

УНИВЕРСИТЕТ ИТМО

Вычислительная математика

Малышева Татьяна Алексеевна, к.т.н., доцент

tamalysheva@itmo.ru

Санкт-Петербург, 2020



Курс «Вычислительная математика»: 108 часов (3 з.е.), 4 семестр.

Курс включает: лекции - 16 часов, лабораторные занятия - 32 часа, СРС - 55,2 часов.

Задачей изучения дисциплины «Вычислительная математика» является формирование у студента необходимых знаний:

- о вычислительной математике как о разделе высшей математики;
- о классификации численных методов;
- о причинах возникновения погрешностей и их учете при оценке результата вычислений;
- об основах численных методах линейной алгебры, о решении нелинейных уравнений и систем, о приближении функций, об основах дифференцирования и интегрирования функций;

В результате лабораторных занятий студент должен уметь:

- выбрать численный метод, которым ему необходимо воспользоваться при решении конкретной задачи;
- написать программное приложение, реализующее данный метод;
- адекватно оценить полученные результаты.



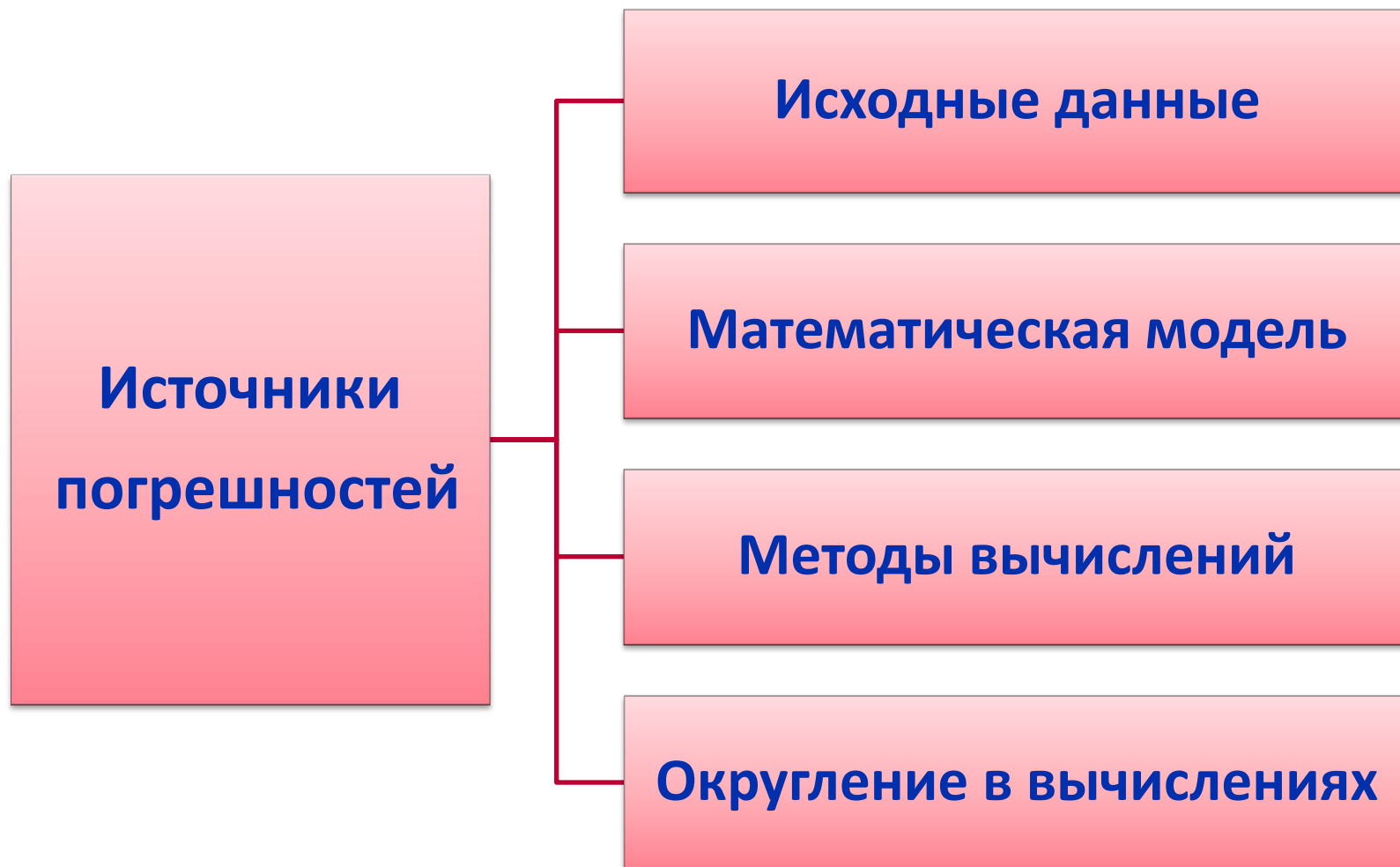
Методы решения математических задач

```
graph TD; A[Методы решения математических задач] --> B[Графические]; A --> C[Аналитические]; A --> D[Численные];
```

Графические

Аналитические

Численные





Абсолютная и относительная погрешности

Абсолютная погрешность :

$$\Delta a^* = |a^* - a|,$$

где a^* – приближенное значение числа a .

Точное число заключено в границах:

$$a = a^* \pm \Delta a^* \text{ или } a^* - \Delta a^* \leq a \leq a^* + \Delta a^*.$$

Предельная абсолютная погрешность (граница абсолютной погрешности) Δ_a :

$$\Delta a^* \leq \Delta_a$$

Относительная погрешность:

$$\delta a^* = \left| \frac{a^* - a}{a^*} \right|$$

Предельная относительная погрешность (граница относительной погрешности) δ_a :

$$\delta a^* \leq \delta_a$$



Вычислительная погрешность

Проведение численных расчётов на компьютере неизбежно связано с погрешностью округления, которые возникают в силу **ограниченности разрядной сетки компьютера** при представлении в нем вещественных чисел.

В современных компьютерах реализован стандарт двоичной арифметики IEEE. Стандарт предусматривает два основных типа чисел с плавающей точкой: числа одинарной и двойной точности.

Тип	Длина	Знак	Мантисса	Порядок
Одинарная точность	4 байта	1 бит	23 бита	8 битов
Двойная точность	8 байтов	1 бит	52 битов	11 битов

Хранение в компьютере в логарифмическом виде – мантисса и порядок: $x = \pm t \cdot a^p$, где t – мантисса, p – порядок, a – основание степени.

