

Краевой конкурс учебно-исследовательских и проектных работ учащихся
«Прикладные вопросы математики»

Алгебра

**Численные методы решения алгебраических уравнений и систем
уравнений**

Булычев Сергей,
МОУ «Лицей №1» г. Перми, 11 кл
Анферов Сергей Дмитриевич,
преподаватель информатики
МОУ «Лицей №1» г. Перми

Метод половинного деления (дихотомия)

Пусть мы нашли такие точки a и b что $f(a)f(b) \leq 0$, т. е. на отрезке $[a, b]$ лежит не менее одного корня уравнения. Найдем середину отрезка $x_c = (a+b)/2$ и вычислим $f(x_c)$. Из двух половин отрезка выберем ту, для которой $f(x_c)f(a)$ или $f(x_c)f(b) \leq 0$, т.е. отрезок на котором функция меняет знак. Затем новый отрезок опять делим пополам и выберем ту половину, на концах которой функция имеет разные знаки, и т. д. (рис. 1).

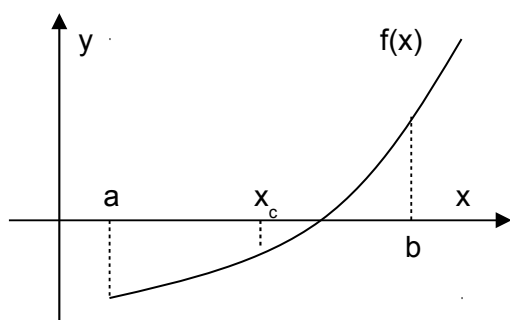


Рис. 1

Если требуется найти корень с точностью ϵ , то продолжаем деление пополам до тех пор, пока длина отрезка не станет меньше 2ϵ . Тогда середина последнего отрезка даст значение корня с требуемой точностью. Дихотомия проста и очень надежна: к простому корню она сходится для любых непрерывных функций $f(x)$, в том числе недифференцируемых; при этом она устойчива к ошибкам округления. Скорость сходимости невелика: за одну итерацию точность увеличивается примерно вдвое, т. е. уточнение трех цифр требует 10 итераций (т.к. длина отрезка, на котором лежит корень, после 10 итераций равна $1/2^{10} = 1/1024 \approx 10^{-3}$). Зато точность ответа гарантируется.

Перечислим недостатки метода.

1. Для начала расчета надо найти отрезок, на котором функция меняет знак.
2. Если в этом отрезке несколько корней, то заранее неизвестно, к какому из них сойдется процесс (хотя к одному из них сойдется).
3. Метод неприменим к корням четной кратности.
4. Для корней высокой нечетной кратности он сходится, но менее

точен и хуже устойчив к ошибкам округления, возникающим при вычислении $f(x)$.

5. Наконец, на системы уравнений дихотомия не обобщается.

Утверждение 1. С помощью данного метода невозможно найти корни чётной кратности.

Доказательство.

Чётно кратный корень это корень уравнения вида

$$(x+a)^{2n}=0, \text{ где } n - \text{ целое, } n \in [0, \infty]. \quad (2)$$

Решением этого уравнения будет корень $x=-a$ кратности $2n$. В общем виде уравнение может иметь как чётно, так и нечётно кратные корни. Можно записать общий вид уравнения имеющего $(k+m)$ только действительных корней так:

$$(x+x_1)^{2n_1}(x+x_2)^{2n_2} \dots (x+x_k)^{2n_k}(x+x_{k+1})^{2n(k+1)+1}(x+x_{k+2})^{2n(k+2)+1} \dots (x+x_{k+m})^{2n(k+m)+1}=0, \quad (3)$$

где $n_1, \dots, n(k+m) \in [0, \infty]$ – целые числа; $x_1 \neq x_2 \neq \dots \neq x_{k+m}$.

В уравнении (3) k чётно кратных и m нечётно кратных корней. Оно раскладывается на $(k+m)$ уравнений, из которых легко получаются корни. Если задать начальный отрезок $[-x_1-r, -x_1+r]$, где r – мало, и проверить условие смены знака функции на его границах, то обнаружим, что знак не меняется в силу чётности степени. А если аналогично проверить нечётно кратные корни, то получим обратную ситуацию.

Следствие 1.

Если корень имеет чётную кратность, то на границах бесконечно малого отрезка с центром в этом корне функция имеет одинаковые знаки.

Следствие 2.

Если корень имеет нечётную кратность, то на границах бесконечно малого отрезка с центром в этом корне функция имеет разные знаки.

Пусть на заданном отрезке $[a, b]$ лежит 1 корень чётной кратности, тогда в силу следствия 1 на границах отрезка знак меняться не будет, что означает

остановку выполнения итераций и недостижение необходимой точности. Если же на отрезке $[a, b]$ лежит 1 чётно кратный корень и 1 нечётно кратный корень, то чётно кратный корень будет просто игнорирован методом, т.к. условие смены знака являющееся также основным условием, с помощью которого определяется корень на текущем полуотрезке, в силу следствия 1 не выполнится. Следовательно, чётно кратный корень не может быть найден с помощью данного метода.

Утверждение 2. Если на концах начального отрезка значения функции имеют один знак, то метод может не сойтись, то есть, возможно, ни один из корней не будет найден с заданной точностью.

