СОДЕРЖАНИЕ

[ВВЕДЕНИЕ 2](#_Toc452569136)

[1. Задачей оптимизации 6](#_Toc452569137)

[1.1.Определение унимодальной функции 7](#_Toc452569139)

[1.2.Методы прямого поиска 9](#_Toc452569140)

[1.3.Пассивный метод поиска минимума. 10](#_Toc452569141)

[1.4.Активные методы поиска минимума. 11](#_Toc452569142)

[1.5.Метод дихотомии 11](#_Toc452569143)

[1.6.Метод золотого сечения. 13](#_Toc452569144)

[1.7.Метод Фибоначчи. 17](#_Toc452569145)

[2.Нахождение оптимального значения унимодальной функции. 23](#_Toc452569146)

[2.2. Сравнение эффективности методов. 26](#_Toc452569147)

[ЗАКЛЮЧЕНИЕ 27](#_Toc452569148)

[СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ 28](#_Toc452569149)

#

# ВВЕДЕНИЕ

В настоящее время оптимизация находит применение в науке, технике и в любой другой области человеческой деятельности.

*Оптимизация* - целенаправленная деятельность, заключающаяся в получении наилучших результатов при соответствующих условиях**.**

Поиски оптимальных решений привели к созданию специальных математических методов. Однако до второй половины 20 века методы оптимизации во многих областях науки и техники применялись очень редко, поскольку практическое использование математических методов оптимизации требовало огромной вычислительной работы, которую без ЭВМ реализовать было крайне трудно, а в ряде случаев - невозможно.

Оптимизация как раздел математики существует достаточно давно. Оптимизация - это выбор, т.е. то, чем постоянно приходится заниматься в повседневной жизни. Термином "оптимизация" в литературе обозначают процесс или последовательность операций, позволяющих получить уточненное решение. Хотя конечной целью оптимизации является отыскание наилучшего или "оптимального" решения, обычно приходится довольствоваться улучшением известных решений, а не доведением их до совершенства. Поэтому под оптимизацией понимают скорее стремление к совершенству, которое, возможно, и не будет достигнуто.

На первоначальном этапе решения принимались без специального математического анализа, просто на основе опыта и здравого смысла.

Практика порождает все новые и новые задачи оптимизации причем их сложность растет. Требуются новые математические модели и методы, которые учитывают наличие многих критериев, проводят глобальный поиск оптимума. Другими словами, жизнь заставляет развивать математический аппарат оптимизации.

Из всего выше сказанного можно сделать вывод об актуальности темы курсовой работы.

*Объект исследования:* метод золотого сечения, метод дихотомии и Фибоначчи.

*Предмет исследования:* активные методы оптимизации

*Цель работы:* изучить вопросы отыскания экстремума функции одной переменной, а также выявить наиболее эффективный метод.

*Задачи*, решаемые в работе:

1. Изучить теорию нахождения безусловного экстремумов функции одной переменной;

2. Рассмотреть задачи минимизации функции одной переменной;

3. Изучить численные методы решения задач поиска безусловного минимума функции.

4.Сравнить эффективности методов оптимизации.

В соответствии с целью, задачами и логикой исследования работа состоит из введения, реферата, содержания, 2 разделов, заключения, списка используемой литературы

В первой главе сделан анализ теоретического материала, посвященного пониманию природы задач оптимизации — выведены необходимые и достаточные условия, которым должна удовлетворять функция. Рассмотрены методы золотого сечения, дихотомии и Фибоначчи.

Во второй главе изложены численные методы отыскания безусловного экстремума, рассмотрены алгоритмы и примеры численных методов оптимизации: методом золотого сечения, методом дихотомии, методом Фибоначчи.

В заключении рассмотрены основные итоги курсовой работы, сформулированы перспективные направления дальнейшего развития его темы.

1. Задачей оптимизации в [математике](http://dic.academic.ru/dic.nsf/ruwiki/750) называется задача о нахождении экстремума (минимума или максимума ) вещественной функции в некоторой заданной области. Как правило, рассматриваются области, принадлежащие Rn и заданные набором равенств и неравенств.

## Постановка задачи оптимизации.

Для того,чтобы корректно поставить задачу оптимизации необходимо

задать:

1. *Допустимое множество* — множество

Χ  ;

1. [Целевую функцию](http://dic.academic.ru/dic.nsf/ruwiki/364163) — отображение ;
2. *Критерий поиска* (*max* или *min*).

Тогда решить задачу означает одно из:

1. Показать, что .
2. Показать, что целевая функция ƒ() не ограничена.
3. Найти
4. Если \*, то найти .

