|  |  |
| --- | --- |
| Gerb-BMSTU_01 | **Министерство науки и высшего образования Российской Федерации**  **Федеральное государственное бюджетное образовательное учреждение**  **высшего образования**  **«Московский государственный технический университет**  **имени Н.Э. Баумана (национальный исследовательский университет)»**  **(МГТУ им. Н.Э. Баумана)** |

|  |  |
| --- | --- |
| ФАКУЛЬТЕТ | *Робототехники и комплексной автоматизации* |
|  |  |
| КАФЕДРА | *Системы автоматизированного проектирования (РК-6)* |

**РАСЧЕТНО-ПОЯСНИТЕЛЬНАЯ ЗАПИСКА**

**К НАУЧНО-ИССЛЕДОВАТЕЛЬСКОЙ РАБОТЕ СТУДЕНТА**

|  |  |  |
| --- | --- | --- |
| Студент |  | Жарова Татьяна Александровна |
| Группа |  | РК6-45Б |
| Тема |  |  |

Студент **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Жарова Т.А.**

*подпись, дата фамилия, и.о.*

Преподаватель **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Волосатова Т.М.**

*подпись, дата фамилия, и.о.*

Консультант **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Дорофеев В.С.**

*подпись, дата фамилия, и.о.*

*Москва, 2022 г.*

**СОДЕРЖАНИЕ**

[ВВЕДЕНИЕ 3](#_Toc104919996)

[ЦЕЛЬ И ЗАДАЧИ РАБОТЫ 4](#_Toc104919997)

[1. МЕТОДЫ ПАРАМЕТРИЧЕСКОЙ ОПТИМИЗАЦИИ 6](#_Toc104919998)

[1.1. Метод сеточного поиска: 6](#_Toc104919999)

[1.2. Метод координатного спуска: 9](#_Toc104920000)

[1.3. Метод золотого сечения: 14](#_Toc104920001)

[СРАВНЕНИЕ МЕТОДОВ 16](#_Toc104920002)

[ПРИМЕНЕНИЕ 16](#_Toc104920003)

[СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ 17](#_Toc104920004)

# **ВВЕДЕНИЕ**

Параметр определяется как числовой или иной измеримый фактор, образующий один из множества, определяющих систему или наборы условий ее функционирования. Параметры - это элементы системы, которые полезны или критичны для идентификации системы или оценки ее производительности, статуса или состояния. Например, параметры уравнения, моделирующего движение, могут включать механику, массы, размеры и формы, а также плотности и вязкости жидкостей в системе.

Оптимизацией называется максимизация или минимизация некоторой функции относительно некоторого набора, часто представляющего диапазон вариантов, доступных в определенной ситуации. Функция позволяет сравнивать различные варианты для определения того, какой из них может быть “лучшим”. [1]

Параметрическая оптимизация рассматривает проблемы оптимизации в зависимости от параметра и описывает, как допустимый набор, функция значений и локальные или глобальные минимизаторы программы зависят от изменений параметра.

Задача параметрической оптимизации заключается в нахождении экстремумов функции на множествах конечномерного векторного пространства, определяемого линейными и нелинейными ограничениями (равенствами и неравенствами). Результатом решения параметрической оптимизацией являются наилучшие структура и значения параметров системы. [2]

По числу оптимизируемых параметров задача оптимизации может быть многопараметрической и однопараметрической, по числу критериев – многокритериальные и однокритериальные.

Чтобы включить противоречивый характер задачи оптимизации в удобный инструмент математического анализа, требуется четкая постановка задачи, которая традиционно состоит из трех компонентов:

1. Целевая функция определяет оптимальное решение, при этом существует три вида назначения целевой функции: максимизация, минимизация или постоянный предел (заданное значение назначения).
2. Ограничения устанавливают неизменяемые параметры управления отношениями и переменные параметры, а из этих зависимостей следуют пределы изменяемых параметров.
3. Граничные условия указывают, в какой степени может быть значение параметров.[2][3]

Алгоритм решения задачи параметрической оптимизацией:

1. Сформулировать цель оптимизации;
2. Перейти от словесного описания задачи к математической модели (выбор целевой функции и ограничений);
3. Нормировать выходные параметры;
4. Выбрать эффективный поисковой метод на основании анализа конкретных особенностей функции цели и ограничений;
5. Рассчитать на ЭВМ.[4]

Проектирование систем традиционно включает в себя определение архитектуры, спецификацию интерфейсов, требования к производительности и т.д.

