

**Министерство высшего и среднего специального образования  
Республики Узбекистан**

**Наманганский инженерно-педагогический институт**

***Кафедра «Информатика и информационные технологии»***

```
function F(x : real) : real;
begin
    F:=.....
end;
begin
    L1: readln('a, b=', a, b);
    if f(a)*f(b)>0 then goto L1;
    readln('eps=', eps);
    L2:C:=(a+b)/2;
    if F(a)*F(c)<0 then b:=c else a:=c;
    if Abs(b-a)>eps then goto L2;
    c:=(a+b)/2;
    writeln ('төнглама ечими= ',c)
end.
```

# *Методическое пособие*

**по дисциплине  
«Численные методы и алгоритмы»**

Данное методическое пособие предназначено для работы со студентами, обучающимися по направлению 5140900 - по (информатика и информационные технологии).

Методическое пособие может быть использовано студентами, изучающими дисциплину «Численные методы, алгоритмы и ППП», магистрами и преподавателями.

Составитель:

асс. Л. Зинина

Обработанное и отредактированное методическое пособие рассмотрено и одобрено на заседании №\_\_\_\_ кафедры «Информатика и информационные технологии» от «\_\_\_\_»\_\_\_\_200\_\_\_\_ г.

Рекомендовано к печати, получив положительный отзыв научно-методического совета Нам ИПИ (Протокол №\_\_\_\_ от «\_\_\_\_»\_\_\_\_200\_\_\_\_ г.).

## **Введение**

В настоящее время в свете Национальной программы по подготовке кадров развитие материально-технической, компьютерной и информационной базы является одним из приоритетных направлений реформирования всех видов образования в Республике Узбекистан. Это само по себе предполагает максимально широкое внедрение информационных технологий всех видов в образовательный процесс.

Очевидной становится актуальность изучения такой дисциплины, как «Численные методы и алгоритмы». Изучение данных методов и реализация их с помощью ЭВМ являются главной задачей студентов, изучающих данную дисциплину.

В настоящем пособии излагаются теоретические сведения об основных численных методах решения уравнений и систем уравнений различного типа.

В первой части пособия приведены численные методы решения алгебраических уравнений и систем уравнений. Во второй части приведены численные методы решения дифференциальных уравнений.

Данное методическое пособие может быть использовано как в дополнение к курсу лекций, так и для самостоятельного изучения дисциплины.

# I. Решение алгебраических уравнений

## 1. Решение нелинейных уравнений с помощью численных методов

### 1.1. Виды нелинейных уравнений

Довольно часто в инженерной практике встречаются задачи, связанные с необходимостью отыскания корней нелинейных уравнений. Такие задачи обычно возникают как элементарные составляющие при решении различных технических и научных проблем.

Рассмотрим в качестве примера процедуру выбора рабочего режима транзисторного усилительного каскада. Предположим, что зависимость коэффициента усиления каскада от тока эмиттера транзистора  $i_e$  может быть представлена формулой

$$K(i_e) = a_0 + a_1 i_e + a_2 i_e^2 + a_3 i_e^3 + a_4 i_e^4 + a_5 i_e^5,$$

где  $a_1, a_2, a_3, a_4, a_5$  – известные постоянные. Если необходимо, чтобы транзистор обеспечивал требуемый коэффициент усиления  $K_{mp}$ , то рабочий ток  $i_e$  должен выбираться таким образом, чтобы выполнялось равенство

$$K(i_e) = K_{mp} \quad \text{или} \quad a_0 + a_1 i_e + a_2 i_e^2 + a_3 i_e^3 + a_4 i_e^4 + a_5 i_e^5 = K_{mp},$$

Последняя запись по сути дела является уравнением, которое можно представить в следующем виде:

$$a_0 + a_1 i_e + a_2 i_e^2 + a_3 i_e^3 + a_4 i_e^4 + a_5 i_e^5 - K_{mp} = 0,$$

Решение этого уравнения и дает искомое значение тока  $i_e$ .

Как видно, задача выбора эмиттерного тока транзистора свелась к решению уравнения вида

$$f(x) = 0, \quad (1.1.1)$$

где роль неизвестного  $x$  играет ток  $i_e$ , а  $f(x)$  является некоторой функцией этого неизвестного. Подобным же образом к решению уравнения могут быть сведены многие проектные и исследовательские задачи, в которых необходимо найти значение какого-либо параметра по известной зависимости.

Следует отметить, что обычно характеристики реальных технических устройств зависят не от одного, а от нескольких параметров. В этом случае проблема выбора параметров приводит к необходимости решения системы уравнений со многими неизвестными. Примером такого рода задачи является выбор сопротивлений резисторов в схеме многокаскадного усилителя. Однако далее мы будем рассматривать только задачи, связанные с решением одного нелинейного уравнения с одним неизвестным.

Очевидно, что способ решения уравнения определяется его видом.

В математике принято подразделять нелинейные уравнения на алгебраические и трансцендентные. К алгебраическим относят уравнения, в которых функция  $f(x)$  является степенным многочленом, то есть:

$$a_0 + a_1 x + a_2 x^2 + a_3 x^3 + a_4 x^4 + \dots + a_n x^n = 0, \quad (1.1.2)$$

где  $a_1, a_2, a_3, a_4, \dots, a_n$  – коэффициенты, а  $n$  – целое число, соответствующее максимальной степени многочлена. Подобное уравнение было рассмотрено выше.

К трансцендентным относят уравнения, содержащие трансцендентные функции, то есть показательную, логарифмическую, тригонометрические функции.

Например, трансцендентными являются следующие уравнения:

$$\sin(2x) \frac{2,1x+1}{0,3x+1} - 0,4x^2 = 0, \quad (1.1.3)$$

$$2^{0,1x} - 6 \lg(44 - x) + 5,5 \sin(x) = 0. \quad (1.1.4)$$

Свойства алгебраических и трансцендентных уравнений существенно различаются. Поэтому различны и подходы к их решению.

Для алгебраических уравнений до четвертого порядка включительно известны прямые точные методы решения. Кроме того, количество корней можно определить по степени многочлена, а их характер по знакам коэффициентов этого многочлена.

Для трансцендентных уравнений общих приемов решения кроме приближенных не существует. Количество корней такого уравнения обычно невозможно определить заранее по его виду. Трансцендентное уравнение может не иметь ни одного вещественного корня, иметь счетное количество корней или бесконечное множество корней. Из двух приведенных выше уравнений первое имеет семь вещественных корней, а второе – пять корней.

К сожалению, во многих практических важных случаях, когда уравнение имеет сложный вид, аналитически его точное решение найти не удается. Отсутствуют методы решения в общем виде алгебраических уравнений высоких степеней. Для трансцендентных уравнений точное решение найти можно в немногих самых простых случаях.

Если решение нельзя найти в явном виде, то для отыскания корня используют другие методы. Например, приближенное решение можно получить методом последовательных приближений. Сравнительно легко (но и весьма неточно) корни уравнения определяются графически – достаточно лишь для уравнения  $f(x) = 0$  построить график функции  $y = f(x)$  и найти точки пересечения кривой с осью абсцисс, в которых эта функция равна нулю. Наконец, корень уравнения можно попытаться определить "методом подбора".

Однако ни один из перечисленных подходов нельзя считать достаточно эффективным при решении инженерных и научных задач на ЭВМ. Более предпочтительны способы, обеспечивающие одновременно как оперативность получения результата, так и высокую точность.

Второе важное требование к методу – универсальность, то есть способность находить решения для разных видов уравнений. В особенности эти требования должны соблюдаться в специальных пакетах программ,

предназначенных для выполнения большого объема расчетов, например, в системах автоматизированного проектирования (САПР).

В связи с этим для решения нелинейных уравнений на ЭВМ широко применяются специальные методы, которые относятся к методам вычислительной математики. На их основе создано большое число алгоритмов, различающихся сложностью и эффективностью.

Когда говорят о методах решения нелинейных уравнений на ЭВМ, то подразумевают в первую очередь итерационные методы. Главным признаком итерационного метода является многократное повторение одного и того же набора действий для получения результата (в переводе с англ. *iteration* – повторение).

В основе итерационного метода лежит итерационная, то есть повторяющаяся процедура. Процедура эта строится таким образом, что после каждого ее выполнения производится очередное приближение к корню. Можно сказать, что итерационный метод несколько напоминает отыскание корня подбором, однако этот подбор производится не "наугад", а по вполне определенному алгоритму. Основные особенности итерационных методов будут рассмотрены позже – при описании самих методов.

При решении практических задач обычно приходится проводить предварительное исследование уравнения до его решения. Дело в том, что если уравнение не удается решить аналитически, то заранее трудно определить, сколько оно имеет корней и какова их природа – сколько из них комплексных или вещественных, сколько отрицательных или положительных. Поиск корней "наугад" без предварительного исследования чреват тем, что правильный ответ так и не будет найден. Кроме того, зачастую некоторые корни не имеют физического смысла, и поэтому нет необходимости определять их точные значения.

Например, поведение резонансной электрической или механической системы обычно описывается так называемым характеристическим уравнением, корни которого соответствуют значениям ее собственных частот.

Сложные объекты, как правило, имеют множество собственных частот, что усложняет анализ. Однако в ходе предварительного исследования уравнения можно, не вычисляя точных значений всех корней, сразу выбрать из них те, которые представляют наибольший интерес.

## 1.2. Графическое исследование уравнения

Как было указано в п. 1.1, примерное положение корней уравнения  $f(x)=0$  на числовой оси легко определить, построив график функции  $y=f(x)$ . Точки пересечения кривой  $y=f(x)$  с осью абсцисс, где  $y=0$ , и будут соответствовать искомым корням.

Отделение корней также нередко выполняют с помощью табличного представления зависимости  $f(x)$ . Для этого формируют таблицу, в которую заносят ряд последовательно расположенных на оси  $x$  точек  $x_i$  и вычисленные в них значения левой части уравнения  $f(x_i)$ .

Затем в таблице выбирают те пары рядом расположенных точек, между которыми функция  $f(x)$  меняет свой знак. При этом для обнаружения корня по сути дела используется тот же признак, что и при графическом исследовании – изменение знака функции.

С целью облегчения поиска корней процедуру вычисления нередко оформляют в виде программы на ЭВМ, включая в ее алгоритм не только вычисление значений  $x_i$  и  $f(x_i)$ , но и автоматическое выявление тех отрезков, в которых предположительно должны находятся корни уравнения.

Однако пользоваться подобными процедурами автоматического отделения корней следует осторожно. Дело в том, что смена знака функции на некотором отрезке  $x_i \leq x \leq x_{i+1}$  не является надежным признаком существования корня.

Во-первых,  $f(x)$  может изменить свой знак в точке разрыва. Во-вторых, даже если функция  $f(x)$  непрерывна, изменение ее знака на рассматриваемом отрезке может быть обусловлено не одним, а несколькими корнями, например, тремя или пятью. И, наоборот, совпадение знаков функции  $f(x)$  на краях отрезка не является доказательством отсутствия корней. К примеру, в случае двух корней на отрезке функция дважды переходит через точки  $y=0$  и дважды меняет свой знак на обратный. Или имеется так называемый кратный корень, когда  $f(x)$  не пересекает, а только касается оси  $x$  в некоторой точке.

Из вышесказанного следует, что табличное отделение корней желательно проводить, выбирая как можно более малый шаг изменения аргумента, и сопровождать его графическим исследованием.

### 1.3. Модификация табличного способа

Несложную итерационную процедуру для отыскания приближенного решения уравнения можно построить на основе рассмотренного выше табличного способа отделения корней. Рассмотрим ее.

Предположим, в результате предварительного исследования определен отрезок  $[a, b]$ , содержащий только один корень уравнения  $f(x)=0$ . Разобьем этот отрезок на некоторое количество равных частей. Для этого возьмем, к примеру,  $N-1=99$  точек, расположенных с постоянным шагом  $\Delta x=(b-a)/100$  на  $[a, b]$  и разделим его таким образом на  $N=100$  частей.

Вычислим в этих точках значения функции  $f(x)$ . Из полученных ста новых отрезков выберем тот, в котором находится корень. Его легко определить по перемене знака  $f(x)$  при переходе от одной точки к другой.

Для дальнейшего уточнения положения корня на числовой оси описанные действия можно повторять многократно, построив, таким образом, на их основе итерационную процедуру. Действительно, на первом итерационном шаге отрезок, в котором заключен корень, уменьшается в  $N=100$  раз.

Уменьшив на втором шаге отрезок еще в сто раз, получим сокращение длины уже в  $N \times N=10000$  раз при суммарном количестве расчетных точек  $N+N=198$ . На третьем шаге общее сокращение достигнет  $100 \times 100 \times$

$100 = 1000000$  раз, а количество расчетных точек возрастет до  $N+N+N=297$ . Общий же коэффициент сужения на  $K$ -м шаге составит  $N^K$ .

Процедуру сжатия отрезка, содержащего корень, следует остановить, если на очередном шаге будет получено значение  $f(x)$  достаточно близкое к нулю. Последнее означает, что найдено приближенное решение уравнения, соответствующее  $f(x) \approx 0$ . Условие прекращения итерационной процедуры обычно записывают в виде

$$|f(x)| \leq \delta, \quad (1.1.5)$$

где  $\delta$  – некоторое малое положительное число. При выборе  $\delta$  руководствуются требованиями к точности решения уравнения и порядком величины  $f(x)$ .