Если минимизируемая функция не является выпуклой, то часто ограничиваются поиском локальных минимумов и максимумов: точек x0 таких, что всюду в некоторой их окрестности и ƒ(*x*) ≥ ƒ(*x0*) для минимума и *f*(*x*) ≤ *f* (*x0*) для максимума.

Если допустимое множество Χ = Rn , то такая задача называется задачей безусловной оптимизации, в противном случае – задачей условной оптимизации.

Остановимся на задаче одномерной оптимизации. В этом случае целевая функция есть функция одного действительного переменного, а допустимое множество представляет собой некоторое подмножество числовой оси. Наиболее важным с практической точки зрения является случай, когда допустимое множество есть числовой промежуток, т.е. интервал, полуинтервал или отрезок.

Напомним, что функция, непрерывная на отрезке, достигает на этом отрезке наибольшего и наименьшего значения. Поэтому, задача минимизации в случае, когда допустимое множество есть отрезок, а целевая функция непрерывна на этом отрезке, всегда имеет решение. Задача минимизации функции одного переменного может не иметь решения, если:

* Допустимое множество есть отрезок, но на этом отрезке целевая функция имеет точки разрыва;
* Допустимое множество есть интервал или полуинтервал.

Например, функция ƒ(*х*), (строго) убывающая на полуинтервале [*a,b*), не достигает наименьшего значения на этом полуинтервале, хотя при этом она может быть непрерывной и ограниченной снизу. Та же функция, доопределенная в точке b значением ƒ(*a*), не достигает наименьшего значения и на отрезке [*a,b*].

Функция может достигать наименьшего значения как в единственной точке, так и на некотором множестве точек, конечном, счетном или несчетном. Например, функция *ƒ(x)* = x4, рассматриваемая на отрезке [*-1;1*], достигает наименьшего значения 0 в единственной точке x\* = 0 , в то время как функция

g *(x)* = 1 – x4 на том же отрезке достигает своего наименьшего значения 0 в двух точках: -1 и 1. Функция h *(x)* = *x2 sin2 ,* доопределенная в точке x = 0 значением 0, достигает наименьшего значения 0 в счетном числе точек, а именно в точках *xk = , k* = ± 1, ± 2 , … , и в точке x0 = 0

1.1. Определение унимодальной функции**.**

В теории одномерной оптимизации выделен важный класс функций, которые на отрезке достигают наименьшего значения и притом в единственной точке. Функцию ƒ(*x*) называют унимодальной функцией на отрезке [*a, b*] если существует такая точка x\* ∈ [*a, b*], что функция ƒ(*х*) на отрезке [*а, x\**] убывает, а на отрезке [*x\*, b*] возрастает. Точка x\* может быть внутренней точкой отрезка [*a;b*] (т.е. *a < x\* < b*) или совпадать с одним из его концов. Унимодальная функция может иметь точки разрыва, но таких точек не более чем счетное множество и все они будут точками разрыва I рода. Ясно, что унимодальная функция ƒ(*х*) достигает наименьшего значения в рубежной точке x\*. Примеры унимодальных функций приведены на рис. 1.1

Рисунок 1.1-Примеры унимодальных функций

Свойство унимодальности обеспечивает выполнение очень важного для поиска точки минимума x\* условия.

Пусть ƒ(*x*) унимодальна на ∆, x1 , x2 ∈ ∆, x1 < x2. Тогда, если

 ƒ(*x1*) ≤ ƒ(*x2*),то x\* ≤ x2; если же ƒ(*x1*) ≥ ƒ( *x2*), то x\* ≥ x1.

Таким образом, на основании вычисленных значений ƒ(*x*) можно указать отрезок ∆′ = [*a′, b′*], в котором заключена точка x\*, меньшей длины, чем исходный отрезок ∆ = [*a, b*], т.е. L′ = b′ − a′ < L = b − a. Говорят, что точка минимума x\* локализована в отрезке [*a′, b*′], при этом сам отрезок называют отрезком локализации минимума. На практических занятиях рассматриваются алгоритмы (методы) минимизации унимодальных функций, использующие информацию лишь о значениях функции (алгоритмы нулевого порядка). При записи алгоритмов и решении задач используются следующие обозначения: ∆i = [*ai , bi*] и,

 Li = bi – ai , i = 1 , 2 , … − соответственно отрезок локализации и его длина после *i* вычислений значений ƒ(*x*), ∆0 ≡ [ *a, b*], L0 ≡ b – a , N – количество вычислений значений ƒ(*x*).