Мантисса записывается в нормализованной форме: $2.578 \cdot 10^2$
Компьютерное представление: **2.578E+02.**



Вычислительная погрешность

Точность	Одинарная	Двойная
Наименьшее значение (UFL), порог машинного нуля	$\approx 10^{-38}$	$\approx 10^{-308}$
Наибольшее значение (OFL), порог переполнения	$\approx 10^{+38}$	$\approx 10^{+308}$
Машинное эpsilon ($\epsilon_{\text{маш}}$), машинная погрешность	$\approx 10^{-8}$	$\approx 10^{-16}$



Свойства численных методов

- Устойчивость – малые погрешности в исходной величине приводят к малым погрешностям в решении (или при небольших изменениях входных данных результаты решения изменяются незначительно).
- Сходимость – численное решение задачи должно стремиться к точному решению задачи.
- Численные методы применяются к корректно поставленным задачам.

Задача называется поставленной корректно, если для любых значений входных данных из некоторого класса выполняются следующие условия:

- 1) решение задачи существует и единственно в рассматриваемом классе решений.
- 2) решение устойчиво.

Лекция №1 МЕТОДЫ РЕШЕНИЯ СИСТЕМ ЛИНЕЙНЫХ АЛГЕБРАИЧЕСКИХ УРАВНЕНИЙ

Рассмотрим систему n линейных алгебраических уравнений с n неизвестными:

[illegible]

В векторно-матричном виде: $Ax = b$,

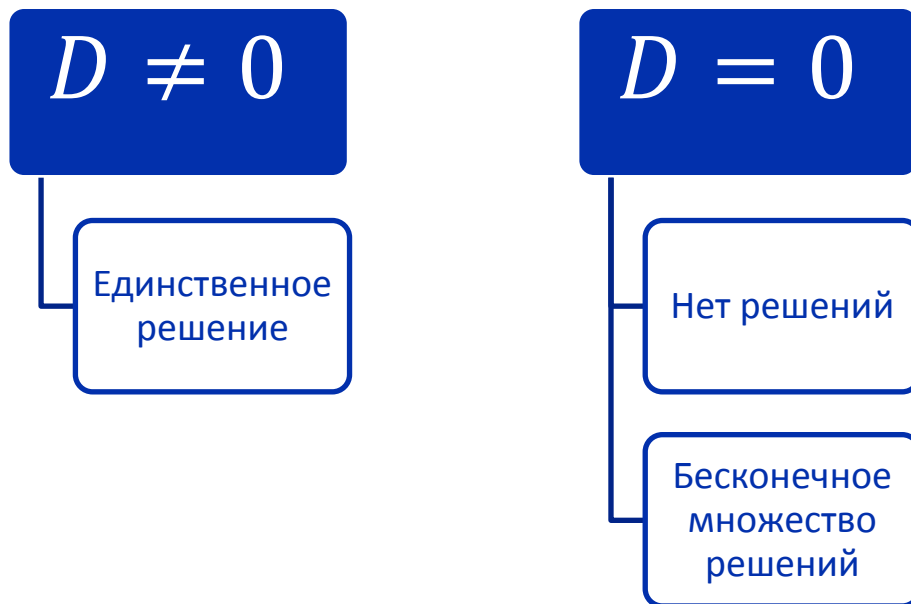
где: A – матрица системы, x – вектор неизвестных, b – вектор правых частей:

$$A = \begin{pmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & a_{22} & \dots & a_{2n} \\ \dots & \dots & \dots & \dots \\ a_{n1} & a_{n2} & \dots & a_{nn} \end{pmatrix} x = \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{pmatrix} \quad b = \begin{pmatrix} b_1 \\ b_2 \\ \vdots \\ b_n \end{pmatrix}$$

Решение задачи

Система линейных алгебраических уравнений может:

- 1) Не иметь решений (быть *вырожденной* или *несовместной*).
- 2) Иметь бесконечное множество решений.
- 3) Иметь единственное решение.





МЕТОДЫ РЕШЕНИЯ ЛИНЕЙНЫХ СИСТЕМ

ПРЯМЫЕ (ТОЧНЫЕ)

ИТЕРАЦИОННЫЕ (ПРИБЛИЖЕННЫЕ)

ПРЯМЫЕ МЕТОДЫ

используют конечные соотношения (формулы) для вычисления неизвестных. Они дают решение за конечное число арифметических операций. Эти методы сравнительно просты и наиболее универсальны, т. е. пригодны для решения широкого класса линейных систем.

Недостатки:

- требуют хранения в оперативной памяти компьютера сразу всей матрицы, и при больших значениях n расходуется много места в памяти.
- не учитывают структуру матрицы при большом числе нулевых элементов в разреженных матрицах (например, клеточных или ленточных); эти элементы занимают место в памяти машины, и над ними проводятся арифметические действия.
- происходит накопление погрешностей в процессе решения, поскольку вычисления на любом этапе используют результаты предыдущих операций.

Применяются для систем ($n < 1000$) с плотно заполненной матрицей и не близким к нулю определителем.



Прямые методы

Формулы Крамера. Каждое неизвестное представляется в виде отношения определителей (детерминантов). Используется для систем размерностью $n=2, 3$.

$$a_{11}x_1 + a_{12}x_2 + \dots + a_{1n}x_n = b_1,$$

$$a_{21}x_1 + a_{22}x_2 + \dots + a_{2n}x_n = b_2,$$

$$a_{n1}x_1 + a_{n2}x_2 + \dots + a_{nn}x_n = b_n.$$

$$x_1 = \frac{\Delta_{x_1}}{\Delta}, x_2 = \frac{\Delta_{x_2}}{\Delta}, x_3 = \frac{\Delta_{x_3}}{\Delta}, \dots, x_n = \frac{\Delta_{x_n}}{\Delta}$$

$$\Delta = \begin{vmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & a_{22} & \dots & a_{2n} \\ \dots & \dots & \dots & \dots \\ a_{n1} & a_{n2} & \dots & a_{nn} \end{vmatrix} \neq 0 - \text{определитель системы}$$

$$\Delta_{x_1} = \begin{vmatrix} b_1 & a_{12} & \dots & a_{1n} \\ b_2 & a_{22} & \dots & a_{2n} \\ \dots & \dots & \dots & \dots \\ b_n & a_{n2} & \dots & a_{nn} \end{vmatrix} \quad \Delta_{x_2} = \begin{vmatrix} a_{11} & b_1 & \dots & a_{1n} \\ a_{21} & b_2 & \dots & a_{2n} \\ \dots & \dots & \dots & \dots \\ a_{n1} & b_n & \dots & a_{nn} \end{vmatrix} \quad \dots \quad \Delta_{x_n} = \begin{vmatrix} a_{11} & a_{12} & \dots & b_1 \\ a_{21} & a_{22} & \dots & b_2 \\ \dots & \dots & \dots & \dots \\ a_{n1} & a_{n2} & \dots & b_n \end{vmatrix}$$

При большом числе уравнений потребуются выполнить огромное число арифметических операций, поскольку для вычислений n неизвестных необходимо найти значения $n + 1$ определителей. Количество арифметических операций: $N \approx n \cdot n!$

При $n=100$ потребуется совершить 10^{158} операций. Много это или сойдет для ЭВМ?

