

Тема 4

Решение систем линейных уравнений.

Дана система линейных уравнений (СЛУ) с n неизвестными:

$$\begin{cases} a_{11} x_1 + a_{12} x_2 + \dots + a_{1n} x_n = b_1 \\ a_{21} x_1 + a_{22} x_2 + \dots + a_{2n} x_n = b_2 \\ \dots \\ a_{n1} x_1 + a_{n2} x_2 + \dots + a_{nn} x_n = b_n \end{cases} \quad (4.1)$$

В матричной форме записи система (4.1) имеет вид:

$$A \cdot X = B \quad (4.2)$$

где: n – порядок системы;

$$A = \begin{pmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & a_{22} & \dots & a_{2n} \\ \dots & & & \\ a_{n1} & a_{n2} & \dots & a_{nn} \end{pmatrix} - \text{матрица коэффициентов системы;}$$

$$B = \begin{pmatrix} b_1 \\ b_2 \\ \dots \\ b_n \end{pmatrix} \text{ – вектор свободных членов; } X = \begin{pmatrix} x_1 \\ x_2 \\ \dots \\ x_n \end{pmatrix} \text{ – вектор неизвестных;}$$

В свернутой форме записи СЛУ имеет вид:

$$\sum_{j=1}^n a_{i,j} \cdot x_j = b_i; \quad i = \overline{1, n} \quad (4.3)$$

Система называется *обусловленной* (не вырожденной, не особенной), если определитель системы $\Delta A \neq 0$, и тогда система (4.1) имеет единственное решение.

Система называется *не обусловленной* (вырожденной, особенной), если

$\Delta A = 0$, и тогда система (4.1) не имеет решений или имеет бесконечное множество решений.

На практике коэффициенты системы a_{ij} и свободные члены b_i часто задаются приближенно, с некоторой неустранимой погрешностью. Поэтому, кроме существования и единственности решения СЛУ, важно еще знать, как влияет такая погрешность на получаемое решение.

Система называется *плохо обусловленной*, если неустранимая погрешность оказывает сильное влияние на решение; у таких систем определитель близок, но не равен 0.

Рассмотрим пример плохо обусловленной системы.

Дана система
$$\begin{cases} x_1 + 0x_2 = 1 \\ x_1 + 0,001x_2 = 1 \end{cases} \quad \Delta A = \det A = \begin{vmatrix} 1 & 0 \\ 1 & 0,001 \end{vmatrix} = 0.001 \neq 0$$

Решение $x_1^* = 1$; $x_2^* = 0$.

Пусть b_2 имеет неустранимую погрешность $\pm 1\%$.

Если $b_2 = 1,01$, то $x_1^* = 1$; $x_2^* = 10$.

Если $b_2 = 0,99$, то $x_1^* = 1$; $x_2^* = -10$.

Решение изменяется очень сильно, следовательно, система плохо обусловлена, о чем говорит значение её определителя.

Рассмотрим геометрическую иллюстрацию обусловленности СЛУ на примере системы двух уравнений с двумя неизвестными:

$$\begin{cases} a_{11}x_1 + a_{12}x_2 = b_1 & \text{уравнение (I)} \\ a_{21}x_1 + a_{22}x_2 = b_2 & \text{уравнение (II)} \end{cases}$$

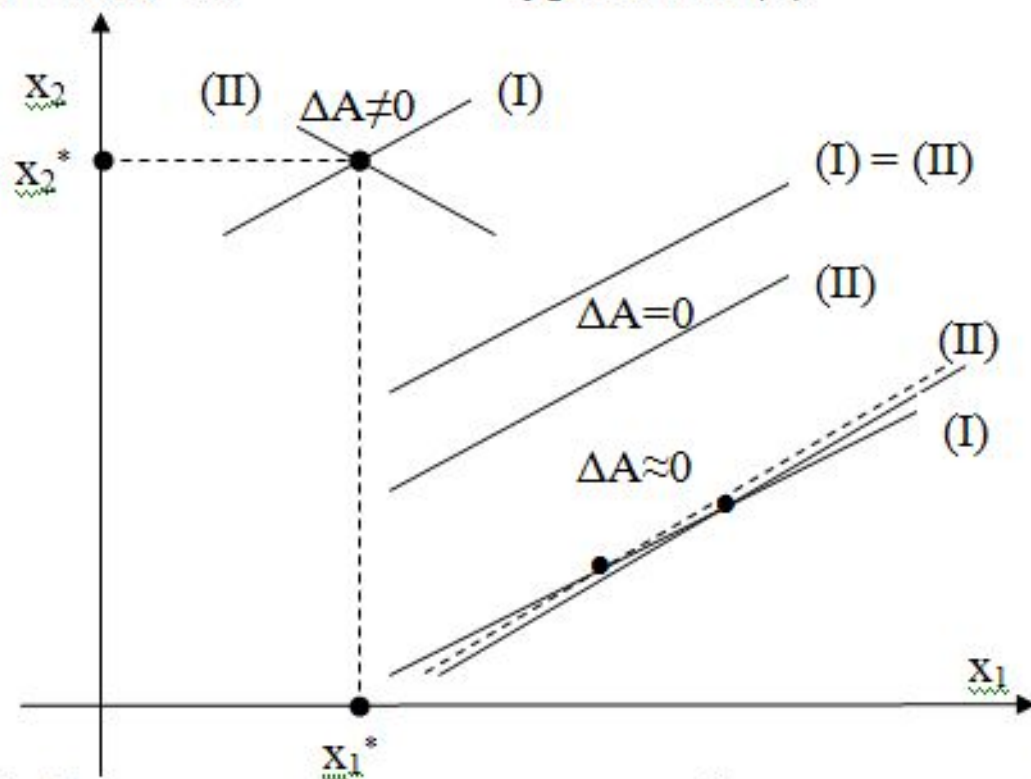


Рис. 4.1. Геометрическая иллюстрация обусловленности СЛУ.