Исходными данными для решения задачи минимизации унимодальной функции являются исходный отрезок локализации ∆0 и N, результатами решения − итоговый отрезок локализации ∆N , а также оценки точки минимума x\* и величины минимума ƒ\*

Существуют разные методы решения задач одномерной оптимизации. Если функция дифференцируема на отрезке, то можно, используя необходимые и достаточные условия локального экстремума, определить все точки локального минимума, добавить к ним концы отрезка и простым перебором среди этих точек найти ту, в которой функция достигается наименьшего значения. Однако, на практике нередки ситуации, когда либо функция не является дифференцируемой, либо неизвестно, является ли она дифференцируемой, либо функция дифференцируемая, но трудно вычислить ее производные. В таких ситуациях применяются методы прямого поиска, характеризующиеся тем, что в них не используются производные функции, а вычисляются лишь значения самой функции.

# 1.2. Методы прямого поиска

Можно выделить две группы методов прямого поиска, обусловленные двумя принципиально различными подходами к решению задачи:

1. Методы пассивного поиска – точки, в которых вычисляются значения функции для определения ее наименьшего значения, выбирают заранее, т.е. до вычисления значений функций в этих точках;

Это самый простой метод, предлагающий разделить отрезок [*a, b*] на *k* равных частей точками

xi = , i =

сравнить значения ƒ(*xi*), i = и найти

 xm: ƒ(*xm*) = .

 Ясно, что точка минимума так будет найдена с погрешностью

ε ≤

1. Методы последовательного поиска – точки, в которых вычисляются значения функций, выбирают последовательно исходя из уже выбранных точек и значений функций в них.

# 1.3. Пассивный метод поиска минимума.

 Метод оптимизации называется пассивным, когда все точки *xi, i =*, вычислений характеристик задачи (в данном случае значений целевой функции) выбираются одновременно до начала вычислений. Если N четное, т.е. ,

 *N = 2l, l =1,2*, то наилучшее (в смысле максимального уменьшения длины отрезка локализации) размещение точек *xi, i =*, получается разбиением их на равноотстоящие ε - пары, т.е. ,

X2j-1 = a + x2j = a + , j = (1.1)

где ε − некоторое малое положительное число.

При этом

LN = .

Если N нечетное, т.е., *N = 2l* +1*, l* =1,2, то наилучшим является равномерное распределение точек, т.е. ,

Xi = a + i, i= (1.2)

При этом

LN = 2

Нетрудно заметить, что использование нечетного числа точек при пассивном методе поиска неэффективно.

После определения точек *xi, i =*, вычисляются значения функции *ƒ(xi)*. Пусть *ƒ(xk )* . Тогда, полагая x0 = a, *xN+1* = b, определяется итоговый отрезок локализации *∆N* =[*xk-1, xk+1* ]*.* Точка *xk* принимается за аппроксимацию 37 (оценку) точки минимума x\*, значение функции *ƒ*(*xk*) – за оценку

ƒ\* = ƒ(*x\**), т.е. *x\**= ~ *xk , ƒ\*=* *~ ƒ*(*xk*).

# 1.4. Активные методы поиска минимума.

Метод оптимизации называется активным, если точки *xi , i =*, вычислений характеристик задачи (в данном случае значений целевой функции) выбираются последовательно, с учетом информации, полученной на предыдущих шагах. Для активных (последовательных) методов поиска принято указывать в используемых обозначениях номер итерации с помощью надстрочного индекса в круглых скобках. В соответствии с этим отрезок локализации после j итераций будет обозначаться *∆(j)* = [*a(j), b(j)*] Если при этом произведено i вычислений значений *ƒ(x),* то *∆(j)* *≡ ∆i* , *a(j) ≡ ai , b(j) ≡ bi.* . На практических занятиях рассматриваются такие активные методы поиска, как метод дихотомии, метод Фибоначчи и метод золотого сечения. Для каждого из этих методов на *j* - й, *j* =1, 2 , …, итерации рассматривается пара точек x1(j) и x2(j), при этом x1(j) < x2(j) . Значения функции в этих точках будут обозначаться соответственно ƒ1(j) и ƒ2(j).

