МЕТОДЫ ОПТИМИЗАЦИИ

Овсянникова Наталья Владимировна
Г-234
IP 9652
nvovsyannikova@mephi.ru

ОТЧЕТНОСТЬ

ПРАКТИКА:

К/р1 – 8 неделя (min 15, max 25)

К/р2 – 15 неделя (min 15, max 25)

ТЕОРИЯ – экзамен

(min 30, max 50)

МАТЕМАТИЧЕСКОЕ ПРОГРАММИРОВАНИЕ

МАТЕМАТИЧЕСКОЕ ПРОГРАММИРОВАНИЕ – математическая дисциплина, посвящённая теории и методам решения задач о нахождении экстремумов функций на множествах, определяемых линейными и нелинейными ограничениями (равенствами и неравенствами).

Методами математического программирования решаются задачи: распределения ресурсов, планирование выпуска продукции, ценообразования, транспортные задачи и т. п.

Постановка задачи: найти экстремум (максимум или минимум) целевой функции

$Z(X) = f(x_1, x_2, \ldots x_n) \rightarrow \max (\min)$

и соответствующие ему переменные при условии, что эти переменные удовлетворяют системе ограничений:

$\varphi_i (x_1, x_2, \ldots x_n) = 0, \ i = 1, 2, \ldots, l$

$\varphi_i (x_1, x_2, \ldots x_n) \leq (\geq)\ 0, \ i = l+1, l+2, \ldots, m$

Система ограничений включает в себя систему уравнений и неравенств, которым удовлетворяют переменные задачи и которые следуют из ограниченности ресурсов или других экономических или физических условий, например положительности переменных и т. п.

Допустимое и оптимальное решение

Допустимым решением задачи математического программирования называется любой n-мерный вектор X = (x_1, x_2,…x_n), удовлетворяющий системе ограничений.
Множество допустимых решений задачи образует область допустимых решений (ОДР).

Оптимальным решением задачи математического программирования называется такое допустимое решение задачи, при котором целевая функция достигает экстремума.

В силу исторических причин задача нахождения оптимального решения (или оптимального плана) получило обобщенное название задачи математического программирования (что не имеет прямого отношения к программированию на ЭВМ), в связи с принятым в термином «program», подразумевающим план/алгоритм.

Примеры задач МП

1. Задача о диете

Имеется два продукта 1 – овощи и 2 – мясо;
заданы два типа питательных веществ: 1 – белки и 2 – углеводы.
Необходимо составить диету, имеющую минимальную стоимость и обеспечивающую необходимую насыщенность питательными веществами.

Математическая постановка задачи:
Матрица содержания питательных веществ в продуктах $A = \begin{pmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{pmatrix}$
a_{11} - содержание белков в овощах
a_{21} - содержание углеводов в овощах
a_{12} - содержание белков в мясе
a_{22} - содержание углеводов в мясе

Вектор ограничений
b_1 - минимальное количество белков, которое должно содержаться в диете;
b_2 – углеводов

$$\bar{b} = \begin{pmatrix} b_1 \\ b_2 \end{pmatrix}$$

Вектор цен

$$\bar{c} = \begin{pmatrix} c_1 \\ c_2 \end{pmatrix}$$

Вектор плана (диета)

$$\bar{x} = \begin{pmatrix} x_1 \\ x_2 \end{pmatrix}$$

Необходимо минимизировать стоимость диеты

$$z = c_1 x_1 + c_2 x_2$$

при ограничениях

$$\begin{cases} a_{11}x_1 + a_{12}x_2 \geq b_1 \\ a_{21}x_1 + a_{32}x_2 \geq b_2 \end{cases}$$

Примеры задач МП

2. Задача о коммивояжере. Состоит в отыскании наилучшего маршрута для коммивояжера, который должен объехать все порученные ему города и вернуться назад в кратчайший срок или с наименьшими затратами на проезд. О сложности этой задачи говорит тот факт: если городов 4, то число возможных маршрутов - 6, а уже при 11 городах существует более 3,5 млн допустимых маршрутов. В общем случае, когда число городов n, количество маршрутов равно (n-1)!

3. Распределительные задачи. Класс экономико-математических задач, связанных с распределением ресурсов по работам, которые необходимо выполнить. Если ресурсов достаточно, чтобы каждую работу выполнить наиболее эффективно, задача не возникает. В обратом случае переброска ресурсов с одной работы на другую приводит к изменению общей эффективности. Поэтому распределительная задача заключается в отыскании наилучшего распределения ресурсов, при котором либо максимизируется общий доход, либо минимизируются затраты.

4. Задачи теории расписаний. Теорией расписаний называется совокупность моделей календарного расписания. Модели теории расписаний позволяют, например, решать такие задачи, как определение оптимальной последовательности обработки деталей на станках, планирование работы производственного участка и т.п.

5. Управление запасами. Это комплекс методов, предназначенных для оптимизации запасов, т. е. ресурсов, находящихся на хранении. В качестве целевой функции выступают суммарные затраты на содержание запасов, на складские операции, потери от порчи при хранении, моральное старение и пр. Управляемыми переменными являются объем запасов, частота и сроки их пополнения и т. п.

