

|  |
| --- |
| МИНОБРНАУКИ РОССИИ |
| Федеральное государственное бюджетное образовательное учреждение  высшего образования  **"МИРЭА** - **Российский технологический университет"**  **РТУ МИРЭА** |

**Институт** Информационных Технологий

**Кафедра** Вычислительной Техники

**ОТЧЁТ ПО ПРАКТИЧЕСКИМ РАБОТАМ**

**по дисциплине**

**«Математическое обеспечение систем поддержки принятия решений»**

Студент группы: ИКБО-14-20 Вежновец Ф.Ю.\_\_ *(Ф.И.О.студента)*

Руководитель \_\_Семенов Р.Э.\_\_

*(Ф.И.О. преподавателя)*

Москва 2023

**Содержание**

[ПРАКТИЧЕСКАЯ РАБОТА №1. МЕТОД НУЛЕВОГО ПОРЯДКА: СИМПЛЕКС МЕТОД 3](#_Toc132142678)

[1.1 Постановка задачи 3](#_Toc132142679)

[1.2 Описание симплекс метода 3](#_Toc132142680)

[1.3 Ручной расчёт итераций симплекса 4](#_Toc132142681)

[1.4 Программная реализация 6](#_Toc132142682)

[1.5 Вывод 10](#_Toc132142683)

[ПРАКТИЧЕСКАЯ РАБОТА №2. МЕТОД НУЛЕВОГО ПОРЯДКА: МЕТОД НЕЛДЕРА-МИДА 11](#_Toc132142684)

[2.1 Постановка задачи 11](#_Toc132142685)

[2.2 Описание симплекс метода 11](#_Toc132142686)

[2.3 Ручной расчёт итераций Нелдера-Мида 13](#_Toc132142687)

[2.4 Программная реализация 15](#_Toc132142688)

[2.5 Вывод 19](#_Toc132142689)

ПРАКТИЧЕСКАЯ РАБОТА №1. МЕТОД НУЛЕВОГО ПОРЯДКА: СИМПЛЕКС МЕТОД

1.1 Постановка задачи

Пользуясь лекционными материалами и методическими указаниями реализовать симплексный метод.

Найти минимум целевой функции.

Симплексным методом с точностью

Размерность задачи n = 2.

Зададим начальную точку симплекса и длину ребра симплекса m = 0,07.

1.2 Описание симплекс метода

Регулярным симплексом в n-мерном пространстве называется правильный многогранник с n+1вершиной. При n = 2 симплексом является правильный треугольник, при n = 3 – тетраэдр и т.д. Отрезок, соединяющий 2 вершины симплекса, называется ребром симплекса.

Поиск симплексным методом ведется по следующей схеме. Устанавливаются координаты вершин симплексов. Определяется вершина с наибольшим значением целевой функции. Вместо нее сроиться новая вершина отражением старой через центр тяжести остальных вершин симплекса. На рисунке 1.1 представлен процесс построения нового симплекса на плоскости.



Рисунок 1.1 – Построение нового симплекса

Если попытка отражения не приводит к уменьшению целевой функции, то выполняется операция редукции, в результате которой формируется новый симплекс с уменьшенными вдвое сторонами. При операции редукции в качестве базовой точки выбирается вершина старого симплекса, в которой функция принимает минимальное значение.

В результате исключения вершин симплексов с наибольшим значением целевой функции процесс поиска сходится к минимальному значению.

Поиск завершается, когда разности между значениями функции в центре тяжести симплекса и вершинах становится достаточно малым.

1.3 Ручной расчёт итераций симплекса

Отобразим изначальную таблицу (Таблица 1.1).

Таблица 1.1 – Изначальные данные

|  |  |  |  |
| --- | --- | --- | --- |
| Номер вершины | Координата 1 | Координата 2 | Значение функции |
| 0 |  |  |  |

Вычислим приращения (Формулы 1.2 и 1.3):

Используя и , вычислим координаты двух остальных вершин симплекса (Формулы 1.4 и 1.5):

Итерация k = 0. Вычислим значение целевой функции в вершинах , , :

Отобразим эти значения в таблицe 1.2.

Таблица 1.2 – Новые данные

|  |  |  |  |
| --- | --- | --- | --- |
| Номер вершины | Координата 1 | Координата 2 | Значение функции |
| 0 |  |  |  |
| 1 |  |  |  |
| 2 | ,067 |  |  |

Наибольшее значение целевой функции соответствует вершине , поэтому необходимо отразить ее относительно центра тяжести остальных вершин и . центр тяжести расположен в точке

Используя свойство регулярности, найдем координаты отраженной вершины

В полученной вершине значение целевой функции 𝑓() = 1216,495

Таким образом, получим таблицу 1.3

Таблица 1.3 – Новые данные

|  |  |  |  |
| --- | --- | --- | --- |
| Номер вершины | Координата 1 | Координата 2 | Значение функции |
| 0 |  |  |  |
| 1 |  |  |  |
| 2 |  |  |  |
| 3 |  |  |  |

Следовательно, наблюдается уменьшение целевой функции < . Новый симплекс образован вершинами , , , которым соответствует значение целевой функции , , .

Проверим условие окончания поиска . Определим центр тяжести симплекса

В полученной точке .

Вычислим , , .

Так как все условия окончания поиска не выполняются, то процесс итераций должен быть продолжен.

1.4 Программная реализация

Реализуем симплекс метод на языке высокого уровня Python.

На рисунках 1.2 – 1.6 представлен результат работы программы. Листинг кода приведен в Приложении А.