1.5. Метод дихотомии**.** В последовательности вложенных отрезков, построенной с помощью процедуры исключения отрезка с симметричным выбором внутренних точек, превышает длина очередного отрезка превышает половину длины предыдущего. Напрашивается такой способ выбора точек, при котором точки *ck* и *dk* на очередном отрезке [*ak ,bk*]расположены максимально близко к середине этого отрезка. Этот способ положен в основу одного из распространенных методов последовательного поиска, называемого методом дихотомии ( или метом деления пополам)

В методе дихотомии в качестве параметра используют достаточно малое число ∆ > 0 ( это число должно быть меньше точности вычисления точки минимума ). На *k*-м шаге вычислений на отрезке [*ak ,bk*]выбираются точки

*сk*= (*ak + bk* ) / 2 - ∆ и *dk*=(*ak + bk* ) / 2 + ∆ отстоящие от середины

(*ak +bk* ) / 2 отрезка [*ak ,bk*]на расстоянии ∆ (рис.1.2)

Рисунок 1.2 **–** Нахождение выбранных точек

Полагая *lk*= *bk* -*ak, k ∈ Z* ,из очевидных геометрических соображений получаем формулу 2*lk+1*- 2∆ = *lk*, *k ∈ Z*, связывающую длины двух последовательных интервалов неопределенности. Из этой формулы находим

*lk+1*- 2∆=( *lk* - 2∆) / 2,откуда

 *lk* - 2∆ = (1.3)

Из равенства (1.3) выводим следующую формулу длины интервала неопределенности, получаемого на k-м шаге метода дихотомии :

*lk* = + 2∆ (1.4)

Из равенства (1.4) следует, что *lk* > 2∆,причем *lk* → 2∆ при k → ∞. Рассматривая величину как удвоенную точность приближенного решения задачи минимизации, полученного на *k*-м шаге ( при выборе в качестве приближения середины интервала неопределенности ), делаем вывод, что параметр ∆ делает оценку снизу погрешности приближения в методе дихотомии. В этой ситуации естественно желание выбирать как можно меньше параметр ∆. Однако на самом деле его нельзя уменьшить неограниченно. Дело в том, что значения функции, как правило, вычисляются приближенно. При очень малом параметре ∆ точка *сk*и *dk*на отрезке [*ak ,bk*]будут расположены настолько близко друг к другу, что не удастся достоверно выяснить, какое из значений функции является большим: различие в значениях функции будет покрываться погрешностью вычислений.

# 1.6. Метод золотого сечения.

Метод дихотомии, как видно из его описания обеспечивает наименьшее количество итераций, необходимых для достижения заданной точности. При этом на каждой итерации метода вычисляются два значения функции (в точках *сk*и *dk*). Однако эффективность метода прямого поиска характеризуется не количеством итераций, а количеством вычисленных значений функции. Существуют методы, в которых на каждой итерации вычисляется только одно значение функций. Количество итераций в них больше, чем в методе дихотомии, но общее количество вычисленных значений функции может быть и меньше.

Можно заметить, что процедура исключения отрезка, примененная к отрезку [*ak , bk* ], дает новый отрезок [ *ak+1, bk+1*] , который содержит внутри себя одну из точек *сk*, *dk*(вторая является одним из концов нового отрезка). В связи с этим возникает идея использовать эту точку повторно, с тем чтобы сократить количество вычисляемых значений целевой функции. В результате возникает следующая схема вычислений.

На начальном отрезке [*a1, b1* ] выбираем точку x1 и симметричную ей точку *y1 = a1 + b1 – x1*. Полагаем *c1 =* и *d1 =* . Проводим процедуру исключения отрезка в соответствии с выбранными точками c1 и d1 и определяем новый отрезок [ *a2, b2*], если будет выбран [*a1, b1*],то внутри него будет располагаться точка *c1*, которую используем в качестве точки *x2* . Если же будет выбран отрезок [ *a1, d1* ],то в качестве *x2* используем точку *d1*. После выбора точки *x2* переходим ко второму шагу вычислений и проводим процедуру исключения отрезка [ *a2, b2* ].

В описанной схеме на каждом шаге необходимо вычислить всего лишь одно значение функции. Но при этом последовательность длин вложенных отрезков устроена сложнее, чем в методе дихотомии. В общем случает трудно объяснить, как быстро эта последовательность стремится к нулю. Кроме того, возможна ситуация, когда на очередном *k*-м шаге точка, сохранения с предыдущего шага, окажется в середине отрезка [ *ak , bk* ]. В такой ситуации не удастся выбрать вторую точку для проведения процедуры исключения отрезка и необходимо выбрать новые точки.

Описанных выше проблем можно избежать, если определенным образом выбрать начальную точку *x1*. Более того, начальную точку *x1* можно выбрать так, что в последовательности {*lk*} длин вложенных отрезков отношение

l*k+1 + 1/ lk* будет оставаться постоянным. Оказывается, что при таком выборе точка *x1* будет осуществлять золотое сечение отрезка [*a1, b1*], т.е. такое его деление на две неравные части, при котором отношение длины всего отрезка к длине его большей части равно отношению длины большей части к длине меньшей.