Примеры задач МП

6. Задачи массового обслуживания. Это класс задач, заключающихся в нахождении параметров систем массового обслуживания.
Критериями качества систем массового обслуживания являются:
· вероятность обслуживания заявки или задержки в обслуживании;
· математическое ожидание числа удовлетворенных (задержанных) заявок за фиксированное время;
· математическое ожидание числа занятых каналов;
· математическое ожидание длины очереди.
Наиболее важным критерием оптимальности являются средние суммарные потери от ожидания требований, с одной стороны, и простоя каналов обслуживания – с другой.

7. Задача о размещении складов. Заключается в минимизации общей суммы транспортных и складских расходов при следующих ограничениях: с каждого завода должна быть отгружена вся продукция, емкость любого склада не должна быть переполнена, потребности всех покупателей должны быть удовлетворены.

8. Задачи о раскрое. Метод решения таких задач помогает с наименьшими отходами использовать листы металла, стекла, картона и др. материалов при раскрое их на заданное количество деталей различного размера. При правильной постановке задачи применение методов линейного программирования гарантирует сокращение отходов до минимально возможного. Часто на предприятиях отходы сокращаются в несколько раз.

Классификация задач оптимизации

1. По характеру взаимосвязи между переменными – **а) линейные б) нелинейные**

В случае а) все функциональные связи в системе ограничений и функция цели – линейные функции; наличие нелинейности хотя бы в одном из упомянутых элементов приводит к случаю б).

2. По характеру изменения переменных – **а) непрерывные б) дискретные**

В случае а) значения каждой из управляющих переменных могут заполнять сплошь некоторую область действительных чисел;
в случае б) все или хотя бы одна переменная могут принимать изолированные числовые значения из некоторого множества.

3. По учету фактора времени **а) статические б) динамические**

В задачах а) моделирование и принятие решений осуществляются в предположении о независимости от времени элементов модели в течение периода времени, на который принимается планово-управленческое решение.
В случае б) необходимо учитывать фактор времени.

4. По наличию информации о переменных **а) задачи в условиях полной определенности (детерминированные) б) задачи в условиях неполной информации в) задачи в условиях неопределенности**

В задачах б) отдельные элементы являются вероятностными величинами, однако известны или дополнительными статистическими исследованиями могут быть установлены их законы распределения. В случае в) можно сделать предположение о возможных исходах случайных элементов, но нет возможности сделать вывод о вероятностях исходов.

5. По числу критериев оценки альтернатив **а) простые, однокритериальные б) сложные, многокритериальные**

В задачах а) экономически приемлемо использование одного критерия оптимальности или удается специальными процедурами (например, «взвешиванием приоритетов») свести многокритериальный поиск к однокритериальному.

Классификация методов МП

В зависимости от особенностей целевой функции и функций, задающих ограничения, в математическом программировании принято выделять ряд разделов:

- Линейное программирование
- Параметрическое программирование
- **Нелинейное программирование**
- Выпуклое программирование
- Динамическое программирование
- Дискретное программирование
- Стохастическое программирование
- Геометрическое программирование

ε

Примерный план курса

- Безусловная одномерная оптимизация
- Безусловная многомерная оптимизация
- Условная многомерная оптимизация
- Многокритериальная оптимизация

Безусловная одномерная оптимизация

1 Алгоритм равномерного поиска
2 Алгоритм деления пополам (дихотомии)
3 Алгоритм Фибоначчи
4 Алгоритм золотого сечения
5 Метод хорд (секущих)
6 Метод касательных (Ньютона)
7 Одномерный метод Монте-Карло
8 Метод аппроксимирующих моделей

ОДНОМЕРНАЯ МИНИМИЗАЦИЯ

1. Алгоритм равномерного поиска

Найти минимум одномерной унимодальной функции Φ(x) , определенной в замкнутой области допустимых значений D=[a;b]

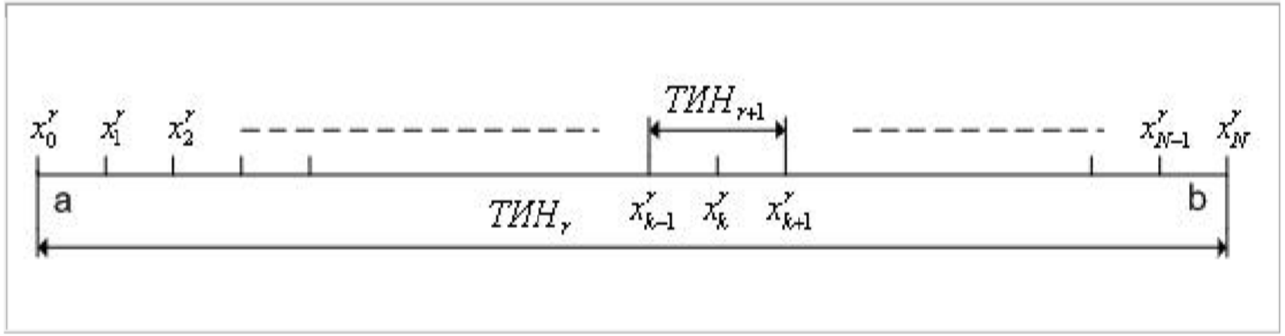
$$\min_{[a;b]} \Phi(x) = \Phi(x^*)$$

Текущий интервал неопределенности будем обозначать ТИН, а его длину |ТИН|.

В алгоритме равномерного поиска испытания проводятся в точках, которые определяются путем равномерного деления интервала [a;b] на N одинаковых подынтервалов. Из вычисленных значений функции Φ(x) выбирается наименьшее. Алгоритм относится к классу пассивных методов поиска.

