ГОСУДАРСТВЕННЫЙ КОМИТЕТ СВЯЗИ, ИНФОРМАТИЗАЦИИ И ТЕЛЕКОММУНИКАЦИОННЫХ ТЕХНОЛОГИЙ РЕСПУБЛИКИ УЗБЕКИСТАН

ФЕРГАНСКИЙ ФИЛИАЛ ТАШКЕНТСКОГО УНИВЕРСИТЕТА ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ

ФАКУЛЬТЕТ «КОМПЬЮТЕРНЫЙ ИНЖИНИРИНГ»

КАФЕДРА ЕСТЕСТВЕННЫХ ДИСЦИПЛИН

«УТВЕРЖДАЮ»
Заместитель директора по
учебной и научной работе
____ доц. Ф. Ю. Полвонов
29 августа 2015 г.

МЕТОДИЧЕСКОЕ ПОСОБИЕ

ПО ПРЕДМЕТУ «ЧИСЛЕННЫЕ МЕТОДЫ И ПРОГРАММИРОВАНИЕ»

ФЕРГАНА – 2015

ГОСУДАРСТВЕННЫЙ КОМИТЕТ СВЯЗИ, ИНФОРМАТИЗАЦИИ И ТЕЛЕКОММУНИКАЦИОННЫХ ТЕХНОЛОГИЙ РЕСПУБЛИКИ УЗБЕКИСТАН

ФЕРГАНСКИЙ ФИЛИАЛ ТАШКЕНТСКОГО УНИВЕРСИТЕТА ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ

ФАКУЛЬТЕТ «КОМПЬЮТЕРНЫЙ ИНЖИНИРИНГ»

КАФЕДРА ЕСТЕСТВЕННЫХ ДИСЦИПЛИН

«УТВЕРЖДАЮ»
Заместитель директора по
учебной и научной работе
____ доц. Ф. Ю. Полвонов
29 августа 2015 г.

МЕТОДИЧЕСКОЕ ПОСОБИЕ

ПО ПРЕДМЕТУ «ЧИСЛЕННЫЕ МЕТОДЫ И ПРОГРАММИРОВАНИЕ»

Рассмотрено на заседании
кафедры, протокол № ___
от ___ 2015 г.

Рекомендовано для утверждения
метод. комиссией, протокол № ___
от ___ 2015 г.

Составил: ст. преп. Ё. А. Юсупов

ФЕРГАНА – 2015

Ё. А. Юсупов. Методическое пособие по предмету «Численные методы и программирование». Фергана: Ферганский филиал ТУИТ, 2015, 128 с.

Данное методическое пособие предназначено для проведения лекционных, лабораторных и практических занятий по дисциплине «Численные методы и программирование» для студентов обучающихся по направлениям ИКТ. В пособии содержится теоретический материал, задачи и необходимый материал для выполнения каждой лабораторный работы, алгоритмы решения в табличном процессоре Excel и программных пакетов как MatLab, Maple, которые каждый студент может легко адаптировать для выполнения индивидуального задания. Приведены образцы решения наиболее важных задач численного анализа.

СОДЕРЖАНИЕ

ВВЕДЕНИЕ

1. Этапы решения технических задач на ЭВМ

Реальные инженерные и физические задачи во всех областях науки и техники обычно решаются посредством использования двух подходов:

– физического эксперимента;

– предварительного анализа конструкций, схем, явлений с целью выбора каких-то их оптимальных параметров.

Первый подход связан с большими и не всегда оправданными затратами материальных и временных ресурсов.

Второй подход связан с *математическим моделированием*, в основе которого заложены знания фундаментальных законов природы и построение на их основе *математических моделей* для произвольных технических и научных задач.

Математические модели представляют собой упрощенное описание исследуемого явления с помощью *математических символов и операций над ними*. Математические модели разрабатываются с соблюдением корректности и адекватности по отношению к реальным процессам, но, как правило, с учетом простоты их технической реализации.

Практика показывает, что возникающие и истребованные технические решения во многом однозначны, что определяет ограниченное число существенно полезных математических моделей, извлекаемых из стандартного справочника «Курс высшей математики». К примеру, из арсенала этих моделей можно назвать такие как линейные и нелинейные уравнения, системы линейных и нелинейных уравнений, дифференциальные уравнения (ДУ), разновидности интегралов, функциональные зависимости, «целевые» функции для решения задач оптимизации и др.

При математическом моделировании важным моментом является первоначальная *математическая постановка задачи*. Она предполагает описание математической модели и указания цели ее исследования. Для одной и той же математической модели могут быть сформулированы и решены различные математические задачи. Например, для наиболее распространенной модели, такой как функциональная зависимость $y = f(x)$ могут быть сформулированы следующие математические задачи:

1) найти экстремальное значение функции $f(x)$: $\max\ f(x)$ или $\min\ f(x)$;

2) найти значение x, при котором $f(x) = 0$;

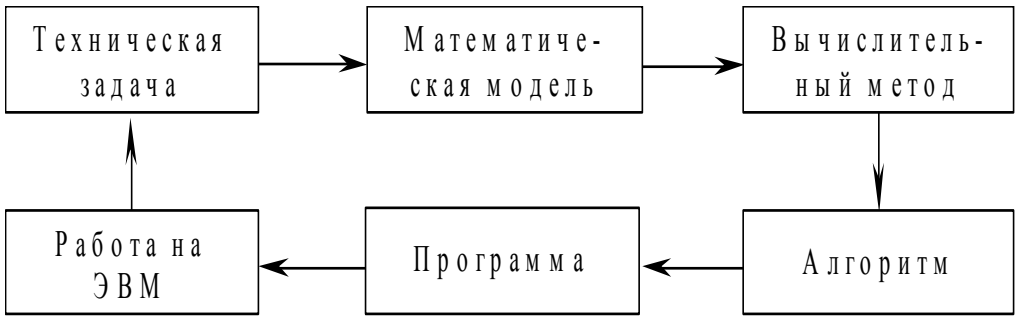
3) найти значение производной $f'(x)$, значение интеграла $\int_a^b f(x)dx$ и т.д.

Бурное развитие вычислительной техники выдвинуло на передний план при решении практических инженерных и научных задач вычислительную математику и программирование.

Вычислительная математика изучает построение и исследование численных методов решения математических задач посредством реализации соответствующих математических моделей.

Программирование обеспечивает техническую реализацию их.

Обобщенную схему математического моделирования можно представить следующим образом:



При реализации данного цикла требуют пристального внимания все его компоненты. Заключительным его этапом является получение численного результата и сопоставление его с целевой установкой и, как правило, для достижения желаемого, или приемлемого результата, всегда возникает необходимость изменения или математической модели, или вычислительного метода, или алгоритма, или программы.

Следует подчеркнуть важность и таких этапов данной технологии решения задач на ЭВМ как проведение расчетов и анализ результатов. (А именно, подготовка исходных данных, обоснование выбора вычислительного метода, корректность и точность решения). Важным моментом является также *экономичность* выбора: способа решения задачи, численного метода, модели ЭВМ, вычислительной среды.

2. Методы реализации математических моделей

Методы реализации математических моделей можно разделить на три группы:

1) графические;

2) аналитические;

3) численные.

Указанные методы используются как самостоятельно, так и совместно.

Графические методы позволяют оценивать порядок искомых величин и направление расчетных алгоритмов.

Аналитические методы (точные, приближенные) упрощают фрагментарные расчеты и позволяют успешно решать задачи оценки корректности и точности численных решений.

Основным инструментом реализации математических моделей являются численные методы, позволяющие свести решение задачи к вычислению конеч-

ного числа арифметических действий над числами и получение этого решения в виде числовых значений. Решение, получаемое численными методами, обычно является приближенным, т.е. содержит некоторую погрешность.

Раздел 1. Элементы теории погрешностей

1.1. Постановка задачи

Важнейшим моментом при математическом моделировании является обеспечение достоверности полученных решений. Но из практики известно, что лишь в редких случаях удается найти метод решения, приводящий к точному результату. Как правило, приближенные решения используются совместно с точными решениями, поэтому, наряду с выбором вычислительного метода, с точки зрения оптимальности алгоритма его реализации, важной задачей является оценка степени точности получаемого решения. Ее принято оценивать некоторой численной величиной, называемой ***погрешностью***.

При решении любой практической задачи необходимо всегда указывать требуемую ***точность результата***. В связи с этим необходимо уметь:

1) зная заданную точность исходных данных, оценивать точность результата (прямая задача теории погрешностей);

2) зная требуемую точность результата, выбирать необходимую точность исходных данных (обратная задача теории погрешностей).

1.2. Источники погрешностей

На рассмотренных выше этапах математического моделирования имеют место следующие источники погрешностей:

1) погрешность математической модели;

2) погрешность исходных данных (неустранимая погрешность):

3) погрешность численного метода;

4) вычислительная погрешность.

Погрешность математической модели возникает из-за стремления обеспечить сравнительную простоту ее технической реализации и доступности исследования. Нужно иметь в виду, что конкретная математическая модель (ММ), прекрасно работающая в одних условиях, может быть совершенно неприменима в других. С точки зрения потребителя, важным является правильная оценка области ее (ММ) применения.

Погрешность численного метода (погрешность аппроксимации), связана, например, с заменой интеграла суммой, с усечением рядов при вычислении функций, с интерполированием табличных значений функциональных зависимостей и т.п. Как правило, погрешность численного метода регулируема и может быть уменьшена до любого разумного значения путем изменения некоторого параметра.

Вычислительная погрешность возникает из-за округления чисел, промежуточных и окончательных результатов счета. Она зависит от правил и необходимости округления, а также от алгоритмов численного решения.

Вспомним технологию округления чисел.

1. Если старший отбрасываемый разряд меньше 5, то предшествующая ему цифра в числе не меняется.

2. Если старший отбрасываемый разряд больше 5, то предшествующая цифра в числе увеличивается на 1.

3. Если старший отбрасываемый разряд равен 5, то по общепринятому соглашению предшествующая ему четная цифра в числе не меняется (например, $c = 3{,}965$; $c^* \approx 3{,}96$), а нечетная – увеличивается на единицу (например, $c = 3{,}915$; $c^* \approx 3{,}92$).

4. При округлении целого числа отброшенные знаки не следует заменять нулями, надо применять умножение на соответствующие степени 10.

В основе процессов округления лежит идея минимальности разности значения c и его округления c^*.

Пример 1. Округлить число c на соответствующее количество знаков:

1) $c = 1{,}9396712$;	2) $c = 245{,}351365$;
$c^* = 1{,}939671$;	$c^* = 245{,}35136$;
$c^* = 1{,}93967$;	$c^* = 245{,}3514$;
$c^* = 1{,}9397$;	$c^* = 245{,}351$;
$c^* = 1{,}940$;	$c^* = 245{,}35$;
$c^* = 1{,}94$;	$c^* = 245{,}4$;
$c^* = 1{,}9$;	$c^* = 245$;
$c^* = 2$;	$c^* = 2{,}4 \cdot 10^2$;
	$c^* = 2 \cdot 10^2$;

Пример 2. Для обоснования необходимости применения округлений в целях экономии памяти приведем следующий пример. Задано выражение

$$S = 25{,}71 \cdot 1{,}42 - 3{,}21 \cdot 7{,}46 + 0{,}93 \cdot 7{,}75 - 4{,}31 \cdot 2{,}69 .$$

1. Вычислить S точно:

$$S = 36{,}5082 - 23{,}9466 + 7{,}2075 - 11{,}5939 = 8{,}1752 .$$

2. Вычислить S и округлить его до двух знаков после запятой:

$$S_1^* = 8{,}18 .$$

3. Вычислить каждое произведение с двумя знаками после запятой и просуммировать:

$$S_2^* = 36{,}51 - 23{,}95 + 7{,}21 - 11{,}59 = 8{,}18 .$$

1.3. Приближенные числа и оценка их погрешностей

При численном решении задач приходится оперировать двумя видами чисел – *точными* и *приближенными*. К точным числам относятся числа, которые дают истинное значение исследуемой величины. К приближенным относятся числа, близкие к истинному значению, причем степень близости и определяется *погрешностью вычислений*.

Результатами вычислений являются, как правило, только приближенные числа. Поэтому для указания области неопределенности результата вводятся некоторые специальные понятия, широко используемые при подготовке исходных данных или (и) оценке погрешности численных решений.

Если x – точное, вообще говоря, неизвестное значение некоторой величины, а a – его приближение, то разность $x–a$ называется *ошибкой*, или *погрешностью приближения*. Часто знак ошибки $x – a$ неизвестен, поэтому используется так называемая ***абсолютная погрешность*** $\Delta(X)$ приближенного числа a, определяемая равенством

$$\Delta(X) = | x - a | , \qquad (1)$$

откуда имеем

$$x = a \pm \Delta(X). \qquad (2)$$

Изучаемая числовая величина x *именованная*, т.е. определяется в соответствующих единицах измерения, например, в сантиметрах, килограммах и т.п. Погрешность (1) имеет ту же размерность.

Однако часто возникает необходимость заменить эту погрешность безразмерной величиной – ***относительной погрешностью***. При этом из-за незнания точного значения изучаемой величины принято называть относительной погрешностью величину

$$\delta(X)=\frac{\Delta(X)}{|a|}=\left|\frac{x-a}{a}\right| . \qquad (3)$$

Относительную погрешность часто выражают в процентах: $\delta(X)=\frac{\Delta(X)}{|a|}\cdot 100\%$.

Это погрешность на единицу измеряемой физической величины. Она сопоставима в идентичных экспериментах, т.е. характеризует качество измерения. А именно, точность результата лучше характеризуется его $\delta(X)$, так как абсолютная погрешность $\Delta(X)$ не достаточна, к примеру, для характеристики качества измерения двух стержней l_1 = 100,8 см ± 0,1 см и l_2 = 5,2 см ± 0,1 см. Очевидно, что качество измерения первого значительно выше.

В связи с тем, что точное значение x, как правило, неизвестно, то формулы (1)–(3) носят сугубо теоретический характер.

Для практических целей вводится понятие ***предельной погрешности***. *Предельная абсолютная погрешность* Δa – это верхняя оценка модуля абсолютной погрешности числа x, т.е.

$$| \Delta x | \le \Delta a.$$

При произвольном выборе, Δa всегда стремятся каким-либо образом взять *наименьшим*.

Истинное значение числа x будет находиться в интервале с границами ($a – \Delta a$) – с недостатком и ($a + \Delta a$) – с избытком, т.е.

$$(a - \Delta a) \le x \le (a + \Delta a) .$$

Условились для приближенных чисел по результатам округлений в качестве Δa принимать единицу или 1/2 единицы оставленного разряда числа. Первое условие называют погрешностью в «широком» смысле, второе в «узком» смысле.

Пример для второго условия:

a	51,7	–0,0031	16	16,00
Δa	0,05	0,00005	0,5	0,005

Предельная относительная погрешность $\delta(a)=\dfrac{\Delta a}{|a|}$ также может выражаться в процентах. При локальных ручных расчетах, и на этапе подготовки исходных данных существуют определенные правила оценки предельных погрешностей для арифметических операций (формулы – (4)):

$$\delta(a\pm b)=\frac{a\delta(a)+b\delta(b)}{a\pm b};\qquad \delta(ab)=\delta(a)+\delta(b);$$

$$\delta(a/b)=\delta(a)+\delta(b);\qquad \delta(a^m)=m\cdot\delta(a);$$

$$\Delta(a\pm\Delta b)=\Delta a+\Delta b;$$

$$\Delta(a\cdot b)=a\cdot b[\,\delta(a)+\delta(b)]=b\Delta a+a\Delta b;$$

$$\Delta(a/b)=\frac{a}{b}[\delta(a)+\delta(b)]=\frac{b\Delta a+a\Delta b}{b^2};$$

$$\Delta(a^m)=m\cdot a^{m-1}\Delta a;$$

где Δ – предельная абсолютная погрешность;

δ – относительная предельная погрешность;

m – рациональное число.

Следует отметить, что приведенные оценки погрешностей приближенных чисел справедливы, если в записи этих чисел все «***значащие***» цифры «***верны***». Определение этих понятий рассмотрим ниже.

1.4. Правила записи приближенных чисел

Запись приближенных чисел должна подчиняться правилам, связанным с понятиями *верных значащих цифр*.

Любое десятичное число

$$x = \pm\alpha_n\,\alpha_{n-1}\ldots\alpha_1\,\alpha_0\,\alpha_{-1}\alpha_{-2}\ldots\alpha_{-m}$$

представимо в виде

$$x = \pm\alpha_n 10^n + \alpha_{n-1}10^{n-1} + \ldots\,\alpha_1 10 + \alpha_0 + \alpha_{-1}10^{-1} + \alpha_{-2}10^{-2} + \ldots + \alpha_{-m}10^{-m},$$

где α_i – цифры числа, 10^i – их позиция ($\pm i$).

Рассмотрим пример:

$$1358{,}7604 = 1\cdot10^3 + 3\cdot10^2 + 5\cdot10 + 8 + 7\cdot10^{-1} + 6\cdot10^{-2} + 0\cdot10^{-3} + 4\cdot10^{-4}.$$

Первая слева отличная от нуля цифра числа x и все расположенные справа от нее цифры называются ***значащими***, т.е. числа 25,047 и –0,00250 имеют соответственно 5 и 3 значащих цифр. Последнее число может быть записано $-2{,}50\cdot10^{-3}$.

Значащая цифра α_i называется ***верной*** (в узком смысле), если абсолютная погрешность числа не превосходит 1/2 единицы разряда, соответствующего этой цифре, т.е. $\Delta a \le 1/2\cdot10^{i}$, где 10^{i} указывает номер разряда ($\pm i$).

Пусть $x^* = 12{,}396$ (x^* приближение x) и известно $\Delta x^* = 0{,}03$. Согласно определению здесь:

$$\Delta x^* > 1/2\cdot10^{-3}; \qquad \Delta x^* > 1/2\cdot10^{-2} \qquad \text{и} \qquad \Delta x^* < 1/2\cdot10^{-1}.$$

Значит, верными знаками будут 1, 2, 3, а 9 и 6 *сомнительные*.

Пусть $x^* = 0{,}037862$ и $\Delta x^* = 0{,}07$. Здесь $\Delta x^* > 1/2\cdot10^{-1}$. Значит все значащие цифры сомнительные.

Если число записано с указанием его абсолютной погрешности

$$S = 20{,}7428; \qquad \Delta S = 0{,}0926\ ,$$

то число верных знаков можно отсчитывать от первой значащей цифры числа до первой значащей цифры его абсолютной погрешности. Здесь верные цифры 2, 0, 7.

Существуют определенные соглашения при оперировании понятиями верных значащих цифр.

1) Если число имеет лишь верные цифры, то и его округление имеет также только верные цифры.

2) Совпадение приближенного значения, имеющего все верные значащие цифры, с точным значением *не обязательно*.

3) Абсолютные и относительные погрешности числа принято округлять в большую сторону, так как при округлениях границы неопределенности числа, как правило, увеличиваются.

4) При изменении формы записи числа количество значащих цифр не должно меняться, т.е. необходимо соблюдать равносильность преобразований, например

$7500 = 0{,}7500\cdot10^{4}$; $0{,}110\cdot10^{2} = 11{,}0$; – равносильные преобразования;

$7500 = 0{,}75\cdot10^{4}$; $0{,}110\cdot10^{2} = 11$; – неравносильные преобразования.

Здесь два нуля в первом и один ноль во втором выражениях переведены в разряд незначащих цифр, поэтому следует использовать записи $7500 = 0{,}7500\cdot10^{4}$ и $0{,}110\cdot10^{2} = 11{,}0$.

5) При вычислениях желательно сохранять такое количество значащих цифр, чтобы их число не превышало числа верных цифр более чем на одну – две единицы.

6) Верные значащие цифры числа характеризуют ориентировочно относительную погрешность по схеме: одна верная цифра 10%, две – 1%, три – 0,1%

и т.д. Верные значащие цифры после запятой характеризуют абсолютную погрешность или в «узком» или в «широком» смысле.

Нормализованная форма числа. Приближенные числа принято записывать таким образом, чтобы все цифры числа, кроме нулей впереди, если они есть, были значащими и верными цифрами.

Обычную форму записи числа, рассмотренную выше, называют записью *с фиксированной точкой*, а числа $0{,}63750\cdot10^6$; $637{,}50\cdot10^3$ и $6{,}3750\cdot10^5$ записаны *в форме с плавающей точкой*. Запись числа с плавающей точкой, как следует из примера, не является однозначной. Для устранения этой неоднозначности принято первый множитель брать меньше единицы, и он должен состоять только из значащих цифр (кроме нуля целых), т.е. первая цифра после запятой всегда отлична от нуля.

Такая форма записи числа называется *нормализованной*. В данном примере ею является запись $0{,}63750\cdot10^6$, а для числа $-0{,}00384$ нормализованная форма $-0{,}384\cdot10^{-2}$.

Итак, запись числа x в нормализованной форме имеет вид

$$x = x^0\cdot10^p; \qquad \text{где} \qquad 0{,}1 \le |\, x^0 \,| < 1.$$

Число x^0 называется мантиссой числа x, а число p – его порядком. Например, для числа $620 = 0{,}620\cdot10^3$ мантиссой является 0,620, а порядком – число 3. Заметим, что в этой записи все цифры после запятой верные.

1.5. Задачи теории погрешностей

Прямая задача теории погрешностей

Пусть в некоторой области G n-мерного числового пространства рассматривается непрерывно дифференцируемая функция

$$y = f(x_1, \dots, x_n).$$

Пусть в точке $(x_1, \dots, x_n)$, принадлежащей области G, нужно вычислить ее (функции) значение. Известны лишь приближенные значения аргументов $(a_1, \dots, a_n) \in G$, и их погрешности. Естественно, что это будет приближенное значение

$$y^* = f(a_1, a_2, \dots, a_n).$$

Нужно оценить его абсолютную погрешность

$$\Delta y^* = |\, y - y^* \,| \lesssim \sum_{i=1}^{n} \Delta a_i \left| \frac{\partial}{\partial a_i} f(a_i, \dots, a_n) \right| .$$

Для функции одного аргумента $y = f(x)$ ее абсолютная погрешность, вызываемая достаточно малой погрешностью Δa, оценивается величиной

$$\Delta y^* = \lesssim |\, f'(a) \,| \cdot \Delta a \,.$$

Обратная задача теории погрешностей

Она состоит в определении допустимой погрешности аргументов по допустимой погрешности функции.

Для функции одной переменной $y = f(x)$ абсолютную погрешность можно вычислить приближенно по формуле

$$\Delta a = \frac{1}{|f'(a)|} \cdot \Delta y, \quad f'(a) \neq 0.$$

Для функций нескольких переменных $y = f(x_1, \ldots, x_n)$ задача решается при следующих ограничениях.

Если значение одного из аргументов значительно *труднее измерить* или вычислить с той же точностью, что и значение остальных аргументов, то погрешность именно этого аргумента и согласовывают с требуемой погрешностью функции.

Если значения всех аргументов можно одинаково легко определить с любой точностью, то применяют ***принцип равных влияний***, т.е. учитывают, что все слагаемые

$$\left| \frac{\partial f}{\partial x_i} \right| \cdot \Delta a_i, \; i = 1, \ldots, n,$$

равны между собой. Тогда абсолютные погрешности всех аргументов определяются формулой

$$\Delta a_i = \frac{\Delta y}{n \cdot |\partial f / \partial x_i|}, \; i = 1, \ldots, n.$$

1.6. Понятия устойчивости, корректности постановки задач и сходимости численного решения

Пусть в результате решения задачи по исходному значению величины x находится значение искомой величины y. Если исходная величина имеет абсолютную погрешность Δx, то решение y имеет погрешность Δy.

Задача называется ***устойчивой по исходному параметру*** x, если решение y непрерывно зависит от x, т.е. малое приращение исходной величины x приводит к малому приращению искомой величины y. Другими словами, малые погрешности в исходной величине приводят к малым погрешностям в результате расчетов.

Отсутствие устойчивости означает, что даже незначительные погрешности в исходных данных приводят к большим погрешностям в решении или вовсе к неверному результату.

Задача называется поставленной ***корректно***, если для *любых* значений исходных данных из некоторого класса ее решение существует, единственно и устойчиво по исходным данным.

Понятие сходимости численного решения вводится для итерационных процессов. По результатам многократного повторения итерационного процесса

получаем последовательность приближенных значений $\overline{x_1}, \overline{x_2}, \ldots, \overline{x_n}, \ldots$. Говорят, что эта последовательность сходится к точному решению, если $\lim\limits_{n\to\infty} \overline{x_n} = \overline{a}$.

Таким образом, для получения решения задачи с необходимой точностью ее постановка должна быть корректной, а используемый численный метод должен обладать устойчивостью и сходимостью. Эти понятия будут рассматриваться в последующих разделах курса.

1.7. Некоторые обобщенные требования к выбору численных методов

Рассмотренные выше вопросы о погрешностях являются одними из важнейших моментов при выборе численного метода. В основе выбора численного метода лежат следующие соображения.

1) Можно утверждать, что нет ни одного метода, пригодного для решения всех задач одного и того же класса. Поэтому всегда стоит задача выбора численного метода (ЧМ), сообразуясь из конкретной технической задачи.

2) Численный метод можно считать удачно выбранным:

– если его погрешность в несколько раз меньше неустранимой погрешности, а погрешность округлений в несколько раз меньше погрешности метода;

– если неустранимая погрешность отсутствует, то погрешность метода должна быть несколько меньше заданной точности;

– завышенное снижение погрешности численного метода приводит не к повышению точности результатов, а к необоснованному увеличению объема вычислений.

3) Предпочтение отдается методу, который:

– реализуется с помощью меньшего числа действий;

– требует меньшего объема памяти ЭВМ;

– логически является более простым.

Перечисленные условия обычно противоречат друг другу, поэтому часто при выборе численного метода приходится соблюдать компромисс между ними.

4) Численный метод должен обладать устойчивостью и сходимостью.

5) По возможности нужно прибегать к существующему программному обеспечению ЭВМ для решения типовых задач.

6) Нужно помнить всегда, что ЭВМ многократно увеличивает некомпетентность Исполнителя технической задачи.

Раздел 2. Решение систем линейных алгебраических уравнений

2.1. Основные понятия и определения

Системы линейных алгебраических уравнений (СЛАУ) являются важной математической моделью линейной алгебры. На их базе ставятся такие практические математические задачи, как:

– непосредственное решение линейных систем;

– вычисление определителей матриц;

– вычисление элементов обратных матриц;

– определение собственных значений и собственных векторов матриц.

Решение линейных систем является одной из самых распространенных задач вычислительной математики. К их решению сводятся многочисленные практические задачи нелинейного характера, решения дифференциальных уравнений и др.

Вторая и третья задачи являются также и компонентами технологии решения самих линейных систем.

Обычно СЛАУ *n*-го порядка записывается в виде

$$\sum_{j=1}^{n} a_{ij} x_j = b_i; \quad i=\overline{1,n}$$

или в развернутой форме

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

или в векторной форме

$$A\bar{x} = \bar{b}\,, \qquad (2)$$

где

$$A = \begin{bmatrix} a_{11}\ a_{12} \cdots a_{1n} \\ a_{21}\ a_{22} \cdots a_{2n} \\ \cdots \\ a_{n1}\ a_{n2} \cdots a_{nn} \end{bmatrix}; \qquad \bar{x} = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix}; \qquad \bar{b} = \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_n \end{bmatrix}.$$

В соотношениях (2):

A называется основной матрицей системы с n^2 элементами;

$\bar{x} = (x_1, x_2, \ldots, x_n)^{\mathrm{T}}$ – вектор-столбец неизвестных;

$\bar{b} = (b_1, b_2, \ldots, b_n)^{\mathrm{T}}$ – вектор-столбец свободных членов.