В самом деле, имеем *lk = b1-a1*. Пусть для определенности точка *x1* расположена правее середины отрезка [ *a1, b1* ], т.е. *d1 = x1*. В результате выполнения первого шага будет выбран новый отрезок [ *a2, b2* ], по длине равный *l2 = x1 - a1* (рис.1.3). Пусть, например, выбран отрезок [ *a2, b2*] = [ *a1, x1*], тогда *x2 = c1 = y1.* Выполнение второго шага будет зависеть от того ,как точка *x2 = y1* расположена по отношению к середине отрезка [ *a1, x1*]. Здесь возможны два случая. В первом случае точка *y1* находится правее середины отрезка [ *a1, x1*], т.е.

 *y1 - a1 > x1 - y1*, и на втором шаге будет выбран отрезок длины *l3 = y1 - a1*. Во втором случае точка *y1* находится левее середины отрезка [ *a1, x1* ], т.е.

*y1 - a1 < x1 - y1*, и на втором шаге будет выбран отрезок длины *l3 = x1 - y1* Рассмотрим каждый из случаев подробнее.

Рисунок 1.3 - Расположение точки *x2 = y1* по отношению к середине отрезка [ *a1, x1*].

В первом случае равенство *l3 / l2 = l2 / l1* принимает вид

Исключив из этого равенства y1 с помощью уравнения

 *x1 + y1 = a1 + b1* , получим

Откуда ,вводя обозначение *r =* , приходим к уравнению

Соответствующему золотому сечению отрезка длины *l* ( величина *r* составляет большую часть этого сечения, а величина *l - r* – меньшую часть ). Уравнение (1.5) сводится к квадратному уравнению *r2 +r - l = 0*, имеющему единственное положительное решение

*r = ≈0,618034*

Это решение мы будем называть коэффициентом золотого сечения. Отметим, что если *x1* осуществляет золотое сечение отрезка [*a1, b1*], т.е*.*

*x1 - a1 = r ( b1 – a1 )*, то

*x1 - a1 <( b1 - a1 ), y1 - a1 = b1 - a1 - x1 - a1 > ( b1 - a1 )*

*и*

*x1 - y1 = 2 (x1 – a1 ) - (b1 – a1) < (b1 –a1)< y1 – a1*,

т.е. золотое сечение соответствует первому случаю при выполнении второго шага.

Во втором случае равенство *l3/ l2 = l2/ ll*принимает вид

Что после исключения *y1* приводит к равенству

Или

 = r

Последнее уравнение имеет единственное решение *r* =1, которое не может быть использовано для выбора точки *x1*: необходимо значение r из интервала ( *0,1*).

Итак, если точка *x1* осуществляет золотое сечение начального отрезка [*a1,b1* ], то одна из точек *x1* и *y1* окажется внутри следующего отрезка [*a2,b2*],

причем эта точка (точка *x2*) будет осуществлять золотое сечение отрезка [*a2, b2*]. Повторяя рассуждения, заключаем, что на каждом k-м шаге точка *xk* будет осуществлять золотое сечение отрезка [*ak , bk*].

Метод последовательного поиска, в котором на *k*-м шаге каждая из точек осуществляет золотое сечение отрезка [*ak , bk* ],называют методом золотого сечения. В этом методе отношение очередного интервала неопределенности к предыдущему равно *r =* . Для длин *lk* интервалов неопределенности выполняются соотношения *lk = rk l1.*

На каждом шаге метода, кроме первого, вычисляется одно значение функции ( на первом шаге требуется вычислить два значения функции в точках x1 и y1)

# 1.7. Метод Фибоначчи.

#### Предположим, что нужно определить минимум как можно точнее, т.е. с наименьшим возможным интервалом неопределенности, но при этом можно выполнить только n вычислений функции

Предположим, что имеется интервал неопределенности (*x1, x3*) и известно значение функции *f* (*x2*) внутри этого интервала (см. рис. 1.4). Если можно вычислить функцию всего один раз в точке х4, то где следует поместить точку х4, для того чтобы получить наименьший возможный интервал неопределенности?


Рисунок 1.4 **–** Графическое представление функции

Положим х2 – х1 = L и х3 – х2 = R, причем L > R, как показано на рис. 1.4, и эти значения будут фиксированы, если известны x1, x2 и х3. Если х4 находится в интервале ( *х1 ; х2*), то:

1. если f (*x4*) < f (*x2*), то новым интервалом неопределенности будет (*x1 , x2*) длиной х2 – х1 = L ;
2. если f (*х4*) > f (*x2*), то новым интервалом неопределенности будет (*х4, х3*) длиной х3 – х4.