Доказательство.

Первым вариантом постоянства знака функции на границах отрезка является отсутствие корня на нём, поэтому исключим этот случай как тривиальный, будем считать, что на отрезке хотя бы один корень существует. Вторым вариантом – существование чётного количества корней.

Если $f(a)f(b) \geq 0$, то продолжать итерации невозможно, т.к. условие смены знака не подтверждается. Если же, тем не менее, на первом шаге не проверять условие смены знака и разделить отрезок пополам, то может возникнуть ситуация, в которой корни распределяться по чётному количеству в каждой половине отрезка. А чётное количество корней означает чётное количество пересечений оси Ox , даже если существуют кратные корни.

Следовательно, условие смены знака вновь не подтвердится для обеих половинок исходного отрезка. Следовательно, дальнейшие итерации не будут выполнены, и не будет достигнута заданная точность.

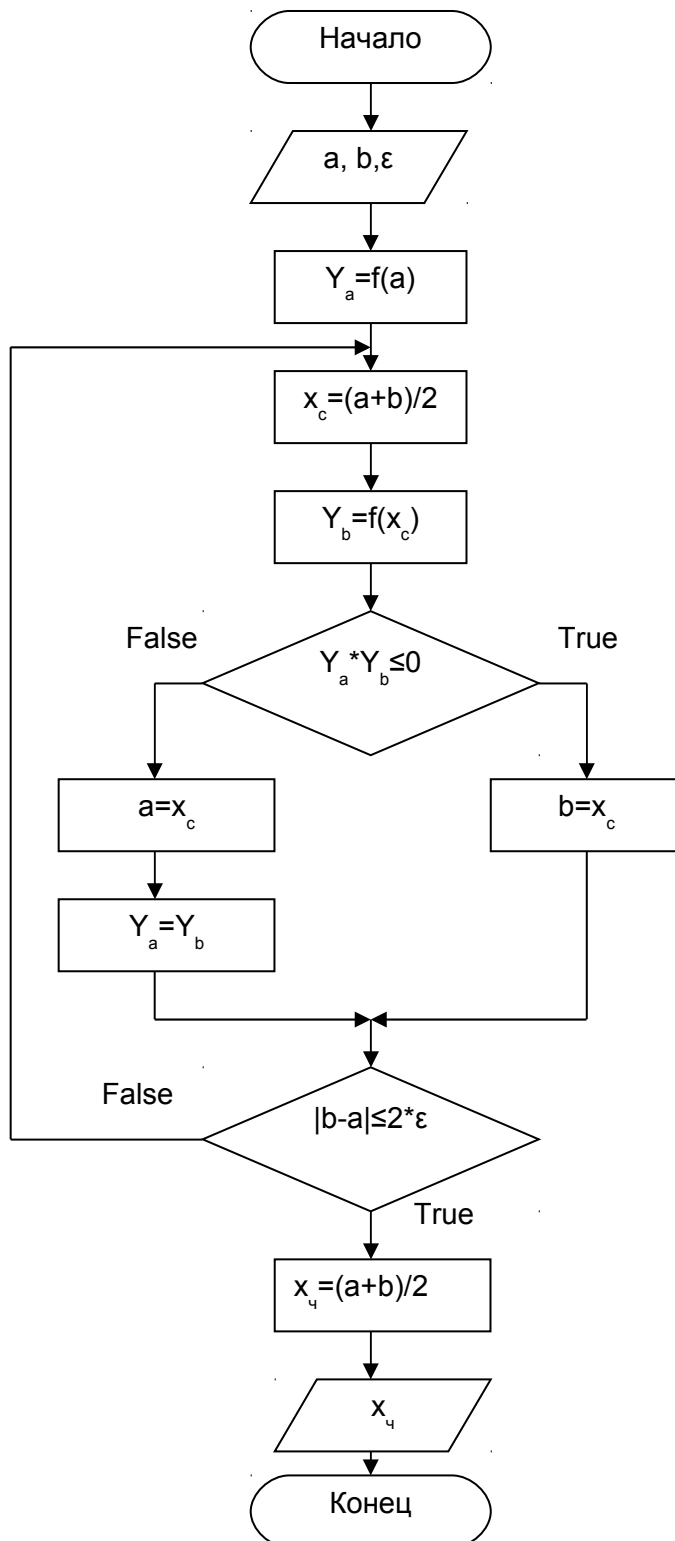
Утверждение 3. Если на концах начального отрезка значения функции имеют разные знаки, то будет найден с заданной точностью один из корней лежащих на нём.

Доказательство.

В силу утверждения 1 будем рассматривать только корни нечётной кратности. Так как функция меняет знак на концах отрезка, предположим,

$f(a) \geq 0, f(b) \leq 0$. Тогда если $f(x_c) \geq 0$, то для дальнейшего приближения выберем отрезок $[x_c, b]$, т.к. $f(b)f(x_c) \leq 0$. Если же $f(x_c) \leq 0$, то для дальнейшего приближения выберем отрезок $[a, x_c]$, т.к. $f(a)f(x_c) \leq 0$. Для второго случая, когда $f(a) \leq 0, f(b) \geq 0$ аналогично доказывается существование одного из полуотрезков, на котором функция меняет знак. Из чего следует, что после каждой итерации для одного из полуотрезков условие смены знака обязательно будет выполнено. Следовательно, нет причин для остановки итерационного процесса, который завершится лишь по достижении заданной точности.

