум функции $f(x)$ многих переменных, т.е. найти такую точку $x^* \in R^n$, что $f(x^*) = \min\limits_{x \in R^n} f(x)$.

Г.1. Адаптивный метод случайного поиска

Стратегия поиска

Задается начальная точка x^0. Каждая последующая точка находится по формуле

$$x^{k+1} = x^k + t_k\,\xi^k,$$

где $t_k > 0$ — величина шага; ξ^k — случайный вектор единичной длины, определяющий направление поиска; k — номер итерации. На текущей итерации при помощи генерирования случайных векторов ξ^k получаются точки, лежащие на гиперсфере радиуса t_k с центром в точке x^k (рис. 7). Если значение функции в полученной точке не меньше, чем в центре, шаг считается неудачным (точки y^1, y^2 при поиске из x^0; y^1, y^3 при поиске из x^1). Если число неудачных шагов из текущей точки достигает некоторого числа M, дальнейший поиск продолжается из той же точки, но с меньшим шагом до тех пор, пока он не станет меньше заранее заданной величины R. Если же значение функции в полученной точке меньше, чем в центре, шаг считается удачным, и в найденном направлении делается увеличенный шаг, играющий роль ускоряющего шага (как при поиске по образцу в методе конфигураций). Если при этом значение функции снова меньше, чем в центре, направление считается удачным и дальнейший поиск продолжается из этой точки (точки $z^3 = x^1$ при поиске из x^0, $z^4 = x^2$ при поиске из x^1). Если же значение функции не стало меньше, чем в центре, направление считается неудачным и поиск продолжается из старого центра (в точке y^2 при поиске из x^1 функция меньше, чем в x^1, а в точке z^2 уже не меньше, поэтому направление $\left(z^2 - x^1\right)$ — неудачное).

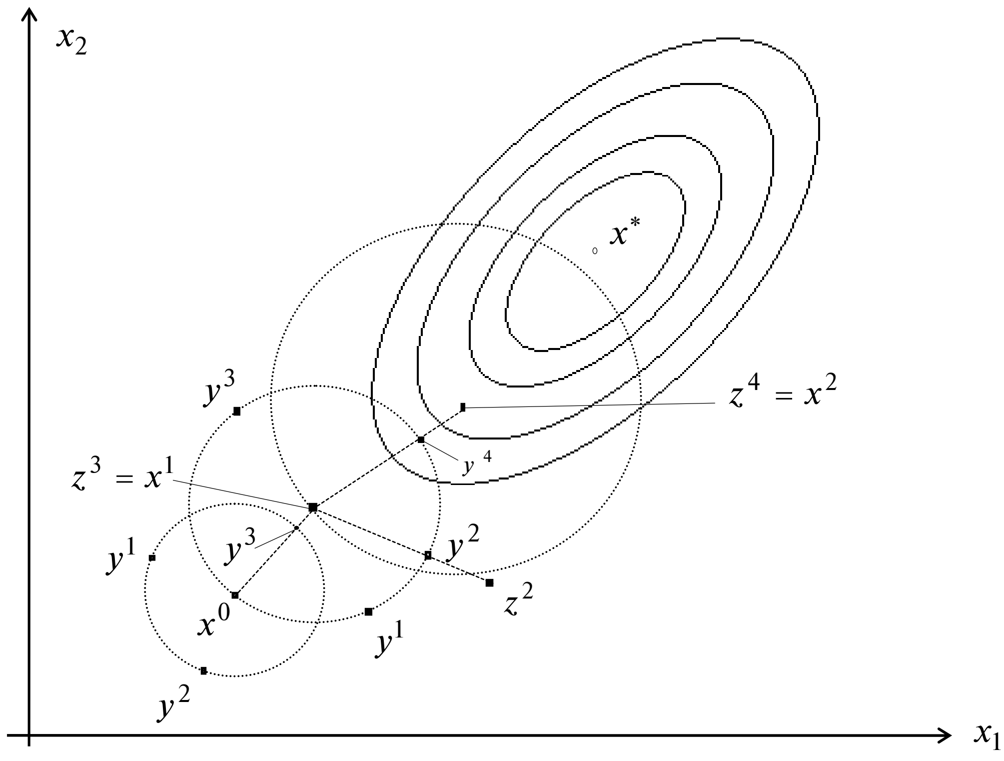


Рис. 7

Алгоритм

Шаг 1. Задать начальную точку x^0, коэффициенты расширения $\alpha \ge 1$ и сжатия $0 < \beta < 1$, M — максимальное число неудачно выполненных испытаний на текущей итерации, $t_0 = 1$ — начальную величину шага, R - минимальную величину шага, N — максимальное число итераций. Положить $k = 0, j = 1$.