#### 1.4. Метод половинного деления

Другие названия: метод бисекции, метод дихотомии (от греч. δίχα – на две части и τομή – сечение).

Метод половинного деления можно рассматривать как дальнейшее усовершенствование описанной выше процедуры поиска корня уравнения. Отличие метода половинного деления состоит в том, что отрезок на каждом шаге разбивается не на любое произвольное число частей  $N$ , а делится только на две части, то есть  $N=2$ .

Графически процедура поиска корня уравнения  $f(x)=0$  методом половинного деления показана на рис. 1.1.

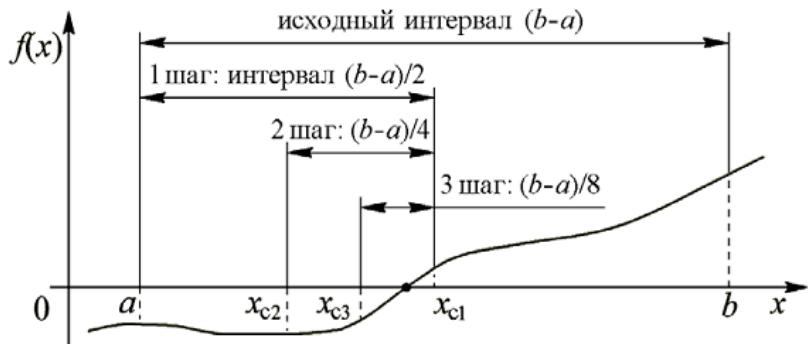


Рис. 1.1. Метод половинного деления

Метод включает следующие операции (см. рис. 1.2). Вначале на концах исходного отрезка  $[a, b]$ , содержащего корень, вычисляют значения функции  $f(a)$  и  $f(b)$ . Затем находят точку, делящую  $[a, b]$  на две равные части, по итерационной формуле

$$x_c = (a+b)/2 \quad (1.1.6)$$

и вычисляют значение функции  $f(x_c)$ . Далее по перемене знака функции выбирают ту половину  $[a, b]$ , в которой расположен корень.

Если знаки  $f(x_c)$  и  $f(a)$  совпадают, то в дальнейшем полагают  $a=x_c$  и  $f(a)=f(x_c)$ . Если же, напротив, знаки  $f(x_c)$  и  $f(a)$  различаются, а знаки  $f(x_c)$  и  $f(b)$  совпадают, то полагают  $b=x_c$  и  $f(b)=f(x_c)$ . В результате этих действий получают новый отрезок, содержащий корень. Этот отрезок имеет длину в два раза меньше, чем исходный.

Точно так же, как и в предыдущем случае, если очередное рассчитанное значение  $f(x)$  достаточно близко к нулю, вычисления прекращаются. Иначе процесс половинного деления продолжается.

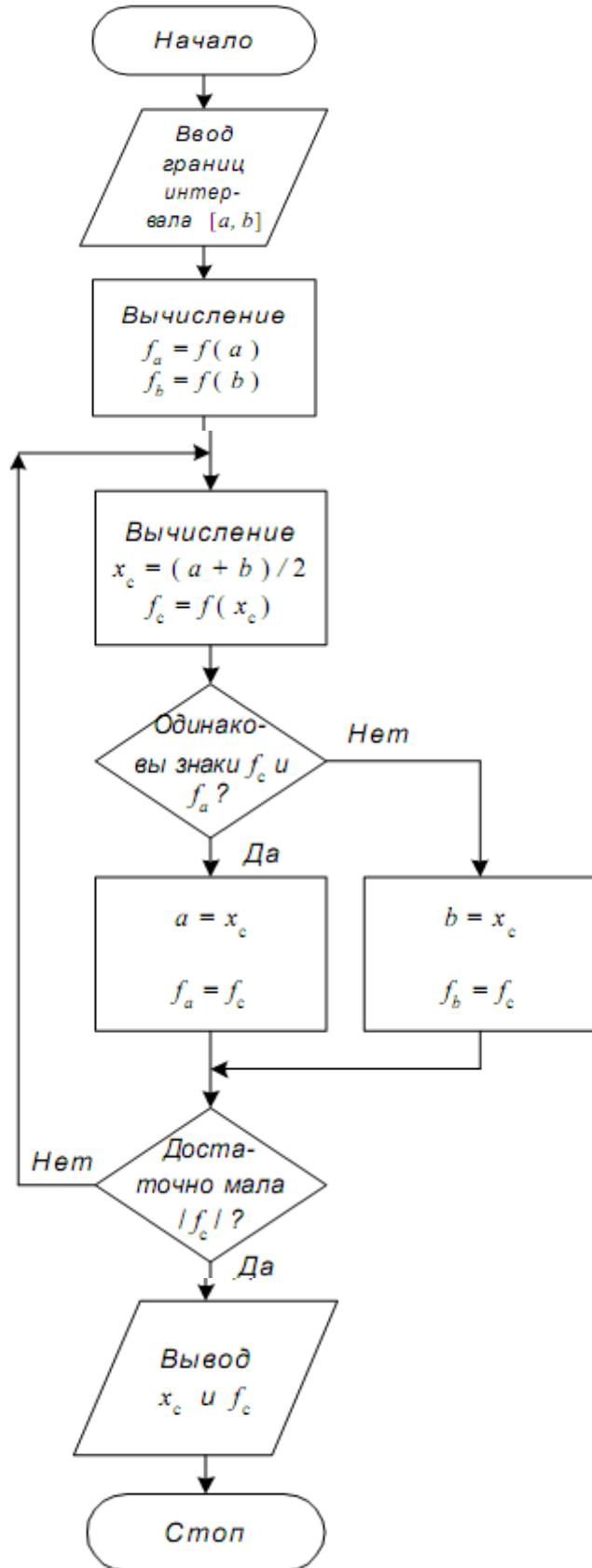


Рис.1.2. Алгоритм метода половинного деления

В некоторых случаях для остановки итерационной процедуры используют условие малости полученного на очередном шаге отрезка, записывая его, например, как

$$|(b-a)/x_c| \leq \delta. \quad (1.1.7)$$

Приняв  $\delta = 0,01$ , можно таким образом получить положение корня с точностью порядка одного процента.

Метод половинного деления позволяет заметно ускорить поиск решения по сравнению с пошаговым поиском табличного метода. Для того чтобы оценить, каков выигрыш, вспомним, что для уменьшения длины исходного отрезка, содержащего корень, в миллион раз в предыдущем случае потребовалось выполнить три итерационных шага и провести вычисление  $f(x)$  в 297 новых точках. В то же время нетрудно подсчитать, что в методе половинного деления для получения аналогичного результата необходимо сделать 20 шагов, так как при  $N = 2$  и  $K = 20$  получается сужение в  $N^K = 2^{20} = 1048576$  раз. А расчет функции  $f(x)$  для этого потребуется провести лишь в  $N \times 20 = 1 \times 20 = 20$  новых точках. В итоге объем и время вычислений по сравнению с ранее рассмотренной процедурой сокращается примерно в пятнадцать раз.

## 1.5. Метод простой итерации

Метод простой итерации основывается на приведении исходного уравнения  $f(x)=0$  к следующему виду:  $x = \psi(x)$ . При этом процесс последовательного приближения к корню строится на основе итерационной формулы

$$x_{i+1} = \psi(x_i).$$

Очевидно, получить расчетную формулу можно, используя следующую цепочку преобразований:

$$f(x) = 0 \Rightarrow bf(x) = 0 \Rightarrow \underbrace{bf(x) + x}_{\psi(x)} = x \Rightarrow x = \psi(x),$$

где  $b$  – некоторый не равный нулю сомножитель.

На рис. 1.3 приведены графические иллюстрации, показывающие приближение к корню в методе простой итерации.

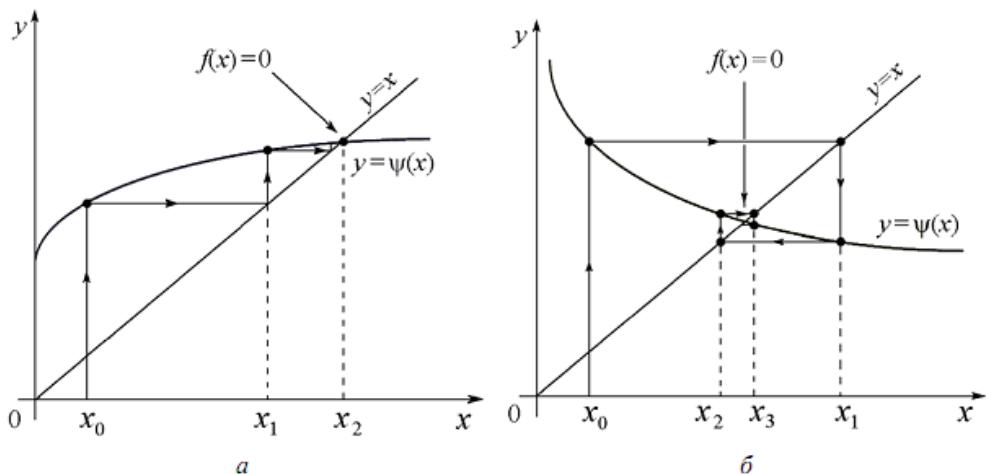


Рис. 1.3. Приближение к корню методом простой итерации

Сходимость процесса приближения к корню в значительной степени определяется видом зависимости  $\psi(x)$ . На рис. 1.3 показаны сходящиеся процессы, а на рис. 1.4 – расходящийся. В последнем случае метод простой итерации не находит решения уравнения. Существенное влияние на сходимость оказывает выбор коэффициента  $b$  – сравните, например, рис. 1.3, а и рис. 1.4.

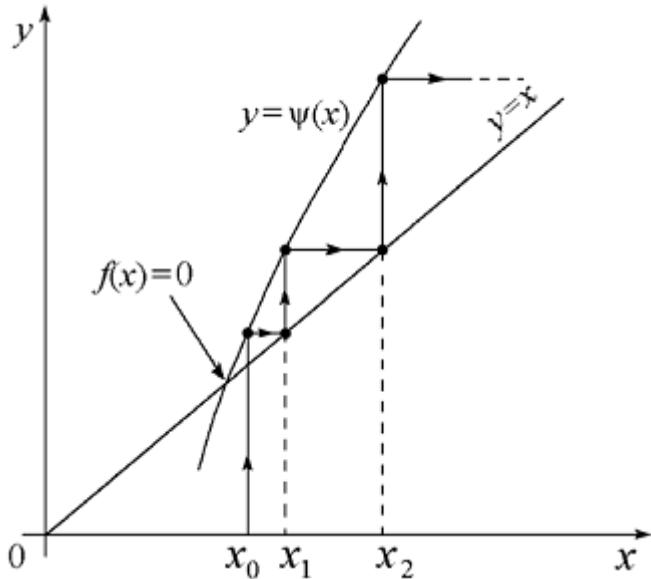


Рис. 1.4. Расходящийся процесс в методе простой итерации

На рис. 1.3 сходимость обеспечивается для медленно изменяющихся функций  $\psi(x)$ , для которых выполняется условие  $|\psi'(x)| < 1$ . На рис. 1.4 расходящийся процесс наблюдается для более быстро меняющейся функции  $|\psi'(x)| > 1$ . Можно сделать вывод, что для обеспечения сходимости метода простой итерации необходимо выполнить условие  $|\psi'(x)| < 1$ .

Алгоритм метода простой итерации приведен на рис. 1.5.

Теоретически можно показать что высокая скорость сходимости обеспечивается при  $b = -1/f'(x)$ . В этом случае метод простой итерации эквивалентен методу Ньютона. Вообще говоря, если в методе Ньютона производная  $f'(x)$  каждый раз вычисляется на очередном шаге, то в методе простой итерации для определения  $b$  можно вычислить производную в начальной точке  $x_0$  и потом сохранять параметр  $b = -1/f'(x)$  неизменным.

## 1.6. Метод Ньютона (метод касательных или метод линеаризации)

Этот метод в отличие от ранее рассмотренных не требует предварительно указывать интервал, в котором располагается корень уравнения. Для начала работы требуется задать лишь одну начальную точку  $x_0$ , расположенную вблизи от предполагаемого корня. Направление поиска определяется из этой точки с помощью линейной экстраполяции  $f(x)$ . Таким образом, при начале расчета из заданной точки  $x_0$  определяется точка  $x_1$ , затем из точки  $x_1$  рассчитывается  $x_2$  и так далее. Продолжение этого процесса да-

лее дает последовательность чисел  $x_0, x_1, x_2, x_3, \dots, x_i, \dots$  последовательно приближающихся к корню уравнения.