Каждому уравнению в плоскости (x_1, x_2) соответствует прямая, а точка пересечения этих прямых является решением этой системы. Если $\Delta A = 0$, то наклоны прямых одинаковы, и они либо параллельны (т.е. не имеют решения), либо совпадают (имеют бесконечное множество решений). Если $\Delta A \neq 0$, то прямые имеют единственную точку пересечения.

Но если система плохо обусловлена ($\Delta A \approx 0$), даже незначительное изменение одного из коэффициентов приведет к сильному изменению решения системы, т.к. прямые почти параллельны.

Для решения СЛУ широко применяются прямые и итерационные методы. Область применения некоторых из них показана в таблице.



Тип	Название метода	Число арифметических действий (при $n = 20$)	Область применения
Прямые	Формулы Крамера	$\sim n! \cdot n^2$ ($9,7 \cdot 10^{20}$)	$n < 5$
	Исключения Гаусса	$\sim \frac{2}{3} n^3 + n^2$ (5733)	$n < 200$
Итерационные	Простых итераций	$\sim n^2$ на каждой итерации (400n)	до 10^5
	Гаусса-Зейделя		

Современная супер-ЭВМ имеет производительность 30 терафлоп – $30 \cdot 10^{12}$ операций с вещественными числами в секунду. Такой машине для решения СЛУ для $n=20$ по формуле Крамера требуется:

$$t = \frac{9,7 \cdot 10^{20}}{30 \cdot 10^{12} \cdot 365 \cdot 24 \cdot 60 \cdot 60} = 1,03 \text{ года.}$$

На решение СЛУ прямым методом сильное влияние оказывает погрешность округления, т.к. требуется огромное количество арифметических действий.

На решение СЛУ итерационным методом погрешность округления практически не влияет, но не всегда удается обеспечить сходимость итерационного процесса.

4.1. Формулы Крамера.

$$\underline{x}_i^* = \Delta A_i / \Delta A, \quad i = 1, n, 1 \quad (4.4)$$

где A_i – вспомогательная матрица, полученная из A заменой i -го столбца вектором свободных членов.

Пример 4.1. Решить СЛУ, используя формулы Крамера.

$$\begin{cases} \underline{x}_1 + 5 \cdot x_2 - x_3 = 2 \\ \underline{x}_1 \quad \quad \quad 2 \cdot x_3 = -1 \\ 2 \cdot x_1 - x_2 - 3 \cdot x_3 = 5 \end{cases}$$

Вычислим определители по правилу треугольников:

$$\Delta A = \begin{vmatrix} 1 & 5 & -1 \\ 1 & 0 & 2 \\ 2 & -1 & -3 \end{vmatrix} = 0 + 1 + 20 + 0 + 15 + 2 = 38 \text{ – система обусловлена.}$$

$$\Delta A_1 = \begin{vmatrix} 2 & 5 & -1 \\ -1 & 0 & 2 \\ -1 & 0 & 2 \end{vmatrix} = 0 + 50 - 1 - 0 + 4 - 15 = 38$$

$$\Delta A_2 = \begin{vmatrix} 1 & 2 & -1 \\ 1 & -1 & 2 \\ 2 & 5 & -3 \end{vmatrix} = 3 + 8 - 5 - 2 + 6 - 10 = 0$$

$$\Delta A_3 = \begin{vmatrix} 1 & 5 & 2 \\ 1 & 0 & -1 \\ 2 & -1 & 5 \end{vmatrix} = 0 - 10 - 2 - 0 - 25 - 1 = -38$$

Вычислим решения:

$$x_1^* = \Delta A_1 / \Delta A = 38/38 = 1;$$

$$x_2^* = \Delta A_2 / \Delta A = 0/38 = 0;$$

$$x_3^* = \Delta A_3 / \Delta A = -38/38 = -1.$$

Проверим полученное решение подстановкой в исходную систему.

$$1 + 5 \cdot 0 - (-1) = 2$$

$$1 + 2 \cdot (-1) = -1$$

$$2 \cdot 1 - 0 - 3 \cdot (-1) = 5$$

Система обращается в тождество, решение верное.

Формулы Крамера применяются редко, только для $n \leq 4$.

4.2. Метод исключений Гаусса.

Решим рассмотренную ранее систему (пример 4.1) методом исключения Гаусса.

Пример 3.2. Решение проводится в два этапа.

