



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 _____ «Информатика и системы управления»

КАФЕДРА _____ «Теоретическая информатика и компьютерные технологии»

Лабораторная работа № 6
по курсу «Численные методы линейной алгебры»
«Изучение скорости сходимости однопараметрического метода»

Студент группы ИУ9-71Б Баев Д.А

Преподаватель Посевин Д. П.

Москва 2023

1 Задание

1. Реализовать однопараметрический метод для положительной симметричной матрицы произвольного размера $N \times N$.

2. Вычислить спектр матрицы A методом Крылова или Данилевского, которые были реализованы ранее, и получить минимальное и максимальное значение спектра λ_{\min} и λ_{\max} . После чего вычислить $\tau_{\text{opt}} = 2/(\lambda_{\min} + \lambda_{\max})$.

3. Построить график зависимости количества итераций n решения уравнения $A \cdot x = f$ однопараметрическим методом в зависимости от значения τ лежащего в пределах от 0 до $2/\lambda_{\max}$. Определить τ_{opt} из графика и сравнить с теоретическим значением полученным в пункте 2.

4. Для каждого эксперимента пункта 3 вывести условие сходимости. Решение системы для оценки неравенства приведенного выше можно получить путем решения $A \cdot x = f$ методом Гаусса. Другими словами требуется убедиться в том, что выполняются условия теоремы о сходимости однопараметрического метода. Обязательно, дополнительно проверить и показать, что для каждого k модуль максимального значения λ_{\max} меньше 1.

2 Исходный код

Исходный код программы представлен в листингах 1– 2.

Листинг 1 — Реализация однопараметрического метода со всеми необходимыми проверками

```
1 def single_parameter(A, b, tao, spectre, delta=1e-7):
2     n = len(A)
3     P = np.eye(n) - tao * A
4     g = tao * b
5
6     x_gauss = np.linalg.solve(A, b)
7     Mu = np.array([1 - tao * spectre[i] for i in range(n)])
8     assert np.max(np.abs(Mu)) < 1
9     k = 0
10    x_prev = np.zeros(shape=(n, ))
11    difference = np.array([1] * n)
12    x_k = None
13    while norm(difference) > delta:
14        x_k = mul_matrix_by_vector(P, x_prev) + g
15        difference = x_k - x_prev
16        r = x_k - x_gauss
17        #if (norm(r) ** 2) > max(Mu[i] ** 2 for i in range(n)) * norm(
18        x_prev - x_gauss) ** 2:
19            # print("One", norm(r) ** 2)
20            # print("Two", max(Mu[i] ** 2 for i in range(n)) * norm(
21            x_prev - x_gauss) ** 2)
22            if np.round(norm(r) ** 2, 7) > np.round(max(Mu[i] ** 2 for i in
23            range(n)) * norm(x_prev - x_gauss) ** 2, 7):
24                print(f"Proverka is not okay: tao - {tao}, iter = {k}")
25                #return None, 0
26                x_prev = deepcopy(x_k)
27                k += 1
28    return x_k, k
```

Листинг 2 — Получение результатов и построение графика количества итераций от значения параметра

```
1 def get_tao_plot(n):
2     A = generate_sim_matrix(n=n, diag=1)
3     x = np.random.uniform(0, 10, size=(n, ))
4     b = mul_matrix_by_vector(A, x)
5
6
7     spectre = eig(A)
8
9     lambda_min, lambda_max = min(spectre), max(spectre)
10    assert lambda_min > 0
11
12    tao_opt = 2 / (lambda_min + lambda_max)
13    _, iter_opt = single_parameter(A, b, tao_opt, spectre)
14    print(f"Analytical solution tao_opt: {tao_opt}")
15    print(f"Optimal iterations: {iter_opt}")
16    tao_array = np.linspace(0, 2 / lambda_max, 100, True)
17    tao_array = tao_array[1:-1]
18    iters_array = []
19
20    for tao in tao_array:
21        _, i = single_parameter(A, b, tao, spectre)
22        iters_array.append(i)
23
24    i_exp = np.argmin(iters_array)
25    plt.plot(tao_array, iters_array)
26    plt.scatter(tao_opt, iter_opt, color="red")
27    plt.scatter(tao_array[i_exp], iters_array[i_exp])
28    print(f"Experiment solution tao_opt: {tao_array[i_exp]}")
29    print(f"Optimal iterations: {iters_array[i_exp]}")
```

3 Результаты

Результаты поиска оптимального значения метода и исследования количества итераций от значений параметра приведены на рисунках 1- 3.

