Федеральное государственное автономное образовательное учреждение высшего образования «Санкт-Петербургский политехнический университет Петра Великого»

Институт прикладной математики и механики

Кафедра прикладной математики

Лабораторная работа

По дисциплине: «Численные методы»

На тему: “Итерационные методы решения Систем Линейных Алгебраических Уравнений ”

Выполнил**:**

Студент гр. 3630102/90003

Попов Иван Дмитриевич

Санкт-Петербург

2020

Оглавление

[Формулировка и формализация задач 3](#_Toc56534883)

[Алгоритмы метода 3](#_Toc56534884)

[Анализ задачи и проверка условий 4](#_Toc56534885)

[Тестовый пример 5](#_Toc56534886)

[Контрольные тесты 6](#_Toc56534887)

[Модульная структура программы 6](#_Toc56534888)

[Численный анализ решения 6](#_Toc56534889)

[Вывод 9](#_Toc56534890)

# Формулировка и формализация задач

Найти численное решение системы линейных алгебраических уравнений с заданной точностью методом простых итераций, т.е. найти такой

Где – численно полученный вектор решений,- вектор точных решений, – матрица коэффициентов, – правая часть СЛАУ

Требуется исследовать эффективность метода простых итераций, получить зависимость количества итераций, необходимых для получения решения с заданной точностью, от заданной погрешности, норму разности точного и численного решения от погрешности, норму невязки от погрешности, а также при фиксированной точности зависимость количества итераций от определителя главной матрицы, норму разности точного и численного решения от определителя и норму невязки от определителя от определителя.

# Алгоритмы метода

В работе реализован метод Якоби – метод, принадлежащий к семейству методов простых итераций.

Условия применимости: для СЛАУ необходимо, чтобы главная матрица была невырожденной и диагонально преобладающей :

,

Алгоритм метода для :

Представим матрицу в виде:

где – диагональная, а – соответственно левая и правая строго треугольные матрицы. Тогда исходную систему можно переписать в следующем виде:

где , при условии, что на диагонали исходной матрицы нет нулей. Теперь, чтобы перейти к итерационному процессу, если – номер приближения, то приближение, записанное в поэлементном виде, будет следующим:

В качестве начального приближения можно взять нулевой вектор **= 0**. Чтобы получить решение с заданной точностью достаточно продолжать итерационный процесс до тех пор, пока выполнено неравенство:

где – некоторое число, удовлетворяющее неравенству ,матрица перехода.

# Анализ задачи и проверка условий

Метод Якоби требует диагонального преобладания и невырожденности главной матрицы, чтобы ответ сходился к решению СЛАУ.

Матрицу коэффициентовбудем строить следующим образом: возьмем случайный вектор, но такой, чтобы его компонентами были числа одного порядка. Построим диагональную матрицу Далее возьмем случайный нормированный вектор. Построим матрицу Хаусхолдера **:**

.

Тогда матрица **.** Матрица коэффициентов – положительно определенная и симметричная, это гарантируется свойствами матрица Хаусхолдера, а ее определитель равен определителю диагональной матрицы. Далее будем увеличивать по модулю элементы на главной диагонали до тех пор, пока матрица не станет диагонально преобладающей. Можно положить вектор правыx частей как Теперь, уменьшить определитель матрицы, будем делить все элементы диагональной матрицы на одно из ее компонент:

Так, чтобы получить наименьший определитель из исходной диагональной матрицы, нужно разделить на наибольший по модулю из ее компонент.

# Тестовый пример

Продемонстрируем принцип работы методом на следующей СЛАУ:

Решение системы — это вектор . Главная матрица невырожденная: Матрица диагонально преобладающая:

Следовательно, метод Якоби простых итераций сходится к ответу. Положим нулевой вектор как начальное приближение. Тогда метод может быть реализован следующим образом:

Первая итерация:

Вторая итерация:

Третья итерация:

# Контрольные тесты

Будем строить по ранее описанному алгоритму матрицы коэффициентов и вектор свободных членов. Меняя погрешность с до , будем решать СЛАУ методом Якоби и вычислять норму разницы между точным решением и полученным решением :

Далее будем вычислять количество итераций и норму невязки :

Теперь, чтобы исследовать зависимость тех же величин от определителя главной матрицы, зафиксируем погрешность в , и будем делить элементы диагональной матрицы на один из ее компонент, сначала на минимальный, во второй раз – на второй по абсолютной величине элемент, и так до максимального компонента, и каждый раз вычислять количество итераций, норму невязки и норму разница между точным и численным ответом.