1 этап **Прямой ход** - матрица A преобразуется к треугольному виду: путем эквивалентных линейных преобразований уравнений системы поддиагональные коэффициенты матрицы A обнуляются.

$$\begin{cases} x_1 + 5x_2 - x_3 = 2 \\ x_1 \qquad \qquad 2x_3 = -1 \\ 2x_1 - x_2 - 3x_3 = 5 \end{cases}$$

Исключим x_1 из 2-го и 3-го уравнения: к 2-му уравнению прибавим 1-ое, умноженное на (-1) ; к 3-му уравнению прибавим 1-ое, умноженное на (-2) .

$$\begin{cases} x_1 + 5x_2 - x_3 = 2 \\ -5x_2 + 3x_3 = -3 \\ -11x_2 - x_3 = 1 \end{cases}$$

Исключим x_2 из 3-го уравнения: к 3-му уравнению прибавим 2-ое, умноженное на $(-11/5)$. Полученный вид системы после прямого хода

$$\begin{cases} x_1 + 5x_2 - x_3 = 2 \\ -5x_2 + 3x_3 = -3 \\ -38/5x_3 = 38/5 \end{cases}$$

2 этап Обратный ход - вычисляются значения неизвестных, начиная с последнего уравнения:

$$\underline{x_3^*} = -1$$

$$-5 \cdot x_2 + 3 \cdot x_3^* = -3 \Rightarrow x_2^* = (3 + 3 \cdot x_3^*) = (3 + 3 \cdot (-1)) = 0$$

$$\underline{x_1} + 5 \cdot x_2^* - x_3^* = 2 \Rightarrow x_1^* = 2 + 5 \cdot x_2^* + x_3^* = 2 + 5 \cdot 0 + (-1) = 1$$

Полученное решение нужно обязательно проверить, подставив в исходную систему!

Словесное описание алгоритма метода исключения Гаусса. Схема алгоритма приведена на рисунках 4.1-4.6.

Алгоритм прямого хода:

Шаг 1. Примем $k=1$

Шаг 2. Выбираем рабочую строку.

Если $\underline{a_{kk}} \neq 0$, то k -ая строка – рабочая.

Если нет, меняем k -ю строку на m -ю ($n \geq m > k$), в которой $\underline{a_{mk}} \neq 0$,

$m = \overline{k+1, n}$. Если такой строки нет, система вырожденная, решение прекратить.

Шаг 3. Для строк $i=k+1, k+2, \dots, n$ вычисляются новые значения коэффициентов.

$$q_i = \frac{a_{ik}}{a_{kk}}, \quad a_{ij} = a_{ij} - q_i a_{kj}, \quad j = \overline{k, n} \quad \text{и новые правые части } b_i = b_i - q_i b_k$$

Шаг 4. Увеличиваем $k = k + 1$. Если $k = n$, прямой ход завершен, иначе алгоритм повторяется со второго шага.

Получаем верхнюю треугольную матрицу A :

$$A = \begin{pmatrix} a_{11} & a_{12} & a_{13} & \dots & a_{1n} \\ 0 & a_{22} & a_{23} & \dots & a_{2n} \\ \dots & & & & \\ 0 & 0 & 0 & \dots & a_{nn} \end{pmatrix}, \quad B = \begin{pmatrix} b_1 \\ b_2 \\ \dots \\ b_n \end{pmatrix}$$

Алгоритм обратного хода:

Шаг 1. Вычислим $x_n^* = \frac{b_n}{a_{nn}}$

Шаг 2. Вычислим:

$$x_i^* = \frac{1}{a_{ii}^*} (b_i - \sum_{j=i+1}^n a_{ij} x_j^*), \quad i = n-1, n-2, \dots, 1$$



Рис. 4.1. Основной алгоритм решения СЛУ методом исключения Гаусса.

Для контроля правильности решения нужно считать невязки δ_i по формуле (4.5).

$$\delta_i = b_i - \sum_{j=1}^n a_{ij} x_j^*, \quad i = \overline{1, n} \quad (4.5)$$

Если невязки велики, задача решена неверно. Причиной может быть сбой машины (крайне редко), ошибки в программе, погрешность округления (при большом n и когда $\Delta A = \det A = 0$ - система плохо обусловлена).

Разновидности метода исключения:

а) Метод исключения Гаусса с выбором главного элемента в столбце. В алгоритме прямого хода на шаге 2 рабочая строка выбирается из условия

$$a_{m,k} = \max_{j=k, n} |a_{j,k}|,$$

т.е. рабочей выбирается та строка, в которой находится наибольший по модулю коэффициент k-го столбца, расположенный на главной диагонали и под ней.

б) Метод Гаусса-Жордана.

В алгоритм прямого хода нужно внести следующие изменения:

- на шаге 3 $i = \overline{1, k-1, k+1, n}$

- на шаге 4 прямой ход завершится при достижении условия $k > n$.

Вид матрицы коэффициентов после прямого хода

$$A' = \begin{pmatrix} a_{11} & 0 & 0 & \dots & 0 \\ 0 & a_{22} & 0 & \dots & 0 \\ 0 & 0 & a_{33} & \dots & 0 \\ \dots & & & & \\ 0 & 0 & 0 & \dots & a_{nn} \end{pmatrix}$$

Упрощается обратный ход: $x_i = b_i / a_{ii}, i = 1, 2, \dots, n$

Недостаток метода – увеличение общего числа действий, и соответственно, влияния погрешности округления.