Построим блок-схему алгоритма вычисления корня уравнения вида (1) с помощью метода дихотомии. Пусть на начальном отрезке $[a, b]$ функция меняет знак, т.е. на этом отрезке существует нечётное количество нечётно кратных корней. Пример такой функции изображён на рис. 1. Необходимо найти корень x_T с точностью ε . Будем считать x_T точным значением корня, x_q – значение корня полученное данным методом, тогда задача считается выполненной, если $x_q \in [x_T - \varepsilon, x_T + \varepsilon]$.



Дихотомия применяется тогда, когда требуется высокая надежность счета, а скорость сходимости малосущественна.

Метод касательных (Ньютона)

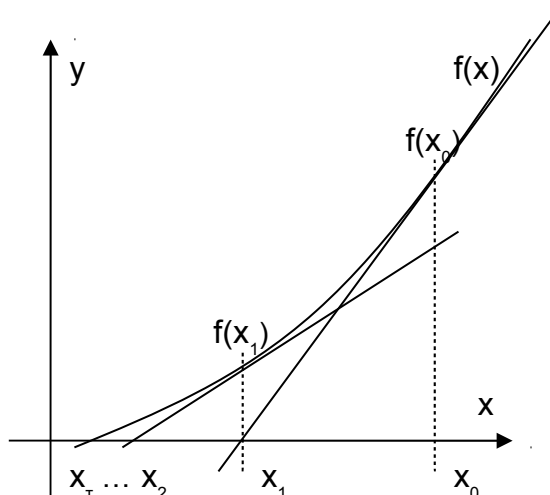


Рис. 3

Метод Ньютона называют также методом касательных и методом линеаризации. Суть метода заключается в том, что в точке приближения к функции строится касательная (Рис. 3). Следующая точка приближения – это точка пересечения полученной прямой с осью Ox . Процесс продолжается вплоть до достижения заданной точности.

Из рисунка очень легко получить итерационную формулу метода, используя геометрический смысл производной. Если $f(x)$ имеет непрерывную производную $f'(x) \neq 0$, тогда получим

$$f'(x_0) = \frac{f(x_0) - f(x_1)}{x_0 - x_1}$$

$$x_1 = x_0 - \frac{f(x_0)}{f'(x_0)}$$

Аналогично получаем x_2, x_3 , и т.д. Таким образом, можем записать общую формулу:

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)}$$

Метод Ньютона можно рассматривать, как частный случай метода простых итераций, если задать

$$\varphi(x) = x - \frac{f(x)}{f'(x)}$$

В этом случае

$$\varphi'(x) = 1 - \frac{f'(x)f'(x) - f(x)f''(x)}{(f'(x))^2} = 1 - 1 + \frac{f(x)f''(x)}{(f'(x))^2} = \frac{f(x)f''(x)}{(f'(x))^2}$$

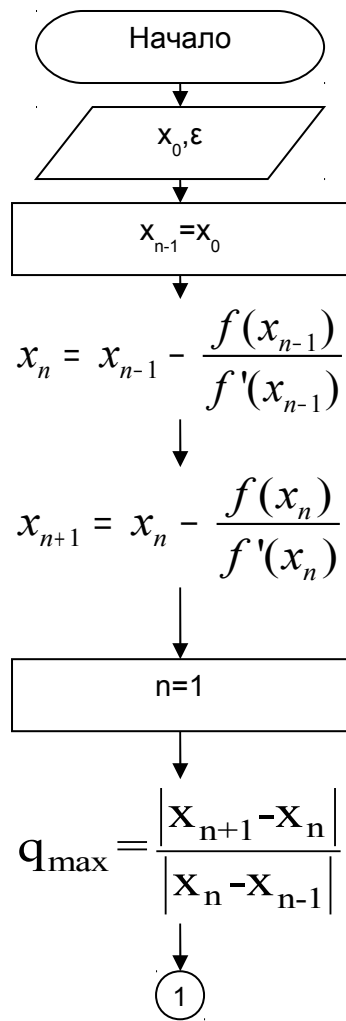
Условие сходимости метода простых итераций можно переписать для метода Ньютона следующим образом:

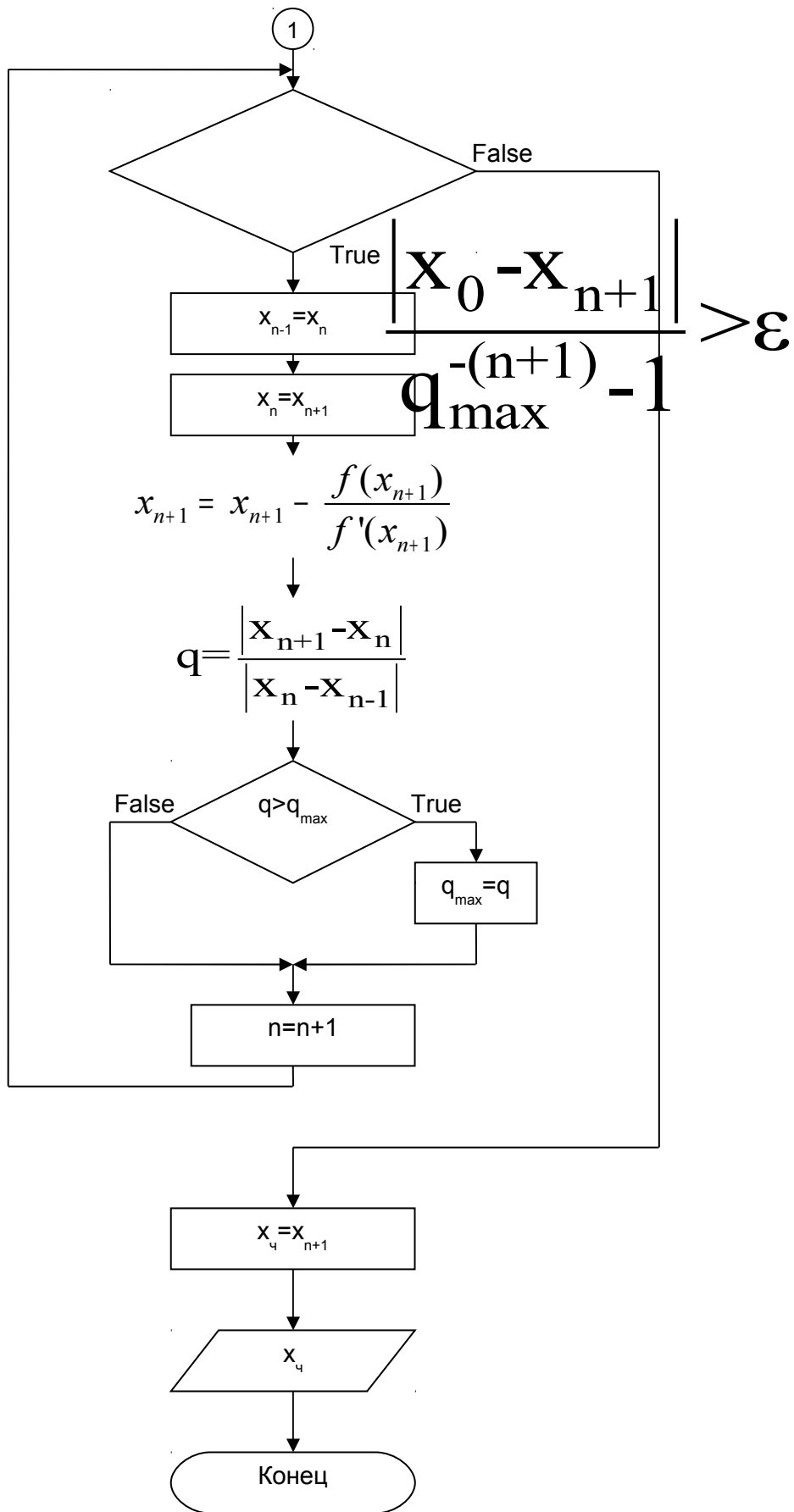
$$|\varphi'(x)| < 1$$

$$\frac{|f(x)f''(x)|}{(f'(x))^2} < 1$$

$$|f(x)f''(x)| < (f'(x))^2 \quad (14)$$

Рассуждения по поводу выбора начального приближения в методе Ньютона такие же, как и в методе простых итераций, только вместо (11) используется (14). Для данного метода также применимо (13). Напишем блок-схему алгоритма метода Ньютона.





Метод Гаусса

Наиболее известным и популярным прямым методом решения СЛАУ является метод Гаусса. Этот метод заключается в последовательном исключении неизвестных. Пусть в системе уравнений

$$\begin{cases} a_{11}^{(0)} \cdot x_1 + a_{12}^{(0)} \cdot x_2 + \dots + a_{1m}^{(0)} \cdot x_m = f_1^{(0)} \\ a_{21}^{(0)} \cdot x_1 + a_{22}^{(0)} \cdot x_2 + \dots + a_{2m}^{(0)} \cdot x_m = f_2^{(0)} \\ \dots \\ a_{m1}^{(0)} \cdot x_1 + a_{m2}^{(0)} \cdot x_2 + \dots + a_{mm}^{(0)} \cdot x_m = f_m^{(0)} \end{cases}$$

первый элемент $a_{11}^{(0)} \neq 0$. Назовем его ведущим элементом первой строки. Поделим все элементы этой строки на $a_{11}^{(0)}$ и исключим x_1 из всех последующих строк, начиная со второй, путем вычитания первой (преобразованной), умноженной на коэффициент при x_1 в соответствующей строке. Получим

$$\begin{cases} x_1 + a_{12}^{(1)} \cdot x_2 + a_{13}^{(1)} \cdot x_3 + \dots + a_{1m}^{(1)} \cdot x_m = f_1^{(1)} \\ a_{22}^{(1)} \cdot x_2 + a_{23}^{(1)} \cdot x_3 + \dots + a_{2m}^{(1)} \cdot x_m = f_2^{(1)} \\ \dots \\ a_{m2}^{(1)} \cdot x_2 + a_{m3}^{(1)} \cdot x_3 + \dots + a_{mm}^{(1)} \cdot x_m = f_m^{(1)} \end{cases}$$