Определителем (детерминантом – det) матрицы A n-го порядка называется число D (det A), равное

$$|A| = D = \det A = \begin{vmatrix} a_{11}\ a_{12} \cdots a_{1n} \\ a_{21}\ a_{22} \cdots a_{2n} \\ \cdots \\ a_{n1}\ a_{n2} \cdots a_{nn} \end{vmatrix} = \sum (-1)^k a_{1\alpha}\, a_{2\beta} \cdots\ a_{n\omega}\ .$$

Здесь индексы α, β, ..., ω пробегают все возможные $n!$ перестановок номеров 1, 2, ..., n; k – число инверсий в данной перестановке.

Первоначальным при решении СЛАУ (1) является анализ вида исходной матрицы A и вектора-столбца свободных членов $\bar{b}$ в (2).

Если все свободные члены равны нулю, т.е. $\bar{b} = 0$, то система $A\bar{x} = 0$ называется ***однородной***. Если же $\bar{b} \neq 0$, или хотя бы одно $b_i \neq 0$ ($i = \overline{1,n}$), то система (2) называется ***неоднородной***.

Квадратная матрица A называется ***невырожденной***, или *неособенной*, если ее определитель $|A| \neq 0$. При этом система (1) имеет единственное решение.

При $|A| = 0$ матрица A называется ***вырожденной***, или *особенной*, а система (1) не имеет решения, либо имеет бесконечное множество решений.

Если $|A| \approx 0$ система (1) называется ***плохо обусловленной***, т.е. решение очень чувствительно к изменению коэффициентов системы.

В ряде случаев получаются системы уравнений с матрицами специальных видов: диагональные, трехдиагональные (частный случай ленточных), симметричные ($a_{ij} = a_{ji}$), единичные (частный случай диагональной), треугольные и др.

Решение системы (2) заключается в отыскании вектора-столбца $\bar{x} = (x_1, x_2, \ldots, x_n)^T$, который обращает каждое уравнение системы в тождество.

Существуют две величины, характеризующие степень отклонения полученного решения от точного, которые появляются в связи с округлением и ограниченностью разрядной сетки ЭВМ, – погрешность ε и «невязка» r:

$$\begin{cases} \varepsilon = \bar{x} - \overline{x^*}; \\ r = \bar{B} - A\overline{x^*}; \end{cases} \qquad (3)$$

где $\overline{x^*}$ – вектор решения. Как правило, значения вектора $\bar{x}$ – неизвестны.

Доказано, что если $\varepsilon \approx 0$, то и $r = 0$. Обратное утверждение не всегда верно. Однако если система не плохо обусловлена, для оценки точности решения используют невязку r.

2.2. Методы решения СЛАУ

Методы решения СЛАУ делятся на две группы:

– прямые (точные) методы;

– итерационные (приближенные) методы.

К ***прямым*** методам относятся такие методы, которые, в предположении, что вычисления ведутся без округлений, позволяют получить точные значения неизвестных. Они просты, универсальны и используются для широкого класса

систем. Однако они не применимы к системам больших порядков ($n < 200$) и к плохо обусловленным системам из-за возникновения больших погрешностей. К ним можно отнести: *правило Крамера*, методы *обратных матриц*, *Гаусса*, *прогонки*, *квадратного корня* и др.

К ***приближенным*** относятся методы, которые даже в предположении, что вычисления ведутся без округлений, позволяют получить решение системы лишь с заданной точностью. Это итерационные методы, т.е. методы последовательных приближений. К ним относятся методы *простой итерации*, *Зейделя*.

2.2.1. Прямые методы решения СЛАУ

1. Правило Крамера

Рассмотрим систему (1). Как отмечалось выше, если определитель этой системы не равен нулю, то будет иметь место единственное решение. Это необходимое и достаточное условие. Тогда по правилу Крамера

$$x_k = \frac{D_k}{D},\ k = \overline{1,n}\ , \tag{4}$$

где D_k – определитель, получающийся из D при замене элементов a_{1k}, a_{2k}, ..., a_{nk} k-го столбца (соответствующими) свободными членами b_1, b_2, ..., b_n из (1), или

$$D_k = \sum_{i=1}^{n} A_{ik}\, b_i\ ,\ k = \overline{1,n}\ ,$$

где A_{ik} алгебраическое дополнение элемента a_{ik} в определителе D. Стоит существенная проблема вычисления определителей высоких порядков.

2. Метод обратных матриц

Дана система $A\bar{x} = \bar{b}$. Умножим левую и правую части этого выражения на A^{-1}:

$$A^{-1} A\, \bar{x} = A^{-1}\bar{b}\,; \qquad \bar{x} = A^{-1}\bar{b}\,.$$

При его реализации стоит проблема нахождения обратной матрицы A^{-1}, с выбором экономичной схемы ее получения и с достижением приемлемой точности. Эти вопросы рассмотрим ниже.

3. Метод Гаусса

Этот метод является наиболее распространенным методом решения СЛАУ. В его основе лежит идея последовательного исключения неизвестных, в основном, приводящая исходную систему к треугольному виду, в котором все коэффициенты ниже главной диагонали равны нулю. Существуют различные вычислительные схемы, реализующие этот метод. Наибольшее распространение имеют схемы с выбором главного элемента либо по строке, либо по столбцу, либо по всей матрице. С точки зрения простоты реализации, хотя и с потерей точности, перед этими схемами целесообразней применять так называемую схему единственного деления. Рассмотрим ее суть.

Посредством первого уравнения системы (1) исключается x_1 из последующих уравнений. Далее посредством второго уравнения исключается x_2 из последующих уравнений и т.д. Этот процесс называется ***прямым ходом Гаусса***. Исключение неизвестных повторяется до тех пор, пока в левой части последнего n-го уравнения не останется одно неизвестное x_n

$$a'_{nn}x_n = b', \tag{5}$$

где a'_{nn} и b' – коэффициенты, полученные в результате линейных (эквивалентных) преобразований.

Прямой ход реализуется по формулам

$$\left.\begin{aligned} a^*_{mi} &= a_{mi} - a_{ki}\frac{a_{mk}}{a_{kk}}, \quad k=\overline{1,n-1}; \quad i=\overline{k,n}; \\ b^*_m &= b_m - b_k\frac{a_{mk}}{a_{kk}}, \quad m=\overline{k+1,n} \end{aligned}\right\} \tag{6}$$

где m – номер уравнения, из которого исключается x_k;

k – номер неизвестного, которое исключается из оставшихся $(n - k)$ уравнений, а также обозначает номер уравнения, с помощью которого исключается x_k;

i – номер столбца исходной матрицы;

a_{kk} – главный (ведущий) элемент матрицы.

Во время счета необходимо следить, чтобы $a_{kk} \neq 0$. В противном случае прибегают к перестановке строк матрицы.

Обратный ход метода Гаусса состоит в последовательном вычислении $x_n, x_{n-1}, ..., x_1$, начиная с (5) по алгоритму

$$x_n = b' / a'_{nn}; \qquad x_k = \frac{1}{a'_{kk}}\left[b'_k - \sum_{i=k+1}^{n} a'_{ki}\,x_i\right], \quad k=\overline{n-1,1}. \tag{7}$$

Точность полученного решения оценивается посредством «невязки» (3). В векторе невязки $(r_1, r_2, ..., r_n)^T$ отыскивается максимальный элемент и сравнивается с заданной точностью ε. Приемлемое решение будет, если $r_{\max} < \varepsilon$. В противном случае следует применить схему уточнения решения.

Уточнение корней

Полученные методом Гаусса приближенные значения корней можно уточнить.

Пусть для системы $A\bar{x} = \bar{b}$ найдено приближенное решение $\overline{x_0}$, не удовлетворяющее по «невязке». Положим тогда $\bar{x} = \overline{x_0} + \bar{\delta}$. Для получения поправки $\delta = (\delta_1, \delta_2, ..., \delta_n)^T$ корня $\overline{x_0}$ следует рассмотреть новую систему

$$A(\overline{x_0 + \delta}) = \bar{b} \qquad \text{или} \qquad A\bar{\delta} = \bar{\varepsilon},$$

где $\bar{\varepsilon} = \bar{b} - A\overline{x_0}$ – невязка для исходной системы.

Таким образом, решая линейную систему с прежней матрицей A и новым свободным членом $\bar{\varepsilon} = (\varepsilon_1, \varepsilon_2, ..., \varepsilon_n)^T$, получим поправки $(\delta_1, \delta_2, ..., \delta_n)$.

Пример решения СЛАУ по методу Гаусса (с точностью до трех знаков). Нужно уточнить корни до 10^{-4}:

$$\begin{cases} 6x_1 - x_2 - x_3 = 11{,}33; \\ -x_1 + 6x_2 - x_3 = 32; \\ -x_1 - x_2 + 6x_3 = 42. \end{cases}$$

В результате $x_1^{(0)} = 4{,}67$; $x_2^{(0)} = 7{,}62$; $x_3^{(0)} = 9{,}05$. Невязки равны $\varepsilon_1^{(0)} = -0{,}02$; $\varepsilon_2^{(0)} = 0$; $\varepsilon_3^{(0)} = -0{,}01$. Получено уточнение $\delta_1^{(0)} = -0{,}0039$; $\delta_2^{(0)} = -0{,}0011$; $\delta_3^{(0)} = -0{,}0025$. Следовательно $x_1 = 4{,}6661$; $x_2 = 7{,}6189$; $x_3 = 9{,}0475$. Невязки будут $\delta_1 = -2 \cdot 10^{-4}$; $\delta_2 = -2 \cdot 10^{-4}$; $\delta_3 = 0$.

4. Модифицированный метод Гаусса

В данном случае помимо соблюдения требования $a_{kk} \neq 0$ при реализации формул (6) накладываются дополнительные требования, чтобы ведущий (главный) элемент в текущем столбце в процессе преобразований исходной матрицы имел максимальное по модулю значение. Это также достигается перестановкой строк матрицы.

Пример. В качестве иллюстрации преимущества модифицированного метода Гаусса, рассмотрим систему третьего порядка:

$$\begin{cases} 10x_1 - 7x_2 = 7; \\ -3x_1 + 2x_2 + 6x_3 = 4; \\ 5x_1 - x_2 + 5x_3 = 6. \end{cases} \qquad (a)$$

Прямой ход метода Гаусса

Исключаем x_1 из второго и третьего уравнений. Для этого первое уравнение умножаем на 0,3 и складываем со вторым, а затем умножаем первое уравнение на (–0,5) и складываем с третьим. В результате получаем

$$\begin{cases} 10x_1 - 7x_2 = 7; \\ -0{,}1x_2 + 6x_3 = 6{,}1; \\ 2{,}5x_2 + 5x_3 = 2{,}5. \end{cases} \qquad (б)$$

Замена второго уравнения третьим не производится, т.к. вычисления выполняются в рамках точной арифметики.

Умножая второе уравнение на 25, и складывая с третьим, получим

$$\begin{cases} 10x_1 - 7x_2 = 7; \\ -0{,}1x_2 + 6x_3 = 6{,}1; \\ 155x_3 = 155. \end{cases} \qquad (в)$$

Обратный ход метода Гаусса

Выполняем вычисления, начиная с последнего уравнения в полученной системе:

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

Подставляя полученное решение [0; –1; 1] в исходную систему, убеждаемся в его истинности.

Теперь изменим коэффициенты системы таким образом, чтобы сохранить прежнее решение, но при вычислении будем использовать округления в рамках арифметики с плавающей точкой сохраняя пять разрядов. Этому будет соответствовать следующая система

$$\begin{cases} 10x_1 - 7x_2 = 7; \\ -3x_1 + 2{,}099x_2 + 6x_3 = 3{,}901; \\ 5x_1 - x_2 + 5x_3 = 6. \end{cases} \qquad (г)$$

Прямой ход метода для системы (*г*) повторим по аналогичной технологии с исходной системой (*а*).

$$\begin{cases} 10x_1 - 7x_2 = 7; \\ -0{,}001x_2 + 6x_3 = 6{,}001; \\ 2{,}5x_2 + 5x_3 = 2{,}5. \end{cases} \qquad (д)$$

После исключения x_2 третье уравнение примет вид (остальные – без изменения)

$$15005\,x_3 = 15004. \qquad (е)$$

Выполняя обратный ход, получим

$$x_3 = \frac{15004}{15005} = 0{,}99993; \; x_2 = \frac{6 \cdot 0{,}99993 - 6{,}001}{0{,}001} = \frac{0{,}0014}{0{,}001} = -1{,}4; \; x_1 = \frac{7 \cdot (-1{,}5) + 7}{10} = -0{,}35.$$

Очевидно, что полученные решения [0; –1; 1] и [–0,35; –1,4; 0,99993] различны. Причиной этого является малая величина ведущего элемента во втором уравнении преобразования в (*д*). Чтобы это исключить, переставим в (*д*) вторую и третью строки

$$\begin{cases} 10x_1 - 7x_2 = 7; \\ -0{,}001x_2 + 6x_3 = 6{,}001; \\ 2{,}5x_2 + 5x_3 = 2{,}5; \end{cases} \equiv \begin{cases} 10x_1 - 7x_2 = 7; \\ 2{,}5x_2 + 5x_3 = 2{,}5; \\ -0{,}001x_2 + 6x_3 = 6{,}001. \end{cases} \qquad (ж)$$

Для данной системы после исключения x_2 из третьего уравнения, оно примет следующий вид

$$6{,}002\,x_3 = 6{,}002. \qquad (з)$$

В данном случае, выполняя обратный ход

$$x_3 = 1; \quad x_2 = \frac{2{,}5 - 5 \cdot 1}{2{,}5} = -1; \quad x_1 = \frac{7 + 7 \cdot (-1)}{10} = 0;$$

мы получим решение системы (*г*) [0; –1; 1], которое в точности совпадает с решением исходной системы.

Решая систему (*г*) мы использовали модифицированный метод Гаусса, в котором на диагонали должен был находиться максимальный в текущем столбце элемент.

Рассмотрим блок-схему модифицированного метода Гаусса (рис. 2.1).

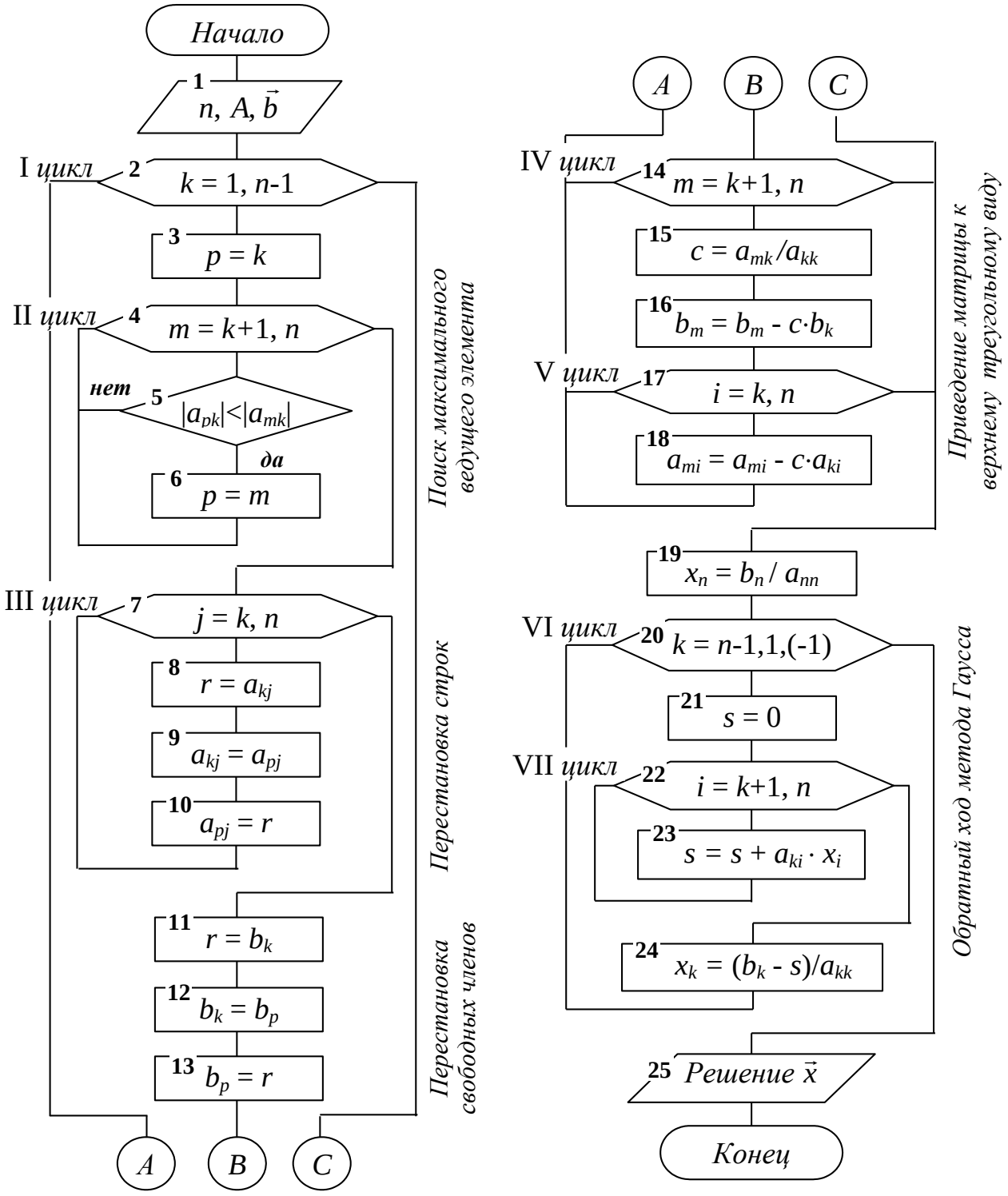


Рис. 2.1. Блок-схема модифицированного метода Гаусса

Проведем анализ предложенной схемы на примере системы n=3 (ε=0,001)

$$\begin{cases} 2x_1 + x_2 - x_3 = 1; \\ 4x_1 + 6x_2 + 2x_3 = 6; \\ 6x_1 + 5x_2 + 8x_3 = 14. \end{cases} \qquad (8)$$

$$A = \begin{bmatrix} 2 & 1 & -1 \\ 4 & 6 & 2 \\ 6 & 5 & 8 \end{bmatrix}; \qquad \bar{b} = \begin{bmatrix} 1 \\ 6 \\ 14 \end{bmatrix}. \qquad (*)$$

Блок 1. Ввод исходных данных: n – порядок системы, A – матрица коэффициентов при неизвестных, b – вектор свободных членов.

Блок 2. I-й цикл прямого хода (для k, изменяющегося от 1 до предпоследнего значения, т.е. до n–1) обеспечивает исключение из главной диагонали матрицы A элемента a_{kk}=0 благодаря поиску максимального элемента a_{kk} в текущем столбце, осуществляемому в блоках 3÷6 с помощью цикла II.

Далее с помощью цикла III в блоках 7÷13 выполняется перестановка текущей строки и строки с максимальным элементом в k-ом столбце (ее номер p).

Затем реализуются расчеты по формулам (6) прямого хода Гаусса в блоках циклов IV и V.

Проведем поблочный анализ в среде рассмотренных циклов I÷V на примере (8).

Блок 3 $p = k = 1$

Вход в цикл II

Блок 4 $m = k+1 = 2$ до $n = 3$

Блок 5 $a_{11} = 2 < a_{21} = 4$ из (*)

Блок 6 $p = 2$

Блок 4 $m = 2+1 = 3$

Блок 5 $a_{21} = 4 < a_{31} = 6$ из (*)

Блок 6 $p = 3$

Выход из цикла II и вход в цикл III, блоки 7÷10 выполняют перестановку строк матрицы A поэлементно

Блок 7 $j = 1$ (j от 1 до 3)

Блок 8 $r = a_{11} = 2$ из (*)

Блок 9 $a_{11} = a_{31} = 6$

Блок 10 $a_{31} = r$

Блок 7 $j = 2$

Блок 8 $r = a_{12} = 1$

Блок 9 $a_{12} = a_{32} = 5$

Блок 10 $a_{32} = r = 1$

Блок 7 $j = 3$ и по аналогии $r = a_{13}$; $a_{13} = a_{33}$; $a_{33} = r = -1$.

Выход из цикла III и вход в *Блок* 11 и далее 12÷13 выполняют аналогичную перестановку значений свободных членов

$r = b_1 = 1;\quad b_1 = b_3 = 14;\ \ b_3 = \mathrm{r} = 1.$

Вход в цикл IV с измененной системой

$$A = \begin{bmatrix} 6 & 5 & 8 \\ 4 & 6 & 2 \\ 2 & 1 & -1 \end{bmatrix}; \qquad \bar{b} = \begin{bmatrix} 14 \\ 6 \\ 1 \end{bmatrix}; \qquad (**)$$

для пересчета b_2 вектора $\vec{b}$

$m = k+1 = 1+1 = 2$ до $n = 3$

$c = a_{mk} / a_{kk} = a_{21} / a_{11} = 4/6$ из (**)

$b_2 = b_2 - c\, b_1 = 6 - 4/6 \cdot 14 = -20/6$ из (**)

Вход во вложенный цикл V для пересчета второй строки

$i = 1$ (i от 1 до 3); $a_{21} = a_{21} - c \cdot a_{11} = 4 - 4/6 \cdot 6 = 0;$

$i = 2;$ $a_{22} = a_{22} - c \cdot a_{12} = 6 - 4/6 \cdot 5 = 16/6;$

$i = 3;$ $a_{23} = a_{23} - c \cdot a_{13} = 2 - 4/6 \cdot 8 = -20/6.$

Выход из цикла V и вход в цикл IV

$m = 3;$ $c = a_{31} / a_{11} = 2/6.$

Вход в *Блок* 16

$b_3 = b_3 - c\, b_1 = 1 - 2/6 \cdot 14 = -22/6.$

Выход из цикла IV и вход в цикл V и вход в *Блок* 17

$i = 1$ (i от 1 до 3); $a_{31} = a_{31} - c \cdot a_{11} = 2 - 2/6 \cdot 6 = 0;$

$i = 2;$ $a_{32} = a_{32} - c \cdot a_{12} = 1 - 2/6 \cdot 5 = -4/6;$

$i = 3;$ $a_{33} = a_{33} - c \cdot a_{13} = -1 - 2/6 \cdot 8 = -22/6.$

Выход из цикла V с преобразованной системой

$$A = \begin{bmatrix} 6 & 5 & 8 \\ 0 & 16/6 & -20/6 \\ 0 & -4/61 & -22/6 \end{bmatrix}; \qquad \bar{b} = \begin{bmatrix} 14 \\ -20/6 \\ -22/6 \end{bmatrix}; \qquad (***)$$

и вход по линии *А* в цикл I

$k = 2;\ p = k = 2;\ m = k+1 = 3;$ вход в *Блок* 5

$| a_{22} | < | a_{32} | = | 16/6 | > | 4/6 |$ из (***).

Выход из цикла II и вход в цикл III

$j = 2$ (j от 2 до 3);

$r = a_{kj} = a_{22} = 16/6;\ a_{22} = a_{22}\ ;\ a_{22} = r = 16/6;$ из (***)

$j = 3;$

$r = a_{23} = -20/6;\ \ a_{23} = a_{23}\ ;\ \ a_{23} = r = -20/6;$ из (***)

В данном случае на диагонали оказался максимальный элемент, поэтому перестановка 2-ой и 3-ей строк не выполняется.

Выход из цикла III и вход в цикл I в *Блок* 11

$r = b_2\ ;$ $b_2 = b_2\ ;$ $b_2 = \mathrm{r} = -20/6.$

Свободный член b_2 остается на своем месте.

Вход в цикл IV

$m = k+1 = 2+1 = 3;$

$c = a_{mk} / a_{kk} = a_{32} / a_{22} = (-4/6) / (16/6);$ из (***)

$b_3 = b_3 - c\ b_2 = -22/6 - (-1/4) \cdot (-20/6) = -27/6$ из (***)

Выход из цикла IV и вход в цикл V

$i = 2$ (i от 2 до 3); $a_{32} = a_{32} - c \cdot a_{22} = -4/6 - (-1/4) \cdot 16/6 = 0;$

$i = 3;$ $a_{33} = a_{33} - c \cdot a_{23} = -22/6 - (-1/4) \cdot (-20/6) = -27/6.$

Выход из цикла V и выход из цикла I.

Обратный ход метода Гаусса

В *Блоках* 19÷24 реализуются формулы (7).

В *Блоке* 19 из последнего уравнения находится значение x_n ($n = 3$)

$x_3 = b_n / a_{nn} = b_3 / a_{33} = (-27/6) / (-27/6) = 1.$

Вход в цикл VI (*Блок* 20), в котором значение переменной цикла k изменяется от $n-1$ до 1 с шагом (–1)

Блок 21 s = 0

Вход в цикл VII (*Блок* 22)

$i = k+1 = 2+1 = 3;\ \ n = 3;\ \ s = s + a_{ki} \cdot x_i = 0 + a_{23} \cdot x_3 = -20/6 \cdot 1 = -20/6.$

Выход из цикла VII на *Блок* 24 в цикл VI:

$k = 2;$ $x_2 = (b_k - \mathrm{s})/\ a_{nn} = (b_2 - \mathrm{s})/\ a_{22} = (-20/6 + 20/6)/\ a_{22} = 0.$

Далее по аналогии

$k = k-1 = 2-1 = 1;$

s = 0;

$i = k + 1 = 2;$ $s = 0 + a_{12} \cdot x_2 = 5 \cdot 0 = 0;$

$i = k + 1 = 3;$ $s = 0 + a_{13} \cdot x_3 = 8 \cdot 1 = 8;$

$x_1 = (b_1 - \mathrm{s})/\ a_{11} = (14 - 8) / 6 = 1.$

Выход из последнего цикла VII.

В *Блоке* 25 (цикл опущен) выполняется вывод на экран полученного решения СЛАУ – вектора $\vec{x}$, т.е. x_i, i=1, ..., n. В нашем случае (1; 0; 1).

5. Метод прогонки

Данный метод также является модификацией метода Гаусса для частного случая *разреженных* систем – систем с матрицей трехдиагонального типа (краевая задача ДУ).

Каноническая форма их записи

$$a_i x_{i-1} + b_i x_i + c_i x_{i+1} = d_i; \quad i=\overline{1,n}; \quad a_1 = c_n = 0, \tag{9}$$

или в развернутом виде

$$\begin{cases} b_1 x_1 + c_1 x_2 = d_1; \\ a_2 x_1 + b_2 x_2 + c_2 x_3 = d_2; \\ a_3 x_2 + b_3 x_3 + c_3 x_4 = d_3; \\ \quad \ldots \\ a_{n-1} x_{n-2} + b_{n-1} x_{n-1} + c_{n-1} x_n = d_{n-1}; \\ a_n x_{n-1} + b_n x_n = d_n . \end{cases} \tag{10}$$

При этом, как правило, все коэффициенты $b_i \neq 0$.

