Численные методы
Лекция 2

Е.А. Яревский

11 ноября 2020

Прямые и итерационные методы решения СЛАУ

Решаем систему

$$Ax = b.$$

1. Прямые методы (конечное число арифметических операций):

$$x = A^{-1}b.$$

Достоинства:

– гарантированный результат.

– известное время вычислений, независимое от значений элементов матрицы.

Недостатки:

– большие требования к памяти и времени.

Прямые и итерационные методы решения СЛАУ

2. Итерационные методы: строим последовательность

$$x^{(0)}, x^{(1)}, x^{(2)}, \ \ldots, \ \text{таких, что} \ \lim_{i\to\infty} x^{(i)} = x.$$

Достоинства:

– малые требования к памяти.

Недостатки:

– возможные проблемы со сходимостью (медленная или её отсутствие).

– зависимость вычислений от значений матрицы и правой части.

Метод Гаусса

Решаем систему $Ax = b$:

$$\begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1N} \\ a_{21} & a_{22} & \cdots & a_{2N} \\ \cdots & \cdots & \cdots & \cdots \\ a_{N1} & a_{N2} & \cdots & a_{NN} \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ \cdots \\ x_N \end{pmatrix} = \begin{pmatrix} b_1 \\ b_2 \\ \cdots \\ b_N \end{pmatrix}.$$

Это – первый шаг метода: $A = A_1$, $b = b_1$.
На втором шаге исключаем x_1: к каждой i-ой строке матрицы добавляем первую, умноженную на $c_{i1} = -a_{i1}/a_{11}$ (то же делаем с вектором).
Получаем систему $A_2 x = b_2$:

$$\begin{pmatrix} a_{11}^{(1)} & a_{12}^{(1)} & \cdots & a_{1N}^{(1)} \\ 0 & a_{22}^{(2)} & \cdots & a_{2N}^{(2)} \\ \cdots & \cdots & \cdots & \cdots \\ 0 & a_{N2}^{(2)} & \cdots & a_{NN}^{(2)} \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ \cdots \\ x_N \end{pmatrix} = \begin{pmatrix} b_1^{(1)} \\ b_2^{(2)} \\ \cdots \\ b_N^{(2)} \end{pmatrix},$$

где $a_{ij}^{(2)} = a_{ij}^{(1)} + c_{i1} a_{1j}^{(1)}$, $b_i^{(2)} = b_i^{(1)} + c_{i1} b_1^{(1)}$, $i \geq 2$.
Далее, к каждой i-ой строке новой матрицы, $i \geq 3$, добавляем новую вторую, умноженную на $c_{i2} = -a_{i2}^{(2)} / a_{22}^{(2)}$.
Получаем систему $A_3 x = b_3$:

$$\begin{pmatrix} a_{11}^{(1)} & a_{12}^{(1)} & a_{13}^{(1)} & \cdots & a_{1N}^{(1)} \\ 0 & a_{22}^{(2)} & a_{23}^{(2)} & \cdots & a_{2N}^{(2)} \\ 0 & 0 & a_{33}^{(3)} & \cdots & a_{3N}^{(3)} \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & 0 & a_{N3}^{(3)} & \cdots & a_{NN}^{(3)} \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ \cdots \\ x_N \end{pmatrix} = \begin{pmatrix} b_1^{(1)} \\ b_2^{(2)} \\ b_3^{(3)} \\ \cdots \\ b_N^{(3)} \end{pmatrix}.$$

На $(k+1)$ шаге:
$a_{ij}^{(k+1)} = a_{ij}^{(k)} + c_{ik} a_{kj}^{(k)}$, $b_i^{(k+1)} = b_i^{(k)} + c_{ik} b_k^{(k)}$,
$c_{ik} = -a_{ik}^{(k)} / a_{kk}^{(k)}$, $i, j > k$.