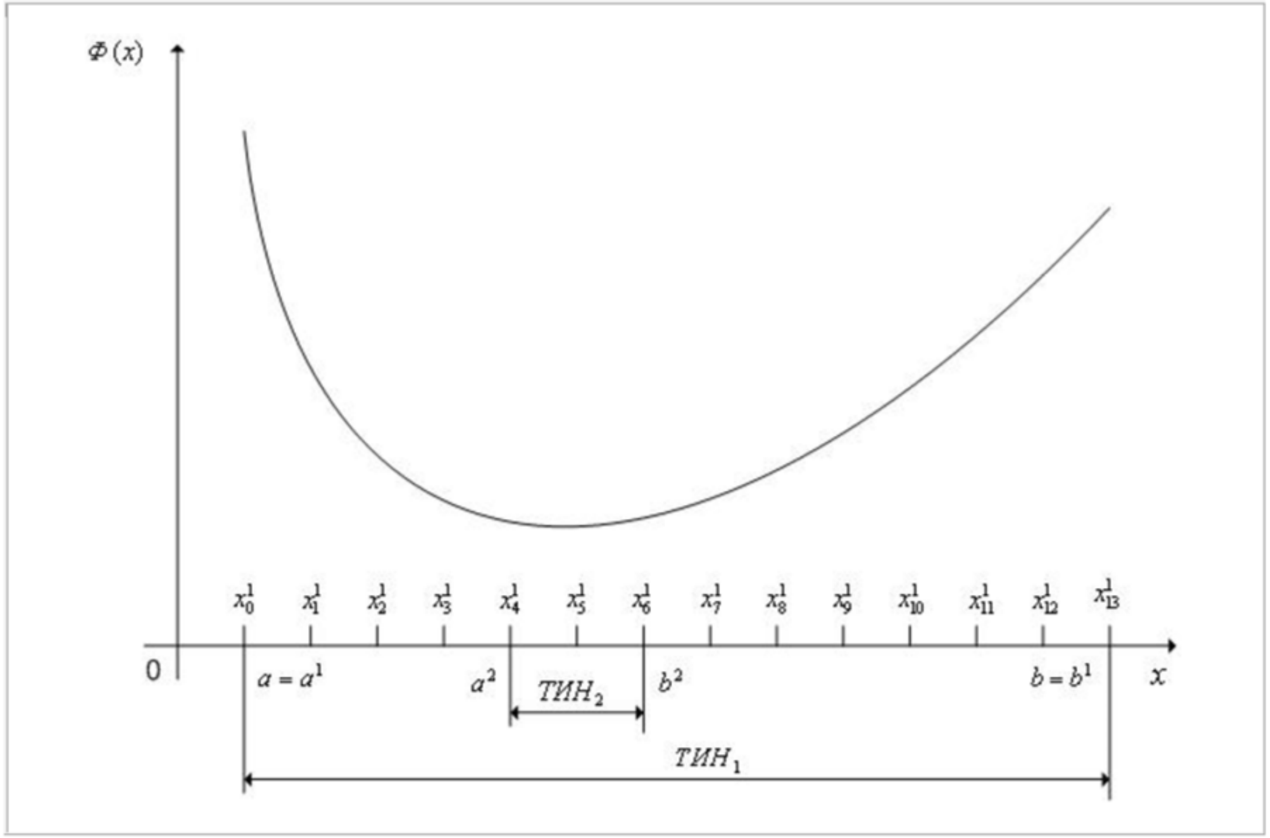
Алгоритм равномерного поиска

1. Выполняем присваивания: номер итерации r=1, $a^1=a$, $b^1=b$, ТИН1=$[a^1;b^1]$
2. На текущем ТИН строим равномерную сетку с N+1 узлами
3. Вычисляем значения функции Φ(x) в узлах построенной сетки: $\Phi(x_0),\dots,\Phi(x_N)$
4. Находим минимальное из этих значений $\min(\Phi(x_0),\dots,\Phi(x_N)) = \Phi(x_k)$
5. Выполняем присваивания $a^{r+1}=x_{k-1}$, $b^{r+1}=x_{k+1}$, $\text{ТИН}^{r+1}=[a^{r+1}; b^{r+1}]$
6. Если $|\text{ТИН}^{r+1}|<epsx$, то заканчиваем вычисления. Иначе r=r+1 и переходим на п.2. Здесь epsx– требуемая точность решения.



В качестве приближенного значения точки минимума x* с равными основаниями может быть принята любая точка последнего ТИН.
После одной итерации алгоритма равномерного поиска ТИН уменьшается в N/2 раз.
Количество   итераций r, необходимых для нахождения минимума функции с точностью epsx , может быть найдено из условия

$$\left(\frac{2}{N}\right)^r (b-a) \le epsx$$



2. Метод деления пополам (дихотомии)

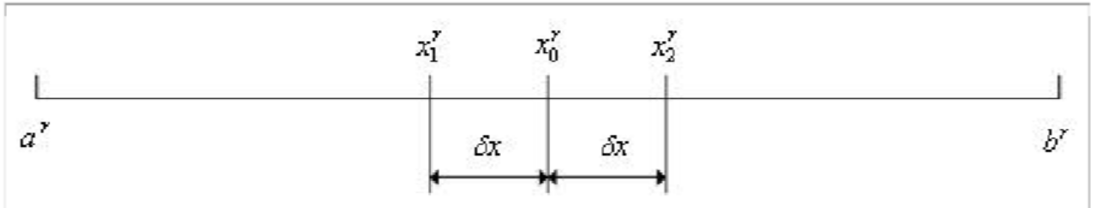
Найти минимум одномерной унимодальной функции Φ(x) , определенной в замкнутой области допустимых значений D=[a;b]

