

ДИСЦИПЛИНА	Вычислительная математика (полное наименование дисциплины без сокращений)
ИНСТИТУТ	информационных технологий
КАФЕДРА	прикладной математики (полное наименование кафедры)
ВИД УЧЕБНОГО МАТЕРИАЛА	Материалы для лекционных занятий (в соответствии с пп.1-11)
ПРЕПОДАВАТЕЛЬ	Митин Михаил Петрович, Самохин Александр Борисович, Аристов Анатолий Игоревич, Держинский Роман Игоревич, Ледовская Екатерина Валерьевна (фамилия, имя, отчество)
СЕМЕСТР	3, 2023-2024 (указать семестр обучения, учебный год)

Лекция №5-6

5. Численные методы линейной алгебры

Рассматриваются численные методы решения систем линейных алгебраических уравнений (СЛАУ), а также нахождения собственных значений и собственных векторов матриц.

5.1. Численное решение СЛАУ

СЛАУ используются во многих областях науки и техники и являются наиболее часто встречающимся типом задач вычислительной математики. В общем виде СЛАУ из n уравнений с n неизвестными записывается в виде:

$$A\vec{x} = \vec{b} \quad (5.1)$$

Здесь \vec{x} - неизвестный вектор решения, \vec{b} - заданный вектор в n -мерном пространстве, а

$$A = \left(\begin{array}{cc|c} a_{1,1} & a_{1,2} & a_{1,n} \\ a_{2,1} & a_{2,2} & a_{2,n} \\ \hline a_{n,1} & a_{n,2} & a_{n,n} \end{array} \right) -$$

линейный оператор в этом пространстве, заданная матрица размером $n \times n$ или в другом виде $A = \{a_{i,j}\}, i, j = 1, 2, \dots, n$.

Доказывается, что если определитель матрицы не равен нулю, то СЛАУ имеет единственное решение. Ниже будем полагать, что это условие выполняется. Однако, отличие определителя A от нуля не могут служить гарантией того, что решение СЛАУ будет найдено численно с заданной точностью. Причи-

ной этого может быть как плохая обусловленность самой системы, так и выбранного алгоритма. Заметим, что близость определителя к нулю и даже весьма малое его значение не свидетельствуют, вообще говоря, о плохой обусловленности системы. В качестве примера можно привести матрицу системы, у которой присутствует только главная диагональ с весьма малыми, но отличными от нуля коэффициентами. Определитель такой матрицы может быть машинный нуль, в тоже время свойства такой матрицы близки к единичной, а ошибка в решении порядка ошибки в задании исходных данных.

Для, так называемых, плохо обусловленных задач их решение принципиально нельзя получить совершенно точно. Для них малые изменения в исходных данных (коэффициентах матрицы и в векторе правой части), которые могут находиться в пределах точности их задания, приводят к несоразмерно большим изменениям в решении. В результате, в пределах точности задания исходных данных (например, в пределах ошибки округления из-за ограниченного формата числовых данных ЭВМ) может существовать множество различных решений, удовлетворяющих системе.

В качестве примера плохо обусловленной системы можно привести СЛАУ с почти линейно зависимыми строками (столбцами) в матрице. Плохо обусловленным алгоритмом для решения СЛАУ можно назвать метод Гаусса без выбора главного элемента.

Для характеристики обусловленности задачи вводят, так называемое, число обусловленности K . Для задачи решения СЛАУ в качестве числа обусловленности можно принять

$$K = \|A\| \cdot \|A^{-1}\|.$$

Здесь $\|\cdot\|$ - какая-либо норма в пространстве n -мерных векторов, которая выражается через норму вектора следующим образом:

$$\|A\| = \max_{\|\vec{x}\| \neq 0} \frac{\|A \cdot \vec{x}\|}{\|\vec{x}\|} = \max_{\|\vec{x}\|=1} \|A \cdot \vec{x}\|$$

Норма матрицы характеризует максимально возможное относительное увеличение по норме ненулевого вектора при воздействии на него матрицы.

Пусть решение \vec{x} СЛАУ получено с относительной ошибкой $\delta\vec{x}$. Тогда для нее справедлива оценка:

$$\|\delta\vec{x}\| < \approx K \varepsilon_{\text{маш.}}$$

Здесь $\varepsilon_{\text{маш.}}$ - машинная константа – наименьшее число, которое при прибавлении к единице ещё изменяет её значение в машинном представлении. Отметим,

что оценка справедлива для малых ошибок в заданной матрице

$$K \|\Delta A\| / \|A\| < 1.$$

Введём понятие невязки \vec{h} решения:

$$\vec{h} = A \cdot \vec{x} - \vec{b} \quad (5.2)$$

Заметим, что малость невязки $\|\vec{h}\| < \varepsilon \|\vec{x}\|$, $\varepsilon \ll 1$ не гарантирует малость ошибки $\Delta\vec{x}$ в решении. Так, для невязки выполняется соотношение

$$\|\vec{h}\| < \approx \|A\| \|\vec{x}\| \varepsilon_{\text{маш}},$$

в то время как для $\Delta\vec{x}$ справедливо:

$$\|\Delta\vec{x}\| < \approx \|A^{-1}\| \|\vec{h}\|$$

Норма обратной матрицы для плохо обусловленной СЛАУ велика, также как и число обусловленности K , характеризующее в этом случае близость матрицы к вырожденной (сингулярной), для которой $\|A^{-1}\| \rightarrow \infty$.

