

Санкт-Петербургский политехнический университет Петра Великого  
Институт прикладной математики и механики  
Высшая школа прикладной математики и вычислительной физики

# Итерационные методы

Отчёт по курсовой работе

МЕТОД СОПРЯЖЁННЫХ ГРАДИЕНТОВ И МЕТОД РИЧАРДСОНА

Работу выполнил:  
студентка I курса  
магистратуры  
Добрецова Е.В.  
Группа:  
5040102/40101  
Преподаватель:  
Елисеев А.А.

Санкт-Петербург  
2025

# Содержание

1. Постановка задачи	3
2. Аппроксимация задачи	3
2.1. Интегральная форма уравнения . . . . .	3
2.2. Аппроксимация потоков . . . . .	4
2.3. Дискретизированное уравнение . . . . .	4
2.4. Пример: сетка $N = M = 4$ . . . . .	4
3. Алгоритмы решения СЛАУ	5
3.1. Метод сопряжённых градиентов (CGM, conjugate gradient method) . . . . .	5
3.2. Метод Рундсона . . . . .	6
4. Численный эксперимент	6
4.1. Влияние числа итераций на точность . . . . .	6
4.2. Исследование устойчивости методов . . . . .	7
4.3. Сравнение скорости сходимости методов . . . . .	7
5. Выводы	7
6. Примечания	7

# 1. Постановка задачи

Необходимо взять эллиптическую краевую задачу с условиями первого рода на границе:

$$\begin{cases} \frac{\partial}{\partial x}(a(x, y) \frac{\partial u}{\partial x}) + \frac{\partial}{\partial y}(b(x, y) \frac{\partial u}{\partial y}) = -f(x, y) \\ \Omega = [0; 1] \times [0; 1] \\ u|_{\partial\Omega} \text{ задано} \end{cases}$$

Необходимо дискретизировать задачу с помощью метода конечных объёмов, сведя решение краевой задачи к решению СЛАУ, решить СЛАУ методом сопряжённых градиентов (CG) и методом Рундсона, сравнить эти методы по таким параметрам, как влияние числа итераций на точность решения, устойчивость, скорость сходимости.

Выберем коэффициенты задачи и точное решение следующим образом:

$$a(x, y) = 3,$$

$$b(x, y) = 2,$$

$$u(x, y) = \sin(3\pi x^2 y) + (y - 1)^3.$$

Таким образом, мы получим краевую задачу со следующей правой частью:

$$f = 18\pi(\pi x^2 \sin(3\pi x^2 y)(x^2 + 6y^2) - y \cos(3\pi x^2 y)) - 12(y - 1)$$

и известным решением.

## 2. Аппроксимация задачи

Для численного решения применяется метод конечных объёмов. Область  $\Omega$  разбивается равномерной сеткой с шагами  $h_x = h_y = h = 1/N$ . Контрольным объёмом является прямоугольная ячейка размером  $h \times h$ .

### 2.1. Интегральная форма уравнения

Интегрируем уравнение по контрольному объёму  $V_{i,j}$ :

$$\iint_{V_{i,j}} \left[ \frac{\partial}{\partial x} \left( a(x, y) \frac{\partial u}{\partial x} \right) + \frac{\partial}{\partial y} \left( b(x, y) \frac{\partial u}{\partial y} \right) \right] dx dy = - \iint_{V_{i,j}} f(x, y) dx dy.$$

Применяя теорему Гаусса–Остроградского (дивергентную теорему), преобразуем левую часть:

$$\iint_{V_{i,j}} \nabla \cdot \vec{F} dx dy = \oint_{\partial V_{i,j}} \vec{F} \cdot \vec{n} ds,$$

где

$$\vec{F} = \left( a(x, y) \frac{\partial u}{\partial x}, b(x, y) \frac{\partial u}{\partial y} \right).$$

Таким образом, получаем:

$$\oint_{\partial V_{i,j}} \vec{F} \cdot \vec{n} ds = - \iint_{V_{i,j}} f(x, y) dx dy.$$

## 2.2. Аппроксимация потоков

Для регулярной сетки (равномерные ячейки) поток через каждую грань аппроксимируется по центральной разности.