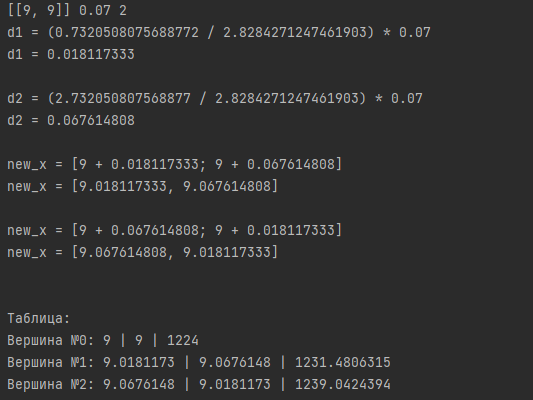


Рисунок 1.2 – Результат работы программы

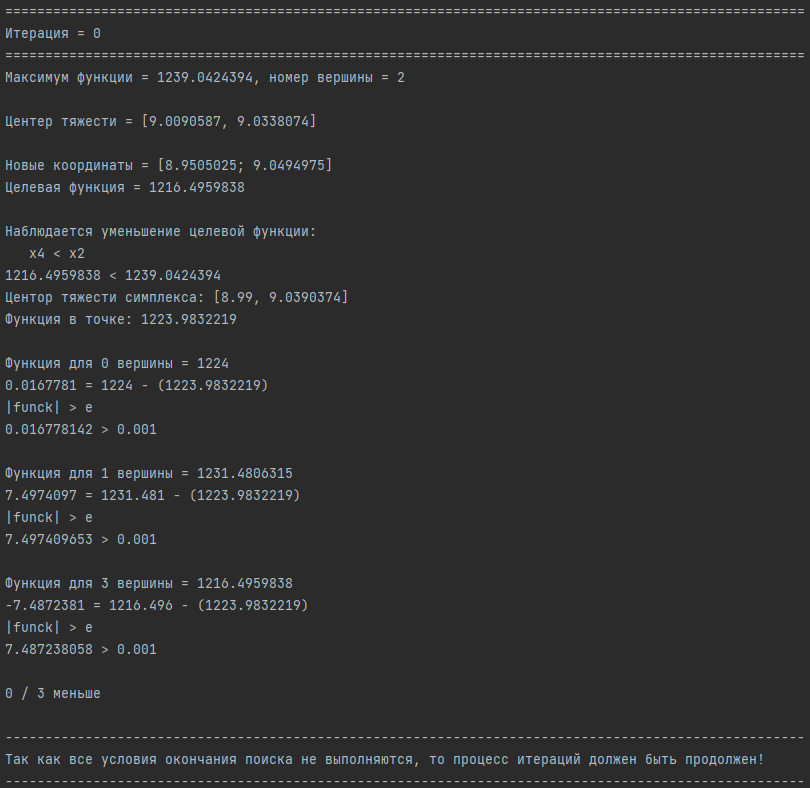


Рисунок 1.3 – Результат работы программы

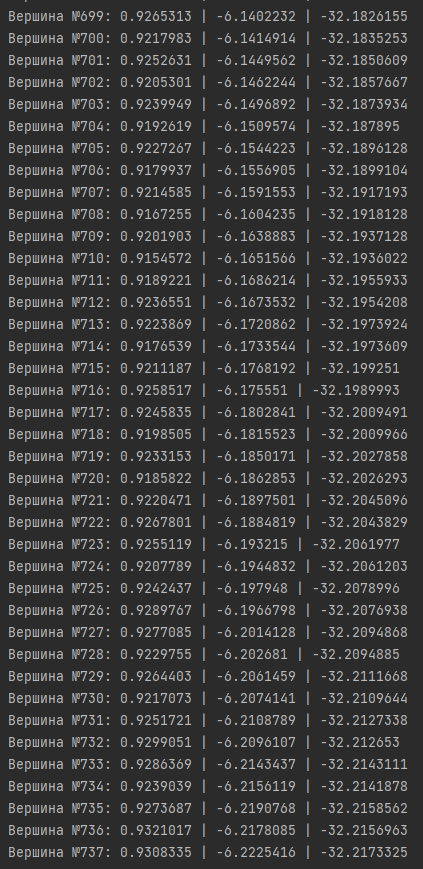


Рисунок 1.4 – Результат работы программы

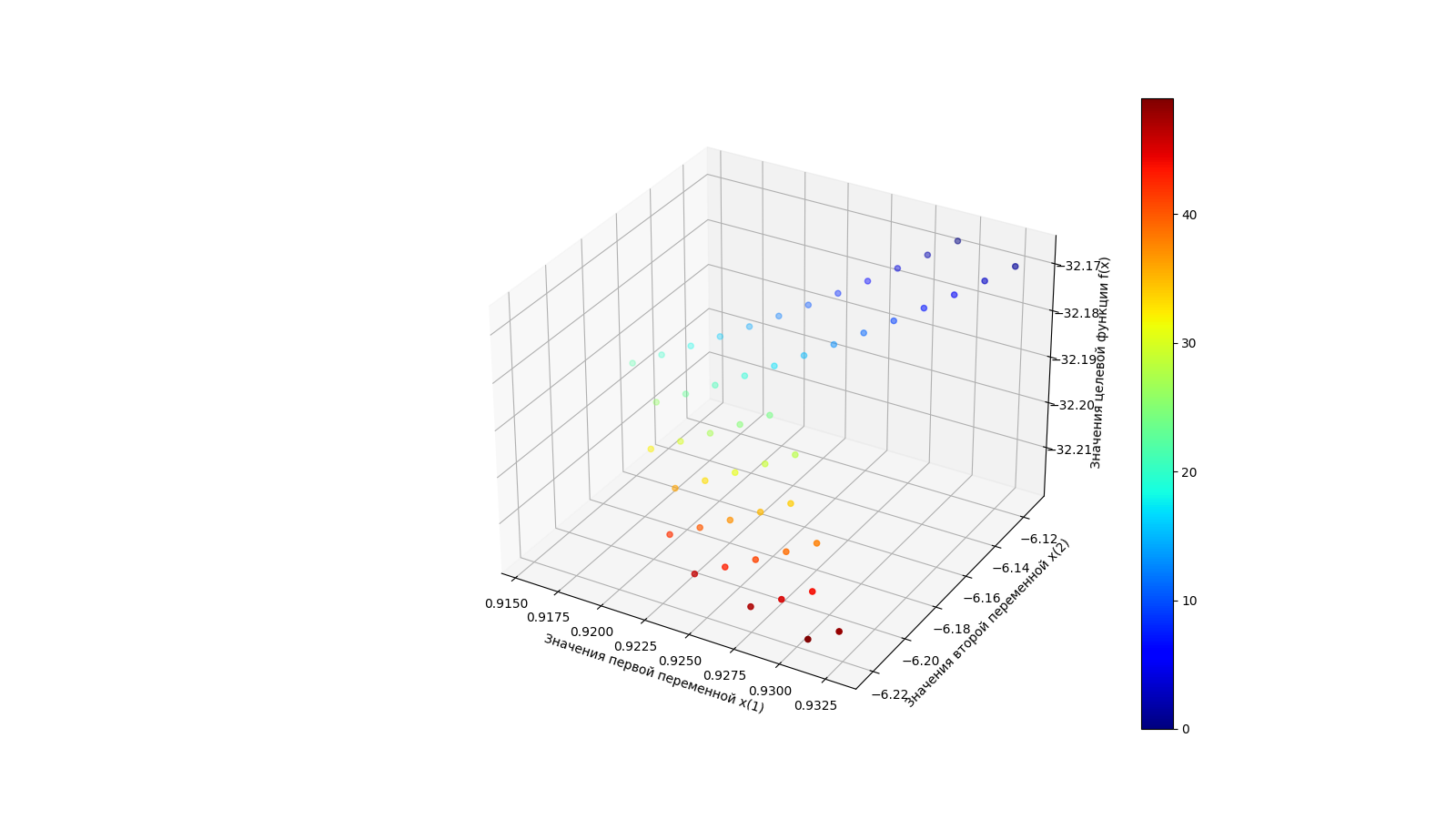


Рисунок 1.5 – Последние 50 элементов таблицы симплекса

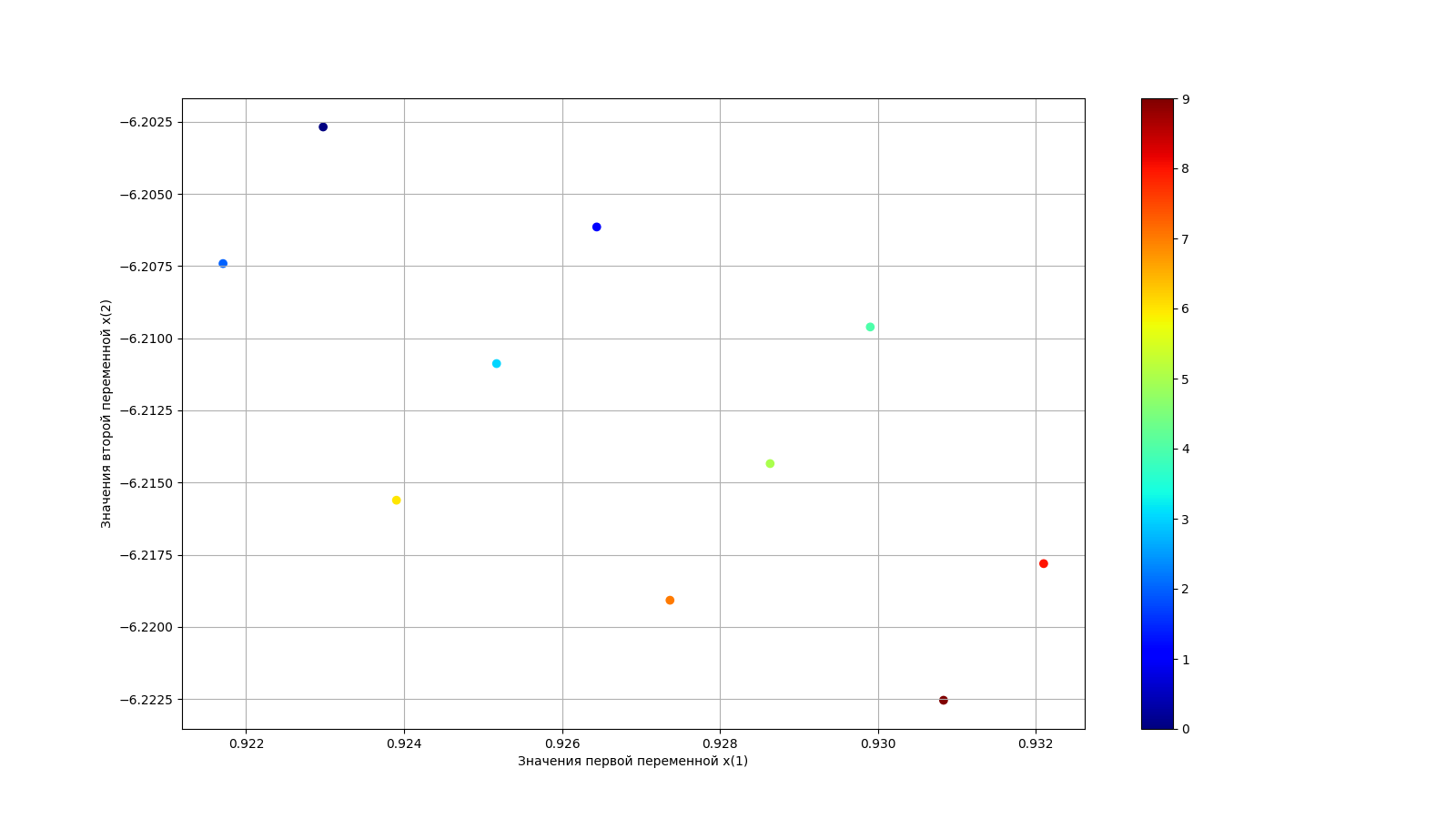


Рисунок 1.6 – Последние 5 элементов таблицы симплекса