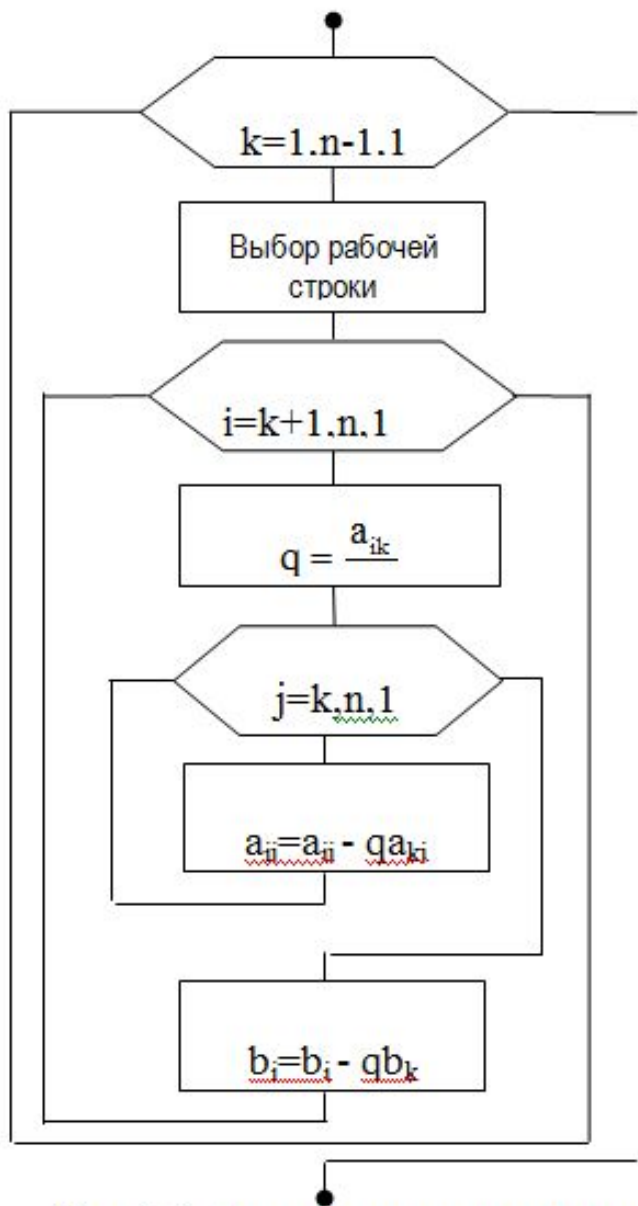


Рис. 4.2. Алгоритм прямого хода

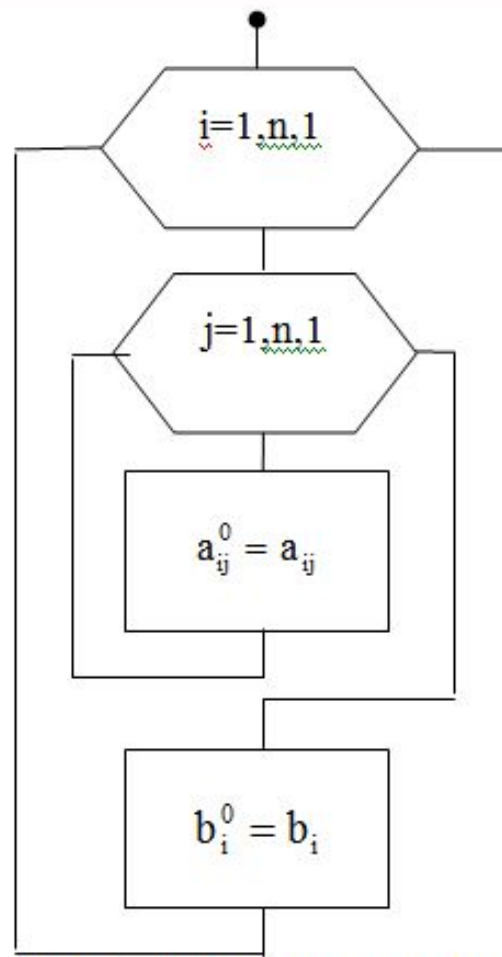
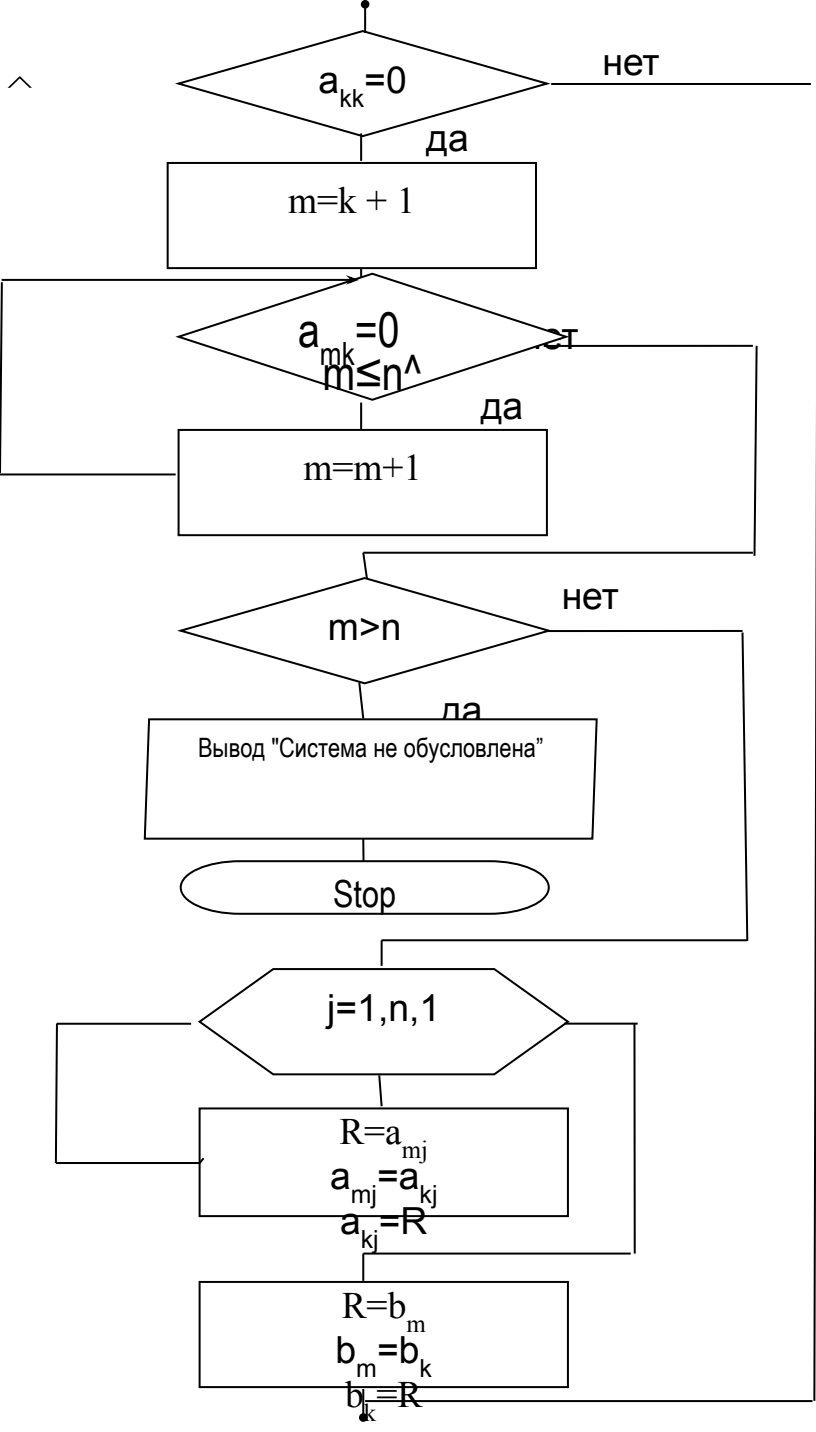