Если $a_{22}^{(1)} \neq 0$, то, продолжая аналогичное исключение, приходим к системе уравнений с верхней треугольной матрицей

$$\begin{cases} x_1 + a_{12}^{(1)} \cdot x_2 + a_{13}^{(1)} \cdot x_3 + \dots + a_{1m}^{(1)} \cdot x_m = f_1^{(1)} \\ x_2 + a_{23}^{(2)} \cdot x_3 + \dots + a_{2m}^{(2)} \cdot x_m = f_2^{(2)} \\ x_3 + \dots + a_{3m}^{(3)} \cdot x_m = f_3^{(3)} \\ \dots \quad \dots \quad \dots \\ x_m = f_m^{(m)} \end{cases}$$

Из нее в обратном порядке находим все значения x_i :

$$\begin{cases} x_m = f_m^{(m)} \\ x_{m-1} = f_{m-1}^{(m-1)} - a_{m-1m}^{(m-1)} \cdot x_m \\ \dots \quad \dots \quad \dots \\ x_1 = f_1^{(1)} - a_{12}^{(1)} \cdot x_2 - a_{13}^{(1)} \cdot x_3 - \dots - a_{1m}^{(1)} \cdot x_m \end{cases}$$

Процесс приведения к системе с треугольной матрицей называется **прямым ходом**, а нахождения неизвестных – **обратным**. В случае если один из ведущих элементов равен нулю, изложенный алгоритм метода Гаусса неприменим. Кроме того, если какие-либо ведущие элементы малы, то это приводит к усилению ошибок округления и ухудшению точности счета. Поэтому обычно используется другой вариант метода Гаусса – схема Гаусса с **выбором главного элемента**. Путем **перестановки строк**, а также столбцов с соответствующей перенумерацией коэффициентов и неизвестных добиваются выполнения условия:

$$|a_{ii}^{(0)}| \geq |a_{ij}^{(0)}|, j = i+1, i+2, \dots, m;$$

т.е. осуществляется выбор первого главного элемента. Переставляя уравнения так, чтобы в первом уравнении коэффициент a_{11} **был максимальным по модулю**. Разделив первую строку на главный элемент, как и прежде, исключают x_1 из остальных уравнений. Затем для оставшихся столбцов и строк выбирают второй главный элемент и т.д.

Рассмотрим применение метода Гаусса с выбором главного элемента на примере следующей системы уравнений:

$$\begin{cases} 0.5x_2 + x_3 = 4 \\ x_1 + x_2 - x_3 = 5 \\ -2x_1 + x_3 = 6 \end{cases}$$

В первом уравнении коэффициент при $x_1=0$, во втором = 1 и в третьем = -2, т.е. максимальный по модулю коэффициент в третьем уравнении. Поэтому переставим третье и первое уравнение:

$$\begin{cases} -2x_1 + x_3 = 6 \\ 0.5x_2 + x_3 = 4 \\ x_1 + x_2 - x_3 = 5 \end{cases}$$

Исключим x_1 из второго и третьего уравнений с помощью первого. Во втором уравнении исключать не надо. Для исключения из третьего уравнения умножим первое на 0.5 и сложим с третьим:

$$\begin{cases} -2x_1 + x_3 = 6 \\ 0.5x_2 + x_3 = 4 \\ x_2 - 0.5x_3 = 8 \end{cases}$$

Рассмотрим второе и третье уравнения. Максимальный по модулю элемент при x_2 в третьем. Поэтому поместим его на место второго:

$$\begin{cases} -2x_1 + x_3 = 6 \\ x_2 - 0.5x_3 = 8 \\ 0.5x_2 + x_3 = 4 \end{cases}$$

Исключим x_2 из третьего уравнения. Для этого умножим второе на -0.5 и сложим с третьим:

$$\begin{cases} -2x_1 + x_3 = 6 \\ x_2 - 0.5x_3 = 8 \\ 1.25x_3 = 0 \end{cases}$$

Обратный ход: $x_3 = 0$, $x_2 = 8$, $x_1 = -3$.

Проверка: $0.5 \cdot 8 + 0 = 4$, $-3 + 8 - 0 = 5$, $-2 \cdot (-3) + 0 = 6$.

Такая перестановка уравнений необходима для того, чтобы уменьшить влияние ошибок округления на конечный результат.

Часто возникает необходимость в решении СЛАУ, матрицы **которые являются**

слабо заполненными, т.е. содержат много нулевых элементов. В то же время эти матрицы имеют определенную структуру. Среди таких систем выделим системы с матрицами ленточной структуры, в которых ненулевые элементы располагаются на главной диагонали и на нескольких побочных диагоналях. Для решения систем с ленточными матрицами коэффициентов вместо метода Гаусса можно использовать более эффективные методы. Например, **метод прогонки**, который мы рассмотрим позже при решении краевой задачи для обыкновенного дифференциального уравнения второго порядка.