Существуют два основных класса методов для решения СЛАУ – прямые и итерационные. Прямые методы характеризуются тем, что при абсолютной точности вычислений (на гипотетической бесконечноразрядной ЭВМ) точное решение СЛАУ может быть получено с помощью конечного числа арифметических операций. Итерационные методы характеризуются тем, что даже при абсолютной точности вычислений за конечное число арифметических операций может быть получено лишь приближенное решение системы, хотя возможно и как угодно близкое к точному значению. Однако при реальных вычислениях на ЭВМ указанное различие теряет свой смысл, и для многих задач итерационные методы оказываются более предпочтительными, чем прямые в силу отсутствия накопления ошибок для сходящегося процесса и возможности приблизиться к решению с заданной точностью.

Рассмотрим сначала прямые методы. Наиболее известным является метод Гаусса, поскольку другие методы являются, как правило, его модификацией.

5.2. Прямые методы решения СЛАУ

Количество операций для решения системы $\sim n^3$. Матрица A либо невыгодно обращается, либо представляется в виде произведения матриц удобных для обращения.

В первом случае матрица A последовательно преобразуется с помощью элементарных (эквивалентных) преобразований:

1. Перестановка столбцов и строк.
2. Умножение столбцов и строк на число.
3. Прибавление к строке (столбцу) другой строки, умноженной на число.

Каждое элементарное преобразование можно представить в виде матрицы L_i , в результате последовательного умножения A на L_i , она преобразуется в единичную матрицу:

$$L_n \dots L_2 L_1 A \cdot \vec{x} = L_n \dots L_1 \vec{b}$$

5.2.1. Метод Гаусса (Метод исключений)

Формально, метод Гаусса основан на последовательном применении матриц

$$L_i = \begin{pmatrix} 1 & 0 & 0 \\ 0 & l_{ii} & 0 \\ 0 & l_{ij} & 0 \\ 0 & l_{ni} & 0 \end{pmatrix}; \quad l_{ii} = \frac{1}{a_{ii}}; \quad l_{ij} = -\frac{a_{ij}}{a_{ii}} \quad (5.2.1.1)$$

Пример для матрицы (3×3) :

$$\begin{pmatrix} a_{11}^{-1} & 0 & 0 \\ -\frac{a_{21}}{a_{11}} & 1 & 0 \\ -\frac{a_{31}}{a_{11}} & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{pmatrix} =$$

Действие матрицы L_i преобразуют элементы i -го столбца матрицы A ниже диагонали в нулевые (т.е. исключают их).

В каждом уравнении выделяется ведущий элемент, на который производится деление; пусть это будет a_{11} . Делим первое уравнение на a_{11} :

Все остальные элементы преобразуются по схеме:

$$\begin{cases} x_1 + c_{12}x_2 + \dots + c_{1n}x_n = g_1 \\ a_{22}^{(1)}x_2 + \dots + a_{2n}^{(1)}x_n = b_2^{(1)} \\ \dots\dots\dots \\ a_{n2}^{(1)}x_2 + \dots + a_{nn}^{(1)}x_n = b_n^{(1)} \end{cases}$$

На втором шаге ведущим элементом выбирается $a_{22}^{(1)}$, на него делится вторая строка, а все остальные элементы преобразуются по формуле:

$$a_{ij}^{(2)} = a_{ij}^{(1)} - a_{i2}^{(1)} \cdot \frac{a_{2j}^{(1)}}{a_{22}^{(1)}}; b_i^{(2)} = b_i^{(1)} - a_{i2}^{(1)} \cdot \frac{b_2^{(1)}}{a_{22}^{(1)}};$$

$$\begin{pmatrix} 1 & c_{12} & \dots & c_{1n} \\ 0 & 1 & \dots & c_{2n} \\ \dots & \dots & \dots & \dots \\ 0 & 0 & \dots & 1 \end{pmatrix} \cdot \begin{pmatrix} x_1 \\ x_2 \\ \dots \\ x_n \end{pmatrix} = \begin{pmatrix} g_1 \\ g_2 \\ \dots \\ g_n \end{pmatrix}$$

Преобразование к верхней треугольной матрице называется прямым ходом.

Далее следует обратный ход: начиная с x_n , последовательно вычисляются компоненты вектора:

$$x_n = g_n; \quad x_{n-1} = g_{n-1} - c_{n-1,n}x_n;$$

$$x_k = g_k - \sum_{i=k+1}^n c_{ki}x_i, \quad k = n, n-1, \dots, 1. \quad (5.2.2.3)$$

В машинных расчетах в качестве ведущего элемента обычно выбирается максимальный элемент i -го столбца с $j > i$ или строки a_{ij} с $i > j$.

Эта строка (или столбец) переставляются на место i -ой строки (столбца). Такой выбор уменьшает ошибки округления. При ручных расчетах элементы матрицы записываются вместе с элементами вектора b в расширенную матрицу:

$$\left(\begin{array}{ccc|c} a_{11} & a_{12} & a_{13} & b_1 \\ a_{21} & a_{22} & a_{23} & b_2 \\ a_{31} & a_{32} & a_{33} & b_3 \end{array} \right) \quad \text{далее}$$

из соображений удобства выбирают ведущий элемент, а преобразование остальных элементов на одном шаге прямого хода метода Гаусса проводят по *правилу прямоугольника*. В матрице выделяется прямоугольник, на главной диагонали которого расположены ведущий и преобразуемый элементы.