Пусть  $u_{i,j}$  — приближённое значение  $u$  в центре ячейки  $V_{i,j}$ .

$$\text{Восточная грань: } F^E = a_{i+\frac{1}{2},j} \cdot \frac{u_{i+1,j} - u_{i,j}}{h} \cdot h,$$

$$\text{Западная грань: } F^W = a_{i-\frac{1}{2},j} \cdot \frac{u_{i,j} - u_{i-1,j}}{h} \cdot h,$$

$$\text{Северная грань: } F^N = b_{i,j+\frac{1}{2}} \cdot \frac{u_{i,j+1} - u_{i,j}}{h} \cdot h,$$

$$\text{Южная грань: } F^S = b_{i,j-\frac{1}{2}} \cdot \frac{u_{i,j} - u_{i,j-1}}{h} \cdot h.$$

Суммарный поток:

$$F_{i,j} = F^E - F^W + F^N - F^S.$$

Правая часть:

$$\iint_{V_{i,j}} f(x, y) dx dy \approx f_{i,j} h^2.$$

## 2.3. Дискретизированное уравнение

$$\frac{a_{i+\frac{1}{2},j}(u_{i+1,j} - u_{i,j}) - a_{i-\frac{1}{2},j}(u_{i,j} - u_{i-1,j})}{h^2} + \frac{b_{i,j+\frac{1}{2}}(u_{i,j+1} - u_{i,j}) - b_{i,j-\frac{1}{2}}(u_{i,j} - u_{i,j-1})}{h^2} = -f_{i,j}.$$

Если  $a(x, y) = a = \text{const}$  и  $b(x, y) = b = \text{const}$ , уравнение упрощается:

$$\frac{a(u_{i+1,j} - 2u_{i,j} + u_{i-1,j})}{h^2} + \frac{b(u_{i,j+1} - 2u_{i,j} + u_{i,j-1})}{h^2} = -f_{i,j}.$$

## 2.4. Пример: сетка $N = M = 4$

При  $h = 0.25$ ,  $a = 3$ ,  $b = 2$ :

$$\frac{3(u_{i+1,j} - 2u_{i,j} + u_{i-1,j}) + 2(u_{i,j+1} - 2u_{i,j} + u_{i,j-1})}{0.0625} = -f_{i,j}.$$

Или:

$$-160u_{i,j} + 48u_{i+1,j} + 48u_{i-1,j} + 32u_{i,j+1} + 32u_{i,j-1} = -f_{i,j}.$$

Для внутренних узлов задаём локальную нумерацию и составляем СЛАУ  $Au = b$ , где:  
-  $u = [u_1, u_2, \dots, u_9]^T$  — значения в 9 внутренних узлах,  
-  $A$  — матрица с коэффициентами, соответствующими шаблону пятиточечного аппроксиматора,  
-  $b$  — вектор правой части с учётом граничных условий.

Внутренних узлов  $3 \times 3 = 9$ . Нумеруем их построчно снизу вверх и слева направо. Тогда:

$$u = [u_1, u_2, u_3, u_4, u_5, u_6, u_7, u_8, u_9]^T$$

Матрица  $A$  имеет структуру:

$$A = \begin{bmatrix} -160 & 48 & 0 & 32 & 0 & 0 & 0 & 0 & 0 \\ 48 & -160 & 48 & 0 & 32 & 0 & 0 & 0 & 0 \\ 0 & 48 & -160 & 0 & 0 & 32 & 0 & 0 & 0 \\ 32 & 0 & 0 & -160 & 48 & 0 & 32 & 0 & 0 \\ 0 & 32 & 0 & 48 & -160 & 48 & 0 & 32 & 0 \\ 0 & 0 & 32 & 0 & 48 & -160 & 0 & 0 & 32 \\ 0 & 0 & 0 & 32 & 0 & 0 & -160 & 48 & 0 \\ 0 & 0 & 0 & 0 & 32 & 0 & 48 & -160 & 48 \\ 0 & 0 & 0 & 0 & 0 & 32 & 0 & 48 & -160 \end{bmatrix}$$

Жирным шрифтом выделены ненулевые диагонали матрицы.