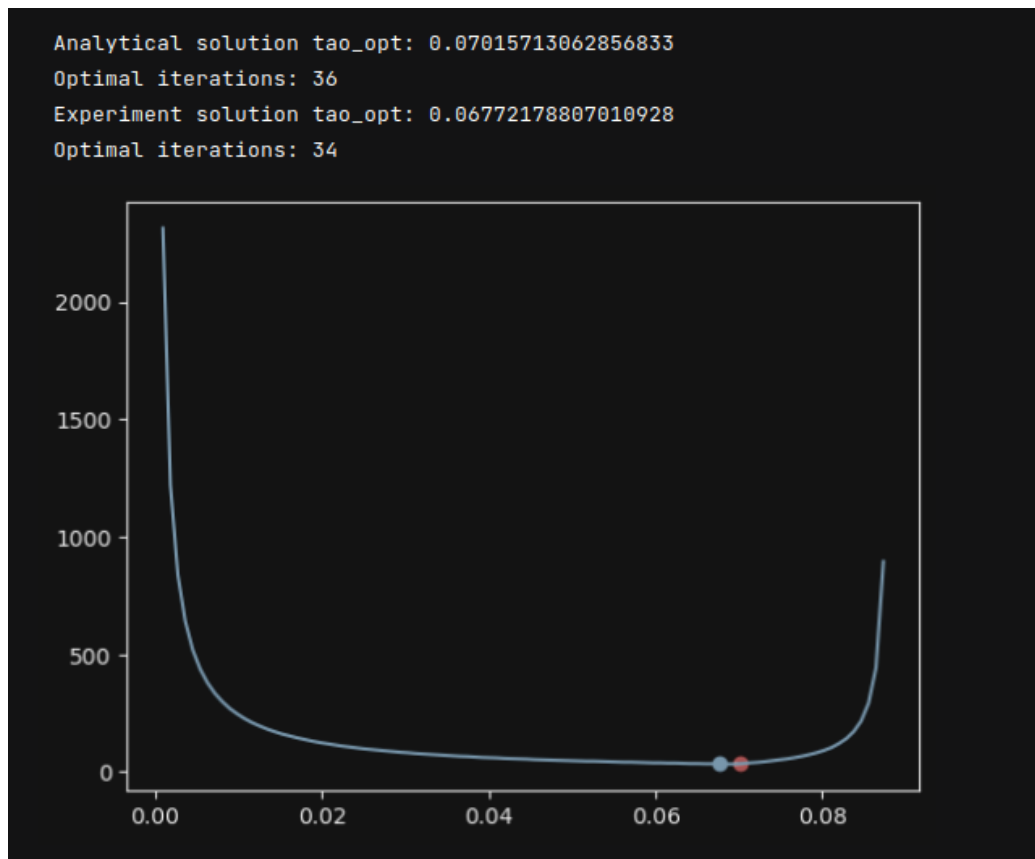


Рис. 1 — График зависимости количества итераций от значения параметра в области сходимости метода для матрицы 4x4