На $(N-1)$ шаге получаем верхнетреугольную систему:

$$\begin{pmatrix} a_{11}^{(1)} & a_{12}^{(1)} & a_{13}^{(1)} & \cdots & \cdots & a_{1N}^{(1)} \\ 0 & a_{22}^{(2)} & a_{23}^{(2)} & \cdots & \cdots & a_{2N}^{(2)} \\ 0 & 0 & a_{33}^{(3)} & \cdots & \cdots & a_{3N}^{(3)} \\ 0 & 0 & 0 & a_{44}^{(4)} & \cdots & a_{4N}^{(4)} \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & 0 & 0 & \cdots & 0 & a_{NN}^{(N)} \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ \cdots \\ x_N \end{pmatrix} = \begin{pmatrix} b_1^{(1)} \\ b_2^{(2)} \\ b_3^{(3)} \\ b_4^{(4)} \\ \cdots \\ b_N^{(N)} \end{pmatrix}.$$

Треугольная система $A_N x = b_N$ легко решается: $(U \equiv A_N,\ f \equiv b_N)$

$$x_N = \frac{f_N}{U_{NN}}, \quad x_k = \frac{1}{U_{kk}} \left(f_k - \sum_{i=k+1}^{N} U_{ki} x_i \right), \quad k = N-1, \ldots, 1.$$

Выбор главного элемента

При прямом ходе метода Гаусса может возникнуть деление на нуль (или очень маленький элемент).

Частичный выбор главного элемента: после каждого шага, k, переставляют строки k, $k+1$, $\ldots$, N так, чтобы на месте kk оказался наибольший по модулю элемент из всех $a_{mk}^{(k)}$ при $m > k$. Переставляются и компоненты вектора b.

Метод Гаусса с выбором главного элемента: на каждом шаге переставляются и строки, и столбцы, так чтобы на месте kk оказался наибольший по модулю элемент из всех $a_{mr}^{(k)}$, $m, r > k$. (Нужно ещё переставлять и элементы x).

Вычислительная сложность метода Гаусса:

прямой ход: $\sim \frac{2}{3}N^3$ операций,
обратный ход: $\sim N^2$ операций.
Существуют методы с меньшей асимптотической сложностью (напр. метод Штрассена), но их устойчивость существенно хуже.

Достоинства метода Гаусса:

– простота и устойчивость алгоритма
– возможность находить ранг и определитель матрицы
– возможность эффективно решать СЛАУ для нескольких правых частей

Недостатки метода Гаусса:

– правые части должны быть заранее известны
– не самое эффективное использование памяти

LU-разложение

Пусть мы смогли представить матрицу в виде:

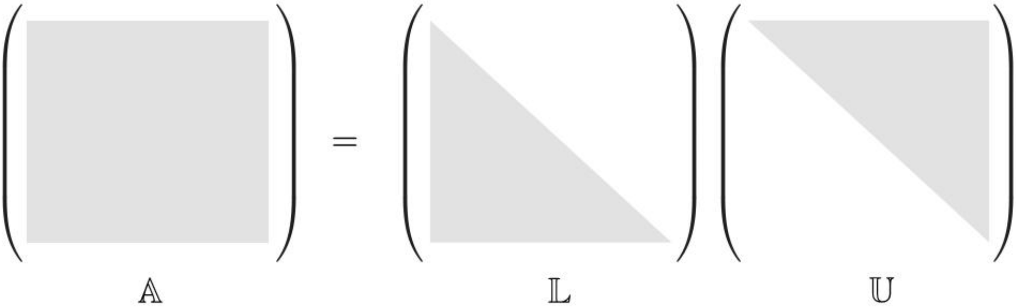
$$A = LU,$$

где

$$L = \begin{pmatrix} L_{11} & 0 & \cdots & 0 \\ L_{21} & L_{22} & \cdots & 0 \\ \vdots & & \ddots & \vdots \\ L_{N1} & L_{N2} & \cdots & L_{NN} \end{pmatrix}, \quad U = \begin{pmatrix} 1 & U_{12} & \cdots & U_{1N} \\ 0 & 1 & \cdots & U_{2N} \\ \vdots & & \ddots & \vdots \\ 0 & 0 & \cdots & 1 \end{pmatrix}$$

Тогда решение $Ax = b$ получается решением двух треугольных СЛАУ:

$$Ly = b, \quad Ux = y.$$



$$y_1 = \frac{b_1}{L_{11}}, \quad y_i = \frac{1}{L_{ii}}\left(b_i - \sum_{j=1}^{i-1} L_{ij} y_j\right), \quad i = 2 \ldots N,$$

$$x_N = y_N, \; x_i = y_i - \sum_{j=i+1}^{N} U_{ij} x_j, \quad i = N-1 \ldots 1.$$

Как найти требуемое разложение?