Теперь составим вектор правой части  $b$ , подставляя значения функции  $f(x, y)$  в узлах.

Для примера, пусть точное решение:

$$u(x, y) = \sin(3\pi x^2 y) + (y - 1)^3,$$

и правая часть:

$$f(x, y) = 18\pi (\pi x^2 \sin(3\pi x^2 y)(x^2 + 6y^2) - y \cos(3\pi x^2 y)) - 12(y - 1).$$

Вычислим значения  $f_{i,j}$  в каждом внутреннем узле  $k = 1, \dots, 9$  по координатам из таблицы, и получим:

$$b_k = -f(x_k, y_k) + \text{вклад граничных узлов (если есть)}.$$

Например, для  $u_2$ :

$$b_2 = -f(0.5, 0.25) + 32 \cdot u(x = 0.5, y = 0) = -f_{2,1} + 32 \cdot (-1).$$

Аналогично считаем  $b_1, b_3, \dots, b_9$ .

Таким образом, мы получили СЛАУ:

$$Au = b,$$

где  $A$  — разреженная симметричная матрица,  $b$  учитывает правую часть и граничные условия.

### 3. Алгоритмы решения СЛАУ

#### 3.1. Метод сопряжённых градиентов (CGM, conjugate gradient method)

Данный метод представляет собой итерационный метод решения СЛАУ  $Ax = b$  с симметричной положительно определённой матрицей. Алгоритм CGM выглядит следующим образом: пусть  $x_0$  - начальное приближение,  $r_0 = Ax_0 - b$  - начальная невязка,  $p_0 = r_0$  - первое направление. Пока  $\|r_k\| > \varepsilon$ :

$$\alpha_{k-1} = \frac{(r_{k-1}, r_{k-1})}{(Ap_{k-1}, p_{k-1})} \text{ - шаг вдоль направления } p_{k-1};$$

$$v_k = v_{k-1} + \alpha_{k-1} p_{k-1} \text{ - обновлённое приближение;}$$

$$r_k = r_{k-1} - \alpha_{k-1} Ap_{k-1} \text{ - обновлённая невязка;}$$

$$\beta_{k-1} = \frac{(r_k, r_k)}{(r_{k-1}, r_{k-1})} \text{ - коэффициент сопряжённости;}$$