Прямые методы. Метод Гаусса

Основан на приведении матрицы системы к треугольному виду так, чтобы ниже ее главной диагонали находились только нулевые элементы

Прямой ход метода Гаусса состоит в последовательном исключении неизвестных из уравнений системы. Сначала с помощью первого уравнения исключается x_1 из всех последующих уравнений системы. Затем с помощью второго уравнения исключается x_2 из третьего и всех последующих уравнений и т.д.

Этот процесс продолжается до тех пор, пока в левой части последнего (n -го) уравнения не останется лишь один член с неизвестным x_n , т. е. матрица системы будет приведена к треугольному виду.

Обратный ход метода Гаусса состоит в последовательном вычислении искомых неизвестных: решая последнее уравнение, находим единственное в этом уравнении неизвестное x_n . Далее, используя это значение, из предыдущего уравнения вычисляем x_{n-1} и т. д. Последним найдем x_1 из первого уравнения.

$$\begin{aligned} a_{11}x_1 + a_{12}x_2 + a_{13}x_3 + \dots + a_{1n}x_n &= b_1, \\ a_{22}^{(1)}x_2 + a_{23}^{(1)}x_3 + \dots + a_{2n}^{(1)}x_n &= b_2^{(1)}, \\ a_{32}^{(1)}x_2 + a_{33}^{(1)}x_3 + \dots + a_{3n}^{(1)}x_n &= b_3^{(1)}, \\ a_{n2}^{(1)}x_2 + a_{n3}^{(1)}x_3 + \dots + a_{nn}^{(1)}x_n &= b_n^{(1)} \end{aligned} \quad (2) \quad \begin{aligned} a_{ij}^{(1)} &= a_{ij} - \frac{a_{i1}}{a_{11}} a_{1j}, i, j = 2, 3 \dots n \\ b_i^{(1)} &= b_i - \frac{a_{i1}}{a_{11}} b_1, i = 2, 3 \dots n \end{aligned}$$



Прямые методы. Метод Гаусса

Шаг 2:

Исключим x_2 из третьего уравнения: умножим второе уравнение на $(-a'_{32}/a'_{22})$ и прибавим к третьему (и т.д. для следующих уравнений)

Исключим x_2 из последнего уравнения: умножим второе уравнение на $(-a'_{n2}/a'_{22})$ и прибавим к последнему.

Получим:

$$\begin{aligned}
 a_{11}x_1 + a_{12}x_2 + a_{13}x_3 + \dots + a_{1n}x_n &= b_1, \\
 a_{22}^{(1)}x_2 + a_{23}^{(1)}x_3 + \dots + a_{2n}^{(1)}x_n &= b_2^{(1)}, \\
 a_{33}^{(2)}x_3 + \dots + a_{3n}^{(2)}x_n &= b_3^{(2)} \\
 a_{n3}^{(2)}x_3 + \dots + a_{nn}^{(2)}x_n &= b_n^{(2)}
 \end{aligned} \tag{3}$$

$$a_{ij}^{(2)} = a_{ij}^{(1)} - \frac{a_{i2}^{(1)}}{a_{22}^{(1)}} a_{2j}^{(1)}, \quad i, j = 3, 4 \dots n \quad b_i^{(2)} = b_i^{(1)} - \frac{a_{i2}^{(1)}}{a_{22}^{(1)}} b_2^{(1)}, \quad i = 3, 4 \dots n$$

Прямые методы. Метод Гаусса

Продолжим до тех пор, пока матрица системы (3) не примет треугольный вид (4):

$$\begin{aligned} a_{11}x_1 + a_{12}x_2 + a_{13}x_3 + \dots + a_{1n}x_n &= b_1, \\ a_{22}^{(1)}x_2 + a_{23}^{(1)}x_3 + \dots + a_{2n}^{(1)}x_n &= b_2^{(1)}, \\ a_{33}^{(2)}x_3 + \dots + a_{3n}^{(2)}x_n &= b_3^{(2)}, \\ a_{nn}^{(n-1)}x_n &= b_n^{(n-1)} \end{aligned} \quad (4)$$

Матрица системы (4) имеет треугольный вид \rightarrow конец *прямого хода*.

Требование : Если в процессе исключения неизвестных, коэффициенты:

$$a_{11}, a_{22}^1, a_{33}^2, \dots = 0,$$

тогда необходимо соответственным образом переставить уравнения системы.

Перестановка уравнений должна быть предусмотрена в вычислительном алгоритме при его реализации на компьютере.



Прямые методы. Метод Гаусса

Обратный ход:

$$x_n = \frac{b_n^{(n-1)}}{a_{nn}^{(n-1)}}$$

.....

$$x_2 = \frac{1}{a_{22}^{(1)}} (b_2^{(1)} - a_{23}^{(1)} x_3 - \dots - a_{2n}^{(1)} x_n)$$