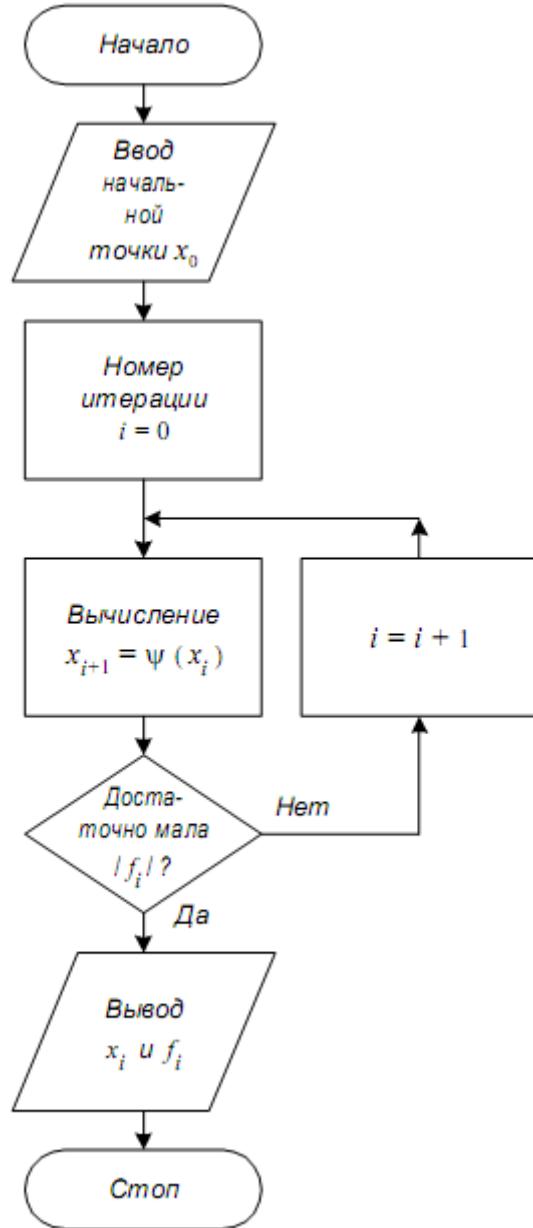


Рис. 1.5. Алгоритм метода простой итерации

Для получения итерационной формулы метода Ньютона воспользуемся разложением функции  $f(x)$  в окрестности точки  $x_i$  в ряд Тейлора:

$$f(x_i + \Delta x) = f(x_i) + \Delta x f'(x_i) + \frac{\Delta x^2}{2!} f''(x_i) + \frac{\Delta x^3}{3!} f'''(x_i) + \dots , \quad (1.1.8)$$

где  $f'(x_i), f''(x_i), f'''(x_i)$  - первая, вторая и третья производные от функции  $f(x)$  по  $x$ .

Сократим (1.1.8), отбросив слагаемые, содержащие  $\Delta x$  во второй и более высоких степенях. Тогда

$$f(x_i + \Delta x) \approx f(x_i) + f'(x_i) \cdot \Delta x .$$

Полагая далее, что в окрестностях  $x_i$  имеется точка  $x_{i+1} = x_i + \Delta x$ , в которой функция  $f(x_{i+1}) = f(x_i + \Delta x)$  равна нулю, получим линейное уравнение

$$f(x_i) + f'(x_i) \cdot \Delta x = 0 ,$$

из которого найдем  $x_{i+1}$ :

$$x_{i+1} = x_i - f(x_i)/f'(x_i) . \quad (1.1.9)$$

Это соотношение является итерационной формулой метода Ньютона. Алгоритм метода Ньютона представлен на рис. 1.6.

Получаемые методом Ньютона точки  $x_i$  образуют ряд чисел  $x_0, x_1, x_2, x_3, \dots$ , который сходится к точному решению, то есть к корню уравнения. Из (1.1.9) следует, что каждый шаг метода Ньютона требует большего объема вычислений чем, например, метод половинного деления, так как приходится находить значение не только функции  $f(x)$ , но и ее производной. Несмотря на это метод Ньютона и его модификации широко используются на практике. Это обусловлено, во-первых, тем, что он не требует задания отрезка  $[a, b]$ , содержащего корень, а может стартовать от одной начальной точки. Во-вторых, он имеет более высокую скорость сходимости, чем ранее рассмотренные методы.

Теоретически можно показать, что метод Ньютона позволяет получить квадратичную сходимость. Это означает, что на каждой итерации погрешность (отклонение очередного приближения  $x_i$  от точного решения) уменьшается по квадратичному закону, то есть количество верных значащих цифр решения удваивается.

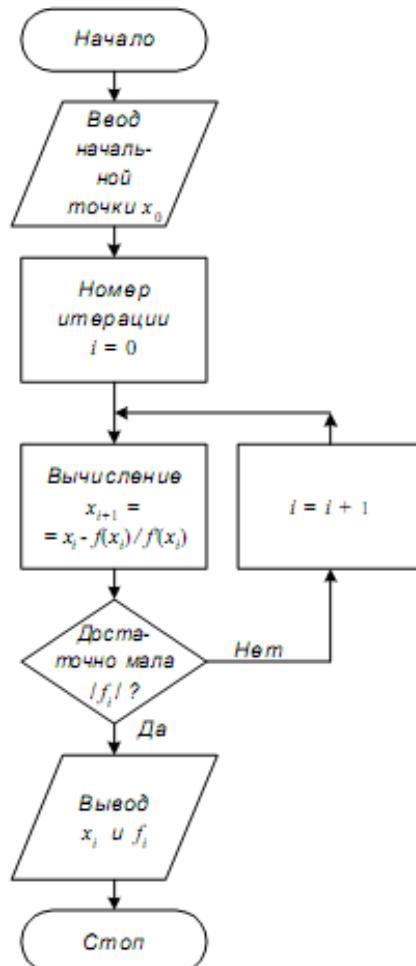


Рис.1.6. Алгоритм метода Ньютона

Если на очередном шаге достигнута погрешность не более 0,5 то за пять-шесть итераций она уменьшится до величины порядка 2–64, что сопоставимо с погрешностью вычислений на ЭВМ. В методе половинного деления для достижения такой же погрешности количество итераций потребовалось бы увеличить более чем на порядок.

На рис. 1.7 представлен ход решения методом Ньютона в графическом виде.

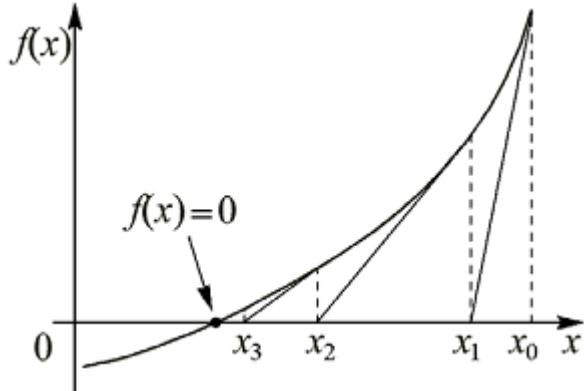


Рис. 1.7. Метод Ньютона

При использовании метода Ньютона следует учитывать ряд его особенностей. Одна из них состоит в необходимости правильного выбора начального приближения.

Метод Ньютона обладает локальной сходимостью, то есть способен найти корень, если начальное приближение задано в некоторой малой его окрестности. Если же начальное приближение взято неудачно и функция не монотонна, метод может дать расходящуюся последовательность  $x_i$ .

Другая проблема заключается в том, что производная  $f'(x)$  в (1.1.9) находится в знаменателе. Это означает, что  $f'(x)$  не должна обращаться в ноль, так как в противном случае итерационная формула перестает работать. Трудности могут возникнуть и в том случае, если  $f'(x)$  не равна нулю, но достаточно мала, вследствие чего результат деления  $f(x)/f'(x)$  может оказаться неприемлемо большим.

Во многих математических пакетах, например, в MathCAD и MATLAB эти проблемы решаются применением комбинированных алгоритмов, сочетающих достоинства различных методов, например, метода половинного деления и метода Ньютона. Первый обеспечивает устойчивую сходимость и используется на начальном этапе решения, а после некоторого числа итераций включается второй, быстрее приближающийся к корню уравнения.

## 2. Решение систем линейных алгебраических уравнений с помощью численных методов

### 2.1. Теоретические сведения о системах линейных алгебраических уравнений

В линейной алгебре рассматриваются четыре класса основных задач: решение систем линейных алгебраических уравнений (СЛАУ), вычисление определителей, нахождение обратных матриц, определение собственных зна-

чений и собственных векторов матриц. Все эти задачи имеют важное прикладное значение при решении различных проблем науки и техники.

Кроме того, задачи линейной алгебры являются вспомогательными при реализации многих алгоритмов вычислительной математики, математической физики, обработки результатов экспериментальных исследований.

Рассмотрим систему линейных алгебраических уравнений (СЛАУ)

$$A \cdot \bar{x} = \bar{f}, \quad (1.2.1)$$

где  $A$  – матрица  $m \times m$ ,  $\bar{x} = (x_1, x_2, \dots, x_m)^T$  – искомый вектор,  $\bar{f} = (f_1, f_2, \dots, f_m)^T$  – заданный вектор. Будем предполагать, что определитель матрицы  $A$  отличен от нуля, т.е. решение системы (1.2.1) существует.

Методы численного решения системы (1.2.1) делятся на две группы: прямые методы («точные») и итерационные методы.

Прямыми методами называются методы, позволяющие получить решение системы (1.2.1) за конечное число арифметических операций. К этим методам относятся метод Крамера, метод Гаусса, LU-метод и т.д.

Итерационные методы (методы последовательных приближений) состоят в том, что решение системы (1.2.1) находится как предел последовательных приближений  $\bar{x}^{(n)}$  при  $n \rightarrow \infty$ , где  $n$  номер итерации. При использовании методов итерации обычно задается некоторое малое число  $\varepsilon > 0$  и вычисления проводятся до тех пор, пока не будет выполнена оценка  $\|\bar{x}^{(n)} - \bar{x}\| < \varepsilon$ . К этим методам относятся метод Зейделя, Якоби, метод верхних релаксаций и т.д.

Следует заметить, что реализация прямых методов на компьютере приводит к решению с погрешностью, т.к. все арифметические операции над переменными с плавающей точкой выполняются с округлением. В зависимости от свойств матрицы исходной системы эти погрешности могут достигать значительных величин.

## 2.2.Метод Гаусса

Запишем систему  $Ax=f$ , в развернутом виде

$$\begin{aligned} a_{11}x_1 + a_{12}x_2 + \dots + a_{1m}x_m &= f_1, \\ a_{21}x_1 + a_{22}x_2 + \dots + a_{2m}x_m &= f_2, \\ \dots & \\ a_{m1}x_1 + a_{m2}x_2 + \dots + a_{mm}x_m &= f_m. \end{aligned}$$

Метод Гаусса состоит в последовательном исключении неизвестных из этой системы. Предположим, что  $a_{11} \neq 0$ . Последовательно умножая первое уравнение на  $-\frac{a_{i1}}{a_{11}}$  и складывая с  $i$ -м уравнение, исключим  $x_1$  из всех уравнений кроме первого. Получим систему

$$a_{11}x_1 + a_{12}x_2 + \dots + a_{1m}x_m = f_1,$$

$$a_{22}^{(I)}x_2 + \dots + a_{2m}^{(I)}x_m = f_2^{(I)},$$

.....

$$a_{m2}^{(I)}x_2 + \dots + a_{mm}^{(I)}x_m = f_m^{(I)},$$

$$\text{где } a_{ij}^{(I)} = a_{ij} - \frac{a_{i1}a_{1j}}{a_{11}}, \quad f_i^{(I)} = f_i - \frac{a_{i1}f_1}{a_{11}}, \quad i, j = 2, 3 \dots m$$

Аналогичным образом из полученной системы исключим  $x_2$ . Последовательно, исключая все неизвестные, получим систему треугольного вида

$$a_{11}x_1 + a_{12}x_2 + \dots + a_{1k}x_k + \dots + a_{1m}x_m = f_1,$$

$$a_{22}^{(I)}x_2 + \dots + a_{2k}^{(I)}x_k + \dots + a_{2m}^{(I)}x_m = f_2^{(I)},$$

.....

$$a_{m-1,m-1}^{(m-1)}x_{m-1} + a_{m-1,m}^{(m-1)}x_m = f_{m-1}^{(m-1)},$$

$$a_{m,m}^{(m-1)}x_m = f_m^{(m-1)}.$$

Описанная процедура называется прямым ходом метода Гаусса. Заметим, что ее выполнение было возможно при условии, что все  $a_{i,i}^{(l)}$ ,  $l = 1, 2 \dots m-1$  не равны нулю.

Выполняя последовательные подстановки в последней системе, (начиная с последнего уравнения) можно получить все значения неизвестных.

$$x_m = \frac{f_m^{(m-1)}}{a_{m,m}^{(m-1)}}, \quad x_i = \frac{1}{a_{i,i}^{(i-1)}}(f_i^{(i-1)} - \sum_{j=i-1}^m a_{ij}^{(i-1)}x_j).$$

Эта процедура получила название обратный ход метода Гаусса..

Метод Гаусса может быть легко реализован на компьютере. При выполнении вычислений, как правило, не интересуют промежуточные значения матрицы  $A$ . Поэтому численная реализация метода сводится к преобразованию элементов массива размерности  $(m \times (m+1))$ , где  $m+1$  столбец содержит элементы правой части системы.