# Модульная структура программы

struct mtr

// структура для хранения матрицы, содержит двумерный массив

double VectorNormInf(double\* vector)

// вычисляет бесконечную норму вектора

// принимает массив с элементами из вектора

//возвращает бесконечную норму вектора

void RightSideProduct(mtr m, double\* b)

// вычисляет произведение матрицы на вектор

// принимает матрицу и массив из элементов вектор

// возвращает вектор, являющийся произведением матрицы и вектора

void Jacobi (mtr m, double \* b, double eps, double \* X, double\* dx, double\* r, int\* iter)

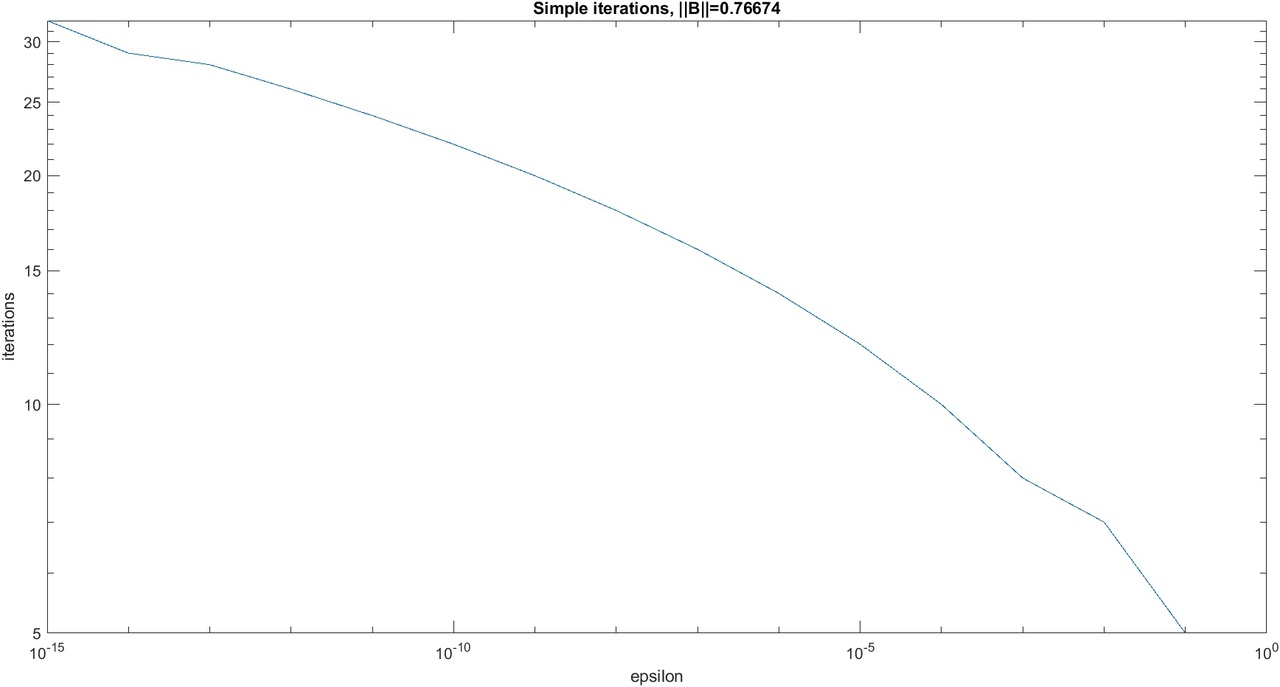
// выполняет решает СЛАУ методом Якоби

// принимает главную матрицу, вектор правой части, погрешность и вектор точного решения, а также контейнеры для записи получившейся фактической погрешности, невязки и кол-ва итераций