$$p_k = r_k + \beta_{k-1} p_{k-1} \text{ - новое направление.}$$

Accuracy on iterations number for different grids

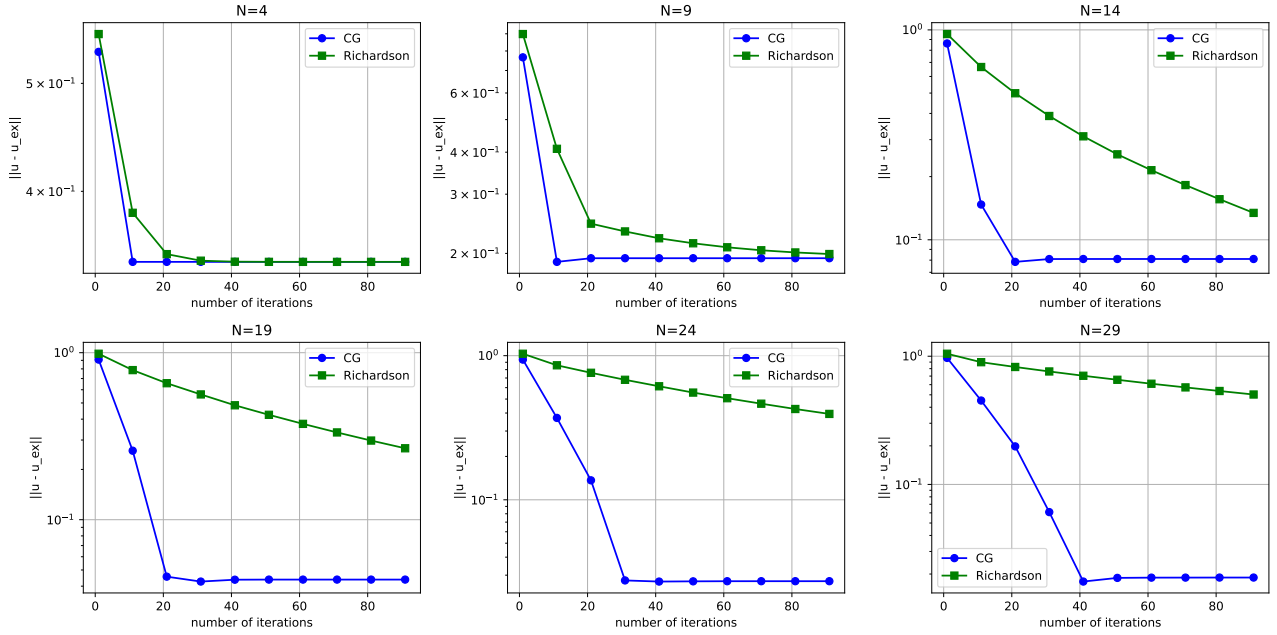


Рисунок 4.1. Зависимость точности решения от номера итерации

### 3.2. Метод Ричардсона

Данный метод представляет собой простой итерационный метод для численного решения СЛАУ вида

$$Ax = b,$$

где  $A$  - заданная матрица, не обязательно симметричная.

Алгоритм метода Ричардсона выглядит следующим образом: выбирается  $x_0$  - начальное приближение, затем вычисляется начальная невязка  $r_0 = Ax_0 - b$ . Затем, до тех пор, пока  $\|r_k\| > \varepsilon$ ,

$$x_{k+1} = x_k - \tau r_k,$$

где  $r_k = Ax_k - b$  - невязка,  $\tau$  - параметр Ричардсона,

$$r_{k+1} = Ax_{k+1} - b.$$

## 4. Численный эксперимент

При проведении всех численных экспериментов будем разбивать область на  $N = 5, 10, 15, 25, 30$  узлов.

### 4.1. Влияние числа итераций на точность

Будем строить график нормы ошибки (разности точного решения задачи и численного) от количества итераций. Он показан на [рисунке 4.1](#). Графики показывают, что ошибка имеет порядок  $10^{-1}$ , это объясняется тем, что дискретизация задачи даёт погрешность  $O(h^2)$ , а решение, полученное итерационным методом, стремится к численному решению дискретной задачи, а не к аналитическому решению. Можно видеть, что метод сопряжённых градиентов быстро (при малом числе узлов за 30-40 итераций) приходит к пределу точности, в то время как ошибка метода Ричардсона убывает гораздо медленнее.

## 4.2. Исследование устойчивости методов

Будем вносить малое (с относительной амплитудой, равной  $10^{-3}$  -  $10^{-8}$ ) возмущение в матрицу левой части и будем исследовать, как будет меняться число итераций, необходимое для достижения заданной точности, в зависимости от внесённого возмущения. График количества итераций от возмущения при различном числе узлов показан на [рисунке 4.2](#).

По графику видно, что в методе сопряжённых градиентов число итераций умеренно растёт по мере роста возмущения, что является абсолютно предсказуемым поведением метода: шум меняет спектр матрицы, увеличивая число её обусловленности, поэтому количество итераций растёт. Метод Рундсона, в свою очередь, движется к решению равномерными маленькими шагами, скорость сходимости определяется выбранным значением параметра  $\tau$ , поэтому кривая данного графика для этого метода практически плоская.

