

Итерационные методы (ИМ) решения СЛАУ

Основная идея ИМ решения СЛАУ состоит в построении последовательности векторов $x^0, x^1, \dots, x^k, \dots$, сходящейся к решению x системы $Ax = b$. Пусть $\|\cdot\|$ заданная норма вектора.

Определение 1. Последовательность векторов $\{x^k\}_{k=0}^{\infty}$ называется сходящейся к вектору x , если $\|x - x^k\| \rightarrow 0$ при $k \rightarrow \infty$.

За приближенное решение СЛАУ принимается вектор x^k при достаточно большом k . В качестве критерия окончания ИМ обычно принимают либо достаточную близость двух соседних приближений, либо достаточную малость вектора невязки, т.е. итерации заканчиваются при выполнении одного из условий (или обоих):

$$\|x^k - x^{k-1}\| \leq \varepsilon \quad \|Ax^k - b\| \leq \varepsilon. \quad (0.1)$$

Вектор x^k называют k -тым приближением к решению или k -той итерацией, k — номером итерации. ИМ (iteration — повтор, повторение) состоит в циклическом выполнении одной и той же группы операций, скажем \mathcal{F} . Двуслойные ИМ имеет вид

$$x^{k+1} = \mathcal{F}(x^k), \quad k = 0, 1, \dots \quad (0.2)$$

Каждое следующее приближение вычисляется лишь по предыдущему и для начала счета необходимо знать (задать) лишь одно начальное приближение x^0 к решению СЛАУ. ИМ вида $x^{k+1} = \mathcal{F}(x^k, x^{k-1})$, $k = 1, 2, \dots$, называются трехслойными ИМ и требуют двух начальных приближений x^0 и x^1 . ИМ конструируются так, чтобы начальные приближения можно было задавать произвольно.

Возникает естественный вопрос: зачем нужны ИМ, если имеются прямые методы, позволяющие найти решение СЛАУ за конечное число операций? Можно привести по крайней мере две причины, когда ИМ будут полезны.

- 1) При реализации прямых методов важно, чтобы исходные и промежуточные данные располагались в оперативной (быстрой) памяти компьютера. Если порядок системы настолько велик, что оперативной памяти для реализации метода недостаточно или число операций метода недопустимо велико, то для таких систем предпочтительнее оказываются ИМ.
- 2) На практике часто встречается ситуация, когда достаточно знать не точное, а приближенное решение СЛАУ, причем начальное приближение к решению имеется. В этом случае ИМ может оказаться предпочтительнее прямого метода, если небольшое число итераций позволит уточнить это приближение и получить приближение с необходимой точностью.

1. Простейшие итерационные методы.

Всюду в дальнейшем через z^k будем обозначать вектор $x - x^k$, где x — решение системы

$$Ax = b, \quad (1.1)$$

т. е. погрешность приближения с номером k . Далее используем представление матрицы A в виде суммы трех матриц:

$$A = L + D + U, \quad D = \text{diag}(a_{11}, a_{22}, \dots, a_{nn}), \quad (1.2)$$

L (U) — нижняя (верхняя) треугольная матрица, поддиагональные (наддиагональные) элементы которой совпадают с соответствующими элементами матрицы A , а все диагональные элементы равны нулю.

1. Метод Якоби¹⁾. Будем считать, что все диагональные элементы матрицы A отличны от нуля. Перепишем систему (1.1) в виде

$$Dx = b - Lx - Ux \quad \Rightarrow \quad x = D^{-1}(b - Lx - Ux). \quad (1.3)$$

Итерационный метод определим по формуле

$$x^{k+1} = D^{-1}(b - Lx^k - Ux^k), \quad k = 0, 1, \dots, \quad (1.4)$$

где начальное приближение $x^0 = (x_1^0, x_2^0, \dots, x_n^0)$ — произвольно задано. Компоненты приближения x^{k+1} определяются по уже найденному вектору x^k при помощи соотношений:

$$x_i^{k+1} = \left(b_i - \sum_{j=1}^{i-1} a_{ij} x_j^k - \sum_{j=i+1}^n a_{ij} x_j^k \right) / a_{ii}, \quad i = 1, 2, \dots, n. \quad (1.5)$$

Формулы (1.4) или (1.5) определяют итерационный метод решения системы (1.1), называемый *методом Якоби*.

Укажем легко проверяемое достаточное условие сходимости этого метода. Напомним, что для матрицы A выполнено условие диагонального преобладания по строкам, если

$$q = \max_{1 \leq i \leq n} \sum_{j=1, j \neq i}^n \frac{|a_{ij}|}{|a_{ii}|} < 1. \quad (1.6)$$

Теорема 1. Пусть матрица A системы (1.1) — матрица с диагональным преобладанием по строкам. Тогда итерационный метод Якоби сходится при любом начальном приближении x^0 ; справедлива следующая оценка скорости сходимости:

$$\|z^k\|_{\infty} \leq q^k \|z^0\|_{\infty}. \quad (1.7)$$

¹⁾Карл Густав Якоб Якоби (Carl Gustav Jacob Jacobi; 1804 — 1851) — немецкий математик.