Пусть a_{ii} - ведущий элемент, тогда

$$a_{kt}^{(1)} = a_{kt} - \frac{a_{ki}a_{it}}{a_{ii}}. \quad (5.2.2.4)$$

Из преобразуемого элемента вычитается произведение элементов, стоящих на побочной диагонали, деленное на ведущий элемент.

5.2.3. Ортогонализация матриц

Матрица называется ортогональной, если $A \cdot A^T = D$, D - диагональная матрица, т.е. в ней отличны от нуля только диагональные элементы, если $A \cdot A^T = E$, то A - ортонормированная матрица. Любая неособенная матрица A может быть представлена в виде: $A = R \cdot T$, R - ортогональная, а T – верхняя треугольная матрица с единичной диагональю.

Рассмотрим матрицу A , как набор вектор – столбцов \vec{a}_i , $A = [\vec{a}_1 | \vec{a}_2 | \dots | \vec{a}_n]$ - вектора \vec{a}_i - линейно независимы, т.к. $\det A \neq 0$. Выберем первый столбец матрицы R - \vec{r}_1 , равным \vec{a}_1 ; $\vec{r}_i = \vec{a}_1$.

Запишем $\vec{a}_2 = t_{12}\vec{r}_1 + \vec{r}_2$, условие ортогональности R $(\vec{r}_1, \vec{r}_2) = 0$ позволяет получить t_{12} :

$$(\vec{r}_1, \vec{a}_2) = t_{12} \cdot (\vec{r}_1, \vec{r}_1), t_{12} = \frac{(\vec{r}_1, \vec{a}_2)}{(\vec{r}_1, \vec{r}_1)},$$

следовательно, известен и вектор $\vec{r}_2 = \vec{a}_2 - t_{12}\vec{r}_1$. Аналогичным образом представляется и $\vec{a}_3 = t_{13} \cdot \vec{r}_1 + t_{23} \cdot \vec{r}_2 + \vec{r}_3$, где

$$t_{13} = \frac{(\vec{r}_1, \vec{a}_3)}{(\vec{r}_1, \vec{r}_1)}, t_{23} = \frac{(\vec{r}_2, \vec{a}_3)}{(\vec{r}_2, \vec{r}_2)}.$$

В общем случае получим выражения:

$$\vec{r}_k = \vec{a}_k - \sum_{i=1}^{k-1} t_{ik} \cdot \vec{r}_i, t_{ik} = \frac{(\vec{r}_i, \vec{a}_k)}{(\vec{r}_i, \vec{r}_i)}. \quad (5.2.3.1)$$

Покажем, что t_{ik} - элементы матрицы T . Действительно:

$$\vec{a}_1 = \vec{r}_1$$

$$\vec{a}_2 = t_{12} \cdot \vec{r}_1 + \vec{r}_2$$

, или иначе:

$$\vec{a}_3 = t_{13} \cdot \vec{r}_1 + t_{23} \cdot \vec{r}_2 + \vec{r}_3$$

.....

$$\begin{bmatrix} a_{11} & a_{12} & a_{13} & \dots \\ a_{21} & a_{22} & a_{23} & \dots \\ a_{31} & a_{32} & a_{33} & \dots \\ \dots & \dots & \dots & \dots \end{bmatrix} = \begin{bmatrix} r_{11} & r_{12} & r_{13} & \dots \\ r_{21} & r_{22} & r_{23} & \dots \\ r_{31} & r_{32} & r_{33} & \dots \\ \dots & \dots & \dots & \dots \end{bmatrix} \cdot \begin{bmatrix} 1 & t_{12} & t_{13} & \dots \\ 0 & 1 & t_{23} & \dots \\ 0 & 0 & 1 & \dots \\ \dots & \dots & \dots & \dots \end{bmatrix}$$

5.2.4. Решение системы уравнений методом ортогонализации

Оптимальной является следующая схема, основанная на свойствах вектора \vec{r} . Запишем систему $A \cdot \vec{x} = \vec{b}$ в виде: $\vec{a}_1 \cdot x_1 + \vec{a}_2 \cdot x_2 + \dots + \vec{a}_n \cdot x_n = \vec{b}$. Из структуры векторов \vec{r} следует, что $(\vec{r}_i, \vec{a}_j) = 0, (i > j)$.

Умножаем систему слева на \vec{r}_n : $\vec{r}_n \cdot (\vec{a}_1 \cdot x_1 + \vec{a}_2 \cdot x_2 + \dots + \vec{a}_n \cdot x_n) = \vec{r}_n \cdot \vec{b}$,
в уравнении остается одно слагаемое: $(\vec{r}_n \cdot \vec{a}_n) \cdot x_n = (\vec{r}_n \cdot \vec{b})$.

$$x_n = \frac{(\vec{r}_n, \vec{b})}{(\vec{r}_n, \vec{a}_n)};$$

$$\vec{a}_1 \cdot x_1 + \dots + \vec{a}_{n-1} \cdot x_{n-1} = \vec{b} - \vec{a}_n \frac{(\vec{r}_n \cdot \vec{b})}{(\vec{r}_n \cdot \vec{a}_n)} = \vec{b}^{(1)}$$

Полученную систему умножим на \vec{r}_{n-1} , находим x_{n-1} и вычисляем $\vec{b}^{(2)}$ и т.д.

$$x_i = \frac{(\vec{r}_i, \vec{b}^{(n-i)})}{(\vec{r}_i, \vec{a}_i)}; \quad \vec{b}^{(i)} = \vec{b}^{(i-1)} - x_{n-i+1} \cdot \vec{a}_{n-i+1}. \quad (5.2.4.1)$$

5.3. Итерационные методы решения СЛАУ

5.3.1. Метод простой итерации

Многие итерационные методы могут быть сведены к процессу простой итерации. При этом исходное уравнение тем или иным способом должно быть сведено к уравнению

$$\vec{x} = B\vec{x} + \vec{b} \quad (5.3.1.1)$$

Здесь \vec{x} - неизвестный вектор, \vec{b} - заданный вектор правой части, B - заданная матрица коэффициентов (оператор). Например, если задана СЛАУ (5.1), то непосредственно принимая

$$B = I - A, \quad (5.3.1.2)$$

где I - единичная матрица, приходим к (5.3.1.1).