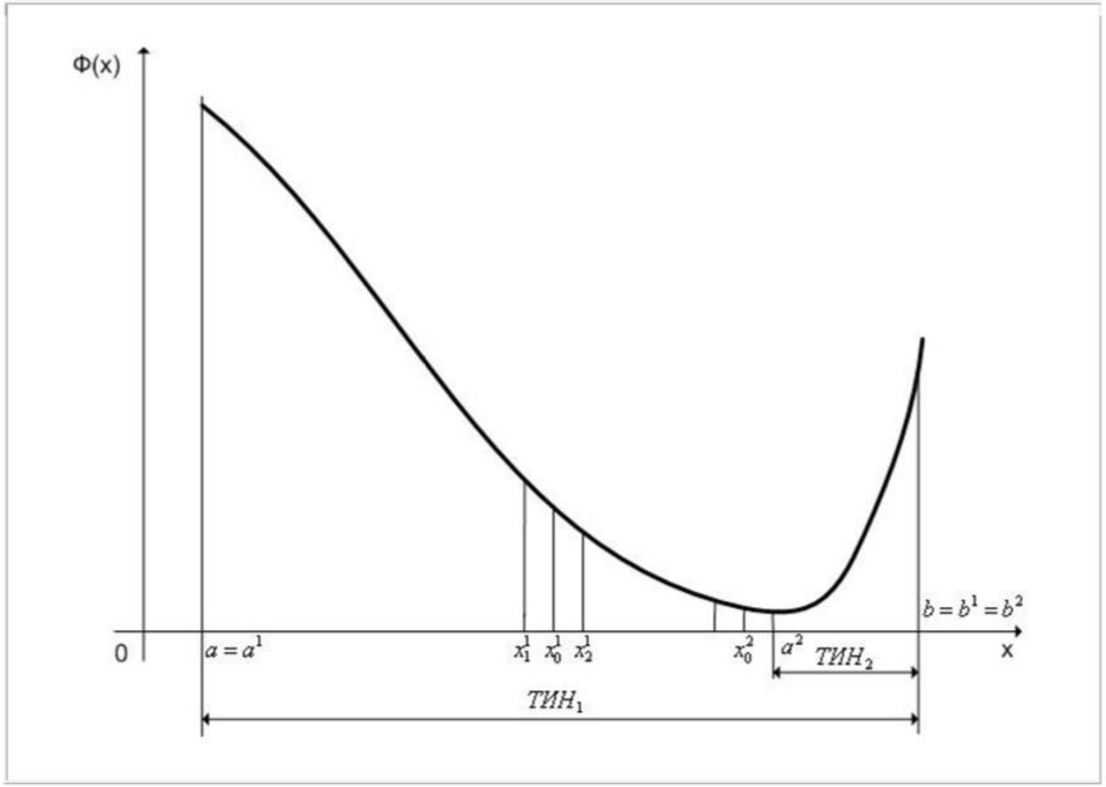
$$\min_{[a;b]} \Phi(x) = \Phi(x^*)$$

В алгоритме деления пополам или алгоритме равномерного дихотомического поиска испытания проводятся парами. Координаты каждой последующей пары испытаний разнесены между собой на величину dx<epsx, где epsx – требуемая точность решения. Испытания производятся в середине ТИН. По значениям Φ(x) , полученным в этих точках, одна половина ТИН в силу унимодальности функции Φ(x) исключается из дальнейшего рассмотрения. Алгоритм относится к классу методов последовательного поиска.

Алгоритм деления пополам

1. Выполняем присваивания r=1, a^1=a, b^1=b, ТИН^1=[a^1;b^1]
2. Вычисляем величины $x_0^r = (a^r + b^r)/2$, $x_1^r = x_0^r - dx/2$,$x_2^r = x_0^r + dx/2$,
3. Вычисляем значения Φ(x_1^r), Φ(x_2^r)
4. Если Φ(x_1^r) < Φ(x_2^r), то $a^{r+1} = a^r$, $b^{r+1} = x_0^r$ ТИН$^{r+1}$=[a^{r+1};b^{r+1}]
Иначе $a^{r+1} = x_0^r$ $b^{r+1} = b^r$
5. Если |ТИН$^{r+1}$|<epsx, то заканчиваем вычисления. Иначе r=r+1 переходим на п.2.





В качестве приближенного значения точки минимума с равными основаниями может быть принята любая точка последнего текущего интервала неопределенности.

Легко видеть, что после одной итерации алгоритма равномерного поиска ТИН уменьшается в 2 раза. Поэтому количество итераций r, необходимых для нахождения минимума функции с точностью epsx , находится из условия

$$(b-a)/2^r \leq epsx$$

3. Метод Фибоначчи

Рассмотрим следующую задачу условной оптимизации: найти минимум одномерной унимодальной функции Φ(x) , определенной в замкнутой области допустимых значений D=[a;b]

min*[a;b]Φx=Φ(x*)*

Числа Фибоначчи задаются рекуррентным соотношением

$$\gamma_0 = \gamma_1 = 1$$

$$\gamma_n = \gamma_{n-1} + \gamma_{n-2}, \quad n \geq 2 - \textit{задается заранее}$$

или

$$\gamma_n = \left[(1/\beta)^{n+1} - (-\beta)^{n+1}\right] / \sqrt{5}, \quad \textit{где}$$

$$\beta = \left(\sqrt{5} - 1\right)/2$$

k	0	1	2	3	4	5	6	7	8	…
γ^k	1	1	2	3	5	8	13	21	34	…

При больших k отношение двух соседних чисел Фибоначчи примерно постоянно и равно β=0,618.