Метод обратной матрицы

Если $\det A \neq 0$, то существует обратная матрица A^{-1} . Тогда решение СЛАУ записывается в виде: $\vec{x} = A^{-1}\vec{f}$. Следовательно, решение СЛАУ свелось к умножению известной обратной матрицы на вектор правых частей. Таким образом, задача решения СЛАУ и задача нахождения обратной матрицы связаны между собой, поэтому часто решение СЛАУ называют задачей обращения матрицы. Проблемы использования этого метода те же, что и при использовании метода Крамера: **нахождение обратной матрицы – трудоемкая операция.**

Метод Крамера

При небольшой размерности системы m ($m = 2, \dots, 5$) на практике часто используют *формулы Крамера* для решения СЛАУ:

$$x_i = \frac{\det A_i}{\det A} \quad (i = 1, 2, \dots, m).$$
 Эти формулы позволяют находить неизвестные в виде дробей, знаменателем которых является определитель матрицы системы, а числителем – **определители матриц A_i , полученных из A заменой столбца коэффициентов при вычисляемом неизвестном столбце свободных членов.** Так A_1 получается из матрицы A заменой первого столбца на столбец правых частей f .

Например, для системы двух линейных уравнений

$$x_1 + x_2 = 1$$

$$2x_1 - x_2 = 0$$

$$\det A_1 = \begin{vmatrix} 1 & 1 \\ 0 & -1 \end{vmatrix} = -1, \quad \det A_2 = \begin{vmatrix} 1 & 1 \\ 2 & 0 \end{vmatrix} = -2,$$

$$\det A = \begin{vmatrix} 1 & 1 \\ 2 & -1 \end{vmatrix} = -1 - 2 = -3, \quad x_1 = (-1)/(-3) = 1/3, \quad x_2 = (-2)/(-3) = 2/3$$

Размерность системы (т.е., число m) является главным фактором, из-за которого формулы Крамера не могут быть использованы для численного решения СЛАУ большого порядка. При непосредственном раскрытии

определителей решение системы с m неизвестными требует порядка $m! \cdot m$ арифметических операций. Таким образом, для решения системы, например, из $m = 100$ уравнений потребуется совершить 10^{158} вычислительных операций (процесс займёт примерно 10^{19} лет), что не под силу даже самым мощным современным ЭВМ

Метод обратной матрицы

Если $\det A \neq 0$, то существует обратная матрица A^{-1} . Тогда решение СЛАУ записывается в виде: $\vec{x} = A^{-1} \vec{f}$. Следовательно, решение СЛАУ свелось к умножению известной обратной матрицы на вектор правых частей. Таким образом, задача решения СЛАУ и задача нахождения обратной матрицы связаны между собой, поэтому часто решение СЛАУ называют задачей обращения матрицы. Проблемы использования этого метода те же, что и при использовании метода Крамера: **нахождение обратной матрицы – трудоемкая операция.**

Метод простой итерации или метод Якоби

Напомним, что нам требуется решить систему линейных уравнений, которая в матричном виде записывается как:

$$\mathbf{A} \vec{x} = \vec{f},$$

где

$$\mathbf{A} = \begin{pmatrix} a_{11} & a_{12} & \dots & a_{1m} \\ a_{21} & a_{22} & \dots & a_{2m} \\ \dots & \dots & \dots & \dots \\ a_{m1} & a_{m2} & \dots & a_{mm} \end{pmatrix}, \quad \vec{f} = \begin{pmatrix} f_1 \\ f_2 \\ \dots \\ f_{m-1} \\ f_m \end{pmatrix}, \quad \vec{x} = \begin{pmatrix} x_1 \\ x_2 \\ \dots \\ x_{m-1} \\ x_m \end{pmatrix}.$$

Предположим, что диагональные элементы матриц A исходной системы не равны 0 ($a_{ii} \neq 0$, $i = 1, 2, \dots, n$). Разрешим первое уравнение системы относительно x_1 , второе относительно x_2 и т.д. Получим следующую эквивалентную систему, записанную в скалярном виде:

$$\begin{aligned}
x_1 &= \frac{1}{a_{11}} (f_1 - (a_{12}x_2 + a_{13}x_3 + \dots + a_{1m}x_m)) \\
x_2 &= \frac{1}{a_{22}} (f_2 - (a_{21}x_1 + a_{23}x_3 + \dots + a_{2m}x_m)) \\
&\dots\dots\dots \\
x_m &= \frac{1}{a_{mm}} (f_m - (a_{m1}x_1 + a_{m2}x_2 + \dots + a_{m,m-1}x_{m-1}))
\end{aligned} \tag{1}$$

Теперь, задав нулевое приближение $x_i^{(0)}$, по рекуррентным соотношениям (1) можем выполнять итерационный процесс, а именно:

(2)

Аналогично находятся следующие приближения $x_i^{(2)}$, где в (2) вместо $x_i^{(0)}$ необходимо подставить $x_i^{(1)}$.

Или в общем случае:

$$\begin{aligned}
x_1^{(k+1)} &= \frac{1}{a_{11}} (f_1 - (a_{12}x_2^{(k)} + a_{13}x_3^{(k)} + \dots + a_{1m}x_m^{(k)})) \\
x_2^{(k+1)} &= \frac{1}{a_{22}} (f_2 - (a_{21}x_1^{(k)} + a_{23}x_3^{(k)} + \dots + a_{2m}x_m^{(k)})) \\
&\dots\dots\dots \\
x_m^{(k+1)} &= \frac{1}{a_{mm}} (f_m - (a_{m1}x_1^{(k)} + a_{m2}x_2^{(k)} + \dots + a_{m,m-1}x_{m-1}^{(k)}))
\end{aligned} \tag{3}$$

$$x_i^{(k+1)} = \frac{1}{a_{ii}} (f_i - \sum_{\substack{j=1 \\ j \neq i}}^m a_{ij} x_j^{(k)})$$

или

Условие окончания итерационного процесса $\max_i |x_i^{(k+1)} - x_i^{(k)}| < \varepsilon$.