Процесс простой итерации строится следующим образом:

$$\bar{x}^{(k+1)} = B\bar{x}^{(k)} + \bar{b}, \quad k=0,1,2,\dots \quad (5.3.1.3)$$

В качестве начального приближения $\bar{x}^{(0)}$ можно принять $\bar{x}^{(0)} = \bar{b}$.

Заметим, что переход от (5.1) к (5.3.1.1) может быть выполнен не единственным способом, что приводит к различным модификациям метода простой итерации. Так, метод (5.3.1.3) с преобразованием (5.3.1.2) известен в литературе как метод Ричардсона. Другие методы простой итерации будут рассмотрены в разделе 5.3.2.

Процесс простой итерации может быть эквивалентно записан также в виде ряда по степеням оператора B , т.е., в виде, так называемого, ряда Неймана

$$\bar{x} = \bar{b} + B\bar{b} + B^2\bar{b} + \dots = \sum_{i=0}^{\infty} B^i \bar{b} \quad (5.3.1.4)$$

Если матрица B постоянна (не зависит от номера итерации k), то такой итерационный процесс называется стационарным.

Пусть \bar{x}^* - «гипотетическое» точное решение, строго удовлетворяющее $\bar{x}^* = B\bar{x}^* + \bar{b}$, а $\Delta\bar{x}^{(k)} = \bar{x}^{(k)} - \bar{x}^*$ - ошибка на k -м шаге. Подставляя в формулу простой итерации получаем для соотношения ошибок на $k+1$ и k -м шаге $\Delta\bar{x}^{(k+1)} = B\Delta\bar{x}^{(k)}$. Для нормы ошибки:

$$\|\Delta\bar{x}^{(k+1)}\| \leq \|B\| \|\Delta\bar{x}^{(k)}\| \leq \|B\|^k \|\Delta\bar{x}^{(1)}\|.$$

Отсюда следует достаточное условие сходимости процесса простой итерации: $\|B\| = \delta, \delta < 1$.

Действительно, тогда

$$\|\Delta\bar{x}^{(k+1)}\| \leq \|B\|^k \|\Delta\bar{x}^{(1)}\| = \delta^k \|\Delta\bar{x}^{(1)}\| < \|\Delta\bar{x}^{(1)}\|$$

Оператор с $\|B\| = \delta < 1$ называется сжимающим, а процесс (5.3.1.2), (5.3.1.3) для него сходящимся, т.к. ошибка убывает с каждым шагом, независимо от её начальной величины.

Спектральным радиусом матрицы (конечномерного оператора) B называется $\rho(B) = \max_i |\beta_i|$, где β_i - собственные числа оператора B (см. 5.4).

Для любой нормы справедливо соотношение $\rho(B) \leq \|B\|$

Доказывается, что необходимым и достаточным условием сходимости процесса простой итерации (5.3.1.3) является

$$\rho(B) < 1, \quad (5.3.1.5)$$

при этом итерации сходятся не хуже геометрической прогрессии со знаменателем $q = \rho(B)$.

Условие (5.3.1.5) является, как правило, сильным ограничением при непосредственном применении метода (5.3.1.2), (5.3.1.3) к заданной СЛАУ. Выбор нового оператора \tilde{B} с другим спектром при эквивалентности исходной системе (5.1) может значительно расширить область сходимости процесса простой итерации с его участием:

$$\bar{x}^{(k+1)} = \tilde{B}\bar{x}^{(k)} + \tilde{b}, \quad k=0,1,2,\dots \quad (5.3.1.6)$$

В качестве условия выхода из вычислительного процесса по достижении заданной точности решения ε , аналогично (3.5.1), можно принять:

$$|\bar{x}^{(k+1)} - \bar{x}^{(k)}| \leq \frac{1-q}{q} \varepsilon, \text{ где } q - \text{спектральный радиус } B \text{ или какая-либо оценка}$$

другой нормы B .

5.3.2. Метод Якоби и метод Зейделя

Исторически одними из самых ранних итерационных методов являются метод Якоби и метод Зейделя, которые могут быть представлены в виде модификации метода простой итерации. Перепишем (5.1) в следующем виде

$$\begin{cases} x_1 = \frac{1}{a_{11}}(b_1 - a_{12}x_2 - a_{13}x_3 - \dots - a_{1n}x_n) \\ \dots\dots\dots \\ x_i = \frac{1}{a_{ii}}(b_i - a_{i1}x_1 - \dots - a_{ii-1}x_{i-1} - a_{ii+1}x_{i+1} - \dots - a_{in}x_n) \\ \dots\dots\dots \\ x_n = \frac{1}{a_{nn}}(b_n - a_{n1}x_1 - a_{n2}x_2 - \dots - a_{nn-1}x_{n-1}) \end{cases} \quad (5.3.2.1)$$