$$x_1 = \frac{1}{a_{11}} (b_1 - a_{12} x_2 - a_{13} x_3 - \dots - a_{1n} x_n)$$



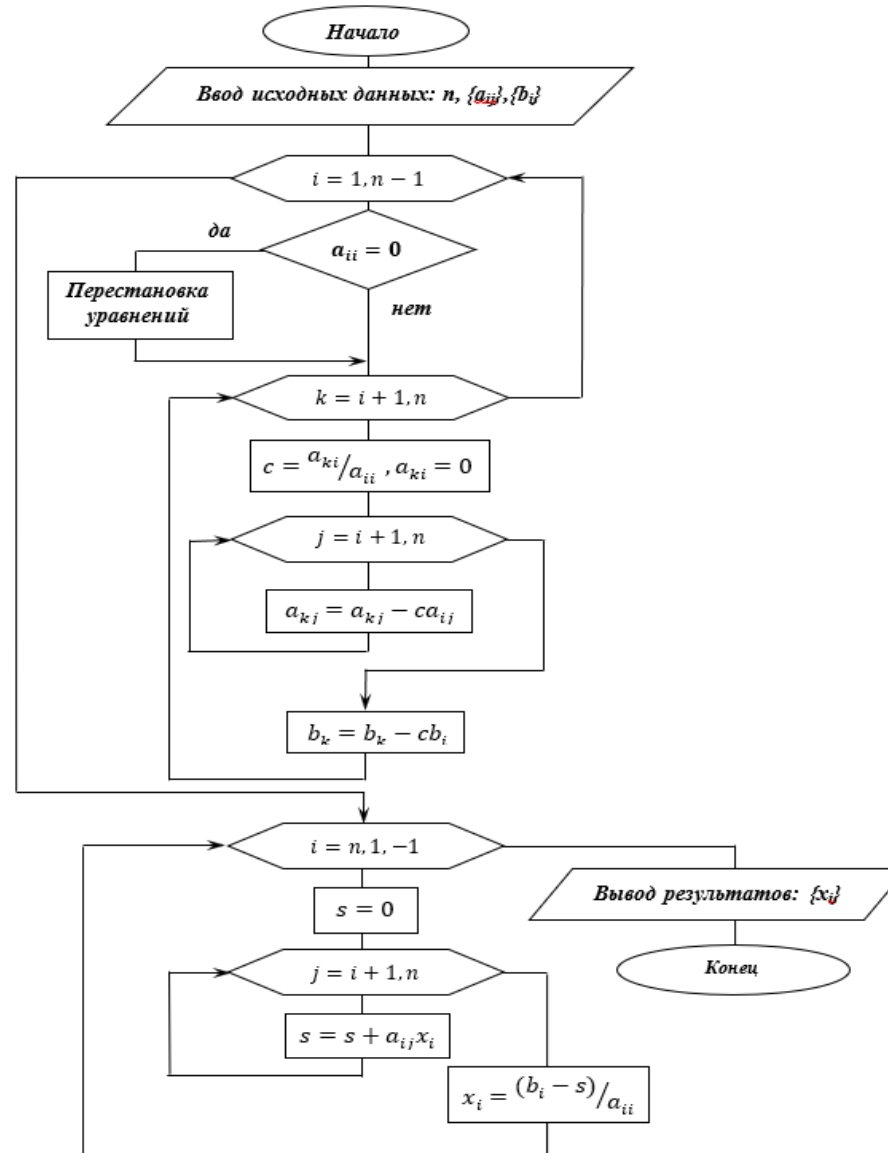
Блок-схема метода Гаусса

Первый цикл с переменной цикла i реализует прямой ход, а второй – обратный ход метода.

i – номер неизвестного, которое исключается из оставшихся $n - 1$ уравнений при прямом ходе (а также номер уравнения, из которого исключается x_i) и номер неизвестного, которое определяется из i -го уравнения при обратном ходе;

k – номер уравнения, из которого исключается неизвестное x_i при прямом ходе;

j – номер столбца при прямом ходе и номер уже найденного неизвестного при обратном ходе.



Метод Гаусса. Пример 1

Рассмотрим алгоритм решения линейной системы методом Гаусса для случая трех уравнений:

$$\begin{cases} 10x_1 - 7x_2 &= 7 \\ -3x_1 + 2x_2 + 6x_3 &= 4 \\ 5x_1 - x_2 + 5x_3 &= 6 \end{cases}$$

Исключим x_1 из второго и третьего уравнений. Для этого сначала умножим первое уравнение на 0.3 и результат прибавим ко второму, а затем умножим первое же уравнение на -0.5 и результат прибавим к третьему. Получим:

$$\begin{cases} 10x_1 - 7x_2 &= 7 \\ -0,1x_2 + 6x_3 &= 6,1 \\ 2,5x_2 + 5x_3 &= 2,5 \end{cases}$$

Исключим x_2 из третьего уравнения (заметим, что ведущий элемент a_{22} мал, поэтому было бы лучше переставить второе и третье уравнения). Однако мы проводим сейчас вычисления в рамках точной арифметики и погрешности округлений не опасны, поэтому продолжим исключение. Умножим второе уравнение на 25 и результат сложим с третьим уравнением. Получим систему в треугольном виде:

$$\begin{cases} 10x_1 - 7x_2 &= 7 \\ -0,1x_2 + 6x_3 &= 6,1 \\ 155x_3 &= 155 \end{cases}$$

На этом заканчивается прямой ход метода Гаусса.

Обратный ход состоит в последовательном вычислении x_3, x_2, x_1

$$x_3 = \frac{155}{155} = 1 \quad x_2 = \frac{6x_3 - 6,1}{0,1} = -1 \quad x_1 = \frac{7x_2 + 7}{10} = 0$$

Подстановкой в исходную систему легко убедиться, что $(0, -1, 1)$ и есть ее решение.

Метод Гаусса. Пример 2

Изменим коэффициенты исходной системы:

$$\begin{cases} 10x_1 - 7x_2 = 7 \\ -3x_1 + \mathbf{2,099}x_2 + 6x_3 = \mathbf{3,901} \\ 5x_1 - x_2 + 5x_3 = 6 \end{cases}$$

Здесь изменены коэффициент при x_2 и правая часть второго уравнения.

Вычисления проведем в рамках арифметики с плавающей точкой, сохраняя пять разрядов числа.

$$\begin{cases} 10x_1 - 7x_2 = 7 \\ -0,001x_2 + 6x_3 = 6,001 \\ 2,5x_2 + 5x_3 = 2,5 \end{cases}$$

Следующий шаг исключения проводим при малом ведущем элементе (-0.001). Чтобы исключить x_2 из третьего уравнения, надо умножить второе уравнение на 2500. При умножении $6,001 \cdot 2500 = 15002,5$ при округлении до пяти разрядов $\rightarrow 15\ 003$.

$$15003 + 2,5 = 15005,5 \rightarrow 15006$$

$$\begin{cases} 10x_1 - 7x_2 = 7 \\ -0,001x_2 + 6x_3 = 6,001 \\ 15005x_3 = 15006 \end{cases} \quad x_3 = \frac{15006}{15005} = 1,0001 \quad x_2 = \frac{6,001x_3 - 6,1}{0,001} = -0,4 \quad x_1 = \frac{7x_2 + 7}{10} = 0,42$$

Вычисления проводились с округлением до пяти разрядов по аналогии с процессом вычислений на компьютере. В результате этого было получено решение (0.42, -0.4, 1.0001) вместо (0, -1, 1). Такая большая неточность результатов объясняется малой величиной ведущего элемента.

Прямые методы. Метод Гаусса с выбором главного элемента

Схема с выбором главного элемента является одной из модификаций метода Гаусса.

Среди ведущих элементов могут оказаться очень маленькие по абсолютной величине. При делении на такие ведущие элементы получается большая погрешность округления (вычислительная погрешность).

Идеей метода Гаусса с выбором главного элемента является такая перестановка уравнений, чтобы на k -ом шаге исключения ведущим элементом a_{ii} оказывался наибольший по модулю элемент k -го столбца.