Поскольку не известно, какая из этих ситуаций будет иметь место, выберем х4 таким образом, чтобы минимизировать наибольшую из длин х3 - х4 и

 х2 - х1. Достигнуть этого можно, сделав длины х3 – х4 и х2 – х1 равными т.е. поместив х4 внутри интервала симметрично относительно точки х2, уже лежащей внутри интервала. Любое другое положение точки х4 может привести к тому, что полученный интервал будет больше L. Помещая х4 симметрично относительно х2, мы ничем не рискуем в любом случае. Если окажется, что можно выполнить еще одно вычисление функции, то следует применить описанную процедуру к интервалу (*х1, х2*), в котором уже есть значение функции, вычисленное в точке х4, или к интервалу (*х4, х3*), в котором уже есть значение функции, вычисленное в точке х2.

Следовательно, стратегия ясна с самого начала. Нужно поместить следующую точку внутри интервала неопределенности симметрично относительно уже находящейся там точке. Парадоксально, но, чтобы понять, как следует начинать вычисления, необходимо разобраться в том, как его следует кончать.

На n - м вычислении n - ю точку следует поместить симметрично по отношению к (*n — 1*) - й точке. Положение этой последней точки в принципе зависит от нас. Для того чтобы получить наибольшее уменьшение интервала на данном этапе, следует разделить пополам предыдущий интервал. Тогда точка х будет совпадать с точкой хn-1. Однако при этом мы не получаем никакой новой информации. Обычно точки хn-1 и хn отстоят друг от друга на достаточном расстоянии, чтобы определить, в какой половине, левой или правой, находится интервал неопределенности. Они помещаются на расстоянии е/ 2 по обе стороны от середины отрезка Ln-1; можно самим задать величину е или выбрать эту величину равной минимально возможному расстоянию между двумя точками.

Интервал неопределенности будет иметь длину Ln, следовательно,

Ln-1 = 2Ln - е (рис.1.5, нижняя часть). На предыдущем этапе точки хn-1 и

хn-2 должны быть помещены симметрично внутри интервала Ln-2 на расстоянии

Ln-2 от концов этого интервала. Следовательно, Ln-2 = Ln-1+Ln ([рис.1.5](http://www.intuit.ru/studies/courses/1020/188/lecture/4929?page=2#image.9.4),средняя часть).


Рисунок 1.5 **–** Графическое представление интервалов неопределенности

**Замечание**. Из рисунка ясно, что на предпоследнем этапе хn-2 остается в качестве внутренней точки.

Аналогично *Ln-3 = Ln-2 + Ln-1* ([pис. 1.5](http://www.intuit.ru/studies/courses/1020/188/lecture/4929?page=2" \l "image.9.4), верхняя часть)

В общем случае *Lj-1 = Lj + Lj+1* при *1 < j < n.*

Таким образом,

*Ln-1 = 2Ln -ε*

*Ln-2 = Ln-1 + Ln = 3Ln - ε*

*Ln-3 = Ln-2 + Ln-1 = 5Ln - 2ε*

*Ln-4 = Ln-3 + Ln-2 = 8Ln -3ε*  и т.д.

Если определить последовательность чисел Фибоначчи следующим образом: F0 = 1, F1 = l, и Fk = Fk-1 + Fk-2 для k = 2 , 3, … , то

|  |  |
| --- | --- |
|  | *Ln-j = Fj+1 Ln – Fj-1 ε , j = 1,2,…,n-1*.  |

Если начальный интервал (*a; b*) имеет длину L = (*b - а*), то

*L1 = Fn Ln – ε Fn-2 , т.е.*

*Ln =*  (1.6)

Следовательно, произведя n вычислений функции, мы уменьшим начальный интервал неопределенности в l / Fn раз по сравнению с его начальной длиной (пренебрегая е), и это - наилучший результат.

Если поиск начат, то его несложно продолжить, используя описанное выше правило симметрии. Следовательно, необходимо найти положение первой точки, которая помещается на расстоянии L2 от одного из концов начального интервала, причем не важно, от какого конца, поскольку вторая точка помещается согласно правилу симметрии на расстоянии L2 от второго конца интервала:

L2 = Fn-1 · Ln – ε ·Fn-3 = Fn -1 = (1.7)

|  |  |
| --- | --- |
|  |  |

После того как найдено положение первой точки, числа Фибоначчи больше не нужны. Используемое значение е может определяться из практических соображений. Оно должно быть меньше L1\ Fn+x, в противном случае мы будем напрасно тратить время на вычисление функции.