Метод реализуется в два этапа – прямой и обратный ходы.

Прямой ход. Каждое неизвестное x_i выражается через x_{i+1}

$$x_i = A_i \cdot x_{i+1} + B_i \text{ для } i = 1,2, ..., n-1, \tag{11}$$

посредством прогоночных коэффициентов A_i и B_i. Определим алгоритм их вычисления.

Из первого уравнения системы (10) находим x_1

$$x_1 = -\frac{c_1}{b_1} x_2 + \frac{d_1}{b_1} .$$

Из уравнения (11) при i=1: $x_1 = A_1 \cdot x_2 + B_1$. Следовательно

$$A_1 = -\frac{c_1}{b_1}; \qquad B_1 = \frac{d_1}{b_1} . \tag{12}$$

Из второго уравнения системы (10) определяем x_2 через x_3, подставляя найденное значение x_1

$$a_2 (A_1 x_2 + B_1) + b_2 x_2 + c_2 x_3 = d_2 ,$$

откуда

$$x_2 = \frac{-c_2 x_3 + d_2 - a_2 B_1}{a_2 A_1 + b_2}; \tag{12*}$$

и согласно (11) при i = 2: $x_2 = A_2 \cdot x_3 + B_2$, следовательно

$$A_2 = -\frac{c_2}{e_2}; \qquad B_2 = \frac{d_2 - a_2 B_1}{e_2} , \text{ где } \quad e_2 = a_2 \cdot A_1 + b_2 .$$

Ориентируясь на соотношения индексов при коэффициентах (12) и (12*) можно получить эти соотношения для общего случая

$$A_i = -\frac{c_i}{e_i}; \qquad B_i = \frac{d_i - a_i B_{i-1}}{e_i}, \quad \text{где} \quad e_i = a_i \cdot A_{i-1} + b_i \ (i=2,3, ..., n-1). \qquad (13)$$

Обратный ход. Из последнего уравнения системы (10) с использованием (11) при $i = n-1$

$$x_n = \frac{d_n - a_n B_{n-1}}{b_n + a_n A_{n-1}}. \qquad (14)$$

Далее посредством (11) и прогоночных коэффициентов (12), (13) последовательно вычисляем $x_{n-1}, x_{n-2}, ..., x_1$.

При реализации метода прогонки нужно учитывать, что при условии

$$| b_i | \geq | a_i | + | c_i |, \qquad (15)$$

или хотя бы для одного b_i имеет место строгое неравенство (15), деление на «0» исключается и система имеет единственное решение.

Заметим, что условие (15) является достаточным, но не необходимым. В ряде случаев для хорошо обусловленных систем (10) метод прогонки может быть устойчивым и при несоблюдении условия (15).

Схема алгоритма метода прогонки может иметь вид, представленный на рис. 2.2.

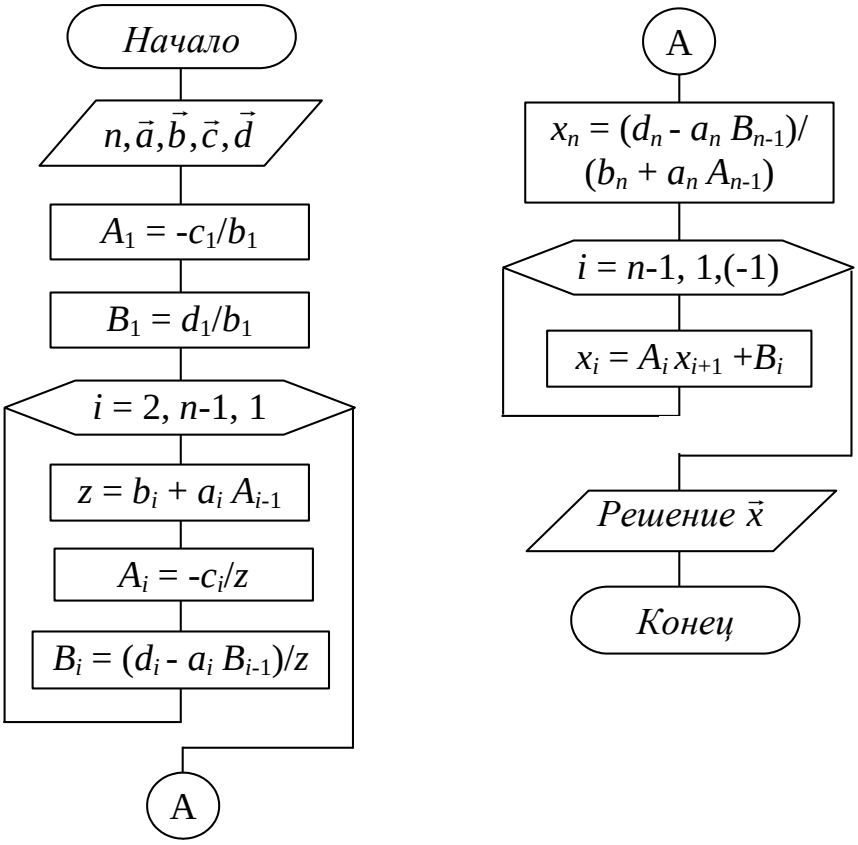


Рис. 2.2. Блок-схема метода прогонки

6. Метод квадратного корня

Данный метод используется для решения линейной системы

$$A\bar{x} = \bar{b}\,, \tag{16}$$

у которой матрица A симметрическая, т.е. $A^T = A$, $a_{ij} = a_{ji}$ (i=j=1,...,n).

Решение системы (16) осуществляется в два этапа.

Прямой ход. Преобразование матрицы A и представление ее в виде произведения двух взаимно транспонированных треугольных матриц:

$$A = S^T \cdot S, \tag{17}$$

где

$$S = \begin{vmatrix} s_{11} & s_{12} & \cdots & s_{1n} \\ 0 & s_{22} & \cdots & s_{2n} \\ & & \cdots & \\ 0 & 0 & \cdots & s_{nn} \end{vmatrix}; \qquad S^T = \begin{vmatrix} s_{11} & 0 & \cdots & 0 \\ s_{12} & s_{22} & \cdots & 0 \\ & & \cdots & \\ s_{1n} & s_{2n} & \cdots & s_{nn} \end{vmatrix}.$$

Перемножая S^T и S, и приравнивая матрице A, получим следующие формулы для определения s_{ij}:

$$\begin{cases} s_{11} = \sqrt{a_{11}}, \quad s_{1j} = a_{1j} / s_{11}, \quad (j > 1); \\ s_{ii} = \sqrt{a_{ii} - \sum\limits_{k=1}^{i-1} s_{ki}^2}, \qquad (1 \le i \le n); \\ s_{ij} = \dfrac{a_{ij} - \sum\limits_{k=1}^{i-1} s_{ki} s_{kj}}{s_{ii}}, \qquad (i < j); \\ s_{ij} = 0, \qquad (i > j). \end{cases} \tag{18}$$

После нахождения матрицы S систему (16) заменяем двумя ей эквивалентными системами с треугольными матрицами (17)

$$S^T \bar{y} = \bar{b}\,, \qquad S\bar{x} = \bar{y}\,. \tag{19}$$

Обратный ход. Записываем системы (19) в развернутом виде:

$$\begin{cases} s_{11} y_1 = b_1; \\ s_{12} y_1 + s_{22} y_2 = b_2\,; \\ \qquad \cdots \\ s_{1n} y_1 + s_{2n} y_2 + ... + s_{nn} y_n = b_n\,; \end{cases} \tag{20}$$

$$\begin{cases} s_{11} x_1 + s_{12} x_2 + ... + s_{1n} x_n = y_1; \\ \qquad s_{22} x_2 + \cdots + s_{2n} x_n = y_2; \\ \qquad\qquad \cdots \\ \qquad\qquad s_{nn} x_n = y_n\,. \end{cases} \tag{21}$$

Используя (20) и (21) последовательно находим

$$y_1 = \frac{b_1}{s_{11}}, \qquad y_i = (b_i - \sum_{k=1}^{i-1} s_{ki}\, y_k)/s_{ii}; \qquad (i > 1); \tag{22}$$

$$x_n = \frac{y_n}{s_{nn}}, \qquad x_i = (y_i - \sum_{k=i+1}^{n} s_{ik}\, x_k)/s_{ii}; \qquad (i < n). \tag{23}$$

Метод квадратных корней дает большой выигрыш во времени по сравнению с рассмотренными ранее прямыми методами, так как, во-первых, существенно уменьшает число умножений и делений (почти в два раза), во-вторых, позволяет накапливать сумму произведений без записи промежуточных результатов. Числовой пример ручного счета можно посмотреть в учебнике: *Копченова Н.В., Марен И.А. «Вычислительная математика в примерах и задачах», 1972.*

Машинная реализация метода предусматривает его следующую трактовку. Исходная матрица A системы (16) представляется в виде произведения трех матриц

$$A = S^T \cdot D \cdot S,$$

где D – диагональная матрица с элементами $d_{ii} = \pm 1$; S – верхняя треугольная ($s_{ik} = 0$, если $i > k$, причем $s_{ii} > 0$); S^T – транспонированная нижняя треугольная.

Требование выполнения условия $s_{ii} > 0$ необходимо для полной определенности разложения. Это и определяет необходимость введения диагональной матрицы D.

Рассмотрим алгоритм разложения матрицы A с использованием матрицы D на примере матрицы второго порядка.

Пусть A – действительная симметричная матрица

$$A = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix}.$$

Будем искать S и D в виде

$$S = \begin{bmatrix} s_{11} & s_{12} \\ 0 & s_{22} \end{bmatrix}, \qquad D = \begin{bmatrix} d_{11} & 0 \\ 0 & d_{22} \end{bmatrix}, \qquad \text{где } d_{ii} = \pm 1.$$

Тогда

$$S^T D S = \begin{bmatrix} s_{11}^2 d_{11} & s_{11} s_{12} d_{11} \\ s_{11} s_{12} d_{11} & s_{12}^2 d_{11} + s_{22}^2 d_{22} \end{bmatrix}.$$

Из условия равенства $A = S^T D S$, получим три уравнения

$$s_{11}^2 d_{11} = a_{11}; \quad s_{11} s_{12} d_{11} = a_{12}; \quad s_{12}^2 d_{11} + s_{22}^2 d_{22} = a_{22}.$$

Из первого уравнения находим

$$d_{11} = \operatorname{sign} a_{11}; \qquad s_{11} = \sqrt{|a_{11}|}.$$

Далее, если $a_{11} \neq 0$, то $s_{12} = a_{12} / (s_{11}\, d_{11})$, и, наконец

$$s_{22}^2 d_{22} = a_{22} - s_{12}^2 d_{11},$$

т.е. $d_{22} = \text{sign}\,(a_{22} - s_{12}^2 d_{11})$; $\quad s_{22} = \sqrt{|a_{22} - s_{12}^2\, d_{11}|}$.

Здесь и для общего случая матрицу S можно по аналогии с числами трактовать как корень квадратный из матрицы A, отсюда и название метода.

Итак, если $S^T D S$ известно, то решение исходной системы $A \cdot \vec{x} = \vec{b}$ сводится к последовательному решению систем:

$$S^T D \cdot \vec{y} = \vec{b}; \quad S \cdot \vec{x} = \vec{y}. \tag{23}$$

Нахождение элементов матрицы S (извлечение корня из A) осуществляется по рекуррентным формулам, избежав проблемы оперирования комплексными числами:

$$d_k = \text{sign}\left(a_{kk} - \sum_{i=1}^{k-1} d_i\, s_{ik}^2 \right);$$

$$s_{kk} = \sqrt{\left| a_{kk} - \sum_{i=1}^{k-1} d_i\, s_{ik}^2 \right|}; \tag{24}$$

$$s_{kj} = \left(a_{kj} - \sum_{i=1}^{k-1} d_i\, s_{ik}\, s_{ij} \right) / (s_{kk}\, d_k);$$

$$k = 1, 2, ..., n; \qquad j = k+1, k+2, ..., n.$$

В этих формулах сначала полагаем $k = 1$ и последовательно вычисляем

$$d_1 = \text{sign}\,(a_{11}); \qquad s_{11} = \sqrt{|a_{11}|}$$

и все элементы первой строки матрицы S (s_{1j}, $j>1$), затем полагаем $k = 2$, вычисляем s_{22} и вторую строку матрицы s_{1j} для $j>2$ и т.д.

Решение систем (23) ввиду треугольности матрицы S осуществляется по формулам, аналогичным обратному ходу метода Гаусса:

$$y_1 = \frac{b_1}{s_{11} d_1}, \qquad y_i = (b_i - \sum_{k=1}^{i-1} d_k\, s_{ki}\, y_k) / (s_{ii}\, d_i); \quad i = 2,3,..., n;$$

$$x_n = \frac{y_n}{s_{nn}}, \qquad x_i = (y_i - \sum_{k=i+1}^{n} s_{ik}\, x_k) / s_{ii}; \quad i = n-1, n-2, ..., 1.$$

Метод квадратного корня почти вдвое эффективнее метода Гаусса, т.к. полезно использует симметричность матрицы.

Схема алгоритма метода квадратного корня представлена на рис. 2.3. Значение функции sign(x) равно +1 для всех $x > 0$ и –1 для всех $x < 0$.

Алгоритм реализован в методическом пособии: А.К.Синицын и др. «Алгоритмы вычислительной математики».

Проиллюстрируем метод квадратного корня, решая систему трех уравнений:

$$\begin{cases} x_1 + x_2 + x_3 = 3; \\ x_1 + 2x_2 + 2x_3 = 5; \\ x_1 + 2x_2 + 3x_3 = 6; \end{cases} \qquad A = \begin{bmatrix} 1 & 1 & 1 \\ 1 & 2 & 2 \\ 1 & 2 & 3 \end{bmatrix}, \qquad \vec{b} = \begin{bmatrix} 3 \\ 5 \\ 6 \end{bmatrix}.$$

Нетрудно проверить, что матрица A есть произведение двух треугольных матриц (здесь $d_{ii} = 1$):

$$A = \begin{bmatrix} 1 & 1 & 1 \\ 1 & 2 & 2 \\ 1 & 2 & 3 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 1 & 1 & 0 \\ 1 & 1 & 1 \end{bmatrix} \times \begin{bmatrix} 1 & 1 & 1 \\ 0 & 1 & 1 \\ 0 & 0 & 1 \end{bmatrix} = S^T \cdot S \ .$$

Исходную систему запишем в виде

$$S^T \cdot S \cdot \vec{x} = \begin{bmatrix} 1 & 0 & 0 \\ 1 & 1 & 0 \\ 1 & 1 & 1 \end{bmatrix} \times \begin{bmatrix} 1 & 1 & 1 \\ 0 & 1 & 1 \\ 0 & 0 & 1 \end{bmatrix} \times \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} 3 \\ 5 \\ 6 \end{bmatrix} .$$

Обозначим

$$S \cdot \vec{x} = \begin{bmatrix} 1 & 1 & 1 \\ 0 & 1 & 1 \\ 0 & 0 & 1 \end{bmatrix} \times \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} y_1 \\ y_2 \\ y_3 \end{bmatrix} .$$

Тогда для вектора $\vec{y}$ получим систему $S^T \vec{y} = \vec{b}$:

$$\begin{bmatrix} 1 & 0 & 0 \\ 1 & 1 & 0 \\ 1 & 1 & 1 \end{bmatrix} \times \begin{bmatrix} y_1 \\ y_2 \\ y_3 \end{bmatrix} = \begin{bmatrix} 3 \\ 5 \\ 6 \end{bmatrix} , \qquad \text{откуда} \qquad y_1 = 3; \ \ y_2 = 2; \ \ y_3 = 1.$$

Зная $\vec{y}$, решаем систему $S \vec{x} = \vec{y}$:

$$\begin{bmatrix} 1 & 1 & 1 \\ 0 & 1 & 1 \\ 0 & 0 & 1 \end{bmatrix} \times \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} 3 \\ 2 \\ 1 \end{bmatrix} , \qquad \text{откуда} \qquad x_3 = 1; \ \ x_2 = 1; \ \ x_1 = 1.$$

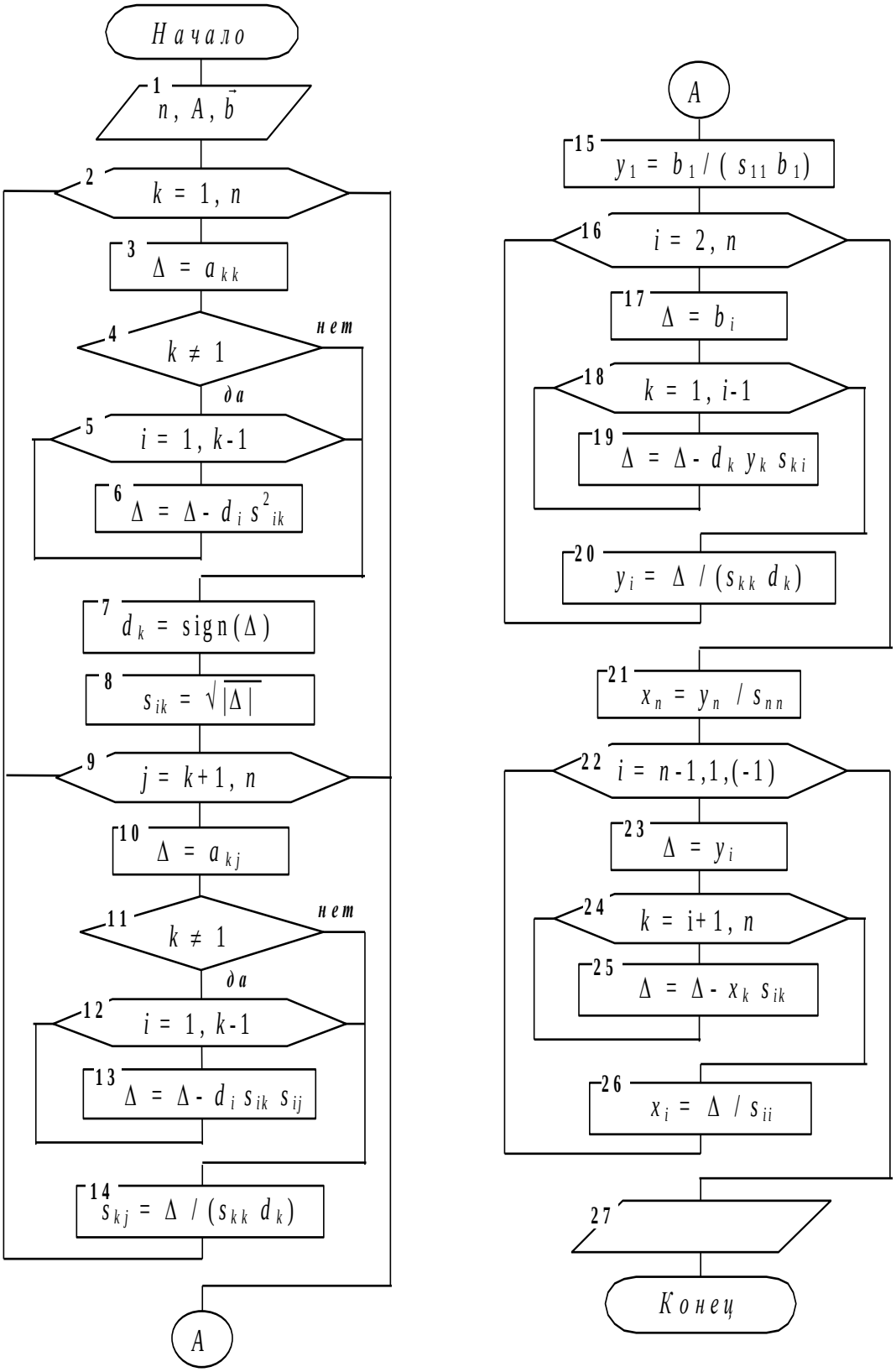


Рис. 2.3. Блок-схема метода квадратного корня

2.2.2. Итерационные методы решения СЛАУ

Напомним, что достоинством итерационных методов является их применимость к плохо обусловленным системам и системам высоких порядков, их самоисправляемость и простота реализации на ЭВМ. Итерационные методы для начала вычисления требуют задания какого-либо начального приближения к искомому решению.

Следует заметить, что условия и скорость сходимости итерационного процесса существенно зависят от свойств матрицы A системы и от выбора начальных приближений.

Для применения метода итераций исходную систему (1) или (2) необходимо привести к виду

$$\bar{x} = G\bar{x} + \bar{f} \tag{25}$$

и затем итерационный процесс выполняется по рекуррентным формулам

$$\bar{x}^{(k+1)} = G\bar{x}^{(k)} + \bar{f}\ ,\ k = 0, 1, 2, \ldots . \tag{25*}$$

Матрица G и вектор $\bar{f}$ получены в результате преобразования системы (1).

Для сходимости (25*) необходимо и достаточно, чтобы $|\lambda_i(G)| < 1$, где $\lambda_i(G)$ – все собственные значения матрицы G. Сходимость будет также и в случае, если $\|G\| < 1$, ибо $|\lambda_i(G)| < \forall\ \|G\|$ ($\forall$ – любой).

Символ $\|\ \ldots\ \|$ означает норму матрицы. При определении ее величины чаще всего останавливаются на проверке двух условий:

$$\|G\| = \max_{1\le i\le n} \sum_{j=1}^{n} |g_{ij}|, \quad \text{или} \quad \|G\| = \max_{1\le j\le n} \sum_{i=1}^{n} |g_{ij}|, \tag{26}$$

где $G = \{g_{ij}\}_1^n$. Сходимость гарантирована также, если исходная матрица A имеет диагональное преобладание, т.е.

$$|a_{ii}| > \sum_{i,\,j=1;\,i\ne j}^{n} |a_{ij}|, \qquad A = \{a_{ij}\}_1^n . \tag{27}$$

Если (26) или (27) выполняются, метод итерации сходится при любом начальном приближении $\bar{x}^{(0)}$. Чаще всего вектор $\bar{x}^{(0)}$ берут или нулевым, или единичным, или сам вектор $\bar{f}$ из (25).

Имеется много подходов к преобразованию исходной системы (2) с матрицей A для обеспечения вида (25) или условий сходимости (26) и (27).

Например, (25) можно получить следующим образом.

Пусть $A = B + C$, $\det B \ne 0$;

тогда $(B+C)\ \bar{x} = \bar{b} \quad \Rightarrow \quad B\ \bar{x} = -C\bar{x} + \bar{b} \quad \Rightarrow \quad B^{-1}B\bar{x} = -B^{-1}\ C\bar{x} + B^{-1}\ \bar{b}$,

откуда $\bar{x} = -B^{-1}\ C\bar{x} + B^{-1}\ \bar{b}$.

Положив $-B^{-1}\ C = G$, $B^{-1}\ \bar{b} = \bar{f}$ и получим (25).

Из условий сходимости (26) и (27) видно, что представление $A = B + C$ не может быть произвольным.

Если матрица A удовлетворяет требованиям (27), то в качестве матрицы B можно выбрать нижнюю треугольную

$$B = \begin{bmatrix} a_{11} & 0 & \cdots & 0 \\ a_{21} & a_{22} & \cdots & 0 \\ & \cdots & & \\ a_{n1} & a_{n2} & \cdots & a_{nn} \end{bmatrix}, \quad a_{ii} \neq 0.$$

Или

$$A\bar{x} = \bar{b}; \quad \Rightarrow \quad A\bar{x} - \bar{b} = 0; \quad \Rightarrow \quad \bar{x} + (A\bar{x} - \bar{b}) = \bar{x}; \quad \Rightarrow$$

$$\bar{x} = \bar{x} + \alpha(A\bar{x} - \bar{b}) = \bar{x} + \alpha A\bar{x} - \alpha\bar{b} = (E + \alpha A)\bar{x} - \alpha\bar{b} = G\bar{x} + \bar{f} \ .$$

Подбирая параметр α можно добиться, чтобы $\|G\| = \|E + \alpha A\| < 1$.

Если имеет место преобладание (27), тогда преобразование к (25) можно осуществить просто, решая каждое i-е уравнение системы (1) относительно x_i по следующим рекуррентным формулам:

$$x_i^k = -\frac{1}{a_{ii}}\left[\sum_{j=1;\, j\neq i}^{n} a_{ij}\, x_j^{k-1} - b_i\right] = \sum_{j=1}^{n} g_{ij}\, x_j^{k-1} + f_i;$$

$$g_{ij} = -a_{ij} / a_{ii}; \qquad g_{ii} = 0; \qquad f_i = b_i / a_{ii}, \qquad (27^*)$$

т.е. $G = \{g_{ij}\}_1^n$.

Если же в матрице A нет диагонального преобладания, его нужно добиться посредством каких-либо ее линейных преобразований, не нарушающих их равносильности.

Для примера рассмотрим систему

$$\begin{cases} 2x_1 - 1{,}8x_2 + 0{,}4x_3 = 1; & (I) \\ 3x_1 + 2x_2 - 1{,}1x_3 = 0; & (II) \\ x_1 - x_2 + 7{,}3x_3 = 0; & (III) \end{cases} \qquad (28)$$

Как видно в уравнениях (*I*) и (*II*) нет диагонального преобладания, а в (*III*) есть, поэтому его оставляем неизменным.

Добьемся диагонального преобладания в уравнении (*I*). Умножим (*I*) на α, (*II*) на β, сложим оба уравнения и в полученном уравнении выберем α и β так, чтобы имело место диагональное преобладание:

$$(2\alpha + 3\beta)\, x_1 + (-1{,}8\alpha + 2\beta)\, x_2 + (0{,}4\alpha - 1{,}1\beta)x_3 = \alpha \ .$$

Взяв $\alpha = \beta = 5$, получим $25x_1 + x_2 - 3{,}5x_3 = 5$.

Для преобразования второго уравнения (*II*) с преобладанием, (*I*) умножим на γ, (*II*) умножим на δ, и из (*II*) вычтем (*I*). Получим

$$(3\delta - 2\gamma)\, x_1 + (2\delta + 1{,}8\gamma)\, x_2 + (-1{,}1\delta - 0{,}4\gamma)x_3 = -\gamma \ .$$

Положим δ = 2, γ = 3, получим $0x_1 + 9{,}4x_2 - 3{,}4x_3 = -3$. В результате получим систему:

$$\begin{cases} 25x_1 + x_2 - 3{,}5x_3 = 5; \\ 9{,}4x_2 - 3{,}4x_3 = -3; \\ x_1 - x_2 + 7{,}3x_3 = 0. \end{cases} \qquad (29)$$

Такой прием можно применять для широкого класса матриц.