Стратегия поиска – это алгоритм выбора следующего варианта значений параметров.

# **ЦЕЛЬ И ЗАДАЧИ РАБОТЫ**

***Цель данной работы:***

Изучить методы параметрической оптимизации и проверить 2 из них на результативность.

***Задачи данной работы:***

* Анализ литературы о реализации и практическом применении методов параметрической оптимизации;
* Обобщение полученной информации о предметной области, оформленное как теоретическая часть данной работы, и включающее в себя:
* Описание алгоритмов методов параметрической оптимизации;
* Практические примеры применения;
* Достоинства и недостатки;
* Практическая реализация двух методов параметрической оптимизации;
* Составление вывода на основе выполнения работы

# **МЕТОДЫ ПАРАМЕТРИЧЕСКОЙ ОПТИМИЗАЦИИ**

1. Включить противоречивый характерзадачи оптимизации в математический анализ удобный инструмент,
2. это требует четкой постановки задачи, которая традиционно состоитиз трех составляющих:
3. 1. Целевая функция определяет оптимальное решение, при этом существуеттри вида
4. назначение целевойфункции: максимизация, минимизация или постоянный предел
5. (задание уставки).
6. 2. Ограничения, устанавливаемыемеждупостоянными параметрами и переменной
7. параметры, а также зависимости этих параметров в пределах варьируемых параметров.
8. 3. Граничные условия указывают коэффициент, к которому может относиться величина оптимизации
9. параметры
10. Включить противоречивый характерзадачи оптимизации в математический анализ удобный инструмент,
11. это требует четкой постановки задачи, которая традиционно состоитиз трех составляющих:
12. 1. Целевая функция определяет оптимальное решение, при этом существуеттри вида
13. назначение целевойфункции: максимизация, минимизация или постоянный предел
14. (задание уставки).
15. 2. Ограничения, устанавливаемыемеждупостоянными параметрами и переменной
16. параметры, а также зависимости этих параметров в пределах варьируемых параметров.
17. 3. Граничные условия указывают коэффициент, к которому может относиться величина оптимизации
18. параметры
19. Включить противоречивый характерзадачи оптимизации в математический анализ удобный инструмент,
20. это требует четкой постановки задачи, которая традиционно состоитиз трех составляющих:
21. 1. Целевая функция определяет оптимальное решение, при этом существуеттри вида
22. назначение целевойфункции: максимизация, минимизация или постоянный предел
23. (задание уставки).
24. 2. Ограничения, устанавливаемыемеждупостоянными параметрами и переменной
25. параметры, а также зависимости этих параметров в пределах варьируемых параметров.
26. 3. Граничные условия указывают коэффициент, к которому может относиться величина оптимизации
27. параметры
28. Включить противоречивый характерзадачи оптимизации в математический анализ удобный инструмент,
29. это требует четкой постановки задачи, которая традиционно состоитиз трех составляющих:
30. 1. Целевая функция определяет оптимальное решение, при этом существуеттри вида
31. назначение целевойфункции: максимизация, минимизация или постоянный предел
32. (задание уставки).
33. 2. Ограничения, устанавливаемыемеждупостоянными параметрами и переменной
34. параметры, а также зависимости этих параметров в пределах варьируемых параметров.
35. 3. Граничные условия указывают коэффициент, к которому может относиться величина оптимизации
36. параметры
37. Включить противоречивый характерзадачи оптимизации в математический анализ удобный инструмент,
38. это требует четкой постановки задачи, которая традиционно состоитиз трех составляющих:
39. 1. Целевая функция определяет оптимальное решение, при этом существуеттри вида
40. назначение целевойфункции: максимизация, минимизация или постоянный предел
41. (задание уставки).
42. 2. Ограничения, устанавливаемыемеждупостоянными параметрами и переменной
43. параметры, а также зависимости этих параметров в пределах варьируемых параметров.
44. 3. Граничные условия указывают коэффициент, к которому может относиться величина оптимизации
45. параметры

Любую задачу о минимизации можно заменить эквивалентной ей задачей максимизации: следует лишь заменить знак всех коэффициентов функции цели, при этом сохранив неизменными ограничения. Соответственно можно сформулировать задачу оптимизации следующим образом: среди элементов *x*, образующих множество *Χ*, найти , что . Корректная постановка задачи оптимизации предполагает задание:

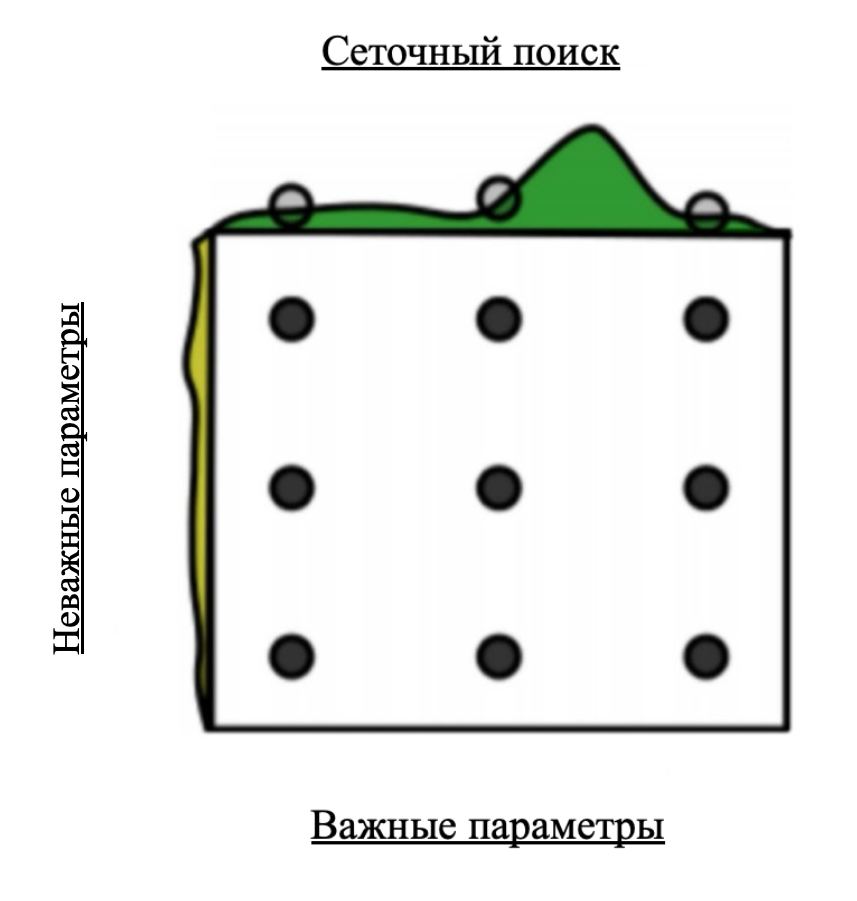
* допустимое множество ;
* целевая функция, т.е. отображенная ;
* критерий поиска . [3]

## **Метод сеточного поиска:**

Поиск по сетке — это метод настройки, который вычисляет оптимальные значения параметров. Это исчерпывающий поиск, который выполняется по конкретным значениям параметров оценочной модели.

Поиск по сетке считается традиционным методом оптимизации параметров, поскольку мы в основном «перебираем» все возможные комбинации. Естественно, лучшим считается прототип с наибольшей точностью.

Поиск по сетке отлично подходит для выборочной проверки комбинаций, которые, как известно, в целом хорошо работают. [5][6][7]



*Рис 1.* Компоновка сетки.

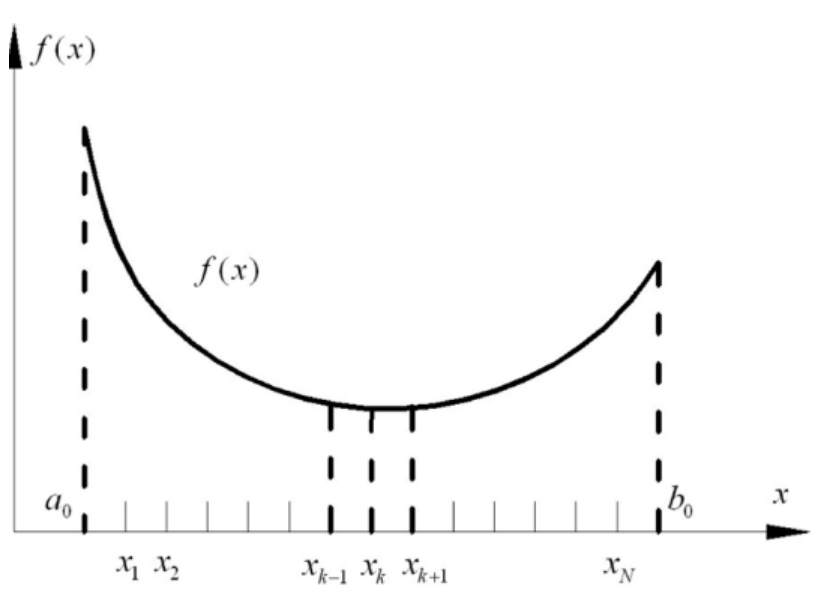
**Алгоритм сеточного поиска для одномерного пространства:**