Для контроля ошибки реализации метода используются так называемые контрольные суммы. Схема контроля основывается на следующем очевидном положении. Увеличение значения всех неизвестных на единицу равносильно замене данной системы контрольной системой, в которой свободные члены равны суммам всех коэффициентов соответствующей строки. Создадим дополнительный столбец, хранящий сумму элементов матрицы по строкам. На каждом шаге реализации прямого хода метода Гаусса будем выполнять преобразования и над элементами этого столбца, и сравнивать их значение с суммой по строке преобразованной матрицы. В случае не совпадения значений счет прерывается.

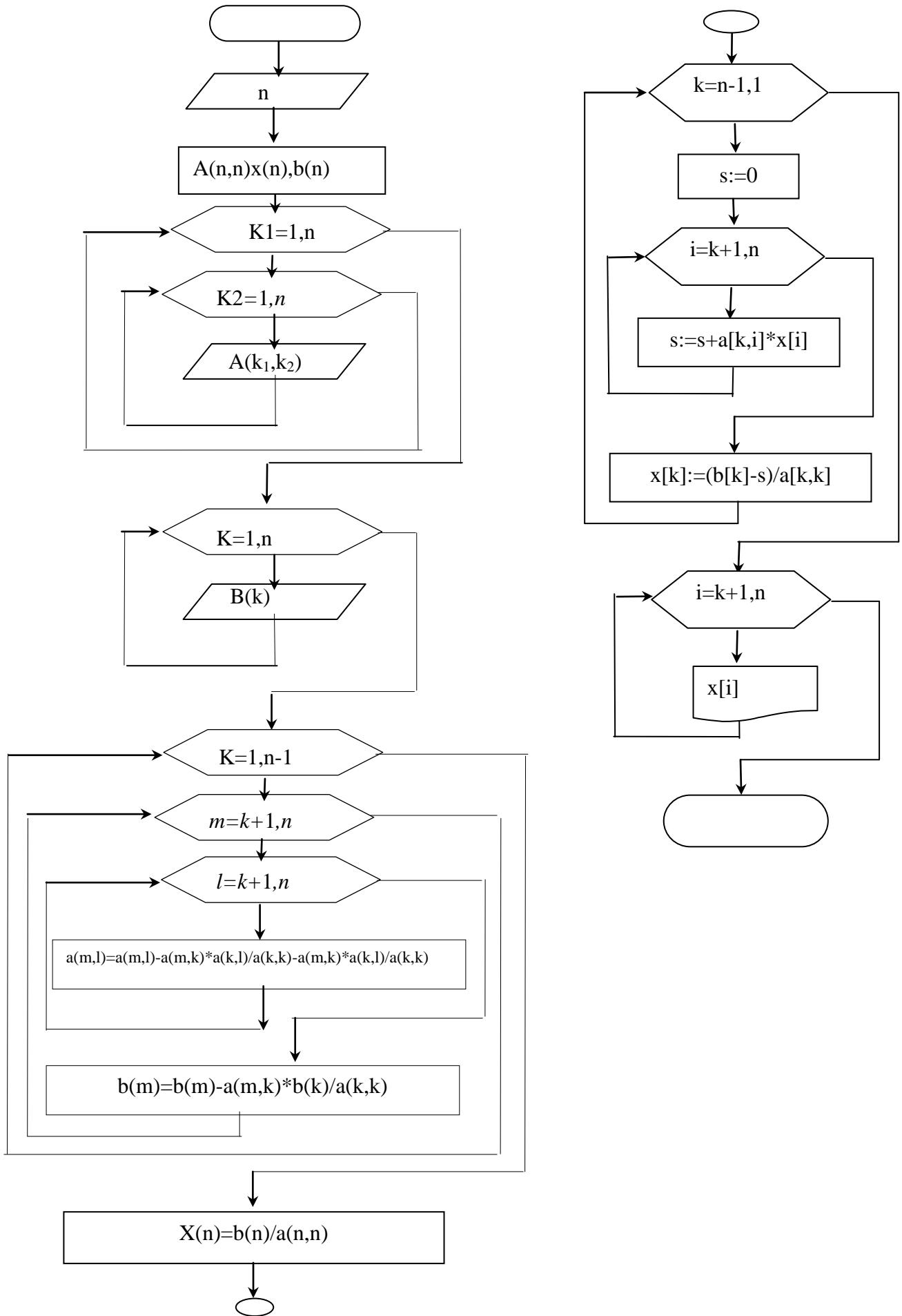


Рис2.1. Алгоритм метода Гаусса

Один из основных недостатков метода Гаусса связан с тем, что при его реализации накапливается вычислительная погрешность. В книге [ Самарский, Гулин] показано, что для больших систем порядка  $m$  число действий умножений и делений близко к  $m^3/3$ .

Для того, чтобы уменьшить рост вычислительной погрешности применяются различные модификации метода Гаусса. Например, метод Гаусса с выбором главного элемента по столбцам, в этом случае на каждом этапе прямого хода строки матрицы переставляются таким образом, чтобы диагональный угловой элемент был максимальным. При исключении соответствующего неизвестного из других строк деление будет производиться на наибольший из возможных коэффициентов и следовательно относительная погрешность будет наименьшей.

Существует метод Гаусса с выбором главного элемента по всей матрице. В этом случае переставляются не только строки, но и столбцы. Использование модификаций метода Гаусса приводит к усложнению алгоритма увеличению числа операций и соответственно к росту времени счета. Поэтому целесообразность выбора того или иного метода определяется непосредственно программистом.

## **II. Решение дифференциальных уравнений**

### **1. Характеристика дифференциальных уравнений**

Инженерные и научные задачи часто связаны с решением дифференциальных уравнений, так как с помощью последних описываются многие физические явления. Соответственно процессы в технических устройствах так же описываются дифференциальными уравнениями.

Природа этих процессов различна. При анализе тепловых режимов аппаратуры рассчитывают тепловые потоки, при изучении электромагнитных процессов – электрические и магнитные поля, при оценке прочности изделий вычисляют механические напряжения и деформации.

К сожалению, для многих практически важных случаев задачи, описываемые дифференциальными уравнениями, весьма сложны, и получить их точное решение оказывается затруднительно или невозможно. Эти трудности могут быть связаны с видом уравнения, например, с его нелинейным характером. Однако решить подобные сложные задачи также как и более простые можно с помощью компьютера. Поэтому методы решения дифференциальных уравнений на ЭВМ широко применяются в инженерной практике.

Дифференциальные уравнения принято делить на две группы: обыкновенные дифференциальные уравнения и уравнения в частных производных.

Далее рассматриваются методы решения задач, описываемых обыкновенными дифференциальными уравнениями. Эти уравнения содержат только одну независимую переменную, в качестве которой может выступать

время или пространственная координата. Иначе говоря, в таких уравнениях все функции зависят только от одной переменной и их производные по этой переменной являются полными.

Уравнения в частных производных содержат более одной независимой переменной. Этими переменными могут быть, например, одновременно пространственные координаты и время или только пространственные координаты для статической задачи. В таких уравнениях производные от функций по любой из независимых переменных являются частными. Кроме того, уравнение может содержать смешанные производные. Подобного рода задачи и методы их решения будут рассмотрены позже.

## 2. Решение обыкновенных дифференциальных уравнений с помощью численных методов

### 2.1. Задача Коши

Важным элементом задач, содержащих дифференциальные уравнения, являются дополнительные условия, которые необходимы для получения количественного решения.

Применительно к обыкновенным дифференциальным уравнениям различают два вида задач: задачу с начальными условиями (задачу Коши) и задачу с краевыми условиями (краевую задачу).

Задачу Коши можно сформулировать следующим образом. Дано обыкновенное дифференциальное уравнение

$$\frac{du(x)}{dx} = f(x, u) \quad (2.2.1)$$

и начальное условие

$$u(x_0) = u_0. \quad (2.2.2)$$

Требуется найти функцию  $u(x)$ , удовлетворяющую уравнению (2.2.1) и начальному условию (2.2.2).

На практике подобные задачи обычно связаны с расчётом переходных электрических, нестационарных тепловых или механических процессов при заданном в некоторый начальный момент времени исходном состоянии системы.

### 2.2. Метод Эйлера

Этот метод является простейшим численным методом решения задачи Коши. Рассмотрим его на примере решения обыкновенного дифференциального уравнения первого порядка (2.2.1) с соответствующим начальным условием (2.2.2).

Расчетную формулу метода Эйлера можно получить, используя разложение функции  $u(x)$  в ряд Тейлора в окрестности некоторой точки  $x_i$ :

$$u(x_i + h) = u(x_i) + h \frac{du(x_i)}{dx} + \frac{h^2}{2!} \frac{d^2u(x_i)}{dx^2} + \frac{h^3}{3!} \frac{d^3u(x_i)}{dx^3} + \dots. \quad (2.2.3)$$

Если приращение  $h$  мало (то есть  $h \ll x_i$ ), то члены ряда, начиная со слагаемого, включающего  $h$  во второй степени, могут быть отброшены как малые величины. Тогда из (2.2.3) в первом приближении получим

$$u(x_i + h) = u(x_i) + h \frac{du(x_i)}{dx} . \quad (2.2.4)$$

Воспользуемся формулой (4), применив ее к единственной известной из условия задачи точке  $x_0$ . Найдем в  $x_0$  производную  $du(x_0)/dx$ , подставив (2.2.2) в (2.2.1):

$$\frac{du(x_0)}{dx} = f[x_0, u(x_0)] = f(x_0, u_0).$$

Подставив последнее выражение в (2.2.4) и полагая  $x_i = x_0$ , получим

$$u(x_0 + h) = u(x_0) + hf(x_0, u(x_0))$$

или, сокращая обозначения, в окончательном виде

$$u_1 = u_0 + hf(x_0, u_0).$$

Таким образом, (4) при известном значении функции  $u_0 = u(x_0)$  в начальной точке  $x_0$  позволяет найти приближенное значение  $u_1 = u(x_1)$  при малом смещении  $h$  от  $x_0$ . На рис. 2.1 графически показан начальный шаг решения методом Эйлера.

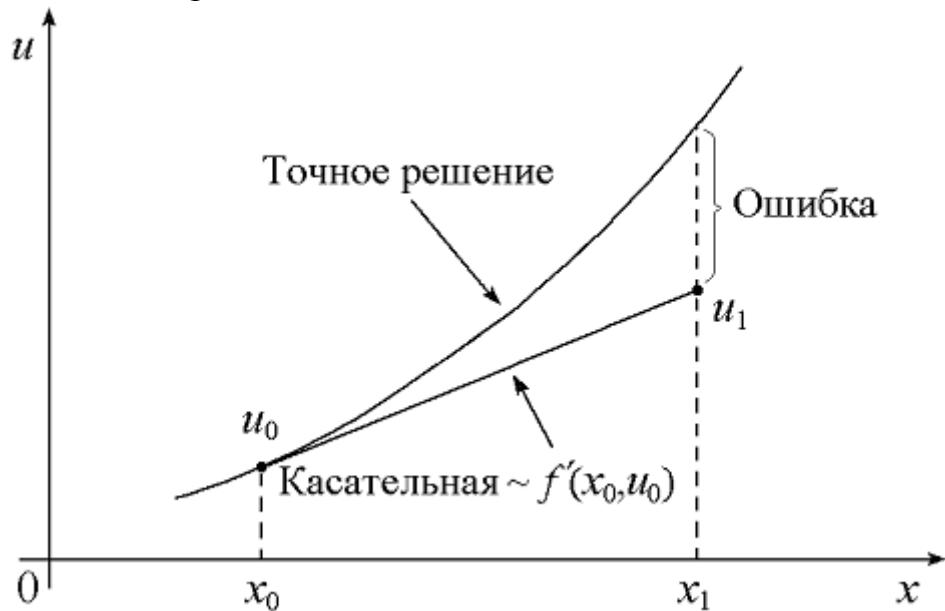


Рис. 2.1. Метод Эйлера

Решение можно продолжить, используя найденное значение функции  $u_1$  для вычисления следующего значения –  $u_2$ . Распространяя эти рассуждения на последующие точки, запишем расчетную формулу метода Эйлера в виде

$$u_{i+1} = u_i + hf(x_i, u_i), \quad i = 0, 1, 2, \dots \quad (2.2.5)$$

Из рис. 2.1 видно, что ошибка метода Эйлера на шаге связана с используемой линейной аппроксимацией  $u(x)$ . Хотя тангенс угла наклона касательной к кривой точного решения в точке  $(x_0, u_0)$  известен и равен  $du(x_0)/dx$ , он изменяется при смещении от  $x_0$  до  $x_1$ . Следовательно, при сохранении начального наклона касательной на всем интервале  $h$  расчет  $u_1$  выполняется с погрешностью. Ошибка метода Эйлера на каждом шаге имеет порядок  $h^2$ , так как члены, содержащие  $h$  во второй и более высоких степенях, отбрасываются – см. (2.2.3) и (2.2.4). Уменьшая  $h$  можно снизить локальную ошибку на шаге.