Далее разделим в (29) каждое уравнение на диагональный элемент, получим

$$\begin{cases} x_1 + 0{,}04x_2 - 0{,}14x_3 = 0{,}2; \\ x_2 - 0{,}36x_3 = -0{,}32; \\ 0{,}14x_1 - 0{,}14x_2 + x_3 = 0. \end{cases} \quad \text{или} \quad \begin{cases} x_1 = -0{,}04x_2 + 0{,}14x_3 + 0{,}2; \\ x_2 = 0{,}36x_3 - 0{,}32; \\ x_3 = -0{,}14x_1 + 0{,}14x_2. \end{cases}$$

Взяв в качестве начального приближения, например, вектор $\bar{x}^{(0)} = (0{,}2; -0{,}32; 0)^T$. Будем решать эту систему по технологии (25*):

$$\begin{aligned} x_1^{(k+1)} &= -0{,}04x_2^{(k)} + 0{,}14x_3^{(k)} + 0{,}2; \\ x_2^{(k+1)} &= 0{,}36x_3^{(k)} - 0{,}32; \qquad k = 0, 1, 2, \ldots \\ x_3^{(k+1)} &= -0{,}14x_1^{(k)} + 0{,}14x_2^{(k)}. \end{aligned}$$

Процесс вычисления прекращается, когда два соседних приближения вектора решения совпадают по точности, т.е.

$$\left| \bar{x}^{(k+1)} - \bar{x}^{(k)} \right| < \varepsilon.$$

1. Метод простой итерации

Технология итерационного решения вида (25*) названа методом ***простой итерации***.

Оценка абсолютной погрешности для метода простой итерации

$$\left\| \bar{x}^* - \bar{x}^{(k+1)} \right\| \leq \|G\|^{k+1} \cdot \left\| \bar{x}^{(0)} \right\| + \frac{\|G\|^{k+1}}{1 - \|G\|} \cdot \left\| \bar{f} \right\|,$$

напомним, символ || ... || означает норму.

***Пример* 1**. Методом простой итерации с точностью ε=0,001 решить систему линейных уравнений

$$\begin{cases} x_1 = 0{,}32x_1 - 0{,}05x_2 + 0{,}11x_3 - 0{,}08x_4 + 2{,}15; \\ x_2 = 0{,}11x_1 + 0{,}16x_2 - 0{,}28x_3 - 0{,}06x_4 - 0{,}83; \\ x_3 = 0{,}08x_1 - 0{,}15x_2 + 0{,}12x_4 + 1{,}16; \\ x_4 = -0{,}21x_1 + 0{,}13x_2 - 0{,}27x_3 + 0{,}44. \end{cases}$$

Число шагов, дающих ответ с точностью до ε = 0,001, можно определить из соотношения

$$\left\| \bar{x}^* - \bar{x}^{(k)} \right\| \quad \le \quad \frac{\|G\|^{k+1}}{1-\|G\|} \cdot \left\| \bar{f} \right\| \le 0{,}001.$$

Оценим сходимость по формуле (26). Здесь $\|G\| = \max\limits_{1\le i\le 4} \sum\limits_{j=1}^{4} |g_{ij}| = \max\{0{,}56;\ 0{,}61;\ 0{,}35;\ 0{,}61\} = 0{,}61 < 1$; $\| \bar{f} \| = 2{,}15$. Значит, сходимость обеспечена.

В качестве начального приближения возьмем вектор свободных членов, т.е. $\bar{x}^{(0)} = (2{,}15;\ -0{,}83;\ 1{,}16;\ 0{,}44)^{\mathrm{T}}$. Подставим значения вектора $\bar{x}^{(0)}$ в формулы (25*):

$$x_1^{(1)} = 0{,}32\cdot 2{,}15 + 0{,}05\cdot 0{,}83 + 0{,}11\cdot 1{,}16 - 0{,}08\cdot 0{,}44 + 2{,}15 = 2{,}9719;$$

$$x_2^{(1)} = 0{,}11\cdot 2{,}15 - 0{,}16\cdot 0{,}83 - 0{,}28\cdot 1{,}16 - 0{,}06\cdot 0{,}44 - 0{,}83 = -1{,}0775;$$

$$x_3^{(1)} = 0{,}08\cdot 2{,}15 + 0{,}15\cdot 0{,}83 + 0{,}12\cdot 0{,}44 + 1{,}16 = 1{,}5093;$$

$$x_4^{(1)} = -0{,}21\cdot 2{,}15 - 0{,}13\cdot 0{,}83 - 0{,}27\cdot 1{,}16 + 0{,}44 = -0{,}4326.$$

Продолжая вычисления, результаты занесем в таблицу:

k	x_1	x_2	x_3	x_4
0	2,15	–0,83	1,16	0,44
1	2,9719	–1,0775	1,5093	–0,4326
2	3,3555	–1,0721	1,5075	–0,7317
3	3,5017	–1,0106	1,5015	–0,8111
4	3,5511	–0,9277	1,4944	–0,8321
5	3,5637	–0,9563	1,4834	–0,8298
6	3,5678	–0,9566	1,4890	–0,8332
7	3,5760	–0,9575	1,4889	–0,8356
8	3,5709	–0,9573	1,4890	–0,8362
9	3,5712	–0,9571	1,4889	–0,8364
10	3,5713	–0,9570	1,4890	–0,8364

Сходимость в тысячных долях имеет место уже на 10-м шаге.

Ответ: $x_1 \approx 3{,}571$; $x_2 \approx -0{,}957$; $x_3 \approx 1{,}489$; $x_4 \approx -0{,}836$.

Это решение может быть получено и с помощью формул (27*).

***Пример* 2**. Для иллюстрации алгоритма с помощью формул (27*), рассмотрим решение системы (только две итерации):

$$\begin{cases} 4x_1 - x_2 - x_3 = 2; \\ x_1 + 5x_2 - 2x_3 = 4; \\ x_1 + x_2 + 4x_3 = 6; \end{cases} \qquad A = \begin{bmatrix} 4 & -1 & -1 \\ 1 & 5 & -2 \\ 1 & 1 & 4 \end{bmatrix}; \qquad \vec{b} = \begin{bmatrix} 2 \\ 4 \\ 6 \end{bmatrix}. \tag{30}$$

Преобразуем систему к виду (25) согласно (27*):

$$\begin{cases} x_1 = (2 + x_2 + x_3)/4; \\ x_2 = (4 - x_1 + 2x_3)/5; \\ x_3 = (6 - x_1 - x_2)/4; \end{cases} \quad \Rightarrow \quad \begin{aligned} x_1^{(k+1)} &= (2 + x_2^{(k)} + x_3^{(k)})/4; \\ x_2^{(k+1)} &= (4 - x_1^{(k)} + 2x_3^{(k)})/5; \\ x_3^{(k+1)} &= (6 - x_1^{(k)} - x_2^{(k)})/4. \end{aligned} \tag{31}$$

Возьмем начальное приближение $\bar{x}^{(0)} = (0; 0; 0)^T$. Тогда для $k = 0$ очевидно, что значение $\bar{x}^{(1)} = (0{,}5; 0{,}8; 1{,}5)^T$. Подставим эти значения в (31), т.е. при k=1, получим $\bar{x}^{(2)} = (1{,}075; 1{,}3; 1{,}175)^T$.

Ошибка $\varepsilon_2 = \max\limits_{1 \le i \le 2} \left| x_i^{(2)} - x_i^{(1)} \right| = \max(0{,}575; 0{,}5; 0{,}325) = 0{,}575.$

Блок-схема алгоритма нахождения решения СЛАУ по методу простых итераций согласно рабочим формулам (27*) представлена на рис. 2.4.

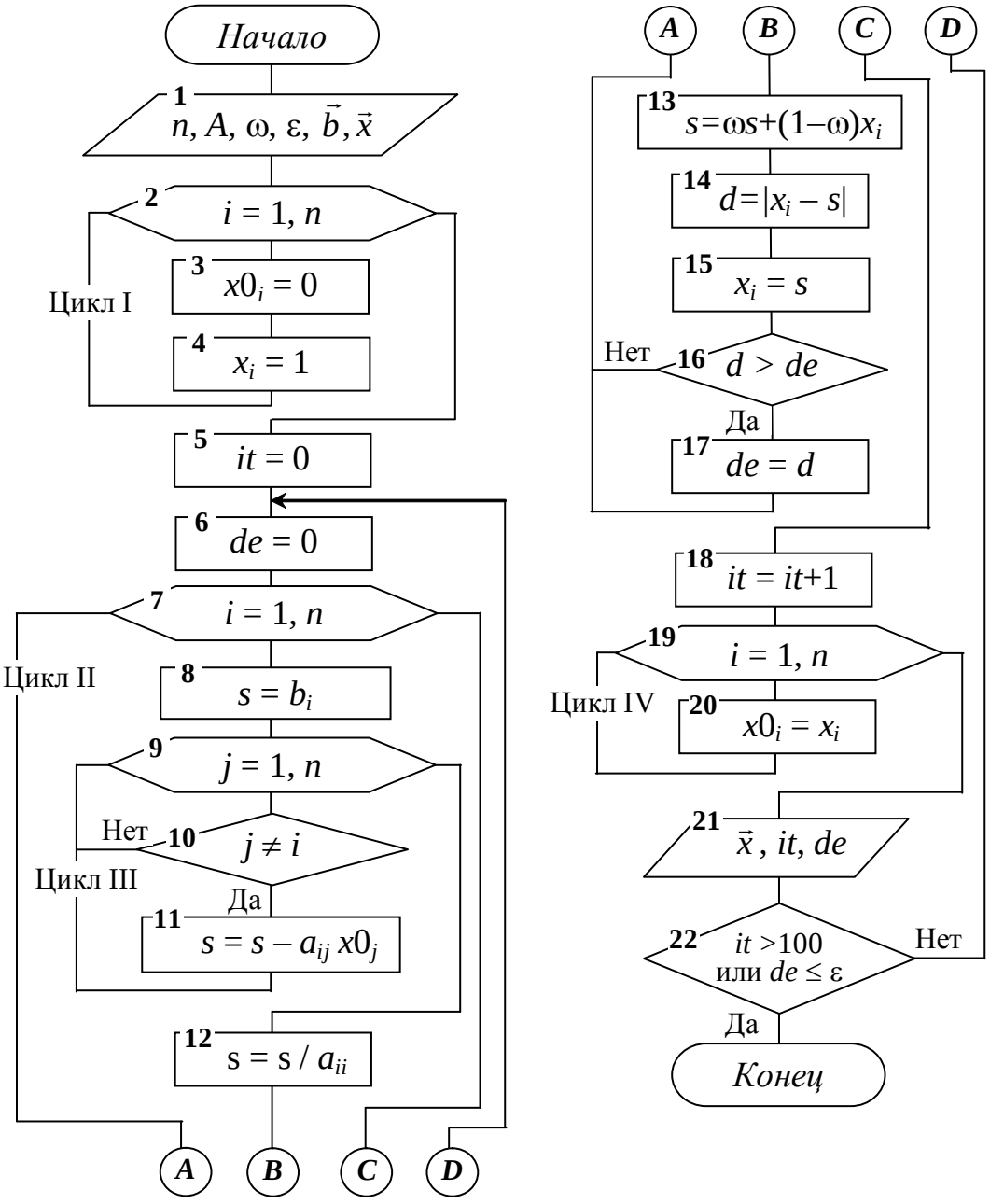


Рис. 2.4. Блок-схема метода простых итераций для решения СЛАУ

Особенностью блок-схемы являются блоки:
(13) – назначение его рассмотрим ниже;
(21) – вывод результатов на экран;

(22) – проверка (индикатор) сходимости.

Проведем анализ предложенной схемы на примере системы (30) (n = 3, ω =1, ε = 0,001):

$$\begin{cases} 4x_1 - x_2 - x_3 = 2; \\ x_1 + 5x_2 - 2x_3 = 4; \\ x_1 + x_2 + 4x_3 = 6; \end{cases} \qquad A = \begin{bmatrix} 4 & -1 & -1 \\ 1 & 5 & -2 \\ 1 & 1 & 4 \end{bmatrix} = \{a_{ij}\}_{i=\overline{1,3}}; \qquad \vec{b} = \begin{bmatrix} b_1 = 2 \\ b_2 = 4 \\ b_3 = 6 \end{bmatrix}.$$

Блок 1. Вводим исходные данные A, $\vec{b}$, $\vec{x}$, ω, ε, n: n = 3, ω =1, ε = 0,001.

Цикл I. Задаем начальные значения векторов $x0_i$ и x_i (i = 1,2,3).

Блок 5. Обнуляем счетчик числа итераций.

Блок 6. Обнуляем счетчик текущей погрешности.

Цикл II – счетчик номеров матрицы A и вектора $\vec{b}$:

i = 1: $S = b_1 = 2$ (*блок* 8).

Переходим во вложенный *Цикл III*, *блок* 9 – счетчик номеров столбцов матрицы A: j = 1.

Блок 10: $j = i$, следовательно, возвращаемся к *блоку* 9 и увеличиваем j на единицу: j = 2.

В *блоке* 10 $j \neq i$ (2 ≠ 1) – выполняем переход к *блоку* 11.

Блок 11: $S = 2 - (-1) \cdot x0_2 = 2 - (-1) \cdot 0 = 2$, и переходим к *блоку* 9, в котором j увеличиваем на единицу: j = 3.

В *блоке* 10 условие $j \neq i$ выполняется, поэтому переходим к *блоку* 11.

Блок 11: $S = 2 - (-1) \cdot x0_3 = 2 - (-1) \cdot 0 = 2$, после чего переходим к *блоку* 9, в котором j увеличиваем на единицу (j = 4). Значение j больше n (n = 3) – заканчиваем цикл и переходим к блоку 12.

Блок 12: $S = S / a_{11} = 2 / 4 = 0{,}5$.

Блок 13: ω = 1; $S = S + 0 = 0{,}5$.

Блок 14: $d = | x_i - S | = | 1 - 0{,}5 | = 0{,}5$.

Блок 15: $x_i = 0{,}5$ (i = 1).

Блок 16: Проверяем условие $d > de$: 0,5 > 0, следовательно, переходим к блоку 17, в котором присваиваем de = 0,5 и выполняем возврат по ссылке «*A*» к следующему шагу *цикла II* – к *блоку* 7, в котором i увеличиваем на единицу:

i = 2: $S = b_2 = 4$ (*блок* 8).

Переходим во вложенный *Цикл III*, *блок* 9: j = 1.

Посредством *блока* 10 $j \neq i$ (1 ≠ 2) – выполняем переход к *блоку* 11.

Блок 11: $S = 4 - 1 \cdot 0 = 4$, и переходим к *блоку* 9, в котором j увеличиваем на единицу: j = 2.

В *блоке* 10 условие не выполняется, поэтому переходим к *блоку* 9, в котором j увеличиваем на единицу: j = 3. По аналогии переходим к *блоку* 11.

Блок 11: $S = 4 - (-2) \cdot 0 = 4$, после чего заканчиваем *цикл III* и переходим к блоку 12.

Блок 12: $S = S / a_{22} = 4 / 5 = 0{,}8$.

Блок 13: $\omega = 1$; $S = S + 0 = 0{,}8$.

Блок 14: $d = |1 - 0{,}8| = 0{,}2$.

Блок 15: $x_i = 0{,}8$ ($i = 2$).

Блок 16: Проверяем условие $d > de$: $0{,}2 < 0{,}5$; следовательно, переходим возврат по ссылке «*A*» к следующему шагу *цикла II* – к *блоку* 7:

$i = 3$: $S = b_3 = 6$ (*блок* 8).

Переходим во вложенный *Цикл III*, *блок* 9: $j = 1$.

Посредством *блока* 10 выполняем переход к *блоку* 11.

Блок 11: $S = 6 - 1 \cdot 0 = 6$, и переходим к *блоку* 9: $j = 2$.

Посредством *блока* 10 выполняем переход к *блоку* 11.

Блок 11: $S = 6 - 1 \cdot 0 = 6$. Заканчиваем *цикл III* и переходим к блоку 12.

Блок 12: $S = S / a_{33} = 6 / 4 = 1{,}5$.

Блок 13: $S = 1{,}5$.

Блок 14: $d = |1 - 1{,}5| = 0{,}5$.

Блок 15: $x_i = 1{,}5$ ($i = 3$).

Согласно *блоку* 16 (с учетом ссылок «*A*» и «*C*») выходим из *цикла II* и переходим к *блоку* 18:

Блок 18. Увеличиваем число итераций $it = it + 1 = 0 + 1 = 1$.

В *блоках* 19 и 20 *цикла IV* заменяем начальные значения $x0_i$ полученными значениями x_i ($i = 1,2,3$).

Блок 21. Выполняем печать промежуточных значений текущей итерации, в нашем случае: $\vec{x} = (0{,}5;\ 0{,}8;\ 1{,}5)^T$, $it = 1$; $de = 0{,}5$.

Посредством *блока* 22 по ссылке «*D*» переходим к *блоку* 6 и $de = 0$.

Переходим к *циклу II* на *блок* 7 и выполняем рассмотренные вычисления с новыми начальными значениями $x0_i$ ($i = 1,2,3$).

После чего получим: $x_1 = 1{,}075$; $x_2 = 1{,}3$; $x_3 = 1{,}175$.

Блок 18. Увеличиваем число итераций $it = it + 1 = 1 + 1 = 2$.

В *блоках* 19 и 20 *цикла IV* заменяем начальные значения $x0_i$ полученными x_i ($i = 1,2,3$).

Блок 21. Выполняем печать значений второй итерации: $\vec{x} = (1{,}075;\ 1{,}3;\ 1{,}175)^T$, $it = 2$; $de = 0{,}575$; и т.д.

2. Метод Зейделя

Данный метод является модификацией метода простой итерации и для системы (25) $\bar{x} = G\bar{x} + \bar{f}$ имеет следующую технологию

$$
\begin{aligned}
x_1^{(k+1)} &= g_{11}x_1^k + \ldots + g_{1n}x_n^k + f_1;\\
x_2^{(k+1)} &= g_{21}x_1^{(k+1)} + \ldots + g_{2n}x_n^{(k)} + f_2;\\
x_3^{(k+1)} &= g_{31}x_1^{(k+1)} + \ldots + g_{31}x_n^{(k)} + f_3;\\
&\ldots\\
x_n^{(k+1)} &= g_{n1}x_1^{(k+1)} + \ldots + g_{nn}x_n^{(k)} + f_n.
\end{aligned}
\qquad (32)
$$

Суть его состоит в том, что при вычислении очередного приближения $x_i^{(k)}$ $(2 \le i \le n)$ в системе (32) и в формуле (27*), если имеет место соотношение (27), вместо $x_i^{k-1}, \ldots, x_{i-1}^{k-1}$ используются уже вычисленные ранее $x_1^k, \ldots, x_{i-1}^k$, т.е. (27*) преобразуется к виду

$$x_i^k = \sum_{j=1}^{i-1} g_{ij} x_j^k + \sum_{j=i+1}^{\bar{n}} g_i x_j^{k-1} + f_i \ , i = 1, \ldots, n. \qquad (33)$$

Это позволяет ускорить сходимость итераций почти в два раза. Оценка точности аналогична методу простой итерации. Схема алгоритма аналогична схеме метода простой итерации, если $x0_j$ заменить на x_j и убрать строки $x0_i = 1$, $x0_i = x_i$.

Пример. Методом Зейделя решить систему линейных уравнений с точностью ε = 0,0001, приводя ее к виду, удобному для итераций.

$$\begin{cases} 4.5x_1 - 1.8x_2 + 3.6x_3 = -1.7 & (I) \\ 3.1x_1 + 2.3x_2 - 1.2x_3 = 3.6 & (II) \\ 1.8x_1 + 2.5x_2 + 4.6x_3 = 2.2 & (III) \end{cases} \qquad (34)$$

Условия (27) для системы не удовлетворяются, поэтому приведем ее к виду соответствующему данному требованию.

$$\begin{cases} 7.6x_1 + 0.5x_2 + 2.4x_3 = 1.9, & (I + II) \\ 2.2x_1 + 9.1x_2 + 4.4x_3 = 9.7, & (2III + II - I) \\ -1.3x_1 + 0.2x_2 + 5.8x_3 = -1.4, & (III - II) \end{cases} \qquad (35)$$

$$\begin{cases} 10x_1 = 2.4x_1 - 0.5x_2 - 2.4x_3 + 1.9; \\ 10x_2 = -2.2x_1 + 0.9x_2 - 4.4x_3 + 9.7; \\ 10x_3 = 1.3x_1 - 0.2x_2 + 4.2x_3 - 1.4; \end{cases}$$

$$\begin{cases} x_1 = 0.24x_1 - 0.05x_2 - 0.24x_3 + 0.19; \\ x_2 = -0.22x_1 + 0.09x_2 - 0.44x_3 + 0.97; \\ x_3 = 0.13x_1 - 0.02x_2 + 0.42x_3 - 0.14. \end{cases}$$

Здесь $\|G\| = \max\limits_{1 \le i \le n} \sum\limits_{j=1}^{n} |g_{ij}| = \max\{0.53;\ 0.75;\ 0.57\} = 0.75 < 1$, значит, процесс Зейделя сходится.

По технологии счета (32) $\overline{x^0} = \{0.19; 0.97; -0.14\}$.

$x_1^{(1)} = 0.24 * 0.19 - 0.05 * 0.97 + 0.24 * 0.14 + 0.19 = 0.2207;$

$x_2^{(1)} = -0.22 * 0.2207 + 0.09 * 0.97 + 0.44 * 0.14 + 0.97 = 1.0703;$

$x_3^{(1)} = 0.13 * 0.2207 - 0.02 * 1.0703 - 0.42 * 0.14 - 0.14 = -0.1915;$

k	x_1	x_2	x_3	k	x_1	x_2	x_3
0	0.19	0.97	–0.14	5	0.2467	1.1135	–0.2237
1	0.2207	1.0703	–0.1915	6	0.2472	1.1143	–0.2241
2	0.2354	1.0988	–0.2118	7	0.2474	1.1145	–0.2243
3	0.2424	1.1088	–0.2196	8	0.2475	1.1145	–0.2243
4	0.2454	1.1124	–0.2226				

Ответ: $x_1 = 0.248$; $x_2 = 1.115$; $x_3 = –0.224$.

Замечание. Если для одной и той же системы методы простой итерации и Зейделя сходятся, то метод Зейделя предпочтительнее. Однако на практике имеет место ситуация, когда области сходимости этих методов могут быть различными, т.е. метод простой итерации сходится, а метод Зейделя расходится и наоборот. Для обоих методов, если $\|G\|$ близка к *единице*, скорость сходимости очень малая.

Для ускорения сходимости тогда используется искусственный прием – так называемый *метод релаксации*. Суть его заключается в том, что полученное по методу итерации очередное значение $x_i^{(k)}$ пересчитывается по формуле:

$$x_i^{(k)} = \omega x_i^{(k)} + (1-\omega)x_i^{(k-1)} \qquad (36)$$

ω – как правило, принято изменять в пределах $0 < \omega \le 2$ с каким-то шагом (можно $h = 0{,}1$ или $0{,}2$). Параметр ω подбирают так, чтобы сходимость метода достигалась за минимальное число итераций.

Релаксация – (физ.тех.) постепенное ослабление какого-либо состояния тела после прекращения действия факторов вызвавших это состояние.

Пример. Рассмотрим результат пятой итерации с применением формулы релаксации. Возьмем $\omega = 1{,}5$.

$$x_1^{(5)} = 1.5 * 0.2467 - 0.5 * 0.2454 = 0.24735;$$
$$x_2^{(5)} = 1.5 * 1.1138 - 0.5 * 1.1124 = 1.1145;$$
$$x_3^{(5)} = 1.5(-0.2237) + (0.5 * 0.2226) = -0.22425.$$

Как видно мы получили почти результат седьмой итерации.

2.3. Вычисление определителей высоких порядков

В отличие от технологии вычисления определителей в методе Крамера, для матриц общего вида, являющихся элементом СЛАУ, для решения этой задачи успешно может использоваться метод Гаусса. Прямой ход метода для системы $A\bar{x} = 0$ позволяет вычислить

$$\Delta = \det A = a_{11} * a_{22}^{(1)} ... a_{nn}^{(n-1)} = \pm \prod_{k=1}^{n} a_{kk} \ ,$$

так как последовательное исключение элементов величину определителя не изменяет. Здесь a_{kk} – элементы преобразованной матрицы A (прямой ход Гаусса). Знак зависит от четности или нечетности перестановок строк исходной матрицы при приведении ее к треугольному виду во избежание делении на «0» или необходимости поиска «*max*» ведущего элемента в текущем столбце на каждом этапе исключение неизвестных.

Для симметричных матриц

$$T = \begin{vmatrix} t_{11} & t_{12} & ... & t_{1n} \\ 0 & t_{22} & ... & t_{2n} \\ & \cdots & & \\ 0 & 0 & ... & t_{nn} \end{vmatrix}; \qquad \Delta = \det A = \left(t_{11} \cdot t_{22} \cdot ... \cdot t_{nn}\right)^2 .$$

2.4. Вычисление обратных матриц

1. ***По методу Гаусса***. Всякая неособенная матрица, для которой $\det A \neq 0$, имеет обратную матрицу. Очевидно, что $A^*A^{-1} = E$. запишем это равенство в виде системы n уравнений с n неизвестными

$$\sum_{k=1}^{n} a_{ik} z_{kj} = \delta_{ij}; \quad i, j = \overline{1,n}; \tag{37}$$

где a_{ik} – элементы матрицы A;

z_{kj} – элементы обратной матрицы (A^{-1});

δ_{ij} – элементы единичной матрицы.

При этом $\delta_{ij} = \begin{cases} 1, i = j; \\ 0, i \neq j. \end{cases}$

Для нахождения элементов одного столбца обратной матрицы необходимо решить соответствующую линейную систему (37) с матрицей A. Так для получения j-го столбца для A^{-1} ($z_{1j,} z_{2j,} \ldots z_{nj}$) решается система:

$$\begin{cases} a_{11} z_{1j} + a_{12} z_{2j} + ... + a_{1n} z_{nj} = 0; \\ \qquad\qquad \cdots \\ a_{j1} z_{1j} + a_{j2} z_{2j} + ... + a_{jn} z_{nj} = 1; \\ \qquad\qquad \cdots \\ a_{n1} z_{1j} + a_{n2} z_{2j} + ... + a_{nn} z_{nj} = 0. \end{cases} \tag{38}$$

Следовательно для обращения матрицы A нужно n раз решить систему (38) при $j=\overline{1,n}$. Поскольку матрица A системы не меняется, то исключение неизвестных

осуществляется только один раз, а (n–1) раз при решении (38) делается только обратный ход с соответствующим изменением правой ее части.

2. Другой подход к определению обратной матрицы A^{-1}

$$A^{-1} = \frac{1}{\Delta}\begin{pmatrix} A_{11} & A_{21}. & . & A_{n1} \\ A_{12} & A_{22} & . & A_{n2} \\ ... & ... & . & ... \\ A_{1n} & A_{2n} & . & A_{nn} \end{pmatrix},$$

где Δ – определитель матрицы, A_{ij} – алгебраические дополнения соответствующих элементов матрицы A.

3. Обращение матрицы А посредством треугольных матриц

Известно, что всякая обратная матрица, если она существует, то по структуре будет такая же, как и исходная, т.к.

$$A^{-1}\cdot A = A\cdot A^{-1} = E = \begin{vmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{vmatrix}. \qquad (39)$$

Рассмотрим пример обращения матрицы 3-го порядка следующего вида:

$$A = \begin{vmatrix} 1 & 0 & 0 \\ 1 & 2 & 0 \\ 1 & 2 & 3 \end{vmatrix}. \qquad (40)$$

Решение. Матрицу A^{-1} ищем в виде

$$A^{-1} = \begin{vmatrix} t_{11} & 0 & 0 \\ t_{21} & t_{22} & 0 \\ t_{31} & t_{32} & t_{33} \end{vmatrix}. \qquad (41)$$

Перемножая A и A^{-1} с учетом (39) будем иметь $t_{11} = 1$; $t_{11} + 2t_{21} = 0$; $2t_{22} = 1$;

$$\begin{cases} t_{11} + 2t_{21} + 3t_{31} = 0; \\ 2t_{22} + 3t_{32} = 0; \\ 3t_{33} = 1. \end{cases}$$

Отсюда последовательно находим $t_{11} = 1$; $t_{21} = –1/2$; $t_{31} = 0$; $t_{22} = 1/2$; $t_{32} = –1/3$; $t_{33} = 1/3$, следовательно

$$A^{-1} = \begin{vmatrix} 1 & 0 & 0 \\ -1/2 & 1/2 & 0 \\ 0 & -1/3 & 1/3 \end{vmatrix}. \qquad (42)$$

Перемножив (42) и (40) получим (39).