Суть метода заключается в том, что в разбиении интервала поиска [a,b] на q равных частей, количество которых определяется по формуле:

где – допустимая погрешность вычислений точки экстремума.

В процессе оптимизации рассматриваем множество точек, численное значение которых вычисляется по формуле:

Вычисляя значение нашей целевой функции и проводя сравнение результатов, находим точку , в которой функция цели имеет экстремальное значение.



*Рис 2. График функции f(x)*

Алгоритм вычислений по методу сеточного поиска следующий:

1. Задаем
2. Определяем число итераций k (циклов вычисления):
3. Вычисляем значения переменной в пробных точках:
4. Находим значения функции цели в пробных точках ;
5. Сравнивая результаты функции в пробных точках, определяем минимальное (максимальное) значение нашей функции. [8]

**Преимущества сеточного поиска:**

* Применяется метод как для многомерной, так и одномерной оптимизации;
* Очень простой для реализации;
* Не требуется предварительно выделять унимодальные области оптимизируемой функции. [9]

***Недостатки сеточного поиска:***

* Результаты предыдущих итераций в последовательности точек сетки не используются для информирования будущих итераций поиска, что и увеличивает время поиска.
* Слишком много вычислений производится в данном методе, поэтому он эффективен для малого количества комбинаций.
* Истинные глобальные минимумы могут быть скрыты между точками сетки, т.е. мы можем просто пропустить искомое значение из-за того, что не рассматривали узел сетки. [10]

## **Метод координатного спуска:**

Координатный спуск - это еще один тип алгоритма оптимизации, применяемый для многомерной оптимизации. Сведение многомерной задачи к последовательным одномерным, которые решаются простыми методами является главной задачей метода координатного спуска.[11]

В циклическом спуске по координатам человек начинает с начальных значений, проходит по направлениям, одновременно минимизирует функцию цели для каждого направления координат. [12]

Суть метода заключается в том, что мы выбираем один параметр и фиксируем все остальные, присваивая им какие-либо значения. После этого проводится оптимизация только для этой переменной и определяется направление, в котором осуществляется оптимизация целевой функции. Пока происходит улучшение (уменьшение или увеличение (в зависимости от задачи)), движение в этом направлении продолжается. После этого выполняем аналогичные действия с другими параметрами. Когда движение (полный цикл) для всех переменных завершено, выполняется новая оптимизация пробного цикла для всех переменных последовательно.

Поиск заканчивается если улучшение решения невозможно. В этом случае полученная многомерная точка принимается за экстремум. [13]

Выбираем одну из переменных , по которой будет осуществляться минимизация (максимизация). При этом на первом этапе оптимизации всем остальным параметрам должны быть присвоены некоторые фиксированные значения (фиксированные значения обычно определяются как начальная пробная точка). После того как переменные заданы постоянными значениями, эта функция преобразуется в функцию с одной переменной:

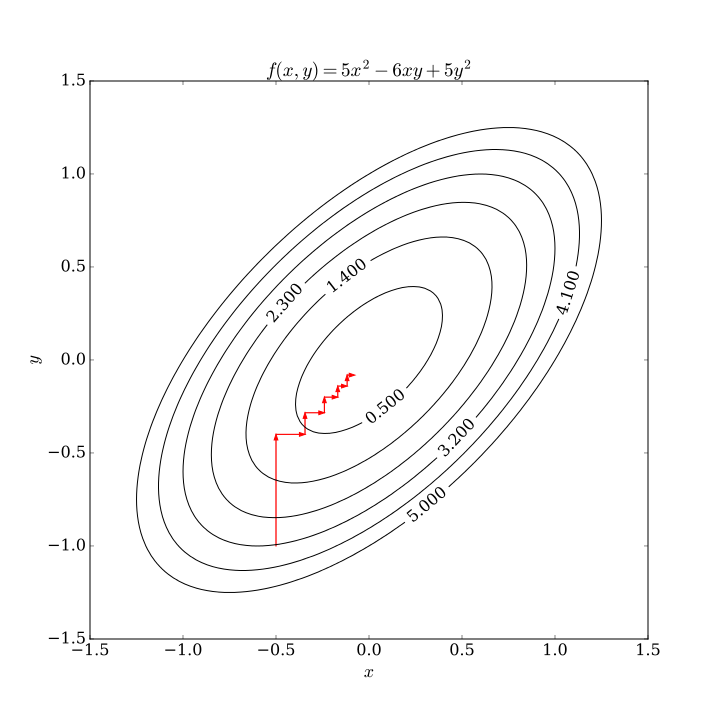
- это погрешность определения местоположения экстремума.

***Алгоритм координатного спуска:***

1. Задаем .
2. Выбираем одну из переменных , по которой мы будем осуществлять минимизация (максимизация).
3. Всем остальным параметрам должны быть присвоены некоторые зафиксированные значения (фиксированные значения обычно определяются как изначальная пробная точка). После того как переменные заданы постоянными значениями, эта функция преобразуется в функцию с одной переменной:
4. Задаем пробные точки переменной с некоторым шагом.
5. Определяем направление изменения значений по значениям нашей функции в пробных точках, которые обеспечивают уменьшение (увеличение).
6. Принимаем за опорную точку значение , при котором функция цели имеет экстремальное значение. В окрестностях этой точки будет осуществляться дальнейший поиск
7. Если такого направления не существует перейти к пункту 2, закончить поиск и перейти к пункту 8, если ни по одной из переменных нет такого направления.
8. Изменяем значение переменной в при заданной величине погрешности в окрестностях опорной точки, найденной по пункту 2 до тех пор, пока происходит уменьшение (увеличение) нашей функции.
9. Минимальное (максимальное) значение функции фиксируем в переменной и переходим к пункту 2.
10. После того как найдете фиксированные значения всех переменных, попробуйте улучшить решение, выполнив заново пункты 2-10 еще раз. В это время, присваивая другим переменным значения, которые вы нашли в процессе решения. Если решение не может быть улучшено, то перейдите к пункту 11.
11. Ищем значение функции в точке с последними фиксированными значениями всех переменных и принимаем это значение за точку экстремума. [14]

Если мы визуализируем контурный график координатного спуска, мы можем увидеть траекторию лестницы, поскольку мы обновляем только одну координату.

Этот процесс проиллюстрирован ниже:



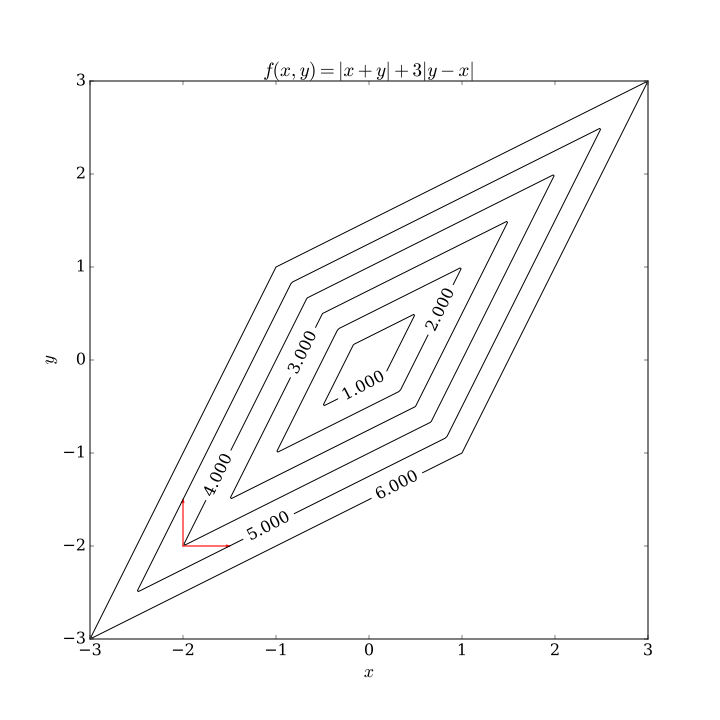
*Рис 3. Сходимость спуска по координатам*