### 2.3. Модифицированный метод Эйлера

Точность метода Эйлера можно существенно повысить, улучшив аппроксимацию  $u(x)$  на рассчитываемом шаге. Для этого при разложении  $u(x)$  в ряд Тейлора учтем дополнительное слагаемое, содержащее  $h^2$  и  $d^2u(x_i)/dx^2$  в (2.2.3). Определим вторую производную, аппроксимировав ее конечной разностью:

$$\frac{d^2u(x_i)}{dx^2} = \frac{d}{dx} \left( \frac{du(x_i)}{dx} \right) \approx \frac{\Delta u'}{\Delta x} = \frac{u'(x_i + h) - u'(x_i)}{h} \quad (2.2.6)$$

где  $\Delta x = h$ ,  $u'(x_i + h) = du(x_i + h)/dx$  и  $u'(x_i) = du(x_i)/dx$ .

Подставляя полученное выражение в (2.2.3) и отбрасывая члены ряда, начиная со слагаемого, содержащего  $h^3$ , запишем

$$u(x_i + h) = u(x_i) + \frac{h}{2} \left[ \frac{du(x_i)}{dx} + \frac{du(x_i + h)}{dx} \right]$$

Заменяя в последнем выражении производные с помощью (2.2.1) так же, как это было сделано в предыдущем параграфе, и, используя сокращенные обозначения, получим расчетную формулу модифицированного метода Эйлера

$$u_{i+1} = u_i + \frac{h}{2} [f(x_i, u_i) + f(x_{i+1}, u_{i+1})]. \quad (2.2.7)$$

Соотношение (2.2.7) дает решение для  $u_{i+1}$  в неявном виде, поскольку  $u_{i+1}$  присутствует одновременно в левой и правой его частях. Следует отметить, что использование неявных методов оправдано тем, что они, как правило, более устойчивы, чем явные.

Формула (2.2.7) может рассматриваться и как явное решение, если в ее правую часть подставить значение  $u_{i+1}^*$ , рассчитав его предварительно методом Эйлера по формуле (2.2.5). При этом значение  $u_{i+1}^*$  является прогнозом, а уточнение результата по формуле (2.2.7) – его коррекцией. Непосредственная подстановка формулы Эйлера (2.2.5) в правую часть (2.2.7) дает расчетное соотношение метода Эйлера-Коши (или метода Хьюна).

Графически модифицированный метод Эйлера представлен на рис.2.2.

Из рис. 2.2 видно, что поправка, учитывающая изменение наклона кривой  $u(x)$ , заметно уменьшает ошибку на шаге  $h$ .

Модифицированный метод Эйлера обеспечивает второй порядок точности. Ошибка на каждом шаге при использовании этого метода пропорциональна  $h^3$ . Повышение точности достигается за счет дополнительных затрат машинного времени при расчете каждого шага.

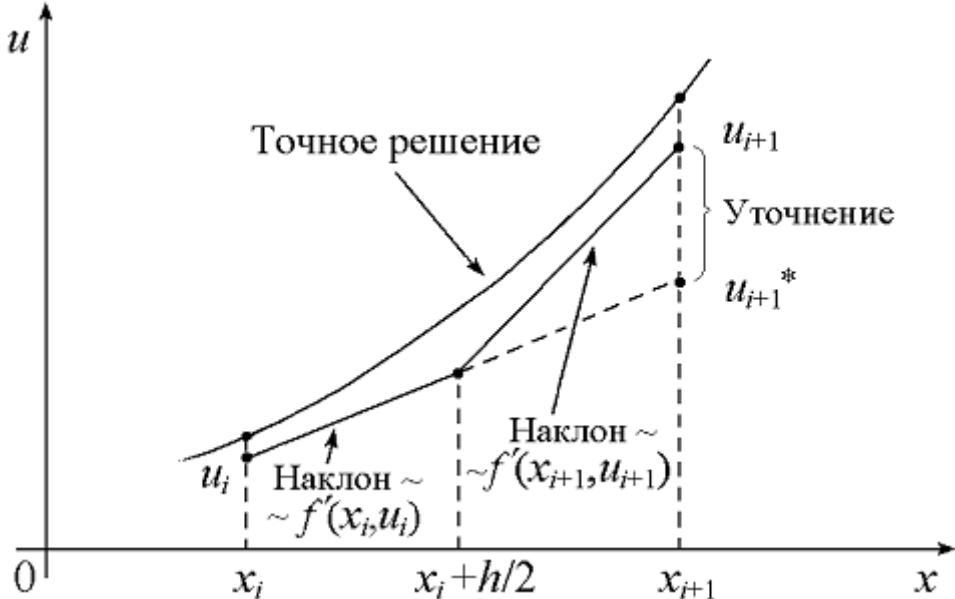


Рис. 2.2. Модифицированный метод Эйлера

Дальнейшее снижение погрешности решения можно получить за счет использования лучшей аппроксимации  $u(x)$ , учитывающей слагаемые высоких порядков. Эта идея положена в основу методов Рунге-Кутта.

#### 2.4. Метод Рунге-Кутта четвертого порядка

В модифицированном методе Эйлера для получения второй производной  $d^2u(x_i)/dx^2$  используется конечно-разностная формула (2.2.6), включающая значения первой производной  $u'(x_i)$  и  $u'(x_i+h)$  в начальной и конечной точках шага. Если подобным же образом вычислить третью производную, рассчитав предварительно вторую производную в двух точках шага, то можно с помощью (2.2.3) построить расчетную формулу метода третьего порядка точности.

Для этого потребуется определить первую производную  $u'(x)$  в дополнительной промежуточной точке между  $x_i$  и  $x_i + h$ .

Аналогичные рассуждения позволяют вывести расчетные формулы методов более высоких порядков, обеспечивающих заметное снижение погрешности решения. Однако на практике их реализация требует существенного повышения объема вычислений с использованием дополнительных промежуточных точек на каждом шаге.

Существуют и другие способы построения численных методов с высоким порядком точности. Один из них, применяемый при построении группы

методов Рунге-Кутта, заключается в аппроксимации решения дифференциального уравнения суммой

$$u(x_i + h) \approx \xi(x_i, h) = u(x_i) + \sum_{n=1}^p A_n k_n(h) \quad (2.2.8)$$

где  $A_n$  – коэффициенты разложения,  $k_n$  – последовательность функций

$$\begin{aligned} k_1 &= hf(x_i, u_i), \\ k_2 &= hf(x_i + \alpha_2 h, u_i + \beta_{21} k_1), \\ k_3 &= hf(x_i + \alpha_3 h, u_i + \beta_{31} k_1 + \beta_{32} k_2), \\ &\dots, \\ k_p &= hf(x_i + \alpha_p h, u_i + \beta_{p,1} k_1 + \dots + \beta_{p,p-1} k_{p-1}), \end{aligned} \quad (2.2.9)$$

$\alpha_n, \beta_{nm}$ ,  $0 < m < n \leq p$  – некоторые параметры.

Неизвестные параметры  $A_n$ ,  $\alpha_n$  и  $\beta_{nm}$  можно выбрать из условия

$$\psi(0) = \psi'(0) = \psi''(0) = \dots = \psi^{(K)}(0) = 0, \quad (2.2.10)$$

где функция  $\psi(h) = u(x_i + h) - \xi(x_i, h)$  показывает отклонение приближенного решения  $\xi(x_i, h)$  от точного  $u(x_i + h)$ . Увеличение параметра  $p$  в (8) позволяет сделать погрешность, связанную с заменой точного решения приближенным, как угодно малой.

Предположим, что  $p=1$ . Тогда, подставляя (2.2.8) в (2.2.10), из условия  $\psi(0)=\psi'(0)=0$  получим  $A_1=1$  и  $\psi''(0)\neq 0$ , откуда

$$u(x_i + h) \approx u(x_i) + \sum_{n=1}^p A_n k_n(h) = u_i + k_1 = u_i + hf(x_i, u_i),$$

что соответствует формуле Эйлера (2.2.5). Таким же образом можно получить формулы более высоких порядков точности, которые называют методами Рунге-Кутта.

Одним из наиболее известных является вариант метода Рунге-Кутта, соответствующий  $p=4$ . Это метод четвертого порядка точности, для которого ошибка на шаге имеет порядок  $h^5$ . Его расчетные формулы имеют следующий вид:

$$u_{i+1} = u_i + \frac{k_1 + 2k_2 + 2k_3 + k_4}{6},$$

где

$$\begin{aligned} k_1 &= hf(x_i, u_i), & k_2 &= hf\left(x_i + \frac{h}{2}, u_i + \frac{k_1}{2}\right), \\ k_3 &= hf\left(x_i + \frac{h}{2}, u_i + \frac{k_2}{2}\right), & k_4 &= hf(x_i + h, u_i + k_3). \end{aligned}$$

Рассмотренные выше метод Эйлера и его модификация по сути дела являются методами Рунге-Кутта первого и второго порядка соответственно.

Несмотря на увеличение объема вычислений метод четвертого порядка имеет преимущество перед методами первого и второго порядков, так как он обеспечивает малую локальную ошибку. Это позволяет увеличивать шаг интегрирования  $h$  и, следовательно, сокращать время расчета.

## 2.5. Погрешность решения и выбор шага

Как было показано выше, порядок точности метода  $p$  определяет ошибку дискретизации  $\sim h^{p+1}$ . Знание порядка ошибки не обеспечивает ее прямую оценку. Получить такую оценку позволяет правило Рунге (формула двойного пересчета).

Пусть одношаговый метод имеет порядок точности  $p$ . Тогда погрешность, равная разности точного решения  $u_T$  и приближенного  $u_{i+1,h}$ , полученного численно с использованием шага  $h$ , имеет порядок  $p+1$ :

$$u_T - u_{i+1,h} \approx Ch^{p+1},$$

Вычитая последнее выражение из предыдущего, определим изменение

$$u_{i+1,\frac{h}{2}} - u_{i+1,h} \approx Ch^{p+1} - C \left(\frac{h}{2}\right)^{p+1} = C \left(\frac{h}{2}\right)^{p+1} (2^{p+1} - 1).$$

Выражая из последнего соотношения постоянную  $C$  и подставляя в предыдущую формулу, получим оценку погрешности по правилу Рунге

$$u_T - u_{i+1,\frac{h}{2}} \approx \frac{u_{i+1,\frac{h}{2}} - u_{i+1,h}}{2^{p+1} - 1}. \quad (2.2.11)$$

Ошибка дискретизации стремится к нулю при стремлении  $h$  к нулю.

Следовательно, уменьшая шаг  $h$  можно сделать локальную ошибку (на шаге) сколь угодно малой. Однако при уменьшении  $h$  необходимо увеличить количество шагов. Поэтому сокращение  $h$  не приводит к такому же снижению глобальной (накапливаемой от шага к шагу) ошибки.

Малые ошибки, появившиеся в начале вычислений, могут совершенно исказить решение, если только не подобрать подходящий численный метод.

Это явление иногда называют «неустойчивостью». Неустойчивость проявляется в катастрофическом нарастании погрешности решения вплоть до возникновения паразитных осцилляций кривой решения.

На практике уменьшению  $h$  препятствуют и ошибки округления, вызванные неточностью представления чисел в компьютере. При уменьшении шага, начиная с некоторого  $h_0$ , вклад ошибок округления преобладает, что приводит к возрастанию погрешности решения (рис. 2.3). Обычно алгоритмы обеспечивают автоматический выбор шага. Для этого выполняется два пробных расчета – с заданным шагом  $h$  и с уменьшенным вдвое  $h/2$ .

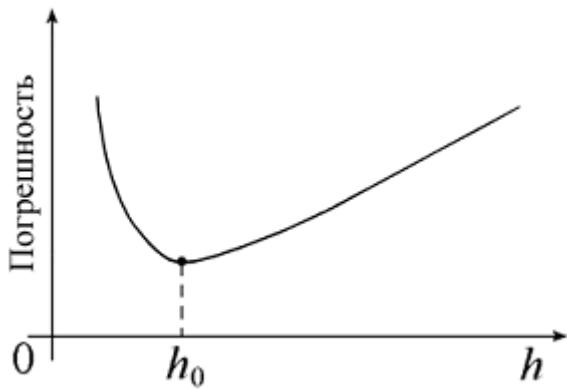


Рис. 2.3. Зависимость погрешности решения от  $h$

В простейшем случае ограничиваются сравнением результатов решений в одной и той же точке:  $|u_{i+1, \frac{h}{2}} - u_{i+1, h}| < \delta$ , где  $\delta$  – некоторое малое положительное число, определяющее требования к точности. Более сложные оценки основываются на формулах подобных правилу Рунге. Если оценка показывает большую ошибку, алгоритм переходит на уменьшенный вдвое шаг.

### 3. Решение дифференциальных уравнений в частных производных

Большое число задач, связанных с анализом физических (и не только физических) полей описываются дифференциальными уравнениями в частных производных. К сожалению, во многих случаях, представляющих практический интерес, найти аналитическое решение таких задач трудно или практически невозможно. Это обычно обусловлено сложной формой или неоднородностью свойств области, в которой отыскивается решение.

Однако результат можно получить численно с помощью компьютера.

Подходы к решению дифференциальных уравнений с частными производными определяются их математической формой. Поэтому рассмотрим классификацию уравнений с этой точки зрения.