Известно, что любая произвольная матрица A может быть представлена в виде двух треугольных.

Например, пусть имеется матрица

$$A = \begin{vmatrix} 1 & -1 & 2 \\ -1 & 5 & 4 \\ 2 & -1 & 14 \end{vmatrix}. \qquad (43)$$

Будем искать $T_1 = \begin{vmatrix} t_{11} & 0 & 0 \\ t_{21} & t_{22} & 0 \\ t_{31} & t_{32} & t_{33} \end{vmatrix}$ и $T_2 = \begin{vmatrix} 1 & r_{12} & r_{13} \\ 0 & 1 & r_{23} \\ 0 & 0 & 1 \end{vmatrix}$. Диагональ в матрице T_2 искусственно берется равной 1. Тогда

$$A = T_1 \cdot T_2 . \qquad (44)$$

Реализуя (44) и сравнивая с (43), получим

$$\begin{vmatrix} 1 & -1 & 2 \\ -1 & 5 & 4 \\ 2 & -1 & 14 \end{vmatrix} = \begin{vmatrix} t_{11} & t_{11}r_{12} & t_{11}r_{13} \\ t_{21} & t_{21}r_{12} + t_{22} & t_{21}r_{13} + t_{22}r_{23} \\ t_{31} & t_{31}r_{12} + t_{32} & t_{31}r_{13} + t_{32}r_{23} + t_{33} \end{vmatrix}.$$

Сравнивая значения правой и левой частей и выполняя простейшие вычисления, очевидно:

$t_{11} = 1$; $t_{11}\, r_{12} = -1$; $t_{11}\, r_{13} = 2$;
$t_{21} = -1$; $t_{21}\, r_{12} + t_{22} = 5$; $t_{21}\, r_{13} + t_{22}\, r_{23} = 4$;
$t_{31} = 2$; $t_{31}\, r_{12} + t_{32} = -1$; $t_{31}\, r_{13} + t_{32}\, r_{23} + t_{33} = 14$;

Решив полученную систему, получим

$t_{11} = 1$; $t_{21} = -1$; $t_{31} = 2$;
$t_{22} = 4$; $t_{32} = 6$; $t_{31} = 1$;
$r_{12} = -1$; $r_{13} = 2$; $r_{23} = 3/2$.

Таким образом $T_1 = \begin{vmatrix} 1 & 0 & 0 \\ -1 & 4 & 0 \\ 2 & 6 & 1 \end{vmatrix}$ и $T_2 = \begin{vmatrix} 1 & -1 & 2 \\ 0 & 1 & 3/2 \\ 0 & 0 & 1 \end{vmatrix}$, тогда $A^{-1} = T_2^{-1} \cdot T_1^{-1}$.

2.5. Применение метода итераций для уточнения элементов обратной матрицы

Точность получения элементов обратной матрицы естественно оценивается соотношением

$$A^{-1} \cdot A = A^0 = E.$$

Однако в общем случае элементы обратной матрицы получаются с некоторой погрешностью, которая появляется в результате округлений в процессе вычисления и большого числа арифметических операций. Для уменьшения по-

грешностей используется ***итерационная*** схема уточнения элементов обратной матрицы.

Пусть для неособенной матрицы A получено приближенное значение элементов матрицы A^{-1}. Обозначим ее через $D_0 \approx A^{-1}$. Тогда для уточнения элементов обратной матрицы строится следующий итерационный процесс:

$$F_{k-1} = E - AD_{k-1}\ ,\ k = 1,2,3\ldots\ ; \qquad (*)$$

$$D_k = D_{k-1}(E + F_{k-1});\ k = 1,2,3\ldots \qquad (**)$$

Доказано, что итерации сходятся, если начальная матрица D_0 достаточно близка к искомой A^{-1}.

В данной итерационной схеме матрица F на каждом шаге как бы оценивает близость матрицы D к A^{-1}.

Схема работает следующим образом.

Сначала по (*) при $k = 1$ находится $F_0 = E - AD_0$, затем находится произведение D_0F_0.

По итерации (**) при $k = 1$ находится $D_1 = D_0 + D_0F_0$.

Чтобы проверить, достигнута ли желаемая точность, вычисляется AD_1, а по (*) при $k = 2$, вычисляется $F_1 = E - AD_1$ и, если наибольший элемент матрицы $F_1 < \varepsilon$, итерации прекращаются и $A^{-1} \approx D_1$.

Раздел 3. Численное решение нелинейных уравнений

3.1. Постановка задачи

Одной из важных практических задач при исследовании различных свойств математической модели в виде функциональной зависимости $y = f(x)$ является нахождение значений x, при которых эта функция обращается в ноль, т.е. решение уравнения

$$f(x) = 0\ . \qquad (1)$$

Как правило, точное решение его можно получить только в исключительных случаях, так как оно в большинстве случаев носит нелинейный характер. Нелинейные уравнения делятся на два класса:

1) ***алгебраические***, содержащие только алгебраические выражения;

2) ***трансцендентные***, содержащие и другие функции (тригонометрические, показательные, логарифмические и др.).

Методы решения нелинейных уравнений делятся на ***прямые*** и ***итерационные*** методы.

Прямые методы позволяют записать корни в виде некоторых конечных соотношений (формул) для простых тригонометрических, логарифмических, показательных и простейших алгебраических уравнений.

Однако подавляющее число практически значимых уравнений могут быть решено только *итерационными методами*, т.е. методами последовательных приближений (численными методами).

Решение уравнений (1) при этом осуществляется в два этапа:

1) определение местоположения, характера интересующего нас корня и выбор его начального значения;

2) вычисление корня с заданной точностью ε, посредством выбранного какого-либо вычислительного алгоритма.

На первом этапе вначале определяют, какие корни требуется найти, например, только действительные или только положительные или наименьший корень и т.д. Затем находят отрезки из области определения функции $y = f(x)$, взятой из (1), содержащие по одному корню.

Имеются различные подходы к решению данной задачи для обоих видов нелинейных уравнений.

На втором этапе используются итерационные методы, позволяющие с помощью некоторого рекуррентного соотношения

$$x_n \approx x^k = \varphi(x^{k-1}, x^{k-2}, ..., x^{k-m}) \qquad (2)$$

при выбранном начальном приближении к x^* построить последовательность (x_n).

Как правило, всегда стоит задача обеспечения сходимости последовательности (2) к истинному значению корня x^*. Сходимость достигается посредством выбора различными способами функций φ в (2), которая зависит от $f(x)$ и в общем случае от номера последовательности решений (n). При этом если при нахождении значения $x_n \approx x^k \approx x^*$, используется одно предыдущее значение m=1, то такой метод называется одношаговым. Если используется m предыдущих значений, то метод называется m-шаговым и, как правило, с увеличением m вычислительные алгоритмы усложняются.

Расчет по рекуррентной последовательности продолжается до тех пор, пока $| x_n - x_{n-1}| < \varepsilon$. Тогда последнее x_n выбирается в качестве приближенного значения корня ($x^* \approx x_n$).

На практике имеется большой выбор законов φ, что обеспечивает многообразие численных итерационных методов решения нелинейных уравнений.

3.2. Отделение корней

3.2.1. Метод половинного деления

Отделить корень x^* уравнения $f(x) = 0$ – значит указать окрестность точки x^*, не содержащую других корней этого уравнения.

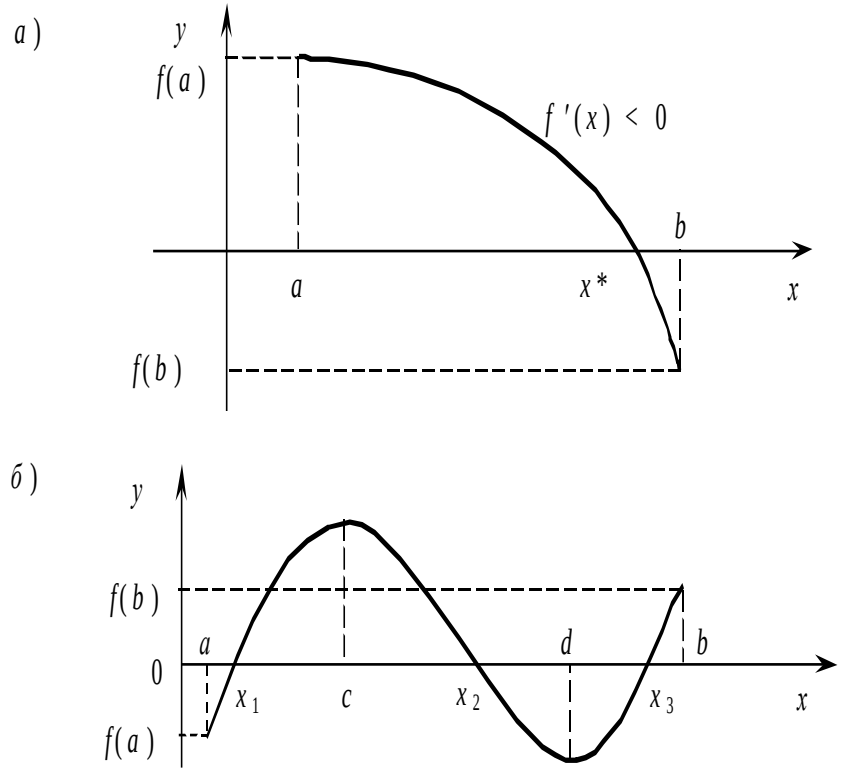


Рис. 3.1

Как известно из анализа, если непрерывная функция $f(x)$ на концах отрезка $[a, b]$ принимает значения разных знаков, т.е. если $f(a)\cdot f(b) < 0$, то внутри этого отрезка существует, по крайней мере, один корень уравнения $f(x) = 0$ (рис 3.1). При этом корень x^* будет единственным, если $f'(x)$ сохраняет знак внутри интервала (a, b) (рис. 3.1*а*).

На практике отделение корней уравнения $f(x) = 0$ на отрезке $[a,b]$ и начинается с проверки условия $f(a)\cdot f(b) < 0$. Если это условие выполнено, то, следовательно, на (a,b) имеется корень и дальнейшая задача состоит в выяснении его единственности или не единственности.

Для отделения корней практически достаточно провести *процесс половинного деления*, в соответствии с которым отрезок $[a, b]$ делится на 2,4,8,… равных частей и последовательно определяются знаки функции в точках деления. При этом если в точках деления x_i, x_{i+1} выполнено условие $f(x_i)\cdot f(x_{i+1}) < 0$, то на интервале (x_i, x_{i+1}) имеется корень уравнения $f(x) = 0$. При определении корней всегда стараются найти интервал (x_i, x_{i+1}) как можно меньшей длины.

Согласно вышеизложенному, получается следующий алгоритм определения корней уравнения $f(x) = 0$:

1) находим участки возрастания и убывания функции $f(x)$ (с помощью производной $f'(x)$, если она существует);

2) составляем таблицу знаков функции $f(x)$ в стационарных точках (или ближайших к ним), а так же в граничных точках области определения $f(x)$;

3) определяем интервалы по правилу $x_i = a + (i - 1)\cdot(b - a)/m - 1$; $i = 1, 2, \ldots, m$, на которых $f(x)$ имеет противоположные знаки. Внутри таких интервалов содержится только по одному корню. На рисунке 3.1*б* интервалы монотонности

функции (a,c), (c,d), (d,b), на концах которых функция имеет противоположные знаки. Корнями уравнения $f(x) = 0$ на отрезке $[a, b]$ в данном случае являются точки x_1, x_2 и x_3.

3.2.2. Графическое отделение корней

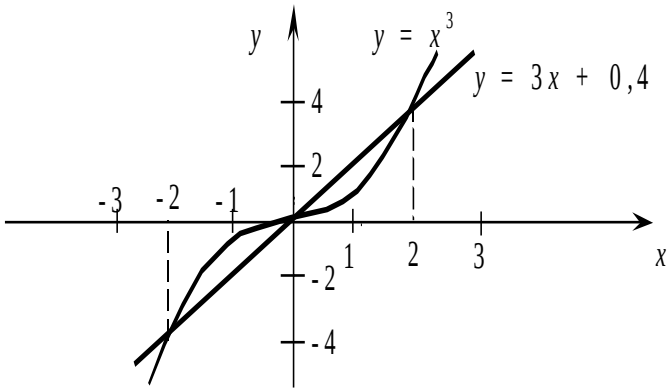
Очевидно, что найти корень уравнения (1) означает найти абсциссу точки пересечения графика $y = f(x)$ с прямой $y = 0$, т.е. осью абсцисс. При этом если построение $y = f(x)$ затруднительно, то ее представляют в эквивалентном виде:

$$f_1(x) = f_2(x) \tag{3}$$

с таким расчетом, чтобы графики $y_1 = f_1(x)$ и $y_2 = f_2(x)$ строились проще. Абсциссы их точек пересечения и будут корнями уравнения (1).

Рассмотрим в качестве примера уравнение $x^3 - 3x - 0{,}4 = 0$. Согласно (3) запишем его как

$$x^3 = 3x + 0{,}4\ . \tag{4}$$



Из рисунка видно, что здесь три корня: $c_1 \in [-2, -1]$; $c_2 \in [-1, 0]$; $c_3 \in [1, 2]$.

При графическом отделении корней уравнения результат зависит от точности построения графиков.

3.3. Итерационные методы уточнения корней

3.3.1. Метод простой итерации

Метод простой итерации применяется к решению уравнения (1), разрешенному относительно x:

$$x = \varphi(x). \tag{5}$$

Переход от записи (1) к эквивалентной записи (5) можно сделать многими способами.

Метод состоит в построении последовательности (2) в виде:

$$x_{n+1} = \varphi(x_n),\ n = 0,1,2,\ldots.$$

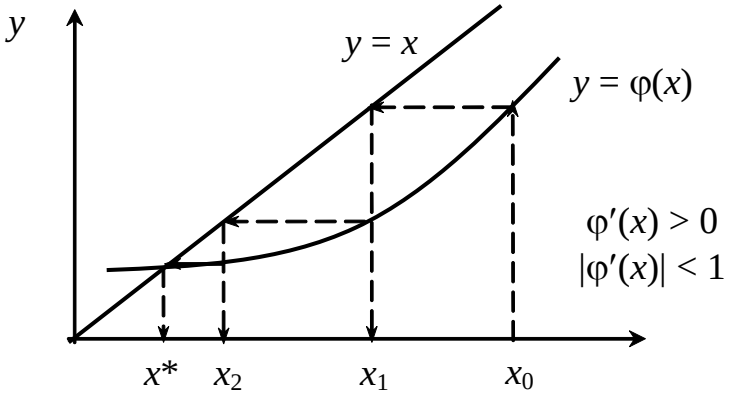
Если $\varphi(x_n)$ – непрерывная функция, а x_n – сходящаяся последовательность, то искомое значение $x^* = \lim\limits_{n\to\infty} x_n$ и будет решением (5), а, следовательно, и (1).

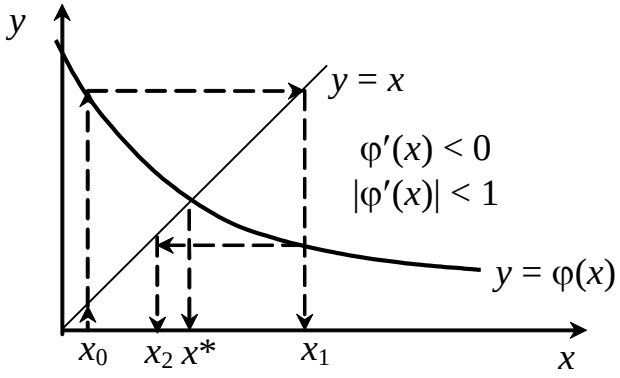
Например, получим (5) из (1) следующим образом: умножим (1) на подобранную специальным образом функцию $\psi(x) \neq 0$ (в частности можно взять $\psi(x) = \text{const}$) и сложим с тождеством $x = x$, тогда (5) будет иметь вид, эквивалентный виду (3):

$$x = x + \psi(x) f(x) = \varphi(x). \qquad (6)$$

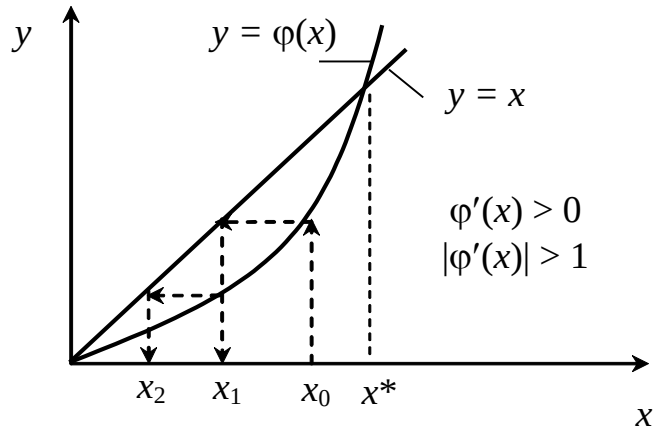
Подбирая $\psi(x)$ добиваются сходимости решения (6). Она может быть монотонной (если $\varphi'(x) > 0$), или колеблющейся (если $\varphi'(x) < 0$).

Метод, очевидно, является одношаговым (m=1) и для начала вычислений нужно знать одно начальное приближение $x_0 = \alpha$, или $x_0 = \beta$, или $x_0 = (\alpha+\beta)/2$.





В методе простой итерации сходимость гарантированна не всегда, например, если $\varphi(x)$ имеет такой характер:



Такая ситуация может быть устранена подбором $\psi(x)$ в (6).

Что касается выбора $\psi(x)$, то можно взять, например, $\psi(x) = \text{Const} = 1/k$. В этом случае необходимо, чтобы $|k| > \max| f'(x)| / 2$. При этом знак k должен совпадать со знаком $f'(x)$.

Доказано, что в общем случае расходимость (несходимость) исключается, если подбирается соотношение

$$| \varphi'(x) | \le q < 1. \qquad (7)$$

При этом скорость сходимости увеличивается при уменьшении величины q.

Максимальный интервал (α, β) при выполнении условия (7) называется областью сходимости. Для данной оценки (7) берется любое $x \in (\alpha,\beta)$; $x^* \in (\alpha,\beta)$.

Итерационный процесс уточнения корня заканчивается, когда $\{|x_n - x_{n-1}|$ или $|f(x_n) - f(x_{n-1})|\} < \varepsilon$.

3.3.2. Метод Ньютона (касательных)

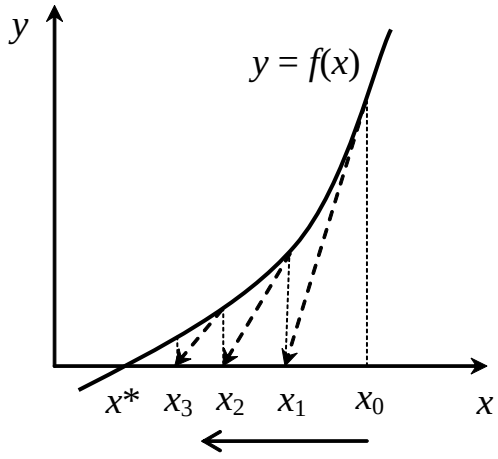
Данный метод является модификацией метода простой итерации. Если функция $f(x)$ непрерывна и дифференцируема, то выбрав в (6) $\psi(x) = -\dfrac{1}{f'(x)}$ получим эквивалентное уравнение в виде $x = x - f(x)/f'(x) = \varphi(x)$, $f'(x) \ne 0$.

Подбором $\psi(x)$ добиваются, чтобы в (7) $q = \varphi'(x^*) \equiv 0$, что обеспечивает большую скорость сходимости *в рекуррентном* соотношении метода в близи искомого корня

$$x_n = x_{n-1} - f(x_{n-1})/ f'(x_{n-1}) = \varphi(x_{n-1}), \; n = 1,2,\ldots \qquad (8)$$

Это также одношаговый метод.

Геометрическая интерпретация метода представлена на рисунке.



Проблематичным является выбор x_0 в виду узости области сходимости вычисления производной. Часто при неудачном выборе x_0 нет монотонного убывания последовательности $|f(x_n)|$, поэтому рекомендуется вычисления проверить по модифицированной схеме

$$x_{n+1} = x_n - \alpha_n \left[f'(x_n) \right]^{-1} f(x_n), \quad n = 0,1,2,\ldots$$

Здесь сомножители $\alpha_n \in [0,1]$ выбирают так, чтобы выполнялось неравенство

$$| f(x_{n+1})| < | f(x_n)| \, .$$

При выборе начального приближения x_0 предпочтительней использовать заведомо сходящийся метод, например, метод деления отрезка пополам.

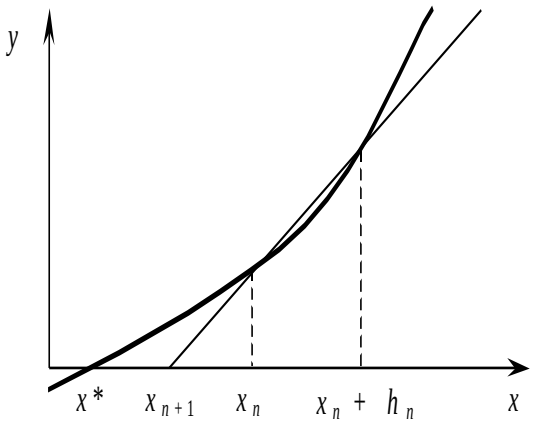
3.3.3. Метод секущих

Этот метод является модификацией метода Ньютона в плане его реализации, т.е. задача поиска корня связана лишь с вычислением значения функции $f(x)$. Заменив производную $f'(x_n)$ в методе Ньютона так называемой разделенной разностью по двум точкам x_n и $x_n + h_n$, где h_n – некоторый малый параметр, получим итерационную формулу

$$x_{n+1} = x_n - \frac{h_n f(x_n)}{f(x_n + h_n) - f(x_n)}, \quad n = 0,1,2,\ldots , \tag{9}$$

которая называется методом секущих.

Приближение x_{n+1} является абсциссой точки пересечения секущей прямой, проведенной через точки $(x_n, f(x_n))$ и $(x_n+h_n, f(x_n + h_n))$ с осью x.



Метод также одношаговый и при удачном подборе параметра h его сходимость, как и у метода Ньютона при упрощении его реализации.

Имеются другие интерпретации формулы (9). В частности, ***метод Вегстейна***, в котором для выбора параметра h используют предыдущую расчетную точку, т.е. берут $h_n = x_{n-1} - x_n$, тогда (9) имеет вид:

$$x_{n+1} = x_n - \frac{(x_n - x_{n-1}) f(x_n)}{f(x_n) - f(x_{n-1})},\ n = 0,1,2,\ldots \qquad (10)$$

Метод Вегстейна, очевидно, двухшаговый ($m = 2$), т.е. для вычисления требуется задать 2 начальные точки приближения, лучше всего $x_0 = a$; $x_1 = b$. Он медленнее метода секущих, однако, требует в 2 раза меньше вычислений $f(x)$ и поэтому оказывается более эффективным.

Целесообразным является использовать подходы к уточнению корня не выпускающие корень из выделенной «вилки», (отрезка $[a, b]$).

Так, если $f(b) \cdot f''(x) > 0$ для $x \in [a,b]$, берут в качестве $x_0 = a$ и уточнение корня производится по формуле

$$x_{n+1} = x_n - \frac{(b - x_{n-1}) f(x_n)}{f(b) - f(x_n)}, \qquad n=0,1,2,\ldots, \qquad (11)$$

а если $f(a) \cdot f''(x) > 0$ для $x \in [a,b]$, берут в качестве $x_0 = b$ и уточнение корня производится по формуле

$$x_{n+1} = x_n - \frac{(x_n - a) f(x_n)}{f(x_n) - f(a)}, \qquad n=0,1,2,\ldots \qquad (12)$$

3.3.4. Метод деления отрезка пополам

Все вышеизложенные методы могут работать, если функция $f(x)$ из (1) является непрерывной и дифференцируемой вблизи искомого корня, в противном случае решение не гарантируется. Данный метод может быть использован даже для разрывных функций.

Его алгоритм реализовывается согласно следующей рекуррентной последовательности: для $x^* \in [\alpha,\beta]$; $x_0 = \alpha$; $x_1 = \beta$, находится $x_2 = (\alpha+\beta)/2$.

Очередная точка x_3 выбирается как середина того из смежных с x_2 интервалов $[x_0, x_2]$ или $[x_2, x_1]$, на котором находится корень. В результате получается следующий алгоритм метода деления отрезка пополам:

1) вычисляем $y_0 = f(x_0)$;

2) вычисляем $x_2 = (x_0 + x_1)/2,\ y_2 = f(x_2)$;

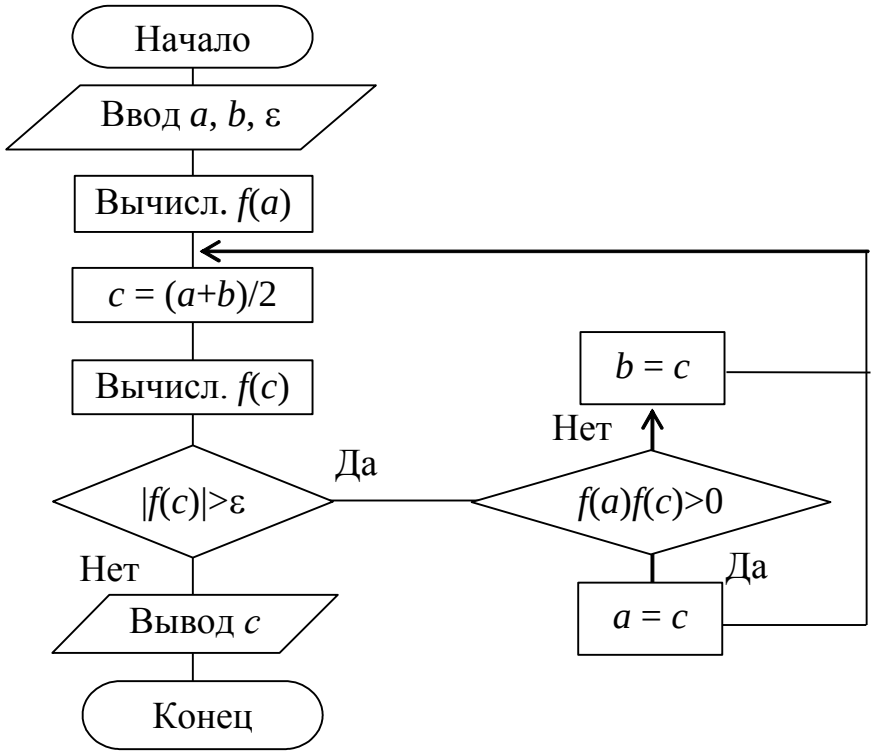
3) если $y_0 \cdot y_2 > 0$, тогда $x_0 = x_2$, иначе $x_1 = x_2$; (13)

4) если $x_1 - x_0 > \varepsilon$, тогда повторять с п. 1;

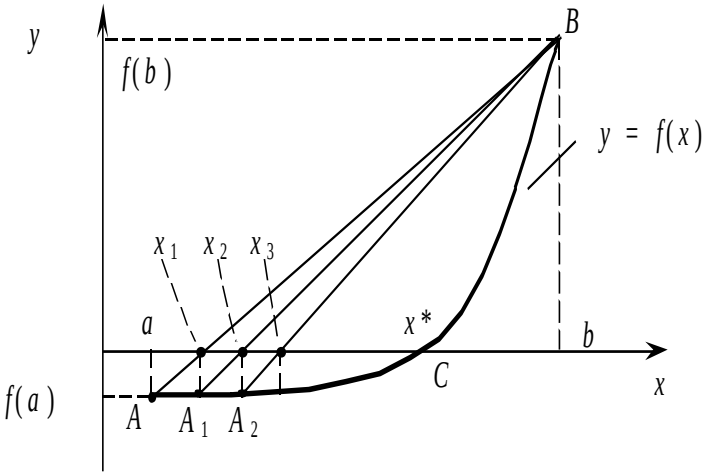
5) вычисляем $x^* = (x_0 + x_1)/2$.