Сравним полученные результаты при помощи сайта WolframAlpha (Рисунок 1.7).

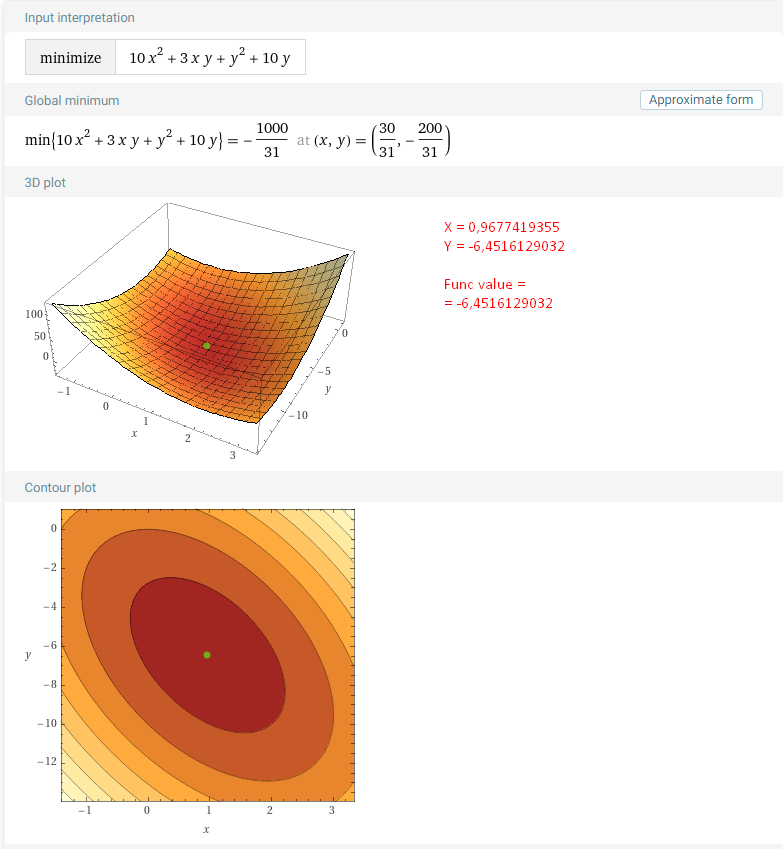


Рисунок 1.7 – Результаты минимизации функции при помощи WolframAlpha

1.5 Вывод

По результатам работы, был реализован симплекс метод (метод нулевого порядка).

ПРАКТИЧЕСКАЯ РАБОТА №2. МЕТОД НУЛЕВОГО ПОРЯДКА: МЕТОД НЕЛДЕРА-МИДА

2.1 Постановка задачи

Пользуясь лекционными материалами и методическими указаниями.