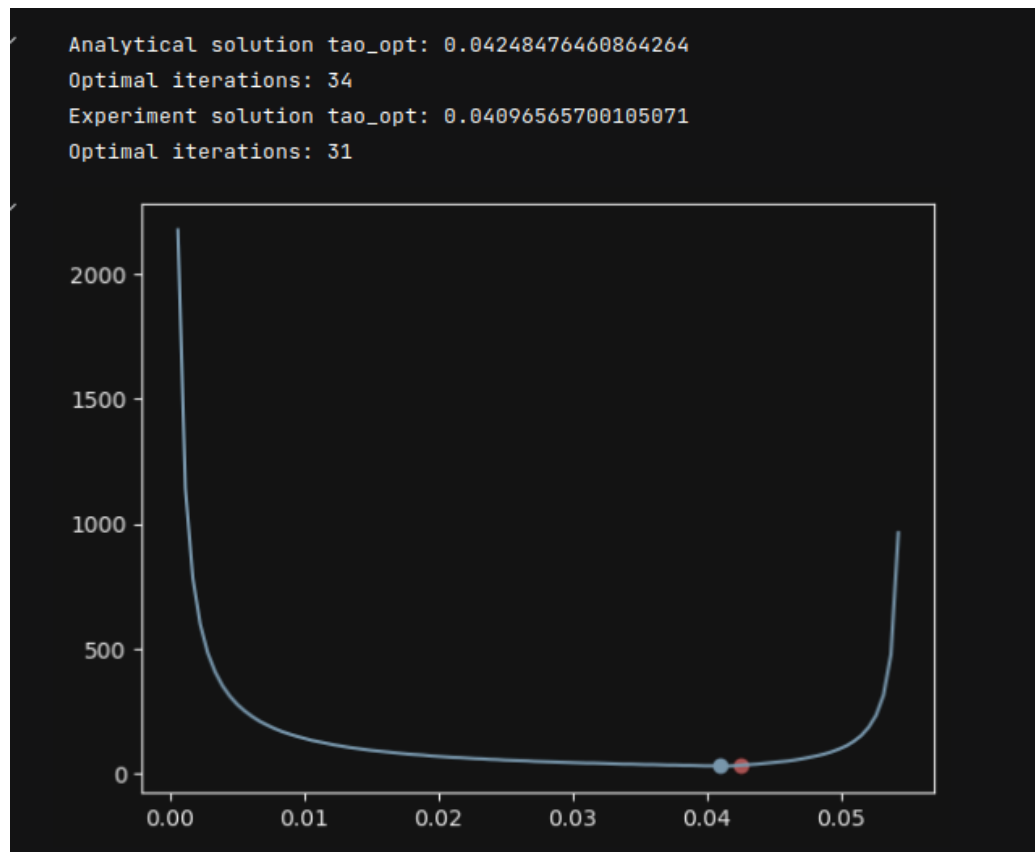


Рис. 2 — График зависимости количества итераций от значения параметра в области сходимости метода для матрицы 7×7

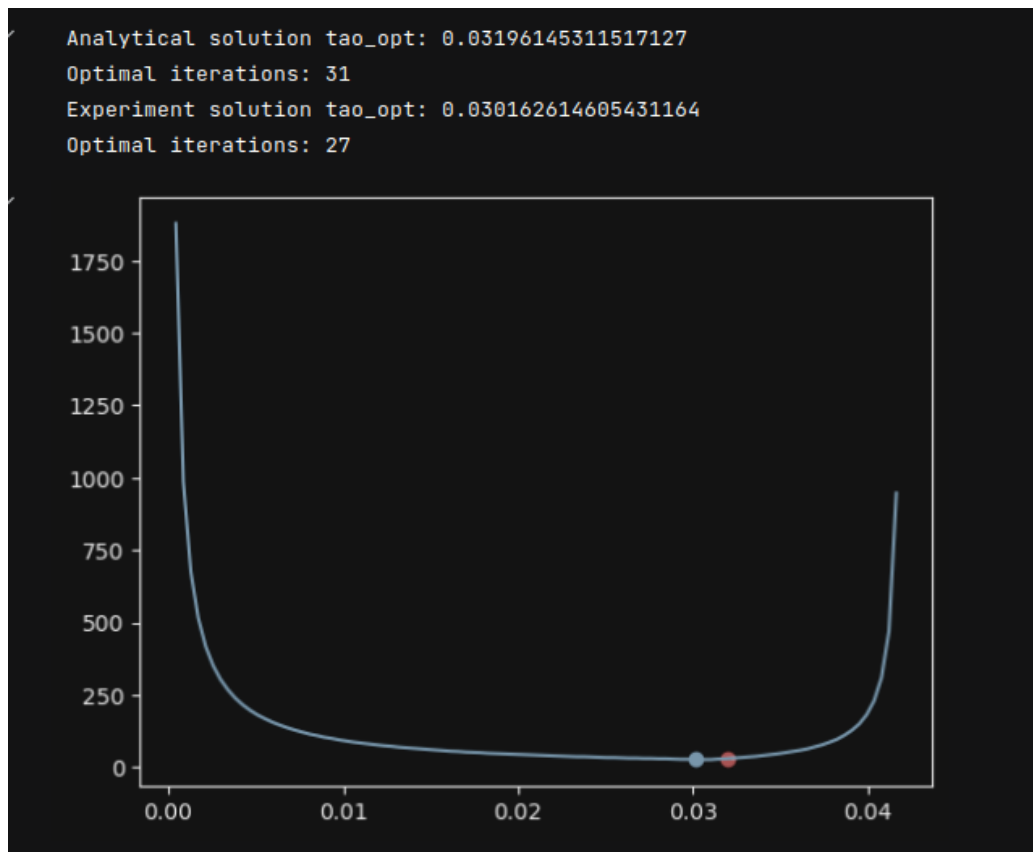


Рис. 3 — График зависимости количества итераций от значения параметра в области сходимости метода для матрицы 10×10

4 Выводы

В результате выполнения лабораторной работы был реализован однопараметрический метод решения СЛАУ с матрицей коэффициентов $N \times N$. В процессе была исследована скорость сходимости метода, а также определены аналитические и экспериментальные оптимальные значения параметра. Также было исследовано условие сходимости метода на каждой его итерации. Стоит заметить, что в практических условиях при значениях параметра, больших чем оптимальное, данное условие будет выполняться не всегда, ввиду неточности вычислений на ЭВМ. Данная проблема решается округлением