За одно вычисление функции погрешность уменьшается вдвое, т.е. скорость сходимости невелика, однако метод устойчив к ошибкам округления и всегда сходится.

Немного подкорректировав алгоритм (13), его более наглядно можно представить в виде блок-схемы:



3.3.5. Метод хорд



Пусть корень C уравнения $f(x)=0$ отделен на $[a,b]$. Функция $f(x)$ непрерывна на отрезке и на его концах имеет разные знаки. Точки A и B имеют координаты соответственно $(a, f(a))$ и $(b, f(b))$

Искомым корнем C будет пресечение $f(x)$ с осью OX. В начале итераций вместо C ищется приближение x_1, как результат пересечения OX с хордой AB.

Уравнение прямой AB запишем в виде $\dfrac{x-b}{y-f(b)}=\dfrac{x-a}{y-f(a)}$.

Полагая $y=0$, находим $x_1=\dfrac{a\cdot f(b)-b\cdot f(a)}{f(b)-f(a)}$. Это можно записать в виде:

$$\left.\begin{array}{l} x_1 = a - f(a)\dfrac{b-a}{f(b)-f(a)}; \\ \text{или} \\ x_1 = b - f(b)\dfrac{b-a}{f(b)-f(a)}; \end{array}\right\} \qquad (14)$$

Если x_1 оказывается недостаточно точным, находят второе приближение:

$$x_2 = x_1 - f(x_1)\frac{b-x_1}{f(b)-f(x_1)}. \qquad (15)$$

На основании (14) и (15) можно записать рекуррентную последовательность:

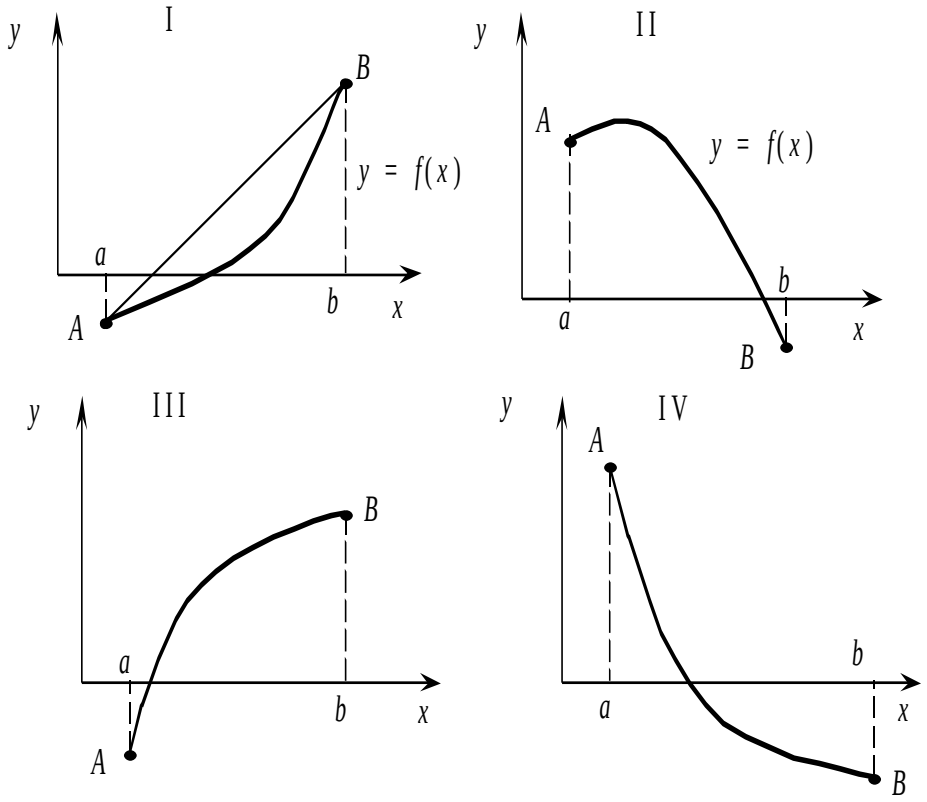
$$x_{k+1} = x_k - f(x_k)\frac{b-x_k}{f(b)-f(x_k)}, \qquad (16)$$

если $f(x_k)\cdot f(b)<0$, и

$$x_{k+1} = x_k - f(x_k)\frac{x_k-a}{f(x_k)-f(a)} \qquad (17)$$

если $f(x_k)\cdot f(a)<0$.

Заметим, что на выделенном интервале $[a, b]$ имеют место четыре типа расположения кривой $f(x)$.



Для I-го $f'(x) > 0$, $f''(x) > 0$, для II-го $f'(x) < 0$, $f''(x) < 0$, для III-го $f'(x) > 0$, $f''(x) < 0$; для IV-го $f'(x) < 0$, $f''(x) > 0$.

Тогда для I-го и для II-го используется (16), т.е. $x_0 = a$. Для III-го и IV-го используется (17), т.е. $x_0 = b$.

В заключение заметим, что во всех методах для определения функции $f(x)$ и ее производных целесообразно использовать схему Горнера.

3.4. Общий алгоритм численных методов решения нелинейных уравнений

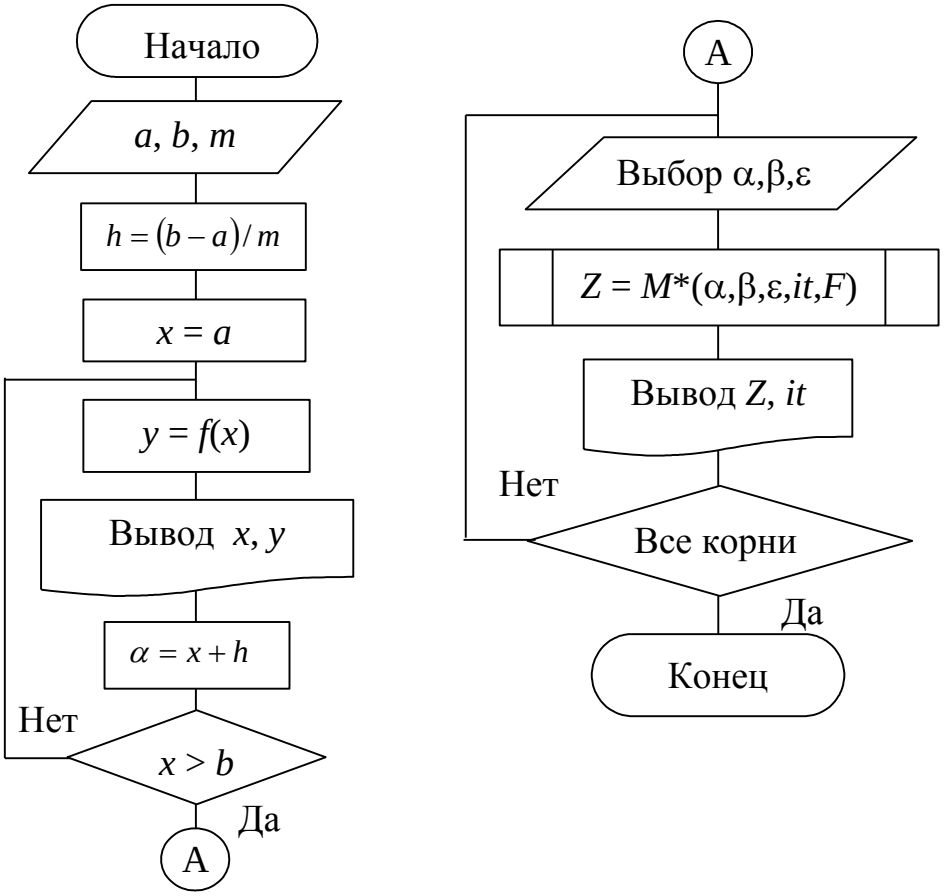
Рассмотрим реализацию двух этапов их решения.

1. Программа должна сначала выдать таблицу значений $y = f(x)$ (отделений корней).

2. Далее делается запрос на ввод начального приближения (это α, β или $(\alpha+\beta)/2$) и точности решения ε.

Расчет функции и вычислительный алгоритм обычно выполняются в виде отдельных подпрограмм.

Примерный алгоритм данных процедур может иметь следующий вид:



Значение *m* выбираем по усмотрению, но с соблюдением принципа «половинного деления», рассмотренного выше.

Раздел 4. Решение систем нелинейных уравнений

4.1. Постановка задачи

Многие практические задачи сводятся к решению систем нелинейных уравнений с *n* неизвестными:

$$\left.\begin{aligned} F_1(x_1, x_2, ..., x_n) &= 0; \\ F_2(x_1, x_2, ..., x_n) &= 0; \\ &\dots \\ F_n(x_1, x_2, ..., x_n) &= 0. \end{aligned}\right\} \quad (1)$$

В отличие от линейных систем прямых методов их решения нет за исключением систем второго порядка, когда одно неизвестное может быть выражено через другое.

Наиболее распространены два метода: метод простой итерации и метод Ньютона.

4.2. Метод простой итерации

Система (1) должна быть представлена в следующем виде:

$$\left.\begin{aligned} x_1 &= f_1(x_1, x_2, ..., x_n); \\ x_2 &= f_2(x_1, x_2, ..., x_n); \\ &\dots \\ x_n &= f_n(x_1, x_2, ..., x_n); \end{aligned}\right\} \quad (2)$$

где $f_i(x_1, x_2, ..., x_n)$ называются итерирующими функциями.

Алгоритм решения аналогичен алгоритму Зейделя или простой итерации для решения систем линейных уравнений.

Пусть известен начальный вектор решения: $x_i = a_i$, $i = 1,2,\dots,n$, тогда

$$\left.\begin{aligned} x_1 &= f_1(a_1, a_2, ..., a_n); \\ x_2 &= f_2(x_1, a_2, ..., a_n); \\ &\dots \\ x_i &= f_i(x_1 ..., x_{i-1}, a_{i,} ..., a_n); \\ &\dots \\ x_n &= f_n(x_1 ..., x_{n-1}, a_n). \end{aligned}\right\}$$

Итерационный процесс продолжается до тех пор, пока изменение всех неизвестных в двух последовательных итерациях не станет меньше заданного значения ε.

Начальные значения должны быть близкими к истинным значениям, иначе итерационный процесс может не сойтись. Стоит проблема их отыскания (т.е. условий сходимости). В случае расходимости (несходимости) в блок-схеме алгоритма срабатывает механизм ограничения числа итераций.

4.2.1. Условия сходимости метода простой итерации для нелинейных систем уравнений второго порядка

Рассмотрим систему из двух уравнений общего вида

$$\left.\begin{aligned} F_1(x,y)=0;\\ F_2(x,y)=0. \end{aligned}\right\} \tag{3}$$

Нужно найти действительные корни x и y с заданной степенью точности ε.

Предположим, что данная система имеет корни и их можно установить. Итак, для применения метода простой итерации систему (3) нужно привести к виду:

$$\left.\begin{aligned} x=\varphi_1(x,y);\\ y=\varphi_2(x,y); \end{aligned}\right\}, \tag{4}$$

где φ_1 и φ_2 – итерирующие функции. По ним и строится итерационный процесс решения в виде:

$$\left.\begin{aligned} x_{n+1}=\varphi_1(x_n,y_n);\\ y_{n+1}=\varphi_2(x_n,y_n); \end{aligned}\right\}, \; n=0,1,2,\ldots \tag{5}$$

где при $n = 0$, x_0 и y_0 – начальные приближения.

Имеет место *утверждение*: пусть в некоторой замкнутой области $R(a \le x \le A;\ b \le y \le B)$ имеется одно и только одно единственное решение $x=\gamma$; $y=\beta$, тогда:

1) если $\varphi_1(x,y)$ и $\varphi_2(x,y)$ определены и непрерывно дифференцируемы в R;

2) если начальное решение x_0, y_0 и все последующие решения x_n, y_n также принадлежат R;

3) если в R выполняются неравенства:

$$\left.\begin{aligned} \frac{|\partial\varphi_1|}{|\partial x|}+\frac{|\partial\varphi_2|}{|\partial x|}\le q_1<1\\ \frac{|\partial\varphi_1|}{|\partial y|}+\frac{|\partial\varphi_2|}{|\partial y|}\le q_2<1 \end{aligned}\right\} \tag{6}$$

или равносильные неравенства:

$$\left.\begin{aligned} \frac{|\partial\varphi_1|}{|\partial x|}+\frac{|\partial\varphi_1|}{|\partial y|}\le q_1<1\\ \frac{|\partial\varphi_2|}{|\partial x|}+\frac{|\partial\varphi_2|}{|\partial y|}\le q_2<1 \end{aligned}\right\} \tag{6`}$$

то тогда итерационный процесс (5) сходится к определенным решениям, т.е. $\lim\limits_{n\to\infty} x_n = \gamma,\ \lim\limits_{n\to\infty} y_n = \beta.$

Оценка погрешности *n*-го приближения дается неравенством:

$$|\gamma - x_n| + |\beta - y_n| \le \frac{M}{1-M}(|x_n - x_{n-1}| + |y_n - y_{n-1}|),$$

где M – наибольшее из чисел q_1 или q_2 в соотношениях (6) и (6`). Сходимость считается хорошей, если $M<1/2$. Если совпадают три значащие цифры после запятой в соседних приближениях, то обеспечивается точность $\varepsilon = 10^{-3}$.

Пример. С заданной точностью решить нелинейную систему второго порядка:

$$\begin{cases} x^3 + y^3 - 6x + 3 = 0 \\ x^3 - y^3 - 6y + 2 = 0 \end{cases}$$

Запишем систему в виде (4)

$$\begin{cases} x = \dfrac{x^3 + y^3}{6} + \dfrac{1}{2} = \varphi_1(x, y) \\ y = \dfrac{x^3 - y^3}{6} + \dfrac{1}{3} = \varphi_2(x, y) \end{cases}$$

Рассмотрим квадрат $0 \le x \le 1;\ 0 \le y \le 1$. Если возьмем x_0 и y_0 из этого квадрата, тогда мы имеем:

$$0 < \varphi_1(x_0, y_0) < 1$$
$$0 < \varphi_2(x_0, y_0) < 1$$

Из анализа вида φ_1 и φ_2 определим область нахождения их компонент при $x=y=1$, в заданном квадрате.

Для $\varphi_1(x, y)$: $0 < \dfrac{x^3 + y^3}{6} < \dfrac{1}{3}$, а для $\varphi_2(x, y)$: $-\dfrac{1}{6} < \dfrac{x^3 - y^3}{6} < \dfrac{1}{6}$, то при любом выборе (x_0, y_0) последовательность (x_k, y_k) останется в прямоугольнике:

$$\frac{1}{2} \le x \le \frac{5}{6}\ ;\ \frac{1}{6} \le y \le \frac{1}{2}\ ;$$

так как 1/3+1/2=5/6, 1/3–1/6=1/6, 1/3+1/6=1/2. Тогда для точек этого прямоугольника

$$\left|\frac{\partial \varphi_1}{\partial x}\right| + \left|\frac{\partial \varphi_1}{\partial y}\right| = q_1 = \frac{x^2}{2} + \frac{y^2}{2} < \frac{25/36 + 1/4}{2} = \frac{34}{72} < 1;$$

$$\left|\frac{\partial \varphi_2}{\partial x}\right| + \left|\frac{\partial \varphi_2}{\partial y}\right| = q_2 = \frac{x^2}{2} + \left|-\frac{y^2}{2}\right| < \frac{34}{72} < 1;$$

– условия удовлетворяются, и система может быть решена по методу простых итераций.

Полагаем $x_0 = 1/2$, $y_0 = 1/2$, тогда

$$x_1 = \frac{1}{2} + \frac{1/8 + 1/8}{6} = 0{,}542\ ;\ y_1 = \frac{1}{3} + \frac{1/8 - 1/8}{6} = 0{,}333\ .$$

Вторая итерация: $x_2 = \frac{1}{2} + \frac{0{,}14615}{6} = 0{,}533$; $y_2 = \frac{1}{3} + \frac{0{,}1223}{6} = 0{,}354$; …

x_3=0,533; y_3=0,351. Вычисляем дальше x_4 = 0,533; y_4 = 0,351 – эти значения и являются ответом.

4.2.2. Общий случай построения итерирующих функций

Рассмотрим вариант построения итерирующих функций вида (4) для системы (3) с соблюдением условий (6). Напишем их в следующем виде:

$$\varphi_1(x, y) = x + \alpha F_1(x, y) + \beta F_2(x, y);$$
$$\varphi_2(x, y) = y + \gamma F_1(x, y) + \delta F_2(x, y);$$

При этом должно выполняться $\alpha\delta \neq \beta\gamma$. Коэффициенты $\alpha, \beta, \gamma, \delta$ находятся из решения следующей системы линейных уравнений, которая составлена по требованиям (6′):

$$\begin{aligned}
&1 + \alpha \frac{\partial F_1(x_0, y_0)}{\partial x} + \beta \frac{\partial F_2(x_0, y_0)}{\partial x} = 0\\
&\alpha \frac{\partial F_1(x_0, y_0)}{\partial y} + \beta \frac{\partial F_2(x_0, y_0)}{\partial y} = 0\\
&\gamma \frac{\partial F_1(x_0, y_0)}{\partial x} + \delta \frac{\partial F_2(x_0, y_0)}{\partial x} = 0\\
&1 + \gamma \frac{\partial F_1(x_0, y_0)}{\partial y} + \delta \frac{\partial F_2(x_0, y_0)}{\partial y} = 0
\end{aligned} \qquad (7)$$

При таком подборе параметров условие (6′) будет соблюдено, если частные производные функций F_1 и F_2 изменяются не очень быстро в окрестности точки (x_0, y_0).

Пример:

$$\begin{cases} x^2 + y^2 - 1 = 0; & \varphi_1(x, y) - ? \\ x^3 - y = 0; & \varphi_2(x, y) - ? \end{cases}$$

при $x_0 = 0{,}8$ и $y_0 = 0{,}55$. Для решения будем искать итерирующие функции в виде:

$$\varphi_1(x, y) = x + \alpha(x^2 + y^2 - 1) + \beta(x^3 - y);$$
$$\varphi_2(x, y) = y + \gamma(x^2 + y^2 - 1) + \delta(x^3 - y).$$

Составим систему (7). Предварительно определим компоненты системы (7) для $x_0 = 0{,}8$; $y_0 = 0{,}55$

$$\frac{\partial F_1}{\partial x}=2x;\qquad \frac{\partial F_1(x_0,y_0)}{\partial x}=1{,}6;$$

$$\frac{\partial F_2}{\partial x}=3x^2;\qquad \frac{\partial F_2(x_0,y_0)}{\partial x}=1{,}92;$$

$$\frac{\partial F_1}{\partial y}=2y;\qquad \frac{\partial F_1(x_0,y_0)}{\partial x}=1{,}1;$$

$$\frac{\partial F_2}{\partial y}=-1;\qquad \frac{\partial F_2(x_0,y_0)}{\partial x}=-1.$$

Тогда система (7) имеет вид:

$$\left.\begin{array}{l}1+1{,}6\alpha+1{,}92\beta=0\\ 1{,}1\alpha-\beta=0\\ 1{,}6\gamma+1{,}92\delta=0\\ 1+1{,}1\gamma-\delta=0\end{array}\right\}\quad\left.\begin{array}{c}\alpha=-0{,}3\\ \gamma=-0{,}5\\ \beta=-0{,}3\\ \delta=0{,}4\end{array}\right\}$$ ее решение.

Следовательно, итерирующие функции имеют вид:

$$\varphi_1(x,y)=x-0{,}3(x^2+y^2-1)-0{,}3(x^3-y);$$
$$\varphi_2(x,y)=y-0{,}5(x^2+y^2-1)+0{,}4(x^3-y);$$

и по (5) строим итерационный процесс.

4.3. Метод Ньютона для систем двух уравнений

Пусть дана система

$$\begin{cases}F(x,y)=0;\\ G(x,y)=0.\end{cases}$$

Согласно методу Ньютона последовательные приближения типа (5) вычисляются по формулам

$$x_{n+1}=x_n-\frac{\Delta x^{(n)}}{J(x_n,y_n)};\qquad y_{n+1}=y_n-\frac{\Delta y^{(n)}}{J(x_n,y_n)},$$

где

$$\Delta x^{(n)}=\begin{vmatrix}F(x_n,y_n) & F_y'(x_n,y_n)\\ G(x_n,y_n) & G_y'(x_n,y_n)\end{vmatrix};\ \Delta y^{(n)}=\begin{vmatrix}F_x'(x_n,y_n) & F(x_n,y_n)\\ G_x'(x_n,y_n) & G(x_n,y_n)\end{vmatrix};\ n=0,1,2,\ldots$$

и, если Якобиан

$$J(x_n,y_n)=\begin{vmatrix}F_x'(x_n,y_n) & F_y'(x_n,y_n)\\ G_x'(x_n,y_n) & G_y'(x_n,y_n)\end{vmatrix}\neq 0$$

решение будет единственным.

Начальные значения x_0 и y_0 определяются грубо (приближенно – графически или «*прикидкой*»). Данный метод эффективен только при достаточной близости начального приближения к истинному решению системы.

Пример. Найти корни системы

$$\begin{cases} F(x,y) = 2x^3 - y^2 - 1 = 0; \\ G(x,y) = xy^3 - y - 4 = 0. \end{cases}$$

Графическим путем можно найти приближенно $x_0 = 1{,}2$ и $y_0 = 1{,}7$.

$$J(x,y) = \begin{vmatrix} 6x^2 & -2y \\ y^3 & 3xy^2 - 1 \end{vmatrix}.$$

В начальной точке

$$J(1{,}2;1{,}7) = \begin{vmatrix} 8{,}64 & -3{,}40 \\ 4{,}91 & 9{,}40 \end{vmatrix} = 97{,}910.$$

По формулам получаем

$$x_1 = 1{,}2 - \frac{1}{97{,}910}\begin{vmatrix} -0{,}434 & -3{,}40 \\ 0{,}1956 & 9{,}40 \end{vmatrix} = 1{,}2 + 0{,}0349 = 1{,}2349;$$

$$y_1 = 1{,}7 - \frac{1}{97{,}910}\begin{vmatrix} 8{,}64 & -0{,}434 \\ 4{,}91 & 0{,}1956 \end{vmatrix} = 1{,}7 - 0{,}0390 = 1{,}6610.$$

Продолжая процесс вычисления при x_1 и y_1, получим $x_2 = 1{,}2343$; $y_2 = 1{,}6615$ и т.д. до достижения желаемой точности.

4.4. Метод Ньютона для систем *n*-го порядка с *n* неизвестными

Для метода Ньютона функции $F_i = (x_1, x_2, ..., x_n)$ из (1) раскладываются в ряд Тэйлора с отбрасыванием производных второго и выше порядков.

Пусть известен результат предварительной итерации при решении (1) дает результат для $\bar{x} = (a_1, a_2, ..., a_n)$.

Задача сводится к нахождению поправок этого решения: $\Delta x_1, \Delta x_2, ..., \Delta x_n$.

Тогда при очередной итерации решение будет:

$$x_1 = a_1 + \Delta x_1;\ x_2 = a_2 + \Delta x_2;\ \ldots,\ x_n = a_n + \Delta x_n. \qquad (8)$$

Для нахождения Δx_i разложим $F_i(x_1, x_2, ..., x_n)$ в ряд Тейлора:

$$\begin{cases} F_1(x_1,...x_n) \approx F_1(a_1,..., a_n) + \dfrac{\partial F_1}{\partial x_1}\Delta x_1 + ... + \dfrac{\partial F_1}{\partial x_n}\Delta x_n; \\ \qquad\qquad \dots \\ F_n(x_1,...x_n) \approx F_n(a_1,..., a_n) + \dfrac{\partial F_n}{\partial x_1}\Delta x_1 + ... + \dfrac{\partial F_n}{\partial x_n}\Delta x_n. \end{cases} \quad (9)$$

Приравняем правые части согласно (1) к нулю и получим систему линейных уравнений относительно Δx_i:

$$\begin{cases} \dfrac{\partial F_1}{\partial x_1}\Delta x_1 + \dfrac{\partial F_1}{\partial x_2}\Delta x_2 + ... + \dfrac{\partial F_1}{\partial x_n}\Delta x_n = -F_1; \\ \qquad\qquad \dots \\ \dfrac{\partial F_n}{\partial x_1}\Delta x_1 + \dfrac{\partial F_n}{\partial x_2}\Delta x_2 + ... + \dfrac{\partial F_n}{\partial x_n}\Delta x_n = -F_n. \end{cases} \quad (10)$$

Значения F_1, F_2, …, F_n и их производных вычисляются при $x_1=a_1$, $x_2=a_2$, ..., $x_n=a_n$. Расчет ведется с учетом (8) по (9) и (10). Процесс прекращается, когда $\max|\Delta x_i| < \varepsilon$. При этом будет иметь место единственное решение системы, если Якобиан

$$j = \begin{vmatrix} \dfrac{\partial F_1}{\partial x_1} & ... & \dfrac{\partial F_1}{\partial x_n} \\ ... & ... & .... \\ \dfrac{\partial F_n}{\partial x_1} & ... & \dfrac{\partial F_n}{\partial x_n} \end{vmatrix} \neq 0.$$

По сходимости этот метод выше метода простой итерации.

Раздел 5. Аппроксимация функций

5.1. Постановка задачи

При решении многих практических задач часто приходится вычислять значения каких-то функциональных зависимостей $y = f(x)$.

При этом, как правило, имеют преобладающее место две ситуации.

1. Явная зависимость между x и y на $[a, b]$ отсутствует, а имеется только таблица экспериментальных данных $\{x_i, y_i\}$, $i=\overline{1,n}$ и возникает необходимость определения $y = f(x)$ на интервале $[x_i, x_{i/2}] \in [a, b]$. К этой задаче относится также уточнение таблиц экспериментальных данных.