## 4.3. Сравнение скорости сходимости методов

Отложим по оси абсцисс номер итерации, по оси ординат – норму вектора текущего направления (т.е. критерий останова алгоритма). Также отметим горизонтальной линией принятую точность решения ( $\varepsilon = 10^{-5}$ ). Построенный график можно видеть на [рисунке 4.3](#). Он показывает, что даже при разреженных сетках, т.е. сетках с малым числом узлов, метод сопряжённых градиентов требует для сходимости количества итераций, меньшего на порядок, чем количество итераций, нужное для сходимости метода Рундсона.

Теперь проведём сравнение времени, затраченного на работу алгоритмов. Количество узлов будем рассматривать от 5 до 55. График можно видеть на [рисунке 4.4](#). По графику видно, что хотя разница во времени выполнения и наблюдается в пользу метода сопряжённых градиентов, она незначительна. Это объясняется временными затратами на выполнение операций Python в реализации метода сопряжённых градиентов (главным образом операции `solve.deercore()`).

## 5. Выводы

В ходе выполнения данной работы была составлена краевая задача с известным решением, далее она была сведена к СЛАУ с помощью дискретизации методом конечных объёмов и решена двумя итерационными методами решения СЛАУ: методом сопряжённых градиентов и методом Рундсона. Затем было проведено сравнение эффективности этих методов. Результаты эксперимента показали, что оба метода дают достаточно большую ошибку по сравнению с аналитическим решением исходной задачи вследствие погрешности дискретизации. Метод сопряжённых градиентов сходится намного быстрее метода Рундсона, но он сильнее зависит от шума, внесённого в матрицу. С другой стороны, разница во времени работы методов не соотносится с разницей в числе итераций, что объясняется временными затратами на библиотечные операции языка Python.

## 6. Примечания

Код реализации обоих методов и численных экспериментов доступен по [ссылке](#).