Рис. 3. Алгоритм запоминания коэффициентов.



Определение номера строки m , в которой коэффициент в k -ом столбце не равен нулю.

Завершение алгоритма при отсутствии ненулевых элементов под главной диагональю в k -ом столбце.

Перестановка элементов строк с номерами m и k

Рис. 4.5. Алгоритм выбора рабочей строки.

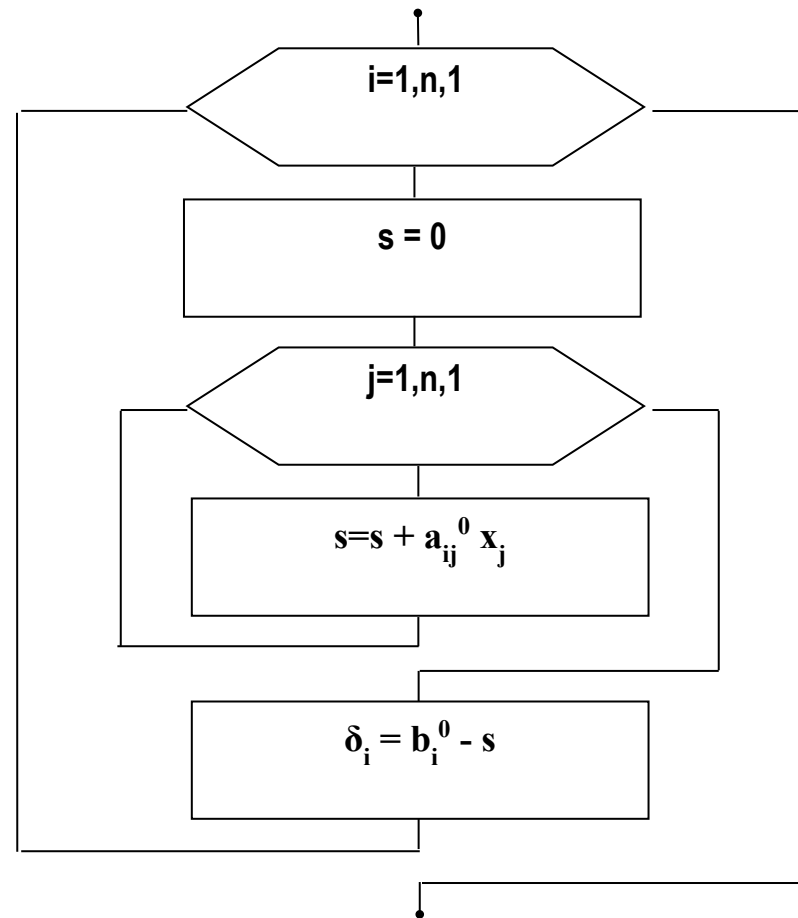
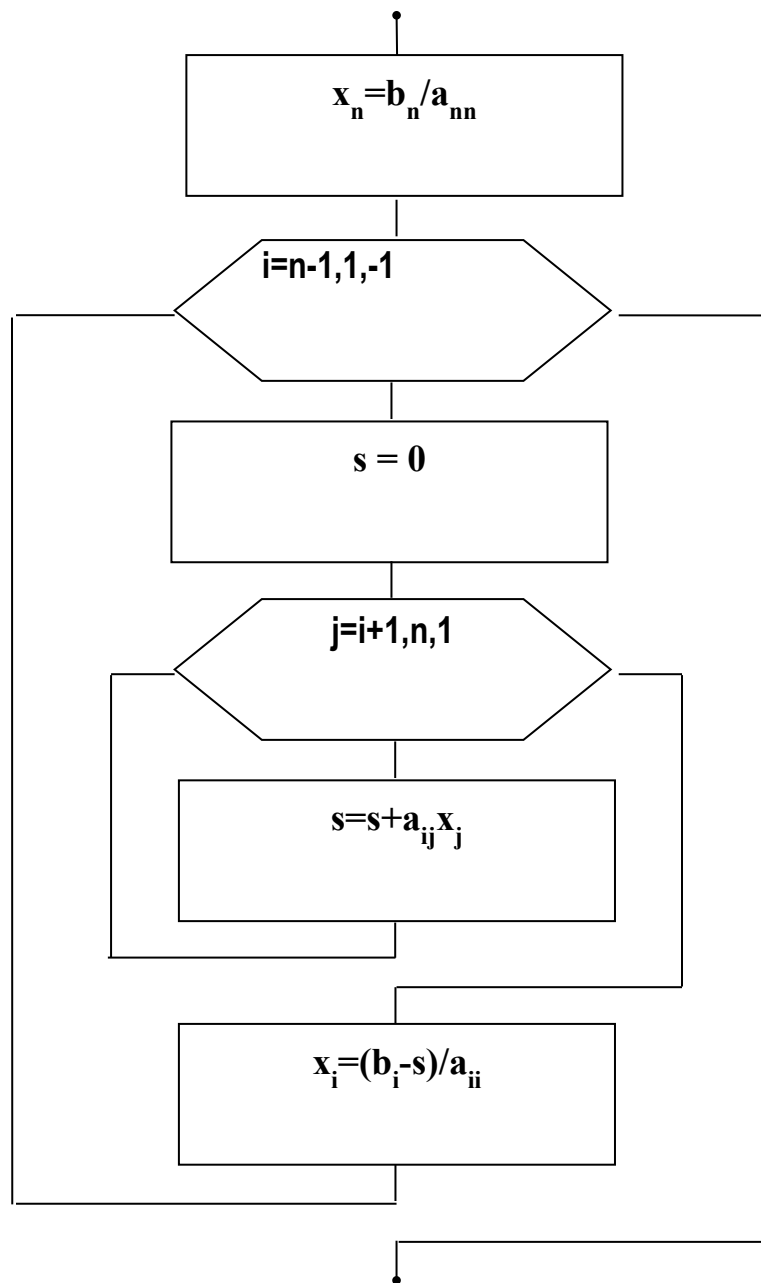