Используем (5.3.2.1) для построения процесса итераций, начиная с $\vec{x}^{(0)} = \vec{b}$ при $k=0$, $k=0,1,2,\dots$:

$$x_i^{(k+1)} = \frac{1}{a_{i,i}} (b_i - a_{i,1} x_1^{(k)} - \dots - a_{i,i-1} x_{i-1}^{(k)} - a_{i,i+1} x_{i+1}^{(k)} - \dots - a_{i,n} x_n^{(k)})$$

$$i=1,2,\dots,n. \quad (5.3.2.2)$$

В матричных обозначениях метод Якоби можно записать следующим образом. Представим $C = D - A$, где D - диагональная матрица, $\{D\}_{i,i} = \{A\}_{i,i}, i=1,2,\dots,n, \{D\}_{i,j} = 0, i \neq j$. C - матрица с нулевой главной диагональю. Тогда справедлива запись уравнения аналогично (5.3.1.6), где

$$\tilde{B} = D^{-1}C, \quad \tilde{b} = D^{-1}\vec{b} \quad (5.3.2.3)$$

Матрица D^{-1} - диагональная и $\{D^{-1}\}_{i,i} = 1/\{A\}_{i,i}, i=1,2,\dots,n$. Необходимые и достаточные условия сходимости метода Якоби

$$\rho(D^{-1}(D-A)) < 1$$

Другой известный метод простой итерации для случая, когда \tilde{B} строится на основе матрицы с нулевой главной диагональю - это метод Зейделя. Он отличается от метода Якоби тем, что при расчете координат вектора $\vec{x}^{(k+1)}$ на текущей $k+1$ -й итерации используются не только координаты вектора на предыдущей k -й итерации $\vec{x}^{(k)}$, но и уже ранее найденные на текущей итерации координаты вектора $\vec{x}^{(k+1)}$:

$$x_i^{(k+1)} = \frac{1}{a_{i,i}} (b_i - a_{i,1} x_1^{(k+1)} - \dots - a_{i,i-1} x_{i-1}^{(k+1)} - a_{i,i+1} x_{i+1}^{(k)} - \dots - a_{i,n} x_n^{(k)})$$

$$i=1,2,\dots,n. \quad k=0,1,2,\dots \quad (5.3.2.4)$$

В матричных обозначениях это соответствует представлению исходной матрицы A как $A = L + D + U$, где L - нижняя треугольная матрица, D -

диагональная матрица, $\{D\}_{i,i} = \{A\}_{i,i}, i=1,2,\dots,n$ и U - верхняя треугольная матрица.

В отличие от метода Якоби действие оператора \tilde{B} на вектор предыдущей итерации разделяется здесь на две части:

$$\tilde{B}\vec{x}^{(k)} = -D^{-1}L\vec{x}^{(k+1)} - D^{-1}U\vec{x}^{(k)} \quad (5.3.2.5)$$

и процесс его воздействия (но не результат!) нельзя свести к воздействию какой-либо матрицы на вектор предыдущей итерации.

Метод Зейделя хорошо алгоритмизируется. Если известна скорость сходимости метода, нет необходимости хранить оба вектора $\vec{x}^{(k+1)}$ и $\vec{x}^{(k)}$.

Достаточными условиями сходимости методов Якоби и Зейделя является диагональное преобладание в матричных элементах:

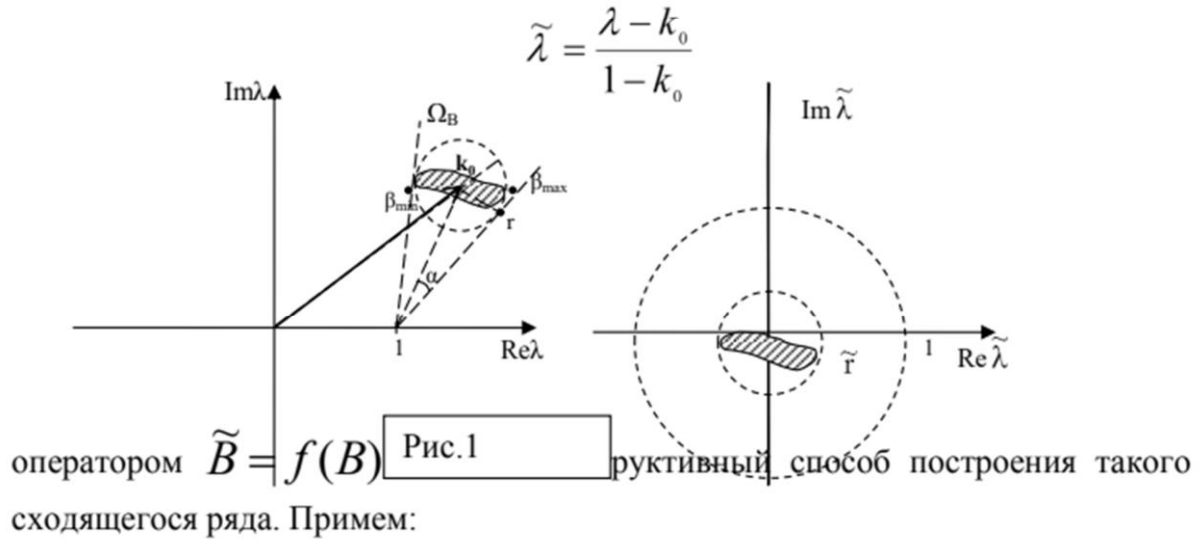
$$q|a_{i,i}| \geq \sum_{j \neq i} |a_{i,j}|, \quad q < 1, \quad \text{для всех } i=1,2,\dots,n,$$

однако на практике область сходимости значительно шире и определяется условием (5.3.1.5) на спектральный радиус матрицы (5.3.2.3) для метода Якоби и оператора (5.3.2.5) для метода Зейделя. Для решения СЛАУ с ленточными матрицами метод Зейделя является превосходным инструментом. Так, для симметричных положительно определенных матриц он будет всегда сходящимся. Однако возможно улучшение сходимости как метода Зейделя, так и любого другого метода простой итерации с помощью изложенного ниже метода оптимального спектрального параметра.