Number of iterations on disturbance amplitude

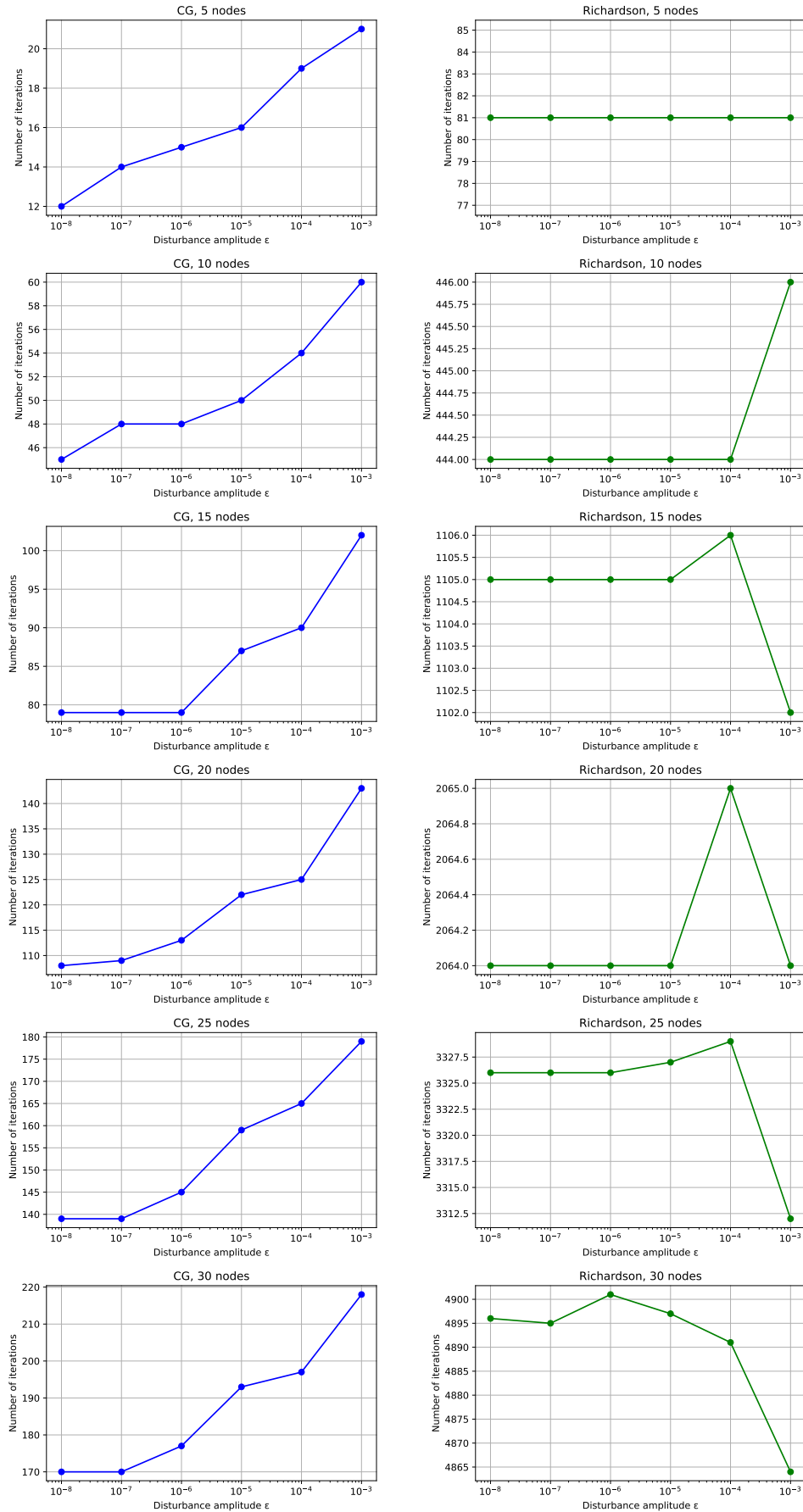


Рисунок 4.2. Зависимость количества итераций от возмущения матрицы левой части



Residual on iteration number,  $\varepsilon = 10^{-5}$