Алгоритм Фибоначчи включает в себя два этапа.

Первый этап состоит из (N-1) итерации r=1,…,N-1

Рассмотрим схему r-й итерации, когда ТИН^r=[a^r;b^r]

1. Вычисляем величины $x_{1r}=a_r+|ТИН_r|\gamma_{N-1-r}/\gamma_{N+1-r}$

   $x_{2r}=a_r+|ТИН_r|\gamma_{N-r}/\gamma_{N+1-r}$

2. Вычисляем значения Ф(x_{1r}), Ф(x_{2r})

3. Если Ф(x_{1r}) < Ф(x_{2r}), то $a_{r+1}=a_r$,

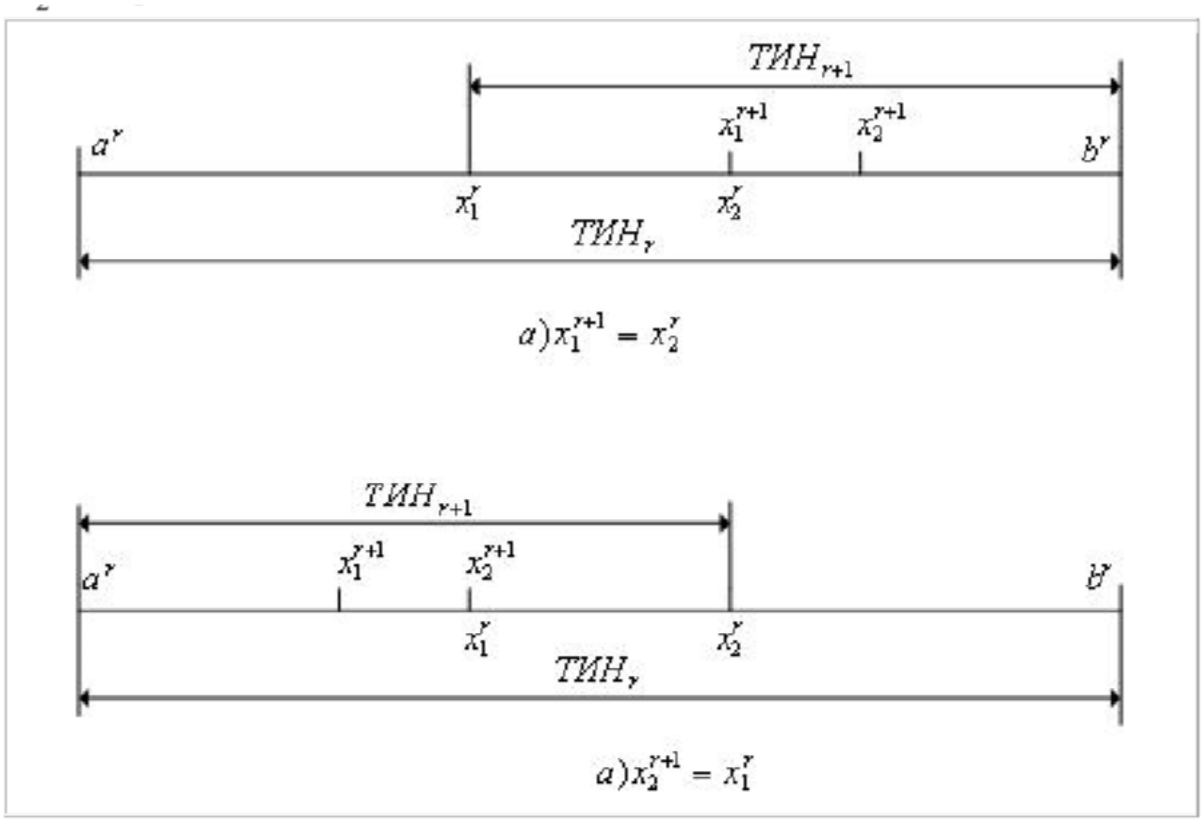
   $b_{r+1}=x_{2r}$

   ТИН$^{r+1}$=[a^{r+1};b^{r+1}]

Иначе $a_{r+1}=x_1$

$b_{r+1}=b_r$

ТИН$^{r+1}$=[a^{r+1};b^{r+1}]



Алгоритм Фибоначчи обладает тем свойством, что после выполнения N-ой итерации сужения ТИН не происходит.

Второй этап призван решить, по какую сторону от точки x^{N-1} лежит точка минимума функции Ф(x) :

1. Находим точку $x^N = x^{N-1}+dx$, где dx< |ТИН$^{N-1}$| (свободный параметр)
2. Вычисляем значение функции Ф(x^N).