Шаг 2. Получить случайный вектор $\xi^j = \left(\xi_1{}^j, \ldots, \xi_n{}^j\right)^T$, где $\xi_i{}^j$ — случайная величина, равномерно распределенная на интервале $[-1,1]$.

Шаг 3. Вычислить $y^j = x^k + t_k \dfrac{\xi^j}{\left\|\xi^j\right\|}$.

Шаг 4. Проверить выполнение условий:

а) если $f\left(y^j\right) < f\left(x^k\right)$, шаг удачный. Положить $z^j = x^k + \alpha\left(y^j - x^k\right)$ и определить, является ли текущее направление $y^j - x^k$ удачным:

- если $f\left(z^j\right) < f\left(x^k\right)$, то направление поиска удачное. Положить $x^{k+1} = z^j$, $t_{k+1} = \alpha t_k$, $k = k+1$ и проверить условие окончания. Если $k < N$, положить $j = 1$ и перейти к шагу 2. Если $k = N$, поиск завершить: $x^* \cong x^k$;
- если $f\left(z^j\right) \ge f\left(x^k\right)$, направление поиска неудачное, перейти к шагу 5;

б) если $f\left(y^j\right) \ge f\left(x^k\right)$, шаг неудачный и перейти к шагу 5.

Шаг 5. Оценить число неудачных шагов из текущей точки:

а) если $j < M$, следует положить $j = j + 1$ и перейти к шагу 2;

б) если $j = M$, проверить условие окончания:

- если $t_k \le R$, процесс закончить: $x^* \cong x^k, f\left(x^*\right) \cong f\left(x^k\right)$;
- если $t_k > R$, положить $t_k = \beta t_k, j = 1$ и перейти к шагу 2.

З а м е ч а н и я.

1. Величина $\xi_i{}^j$, равномерно распределенная на интервале $[-1,1]$, генерируется обычно с помошью датчиков псевдослучайных чисел на ЭВМ. Вырабатывается случайная величина η_i^j, равномерно распределенная на $[0,1]$, а затем используется линейное преобразование: $\xi_i^j = 2\eta_i^j - 1$.

2. Шумер и Стейглиц (Schumer, Steiglitz) рекомендуют следующие параметры алгоритма: $\alpha = 1{,}618$; $\beta = 0{,}618$; $M = 3n$. При $\alpha = 1$ точка z^j на шаге 4 совпадает с y^j, т.е. аналог поиска по образцу не производится. Начальный шаг $t_0 \ge R$ можно задать произвольно.

3. Если выполнено условие окончания $t_k \le R$, то в качестве ответа можно использовать любую точку внутри шара с радиусом t_k и центром в точке x^k.

Г.2. Метод случайного поиска с возвратом при неудачном шаге

Стратегия поиска

Задается начальная точка x^0. Каждая последующая точка находится по формуле

$$x^{k+1} = x^k + t_k \, \xi^k,$$

где $t_k > 0$ – величина шага; ξ^k – случайный вектор единичной длины, определяющий направление поиска; k – номер итерации. На текущей итерации при помощи генерирования случайных векторов ξ^k получаются точки, лежащие на гиперсфере радиуса t_k с центром в точке x^k (рис. 8). Если значение функции в полученной точке не меньше, чем в центре, шаг считается неудачным (точки y^1, y^2 при поиске из x^0; y^1, y^2, y^3 при поиске из x^1), происходит возврат в текущий центр и поиск продолжается. Если число неудачных шагов из текущей точки достигает некоторого числа M, дальнейший поиск продолжается из той же точки, но с меньшим шагом до тех пор, пока он не станет меньше заранее заданной величины R. Если же значение функции в полученной точке меньше, чем в центре, шаг считается удачным и дальнейший поиск продолжается из этой точки.