#### 3.1. Классификация уравнений по математической форме

Во многих случаях для описания физических процессов используют уравнений с частными производными до второго порядка включительно.

Так, например, изучение свободных колебаний различной природы приводит к волновым уравнениям вида

$$\left( \frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} + \frac{\partial^2 u}{\partial z^2} \right) - \frac{1}{c^2} \frac{\partial^2 u}{\partial t^2} = 0, \quad (2.3.1)$$

где  $u(x, y, z, t)$  – функция, описывающая волновой процесс,  $x, y, z$  – координаты,  $c$  – скорость распространения волны в данной среде,  $t$  – время. Опе-

ратор  $\left(\frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2} + \frac{\partial^2}{\partial z^2}\right)$  принято обозначать значком  $\Delta$ , который в этом случае носит название оператора Лапласа. Процессы распространения тепловой энергии описываются уравнением теплопроводности

$$\rho C \frac{\partial T}{\partial t} - k \left( \frac{\partial^2 T}{\partial x^2} + \frac{\partial^2 T}{\partial y^2} + \frac{\partial^2 T}{\partial z^2} \right) = Q, \quad (2.3.2)$$

где  $\rho$  и  $C$  – плотность и теплоемкость вещества,  $T$  – температура,  $k$  – коэффициент теплопроводности,  $Q$  – плотность источников тепла. Анализ стационарных состояний, например, статических тепловых, электрических, магнитных полей или деформаций при статических нагрузках проводят, используя уравнение Пуассона

$$\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} + \frac{\partial^2 u}{\partial z^2} = -f(x, y, z), \quad (2.3.3)$$

где  $u(x, y, z)$  – функция, описывающая статическое поле,  $f(x, y, z)$  – распределенные источники. Если  $f(x, y, z) = 0$ , то (2.2.3) обращается в уравнение Лапласа:

$$\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} + \frac{\partial^2 u}{\partial z^2} = 0. \quad (2.3.4)$$

Известны и другие виды задач и соответствующие им дифференциальные уравнения в частных производных, например, уравнение диффузии или уравнение Гельмгольца. Несмотря на различие процессов, описываемых рассмотренными уравнениями, и форм их записи, все они с математической точки зрения могут быть представлены как частные случаи обобщенной формы дифференциального уравнения второго порядка.

Рассмотрим уравнение второго порядка с двумя независимыми переменными  $x$  и  $y$ :

$$A \frac{\partial^2 u}{\partial x^2} + 2B \frac{\partial^2 u}{\partial x \partial y} + C \frac{\partial^2 u}{\partial y^2} + D = 0, \quad (2.3.5)$$

где  $A, B, C$  и  $D$  – некоторые функции, зависящие в общем случае от  $x, y, u$ ,  $du/dx$  и  $du/dy$ , причем  $A, B$  и  $C$  одновременно не обращаются в ноль. Дифференциальные уравнения, описывающие физические поля, могут быть нелинейными. Однако на практике многие задачи рассматриваются в линейном приближении, когда уравнение с частными производными линейно относительно неизвестной функции  $u$  и ее частных производных.

На основании того, что уравнению (2.3.5) можно поставить в соответствие квадратичную форму  $A\zeta_1^2 + B\zeta_1\zeta_2 + C\zeta_2^2 = 0$ , по математической природе различают следующие типы квазилинейных уравнений:

- 1) гиперболический, если  $B^2 - 4AC > 0$  – его аналогом является волновое уравнение (2.3.1);
- 2) параболический, если  $B^2 - 4AC = 0$  – его аналог уравнение теплопроводности (2.3.2);
- 3) эллиптический, если  $B^2 - 4AC < 0$  – аналог уравнение Пуассона (2.3.3) или Лапласа (2.3.4).

В задачах, описываемых дифференциальными уравнениями в частных производных, другой важной составляющей помимо самого уравнения является формулировка дополнительных условий.

Для задач с уравнениями гиперболического или параболического типа, содержащих в качестве независимой переменной время  $t$ , условия по  $t$  обычно формулируются как начальные, описывающие исходное состояние системы. По координатам  $x$ ,  $y$  и  $z$  задают граничные условия. В тепловых задачах они, например, описывают распределение температуры на границе расчетной области. В задачах с уравнениями эллиптического типа, не содержащими переменную  $t$ , используют только граничные условия по координатам  $x$ ,  $y$  и  $z$ , а саму задачу называют краевой.

Если краевое условие задает распределение функции  $u$  на границе, то его принято называть условием Дирихле. Условие, определяющее производную  $\bar{n} \cdot \overline{\text{grad}(u)} \equiv \bar{n} \cdot \overline{\nabla u}$  на границе расчетной области, называют условием Неймана. Здесь  $\bar{n}$  – единичная нормаль к границе. Условия, представляющие собой комбинацию двух вышесказанных, называют смешанными.

С помощью дифференциальных уравнений формулируют и другой вид задач – задачи на собственные значения, связанные, например, с определением собственных волн (частот) колебательных систем или волноведущих структур. Однако здесь они не рассматриваются.

Приведенная классификация позволяет определить общие подходы к решению дифференциальных уравнений в задачах различных по физической сути, но сходных с математической точки зрения. В настоящее время широкое распространение получили метод конечных разностей и метод конечных элементов, основы которых и будут рассмотрены ниже.

### 3.2. Основы метода конечных разностей

Метод конечных разностей заключается в том, что дифференциальное уравнение в частных производных заменяется соответствующей ему системой алгебраических уравнений. Решение этой системы дает приближенное решение для искомой функции  $u(x, y, z, t)$ .

Метод включает следующие основные этапы:

- 1) построение сетки, охватывающей рассматриваемую область, например, элемент конструкции какого-нибудь устройства;
- 2) построение на полученной сетке конечно-разностной аппроксимации, эквивалентной исходному дифференциальному уравнению и дополнительным условиям;

3) формирование на основе конечно-разностной аппроксимации системы алгебраических уравнений и ее решение.

Рассмотрим перечисленные этапы на примере двухмерных задач.

Формирование сетки производится с учетом геометрии задачи, например, формы детали, для которой выполняется расчет. Обычно для деталей, имеющих прямоугольную форму, используют декартову систему координат и соответственно прямоугольную сетку. На рис. 2.4 приведен пример такой двухмерной сетки, нанесенной на прямоугольную пластину.

В методе конечных разностей применяют и другие виды сеток. Например, если исследуемая конструкция содержит элементы с осевой симметрией, используют полярную сетку.

В дальнейшем решение задачи строят, опираясь на узлы сетки, то есть на точки пересечения ее линий.

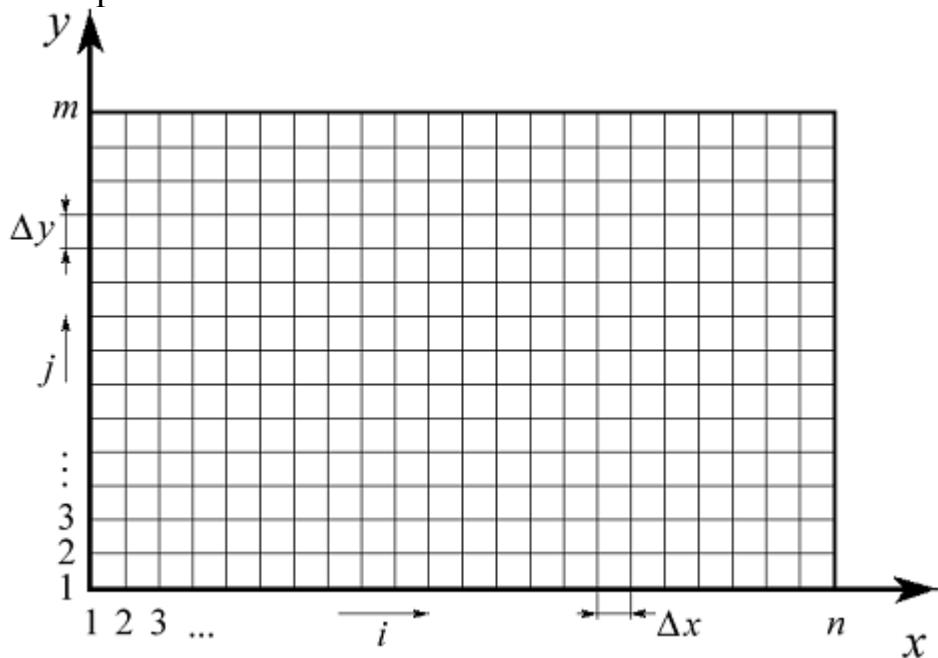


Рис. 2.4. Прямоугольная сетка

Конечно-разностная аппроксимация производных в дифференциальном уравнении строится путем замены этих производных на их приближенные аналоги с помощью сетки. Так, например, частную производную  $\frac{\partial u}{\partial x} = \lim_{\Delta x \rightarrow 0} \frac{[u(x + \Delta x) - u(x)]}{\Delta x}$  в точке  $(x_i, y_i)$  можно заменить приближенным значением так называемой "правой производной"

$$\frac{\partial u}{\partial x} \approx \frac{\Delta u}{\Delta x} \bigg|_{\text{ПРАВАЯ}} = \frac{u_{i+1} - u_i}{x_{i+1} - x_i} = \frac{u_{i+1} - u_i}{\Delta x} \quad (2.3.6)$$

или "левой производной"

$$\frac{\partial u}{\partial x} \approx \frac{\Delta u}{\Delta x} \bigg|_{\text{ЛЕВАЯ}} = \frac{u_i - u_{i-1}}{x_i - x_{i-1}} = \frac{u_i - u_{i-1}}{\Delta x} \quad (2.3.7)$$

где  $\Delta u$  и  $\Delta x$  – приращения функции и аргумента,  $u_i$ ,  $x_i$  и  $u_{i+1}$ ,  $x_{i+1}$  – значения функции и аргумента в узлах  $i$  и  $i+1$ , причем  $\Delta x$  – шаг сетки по координате  $x$ . Аналогично получается формула для второй производной  $\partial^2 u / \partial x^2$ :

$$\frac{\partial^2 u}{\partial x^2} = \frac{\partial}{\partial x} \left( \frac{\partial u}{\partial x} \right) \approx \frac{\frac{\Delta u}{\Delta x} \Big|_{\text{ПРАВАЯ}} - \frac{\Delta u}{\Delta x} \Big|_{\text{ЛЕВАЯ}}}{\Delta x} = \frac{\left( \frac{u_{i+1} - u_i}{\Delta x} \right) - \left( \frac{u_i - u_{i-1}}{\Delta x} \right)}{\Delta x} = \frac{u_{i+1} - 2u_i + u_{i-1}}{\Delta x^2}. \quad (2.3.8)$$

В полученных выражениях в отличие от точных производных используются малые, но не бесконечно малые разности  $\Delta u$  и  $\Delta x$ . Поэтому сам метод и получил название метода конечных разностей. Формулы для производных по независимым переменным  $y$ ,  $z$ ,  $t$  получают аналогично.

### 3.3. Аппроксимация уравнения эллиптического типа

Преобразование уравнения эллиптического типа (2.3.3) для двухмерной задачи (когда  $\partial^2 u / \partial z^2 \equiv 0$ ) производится путем замены в нем производных  $\partial^2 u / \partial x^2$  и  $\partial^2 u / \partial y^2$  конечно-разностными формулами. Заменив в (2.3.3)  $\partial^2 u / \partial x^2$  с помощью (2.3.8) и используя аналогичное выражение для  $\partial^2 u / \partial y^2$ , получим

$$\frac{u_{i+1,j} - 2u_{i,j} + u_{i-1,j}}{\Delta x^2} + \frac{u_{i,j+1} - 2u_{i,j} + u_{i,j-1}}{\Delta y^2} = -\frac{P}{W}, \quad (2.3.9)$$

где индексы  $i$  и  $j$  отсчитываются соответственно по осям  $X$  и  $Y$ .

Для упрощения анализа предположим, что в сетке используются квадратные ячейки, то есть  $\Delta x = \Delta y = h \neq 0$ , тогда

$$u_{i+1,j} + u_{i-1,j} + u_{i,j+1} + u_{i,j-1} - 4u_{i,j} = -\frac{P}{W}h^2. \quad (2.3.10)$$

Уравнение (2.3.10) связывает между собой неизвестное значение функции  $u_{i,j}$  с ее значениями в четырех соседних узлах. На сетке эти узлы образуют пятиточечный шаблон (рис. 2.5), позволяющий легко определить индексы в (2.3.10) для любого произвольно выбранного на сетке узла  $i,j$ .