5.3.3. Метод оптимального спектрального параметра (ОСП) для простой итерации

Рассмотрим случай, когда спектр оператора B выходит за границы единичного круга на комплексной β -плоскости собственных чисел. В этом случае ряд простой итерации (5.3.1.3) расходится.

Определим выпуклую оболочку спектра оператора B как выпуклую замкнутую кривую наименьшей меры, полностью охватывающую спектр оператора на β -плоскости. Доказывается, что если точка 1 находится вне выпуклой оболочки спектра, то можно построить сходящийся ряд простой итерации с



$$\tilde{B} = \frac{B - kI}{1 - k}, \quad \tilde{b} = \frac{\vec{b}}{1 - k},$$

(5.3.3.1)

где k - комплексный параметр. При $k \neq 1$ исходные уравнения (5.3.1.1) с операторами B и \tilde{B} эквивалентны. Выбором k попробуем добиться сходимости ряда (5.3.1.6).

Пусть Ω_B - один из множества кругов радиуса r , полностью охватывающих спектр оператора B , и пусть при этом точка $1 \notin \Omega_B$ (Рис.1). Очевидно, что Ω_B включает в себя выпуклую оболочку спектра. Вектор из начала $\beta = 0$ в центр этого круга обозначим \vec{k}_0 . При дробно-линейном преобразовании (5.3.3.1) с $k = k_0$ круг Ω_B переходит в круг $\Omega_{\tilde{B}}$ с центром в точке $\beta = 0$ и радиусом $|\tilde{r}| = \left| \frac{r}{1 - k_0} \right|$. Если $|\tilde{r}| < 1$, то ряд (5.3.1.6) сходится.

Найдем минимум значения $|\tilde{r}|$. Пусть круг Ω_B «виден» из точки 1 под углом 2α . Пусть \vec{r} – вектор из центра круга \vec{k}_0 в точку касания луча из т.1 и круга. Из рис. 1 очевидно, что $|\tilde{r}| = \left| \frac{r}{1 - \vec{k}_0} \right| = \sin(\alpha)$ и, следовательно,

$$|\tilde{r}| < 1.$$

Таким образом, если Ω_B такой круг, что точка $1 \notin \Omega_B$ и «видимый» из точки 1 под наименьшим углом 2α , то комплексное расстояние до центра этого круга есть оптимальный параметр для сходимости (5.3.1.6), а скорость сходимости ряда (5.3.1.6) не хуже, чем у геометрической прогрессии со знаменателем $\sin(\alpha)$.

Пусть для спектра $\{\beta_\nu\}$ известны оценки для β_{\min} , β_{\max} – минимального и максимального по модулю собственного числа (или нижней и верхней границы расстояния от т.0 до области расположения спектра в случае непрерывного спектрального множества). Тогда, если весь спектр оператора размещается в круге Ω_B , натянутом на точки β_{\min} , β_{\max} как на концевые точки диаметра и точка $1 \notin \Omega_B$, для оптимального параметра верна простая приближенная формула

$$k_0 = \frac{\beta_{\min} + \beta_{\max}}{2} \quad (5.3.3.2)$$

Если граница круга принадлежит спектру, то формула (5.3.3.2) точная. Точная она также и в случае вещественного спектра. Формулу (5.3.3.2) можно улучшить, учитывая более точную конфигурацию спектральной области, например, если область расположения спектра – прямая линия. С помощью формулы (5.3.3.2) во многих случаях можно найти значение близкое к оптимальному параметру в условиях неполного знания свойств спектра, но при известных минимальных и максимальных по модулю собственных числах.

Сходимость каждого из рассмотренных методов простой итерации зависит от конкретного вида исходной матрицы, а точнее, от свойств её спектра. Можно привести примеры матриц, для которых сходится только один из рассмотренных методов, однако комбинация метода простой итерации, Зейделя

или Якоби с методом оптимального спектрального параметра (ОСП) позволяют добиться сходимости в случаях, когда каждый из этих методов по отдельности расходится.

Рассмотрим применение метода ОСП на примерах конкретных матричных задач.

Пусть элементы матрицы A при $n=2$ следующие: $a_{11}=2$, $a_{22}=-2$, $a_{12}=3$, $a_{21}=7$. Собственные числа матрицы B (5.3.1.2) равны $\beta_1=-4$, $\beta_2=6$ и располагаются по разные стороны от точки 1 на прямой, проходящей через неё. В этом случае точка 1 принадлежит выпуклой оболочке спектра и дробно-линейным преобразованием (5.3.3.1) нельзя добиться сходимости итерационного процесса. Собственные же числа матрицы Якоби (5.3.2.3) равны $Y_1=-2.3i$, $Y_2=2.3i$ (здесь i - мнимая единица) и точка 1 находится вне выпуклой оболочки спектра. То же самое можно утверждать и о спектре оператора Зейделя. Однако, непосредственное применение метода Якоби или Зейделя не приведёт к сходящемуся ряду, т.к. $|Y_V| > 1$ и не выполняется (5.3.1.5). Закладывая спектр Y_V в круг Ω_Y с центром в т. -8 приходим к сходящемуся методу Якоби – ОСП с параметром $k=8$. Для метода Зейделя - ОСП оптимальный параметр $k=1$ приводит к быстро сходящемуся процессу. Решение СЛАУ (5.1) с правой частью $b_V=1, i=1,2$ и точностью $\varepsilon=10^{-5}$ достигается за $m=20$ итераций ряда (5.3.1.6).