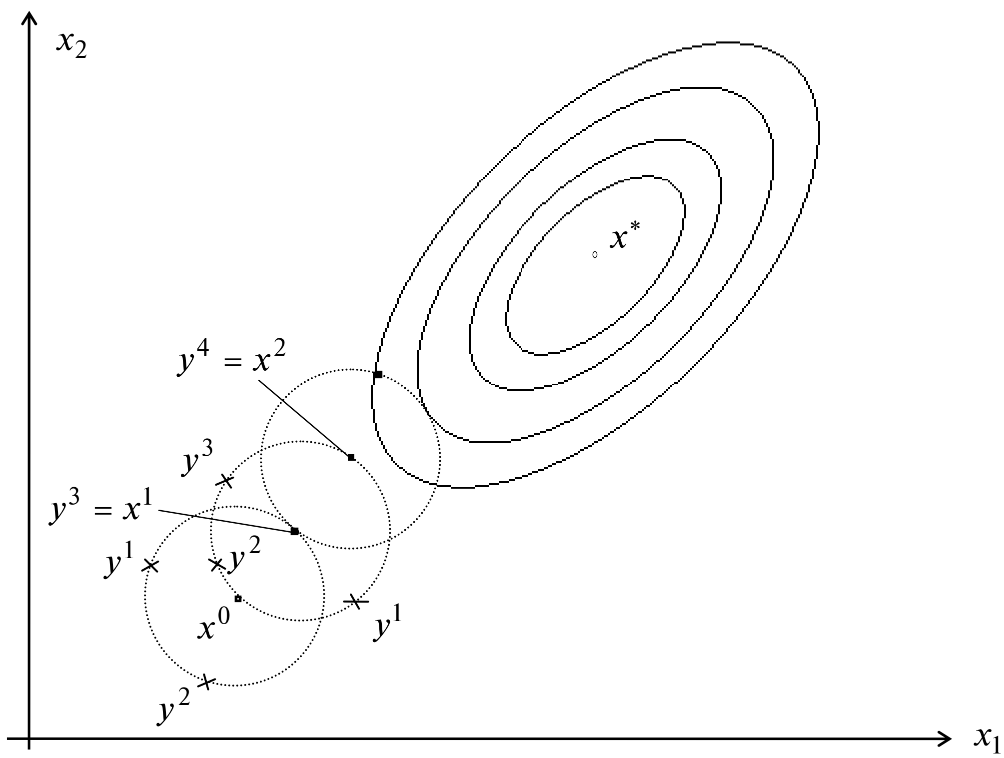


Рис. 8

Г.3. Метод наилучшей пробы

Стратегия поиска

Задается начальная точка x^0. Каждая последующая точка находится по формуле

$$x^{k+1} = x^k + t_k\,\xi^k,$$

где $t_k > 0$ – величина шага; ξ^k – случайный вектор единичной длины, определяющий направление поиска; k – номер итерации. На текущей итерации при помощи генерирования случайных векторов ξ^k получается M точек $y^1,\dots,y^M$, лежащих на гиперсфере радиуса t_k с центром в точке x^k (рис. 9). Среди полученных точек выбирается точка y^m, в которой значение функции наименьшее. Если в выбранной точке значение функции меньше, чем в центре, то дальнейший поиск продолжается из этой точки. Иначе поиск продолжается из старого центра, но с меньшим шагом до тех пор, пока он не станет меньше заранее заданной величины R.

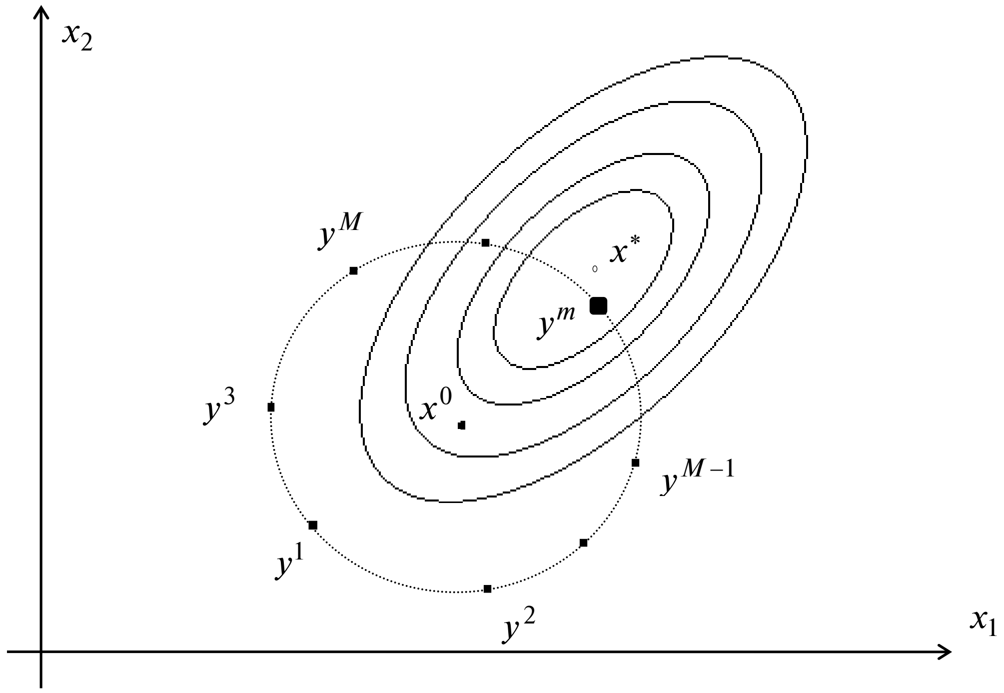


Рис. 9