Рис. 4.6. Алгоритм расчета невязок
 Рис. 4.5. Алгоритм обратного хода.

Нужно подчеркнуть, что для вычисления значения определителя квадратной матрицы можно использовать алгоритм прямого хода: для треугольной или диагональной квадратной матрицы определитель равен произведению элементов главной диагонали.

4.3. Метод простых итераций.

Рассмотрим особенности решения СЛУ методом простых итераций на примере.

Пример 4.3. Требуется найти решение системы с точностью $\varepsilon=0,001$.

$$\begin{cases} \underline{x_1} + 5 \cdot x_2 - x_3 = 2 \\ \underline{x_1} \qquad \qquad \qquad 2 \cdot x_3 = -1 \\ 2 \cdot x_1 - x_2 - 3 \cdot x_3 = 5 \end{cases}$$


Приведем систему к новому, **каноническому виду** метода простых итераций. Для этого нужно преобразовать исходную систему так, чтобы в каждой строке новой матрицы A коэффициент, расположенный на главной диагонали, превышал по абсолютной величине сумму абсолютных значений остальных коэффициентов в этой строке.

При выполнении эквивалентных линейных преобразований системы нужно соблюдать следующие требования: каждое уравнение исходной системы должно участвовать хотя бы в одном преобразовании.

В первом уравнении исходной системы коэффициент при x_2 больше суммы модулей других коэффициентов: $5 > 1+1$. Поэтому это уравнение в новой системе нужно записать вторым уравнением. Для получения нового первого уравнения можно второе уравнение умножить на 2 и сложить с третьим уравнением. Для получения нового третьего уравнения можно из третьего уравнения вычесть второе.

В итоге описанных преобразований получится следующая система:

$$\begin{cases} 4x_1 - x_2 + x_3 = 3 \\ x_1 + 5x_2 - x_3 = 2 \\ x_1 - x_2 - 5x_3 = 6 \end{cases}$$

Важно отметить, что подобные преобразования не меняют решения 

темы.

Выразим явно из каждого нового уравнения очередное неизвестное – получим формулы итерационного процесса.

$$\begin{cases} x_1 = \frac{1}{4}(3 + x_2 - x_3) \\ x_2 = \frac{1}{5}(2 - x_1 + x_3) \\ x_3 = -\frac{1}{5}(6 - x_1 + x_2) \end{cases} \quad (*)$$

Возьмем любое начальное приближение $X^{(0)} = (x_1^{(0)}, x_2^{(0)}, x_3^{(0)})$, например $x_1^{(0)} = 0, x_2^{(0)} = 0, x_3^{(0)} = 0$.