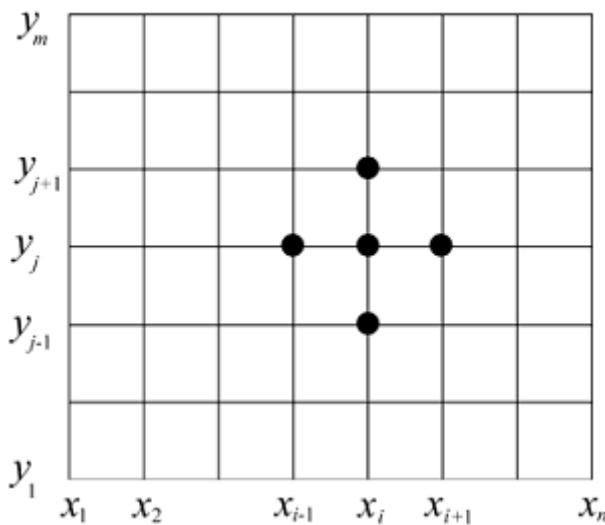


Рис. 2.5. Шаблон "крест" для уравнения эллиптического типа

Записывая (2.3.10) для каждого узла  $2 < i < n-1, 2 < j < m-1$  и подставляя вместо  $i$  и  $j$  соответствующие номера, получим систему связанных уравнений. Количество уравнений будет равно количеству узлов, в которых необходимо найти неизвестные  $u_{i,j}$ . Иначе говоря, число неизвестных равно числу уравнений и система будет замкнутой. Значения функции  $u$  в узлах сетки, лежащих на границе рассматриваемой области, определяются заданными граничными условиями. Например, если в задаче об изгибе пластины ее края считаются жестко закрепленными, то смещение в граничных узлах полагается нулевым:  $u_{1,j}=u_{n,j}=u_{i,1}=u_{i,m}=0$ .

Решение системы алгебраических уравнений, получаемой в результате конечно-разностной аппроксимации уравнения эллиптического типа, является одним из наиболее тяжелых по вычислительным затратам этапов расчета.

Для повышения точности решения приходится использовать сетки с большим числом узлов, на которых формируются и довольно большие системы – нередко до нескольких тысяч алгебраических уравнений. Одним из способов уменьшения числа узлов и является использование сеток с неравномерным шагом. При этом сетку сгущают в наиболее важных с точки зрения точности участках, например, вблизи углов или отверстий.

В то же время решение задачи облегчается тем, что каждое из алгебраических уравнений содержит небольшое количество неизвестных. В качестве примера ниже приведена система с разреженной матрицей ленточного типа, полученной из (2.3.10) для прямоугольной области (рис. 2.4) при  $n=m=5$ .

В правой части записаны  $u_{i,j}$  относящиеся к узлам, лежащим на границах.

$$\begin{bmatrix} -4 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & -4 & 1 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & -4 & 0 & 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & -4 & 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 & -4 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 & -4 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 & 0 & -4 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 1 & -4 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & -4 \end{bmatrix} \begin{bmatrix} u_{22} \\ u_{23} \\ u_{24} \\ u_{32} \\ u_{33} \\ u_{34} \\ u_{42} \\ u_{43} \\ u_{44} \end{bmatrix} = \begin{bmatrix} -Ph^2/W - u_{21} - u_{12} \\ -Ph^2/W - u_{13} \\ -Ph^2/W - u_{14} - u_{25} \\ -Ph^2/W - u_{31} \\ -Ph^2/W \\ -Ph^2/W - u_{35} \\ -Ph^2/W - u_{41} - u_{52} \\ -Ph^2/W - u_{53} \\ -Ph^2/W - u_{45} - u_{54} \end{bmatrix}$$

Для решения подобных систем используют специальные методы, учитывающие разреженность матрицы коэффициентов. К специальным прямым относятся некоторые матричные методы и метод прогонки (аналог метода Гаусса). Из итерационных применяют метод Якоби (одновременных смещений) и метод Гаусса-Зейделя (последовательных смещений), а также модификации последнего, например, метод верхней релаксации.

### 3.4. Аппроксимация уравнения гиперболического типа

Построение алгебраических уравнений на основе дифференциального уравнения гиперболического типа (2.3.1) выполняется, так же как и в предыдущем случае, заменой производных конечно-разностными аналогами.

В качестве примера рассмотрим задачу продольных колебаний тонкого однородного стержня длиной  $L$  (рис. 2.6), когда его деформация  $u$  зависит только от продольной (вдоль оси стержня) координаты  $x$  и времени  $t$ .

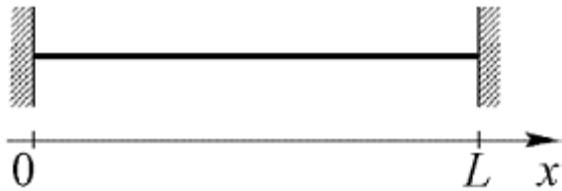


Рис. 2.6. Модель стержня

Колебания стержня описываются дифференциальным уравнением

$$\frac{\partial^2 u}{\partial x^2} - \frac{1}{a^2} \frac{\partial^2 u}{\partial t^2} = 0, \quad (2.3.11)$$

где  $a = \sqrt{E/\rho}$ ,  $E$  и  $\rho$  – модуль упругости и плотность материала стержня. Аппроксимация уравнения производится на сетке в координатах  $t$  и  $x$ . Примерный вид сетки показан на рис. 2.7. Данная задача не имеет верхней границы по координате  $t$ . Это объясняется тем, что с формальной точки зрения колебания в стержне могут продолжаться неопределенно долгое время, даже если будут учтены потери, приводящие к их затуханию.

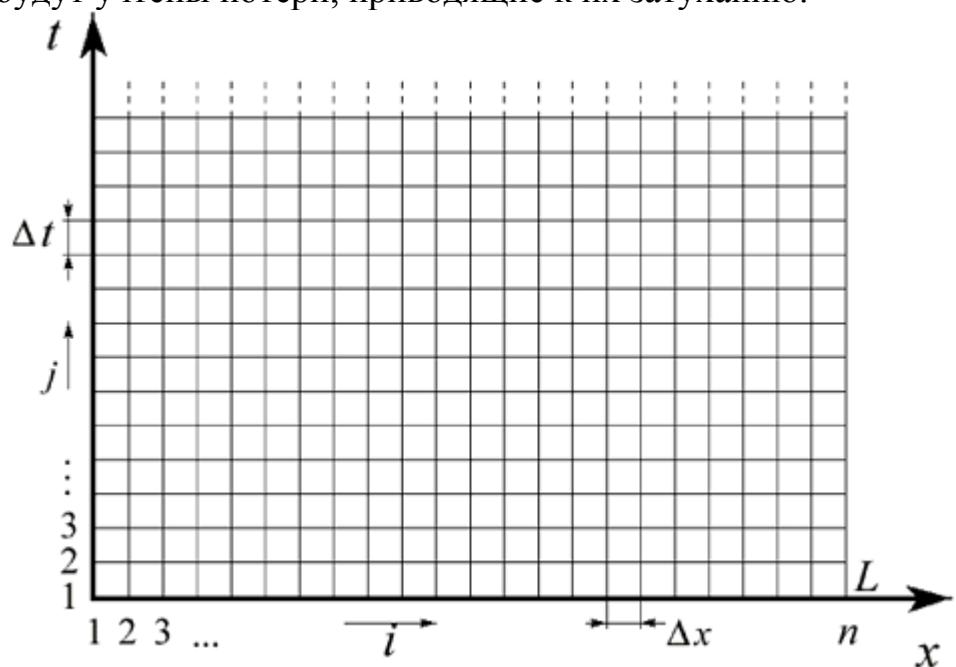


Рис. 2.7. Сетка в координатах  $t$  и  $x$

Используя сетку, запишем в конечно-разностях уравнение, эквивалентное (2.3.11):

$$\frac{u_{i,j+1} - 2u_{i,j} + u_{i,j-1}}{\Delta t^2} = a^2 \frac{u_{i+1,j} - 2u_{i,j} + u_{i-1,j}}{\Delta x^2} \quad (2.3.12)$$

или

$$u_{i,j+1} = 2(1 - \beta^2)u_{i,j} + \beta^2(u_{i+1,j} + u_{i-1,j}) - u_{i,j-1}, \quad (2.3.13)$$

где  $\beta = a \Delta t / \Delta x$ . Из (2.3.12) и (2.3.13) видно, что форма шаблона уравнения гиперболического типа подобна форме шаблона уравнения эллиптического типа. Аналогично предыдущей задаче запишем уравнение (2.3.13) для каждого узла сетки  $i$ , подставляя в него вместо  $i$  и  $j$  соответствующие этим узлам номера, получим систему связанных алгебраических уравнений.

В качестве граничных условий по  $x$  в данной задаче могут использоваться любые условия, описывающие способ закрепления стержня.

Например, жесткое закрепление предполагает нулевой сдвиг на концах стержня. Это соответствует условию  $u(x=0,t)=0$  и  $u(x=L,t)=0$ , где  $x=0$  и  $x=L$  – координаты концов стержня.

По времени  $t$  в качестве начальных условий зададим при  $t=0$  исходную деформацию стержня и начальную скорость его колебаний

$$u(x, t=0) = f_d(x), \quad \left. \frac{\partial u}{\partial t} \right|_{t=0} = v(x) \Big|_{t=0}. \quad (2.3.14)$$

Решение системы уравнений для рассматриваемой задачи можно получить с помощью сравнительно простой процедуры, называемой явной схемой. Эта схема строится на том, что все уравнения системы последовательно связаны между собой.

Расчет будем проводить в следующем порядке. Вначале определим деформацию стержня в моменты  $t=0$  и  $t=0+\Delta t$ . Для  $t=0$  деформация  $u(x,0) \equiv u_{i,1}$  известна из заданных начальных условий (2.3.14). Для следующего момента времени  $t=\Delta t$  деформацию  $u(x,\Delta t) \equiv u_{i,2}$  определим с помощью второго начального условия, задающего скорость  $\partial u / \partial t$  при  $t=0$ :

$$\left. \frac{\partial u}{\partial t} \right|_{t=0} = v(x) \Rightarrow \frac{u_{i,2} - u_{i,1}}{\Delta t} = v_{i,1}, \quad \text{тогда } u_{i,2} = u_{i,1} + v_{i,1} \Delta t. \quad (2.3.15)$$

При известных из (2.3.14) и (2.3.15)  $u_{i,1}$  и  $u_{i,2}$  начнем решение задачи следующим образом. Полагая, что  $j=2$ , то есть  $u_{i,j-1} = u_{i,1}$  и  $u_{i,j} = u_{i,2}$ , подставим в (2.3.13) известную из (2.3.14) соответствующую  $t=0$  начальную деформацию  $u_{i,1} \equiv u(x,t=0) = f_d(x)$ , и соответствующую  $t=\Delta t$  деформацию  $u_{i,2} = u_{i,1} + v_{i,1} \Delta t$  (см. (2.3.14)). Вычисление правой части (2.3.13) позволяет определить  $u_{i,j+1} = u_{i,3}$  в момент времени  $t=2\Delta t$ .

Далее действуя аналогично и сдвигая шаблон решения на одну линию сетки по координате  $t$ , вычисляются последовательно фазы колебаний  $u_{i,4}$  – из  $u_{i,2}$  и  $u_{i,3}$ , затем  $u_{i,5}$  – из  $u_{i,3}$  и  $u_{i,4}$  и так далее. То есть очередной временной слой  $j+1$  рассчитывается из предыдущих – с индексами  $j$  и  $j-1$ .

При решении гиперболического уравнения следует обращать внимание

на выбор шага сетки по  $x$  и  $t$ . Теоретически можно показать, что приближенное решение, получаемое с помощью (2.3.13), сходится к точному при  $\Delta x \rightarrow 0$  и  $\Delta t \rightarrow 0$  со скоростью  $O(\Delta x^2 + \Delta t^2)$ , если  $\beta = a\Delta t / \Delta x < 1$ . Иначе говоря, если выбран шаг сетки  $\Delta x$  по координате  $x$ , то появляется ограничение на шаг по времени  $\Delta t$ .

При  $\beta > 1$  метод становится неустойчивым как в абсолютном, так и в относительном смысле. Последнее означает, что по мере продолжения вычислений ошибки катастрофически нарастают. Теоретически показано, что при  $\beta = 1$  метод устойчив и конечно-разностное решение совпадает с точным. При  $\beta < 1$  решение хотя и устойчиво, но его точность с уменьшением  $\beta$  убывает.

### 3.5. Аппроксимация уравнения параболического типа

Решение двухмерной задачи с уравнением параболического типа (2.3.2) выполняется с помощью сетки аналогичной приведенной на рис. 2.7.

Рассмотрим процесс теплопередачи по длинному однородному стержню длиной  $L$ , ось которого совпадает с осью  $x$ . Предположим, что в исходном состоянии стержень по всей длине имеет температуру  $T = T_0$ . Затем, начиная с момента времени  $t = 0$  температура на его правом конце  $x = L$  скачком возрастает до  $T_L$ , в то время как на левом конце  $x = 0$  поддерживается температура  $T = T_0$ . Теплопередачей через боковую поверхность стержня будем пренебречь.

Учитывая, что в стержне отсутствуют источники тепла ( $Q = 0$ ), запишем в конечных разностях уравнение, эквивалентное (2.3.2):

$$\frac{T_{i,j+1} - T_{i,j}}{\Delta t} = \frac{k}{\rho C} \frac{T_{i+1,j} - 2T_{i,j} + T_{i-1,j}}{\Delta x^2} \quad (2.3.16)$$

$$\text{или } T_{i,j+1} = \beta T_{i+1,j} + (1 - 2\beta) T_{i,j} + \beta T_{i-1,j}, \quad (2.3.17)$$

где  $\beta = (k / \rho C) (\Delta t / \Delta x^2)$ . Из (2.3.16) и (2.3.17) видно, что шаблон для уравнения параболического типа напоминает перевернутую букву Т.