Шаги алгоритма Гаусса:

1) Деление i-го уравнения на $a_{ii}^{(i-1)}$ – равносильно умножению слева на матрицу

$$C_i = \begin{pmatrix} 1 & & & & \\ & \ddots & & & \\ & & (a_{ii}^{(i-1)})^{-1} & & \\ & & & \ddots & \\ & & & & 1 \end{pmatrix}.$$

2) Вычитание i-го уравнения, умноженного на $a_{ik}^{(i-1)}$, из следующих. Равносильно умножению слева на

$$C_i' = \begin{pmatrix} 1 & & & & \\ & \ddots & & & \\ & & 1 & & \\ & & -a_{i+1,i}^{(i-1)} & 1 & \\ & & \vdots & & \ddots & \\ & & -a_{N,i}^{(i-1)} & & & 1 \end{pmatrix}.$$

Весь прямой ход метода Гаусса:

$$CAx = Cb, \quad C = C_N \dots C_1' C_1.$$

C – левая треугольная, как произведение левых треугольных.
Обратная $L = C^{-1}$ – тоже левая треугольная.
Обозначим $CA = U$. По построению, все $U_{ii} = 1$, и матрица U – правая треугольная.
Тогда

$$A = C^{-1} U = LU.$$

Это равенство и $U_{ii} = 1$ дают систему уравнений: т.к. $L_{ij} = 0$ при $i < j$ и $U_{jk} = 0$ при $k < j$, то

$$\sum_{j=1}^{\min(i,k)} L_{ij} U_{jk} = a_{ik}.$$

Иначе:

$$\sum_{j=1}^{k} L_{ij} U_{jk} = a_{ik} \text{ при } k \le i, \quad \sum_{j=1}^{i} L_{ij} U_{jk} = a_{ik} \text{ при } i < k.$$

Тогда для элементов находим:

$$L_{ik} = a_{ik} - \sum_{j=1}^{k-1} L_{ij} U_{jk} \text{ при } k \le i,$$

$$U_{ik} = \frac{1}{L_{ii}} \left(a_{ik} - \sum_{j=1}^{i-1} L_{ij} U_{jk} \right) \text{ при } i < k.$$

Вычисляется последовательно для
$(i, k) = (1, 1), \ldots, (1, N), (2, 1), \ldots, (2, N), \ldots, (N, 1), \ldots, (N, N).$

Кол-во операций для факторизации $\sim N^3$.
Кол-во операций для решения СЛАУ $\sim N^2$.
Вначале делается факторизация, СЛАУ могут решаться позже, по мере появления правых частей.

$LDL^\top$-разложение (тройная факторизация):

$$A = LDU,$$

D – диагональная матрица.
Если A-симметричная, то

$$A = LDL^\top.$$

Ленточные матрицы

Ленточная матрица: ширина ленты

$$\beta(A) = \max\{|i-j| : a_{ij} \neq 0\}.$$

Лента:

$$Band(A) = \{\{i,j\} : 0 \le |i-j| \le \beta(A)\}.$$

Хранение: по столбцам в прямоугольном массиве с размерами: $N \times (2\beta(A)+1)$.
Число операций для разложения матрицы: $\sim \beta^2 N$.
Число операций для решения системы: $\sim \beta N$.