Т.е. на очередном шаге k в уравнениях, начиная от k до последнего ($i=k, k+1, \dots, n$) в столбце k выбирают максимальный по модулю элемент и строки i и k меняются местами. Это выбор главного элемента «по столбцу».

Выбор главного элемента «по строке» - на очередном шаге k в строке k , начиная со столбца k ($j=k, k+1, \dots, n$) справа выбирается максимальный по модулю элемент. Столбцы j и k меняются местами.



Прямые методы. Метод Гаусса с выбором главного элемента

Выбор главного элемента «по столбцу»:

$$\begin{pmatrix} 2 & 2 & 4 \\ 6 & 9 & 3 \\ 3 & 8 & 2 \end{pmatrix} \begin{pmatrix} 7 \\ 5 \\ 1 \end{pmatrix} \rightarrow \begin{pmatrix} 6 & 9 & 3 \\ 2 & 2 & 4 \\ 3 & 8 & 2 \end{pmatrix} \begin{pmatrix} 5 \\ 7 \\ 1 \end{pmatrix} \rightarrow \begin{pmatrix} 6 & 9 & 3 \\ 1 & -3 & \\ -3,5 & -0,5 & \end{pmatrix} \begin{pmatrix} 5 \\ -5,5 \\ 1,5 \end{pmatrix} \\ \rightarrow \begin{pmatrix} 6 & 9 & 3 \\ -3,5 & -0,5 & \\ 1 & -3 & \end{pmatrix} \begin{pmatrix} 5 \\ 1,5 \\ -5,5 \end{pmatrix}$$

Выбор главного элемента «по строке»:

$$\begin{pmatrix} 2 & 2 & 4 \\ 6 & 1 & 3 \\ 3 & 8 & 2 \end{pmatrix} \begin{pmatrix} 7 \\ 5 \\ 1 \end{pmatrix} \rightarrow \begin{pmatrix} 4 & 2 & 2 \\ 3 & 1 & 6 \\ 2 & 8 & 3 \end{pmatrix} \begin{pmatrix} 7 \\ 5 \\ 1 \end{pmatrix} \rightarrow \begin{pmatrix} 4 & 2 & 2 \\ 0,5 & -4,5 & \\ -7 & -2 & \end{pmatrix} \begin{pmatrix} 7 \\ -0,25 \\ 2,5 \end{pmatrix} \\ \rightarrow \begin{pmatrix} 4 & 2 & 2 \\ -4,5 & 0,5 & \\ -2 & -7 & \end{pmatrix} \begin{pmatrix} 5 \\ -0,25 \\ 2,5 \end{pmatrix}$$

Метод Гаусса с выбором главного элемента. Пример 3

До исключения x_2 из третьего уравнения переставим уравнения системы (см. пример 2):

$$\begin{cases} 10x_1 - 7x_2 = 7 \\ 2,5x_2 + 5x_3 = 2,5 \\ -0,001x_2 + 6x_3 = 6,001 \end{cases}$$

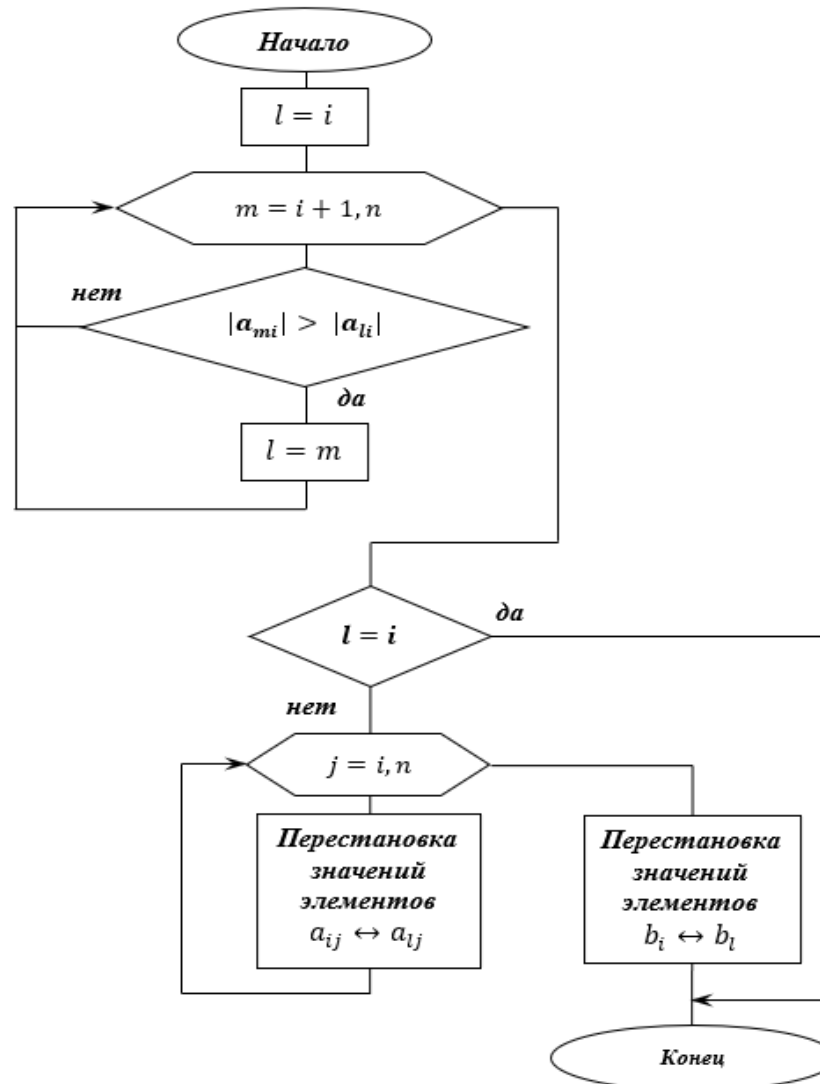
Исключим теперь x_2 из третьего уравнения, получим: $6,002x_3 = 6,002$

Отсюда находим $x_3 = 1$. Далее: $x_2 = \frac{2,5 - 5x_3}{2,5} = -1$ $x_1 = \frac{7x_2 + 7}{10} = 0$

Таким образом, в результате перестановки уравнений, т. е. *выбора наибольшего по модулю из оставшихся в данном столбце элементов*, погрешность решения в рамках данной точности исчезла.



Блок-схема метод Гаусса с выбором главного элемента



l – номер наибольшего по абсолютной величине элемента матрицы в столбце с номером i (т. е. среди элементов $a_{ii}, \dots, a_{mi}, \dots, a_{ni}$);
 m – текущий номер элемента, с которым происходит сравнение;
Выбор главного элемента осуществляется по столбцу



Погрешности решения

Решения, получаемые с помощью прямых методов, обычно содержат погрешности. Они возникают из-за погрешностей округлений при выполнении операций над числами с плавающей точкой, связанных с ограниченностью разрядной сетки машины. В ряде случаев эти погрешности могут быть значительными.