Наоборот, если матрица Якоби (оператор Зейделя) имеют спектр, выпуклая оболочка которого содержит т.1, то никакие модификации этих методов не приведут к сходящемуся процессу. Применение метода ОСП непосредственно к исходной матрице в виде (1.2) может привести в этом случае к сходимости. Такова матрица с элементами $a_{11}=5$, $a_{22}=-0.7$, $a_{12}=-4$, $a_{21}=2$, для которой собственные числа матрицы (1.2) $\beta_1=-1.5$, $\beta_2=0.8$, а собственные числа матрицы (2.3) $-Y_1=-1.5$, $Y_2=1.5$. Применение методов Якоби и Зейделя и их модификаций дают расходящийся процесс, т.к. точка 1 принадлежит выпуклой оболочке спектра. Применение же метода ОСП к простой итерации с матрицей (5.3.1.2) дает быстро сходящийся ряд. Решение СЛАУ (5.1) с точностью $\varepsilon=10^{-5}$ достигается за $m=9$ итераций ряда (5.3.1.6).

Применение метода ОСП наиболее успешно в том случае, когда спектр оператора B в (1.1) локализован в небольшой окрестности с центром в т. k_0 вдали от точки 1. Тогда применение этого метода с оптимальным параметром $k=-k_0$ является самым удачным среди одношаговых стационарных методов и приводит к быстро сходящемуся ряду простой итерации. В качестве примера рассмотрим СЛАУ с матрицей $a_{11}=7$, $a_{22}=3$, $a_{12}=4$, $a_{21}=-1$. В этом примере для матриц (1.2) и (2.3) имеем следующие собственные числа $\beta_1=-4$

, $\beta_2 = -4$ и $Y_1 = -0.436i$, $Y_2 = 0.436i$. Значение оптимального параметра $k_0 = -\frac{\beta_1 + \beta_2}{2} = 4$ переводит в данном случае точку $\beta = 4$, в которой находится весь спектр матрицы B , в точку $\tilde{\beta} = 0$, в которой находится спектр матрицы \tilde{B} . Таким образом, скорость сходимости ряда (5.3.1.6) с матрицей (5.3.3.1), (5.3.1.2) в данном случае очень высокая, т.к. $\rho(\tilde{B}) = 0$. Решение СЛАУ (1) с точностью до машинной константы достигается за $m = 2$ итерации. Решение той же задачи методами Якоби и Зейделя требует гораздо большего количества итераций - $m = 48$ и $m = 23$ соответственно. Для метода Якоби применение ОСП не даст улучшения сходимости, т.к. центр спектра и так находится в точке 0 и оптимальный параметр $k_0 = -\frac{Y_1 + Y_2}{2} = 0$. Для метода же Зейделя спектр оператора (5.3.2.5) отличается от спектра матрицы (5.3.2.3) и использование метода Зейделя-ОСП с оптимальным параметром $k_0 = 0.076$, т.е. ряда (1.6) с оператором (5.3.3.1), (5.3.2.5), приводит к уменьшению требуемого количества итераций - $m = 15$.

Пусть рассмотренная матрица продолжена на большую трехдиагональную матрицу с $n = 100$ и такими же элементами, т.е. на главной диагонали чередуются значения 7 и 3, а на двух соседних соответственно 4 и -1 . Спектр исходной матрицы существенно трансформируется из точки в протяженную область на комплексной плоскости, но при этом значение оптимального параметра, полученного по формуле (5.3.3.2) с участием минимального и максимального по модулю собственного числа матрицы B (5.3.1.2), остается неизменным $k_0 = 4$. Это справедливо для любой трехдиагональной матрицы, полученной таким периодическим продолжением из малой матрицы. Однако это значение k_0 все же приближенное в силу того, что матрица не является положительно определенной и другие комплексные собственные числа выходят за пределы круга, натянутого на $[\beta_{\min}, \beta_{\max}]$ как на диаметр. Опытным путем для сравнительно малых матриц с $n = 10$ значение оптимального параметра можно уточнить до $k_0 = 5.9$ и это значение остается практически неизменным для всех больших матриц такого вида. Для параметров $k_0 = 4$ и $k_0 = 5.9$ и

точности решения $\varepsilon = 10^{-5}$ получаем соответственно число требуемых итераций $m = 129$ и $m = 46$. Впечатляющий результат для данной задачи приносит метод Зейделя-ОСП. Если для обычного метода Зейделя число итераций $m = 190$, то с применением ОСП при $k_0 = 0.17$ число требуемых итераций снижается до $m = 6$!