Таким образом, поиск методом Фибоначчи, названный так ввиду появления при поиске чисел Фибоначчи, является итерационной процедурой. В процессе поиска интервала (*x1; x2*) с точкой х2, уже лежащей в этом интервале, следующая точка х2 всегда выбирается такой, что х3 – х4 = х2 – х1 или

 х4 - х1 = х3 - x2, т.е. x4 = х1 - х2 + х3.

Если *f* (*x2*) = *f*2 и *f* (*x4*) = *f*4, то можно рассмотреть четыре случая ([рис.1.6](http://www.intuit.ru/studies/courses/1020/188/lecture/4929?page=2#image.9.5)).

Рисунок 1.6а - х4 < x2 ; *f*4 < *f*2 . Новый интервал (*х1, х2*),содержащий точку х4.

Рисунок 1.6б - х4 > x2 ; *f*4 < *f*2. Новый интервал (*х2,х3*), содержащий точку х4.

Рисунок 1.6в - х4 < x2; *f*4 > *f*2 . Новый интервал (*х4,х3*), содержащий точку х2.



Рисунок 1.6г - х4 > x2; *f*4 > *f*2 . Новый интервал (*х1, х4*), содержащий точку х2.

## 2.  Нахождение оптимального значения унимодальной функции.

*Задание.* Определить методом дихотомии минимум функции *ƒ(x) = x4 - 6x2 + 10*, заданной на отрезке ∆=[*1;3*], при N = 8, ε = 0,1.

*Решение*. В данном случае будут выполнены N/ 2 = 4 итерации.

Результаты вычислений заносим в таблицу 2.1

Таблица 2.1- Результаты вычислений методом дихотомии

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Номер итерации | X1(j) | X2(j) | ƒ1(j) | ≤> | ƒ2(j) | a(j) | b(j) |
| 0 |  --- |  --- |  --- |  |  --- |  1 |  3 |
| 1 | 1,95 | 2,05 | 1,644 | < | 2,446 |  1 | 2,05 |
| 2 | 1,475 | 1,575 | 1,680 | > | 1,270 | 1,475 | 2,05 |
| 3 | 1,713 | 1,813 | 1,004 | < | 1,082 | 1,475 | 1,813 |
| 4 | 1,594 | 1,694 | 1,211 | > | 1,217 | 1,594 | 1,813 |

Поскольку j = N/ 2 = 4,то вычисления завершаются.

Точка минимума локализована на отрезке ∆8 = [1,594;1,813].На данном отрезке исследованы 4 точки:

x\* **≅**x1(3) =1,713, ƒ\***≅**ƒ(x1(3))=1,004

*Ответ*: ∆8 = [1,594;1,813], x\* **≅** 1,713, ƒ\***≅** 1,004.

*Задание*. Определить методом Фибоначчи минимум функции

*ƒ(x)=x4 - 6x2 + 10*, заданной на отрезке ∆=[*1;3*], при N = 4.

*Решение .* В данном случае будут выполнены N – 1 = 3 итерации.

Определим числа Фибоначчи Fk , k = :

F0 = F1 = 1, F2 = 2 ,F3 = 3, F4 = 5, F5 = 8

ε <

Выбираем ε = 0,1.

*Первая итерация*.

*Вторая итерация*.

.

*Третья итерация.*

.

Результаты вычислений заносим в таблицу 2.2

Таблица 2.2 - Результаты вычислений методом Фибоначчи

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Номер итерации |  X1(j) |  X2(j) |  ƒ1(j) |  ≤ > |  ƒ2(j) |  a(j) |  b(j) |
|  0 |  --- |  --- |  --- |  |  --- | 1 | 3 |
|  1 |  1,78\* | 2,22\* | 1,028 | < | 4,719 | 1 | 2,22 |
|  2 |  1,44\* | 1,78 | 1,858 | > | 1,028 | 1,44 | 2,22 |
|  3 |  1,78 | 1,88\* | 1,028 | < | 1,286 | 1,44 | 1,88 |

Примечание. Знаком \* помечаем точки Xi(j), i = 1,2,вычисляемые на j-й итерации.

Поскольку j = N – 1 = 3 ,то вычисления завершаются.

Точка минимума локализована на отрезке ∆4 = [1,44;1,88], x\* **≅**x1(3) =1,78, ƒ\***≅**ƒ(x1(3))=1,028.

*Ответ*: ∆4 = [1,44;1,88], x\* **≅** 1,78, ƒ\***≅** 1,028.