***Преимущества координатного спуска:***

* + - Минимизирует одну координату (т.е.) сразу, сохраняя фиксированными другие. Следовательно, решение становится намного проще.
    - Может использоваться (в большинстве случаев), даже если для функции цель/стоимость нет близкого решения. [15]

***Недостатки спуска по координатам:***

Есть два недостатка у координатного спуска. Одним из них является негладкая функция многих переменных. Как показано на рисунке 4, кривые уровня функции негладкие. В таком случае итерация координатного спуска может застрять в нестационарной точке. Допустим, что координатами начальной точки являются (-2, -2); тогда есть два направления, выровненных по осям, которые алгоритм может учитывать для выполнения шага (направления обозначены на рисунке красными линиями). Однако в этих двух направлениях каждый шаг будет увеличивать (при условии задачи минимизации) значение функции цели поэтому алгоритм не выполнит ни одного шага, даже если оба шага вместе приблизят к оптимуму. Хотя данный пример показывает, что координатный спуск необязательно должен сходиться к оптимуму, формальную сходимость можно показать при разумных условиях.



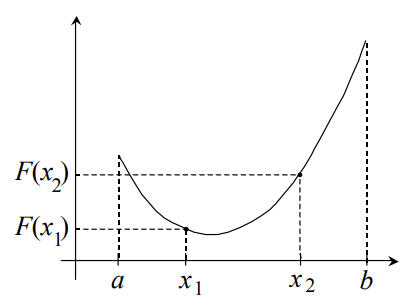
*Рис 4. Расходимость спуска по координатам*

Другой недостаток - это сложность в распараллеливании. Так как природа координатного спуска заключается в прохождении направлений циклично и минимизации (максимизации) функции цели по отношению к каждому координатному направлению, данный метод не является явным кандидатом для распараллеливания. Недавние исследовательские работы показали, что распараллеливание может быть применен к спуску по координатам за счет ослабления изменения функции цели по отношению к каждому направлению координат. [16]

## **Метод золотого сечения:**

Метод Золотого сечения используется для нахождения максимума или минимума унимодальной функции. (Унимодальная функция содержит только один минимум или максимум на интервале [a, b].) Предположим, что мы пытаемся найти минимум функции. [17]

Как показано на рисунке, выберем три точки *a*, *b* и () вдоль оси x с соответствующими значениями функции . Так как и , минимум должен находиться между *a* и *b*. Теперь четвертая точка, обозначенная как , выбрана так, чтобы находиться между большим из двух интервалов и . Предполагая, что интервал больше, чем , мы обозначаем точку в интервале . Если , то новые три точки будут *a*, ,; иначе, если , то новые три точки будут ,, *b*. Этот процесс продолжается до тех пор, пока расстояние между внешними точками не станет достаточно малым.[18]



*Рис 5. График функции f(x)*

Следующий алгоритм может быть использован для определения минимума функции f(x).

1. Определите две промежуточные точки и такие, что

, где

1. Вычислите и .

Если , то определите новые *a*, ,, *b* как показано ниже. Обратим внимание, что поменялась только одна точка .

Если , то определите новые *a*, ,, *b* как показано ниже. Обратим внимание, что поменялась только одна точка .

1. Если (достаточно малое число), то является минимумом, то прекратите итерацию, иначе перейдите к шагу 2. [19]

***Преимущества золотого сечения:***

* + - На второй и последующих итерациях требуется вычисление только одного значения функции.
    - Обладает высокой эффективностью, наиболее подходит для решения одномерных унимодальных задач оптимизации[20]

***Недостатки золотого сечения:***

* Большое количество итераций
* Неустойчивость относительно ошибок округления
* Невозможность отыскать все экстремумы функции[21]

# **СРАВНЕНИЕ МЕТОДОВ**

Сравнивая данные методы можно определить где лучше всего нам применять их.

Сеточный поиск является самым простым методом для реализации. Но при этом слишком много вычислений производится, что увеличивает время поиска экстремума. Соответственно, сеточный поиск эффективен для малого количества комбинаций.

Метод координатного спуска также простой, так как сводится к нескольким задачам с одной переменной. Его нельзя использовать для негладких функций многих переменных, потому что алгоритм просто не сможет найти экстремум.

Метод золотого сечения считается самым эффективным и обладает высокой точностью, но, как и в сеточном поиске, имеет большое количество итераций.