Достаточное условие сходимости: Если выполнено условие диагонального

$$|a_{ii}| > \sum_{j=1, j \neq i}^n |a_{ij}|, \quad i = 1, 2, \dots, m.$$

преобладания, т.е. $|a_{ii}| > \sum_{j=1, j \neq i}^n |a_{ij}|$, то итерационный процесс (3) сходится при любом выборе начального приближения. Если исходная система уравнений не удовлетворяет условию сходимости, то ее приводят к виду с диагональным преобладанием.

Выбор начального приближения влияет на количество итераций, необходимых для получения приближенного решения. Наиболее часто в качестве начального приближения берут $x_i^{(0)} = \beta_i = f_i / a_{ii}$ или $x_i^{(0)} = 0$.

Замечание. Указанное выше условие сходимости является достаточным, т.е. если оно выполняется, то процесс сходится. Однако процесс может сходиться и при отсутствии диагонального преобладания, а может и не сойтись.

Пример.

Решить систему линейных уравнений с точностью $\varepsilon = 0.01$:

$$\begin{cases} 8x_1 + 4x_2 + 2x_3 = 10 \\ 3x_1 + 5x_2 + x_3 = 5 \\ 3x_1 - 2x_2 + 10x_3 = 4 \end{cases}$$

	8	4	2		10	x_1	
A =	3	5	1	\vec{f} =	5	=	x_2
	3	-2	10		4	x_3	

Решение прямыми методами, например, обратной матрицей, даёт решение:

$$x_1 = 1.308, \quad x_2 = 0.346, \quad x_3 = 0.158$$

Найдем решение методом простой итерации. Проверяем условие диагонального преобладания: $|8| > |4| + |2|$, $|5| > |3| + |1|$, $|4| > |-2| + |3|$.

Приводим систему уравнений к виду (1):

$$\begin{cases} x_1 = -0.5x_2 - 0.25x_3 + 1.25 \\ x_2 = -0.6x_1 - 0.2x_3 + 1 \\ x_3 = -0.3x_1 + 0.2x_2 + 0.4 \end{cases} \quad \begin{cases} x_1^{(k+1)} = -0.5x_2^{(k)} - 0.25x_3^{(k)} + 1.25 \\ x_2^{(k+1)} = -0.6x_1^{(k)} - 0.2x_3^{(k)} + 1 \\ x_3^{(k+1)} = -0.3x_1^{(k)} + 0.2x_2^{(k)} + 0.4 \end{cases}$$

Начальное приближение $x_i^{(0)} = 0$. Дальнейшие вычисления оформим в виде таблицы:

k	x ₁	x ₂	x ₃	ТОЧНОСТЬ
0	0	0	0	
1	1.250	1.000	0.400	1.2500
2	0.650	0.170	0.225	0.8300
3	1.109	0.565	0.239	0.4588
			
4	0.908	0.287	0.180	0.2781
5	1.061	0.419	0.185	0.1537
6	0.994	0.326	0.165	0.0931
7	1.046	0.370	0.167	0.0515
8	1.023	0.594	0.160	0.2235
9	0.913	0.582	0.212	0.1101
10	0.906	0.505	0.242	0.0764
11	0.937	0.495	0.229	0.0305
12	0.945	0.516	0.218	0.0210
			
13	0.937	0.523	0.220	0.0077

Здесь $x_1^{(1)} = 1.25,$
 $x_2^{(1)} = 1,$
 $x_3^{(1)} = 0.4,$

$$\max \begin{pmatrix} |x_1^{(1)} - x_1^{(0)}| \\ |x_2^{(1)} - x_2^{(0)}| \\ |x_3^{(1)} - x_3^{(0)}| \end{pmatrix} = \max \begin{pmatrix} 1.25 \\ 1 \\ 0.4 \end{pmatrix} = 1.25 > \varepsilon = 0.01,$$

$$\begin{aligned} x_1^{(2)} &= -0.5 \cdot 1 - 0.25 \cdot 0.4 + 1.25 = 0.65, \\ x_2^{(2)} &= -0.6 \cdot 1.25 - 0.2 \cdot 0.4 + 1 = 0.17, \\ x_3^{(2)} &= -0.3 \cdot 1.25 + 0.2 \cdot 1 + 0.4 = 0.225, \end{aligned} \quad \max \begin{pmatrix} |x_1^{(2)} - x_1^{(1)}| \\ |x_2^{(2)} - x_2^{(1)}| \\ |x_3^{(2)} - x_3^{(1)}| \end{pmatrix} = \max \begin{pmatrix} 0.6 \\ 0.83 \\ 0.175 \end{pmatrix} > \varepsilon,$$

$$\begin{aligned}
x_1^{(3)} &= -0.5 \cdot 0.17 - 0.25 \cdot 0.225 + 1.25 = 1.109 \\
x_2^{(3)} &= -0.6 \cdot 0.65 - 0.2 \cdot 0.225 + 1 = 0.565 \\
x_3^{(3)} &= -0.3 \cdot 0.65 + 0.2 \cdot 0.17 + 0.4 = 0.239
\end{aligned}
\quad , \quad
\max \begin{pmatrix} |x_1^{(2)} - x_1^{(3)}| \\ |x_2^{(2)} - x_2^{(3)}| \\ |x_3^{(2)} - x_3^{(3)}| \end{pmatrix} = \max \begin{pmatrix} 0.459 \\ 0.395 \\ 0.014 \end{pmatrix} > \varepsilon,$$

И т.д., пока не получим, в последнем столбце величину меньшую 0.01, что произойдет на 13 – ой итерации.