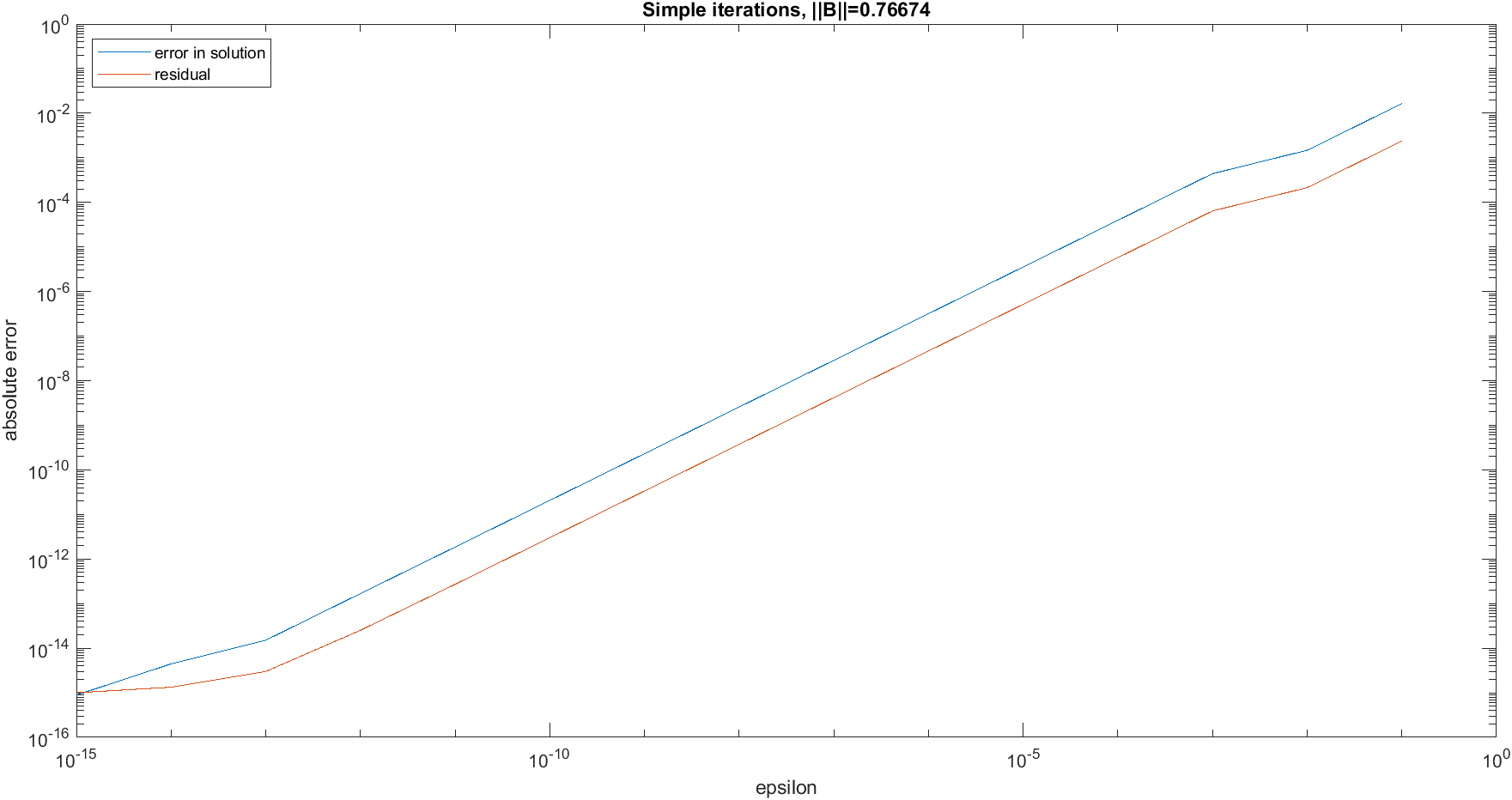
// ничего не возвращает

# Численный анализ решения

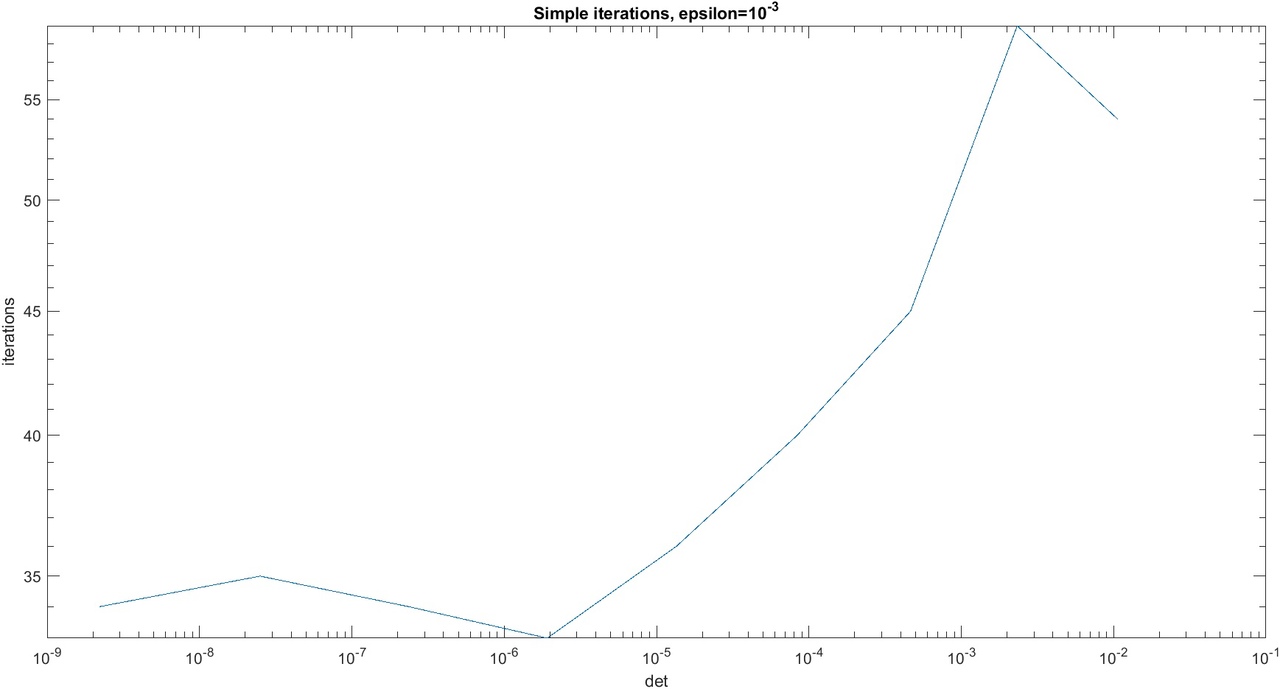
Ниже представлены графики, демонстрирующие зависимость количества итераций от заданной погрешности, нормы фактической погрешности в решении и невязки от заданной погрешности, количества итераций от определителя главной матрицы, а также нормы фактической погрешности в решении и невязки от определителя. Все нормы бесконечные.