2. Зависимость $y = f(x)$ известна и непрерывна, но настолько сложна, что не пригодна для практических расчетов. Стоит задача упрощения вычисления значений $y = f(x)$ и ее характеристик ($f'(x)$, $\max f(x)$, $\int_a^b f(x)dx$, и т.д.). Поэтому, с точки зрения экономии времени и материальных ресурсов, приходят к необходимости построения какой-то другой функциональной зависимости $y = F(x)$, которая была бы близка к $f(x)$ по основным ее параметрам, но более проста и удобна в реализации при последующих расчетах, т.е. ставится задача о приближении (аппроксимации) в области определения $y = f(x)$. Функцию $y = F(x)$ называют аппроксимирующей.

Основной подход к решению данной задачи заключается в том, что $y=F(x)$ выбирается зависящей от каких-то свободных параметров эксперимента, т.е. $y = F(x) = \varphi(x, c_1, c_2, \ldots, c_n) = \varphi(x, \bar{c})$. Значения вектора $\bar{c}$ выбираются из каких-то условий близости для $f(x)$ и $F(x)$.

В зависимости от способа подбора вектора $\bar{c}$, получают различные виды аппроксимации.

Если приближение строится на каком-то дискретном множестве $\{x_i\}$, $i=\overline{1,n}$, то аппроксимация называется ***точечной***. К ней относится интерполирование, среднеквадратичное приближение (МНК). Если множество $\{x_i\}$ непрерывно, например, в виде отрезка $[a, b]$, аппроксимация называется ***непрерывной*** или интегральной (полиномы Чебышева).

В настоящее время на практике хорошо изучена и широко применяется линейная аппроксимация, при которой $\varphi(x, \bar{c})$ выбирается линейно-зависящий от параметров $\bar{c}$ в виде так называемого ***обобщенного многочлена***:

$$F(x) = \varphi(x, \bar{c}) = c_1\varphi_1(x) + c_2\varphi_2(x) + \ldots + c_n\varphi_n(x) = \sum_{k=1}^{n} c_k \varphi_k(x); \qquad (1)$$

здесь $\varphi_k(x)$ – какая-то выбранная линейно-независимая система базисных функций. В качестве их могут быть, например,

– алгебраическая: $1, x, x^2, ..., x^n, ...$;

– тригонометрическая: $1, \sin(x), \cos(x), \ldots \sin(nx), \cos(nx), \ldots$;

– экспоненциальная: $e^{\alpha_0 x}, e^{\alpha_1 x}, \ldots, e^{\alpha_n x}, \ldots$;

где $\{\alpha_i\}$ – некоторая числовая последовательность попарно различных действительных чисел.

Важным является, чтобы эта система была полной, т.е. обеспечивающей аппроксимацию посредством (1) с заданной точностью на всех интервалах $[a,b]$ определения $y = f(x)$.

Для большинства практических задач наиболее удобна первая из них, представляющая собой в итоге обычные алгебраические многочлены.

5.2. Интерполирование функций

Интерполирование по определению предполагает нахождение промежуточных значений величины заданной таблицей или графиком по некоторым ее значениям. Относительно функциональных зависимостей она является одним из основных видов точечной аппроксимации. Суть интерполирования в данном случае заключается в следующем:

Пусть функция $f(x)$ определена на отрезке $[a, b]$, на котором должна быть обеспечена близость $f(x)$ и $\varphi(x)$. На данном отрезке выбирается система точек, называемых узлами, по правилу:

$$a \le x_0 < x_1 < x_2 < \ldots < x_n \le \mathrm{b}.$$

Их число равно количеству параметров $\bar{c}$ в (1).

Известны значения функции $f(x)$ в этих узлах, т.е.

$$y_i = f(x_i),\ i = \overline{0,n}.$$

Задача интерполирования сводится к подбору многочлена согласно (1) вида:

$$P(x) = c_0 x^n + c_1 x^{n-1} + \ldots + c_{n-1} x + c_n = \sum_{k=0}^{n} c_k x^{n-k}\ , \qquad (2)$$

с действительными коэффициентами c_k, найденными по правилу:

$$\sum_{k=0}^{n} c_k x_i^{n-k} = f(x_i) = y_i\ , \qquad i = \overline{0,n}. \qquad (3)$$

Такой многочлен называют интерполяционным многочленом.

Процедуру (2) с использованием условий (3) называют ***глобальной интерполяцией.*** Если же многочлен (2) строится только для отдельных участков отрезка $[a, b]$ (области определения $f(x)$), т.е. для m интерполяционных узлов, где $m < n$, то интерполяцию называют ***локальной.***

Матрица системы (3) и ее определитель имеют следующий вид:

$$G = \begin{vmatrix} x_0^n & x_0^{n-1},\ldots & 1 \\ x_1^n & x_1^{n-1},\ldots & 1 \\ x_n^n & x_n^{n-1},\ldots & 1 \end{vmatrix}; \qquad |G| \neq 0, \qquad (4)$$

так как узлы выбранной системы точек различны. Следовательно, система (3) имеет единственное решение, т.е. коэффициенты многочлена (2) находятся однозначно.

Заметим, что условие (3) обеспечивает близость $f(x)$ и $F(x)$, по любой технологии ее получения, т.е. в узлах интерполяции $f(x)$ и $F(x)$ совпадают по их значениям.

Если (2) и (3) используются для вычисления значений функции для случая $x < x_0$ и $x > x_n$ такое приближение называется ***экстраполяцией***.

5.3. Типовые виды локальной интерполяции

5.3.1. Линейная интерполяция

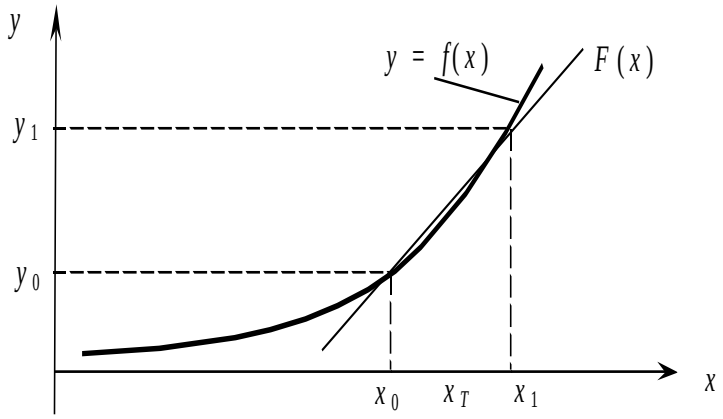
Линейная интерполяция состоит в том, что заданные точки таблицы (x_i, y_i), ($i=\overline{0,n}$) соединяются прямыми линиями и исходная функция $f(x)$ приближается на интервале $[a, b]$ к ломаной с вершинами в узлах интерполяции. В общем случае частичные интервалы $[x_{i-1}, x_i] \in [a, b]$ различны. Для каждого отрезка ломаной можно написать уравнение прямой, проходящей через точки (x_{i-1}, y_{i-1}) и (x_i, y_i). В частности, для i-го интервала в виде:

$$\frac{y - y_{i-1}}{y_i - y_{i-1}} = \frac{x - x_{i-1}}{x_i - x_{i-1}}.$$

Тогда рабочую формулу можно записать:

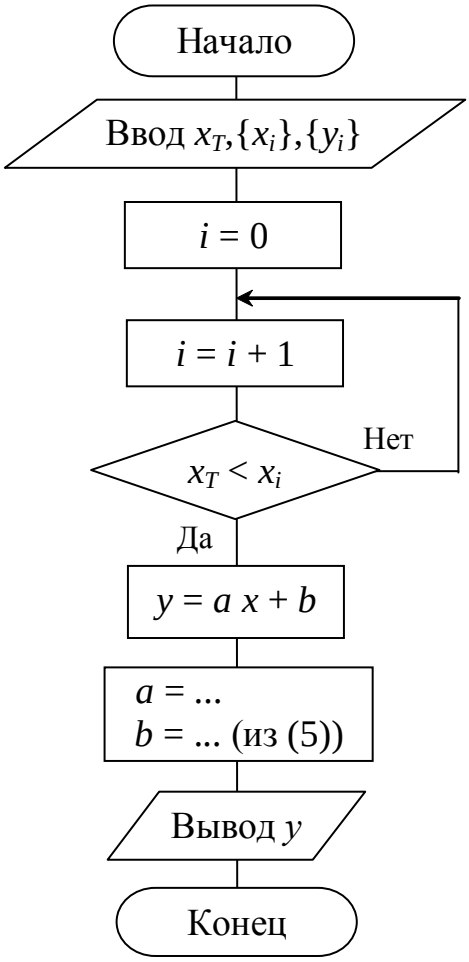
$$y = a_i x_T + b_i, \;\; x_{i-1} \le x_T \le x_i, \qquad (5)$$

где $a_i = \dfrac{y_i - y_{i-1}}{x_i - x_{i-1}}, b_i = y_{i-1} - a_i x_{i-1}, \; i=\overline{1,n}$



Из графической иллюстрации видно, что для реализации (5) сначала нужно определить интервал, в который попадает значение x_T, а затем воспользоваться его границами.

Блок-схема данного алгоритма:



Теоретическая погрешность $R(x) = f(x) - F(x) \neq 0$ в точках, отличных от узлов.

$$R_1(x) = \frac{M_2}{8} h^2, \text{ где } M_2 = \max\left|f''(x)\right|, \; x \in [x_{i-1}, x_i].$$

5.3.2. Квадратичная (параболическая) интерполяция

В данном случае в качестве интерполяционного многочлена используется квадратный трехчлен на отрезке $[x_{i-1}, x_{i+1}] \in [a, b]$ в виде:

$$y = a_i x^2 + b_i x + c_i, \tag{6}$$

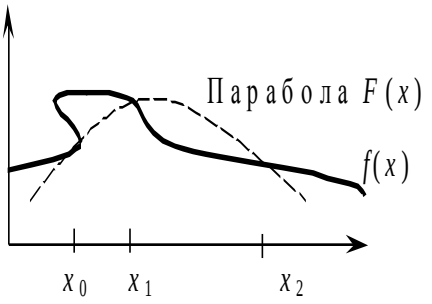
$$x_{i-1} \le x_T \le x_{i+1}.$$

Для определения коэффициентов a_i, b_i, c_i составляется система из трех уравнений согласно условиям (3), а именно:

$$\begin{cases} a_i x_{i-1}^2 + b_i x_{i-1} + c_i = y_{i-1}; \\ a_i x_i^2 + b_i x_i + c_i = y_i; \\ a_i x_{i+1}^2 + b_i x_{i+1} + c_i = y_{i+1}. \end{cases} \tag{7}$$

Алгоритм вычисления аналогичен предыдущему, только вместо соотношений (5) используется соотношение (6) с учетом решения (7). Очевидно, что для $x_T \in [x_0, x_n]$ используются три ближайшие точки.

Графическая иллюстрация метода



Теоретическая погрешность вне узлов интерполяции

$$R(x) = (x - x_0) \cdot (x - x_1) \cdot (x - x_2) \frac{f'''(x)}{6}.$$

5.4. Типовые виды глобальной интерполяции

5.4.1. Интерполяция общего вида

В данном случае интерполяционный многочлен ищется в виде (2) для всего интервала области определения x_T, т.е. для $[x_0, x_n]$ в виде:

$$\varphi(x) = a_0 + a_1 x + a_2 x^2 + ... + a_n x^n . \qquad (8)$$

Для получения коэффициентов a_i составляется система уравнений (3)

$$\begin{cases} a_0 + a_1 x_0 + ... + a_n x_0^n = y_0; \\ a_0 + a_1 x_1 + ... + a_n x_1^n = y_1; \\ \quad\quad\quad \dots \\ a_0 + a_1 x_n + ... + a_n x_n^n = y_n. \end{cases} \qquad (9)$$

Известно, что если $x_i \neq x_j$ при $i \neq j$ система имеет единственное решение. Для решения (9) можно использовать методы, рассмотренные ранее для СЛАУ. Прямое решение системы (9) и получение $F(x)$ в виде (8) выгодно, когда производится много вычислений по одной и той же таблице. Для разового вычисления $y = f(x_T)$ предложены другие алгоритмы, при которых не нужно находить параметры вектора $\bar{a}$, а интерполяционные многочлены записываются через значения таблиц $\{x_i, y_i\}$, $i = \overline{0,n}$. Это интерполяционные многочлены Лагранжа и Ньютона.

5.4.2. Интерполяционный многочлен Лагранжа

1. Формула Лагранжа для произвольной системы интерполяционных узлов

Многочлен Лагранжа ищется в виде линейной комбинации из значений $f(x)$ в узлах интерполяции и каких-то специально построенных из системы узлов интерполяции многочленов n-ой степени в виде:

$$L_n(x) = \sum_{i=0}^{n} y_i l_i(x) = y_0 l_0(x) + y_1 l_1(x) + ... + y_n l_n(x). \qquad (10)$$

Итак, сначала строится вспомогательный многочлен $(n+1)$-й степени

$$\omega(x) = (x - x_0)(x - x_1)...(x - x_n) \qquad (11)$$

и многочлен n-й степени

$$\varphi_i(x) = \frac{\omega(x)}{x - x_i} = (x - x_0)...(x - x_{i-1})(x - x_{i+1})...(x - x_n). \qquad (12)$$

Очевидно, что многочлен (11) обращается в нуль в узлах интерполяции x_i, т.е. $\omega(x_i) = 0$, $i = \overline{0,n}$, а многочлен (12) $\varphi_i(x)$ обращается в ноль во всех узлах, кроме узла x_i, т.е.:

$$\varphi_i(x_j) = \begin{cases} 0, & j \neq i; \\ (x_j - x_0)...(x_j - x_{i-1})(x_j - x_{i+1})...(x_i - x_n) \neq 0, & j = i. \end{cases} \qquad (13)$$

Из равенств (12) и (13) следует, что построенный новый многочлен

$$l_j(x) = \frac{\omega(x)}{(x - x_i)(x_j - x_0)...(x_j - x_{i-1})(x_j - x_{i+1})...(x_j - x_n)}$$

принимает нулевое значение во всех узлах, кроме j-го, а в узле x_j его значение будет равно ***единице***, т.е.

$$l_j(x_i) = \begin{cases} 0, & i \neq j; \\ 1, & i = j; \end{cases} \qquad i, j = \overline{0,n}.$$

Тогда j-й многочлен из (10) $l_j(x_i) \cdot y_j$ будет принимать нулевые значения во всех узлах, кроме x_j, и значение y_j в узле x_j, т.е.

$$l_j(x_i) \cdot y_j = \begin{cases} 0, & i \neq j; \\ y_j, & i = j; \end{cases} \qquad i, j = \overline{0,n}$$

Согласно (10) составим многочлен

$$L_n(x) = \sum_{j=0}^{n} y_j l_j(x) = \sum_{j=0}^{n} y_j \frac{\omega(x)}{(x - x_i)\omega'(x_j)},$$

где $\omega'(x_j) = (x_j - x_0)...(x_j - x_{j-1})(x_j - x_{j+1})...(x_j - x_n)$.

Или в более свернутой форме

$$L_n(x) = \sum_{j=0}^{n} y_j \prod_{\substack{i=0 \\ i \neq j}}^{n} \frac{x - x_i}{x_j - x_i} \ ; \qquad (14)$$

Его погрешность $R_n(x) = f(x) - L_n(x) = \dfrac{\left| f^{(n+1)}(\xi) \right|}{(n+1)!} \cdot \omega(x)$, где $\xi \in [a, b]$.

В отличие от полинома (8) здесь не требуется предварительного определения всех его коэффициентов. Однако, для каждого x_T нужно рассчитывать полином Лагранжа по технологии (14). Поэтому объем вычислений фактически не меньше, чем при технологии расчета (9).

На практике, если необходим повторный расчет при различных x_T в большем количестве, то схема (8) будет предпочтительнее. Однако полином Лагранжа широко используется при реализации других численных методов. Следует подчеркнуть, что при $n = 1$ – это линейная, а при $n = 2$ – квадратичная интерполяция.

2. Полином Лагранжа на системе равноотстоящих интерполяционных узлов

Величина $h = x_{i+1} - x_i = \text{const}$. Тогда произвольный узел $x_i = x_0 + i \cdot h, \ i = \overline{0, n}$. Введем переменную $t = (x - x_0)/ h$. Тогда

$$x - x_i = x_0 + th - x_0 - ih = (t - i)h \ . \qquad (15)$$

Подставив разности (15) в равенство (11) получим:

$$\omega(x) = (x - x_0)(x - x_1)...(x - x_n) = th(t-1)h...(t-n)h = t(t-1)...(t-n)h^{n+1}.$$

Далее, так как

$$x_j - x_i = (x_0 + jh) - (x_0 + ih) = (j - i)h,$$

то с учетом (15) формула Лагранжа примет вид:

$$L_n(x) = \sum_{j=0}^{n} y_j \prod_{\substack{i=0 \\ i \neq j}}^{n} \frac{t - i}{j - i}, \qquad (16)$$

где $t = (x - x_0)/h$.

Его погрешность $R_n(x) = h^{n+1} t(t-1)...(t-n) \dfrac{f^{(n+1)}(\xi)}{(n+1)!}$.

5.4.3. Интерполяционный многочлен Ньютона

Как и в предыдущем случае строится многочлен (2) с соблюдением условий (3) специфического вида. Интерполяционный многочлен Ньютона ищется в следующем виде:

$$N(x) = a_0 + a_1(x - x_0) + a_2(x - x_0)(x - x_1) + \ldots + a_n(x - x_0)(x - x_1)\ldots(x - x_{n-1}). \qquad (17)$$

Как и в случае (8) для получения рабочей формулы Ньютона необходимо определить значения коэффициентов a_i. В отличие от технологии расчета (9) для построения интерполяционного многочлена Ньютона вводится рабочий аппарат в виде, так называемых, ***конечных разностей*** для системы равноотстоящих интерполяционных узлов и в виде ***разностных отношений*** (разделенные разности) для произвольной системы узлов.

Пусть заданны равноотстоящие узлы $x_k = x_0 + kh$, $h = x_{i+1} - x_i = \text{const} > 0$. Значения $f(x)$ в них обозначим $f(x_k) = f_k = y_k$, $k = \overline{0,n}$

Конечными разностями первого порядка принято называть величины

$$\Delta f(x_i) = \Delta f_i = f_{i+1} - f_i\,; \qquad i = \overline{0,n}.$$

Конечные разности второго порядка определяются равенствами

$$\Delta^2 f_i = \Delta(\Delta f_i) = \Delta f_{i+1} - \Delta f_i,\ i = \overline{0,n}.$$

Конечные разности (k+1)-го порядка определяются через разности k-го порядка

$$\Delta^{k+1} f_i = \Delta^k f_{i+1} - \Delta^k f_i,\ i = \overline{0,n};\ k = \overline{1,n}. \qquad (18)$$

Конечные разности, как правило, вычисляются по следующей схеме:

Таблица 1

i	f_i	Δf_i	$\Delta^2 f_i$	$\Delta^3 f_i$	…
0	f_0				
		Δf_0			
1	f_1		$\Delta^2 f_0$		
		Δf_1		$\Delta^3 f_0$	
2	f_2		$\Delta^2 f_1$		
		Δf_2		$\Delta^3 f_1$	
3	f_3		$\Delta^2 f_2$		
		Δf_3			
4	f_4				
…	…				

Каждая последующая конечная разность получается путем вычитания в предыдущей колонке верхней строки из нижней строки. Последняя колонка $\Delta^k f_i$ будет равна нулю. Заметим, что конечные разности можно выразить непосредственно через значения функций. Так для i-го узла рабочая формула имеет вид:

$$\Delta^k f_i = f_{k+i} - k f_{k+i-1} + \frac{k(k-1)}{2!} f_{k+i-2} + \ldots + (-1)^k f_i\,; \quad i = \overline{0,n};\ k = 1,2,\ldots \qquad (19)$$

Разностными отношениями (разделенными разностями) первого порядка называются величины

$$f(x_0, x_1) = \frac{f(x_1) - f(x_0)}{x_1 - x_0}; \quad f(x_1, x_2) = \frac{f(x_2) - f(x_1)}{x_2 - x_1}; \ \ldots$$

Здесь x_i – произвольные узлы с соблюдением приоритетности по величине.

По этим соотношениям составляются разностные отношения второго порядка:

$$f(x_0, x_1, x_2) = \frac{f(x_1, x_2) - f(x_0, x_1)}{x_2 - x_0}; \quad f(x_1, x_2, x_3) = \frac{f(x_2, x_3) - f(x_1, x_2)}{x_3 - x_1}; \quad \ldots$$

Разделенные разности порядка (k+1), k = 1,2, ... определяются при помощи разделенных разностей предыдущего порядка k по формуле:

$$f(x_0, x_1, \ldots, x_{k+1}) = \frac{f(x_1, x_2, \ldots, x_{k+1}) - f(x_0, x_1, \ldots, x_k)}{x_{k+1} - x_0} \ . \qquad (20)$$

Разностные отношения вычисляются по следующей схеме:

Таблица 2

i	x_i	f_i	$f(x_i,x_{i+1})$	$f(x_i,x_{i+1},x_{i+2})$	…
0	x_0	f_0			
			$f(x_0,x_1)$		
1	x_1	f_1		$f(x_0,x_1,x_2)$	
			$f(x_1,x_2)$		
2	x_2	f_2		$f(x_1,x_2,x_3)$	
			$f(x_2,x_3)$		
3	x_3	f_3		$f(x_2,x_3,x_4)$	
...	…	…	…	…	…

Для равноотстоящих узлов $x_k = x_0 + kh$ ($k = \overline{0,n}$) имеет место соотношение между разделенными разностями и конечными разностями

$$f(x_0, x_1, \ldots, x_k) = \frac{\Delta^k f_0}{h^k k!}; \ k = 0,1,2, \ldots \qquad (21)$$

Конечная разность и разделенная разность порядка n от многочлена степени (n) равны постоянной величине, и, следовательно, они для более высокого порядка равны ***нулю***.

1. Интерполяционный многочлен Ньютона для системы равноотстоящих узлов

В случае равностоящих узлов имеется много различных формул, построение которых зависит от расположения точки интерполирования x_T по отношению к узлам интерполирования.

Пусть функция $f(x)$ задана таблицей значений $f_k = f(x_k) = y_k$ в узлах $x_k = x_0 + kh$ ($k = \overline{0,n}$), $h = x_{k+1} - x_k = \text{const}$.

На основании условий (3) и аппарата конечных разностей для определения коэффициентов для искомого многочлена (17) получена формула

$$a_k = \frac{\Delta^k y_0}{k! h_k}, \ k = \overline{0,n}; \ \text{при условии, что } \Delta^0 = 1; \ 0! = 1. \qquad (22)$$

Подставляя (22) в (17) получим формулу Ньютона для интерполирования в начале таблицы

$$N(x) = y_0 + \frac{\Delta y_0}{1!h}(x - x_0) + \frac{\Delta^2 y_0}{2!h^2}(x - x_0)(x - x_1) + \\ + ... + \frac{\Delta^n y_0}{n!h^n}(x - x_0)(x - x_1)...(x - x_{n-1}). \qquad (23)$$

При этом конечные разности определяются или по схеме (табл.1) или по формуле для произвольного узла (19).

Для практического удобства формула (23) часто записывают в другом виде. Вводится новая переменная $t = (x - x_0)/h$. Тогда имеем:

$$x = x_0 + kh; \quad \frac{x - x_1}{h} = (x - x_0 - h)/h = t - 1;$$

$$\frac{x - x_2}{h} = t - 2, \ \ldots, \ \frac{x - x_{n-1}}{h} = t - n + 1;$$

и (23) примет вид

$$N(x_0 + th) = y_0 + t\Delta y_0 + \frac{t(t-1)}{2!}\Delta^2 y_0 + ... + \frac{t(t-1)...(t-n+1)}{n!}\Delta^n y_0. \qquad (24)$$

Выражение (24) может аппроксимировать $y = f(x)$ на всем отрезке $[x_0, x_n]$. Однако, с точки зрения повышения точности расчетов, и уменьшения числа членов в (24), рекомендуется ограничиться случаем $t < 1$, т.е. использовать формулу (24) для интервала $x_0 \le x \le x_1$. Для других значений аргумента, например, для $x_1 \le x \le x_2$, вместо x_0 лучше взять значение x_1. Тогда (24) можно записать в виде

$$N(x_i + th) = y_i + t\Delta y_i + \frac{t(t-1)}{2!}\Delta^2 y_i + ... + \frac{t(t-1)...(t-n+1)}{n!}\Delta^n y_i; \ i = 0,1,\ldots \qquad (25)$$

Выражение (25) называется ***первым*** интерполяционным многочленом Ньютона для интерполирования вперед. Он используется для вычисления значений функций в точках левой половины рассматриваемого отрезка. Это объясняется тем, что разности $\Delta^k y_i$ вычисляются через значение функции y_i, y_{i+1}, ..., y_{i+k}, причем $i + k \le n$. Поэтому при больших значениях i нельзя вычислить значения разностей высших порядков ($k \le n - i$). Например, при $i = n - 3$ в (25) можно учесть только Δy, $\Delta^2 y$, $\Delta^3 y$.

Для правой половины отрезка разности рекомендуется вычислять ***справа налево***. В этом случае $t = (x - x_n) / h$, т.е. $t < 0$ и (25) можно получить в виде

$$N(x_n + th) = y_n + t\Delta y_{n-1} + \frac{t(t+1)}{2!}\Delta^2 y_{n-2} + ... + \frac{t(t+1)...(t+n-1)}{n!}\Delta^n y_0. \qquad (26)$$

Полученная формула называется ***вторым*** интерполяционным многочленом Ньютона для интерполирования назад. Для интерполирования в средине отрез-

ка можно использовать интерпретации многочлена Ньютона – это многочлены Стирлинга, Гаусса, Бесселя.

Погрешность метода Ньютона:

$$R_N(x) = f(x) - N_n(x) = \frac{t(t-1)...(t-n)}{(n+1)!} f^{(n+1)}(\xi) \cdot h^{n+1},$$

при $t = \frac{x - x_0}{h}$, ξ – принадлежит отрезку.

Рассмотрим пример. Вычислить значение функции $y = f(x)$, заданной таблицей в точках $x = 0{,}1$ и $x = 0{,}9$. Строим таблицу 1 для конечных разностей

x	$y = f(x)$	Δy	$\Delta^2 y$	$\Delta^3 y$	$\Delta^4 y$	$\Delta^5 y$
0	1,2715					
		1,1937				
0,2	2,4652		–0,0146			
		1,1791		0,0007		
0,4	3,6443		–0,0139		–0,0001	
		1,1652		0,0006		0,0000
0,6	4,8095		–0,0133		–0,0001	
		1,1919		0,0005		
0,8	5,9614		–0,0128			
		1,1391				
1	7,1005					

Используя для расчета верхние значения конечных разностей, получим при $x = 0{,}1$ значение $t = (x - x_0)/h = (0{,}1 - 0) / 0{,}2 = 0{,}5$. По формуле (24) получим

$$f(0{,}1) \approx N(0{,}1) = 1{,}2715 + 0{,}5 \times$$

$$\times 1{,}1937 + \frac{0{,}5(0{,}5-1)}{2!}(-0{,}0146) + \frac{0{,}5(0{,}5-1)(0{,}5-2)}{3!}0{,}0007 +$$

$$+ \frac{0{,}5(0{,}5-1)(0{,}5-2)(0{,}5-3)}{4!}(-0{,}0001) = 1{,}8702 .$$

По формуле линейной интерполяции $f(0{,}1) \approx 1{,}8684$; $\Delta = \{0{,}0018\}$.