ДОКАЗАТЕЛЬСТВО. Пусть x — решение системы уравнений (1.1). Вычитая из второго равенства (1.3) равенство (1.4), получим $z^{k+1} = -D^{-1}(Lz^k + Uz^k)$ или

$$z_i^{k+1} = -\sum_{j=1}^{i-1} \frac{a_{ij}}{a_{ii}} z_j^k - \sum_{j=i+1}^n \frac{a_{ij}}{a_{ii}} z_j^k, \quad i = 1, 2, \dots, n.$$

Следовательно, для $i = 1, 2, \dots, n$:

$$\begin{aligned} |z_i^{k+1}| &\leq \sum_{j=1}^{i-1} \frac{|a_{ij}|}{|a_{ii}|} |z_j^k| + \sum_{j=i+1}^n \frac{|a_{ij}|}{|a_{ii}|} |z_j^k| \leq \\ &\leq \left(\sum_{j=1}^{i-1} \frac{|a_{ij}|}{|a_{ii}|} + \sum_{j=i+1}^n \frac{|a_{ij}|}{|a_{ii}|} \right) \max_{1 \leq j \leq n} |z_j^k| = q \max_{1 \leq j \leq n} |z_j^k|, \end{aligned}$$

откуда вытекает, что

$$\|z^{k+1}\|_{\infty} \leq q \|z^k\|_{\infty}$$

для любого $k = 0, 1, \dots$, поэтому

$$\|z^k\|_{\infty} \leq q^k \|z^0\|_{\infty} \rightarrow 0$$

при $k \rightarrow \infty$, поскольку $0 < q < 1$, а это и означает, что $x^k \rightarrow x$. \square

Оценка (1.7) показывает, что, чем меньше q , т. е. чем выше диагональное преобладание матрицы A , тем быстрее сходится метод Якоби.

2. Метод Зейделя. Формулы (1.5) допускают естественную модификацию. Именно, при вычислении x_i^{k+1} будем использовать уже найденные компоненты вектора x^{k+1} , т. е. $x_1^{k+1}, x_2^{k+1}, \dots, x_{i-1}^{k+1}$. В результате приходим к итерационному *методу Зейделя*¹⁾:

$$x_i^{k+1} = \left(b_i - \sum_{j=1}^{i-1} a_{ij} x_j^{k+1} - \sum_{j=i+1}^n a_{ij} x_j^k \right) / a_{ii}, \quad i = 1, 2, \dots, n. \quad (1.8)$$

В матричных обозначениях он запишется в виде (проверьте!):

$$x^{k+1} = D^{-1}(b - Lx^{k+1} - Ux^k), \quad k = 0, 1, \dots \quad (1.9)$$

Метод Зейделя позволяет более экономно расходовать память компьютера, поскольку в данном случае вновь получаемые компоненты вектора x^{k+1} можно размещать на месте соответствующих компонент вектора x^k , в то время как при реализации метода Якоби все компоненты векторов x^k, x^{k+1} должны одновременно находиться в памяти компьютера.

¹⁾Филипп Людвиг Зейдель (Philipp Ludwig von Seidel; 1821 — 1896) — немецкий математик и астроном.

Теорема 2. Пусть матрица A — матрица с диагональным преобладанием по строкам. Тогда метод Зейделя сходится при любом начальном приближении x^0 и справедлива оценка

$$\|z_j^k\|_\infty \leq q^k \|z^0\|_\infty, \quad (1.10)$$

где q определяется (1.6).

ДОКАЗАТЕЛЬСТВО. Аналогично методу Якоби имеем

$$z_i^{k+1} = - \sum_{j=1}^{i-1} \frac{a_{ij}}{a_{ii}} z_j^{k+1} - \sum_{j=i+1}^n \frac{a_{ij}}{a_{ii}} z_j^k, \quad i = 1, 2, \dots, n. \quad (1.11)$$

Пусть $\max_{1 \leq j \leq n} |z_j^{k+1}| = |z_l^{k+1}|$. Из l -того уравнения (1.11) следует

$$|z_l^{k+1}| \leq \alpha_l \max_{1 \leq j \leq n} |z_j^{k+1}| + \beta_l \max_{1 \leq j \leq n} |z_j^k|,$$

где

$$\alpha_l = \sum_{j=1}^{l-1} \frac{|a_{lj}|}{|a_{ll}|}, \quad \beta_l = \sum_{j=l+1}^n \frac{|a_{lj}|}{|a_{ll}|},$$

следовательно,

$$\|z^{k+1}\|_\infty \leq \frac{\beta_l}{1 - \alpha_l} \|z^k\|_\infty.$$

Из условия (1.6) получаем, что $\alpha_l + \beta_l \leq q < 1$, но тогда и $q\alpha_l + \beta_l \leq q$, таким образом, $\beta_l/(1 - \alpha_l) \leq q$, поэтому $\|z^{k+1}\|_\infty \leq q \max \|z^k\|_\infty$ для любого $k \geq 0$. Дальнейшие рассуждения совпадают с соответствующими рассуждениями из доказательства предыдущей теоремы. \square

3. Метод релаксации. Зачастую существенного ускорения сходимости можно добиться за счет введения в расчетные формулы числового параметра. В качестве примера приведем ИМ

$$x_i^{k+1} = (1 - \omega) x_i^k + \omega \left(b_i - \sum_{j=1}^{i-1} a_{ij} x_j^{k+1} - \sum_{j=i+1}^n a_{ij} x_j^k \right) / a_{ii}, \quad (1.12)$$

$i = 1, 2, \dots, n$, $k = 0, 1, \dots$. Этот метод называется *методом релаксации*, число ω — *релаксационным параметром*. При $\omega = 1$ метод переходит в метод Зейделя. В матричных обозначениях получаем

$$x^{k+1} = (1 - \omega) x^k + \omega D^{-1}(b - Lx^{k+1} - Ux^k), \quad k = 0, 1, \dots \quad (1.13)$$

Ясно, что по затратам памяти и объему вычислений на каждом шаге итераций метод релаксации не отличается от метода Зейделя.