Если Ф(x^N)> Ф(x^{N-1}), то ТИН$_N$=[a^{N-1};x^{N-1}]

Иначе ТИН$_N$=[x^{N-1};b^{N-1}]

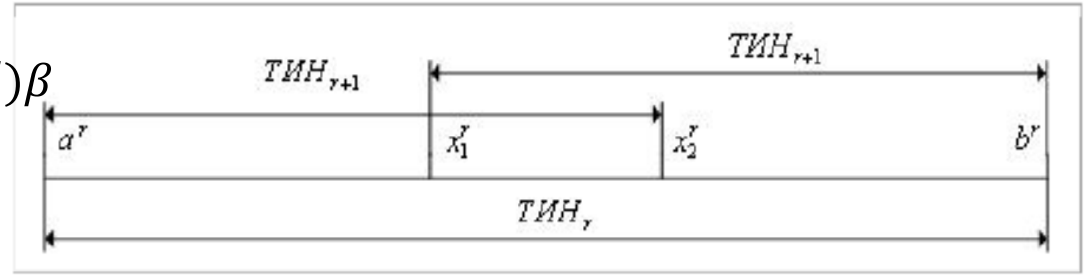
В качестве приближенного значения точки минимума x* с равными основаниями может быть принята любая точка ТИН$_N$.

Точность $(b-a)/\gamma^N \leq epsx$

4. Метод золотого сечения

Говорят, что точка С выполняет золотое сечение отрезка [A;B], если
(C – A)/(B – A) = β
Где β является корнем уравнения $\beta^2 + \beta - 1 = 0$
β =$\left(\sqrt{5}-1\right)/2 \approx 0.618$

Алгоритм золотого сечения относится к классу последовательных методов поиска.
1. Выполняем присваивания r=1, $a^1=a$, $b^1=b$, $ТИН^1=[a^1;b^1]$
2. Вычисляем величины $x_1^r = b^r - (b^r - a^r)\beta \quad x_2^r = a^r + (b^r - a^r)\beta$
3. Вычисляем значения Φ(x_1^r), Φ(x_2^r)
4. Если Φ(x_1^r) < Φ(x_2^r), то то $a^{r+1} = a^r$, $b^{r+1} = x_2^r$ $ТИН^{r+1}=[a^{r+1};b^{r+1}]$
Иначе $a^{r+1} = x_1^r \quad b^{r+1} = b^r$ $ТИН^{r+1}=[a^{r+1};b^{r+1}]$
5. Если $|ТИН^{r+1}|$<epsx, то заканчиваем вычисления. Иначе – выполняем присваивание r=r+1 и переходим на п.2. Здесь epsx– требуемая точность решения.



В качестве приближенного значения точки минимума с равными основаниями может быть принята любая точка последнего текущего интервала неопределенности.
В результате одной итерации алгоритма золотого сечения длина ТИН сокращается в β раз, поэтому количество итераций, необходимых для достижения точности epsx, находится из условия

$$epsx \le (b-a)\beta^N$$

При больших N алгоритм Фибоначчи практически идентичен алгоритму золотого сечения.

5. Метод секущих (хорд)

найти минимум одномерной унимодальной функции Ф(x) , определенной в замкнутой области допустимых значений D=[a;b]

$\min_{[a;b]} \Phi x = \Phi(x^*)$

Необходимым условием минимума функции является условие Ф’(x)=0, x϶[a;b]

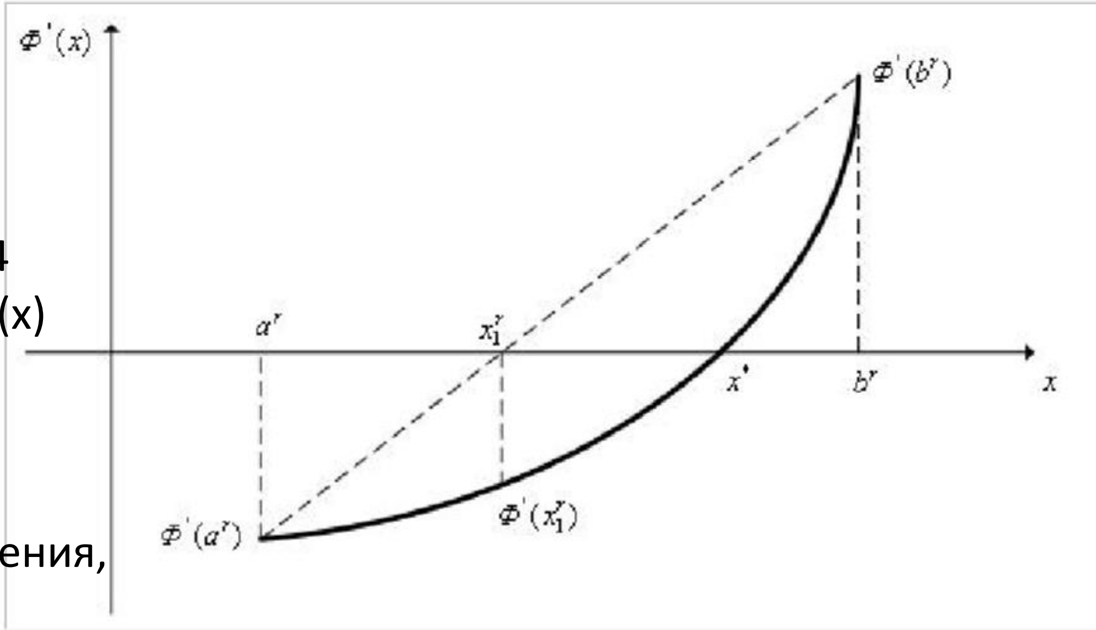
Алгоритм метода хорд

1. Выполняем присваивания r=1, $a^1=a$, $b^1=b$
2. Вычисляем значения производных Ф’(a^r), Ф’(b^r)
3. Если производные Ф’(a^r), Ф’(b^r) имеют одинаковые знаки – завершаем вычисления (точки a и b выбраны неверно), иначе п.4
4. Вычисляем приближение x_1^r к стационарной точке функции Ф(x)

$$x_1^r = b^r - \frac{\Phi'(b^r)(b^r - a^r)}{\Phi'(b^r) - \Phi'(a^r)}$$

и значение производной Ф’(x_1^r).