# **ПРИМЕНЕНИЕ**

Параметрическая оптимизация применяется практически во всех технических сферах. Она экономит время поиска значений, а также помогает человеку в вычислениях (всегда есть алгоритм, по которому можно найти наш экстремум). Методы, которые мы рассмотрели в данной работе, находят применение при решении классических задач и задач с ограничениями в виде уравнений. Они просты в реализации и обладают довольно-таки высокой точностью. [22]

Также параметрическая оптимизация применяется в прогнозировании погоды, расчете различных параметров электродвигателей, а также в конструкторском проектировании САПР. [23]

На данный момент параметрическую оптимизацию используют во многих областях: аналитика, экономика, математика и т.д. Способов ее применения огромное множество, причем не только в технических сферах, но и в профессиях, которые никак не связаны с инженерной деятельностью. [24]

# **СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ**

1. В.В. Кийков, В.Ф. Кочкина, К.А. Вдовкин. Параметрическая оптимизация радиоэлектронных схем // Федеральное агентство по образованию ГОУ ВПО «Уральский государственный технический университет – УПИ». Екатеринбург, 2005
2. А.Н. Рязанцев Методические указания для студентов специальностей 1-36 01 01 «Технология машиностроения», 1-53 01 01 «Автоматизация технологических процессов и производств» Оптимизация технологических процессов // Государственное учреждение высшего профессионального образования «белорусско-российский университет», 2014
3. О.А. Хнаев, И.А. Пчелинцев. Параметрическая оптимизация систем. Методы решения экстремальных задач, 2012
4. Р.Ф. Маликов. Этапы программирования и моделирования // Башкирский государственный педагогический университет им. М. Акмуллы, 2012
5. [Collins Ayuya](https://www.section.io/engineering-education/authors/collins-ayuya/). Using Grid Search to Optimize Hyperparameters, 2021
6. Jinhyo Kim Iterated Grid Search Algorithm on Unimodal Criteria, 1997
7. В.А. Аганесов Поиск по сетке в Python, 2021
8. Корнилов А.Г. Методические материалы для изучения алгоритмов реализации методов безусловной оптимизации непрерывных одномерных и многомерных унимодальных функций // Казанский Государственный Технический Университет им. А. Н. Туполева. Казань, 2003
9. Б.Я. Советов С.А. Яковлев. Моделирование систем Издание третье, переработанное и дополненное, «Издательство «Высшая школа», 2001
10. [Gianluca Malato](https://www.yourdatateacher.com/author/admin/) Hyperparameter tuning. Grid search and random search, 2021
11. [Abhisek Jana](https://www.adeveloperdiary.com/author/adeveloperdiarygmail-com/) Introduction to Coordinate Descent using Least Squares Regression, 2018
12. В.А. Гончаров Методы оптимизации Министерство образования и науки Российской Федерации Московский государственный институт электронной техники (технический университет). Москва, 2008
13. М. Э. Аббасов Методы оптимизации Санкт-петербургский государственный университет Факультет прикладной математики – процессов управления, 2014
14. С.С.Михалкович, А.В.Олифер, А.М.Столяр ЧИСЛЕННЫЕ МЕТОДЫ Выпуск III Оптимизация. Системы нелинейных уравнений. Ростов-на-Дону, 2000
15. University of Illinois at Chicago, Department of Mathematics, Statistics and Computer Science, 2005
16. Орлов Н.Н. Системы автоматизированного проектирования электромеханических устройств, 1989
17. Гилл Ф.Н. Численные методы условной оптимизации, 1977
18. Фадеева О. С. [Информатика\_Курсовой проект\_вар.30](https://studfile.net/users/Oksana/folder:6165/), [Томский Государственный Университет Систем Управления и Радиоэлектроники](https://studfile.net/tusur/), 2014
19. Лаптева Е.Н., Кулай Н.В. Оптимизация. Методы многомерного поиска, 2002
20. В.В. Аюпов Математическое моделирование технических систем, 2017
21. Фадеева О. С. Параметрическая оптимизация электрической цепи, 2014
22. Способы постановки задач параметрической оптимизации, 2017
23. О. О. Луковенкова, Ю. В. Марапулец, А. Б. Тристанов, А. А. Ким, И. А. Кашутина Оптимизация метода адаптивного согласованного преследования для анализа сигналов геоакустической эмиссии, 2018
24. Roger D. Peng Advanced Statistical Computing, 2021