Вычислим новое приближение решения $X^{(1)} = (x_1^{(1)}, x_2^{(1)}, x_3^{(1)})$, подставив в правую часть (*) начальное приближение:

$$x_1^{(1)} = \frac{1}{4}(3 + x_2^{(0)} - x_3^{(0)}) = \frac{3}{4} = 0,75$$

$$x_2^{(1)} = \frac{1}{5} (2 - x_1^{(0)} + x_3^{(0)}) = \frac{2}{5} = 0,4$$

$$x_3^{(1)} = -\frac{1}{5} (6 - x_1^{(0)} + x_2^{(0)}) = -\frac{6}{5} = -1,2$$

Оценим достигнувшую точность δ по формуле:

$$\delta = \max_{i=1,3} |x_i^{(1)} - x_i^{(0)}| = \max(|x_1^{(1)} - x_1^{(0)}|, |x_2^{(1)} - x_2^{(0)}|, |x_3^{(1)} - x_3^{(0)}|) = \max(0,75; 0,4; 1,2) = 1,2$$

Итерационный процесс нужно продолжить, т.к. $\delta > \varepsilon$.

Вычислим второе приближение $X^{(2)} = (x_1^{(2)}, x_2^{(2)}, x_3^{(2)})$, подставив в правую часть (*) первое приближение:

$$x_1^{(2)} = \frac{1}{4} (3 + x_2^{(1)} - x_3^{(1)}) = \frac{1}{4} (3 + 0,4 + 1,2) = \frac{4,6}{4} = 1,15$$

$$x_2^{(2)} = \frac{1}{5} (2 - x_1^{(1)} + x_3^{(1)}) = \frac{1}{5} (2 - 0,75 - 1,2) = \frac{0,05}{5} = 0,01$$

$$x_3^{(2)} = -\frac{1}{5} (6 - x_1^{(1)} + x_2^{(1)}) = -\frac{1}{5} (6 - 0,75 + 0,4) = -\frac{5,65}{5} = -1,13$$

$$\delta = \max_{i=1,3} |x_i^{(2)} - x_i^{(1)}| = \max(|1,15 - 0,75|; |0,01 - 0,4|; |-1,13 + 1,2|) = 0,4$$

Третье приближение:

$$x_1^{(3)} = \frac{1}{4} (3 + 0,01 + 1,13) = \frac{4,14}{4} = 1,0350$$

$$x_2^{(3)} = \frac{1}{5} (2 - 1,15 - 1,13) = -\frac{0,28}{5} = -0,056$$

$$x_3^{(3)} = -\frac{1}{5} (6 - 1,15 + 0,01) = -\frac{4,86}{5} = -0,972 \quad \delta = 0,158$$

Четвертое приближение:

Четвертое приближение:

$$x_1^{(4)} = 1,007, \quad x_2^{(4)} = -0,0014, \quad x_3^{(4)} = -0,9818, \quad \delta = 0,0546$$

Очевидно, что итерационный процесс сходится, т.к. значение δ монотонно убывает. Для достижения требуемой точности $\varepsilon=0,001$ потребуется еще несколько итераций.

Скорость сходимости зависит от уровня преобладания значений диагональных коэффициентов.

Основные расчетные зависимости метода простых итераций:

Формула итерационного процесса:

$$x_i^{(k)} = \frac{1}{a_{ii}} \left(b_i - \sum_{\substack{j=1 \\ i \neq j}}^n a_{ij} \cdot x_j^{(k-1)} \right), \quad i = \overline{1, n} \quad (4.6)$$

где: $k = 1, 2, \dots$ – номер приближения.

$x_i^{(0)}$ – начальное приближение, $i = \overline{1, n}$;

Условия завершения итерационного процесса:

$$\delta \leq \varepsilon \quad (4.7)$$

где ε – требуемая точность;

$$\delta - \text{оценка достигнутой точности, } \delta = \max_{i=\overline{1,n}} |x_i^{(k)} - x_i^{(k-1)}| \quad (4.8)$$

$$\text{или } \delta = \frac{1}{n} \sum_{i=1}^n |x_i^{(k)} - x_i^{(k-1)}| \quad (4.9)$$

Условие сходимости итерационного процесса (условие преобладания диагональных коэффициентов):

$$|a_{ii}| > \sum_{\substack{j=1 \\ i \neq j}}^n |a_{ij}| \quad \forall i = \overline{1,n} \quad (4.10)$$

Схема алгоритма метода представлена на рис. 4.7.

Если в полученных результатах значения $\delta > \varepsilon$ и $k > k_{\max}$, то задача не решена, т.е. $x(1:n)$ не является решением системы. Необходимо проверить условия сходимости или увеличить k_{\max} .

4.4 Метод Гаусса-Зейделя

В формуле итерационного процесса метода простых итераций (4.6) к моменту вычисления $x_i^{(k)}$ уже вычислены значения $x_1^{(k)}, x_2^{(k)}, \dots, x_{i-1}^{(k)}$.

Очевидно, что эти значения в большинстве случаев ближе к решению и их можно использовать для вычисления $x_i^{(k)}$. Исходя из этого, Гаусс и Зейдель предложили видоизмененную формулу итерационного процесса

$$X_i^{(k)} = \frac{1}{a_{i,i}} \left(b_i - \sum_{j=1}^{i-1} a_{i,j} x_j^k - \sum_{j=i+1}^n a_{i,j} x_j^{k-1} \right), \quad \overline{i = 1, n} \quad (4.11)$$

Условие завершения итерационного процесса (4.7) и условия сходимости (4.10) справедливы и для данного метода. Поэтому схема алгоритма Гаусса-Зейделя отлична только формулой расчета нового приближения:

$$x_i = \frac{1}{a_{i,i}} \left(b_i - \sum_{\substack{j=1 \\ i \neq j}}^n a_{i,j} x_j \right), \quad \overline{i = 1, n}$$

Метод этот, как правило, позволяет достичь требуемой точности ε за меньшее число итераций, т.е. имеет лучшую сходимость.