5. Если | Ф’(x_1^r)|≤eps, то принимаем x*=x_1^rи заканчиваем вычисления, иначе п.6
6. Если производные Ф’(a^r), Ф’(x_1^r) имеют разные знаки, то $a^{r+1}=a^r$
$b^{r+1}= x_1^r$ r=r+1
и переходим на п.4, иначе п.7
7. Если производные Ф’(x_1^r) и Ф’(b^r) имеют разные знаки, то $a^{r+1}= x_1^r$
$b^{r+1}=b^r$ r=r+1 и переходим на п.4.



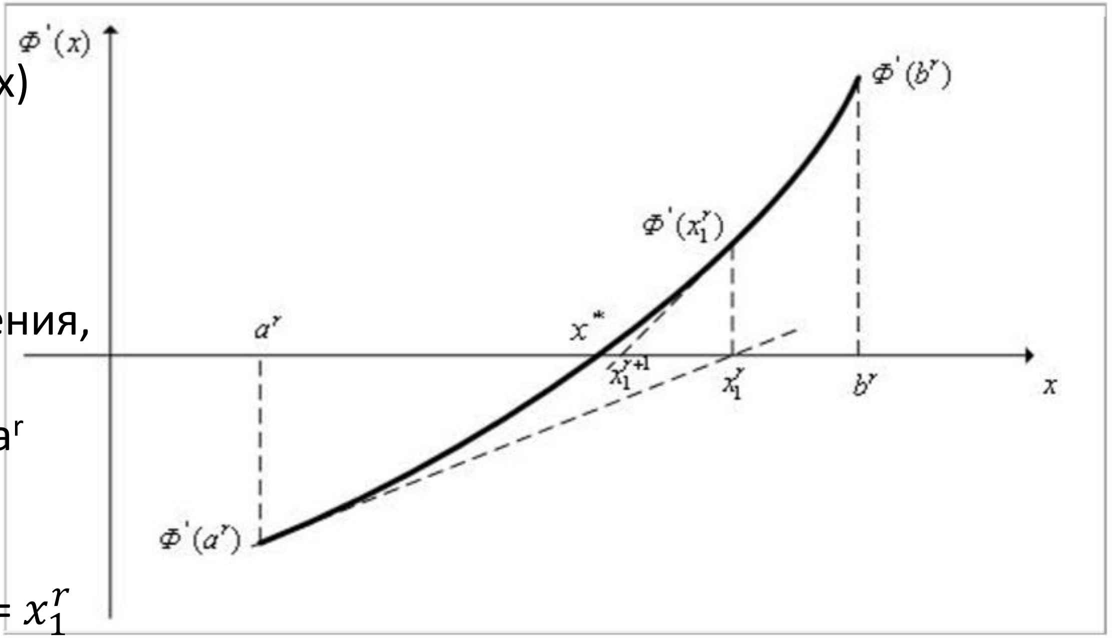
6. Метод касательных (Ньютона)

Найти минимум одномерной унимодальной функции Φ(x) , определенной в замкнутой области допустимых значений D=[a;b]
min*[a;b]Φx=Φ(x*)*
Метод касательных ориентирован на нахождение корня уравнения Φ'(x)=0 в случае, когда на границах интервала знаки производной Φ'(x) различны. Такая ситуация, очевидно, возможна, если точка минимума функции Φ(x) является внутренней точкой интервала [a;b].
Метод требует, чтобы функция Φ(x) была определена и дважды дифференцируема в области допустимых значений D=[a;b].

Алгоритм метода касательных
1. Выполняем присваивания r=1, $a^1=a$, $b^1=b$
2. Вычисляем значения производных Φ'(a^r), Φ'(b^r) имеют одинаковые знаки – завершаем вычисления (точки a и b выбраны неверно), иначе п.4
4. Вычисляем приближение x_1^r к стационарной точке функции Φ(x)

$$x_1^r = a^r - \frac{\Phi'(a^r)}{\Phi''(a^r)}$$

Вычисляем Φ'(x_1^r).
5. Если | Φ'(x_1^r)|≤eps, то принимаем x*=x_1^rи заканчиваем вычисления, иначе п.6
6. Если производные Φ'(a^r), Φ'(x_1^r) имеют разные знаки, то $a^{r+1}=a^r$
$b^{r+1}= x_1^r$ r=r+1
и переходим на п.4, иначе п.7
7. Если производные Φ'(x_1^r) и Φ'(b^r) имеют разные знаки, то $a^{r+1}= x_1^r$
$b^{r+1}=b^r$ r=r+1 и переходим на п.4.