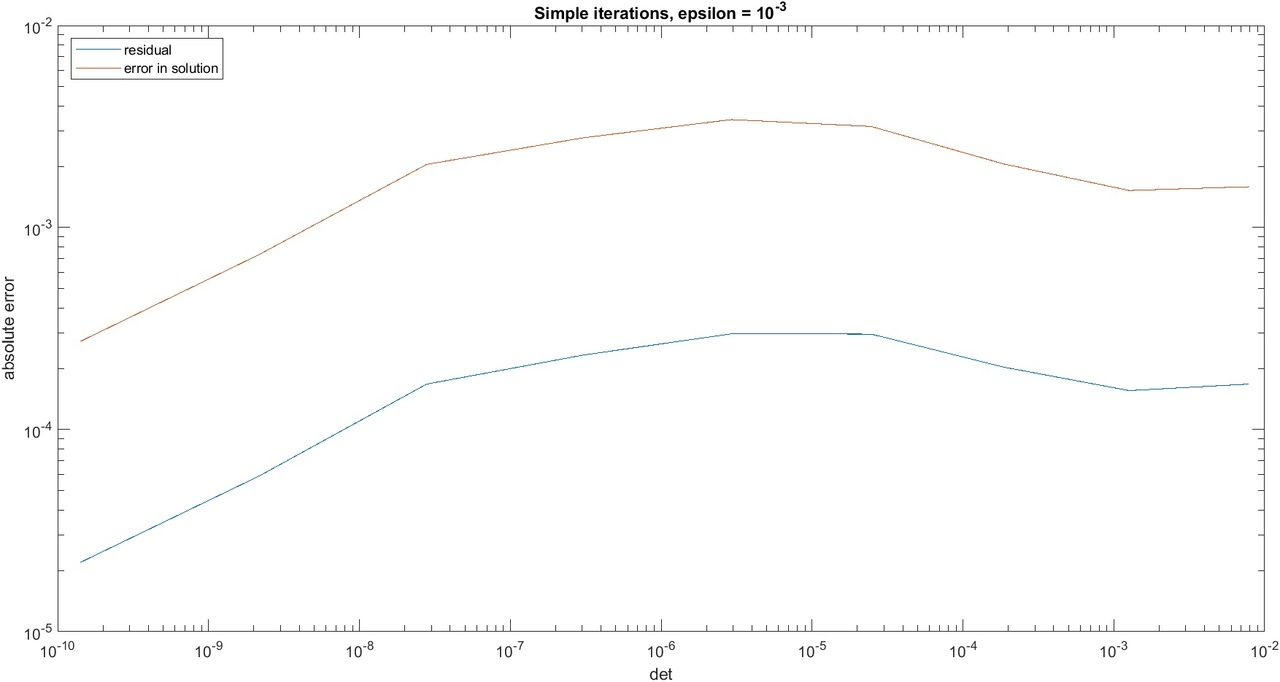
*Рис. 1,* *Количество итераций от погрешности*



*Рис. 2, Фактическая ошибка и невязка от погрешности*



*Рис. 3, Количество итераций от определителя*



*Рис. 4 Фактическая ошибка и невязка от определителя*

На графиках наблюдаем, что с ростом определителя и с уменьшением погрешности растет количество итераций. Невязка и фактическая ошибка в решении растут с увеличением определителя или погрешности. Также можно заметить, что невязка и фактическая ошибка растут одинаково и отличаются на одну и ту же постоянную величину.

# Вывод

По итогу проделанных измерений можно сделать следующие выводы:

* Метод простых итераций, конкретно – метод Якоби, достаточно прост в реализации, при относительно небольшом определителе сходится очень быстро, и может быть гораздо эффективнее точных методов в таком случае. Однако, на матрицу накладываются серьезные ограничения, для метода Якоби – в особенности.