Найти минимум целевой функции.

методом Нелдера-Мида с точностью

Размерность задачи .

Длину ребра симплекса

Параметр растяжения

Параметр сжатия

Зададим начальную точку симплекса .

2.2 Описание симплекс метода

В 1964 году Нелдер и Мид предложили модификацию, в которой симплекс может изменять свою форму (растягиваясь и сжимаясь) в зависимости от свойств поверхности целевой функции. Так как в этом случае симплекс не будет уже регулярным, метод назвали поиском по деформируемому многограннику.

И так модифицируем рассмотренный на предыдущей лекции алгоритм минимизации целевой функции по регулярному симплексу, добавив к процедуре отражения при построении нового симплекса процедуры сжатия и растяжения. Геометрическая иллюстрация этих процедур для случая 𝑛=2 представлена на рис. 1, 2 и 3, где введены следующие обозначения:

– наибольшее значение целевой функции;

– следующее по величине за наибольшим значение целевой функции;

– наименьшее значение целевой функции;

– текущие значения целевой функции

1. Операция отображения (Рисунок 2.1)



Рисунок 2.1 – Операция отображения

1. Если то выполняется операция растяжения , где 𝛽 – параметр растяжения (Рисунок 2.2).



Рисунок 2.2 – Операция растяжения

1. Если , то выполняется операция сжатия , где – параметр сжатия (Рисунок 2.3).



Рисунок 2.3 – Операция сжатия

При решении практически задач минимизации параметры растяжения и сжатия 𝛾 Нелдер и Мид рекомендует брать , , но Правильнее – выбирать эти параметры из интервалов и .

2.3 Ручной расчёт итераций Нелдера-Мида

Отобразим изначальную таблицу (Таблица 2.1).

Таблица 2.1 – Изначальные данные

|  |  |  |  |
| --- | --- | --- | --- |
| Номер вершины | Координата 1 | Координата 2 | Значение функции |
| 0 |  |  |  |

Вычислим приращения (Формулы 1.2 и 1.3):

Используя и , вычислим координаты двух остальных вершин симплекса (Формулы 1.4 и 1.5):

Итерация k = 0. Вычислим значение целевой функции в вершинах , , и обозначим наибольшее значение функции , следующее за наибольшим значением , наименьшее значение функции :

, , ;

Отобразим эти значения в таблицe 2.2.

Таблица 2.2 – Новые данные

|  |  |  |  |
| --- | --- | --- | --- |
| Номер вершины | Координата 1 | Координата 2 | Значение функции |
| 0 |  |  |  |
| 1 |  |  |  |
| 2 | ,067 |  |  |

Наибольшее значение целевой функции соответствует вершине , поэтому необходимо отразить ее относительно центра тяжести остальных вершин и . центр тяжести расположен в точке

Используя свойство регулярности, найдем координаты отраженной вершины

В полученной вершине значение целевой функции ) = 1216,495

Таким образом, получим таблицу 2.3

Таблица 2.3 – Новые данные

|  |  |  |  |
| --- | --- | --- | --- |
| Номер вершины | Координата 1 | Координата 2 | Значение функции |
| 0 |  |  |  |
| 1 |  |  |  |
| 2 |  |  |  |
| 3 |  |  |  |

Следовательно, наблюдается уменьшение целевой функции < .

Так как не выполняется условие то выполним операцию растяжения симплекса.

В полученной вершине значение целевой функции .

Условие растяжения выполнено

Проверим условие окончания поиска. Определим координаты центра тяжести симплекса

В полученной точке .

Вычислим 𝜎 (сигма)



Так как условие окончания поиска не выполняется, то процесс итерации должен быть продолжен.

2.4 Программная реализация

Реализуем симплекс метод на языке высокого уровня Python.

На рисунках 2.2 – 2.7 представлен результат работы программы. Листинг кода приведен в Приложении А.