Существуют две величины, характеризующие степень отклонения полученного решения от точного:

погрешность $\Delta x = x - x^*$, где x – точное решение, x^* – решение, вычисленное по методу Гаусса.

невязка $r = Ax^* - b$, разность между левой и правой частями уравнений при подстановки в них решения x^* .

В практических расчетах, если система не является плохо обусловленной, контроль точности решения осуществляется с помощью невязки (погрешность же обычно вычислить невозможно, поскольку неизвестно точное решение).

Необходимое условие, но не достаточное, плохо обусловленной матрицы: $D \approx 0$.

Можно отметить, что метод Гаусса с выбором главного элемента в этих случаях дает малые невязки.



Определитель

Определитель треугольной матрицы: равен произведению ее диагональных элементов.

Для приведения матрицы к треугольному виду может быть использован метод исключения, т. е. прямой ход метода Гаусса. В процессе исключения элементов величина определителя не меняется. Знак определителя меняется на противоположный при перестановке его столбцов или строк.

Определитель после приведения матрицы A к треугольному виду вычисляется по формуле:

$$\det A = (-1)^k \prod_{i=1}^n a_{ii}$$

Диагональные элементы a_{ii} берутся из преобразованной (а не исходной) матрицы.

k – число перестановок строк (или столбцов) матрицы при ее приведении к треугольному виду (для получения ненулевого или максимального по модулю ведущего элемента на каждом этапе исключения).

Благодаря методу исключения можно вычислять определители порядка $n > 1000$ -го и объем вычислений будет значительно меньший, чем в проведенных ранее оценках.



Другие прямые методы

Метод прогонки используется при решении краевых задач для дифференциальных уравнений.

Схема Жордана – система приводится к диагональному виду (а не к треугольному), но облегчается обратный ход.

Метод квадратного корня используется в тех случаях, когда матрица системы является симметричной.

Метод оптимального исключения удобен при построчном вводе матрицы системы в оперативную память.

Клеточные методы могут использоваться для решения больших систем, когда матрица и вектор правых частей целиком не помещаются в оперативной памяти.



МЕТОДЫ РЕШЕНИЯ ЛИНЕЙНЫХ СИСТЕМ

ПРЯМЫЕ (ТОЧНЫЕ)

ИТЕРАЦИОННЫЕ (ПРИБЛИЖЕННЫЕ)

ИТЕРАЦИОННЫЕ МЕТОДЫ

это методы последовательных приближений.

Задается некоторое начальное приближение. Далее с помощью определенного алгоритма проводится один цикл вычислений - итерация. В результате итерации находят новое приближение. Итерации проводятся до получения решения с требуемой точностью.

Достоинства:

- требуют хранения в памяти машины не всей матрицы системы, а лишь нескольких векторов с n компонентами. Иногда элементы матрицы можно совсем не хранить, а вычислять их по мере необходимости.
- погрешности не накапливаются, поскольку точность вычислений в каждой итерации определяется лишь результатами предыдущей итерации и практически не зависит от ранее выполненных вычислений.

Недостатки:

- Алгоритмы итерационных методов более сложные по сравнению с прямыми методами.

Итерационные методы

Итерационные методы дают возможность для системы (1) построить последовательность векторов $\vec{x}^{(0)}, \vec{x}^{(1)}, \dots, \vec{x}^{(k)}$, пределом которой должно быть точное решение $\vec{x}^{(*)}$:

$$\vec{x}^{(*)} = \lim_{k \rightarrow \infty} \vec{x}^{(k)}$$

На практике построение последовательности заканчивается, как только достигается желаемая точность.

Критерии окончания итерационного процесса:

Критерий по абсолютным отклонениям, наиболее простой и часто используемый способ – это сравнение между собой соответствующих неизвестных по двум соседним итерациям (k) и (k-1):

$$\max_{1 \leq i \leq n} \left| x_i^{(k)} - x_i^{(k-1)} \right| \leq \varepsilon$$

Критерий по относительным разностям:

$$\max_{1 \leq i \leq n} \left| \frac{x_i^{(k)} - x_i^{(k-1)}}{x_i^{(k-1)}} \right| \leq \varepsilon$$

Критерий по невязке, показывающий насколько полученное приближение x^k отличается от точного решения:

$$\max_{1 \leq i \leq n} \left| r_i^{(k)} = Ax_i^{(k)} - b \right| \leq \varepsilon$$

Итерационные методы. Метод простой итерации

Рассмотрим систему линейных уравнений с невырожденной матрицей ($\det A \neq 0$):

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

Приведем систему уравнений к виду (2), выразив неизвестные x_1, x_2, x_3 соответственно из первого, второго и третьего уравнений системы (1):

$$\begin{cases} x_1 = c_{11}x_1 + c_{12}x_2 + \dots + c_{1n}x_n + d_1 \\ x_2 = c_{21}x_1 + c_{22}x_2 + \dots + c_{2n}x_n + d_2 \\ \dots \dots \\ x_n = c_{n1}x_1 + c_{n2}x_2 + \dots + c_{nn}x_n + d_n \end{cases} \quad (2)$$

Или в векторно-матричном виде: $x = Cx + D$, где x – вектор неизвестных, C – матрица коэффициентов преобразованной системы размерности $n \times n$, D – вектор правых частей преобразованной системы.

Итерационные методы. Метод простой итерации

Систему (2) представим в сокращенном виде:

$$x_i = \sum_{j=1}^n c_{ij} x_j + d_i, \quad i = 1, 2, \dots, n$$

$$c_{ij} = \begin{cases} 0, & \text{при } i = j \\ -\frac{a_{ij}}{a_{ii}}, & \text{при } i \neq j \end{cases} \quad d_i = \frac{b_i}{a_{ii}} \quad i = 1, 2, \dots, n$$

Рабочая формула метода простой итерации:

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

где k – номер итерации.

За начальное (нулевое) приближение выбирают вектор свободных членов: $x^{(0)} = D$ или нулевой вектор.

Следующее приближение: $\vec{x}^{(1)} = c\vec{x}^{(0)} + \vec{d}$, $\vec{x}^{(2)} = c\vec{x}^{(1)} + \vec{d} \dots$

$$\vec{x}^{(k)} = c\vec{x}^{(k-1)} + \vec{d}$$

Итерационные методы. Условия сходимости

Теорема. Достаточным условием сходимости итерационного процесса к решению системы при любом начальном векторе $x_i^{(0)}$ является выполнение условия преобладания диагональных элементов или доминирование диагонали:

$$|a_{ii}| \geq \sum_{j \neq i} |a_{ij}|, \quad i = 1, 2, \dots, n$$

При этом хотя бы для одного уравнения неравенство должно выполняться строго. Эти условия являются достаточными для сходимости метода, но они не являются необходимыми, т. е. для некоторых систем итерации сходятся и при нарушении этого условия.