Трёхдиагональные матрицы: метод прогонки

$$A = \begin{pmatrix} c_1 & -b_1 & 0 & \dots & 0 & 0 \\ -a_2 & c_2 & -b_2 & \dots & 0 & 0 \\ 0 & -a_3 & c_3 & -b_3 & 0 & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & 0 & \dots & -a_N & c_N \end{pmatrix}.$$

Для решения $At = s$ применяется метод прогонки.
Система записывается в виде:

$$-t_{k-1}a_k + t_k c_k - t_{k+1}b_k = s_k, \quad k = 1, 2, \dots, N.$$

Выразим t_i через следующий t_{i+1}:
Из первого уравнения:

$$t_1 = \frac{b_1}{c_1} t_2 + \frac{s_1}{c_1},$$

из второго

$$t_2 = \frac{b_2 t_3}{c_2 - \frac{b_1}{c_1} a_2} + \frac{s_2 + \frac{a_2}{c_1} s_1}{c_2 - \frac{b_1}{c_1} a_2}.$$

Определяем закономерность (подставляем $t_{k-1} = \alpha_{k-1} t_k + \beta_{k-1}$):

$$t_k = \alpha_k t_{k+1} + \beta_k,$$

$$\alpha_k = \frac{b_k}{c_k - \alpha_{k-1} a_k}, \quad \beta_k = \frac{s_k + \beta_{k-1} a_k}{c_k - \alpha_{k-1} a_k}.$$

Доказываем по индукции (база + шаг индукции).

Применение метода:
1) Прямой ход: определение α_k, β_k через значения матрицы, вектора и предыдущие значения.
2) Обратный ход: определение решения t_k:

$$t_N = \beta_N, \quad t_k = \alpha_k t_{k+1} + \beta_k.$$

Достаточное условие разрешимости прогонки:

Если $|c_k| > |b_k| + |a_k|\ \forall k$, то $\det A \neq 0$.

Доказательство:

Единственная возможная проблема – деление на нуль в процессе прямого хода.

Если $|\alpha_k| < 1$, то этого не происходит:

$$|c_k - \alpha_{k-1} a_k| \geq |c_k| - |\alpha_{k-1}||a_k| > |c_k| - |a_k| > |b_k| \geq 0.$$

Оценим α_k по индукции:

$$|\alpha_1| = |\frac{b_1}{c_1}| < 1, \quad |\alpha_k| = \frac{|b_k|}{|c_k - \alpha_{k-1} a_k|} < \frac{|b_k|}{|b_k|} = 1.$$

Упражнение 2: Решение СЛАУ

Рассмотрим краевую задачу для ОДУ:

$$-u''(x) + u(x) = x, \quad 0 \le x \le 1, \quad u(0) = u(1) = 0.$$

Постройте его аналитическое решение, которое можно будет использовать для проверки правильности вычислений.

Для численного решения этого уравнения может использоваться метод конечно-разностной дискретизации. Разделим интервал $[0, 1]$ на $N + 1$ отрезок одинаковой длины $h = 1/(N + 1)$. В каждой точке $x_i = ih$, $i = 1, 2, \ldots N$, построим конечно-разностную аппроксимацию второй производной:

$$-u''(x_i) \approx -h^{-2}[u(x_{i-1}) - 2u(x_i) + u(x_{i+1})].$$

Тогда, с учётом граничных условий, для вектора, состоящего из неизвестных, $\vec{u} = (u(x_1), u(x_2), \ldots, u(x_N))^\top$, получим СЛАУ

$$A\vec{u} = \vec{b},$$

где $\vec{b} = (h, 2h, \dots, Nh)^\top$. Трехдиагональная матрица A размерности $N \times N$ содержит на субдиагоналях одинаковые значения $-h^{-2}$, а её диагональные элементы равны $2h^{-2} + 1$.

1) Сформируйте квадратную матрицу A и вектор $\vec{b}$, найдите решение $\vec{u}$ с помощью встроенной функции для решения СЛАУ. Постройте решение, сравните его с точным. Используя встроенные функции для измерения времени, постройте график зависимости времени решения СЛАУ от её размерности N.

2) Реализуйте метод прогонки для решения построенной системы. Убедитесь, что решение построено правильно. Постройте график зависимости времени работы метода прогонки от N, сравните его с предыдущим.

Литература

1. Бахвалов Н.С., Жидков Н.П., Кобельков Г.М., *Численные методы*, М.: Бином. Лаборатория знаний, 2012. - 636 с. Параграфы 6.1, 6.3, 6.6, 6.7, 6.11.
2. Faires J.D., Burden R.L. *Numerical methods* (3ed., Thomson Brooks Cole, 2003). Глава 7.