Конечно, задача определения спектра матрицы в общем случае ничем не проще задачи решения СЛАУ прямыми методами. Однако, для ряда матриц приближенное значение оптимального параметра k_0 для метода ОСП в применении к простой итерации (5.3.1.2), (5.3.1.3) находится весьма просто через её коэффициенты. Например, для большой трехдиагональной матрицы с двумя постоянными диагоналями возле главной и с чередующимися значениями a и b коэффициентов на главной диагонали. Для такой матрицы A в (5.1) значение оптимального параметра в (5.3.1.6) с (5.3.3.1), (5.3.1.2) равно $k_0 = (a+b)/2 - 1$ и, если A - положительно определенная матрица, то это значение точное. Это не значит, что для любой матрицы такого типа можно построить сходящийся итерационный процесс, но если можно добиться сходимости, то при таком $k = k_0$ метод сходится.

Кроме того, для физических и технических задач область локализации спектра оператора часто известна, т.к. она соответствует физически нерегулярным и резонансным решениям.

Преобразование оператора (5.3.3.1) можно использовать в условиях неполной информации об его спектре. Так, например, если известна в точности только одна граница вещественного спектра. Более определенно, пусть известно, что собственные числа β_v находятся на интервале $[-M, m]$ и значение $M \geq 1$ известно точно, а для m известно лишь, что $m \in (0, 1)$. Т.к. для данного случая $\rho(\beta_v) \geq 1$, то ряд простой итерации расходится, но в силу того, что $1 \notin \Omega_B$ можно построить сходящийся ряд. Действительно, принимая $k = M$, получаем сходящийся ряд простой итерации для оператора \tilde{B} , спектр которого лежит на интервале $\left[0, \frac{m+M}{1+M}\right]$, причем $\frac{m+M}{1+M} < 1$, т.е. $\rho(\tilde{\beta}_v) < 1$. Можно показать также, что в условиях неопределенности данной задачи $0 < m < 1$ лучший результат даст $k = M/2$.

Если даже приходится детально исследовать спектр задачи для построения быстро сходящегося итерационного процесса то, однажды его построив, можно затем многократно использовать для расчетов с различными источниками - правыми частями \vec{b} .

Преимущества же быстро сходящихся итерационных процессов перед прямыми методами известны. Это:

- количество арифметических операций $\approx m \cdot n^2$ (здесь m - число итераций), вместо n^3 ;
- отсутствие накопления ошибок в процессе итераций со сжимающим оператором;
- пониженные требования к оперативной памяти ЭВМ.

Особенно эти преимущества заметны для задач с большими матрицами $n \geq 1000$. Решение СЛАУ с $n = 1000$ стандартным методом Mathcad на ЭВМ Р-2 750Мгц занимает около 3 мин машинного времени, в то время как решение той же системы быстро сходящимся итерационным методом с $m \approx 10..20$ требует всего около 1..2 сек.

5.4. Нахождение собственных векторов и собственных значений матриц

Собственными векторами и собственными значениями матрицы A называются вектора и числа, удовлетворяющие соотношению: $A\vec{x} = \lambda\vec{x}$, причем собственный вектор определен с точностью до постоянного множителя.

В дальнейшем рассматриваются невырожденные матрицы, имеющие различные собственные значения. Для нахождения собственных значений необходимо решить уравнение: $\det(A - E\lambda) = 0$. Нахождение коэффициентов характеристического полинома:

$$\det(A - E\lambda) = (-1)^n D(\lambda) = (-1)^n (\lambda^n + d_1\lambda^{n-1} + \dots + d_n)$$

непосредственным раскрытием определителя достаточно громоздко. В методе Крылова используется то, что подстановка в характеристический полином вместо переменной матрицы A дает в результате нулевую матрицу: $D(A) = O$.

Это тождество умножается слева на произвольный вектор \vec{y}_0 :

$\vec{y}_n + d_1\vec{y}_{n-1} + \dots + d_n\vec{y}_0 = 0$, где $\vec{y}_k = A^k \vec{y}_0$, то есть получается СЛАУ относительно коэффициентов характеристического полинома d_k . Для определения собственных векторов вводится система полиномов

$$R_i(\lambda) = \frac{D(\lambda)}{\lambda - \lambda_i} = \lambda^{n-1} + r_{1,i}\lambda^{n-2} + \dots + r_{n-1,i},$$

$$R_i(\lambda_j) = 0, \text{ если } i \neq j.$$

Учитывается, что собственные вектора \vec{x}_k линейно независимые, то есть любой вектор можно представить в виде их линейной комбинации:

$$\vec{y}_0 = c_1 \vec{x}_1 + c_2 \vec{x}_2 + \dots c_n \vec{x}_n.$$

Собственный вектор является линейной комбинацией векторов \vec{y}_k и коэффициентов полинома $R_i(\lambda)$. Действительно:

$$\begin{aligned} & \vec{y}_{n-1} + r_{1,i} \vec{y}_{n-2} + \dots + r_{n-1,i} \vec{y}_0 = \\ & c_1 \lambda_1^{n-1} \vec{x}_1 + c_2 \lambda_2^{n-1} \vec{x}_2 + \dots + c_n \lambda_n^{n-1} \vec{x}_n + \\ & + r_{1,i} (c_1 \lambda_1^{n-2} \vec{x}_1 + c_2 \lambda_2^{n-2} \vec{x}_2 + \dots + c_n \lambda_n^{n-2} \vec{x}_n) + \\ & \dots \\ & + r_{n-1,i} (c_1 \vec{x}_1 + c_2 \vec{x}_2 + \dots + c_n \vec{x}_n) \end{aligned}$$

Коэффициенты при собственных векторах представляют собой $c_j R_i(\lambda_j)$, которые все равны нулю кроме коэффициента с $i = j$, стоящего перед \vec{x}_i . То есть данная линейная комбинация является собственным вектором.