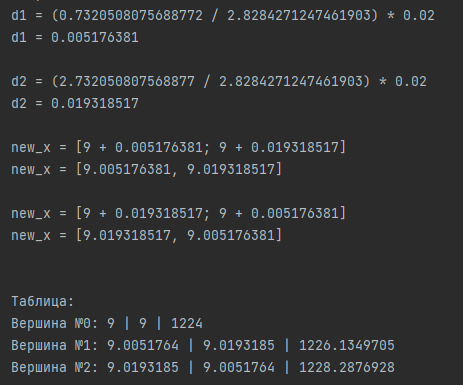


Рисунок 2.2 – Результат работы программы

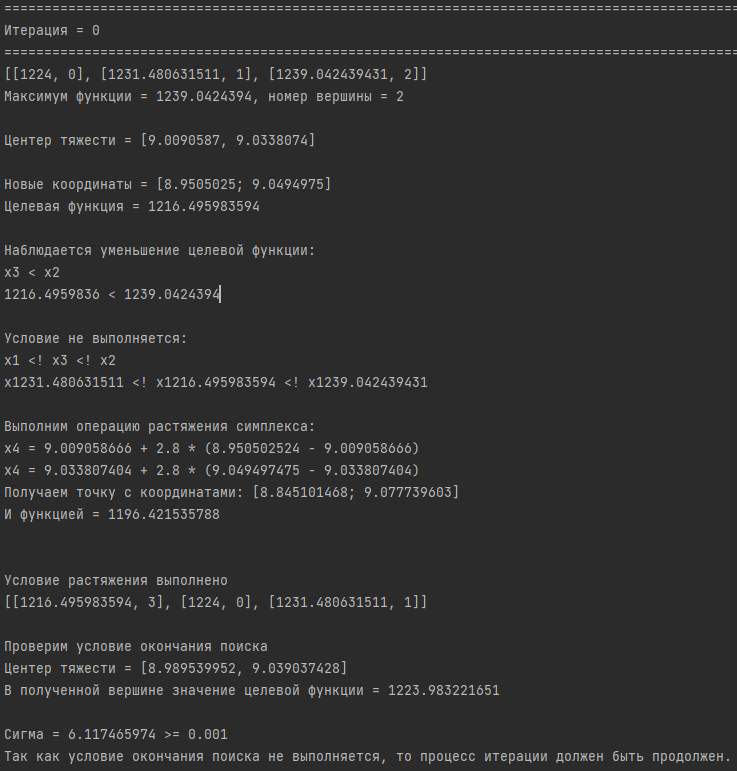


Рисунок 2.3 – Результат работы программы

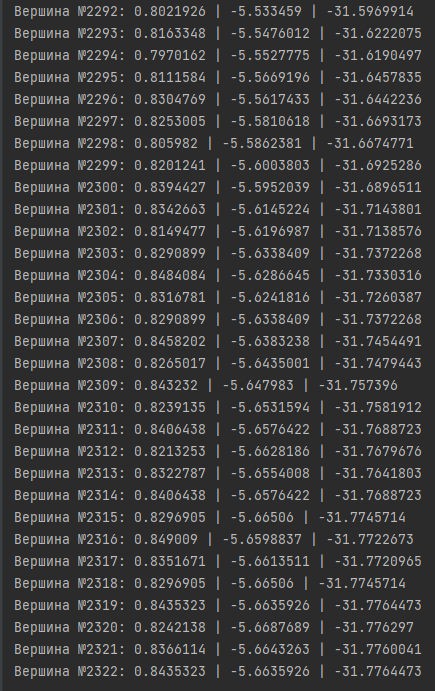


Рисунок 2.4 – Результат работы программы

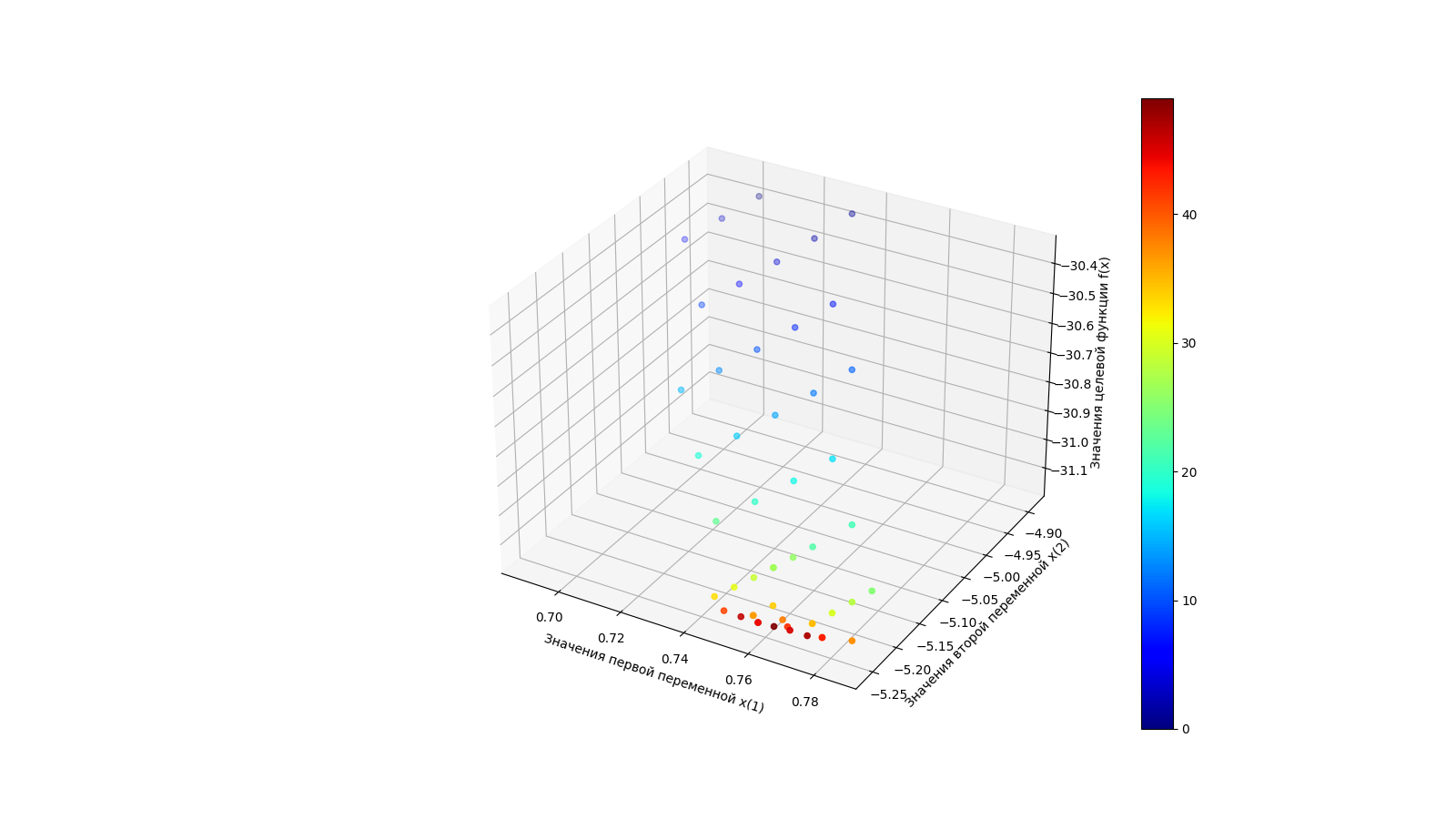


Рисунок 2.5 – Последние 50 элементов таблицы Нелдера Мида

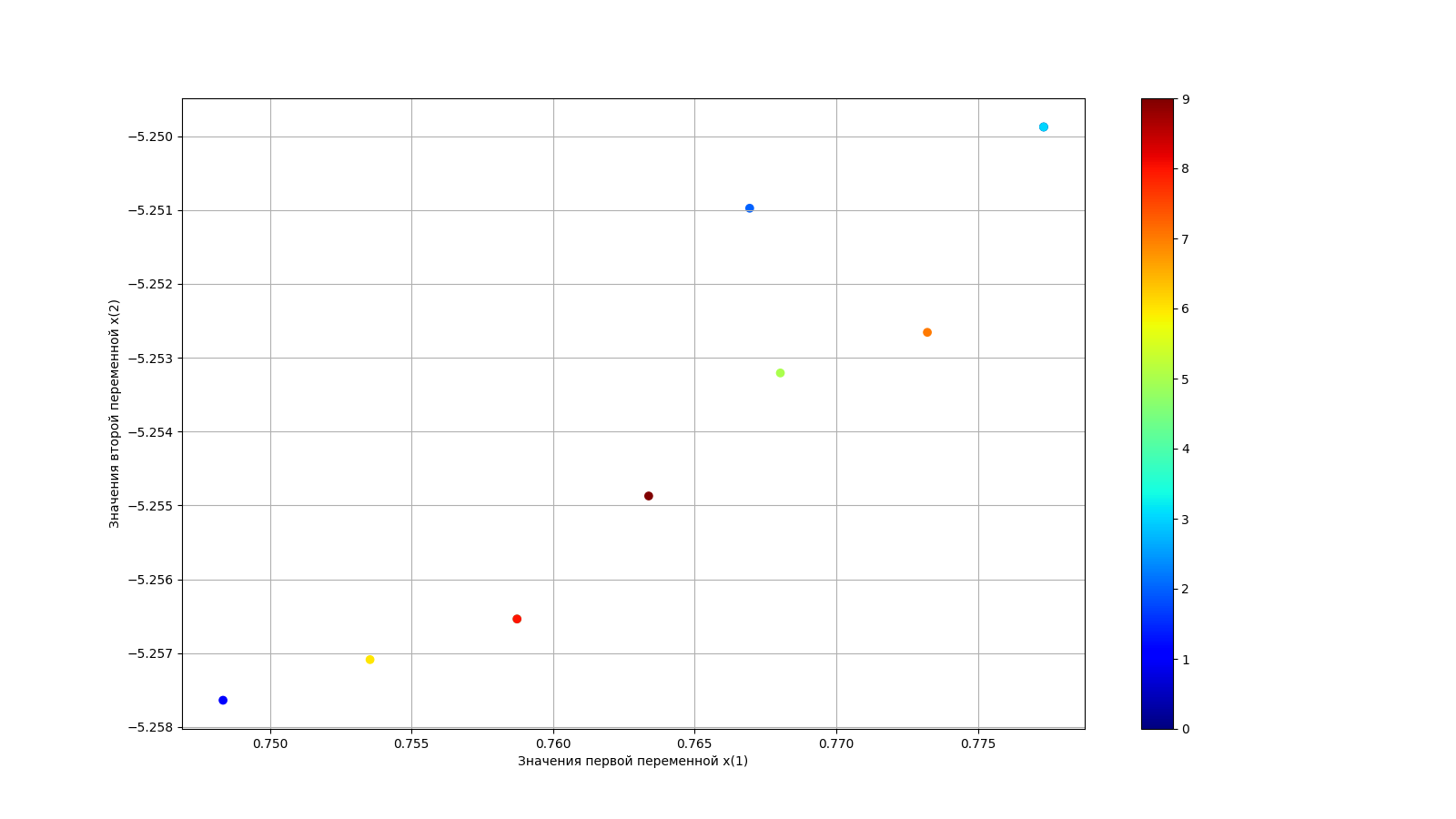


Рисунок 2.6 – Последние 5 элементов таблицы Нелдера Мида

Сравним полученные результаты при помощи сайта WolframAlpha.



Рисунок 2.7 – Результаты минимизации функции при помощи WolframAlpha

2.5 Вывод

По результатам работы, был реализован метод Нелдера Мида (метод нулевого порядка).

ПРАКТИЧЕСКАЯ РАБОТА №3. МЕТОД ПЕРВОГО ПОРЯДКА: МЕТОД ГРАДИЕНТНОГО СПУСКА П ПОСТОЯННЫМ ШАГОМ

3.1 Постановка задачи

Пользуясь лекционными материалами и методическими указаниями.

Найти минимум целевой функции

методом градиентного спуска с постоянным шагом с точностью

Зададим начальную точку .

Начальную величину шага .

3.2 Описание симплекс метода

Сущность метода градиентного спуска с постоянным шагом заключается в следующем. Выбирается начальная точка из области определения функции . Координаты новой точки вычисляются по формуле

где *k* − номер итерации *k* = 0, 1, …,

− величина шага,

– градиент функции в точке ,

Начальная величина шага задаётся пользователем. В каждом новой точке поиска проверяется условие убывания функции . Если условие нарушается, то постепенно уменьшается величина шага , т.е. точка приближается к точке до тех пор, пока условие не выполнится. В полученной точке определяется новое направление градиента и осуществляется новый спуск. Процесс продолжается пока не будет выполнено условие окончания поиска. В качестве условия окончания поиска используется близость к нулю нормы градиента .

Геометрическая иллюстрация поиска минимума целевой функции методом градиентного спуска с постоянным шагом для случая *n* = 2 представлена на рисунке 3.1.