Достоинства итерационных методов:

1. Погрешность округления не накапливается от итерации к итерации.
2. Число итераций при $n > 100$ обычно меньше n , поэтому общее число действий меньше n^3 , т.е. меньше, чем в методе исключений Гаусса.
3. Не требуется большой объем памяти.
4. Итерационные методы особенно выгодны для систем с большим количеством нулевых коэффициентов (систем с разреженной итерацией). Методы исключения наоборот: чем больше нулей, тем чаще требуется выбирать новую рабочую строку.

Недостаток - не всегда можно обеспечить
сходность итерационного процесса. С
увеличением размерности системы труднее
выполнить линейные преобразования для
обеспечения сходимости.

1
SimpleIterations

2
 $K = 0$

3
 $x_i = 0$
 $i = \overline{1, n}$

4
 $k = k + 1$
 $x0_i = x_i,$
 $i = \overline{1, 2, \dots, n}$

5
Вычисление
нового при-
ближения

Входные данные:
 n – порядок системы;
 a ($1 \times n, 1 \times n$), b ($1 \times n$) – коэффициенты системы в каноническом виде;
– точность задания;
 K_{\max} – максимальное число итераций.

Задаётся нулевое начальное приближение

Увеличивается счётчик итераций
Запоминается старое приближение

$$x_i = \frac{1}{a_{ii}} (b_i - \sum_{\substack{j=1 \\ i \neq j}}^n a_{ij} \cdot x0_j), \quad i = 1, 2, \dots, n$$

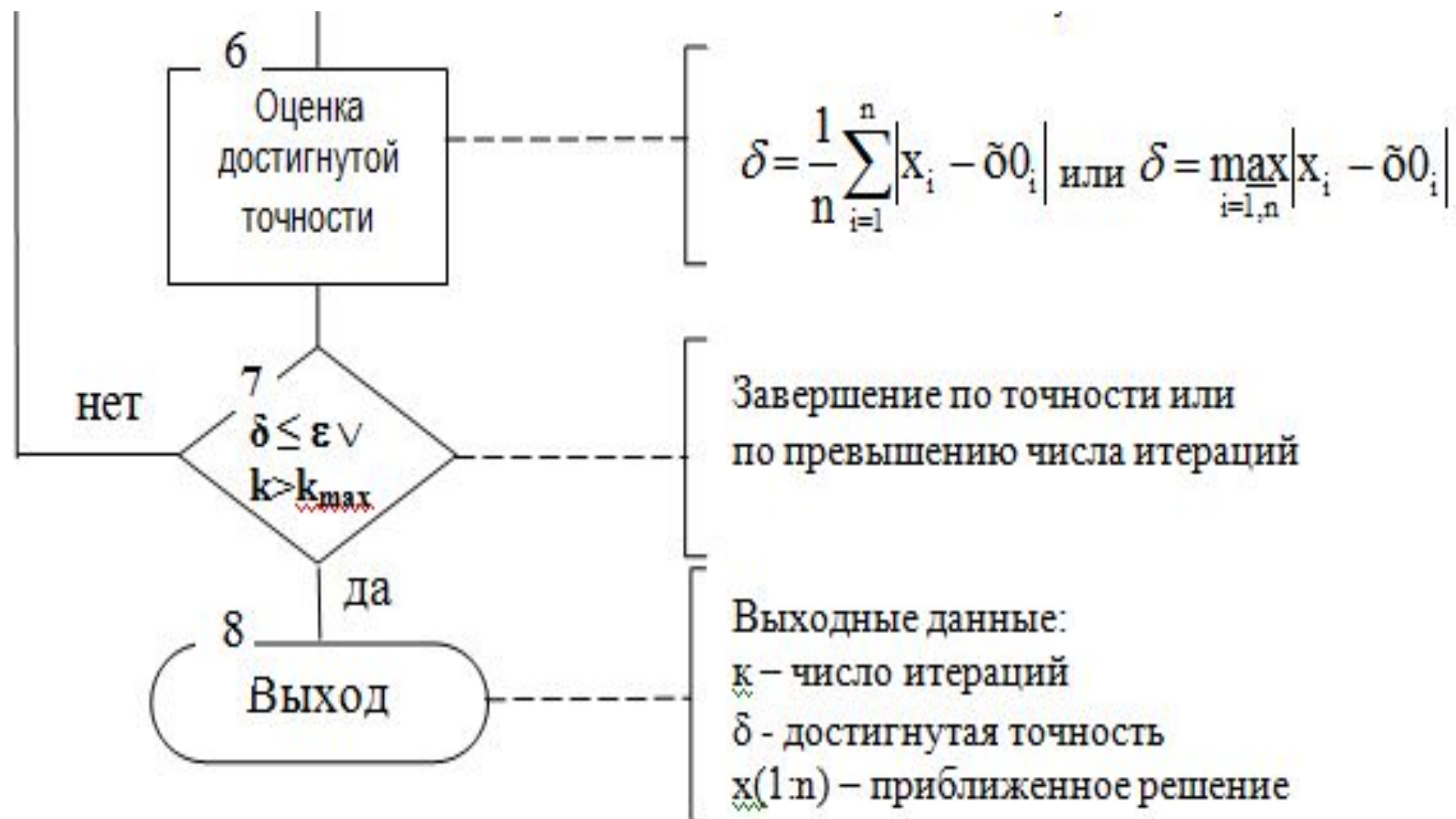


Рис. 4.7. Схема алгоритма метода простых итераций