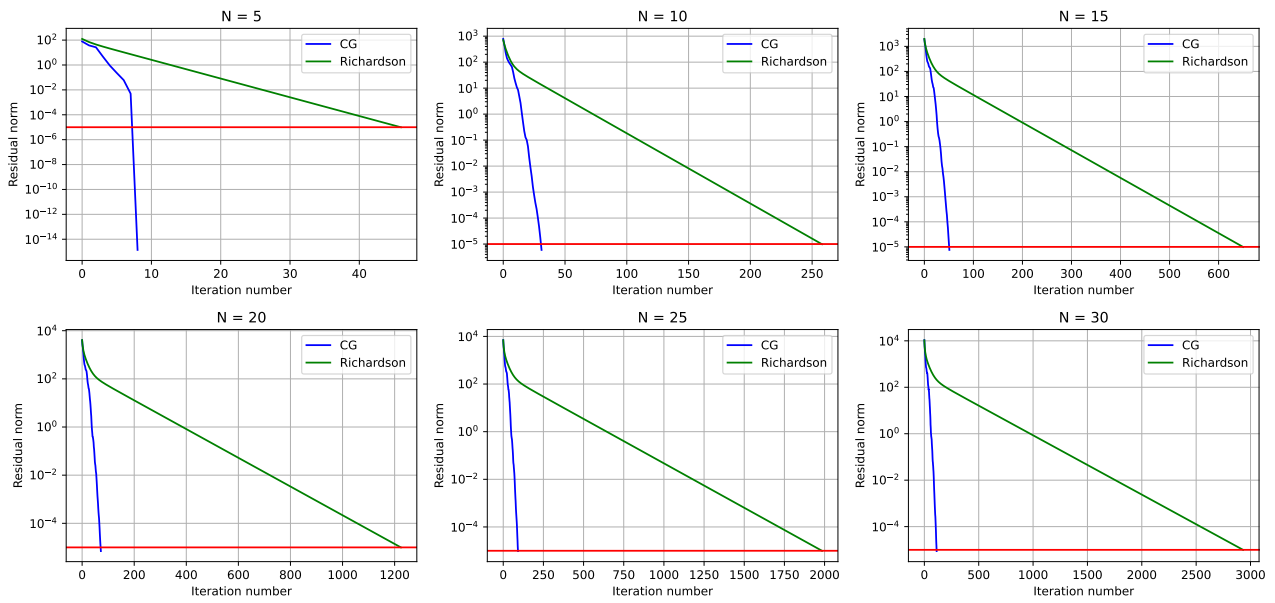


Рисунок 4.3. Зависимость нормы невязки от номера итерации

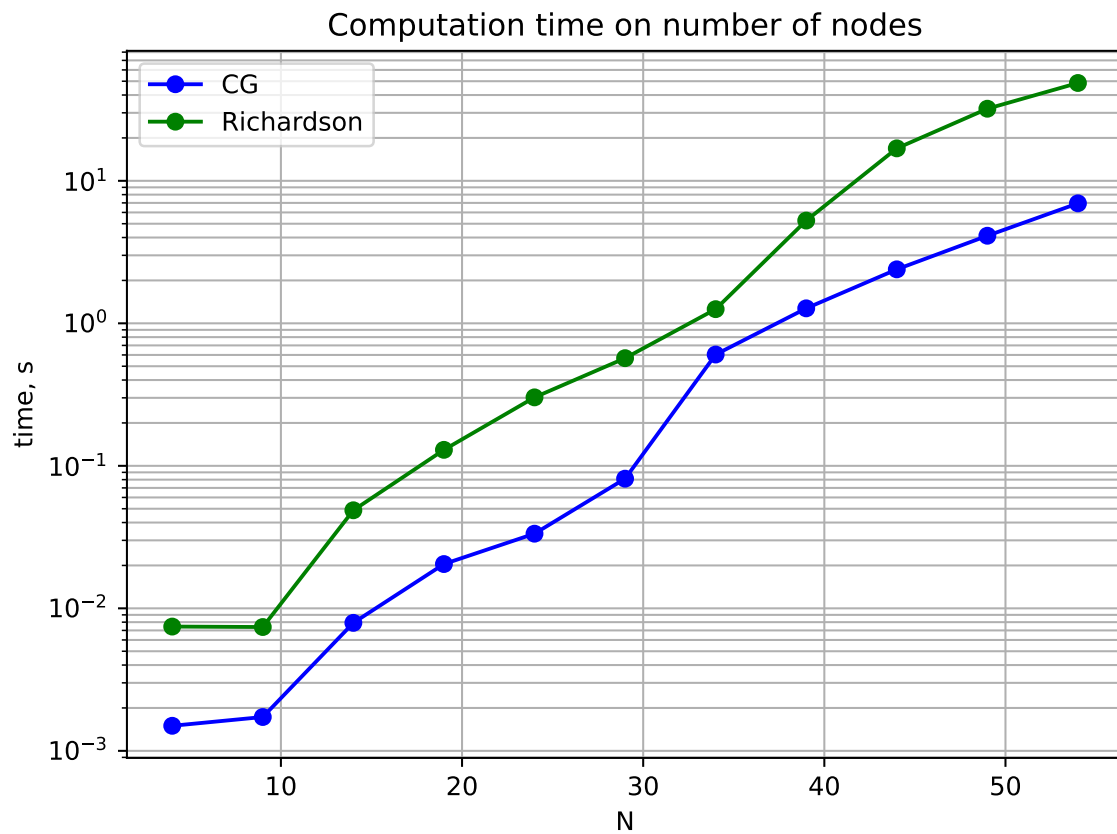


Рисунок 4.4. Зависимость времени работы алгоритма от количества узлов