*Задание*. Определить методом золотого сечения минимум функции

*ƒ(x) = x4 - 6x2 + 10*, заданной на отрезке ∆ = [*1;3*] , при N = 4.

*Решение*. В данном случае будут выполнены N – 1 = 3 итерации.

Результаты вычислений заносим в таблицу 2.3

Таблица 2.3 - Результаты вычислений методом золотого сечения

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Номер итерации | X1(j) | X2(j) | ƒ1(j) | ≤> | ƒ2(j) |  a(j) | b(j) |
|  0 |  --- |  --- |  --- |  |  --- |  1 |  3 |
|  1 | 1,764\* | 2,236\* | 1,012 |  < | 4,999 |  1 | 2,236 |
|  2 | 1,472\* | 1,764 | 1,694 |  > | 1,012 | 1,472 | 2,236 |
|  3 | 1,764 | 1,944\* | 1,012 |  < | 1,607 | 1,472 | 1,944 |

Поскольку j = N – 1 = 3,то вычисления завершаются.

Точка минимума локализована на отрезке ∆4 = [1,472;1,944], x\***≅**x1(3) =1,764, ƒ\***≅**ƒ(x1(3))=1,012.

*Ответ*: ∆4=[1,472;1,944], x\* **≅** 1,764, ƒ\***≅** 1,012.

# 2.2. Сравнение эффективности методов.

При сравнении прямых методов минимизации обычно учитывают количество **N** значений *f* (*x*), гарантирующее заданную точность определения точки *х\** тем или иным методом. Чем меньше **N**, тем эффективнее считается метод. При этом вспомогательные операции такие, как выбор пробных точек, сравнение значений *f* (*x*) и т.п., не учитываются. Во многих практических случаях определение значений целевой функции требует больших затрат (например, времени ЭВМ или средств для проведения экспериментов) и вспомогательными вычислениями можно пренебречь. А эффективность метода минимизации особенно важна именно в таких случаях, поскольку позволяет сократить указанные затраты.

Эффективность методов минимизации можно также сравнивать, на основании гарантированной точности ε (***N***) нахождения точки *х\**, которую они обеспечивают в результате определения **N** значений *f* (*x*). Метод золотого сечения считают более точным, т.к. он позволяет исключать интервалы, вычисляя только одно значение функции на каждой итерации , в отличии от метода дихотомии и Фибоначчи.

Проверить точность эффективность методов и точность можно с помощью метода производных. Так x\* **≅** 1,737. Сопоставляя полученные x\* , можно сделать вывод о том, что действительно метод золотого сечения является наиболее точным и удобным для использования, нежели другие методы, однако разница в точности в данном случае незначительна.

# ЗАКЛЮЧЕНИЕ

В курсовой работе была произведена минимизация функций с помощью активных методов поиска – методом золотого сечения, дихотомии и методом Фибоначчи. В соответствии с поставленной целью в ходе работы было выяснено, что метод Фибоначчи является наилучшим (в смысле максимального уменьшения длины отрезка локализации) среди активных методов поиска, но недостатком метода Фибоначчи является то, что должно быть задано количество вычислений N. Дихотомическое деление привлекательно своей простотой, а метод золотого сечения почти столь же эффективен, как и метод Фибоначчи, но при этом не зависит от N.

 Анализируя полученные результаты, можно сделать вывод о том, что по числу итераций более эффективным является метод золотого сечения. Однако следует отметить, что эффективность этих методов может изменяться в зависимости от выбора начального приближения и вида функции

Оптимизация в широком смысле слова находит применение в науке, технике и в любой другой области человеческой деятельности. Указанные методы используются при решении нелинейных задач с ограничениями и линейных целочисленных задач.

# СПИСОК ИСПОЛЬЗУЕМЫХ ИСТОЧНИКОВ

1. Методы оптимизации: Учебное пособие/ Б.Ф.Харчистов: Изд-во ТРТУ, 2004. − 140с.

# Введение в методы оптимизации: Учебное пособие /Аттетков А.В., Зарубин В.С., Канатников А.Н.: Инфра-М. 2008. — 272 с

1. Основы автоматизированного проектирования : учеб. для вузов / И.П.Норенков – 4 - е изд.перераб. и доп.- М.: Изд-во МГТУ им. Н.Э.Баумана,2009.-430, [2] с.: ил. - («Информатика в техническом университете»)
2. Численные методы оптимизации: учебное пособие /В.И. Рейлин; Томский политехнический университет .- Томск: Изд - во Томского политехнического университета,2011-105с.