Следовательно, приближенное решение имеет вид: $x_1 = 0.937, x_2 = 0.523, x_3 = 0.22$

Метод Гаусса – Зейделя

Расчетные формулы имеют вид:

$$x_i^{(k+1)} = \frac{1}{a_{ii}} \left[f_i - \sum_{j=1}^{i-1} a_{ij} x_j^{(k+1)} - \sum_{j=i+1}^n a_{ij} x_j^{(k)} \right]$$

т.е. для подсчета i -й компоненты $(k+1)$ -го приближения к искомому вектору используется уже вычисленное на этом, т.е. $(k+1)$ -м шаге, новые значения первых $i-1$ компонент.

Подробные формулы имеют вид:

$$x_1^{(k+1)} = \frac{1}{a_{11}} \left(f_1 - a_{12} x_2^{(k)} - a_{13} x_3^{(k)} - \dots - a_{1m} x_m^{(k)} \right)$$

$$x_2^{(k+1)} = \frac{1}{a_{22}} \left(f_2 - a_{21} x_1^{(k+1)} - a_{23} x_3^{(k)} - a_{24} x_4^{(k)} - \dots - a_{2m} x_m^{(k)} \right)$$

.....

$$x_i^{(k+1)} = \frac{1}{a_{ii}} \left(f_i - a_{i1} x_1^{(k+1)} - a_{i2} x_2^{(k+1)} - \dots - a_{i,j-1} x_{j-1}^{(k+1)} - a_{i,j+1} x_{j+1}^{(k)} - \dots - a_{im} x_m^{(k)} \right)$$

.....

$$x_m^{(k+1)} = \frac{1}{a_{m1}} \left(f_m - a_{m1} x_1^{(k+1)} - a_{m2} x_2^{(k+1)} - \dots - a_{m,m-1} x_{m-1}^{(k+1)} \right)$$

Достаточное условие сходимости этого метода такое же, как и для метода простой итерации, т.е. диагональное преобладание:

$$|a_{ii}| > \sum_{\substack{j=1 \\ j \neq i}}^m |a_{ij}|, \quad i = 1, 2, \dots, m.$$

Начальное приближение:

$$x_i^{(0)} = 0$$

Найдем решение предыдущей системы уравнений методом Гаусса – Зейделя.

$$\begin{cases} 8x_1 + 4x_2 + 2x_3 = 10 \\ 3x_1 + 5x_2 + x_3 = 5 \\ 3x_1 - 2x_2 + 10x_3 = 4 \end{cases}$$

Расчетные формулы:

$$\begin{cases} x_1^{(k+1)} = -0.5x_2^{(k)} - 0.25x_3^{(k)} + 1.25 \\ x_2^{(k+1)} = -0.6x_1^{(k+1)} - 0.2x_3^{(k)} + 1 \\ x_3^{(k+1)} = -0.3x_1^{(k+1)} + 0.2x_2^{(k+1)} + 0.4 \end{cases}$$

k	x1	x2	x3	точность
0	0	0	0	
1	1.2500	0.2500	0.075	1.2500
2	1.1060	0.3210	0.132	0.1438
3	1.0560	0.3400	0.151	0.0500
4	1.0420	0.3440	0.156	0.0139
5	1.0390	0.3460	0.157	0.0036

$$\begin{aligned}
 x_1^{(1)} &= -0.5 \cdot 0 - 0.25 \cdot 0 + 1.25 = 1.25, \\
 x_2^{(1)} &= -0.6 \cdot 1.25 - 0.2 \cdot 0 + 1 = 0.25, \\
 x_3^{(1)} &= -0.3 \cdot 1.25 + 0.2 \cdot 0.25 + 0.4 = 0.075,
 \end{aligned}
 \quad
 \max \begin{pmatrix} |x_1^{(1)} - x_1^{(0)}| \\ |x_2^{(1)} - x_2^{(0)}| \\ |x_3^{(1)} - x_3^{(0)}| \end{pmatrix} = \max \begin{pmatrix} 1.25 \\ 0.25 \\ 0.075 \end{pmatrix} = 1.25 > \varepsilon = 0.01,$$

$$\max \begin{pmatrix} |x_1^{(2)} - x_1^{(1)}| \\ |x_2^{(2)} - x_2^{(1)}| \\ |x_3^{(2)} - x_3^{(1)}| \end{pmatrix} = \max \begin{pmatrix} 0.144 \\ 0.071 \\ 0.057 \end{pmatrix} = 0.144 > \varepsilon,$$

Из таблицы видно, что нужная точность достигнута уже на 5-ой итерации вместо 13-ой по методу простой итерации и значения корней более близки к значениям, полученным методом обратной матрицы. $x_1 = 1.039$, $x_2 = 0.346$, $x_3 = 0.157$