Границные условия по координате  $x$  в данной задаче включают температуру на концах стержня:  $T_{1,j} = 0$  при  $x = 0$  и  $T_{n,j} = T_L$  при  $x = L$ . По времени  $t$  начальное условие задает исходное распределение температуры в стержне  $T(x, t=0) = T_0$ . Запишем уравнение (2.3.17) для каждого узла сетки  $i$ , подставляя в него вместо  $i$  и  $j$  соответствующие этим узлам номера, получим систему связанных уравнений.

Решение системы уравнений для данной задачи, так же как и в предыдущем случае вычисляется с использованием явной схемы. При этом расчет упрощается за счет того, что распределение температуры в стержне для каждого последующего временного слоя  $j+1$  определяется из известного распределения только в одном предыдущем слое  $j$ .

При решении уравнения параболического типа также важен выбор шага  $\Delta t$ . Для обеспечения сходимости и устойчивости метода желательно, что-

бы параметр  $\beta = (k/\rho C)(\Delta t/\Delta x^2)$  в (2.3.17) не превышал 0,5. Нарушение этого условия приводит к расходящемуся или колеблющемуся решению.

### 3.6. Погрешность решения

Погрешность решения методом конечных разностей в первую очередь определяется ошибкой, вносимой при замене исходного дифференциального уравнения на его конечно-разностный аналог. Вначале оценим погрешность аппроксимации (2.3.6) для первой производной, используя разложение  $u(x)$  в окрестностях точки  $x$  в ряд Тейлора:

$$u(x_i + \Delta x) = u(x_i) + \Delta x \frac{\partial u(x_i)}{\partial x} + \frac{\Delta x^2}{2!} \frac{\partial^2 u(x_i)}{\partial x^2} + \frac{\Delta x^3}{3!} \frac{\partial^3 u(x_i)}{\partial x^3} + \frac{\Delta x^4}{4!} \frac{\partial^4 u(x_i)}{\partial x^4} + \dots, \quad (2.3.18)$$

откуда

$$\frac{\partial u(x_i)}{\partial x} = \frac{u(x_i + \Delta x) - u(x_i)}{\Delta x} - \frac{\Delta x}{2!} \frac{\partial^2 u(x_i)}{\partial x^2} - \frac{\Delta x^2}{3!} \frac{\partial^3 u(x_i)}{\partial x^3} - \dots. \quad (2.3.19)$$

Согласно (2.3.18) погрешность конечно-разностной аппроксимации по формуле (2.3.6) обусловлена тем, что в ней не учитываются слагаемые высоких порядков, начиная с  $(\Delta x/2!)(\partial^2 u/\partial x^2)$ . Можно утверждать, что в (2.3.19) слагаемые убывают по мере увеличения их порядка. Поэтому ошибка (2.3.6) приближенно равна  $(\Delta x/2)(\partial^2 u/\partial x^2)$ .

Аналогичную оценку нетрудно провести и для второй производной. Для этого необходимо воспользоваться (2.3.18) и аналогичным разложением, записанным для  $u(x_i - \Delta x)$ :

$$u(x_i - \Delta x) = u(x_i) - \Delta x \frac{\partial u(x_i)}{\partial x} + \frac{\Delta x^2}{2!} \frac{\partial^2 u(x_i)}{\partial x^2} - \frac{\Delta x^3}{3!} \frac{\partial^3 u(x_i)}{\partial x^3} + \frac{\Delta x^4}{4!} \frac{\partial^4 u(x_i)}{\partial x^4} - \dots \quad (2.3.20)$$

Сложив (2.3.18) и (2.3.20) получим выражение для второй производной:

$$\frac{\partial^2 u(x_i)}{\partial x^2} = \frac{u(x_i + \Delta x) - 2u(x_i) + u(x_i - \Delta x)}{\Delta x^2} - \frac{\Delta x^2}{12} \frac{\partial^4 u(x_i)}{\partial x^4} - \dots. \quad (2.3.21)$$

Из сравнения (2.3.21) и (2.3.8) видно, что погрешность (2.3.8) определяется не учтеными в ней слагаемыми высоких порядков, начиная с  $(\Delta x^2/12)(\partial^4 u/\partial x^4)$ .

Поэтому ошибка (8) уменьшается пропорционально квадрату  $\Delta x$ . Данный результат полезно учитывать при выборе шага сетки. Так, например, уменьшение вдвое шага  $\Delta x = \Delta y = h$  приводит к снижению ошибки аппроксимации для уравнения эллиптического типа в четыре раза.

Нельзя утверждать, что уменьшение шага сетки однозначно повышает точность решения методом конечных разностей. С увеличением количества узлов сетки возрастает объем вычислений и, следовательно, растут вычислительные погрешности. На практике для оценки погрешности решения можно провести ряд пробных расчетов с разными значениями шага сетки и выбрать вариант, обеспечивающий приемлемую точность при невысоких вычислительных затратах.

## Источники погрешности решения задачи на ЭВМ

Рассмотренные итерационные методы поиска корней нелинейных уравнений по своей природе являются приближенными в отличие от прямых методов, дающих точное решение. С точки зрения точности результата использование прямых методов может показаться более предпочтительным.

Однако на самом деле при решении задачи на компьютере ответ все равно будет содержать погрешность.

В качестве основных источников погрешности обычно рассматривают три вида ошибок. Это так называемые ошибки усечения, ошибки округления и ошибки распространения. Рассмотрим их.

### Ошибки усечения

Этот вид ошибок связан с погрешностью, заложенной в самой задаче. Он может быть обусловлен неточностью определения исходных данных. Например, если в условии задачи заданы какие-либо размеры, то на практике для реальных объектов эти размеры известны всегда с некоторой точностью.

То же самое касается любых других физических параметров. Сюда же можно отнести неточность расчетных формул и входящих в них числовых коэффициентов.

Большое число расчетных формул являются эмпирическими и дают результат с некоторой погрешностью, содержат подгоночные коэффициенты, обеспечивающие приемлемую ошибку в ограниченном диапазоне входных параметров. Поэтому, как правило, если исходные данные известны с некоторой погрешностью, вряд ли стоит пытаться получить результат с меньшей погрешностью.

### Ошибки распространения

Данный вид ошибок связан с применением того или иного способа решения задачи. В ходе вычислений неизбежно происходит накопление или, иначе говоря, распространение ошибки. Помимо того, что сами исходные данные не являются точными, новая погрешность возникает при их перемножении, сложении и т. п. Накопление ошибки зависит от характера и количества арифметических действий, используемых в расчете.

Обычно для решения одной и той же задачи может быть использован ряд различных методов решения. Например, систему линейных алгебраических уравнений можно решить методом Гаусса или через определители (методом Крамера). Теоретически оба метода позволяют получить точное решение. Однако на практике при решении больших систем уравнений метод Гаусса обеспечивает меньшую погрешность, чем метод Крамера, так как использует меньший объем вычислений.

### Ошибки округления

Это тип ошибок связан с тем, что истинное значение числа всегда точно сохраняется компьютером. При сохранении вещественного чис-

ла в памяти компьютера оно записывается в виде мантиссы и порядка, примерно так же, как отображается число на дисплее калькулятора (см. рис.1).

$\pm$	$R_1$	$R_2$	$R_3$	.	.	.	.	.	$R_n$	$\pm$	$D_1$	$D_2$	.	.	.	$D_m$
Мантисса								Порядок								

Рис.1 Структура записи вещественного числа

Здесь  $R_1, R_2, R_3, \dots, R_n$  – разряды мантиссы,  $D_1, D_2, \dots, D_m$  – разряды порядка. На самом деле конечно, в отличие от дисплея калькулятора, мантисса и порядок числа, включая их знаки, в памяти компьютера хранятся в двоичном виде. Но для обсуждения природы ошибок округления это различие не столь принципиально.

Понятно, что иррациональные числа такие, как  $\pi=3,14159\dots$  и  $e=2,712\dots$  не могут быть представлены в памяти компьютера в принципе.

Однако же и рациональные числа, если количество их значащих цифр превышает число отведенных разрядов мантиссы (см. рис. 1), будут представлены не точно. При этом цифра последнего сохраняемого в ЭВМ разряда может быть записана с округлением или без него.

Фактически при заданной структуре хранения числа компьютер может использовать не бесконечное, а конечное число рациональных чисел, которые вписываются в приведенную на рис. схему. Поэтому любой входной параметр решаемой задачи, ее промежуточный результат и окончательный ответ всегда округляются до разрешенных в компьютере чисел.

Следующий важный вывод касается диапазона представления чисел в ЭВМ. Если проводить рассуждения для десятичной системы счисления, то максимальное по модулю число, которое может быть представлено в соответствии со схемой на рис.1 , равно

$$\pm X_{\infty} = \pm 999 \dots 9 \times 10^{+99 \dots 9}.$$

Все числа, превышающие по модулю  $X_{\infty}$ , не представимы в ЭВМ и рассматриваются как машинная бесконечность. Если в ходе расчетов будет получен результат, превышающий  $X_{\infty}$ , то произойдет аварийное завершение вычислений по переполнению.

Минимальное по модулю число, сохраняемое в памяти компьютера, по схеме на рис.1 равно

$$\pm X_0 = \pm 000 \dots 1 \times 10^{-99 \dots 9}.$$

Числа, модуль которых меньше  $X_0$ , воспринимаются ЭВМ как нуль, точнее как машинный нуль. Если при выполнении расчетов будет получен результат меньше, чем  $X_0$ , то это будет воспринято как потеря порядка. Обычно в подобной ситуации результат полагается равным нулю, и вычисления продолжаются.

На рис. 2 показана "машинная" числовая ось, на которой отмечены  $X_0$  и  $X_{\infty}$ . Числа располагаются на оси неравномерно. Их плотность возрастает по мере приближения к нулю.



Рис. 2. "Машинная" числовая ось

На рис. 2 вблизи единицы отмечена небольшая область  $\epsilon_m$ , которую называют машинное эпсилон. Параметр  $\epsilon_m$  весьма важен, так как он характеризует относительную точность представления чисел в компьютере. В зависимости от способа округления чисел в ЭВМ величина  $\epsilon_m$  определяется первым отбрасываемым или последним сохраняемым разрядом мантиссы.

Следует иметь ввиду, что длина мантиссы в памяти компьютера устанавливается программно.

## **Литература**

1. Бахвалов Н.С. Численные методы. – М.: Наука, 1975.
2. Калиткин, Н. Н. Численные методы. М.: Наука, 1978.
3. Самарский, А. А. Введение в численные методы. М.: Наука, 1987.
4. Самарский А.А., Гулин А.В. Численные методы: Учеб. пособие для вузов.—М.: Наука, 1989.

## Оглавление

### Введение 3

### I. Решение алгебраических уравнений 4

1. РЕШЕНИЕ НЕЛИНЕЙНЫХ УРАВНЕНИЙ С ПОМОЩЬЮ ЧИСЛЕННЫХ МЕТОДОВ.....	4
1.1. Виды нелинейных уравнений.....	4
1.2. Графическое исследование уравнения.....	6
1.3. Модификация табличного способа .....	7
1.4. Метод половинного деления.....	8
1.5. Метод простой итерации.....	10
1.6. Метод Ньютона (метод касательных или метод линеаризации).....	11
2. РЕШЕНИЕ СИСТЕМ ЛИНЕЙНЫХ АЛГЕБРАИЧЕСКИХ УРАВНЕНИЙ С ПОМОЩЬЮ ЧИСЛЕННЫХ МЕТОДОВ .....	14
2.1. Теоретические сведения о системах линейных алгебраических уравнений .....	14
2.2. Метод Гаусса .....	15

### II. Решение дифференциальных уравнений 18

1. ХАРАКТЕРИСТИКА ДИФФЕРЕНЦИАЛЬНЫХ УРАВНЕНИЙ.....	18
2. РЕШЕНИЕ ОБЫКНОВЕННЫХ ДИФФЕРЕНЦИАЛЬНЫХ УРАВНЕНИЙ С ПОМОЩЬЮ ЧИСЛЕННЫХ МЕТОДОВ.....	19
2.1. Задача Коши .....	19
2.2. Метод Эйлера.....	19
2.3. Модифицированный метод Эйлера.....	21
2.4. Метод Рунге-Кутта четвертого порядка .....	22
2.5. Погрешность решения и выбор шага.....	24
3. РЕШЕНИЕ ДИФФЕРЕНЦИАЛЬНЫХ УРАВНЕНИЙ В ЧАСТНЫХ ПРОИЗВОДНЫХ .....	25
3.1. Классификация уравнений по математической форме .....	25
3.2. Основы метода конечных разностей.....	27
3.3. Аппроксимация уравнения эллиптического типа .....	29
3.4. Аппроксимация уравнения гиперболического типа.....	31
3.5. Аппроксимация уравнения параболического типа .....	33
3.6. Погрешность решения.....	34
ПРИЛОЖЕНИЕ.....	35
ЛИТЕРАТУРА .....	38