Итерационные методы. Метод простой итерации

Теорема. Достаточным условием сходимости итерационного метода к решению системы при любом начальном векторе $x_i^{(0)}$ является требование к норме матрицы C :

$\|C\| < 1$ (по строкам, или по столбцам, или по всей матрице)

$$\|C\| = \max_{1 \leq i \leq n} \sum_{j=1}^n |c_{ij}| < 1$$

$$\|C\| = \max_{1 \leq j \leq n} \sum_{i=1}^n |c_{ij}| < 1$$

$$\|C\| = \sqrt{\sum_{i=1}^n \sum_{j=1}^n c_{ij}^2} < 1$$

Условие сходимости $\|C\| < 1$ в этом методе равносильно условию диагонального преобладания.

Итерационные методы. Метод простой итерации

Следует особо отметить, что в решении СЛАУ этим методом наиболее сложным и трудоемким является выполнение преобразования системы из вида (1) к виду (2). Эти преобразования должны быть эквивалентными, т.е. не меняющими решения исходной системы и обеспечивающие величину нормы матрицы $C < 1$. Единого рецепта для выполнения таких преобразований не существует.

Здесь в каждом конкретном случае необходимо проявлять

творчество!!!!

Достоинства метода:

- Является универсальным и простым для реализации на ЭВМ

Недостатки метода:

- Является трудоемким
- Обладает медленной скоростью сходимости



Метод простой итерации. Пример 4

Методом простых итераций с точностью $\varepsilon = 0,01$ решить систему линейных алгебраических уравнений:

$$\begin{cases} 2x_1 + 2x_2 + 10x_3 = 14 \\ 10x_1 + x_2 + x_3 = 12 \\ 2x_1 + 10x_2 + x_3 = 13 \end{cases}$$

Условие преобладания диагональных элементов не выполняется, так как $|2| < |2| + |10|$, $|1| < |10| + |1|$, $|1| < |2| + |10|$. Переставим уравнения местами так, чтобы выполнялось условие преобладания диагональных элементов:

$$\begin{cases} 10x_1 + x_2 + x_3 = 12 \\ 2x_1 + 10x_2 + x_3 = 13 \\ 2x_1 + 2x_2 + 10x_3 = 14 \end{cases}$$

Выразим из первого уравнения x_1 , из второго x_2 , из третьего x_3 :

$$\begin{cases} x_1 = -0,1x_2 - 0,1x_3 + 1,2 \\ x_2 = -0,2x_1 - 0,1x_3 + 1,3 \\ x_3 = -0,2x_1 - 0,2x_2 + 1,4 \end{cases}$$

Метод простой итерации. Пример 4

$$C = \begin{pmatrix} 0 & -0,1 & -0,1 \\ -0,2 & 0 & -0,1 \\ -0,2 & -0,2 & 0 \end{pmatrix}$$

$$d = \begin{pmatrix} 1,2 \\ 1,3 \\ 1,4 \end{pmatrix}$$

Заметим, что норма преобразованной матрицы:

$\|C\| = \max(0,2; 0,3; 0,4) = 0,4 < 1$, следовательно, условие сходимости выполнено.

Зададим начальное приближение:

$$x^0 = d = \begin{pmatrix} 1,2 \\ 1,3 \\ 1,4 \end{pmatrix}$$

Выполним расчеты по формуле: $x^{k+1} = Cx^k + d$ или:

$$x_1^{k+1} = -0,1x_2^k - 0,1x_3^k + 1,2$$

$$x_2^{k+1} = -0,2x_1^k - 0,1x_3^k + 1,3$$

$$x_3^{k+1} = -0,2x_1^k - 0,2x_2^k + 1,4$$

Метод простой итерации. Пример 4

Для первого приближения получаем:

$$x_1^1 = -0,1 \cdot 1,3 - 0,1 \cdot 1,4 + 1,2 = 0,93$$

$$x_2^1 = -0,2 \cdot 1,2 - 0,1 \cdot 1,4 + 1,3 = 0,92$$

$$x_3^1 = -0,2 \cdot 1,2 - 0,2 \cdot 1,3 + 1,4 = 0,9$$

$$\max \begin{pmatrix} |x_1^1 - x_1^0| \\ |x_2^1 - x_2^0| \\ |x_3^1 - x_3^0| \end{pmatrix} = \max \begin{pmatrix} 0,27 \\ 0,38 \\ 0,5 \end{pmatrix}$$

k	x_1^k	x_2^k	x_3^k	$\max x_i^{(k)} - x_i^{(k-1)} $
0	1,200	1,3000	1,4000	-
1	0,9300	0,9200	0,9000	0,5
2	1,0180	1,0240	1,0300	0,13
3	0,9946	0,9934	0,9916	0,0384
4	1,0015	1,0020	1,0024	0,0108
5	0,9996	0,9995	0,9993	0,0027 < ε

Приближенное решение задачи: (0,9996; 0,9995; 0,9993). Очевидно, точное решение: (1,1,1)



Метод Гаусса-Зейделя

Метод Гаусса-Зейделя является модификацией метода простой итерации и обеспечивает более быструю сходимость к решению системы

уравнений. Идея метода: при вычислении компонента $x_i^{(k+1)}$ вектора неизвестных на $(k+1)$ -й итерации используются $x_1^{(k+1)}, x_2^{(k+1)}, \dots, x_{i-1}^{(k+1)}$, уже вычисленные на $(k+1)$ -й итерации. Значения остальных компонент $x_{i+1}^{(k+1)}, x_{i+2}^{(k+1)}, \dots, x_n^{(k+1)}$ берутся из предыдущей итерации.