7. Одномерный метод Монте-Карло

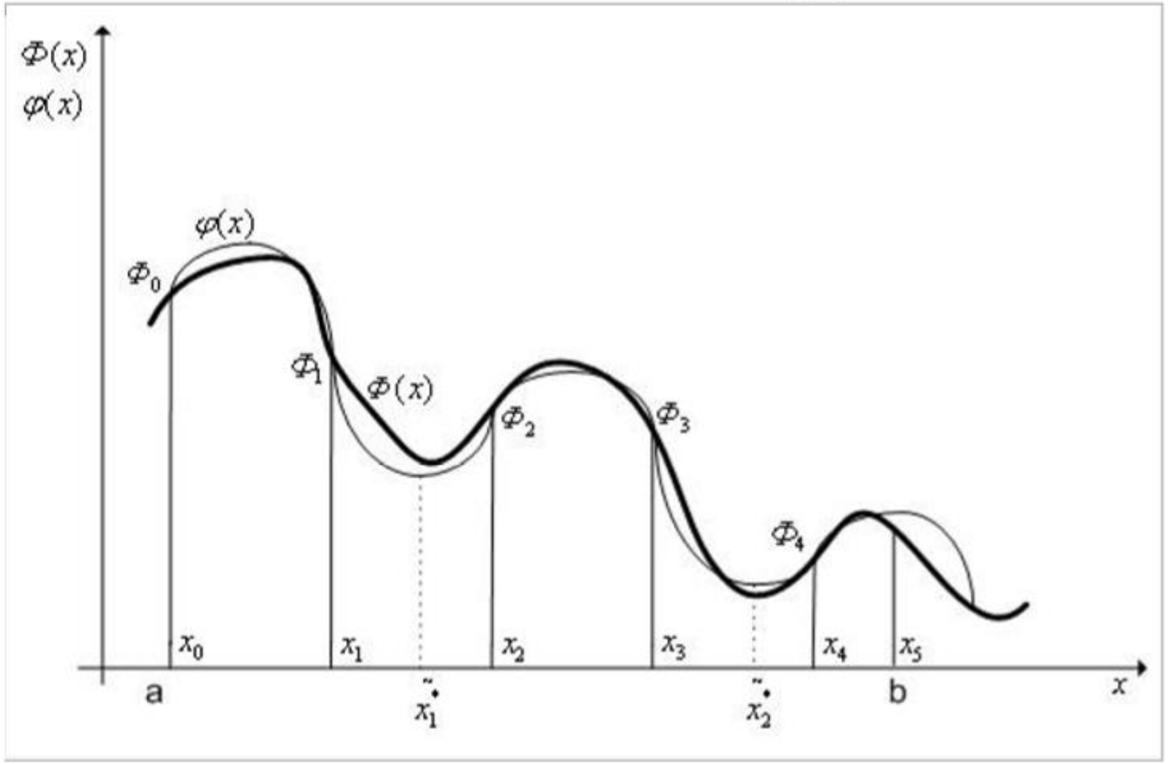
Рассматривается следующая одномерная задача условной **глобальной** оптимизации): найти минимум, вообще говоря, многоэкстремальной функции Ф(x), определенной в замкнутой области допустимых значений

Алгоритм метода Монте-Карло

1. Генерируем с помощью программного генератора случайных чисел, равномерно распределенных в интервале [a, b], случайное число x1 .
2. Вычисляем значения Ф(x1), xmin=x1.
3. Полагаем r=2.
4. Аналогично п. 1 генерируем случайное число xr.
5. Вычисляем значение Ф(xr)
6. Если Ф(xr)< Ф(xmin), то выполняем присваивания xmin=xr.
7. Если r<N, то выполняем присваивание r=r+1 и переходим на п. 4. Иначе – заканчиваем вычисления.
8. Принимаем xmin в качестве приближенного значения точки глобального минимума функции Ф(x) на интервале [a, b]

При достаточно большом N метода гарантирует нахождение глобального минимума с высокой вероятностью.

8. Метод аппроксимирующих моделей



Метод квадратичной аппроксимации

Найти оптимум f(x) на отрезке [a;b]
Требование – унимодальность функции в окрестности поиска экстремума

$$\text{Пусть } f(x) = c_0 + c_1 x + c_2 x^2$$

$$\begin{cases} f(\alpha) = c_0 + c_1\alpha + c_2\alpha^2 \\ f(\beta) = c_0 + c_1\beta + c_2\beta^2 \\ f(\gamma) = c_0 + c_1\gamma + c_2\gamma^2 \end{cases}$$

$$df/d\lambda = c_1 + 2c_2\lambda = 0 \quad \Rightarrow \lambda = -{c_1}/{2c_2} \Rightarrow$$

$$\lambda = -\frac{1}{2}\begin{vmatrix} 1 & f(\alpha) & \alpha^2 \\ 1 & f(\beta) & \beta^2 \\ 1 & f(\gamma) & \gamma^2 \end{vmatrix} \Big/ \begin{vmatrix} 1 & \alpha & f(\alpha) \\ 1 & \beta & f(\beta) \\ 1 & \gamma & f(\gamma) \end{vmatrix} =$$

$$= -\frac{1}{2}\cdot\frac{(\beta^2-\gamma^2)f(\alpha) + (\gamma^2-\alpha^2)f(\beta) + (\alpha^2-\beta^2)f(\gamma)}{(\beta-\gamma)f(\alpha) + (\gamma-\alpha)f(\beta) + (\alpha-\beta)f(\gamma)}$$