Рисунок 3.1 - Графическая иллюстрация поиска точки минимума методом градиентного спуска с постоянным шагом

3.3 Ручной расчёт итераций методом градиентного спуска с постоянным шагом

Отобразим изначальную таблицу (Таблица 3.1).

Таблица 3.1 – Изначальные данные

|  |  |  |  |
| --- | --- | --- | --- |
| Номер вершины | Координата 1 | Координата 2 | Значение функции |
| 0 |  |  |  |

Найдем градиент функции в произвольной точке

***Итерация k = 0.*** Вычислим значение целевой функции 𝑓 (𝑥 (0)) и градиент

в начальной точке 𝑥 (0): 𝑓 (𝑥 (0)) = 1224; ∇𝑓 (𝑥 (0)) = (207, 55) 𝑇.

Определим координаты точки

и значение целевой функции в этой точке

Сравним и . Поскольку , то условие убывания функции не выполнено. Уменьшим величину шага и повторим вычисления координат точки .

Определим координаты точки

и значение целевой функции в этой точке

Сравним и . Поскольку , то условие убывания функции не выполнено. Уменьшим величину шага и повторим вычисления координат точки .

Определим координаты точки

и значение целевой функции в этой точке

Сравним и . Поскольку , то условие убывания функции не выполнено. Уменьшим величину шага и повторим вычисления координат точки .

Определим координаты точки

и значение целевой функции в этой точке

Сравним и . Поскольку , то условие убывания функции выполнено. Составим таблицу 3.2.

Таблица 3.2 – Новая таблица данных

|  |  |  |  |
| --- | --- | --- | --- |
| Номер вершины | Координата 1 | Координата 2 | Значение функции |
| 0 |  |  |  |
| 1 |  |  |  |

Проверим условие окончания поиска. Для этого вычислим вектор градиента в точке :

Найдем норму вектора градиента

Так как условие окончания поиска не выполняется, то процесс итерации должен быть продолжен.

3.4 Программная реализация

Реализуем симплекс метод на языке высокого уровня Python.

На рисунках 3.2 – 3.10 представлен результат работы программы. Листинг кода приведен в Приложении А.