Так же как и в методе простых итераций строится эквивалентная СЛАУ и за начальное приближение принимается вектор правых частей (как правило, но может быть выбран и нулевой вектор): $x_i^0 = (d_1, d_2, \dots, d_n)$.

$$x_1 = c_{11}x_1 + c_{12}x_2 + \dots + c_{1n}x_n + d_1$$

$$x_2 = c_{21}x_1 + c_{22}x_2 + \dots + c_{2n}x_n + d_2$$

... ..

$$x_n = c_{n1}x_1 + c_{n2}x_2 + \dots + c_{nn}x_n + d_n$$

$$|x_1^{(k)} - x_1^{(k-1)}| \leq \varepsilon, \quad |x_2^{(k)} - x_2^{(k-1)}| \leq \varepsilon, \quad |x_3^{(k)} - x_3^{(k-1)}| \leq \varepsilon$$

Метод Гаусса-Зейделя

Достоинства метода:

- Является универсальным и простым для реализации на ЭВМ.
- Обеспечивает более быструю сходимость (по сравнению с методом простой итерации)

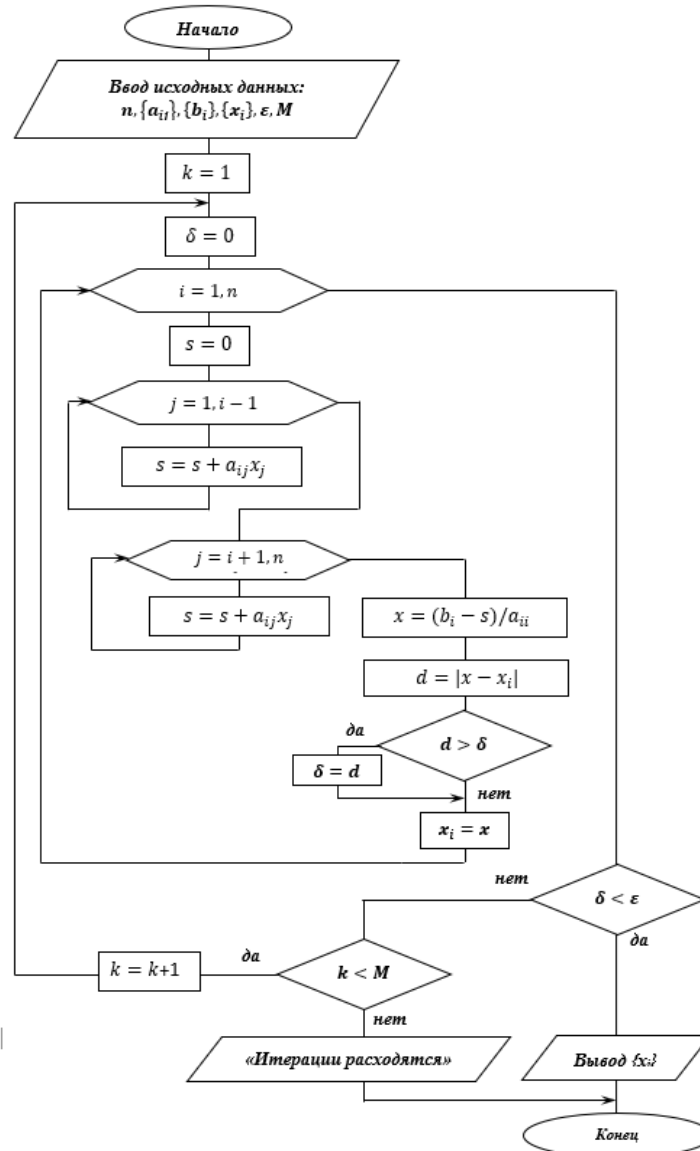
Недостатки метода:

- Является трудоемким



Блок-схема метод Гаусса-Зейделя

n – порядок матрицы,
 ε – погрешность вычислений,
 a_{ij}, b_i – коэффициенты и правые части уравнений системы,
 x_i – начальные приближения,
 M – максимально допустимое число итераций,
 k – порядковый номер итерации;
 i – номер уравнения, а также переменного, которое вычисляется в соответствующем цикле;
 j – номер элемента вида $a_{ij}x_j^{(k)}$ или $a_{ij}x_j^{(k-1)}$ в правой части соотношения.
Итерационный процесс прекращается либо при выполнении условия:
 $\max_{1 \leq i \leq n} |x_i^{(k)} - x_i^{(k-1)}| < \varepsilon$, либо при $k = M$, т.е. итерации не сходятся.



Метод Гаусса-Зейделя. Пример 5

Методом Гаусса-Зейделя с точностью $\varepsilon = 0,01$ решить систему линейных алгебраических уравнений:

$$\begin{cases} 2x_1 + 2x_2 + 10x_3 = 14 \\ 10x_1 + x_2 + x_3 = 12 \\ 2x_1 + 10x_2 + x_3 = 13 \end{cases}$$

Приведем систему $Ax = b$ к виду $x = Cx + d$:

$$\begin{cases} x_1 = -0,1x_2 - 0,1x_3 + 1,2 \\ x_2 = -0,2x_1 - 0,1x_3 + 1,3 \\ x_3 = -0,2x_1 - 0,2x_2 + 1,4 \end{cases}$$

$$C = \begin{pmatrix} 0 & -0,1 & -0,1 \\ -0,2 & 0 & -0,1 \\ -0,2 & -0,2 & 0 \end{pmatrix} \quad d = \begin{pmatrix} 1,2 \\ 1,3 \\ 1,4 \end{pmatrix}$$

Норма матрицы $\|C\| = \max(0,2; 0,3; 0,4) = 0,4 < 1$, условие сходимости выполняется.

Метод Гаусса-Зейделя. Пример 5

Зададим начальное приближение:

$$x^0 = d = \begin{pmatrix} 1,2 \\ 1,3 \\ 1,4 \end{pmatrix}$$

Выполним расчеты по формуле:

$$x_1^{k+1} = -0,1x_2^k - 0,1x_3^k + 1,2$$

$$x_2^{k+1} = -0,2x_1^{k+1} - 0,1x_3^k + 1,3$$

$$x_3^{k+1} = -0,2x_1^{k+1} - 0,2x_2^{k+1} + 1,4$$

Для первого приближения получаем:

$$x_1^1 = -0,1 \cdot 1,3 - 0,1 \cdot 1,4 + 1,2 = 0,93$$

$$x_2^1 = -0,2 \cdot 0,93 - 0,1 \cdot 1,4 + 1,3 = 0,974$$

$$x_3^1 = -0,2 \cdot 0,93 - 0,2 \cdot 0,974 + 1,4 = 1,0192$$

$$\max \begin{pmatrix} |x_1^1 - x_1^0| \\ |x_2^1 - x_2^0| \\ |x_3^1 - x_3^0| \end{pmatrix} = \max \begin{pmatrix} 0,27 \\ 0,326 \\ 0,3808 \end{pmatrix}$$

Метод Гаусса-Зейделя. Пример 5

k	x_1^k	x_2^k	x_3^k	$\max x_i^{(k)} - x_i^{(k-1)} $
0	1,200	1,3000	1,4000	-
1	0,9300	0,9740	1,0192	0,3808
2	1,0007	0,9979	1,0028	0,0707
3	0,9997	0,9998	1,0001	0,0027 < ε

Как видно, итерационный процесс завершился быстрее, чем в методе простой итерации (5 итераций).