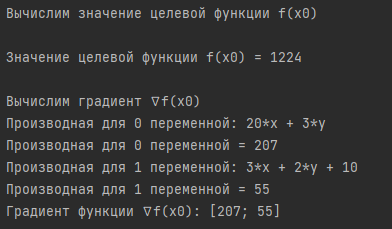


Рисунок 3.2 – Результат работы программы

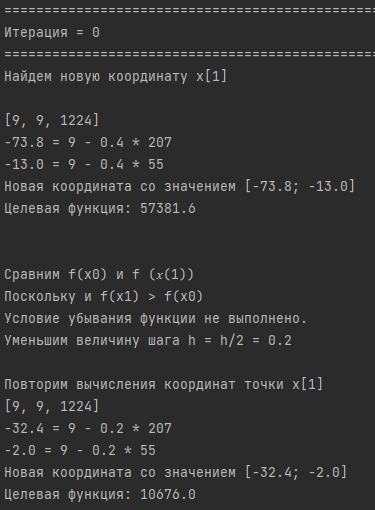


Рисунок 3.3 – Результат работы программы

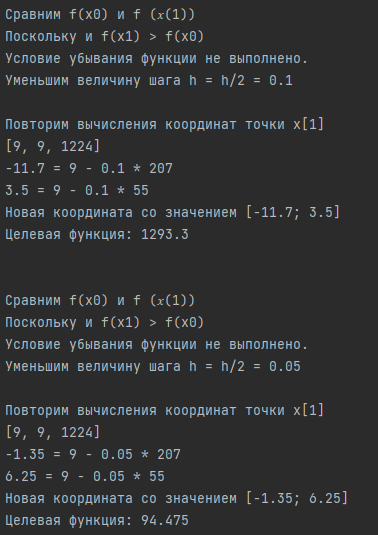


Рисунок 3.4 – Результат работы программы

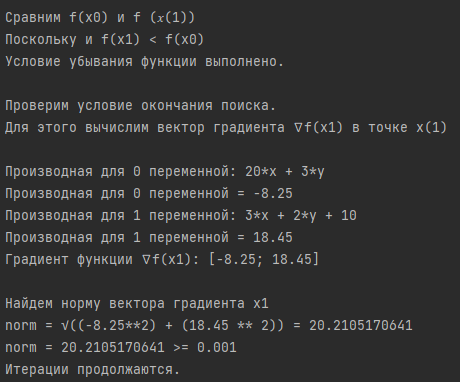


Рисунок 3.5 – Результат работы программы



Рисунок 3.6 – Результат работы программы

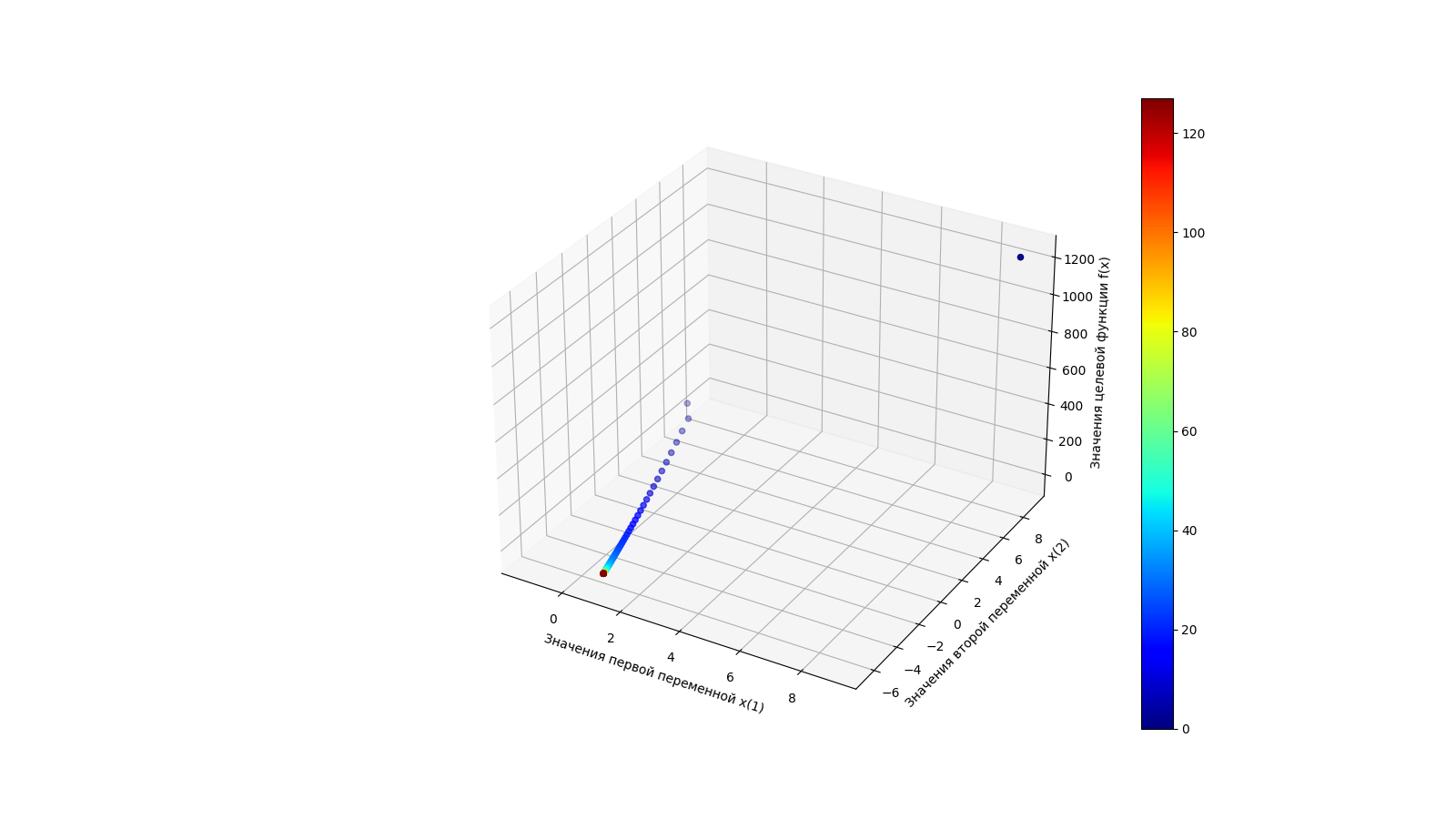


Рисунок 3.7 – Все элементов таблицы методом градиентного спуска с постоянным шагом

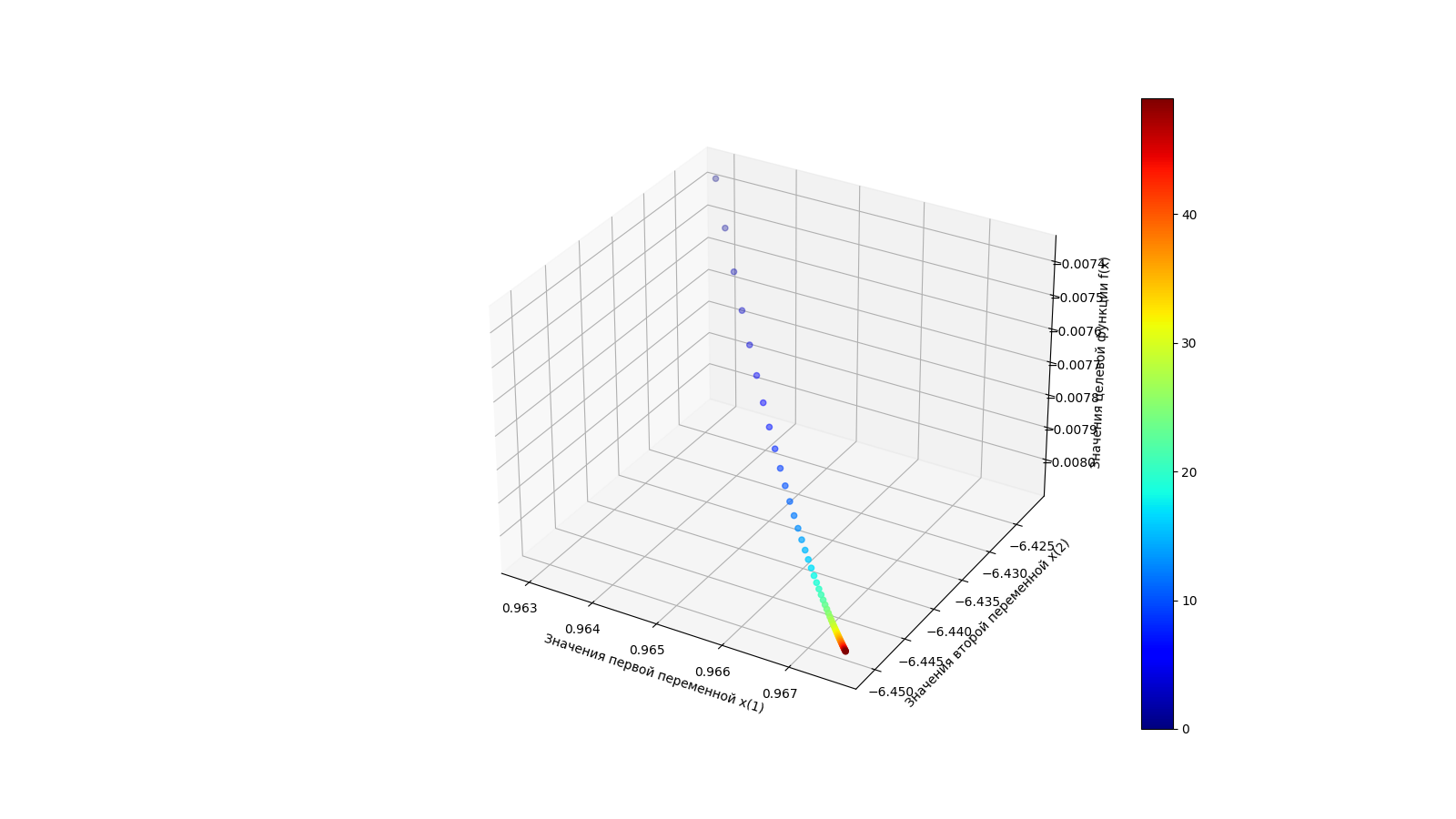


Рисунок 3.8 – Последние 50 элементов таблицы методом градиентного спуска с постоянным шагом

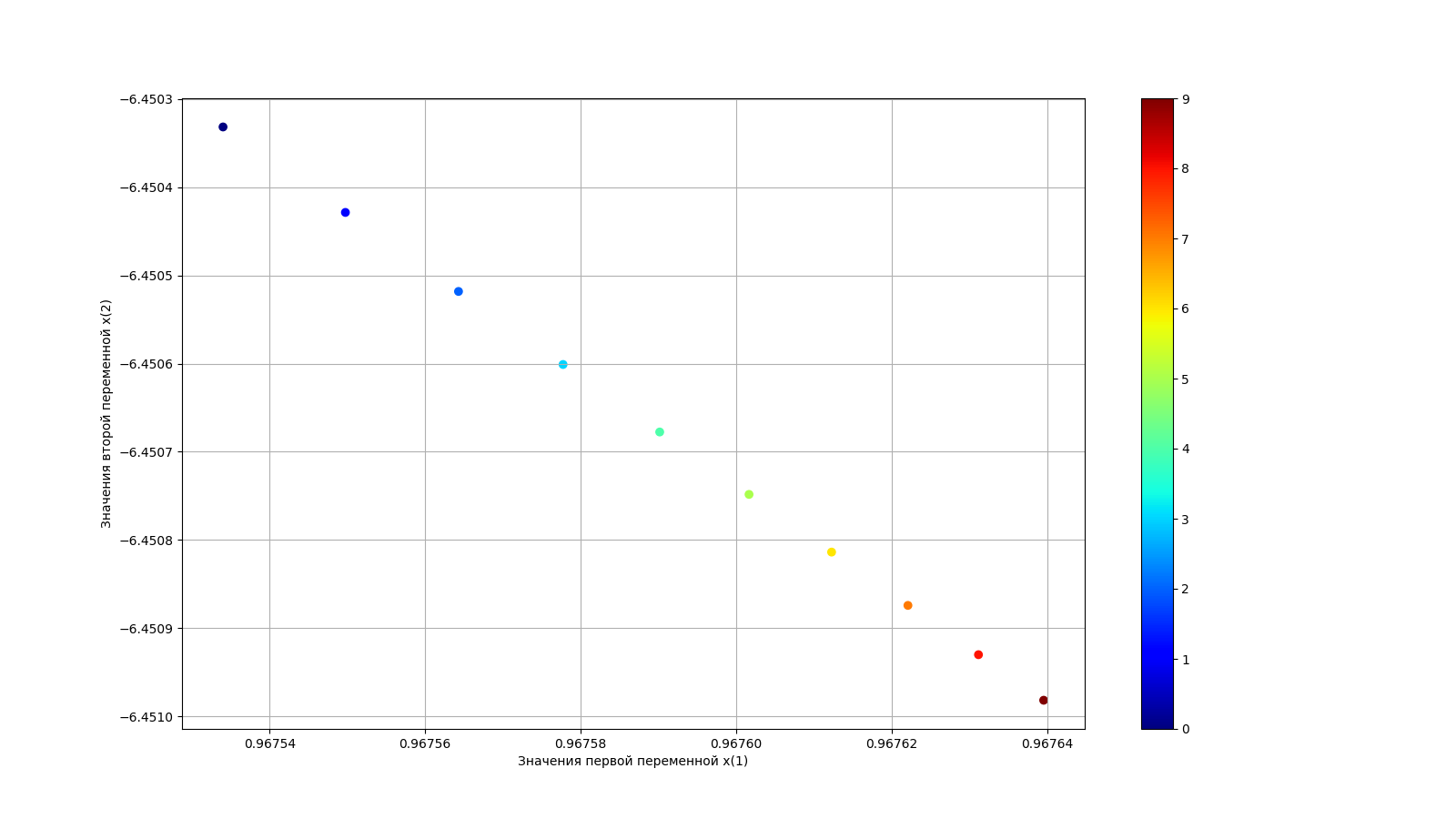


Рисунок 3.9 – Последние 5 элементов таблицы методом градиентного спуска с постоянным шагом

Сравним полученные результаты при помощи сайта WolframAlpha.



Рисунок 3.10 – Результаты минимизации функции при помощи WolframAlpha

3.5 Вывод

По результатам работы, был реализован методом градиентного спуска с постоянным шагом (метод первого порядка).