Значение функции в точке $x = 0{,}9$ вычислим по формуле (26). В данном случае $t = (x - x_n) / h = (0{,}9 - 1) / 0{,}2 = -0{,}5$. Используя нижние значения конечных разностей, получим

$$f(0{,}9) \approx N(0{,}9) = 7{,}1005 - 0{,}5 \cdot 1{,}1391 -$$

$$- \frac{0{,}5(-0{,}5+1)}{2!}(-0{,}0128) - \frac{0{,}5(-0{,}5+1)(-0{,}5+2)}{3!}0{,}0005 -$$

$$- \frac{0{,}5(-0{,}5+1)(-0{,}5+2)(-0{,}5+3)}{4!}(-0{,}0001) = 6{,}5325 .$$

Если считать по (24) $f(0{,}9) = 6{,}532522641$.

Линейная интерполяция $f(0{,}9) = 6{,}53095$; $\Delta = \{0{,}00155\}$.

2. Интерполяционный многочлен Ньютона для системы произвольно расположенных узлов

Данный многочлен строится с помощью аппарата разделенных разностей (табл. 2) в виде (17) на основании имеющего места соотношения (21)

$$N_n(x) = f(x_0) + (x–x_0)\, f(x_0,x_1) + (x–x_0)(x–x_1)\, f(x_0, x_1, x_2) +\ldots+$$
$$+ (x–x_0)(x–x_1)\ldots(x–x_{n–1})\, f(x_0, x_1,\ldots, x_n) \qquad (27)$$

Здесь, как и раньше, $N_n(x_k) = f(x_k)$, $(k = 0, 1, 2,\ldots,n)$.

Остаточный член

$$R_n(x) = f(x) – N_n(x) = f(x, x_0, x_1,\ldots, x_n)\,(x–x_0)(x–x_1)\ldots\,(x–x_n).$$

При $n = 1$ – это линейная интерполяция, при $n = 2$ – квадратичная.

Замечания

1. Разные способы построения многочленов Лагранжа и Ньютона дают тождественные рабочие формулы при заданной таблице $f(x)$. Это следует из единственности интерполяционного многочлена заданной степени на упорядоченной системе узлов.

2. Повышение точности интерполирования предположительно проводить за счет увеличения числа узлов n и соответственно степени полинома $P_n(x)$. Однако при таком подходе увеличивается погрешность из-за роста $| f^{(n)}(x) |$ и, кроме того, увеличивается вычислительная погрешность.

Эти соображения приводят к другому способу приближения функций с помощью сплайнов (будет рассмотрено дальше).

3. Повышение точности интерполирования осуществляется и посредством специального расположения узлов интерполяции на рассматриваемом отрезке $[a, b]$ области определения функции $f(x)$. Известно, что если сконцентрировать узлы x_i вблизи одного конца отрезка $[a, b]$, то погрешность $R_n(x)$ при длине отрезка $l = b – a > 1$ будет велика в точках x_i близких к другому концу. Поэтому всегда возникает задача о наиболее рациональном выборе x_i (при заданном числе узлов n).

Эта задача была решена Чебышевым, т.е. оптимальный выбор узлов нужно производить по формуле:

$$x_i = \frac{b+a}{2} + \frac{b-a}{2}\xi_i\,,$$

где $\xi_i = -\cos\dfrac{2i+1}{2n+2}\pi$ $(i = 0,1,2, ..., n)$ – есть нули полинома Чебышева $T_{n+1}(x)$.

Пример. Найти значение $y = f(x)$ при $x = 0{,}4$ заданной таблично:

i	0	1	2	3
x_i	0	0,1	0,3	0,5
y_i	–0,5	0	0,2	1

3. Локальная интерполяция

3.1. Линейная интерполяция

$$y = a_i x + b_i$$

$a_i = (y_i - y_{i-1})/(x_i - x_{i-1})$, $b_i = y_{i-1} - a_i x_{i-1}$.

$x_t = 0{,}4$; $0{,}3 \le x_t \le 0{,}5$; $x_{i-1} = 0{,}3$; $x_i = 0{,}5$

$y_{i-1} = 0{,}2$; $y_i = 1$

$a_3 = (1 - 0{,}2)/(0{,}5 - 0{,}3) = 0{,}8/0{,}2 = 4$; $b_3 = -1$;

$y = 4x - 1$, при $x = 0{,}4$; $y = 4 \cdot 0{,}4 - 1 = 0{,}6$. (28)

3.2. Квадратичная интерполяция

$$y = a_i x^2 + b_i x + c_i$$

Выбираем три ближайшие точки к $x_t = 0{,}4$

$x_{i-1} = 0{,}1$; $x_i = 0{,}3$; $x_{i+1} = 0{,}5$.

$y_{i-1} = 0$; $y_i = 0{,}2$; $y_{i+1} = 1$.

$$\left.\begin{array}{c} a_i x_{i-1}^2 + b_i x_{i-1} + c_i = y_{i-1} \\ a_i x_i^2 + b_i x_i + c_i = y_i \\ a_i x_{i+1}^2 + b_i x_{i+1} + c_i = y_{i+1} \end{array}\right\} \Rightarrow \left\{\begin{array}{l} 0{,}01 a_i + 0{,}1 b_i + c_i = 0; \\ 0{,}09 a_i + 0{,}3 b_i + c_i = 0{,}2; \\ 0{,}25 a_i + 0{,}5 b_i + c_i = 1; \end{array}\right.$$

$$A = \begin{vmatrix} 0{,}01 & 0{,}1 & 1 \\ 0{,}09 & 0{,}3 & 1 \\ 0{,}25 & 0{,}5 & 1 \end{vmatrix}; \quad \overline{B} = \begin{vmatrix} 0 \\ 0{,}2 \\ 1 \end{vmatrix}; \quad \overline{X} = \begin{Bmatrix} a \\ b \\ c \end{Bmatrix} = A^{-1} \overline{B}.$$

Найдем $A^{-1} = \begin{vmatrix} \frac{75}{6} & \frac{-75}{3} & \frac{25}{2} \\ -10 & \frac{45}{3} & -5 \\ \frac{15}{8} & -\frac{10}{8} & \frac{3}{8} \end{vmatrix}$;

$$\overline{X} = \begin{vmatrix} a \\ b \\ c \end{vmatrix} = \begin{vmatrix} \frac{75}{6} & -\frac{75}{3} & \frac{25}{2} \\ -10 & \frac{45}{3} & -5 \\ \frac{15}{8} & -\frac{10}{8} & \frac{3}{8} \end{vmatrix} \cdot \begin{vmatrix} 0 \\ 0{,}2 \\ 1 \end{vmatrix};$$

$a = 0 - \frac{75}{3} * \frac{1}{5} + \frac{25}{2} = 7{,}5$; $b = -2$; $c = 0{,}125$;

$y = 7{,}5x^2 - 2x + 0{,}125$; при $x = 0{,}4$; $y = 0{,}525$. (29)

4. Глобальная интерполяция

4.1. Интерполяционный многочлен Лагранжа

$$L(x) = \sum_{i=0}^{n} y_i \prod_{\substack{j=0 \\ j\neq i}}^{n} \frac{x - x_j}{x_i - x_j};$$

$$L(x) = \sum_{i=0}^{3} y_i \prod_{\substack{j=0 \\ j\neq i}}^{3} \frac{x - x_j}{x_i - x_j} = y_0 \frac{(x-x_1)(x-x_2)(x-x_3)}{(x_0-x_1)(x_0-x_2)(x_0-x_3)} + y_1 \frac{(x-x_0)(x-x_2)(x-x_3)}{(x_1-x_0)(x_1-x_2)(x_1-x_3)} +$$

$$+ y_2 \frac{(x-x_1)(x-x_0)(x-x_3)}{(x_2-x_0)(x_2-x_1)(x_2-x_3)} + y_3 \frac{(x-x_0)(x-x_1)(x-x_2)}{(x_3-x_0)(x_3-x_1)(x_3-x_2)} =$$

$$= \frac{125}{3} x^3 - 30x^2 + \frac{91}{12} x - 0{,}5;$$

при $x = 0{,}4$; $y \approx L(x) = 0{,}3999$.

Найдем выражение для полинома Лагранжа для данной таблицы при n=1 и для $x_T = 0{,}4$;

$$L(x) = \sum_{i=0}^{1} y_i \prod_{\substack{j=0 \\ j\neq i}}^{1} \frac{x - x_j}{x_i - x_j} = y_0 \frac{x - x_1}{x_0 - x_1} + y_1 \frac{(x - x_0)}{(x_1 - x_0)} = 0{,}2 \frac{x - 0{,}5}{0{,}3 - 0{,}5} + 1 \frac{(x - 0{,}3)}{(0{,}5 - 0{,}3)} =$$

$$= 5x - 1{,}5 - x + 0{,}5 = 4x - 1;$$

это соответствует (28).

Для $n = 2$ при $x_T = 0{,}4$ $y \approx L(x) = \sum_{i=0}^{2} y_i \prod_{\substack{j=0 \\ i\neq j}}^{2} \frac{x - x_j}{x_i - x_j}$;

Для рассматриваемого интервала $[x_1, x_3]$, берем $x_0 = 0{,}1$; $x_1 = 0{,}3$; $x_2 = 0{,}5$; $y_0 = 0$; $y_1 = 0{,}2$; $y_2 = 1$. Тогда

$$y \approx L(x) = 0{,}2 \cdot \frac{(x - 0{,}1)(x - 0{,}5)}{(0{,}3 - 0{,}1)(0{,}3 - 0{,}5)} + 1 \cdot \frac{(x - 0{,}1)(x - 0{,}3)}{(0{,}5 - 0{,}1)(0{,}5 - 0{,}3)} = 7{,}5x^2 - 2x + 0{,}125;$$

что соответствует (29).

Алгоритм расчета интерполяционного многочлена Лагранжа, реализованный в виде функции *PL* с параметрами:

x_T – значение текущей точки;

$\vec{x}$, $\vec{y}$ – одномерные массивы известных значений x и $f(x)$;

n – размер массивов $\vec{x}$, $\vec{y}$;

представлен на рис. 5.1.

В схеме введены следующие обозначения:

p – значение накапливаемой суммы, результат которой равен $L(x_T)$;

e – значение очередного члена произведения;

Результатом функции *PL* является значение p.

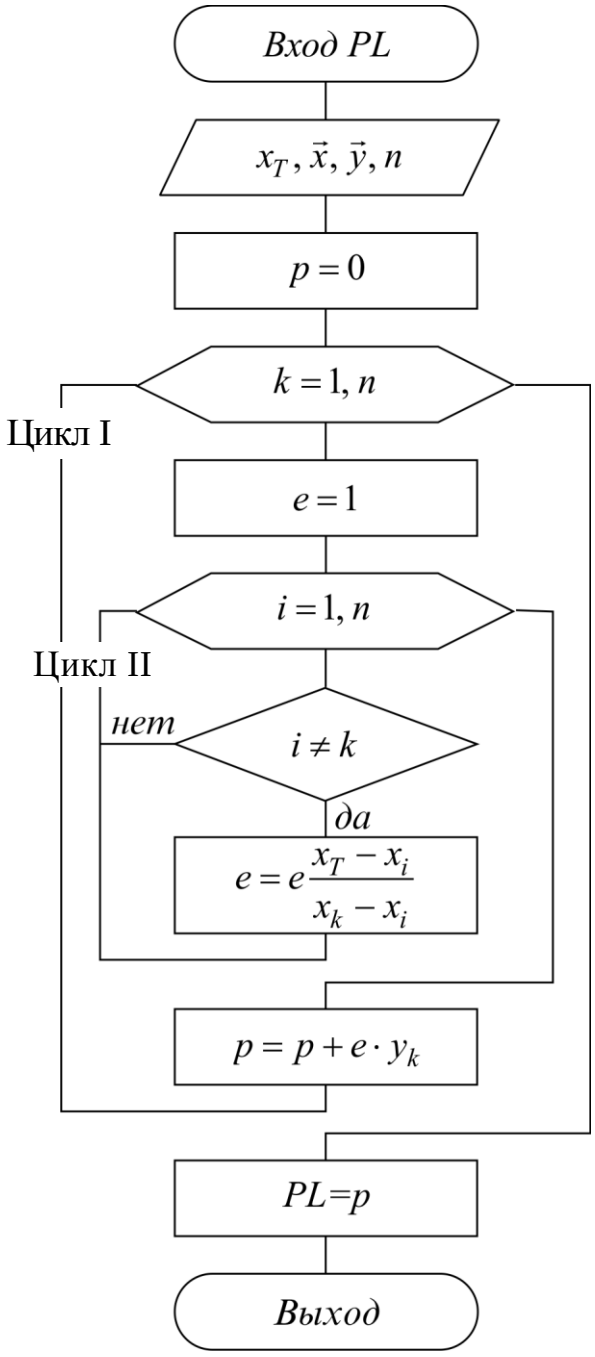


Рис. 5.1. Схема расчета интерполяционного многочлена Лагранжа

4.2. Интерполяционный многочлен Ньютона

Имеем случай неравностоящих узлов, $n = 3$;

$$N_3(x) = f(x_0) + (x-x_0)f(x_0,x_1) + (x-x_0)(x-x_1)f(x_0,x_1,x_2) + (x-x_0)(x-x_1)(x-x_2)f(x_0,x_1,x_2,x_3).$$

По схеме таблицы 2 находим раздельные разности

$$f(x_0,x_1) = \frac{f(x_1) - f(x_0)}{x_1 - x_0} = \frac{0-(-0{,}5)}{0{,}1} = 5;$$

$$f(x_1,x_2)=\frac{f(x_2)-f(x_1)}{x_2-x_1}=\frac{0,2-0}{0,3-0,1}=1;$$

$$f(x_2,x_3)=\frac{f(x_3)-f(x_2)}{x_3-x_2}=\frac{1-0,2}{0,5-0,3}=4\ ;$$

$$f(x_0,x_1,x_2)=\frac{f(x_1,x_2)-f(x_0,x_1)}{x_2-x_0}=\frac{1-5}{0,3-0}=-\frac{40}{3};$$

$$f(x_1,x_2,x_3)=\frac{f(x_2,x_3)-f(x_1,x_2)}{x_3-x_1}=\frac{4-1}{0,5-0,1}=\frac{15}{2};$$

$$f(x_0,x_1,x_2,x_3)=\frac{f(x_1,x_2,x_3)-f(x_0,x_1,x_2)}{x_3-x_0}=\frac{\frac{15}{2}+\frac{40}{3}}{0,5}=\frac{125}{3}.$$

Результаты расчетов поместим в таблицу:

n	x_n	f_n	$f(x_n, x_{n+1})$	$f(x_n, x_{n+1}, x_{n+2})$	$f(x_n, x_{n+1}, x_{n+2}, x_{n+3})$
0	0	–0,5			
1	0,1	0	5	–40/3	125/3
2	0,3	0,2	1	15/2	
3	0,5	1	4		

Используя первые в столбцах разделенные разности, получим

$$N_3(x) = -0,5 + (x-0)\cdot 5 + (x-0)(x-0,1)(-\frac{40}{3}) + (x-0)(x-0,1)(x-0,3)\frac{125}{3} =$$

$$= \frac{125}{3}x^3 - 30x^2 + \frac{91}{12}x - 0,5\ . \qquad (30)$$

Аналогично расчету по Лагранжу.

Напомним, что расчеты интерполяционного многочлена Ньютона выполняются по формуле

$$N_{n-1}(x_T) = y_1 + \sum_{k=1}^{n-1}(x_T - x_1)(x_T - x_2)...(x_T - x_k)\Delta_1^k\ ,$$

где x_T – текущая точка, в которой надо вычислить значение многочлена;

Δ_1^k – разделенные разности порядка k, которые вычисляются по следующим рекуррентным формулам:

$$\Delta_i^1 = \frac{y_i - y_{i+1}}{x_i - x_{i+1}}, \qquad i = 1,..., (n-1);$$

$$\Delta_i^2 = \frac{\Delta_i^1 - \Delta_{i+1}^1}{x_i - x_{i+2}}, \qquad i = 1,..., (n-2);$$

$$\Delta_i^k = \frac{\Delta_i^{k-1} - \Delta_{i+1}^{k-1}}{x_i - x_{i+k}},\ i = 1,..., (n-k).$$

Схема алгоритма расчета многочлена Ньютона, реализованная в виде функции *PN* с параметрами, значения которых аналогичны рассмотренной ранее функции *PL*, представлена на рис. 5.2.

Результатом функции *PN* является значение *N*.

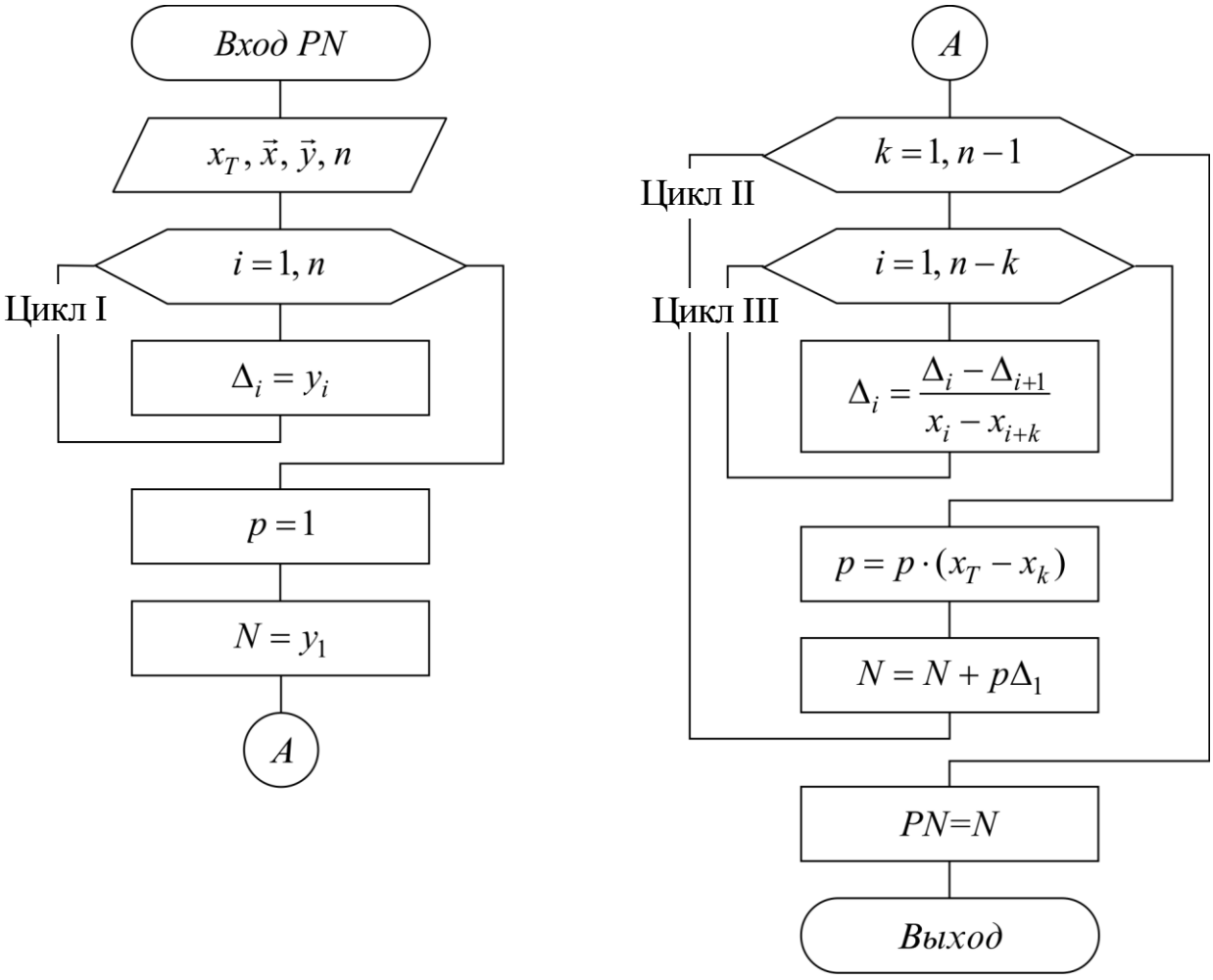
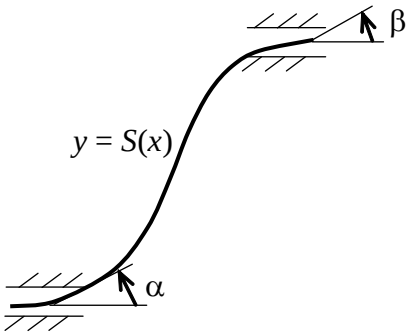


Рис. 5.2. Схема расчета многочлена Ньютона

5.5. Сплайны

Пусть интервал [*a*,*b*] разбит узлами x_i, как и выше, на *n* отрезков, $0 \le i \le n$ Сплайном $S_n(x)$ называется функция, определенная на [*a*, *b*], принадлежащая $C^k[a, b]$ и такая, что на каждом отрезке $[x_i, x_{i+1}]$, $0 \le i \le n-1$ – это полином *n*-й степени.

В частности, это могут быть, построенные специальным образом, многочлены 3-й степени (кубический сплайн), которые являются математической моделью гибкого тонкого стержня, закрепленного в двух точках на концах с заданными углами наклона α и β.



В данной физической модели стержень принимает форму, минимизирующую его потенциальную энергию. Пусть форма стержня определяется какой-то функцией $y = S(x)$. Из курса сопротивления материалов известно, что уравнение свободного равновесия имеет вид $S^{(IV)}(x) = 0$. А этому состоянию соответствует многочлен третьей степени между двумя соседними узлами интерполяции. Его выбирают в виде

$$S(x) = a_i + b_i(x - x_{i-1}) + c_i(x - x_{i-1})^2 + d_i(x - x_{i-1})^3 \; ; \; x_{i-1} \le x \le x_i \; . \qquad (31)$$

Стоит проблема нахождения a_i, b_i, c_i, d_i. Для определения их на всех n элементарных участках интервала $[a,b]$ необходимо составить $4n$ уравнений. Часть этих уравнений в составе $2n$ получают из условия прохождения $S(x)$ через заданные точки, т.е.

$$S(x_{i-1}) = y_{i-1}; \quad S(x_i) = y_i \; .$$

Эти условия можно записать, используя (31) в виде:

$$S(x_{i-1}) = a_i = y_{i-1}; \qquad (32)$$

$$\left.\begin{aligned} &S(x_i) = a_i + b_i h_i + c_i h_i^2 + d_i h_i^3 = y_i; \\ &h_i = x_i - x_{i-1}; \qquad i = 1,2,...,n. \end{aligned}\right\} \qquad (33)$$

Уравнения в количестве $(2n–2)$ получают из условия непрерывности первых и вторых производных в узлах интерполяции. Условие гладкости.

Вычислим производные многочлена (31)

$$S'(x) = b_i + 2c_i(x - x_{i-1}) + 3d_i(x - x_{i-1})^2,$$

$$S''(x) = 2c_i + 6d_i(x - x_{i-1}); \quad \text{при} \;\; x_{i-1} \le x \le x_i \; . \qquad (34)$$

Приравнивая в каждом внутреннем узле $x = x_i$ значения этих производных, вычисленных на концах рассматриваемого отрезка, получают $(2n–2)$ уравнений

$$b_{i+1} = b_i + 2h_i c_i + 3h_i^2 d_i \; ; \qquad i = 1,2,\ldots,n-1; \qquad (35)$$

$$c_{i+1} = c_i + 3h_i d_i \; ; \qquad i = 1,2,\ldots,n-1 \; . \qquad (36)$$

Оставшиеся 2 уравнения получают из естественного предположения условия о нулевой кривизне этой функции на концах отрезка.

$$\left.\begin{aligned} &S''(x_0) = c_1 = 0; \\ &S''(x_n) = 2c_n + 6d_n h_n = 0. \end{aligned}\right\} \qquad (37)$$

Система, составленная из (32) – (37), решается одним из методов решения СЛАУ.

Для упрощения машинных расчетов эта система уравнений приводится к более удобному виду посредством следующего алгоритма.

1. Из условия (32) можно сразу найти a_i.

2. Из (36) – (37) находят:

$$\left.\begin{aligned} d_i &= \frac{c_{i+1}-c_i}{3h_i}; \quad i=1,2,...,n-1; \\ d_n &= -\frac{c_n}{3h_n}. \end{aligned}\right\} \tag{38}$$

3. После подстановки (38) и (32) в (33) находят коэффициенты b_i.

$$b_i = \frac{y_i - y_{i-1}}{h_i} - \frac{h_i}{3}(c_{i+1} + 2c_i); \quad i=1,2,...,n-1;$$

$$b_n = \frac{y_n - y_{n-1}}{h_n} - \frac{2}{3}h_n c_n. \tag{39}$$

4. Учитывая (38) и (39) из уравнения (35) исключаются d_i и b_i, тогда исходная система приводится к трехдиагональной матрице, содержащей только коэффициенты c_i. Получаем систему

$$h_{i-1}c_{i-1} + 2(h_{i-1} + h_i)c_i + h_i c_{i+1} = 3\left(\frac{y_i - y_{i-1}}{h_i} - \frac{y_{i-1} - y_{i-2}}{h_{i-1}}\right), \qquad i=2,3,\ldots,n. \tag{40}$$

При этом $c_1 = 0$, $c_{n+1} = 0$. Система (40) может быть решена методом прогонки. Зная c_i по (38) и (39), определяют b_i и d_i. Тогда кубический многочлен определяется для всех интервалов.

***Пример составления системы* (40)**. Пусть функция $f(x)$ задана таблицей

i	0	1	2	3	4	5
x	0,1	0,15	0,19	0,25	0,28	0,30
$y = f(x)$	1,1052	1,1618	1,2092	1,2840	1,3231	0,3499
h		0,05	0,04	0,06	0,03	0,02

$c_1 = 0$;

$$0{,}05c_1 + 0{,}18c_2 + 0{,}04c_3 = 3\left[\frac{(1{,}2092 - 1{,}1618)}{0{,}04} - \frac{(1{,}1618 - 1{,}1052)}{0{,}05}\right] = 0{,}159;$$

(коэффициент при c_2 получен следующим образом: $2\cdot(0{,}05+0{,}04) = 0{,}18$;)

$$0{,}04c_2 + 0{,}2c_3 + 0{,}06c_4 = 3\left[\frac{(1{,}2840 - 1{,}2092)}{0{,}05} - \frac{(1{,}2092 - 1{,}1618)}{0{,}04}\right] = -0{,}185;$$

$$0{,}06c_3 + 0{,}18c_4 + 0{,}03c_5 = 3\left[\frac{(1{,}3231 - 1{,}2840)}{0{,}03} - \frac{(1{,}2840 - 1{,}2092)}{0{,}06}\right] = -0{,}170;$$

$$0{,}03c_4 + 0{,}1c_5 = 3\left[\frac{(0{,}3499 - 1{,}3231)}{0{,}02} - \frac{(0{,}3231 - 1{,}2840)}{0{,}03}\right] = -1{,}50.$$

$c_6 = 0.$

В результате получим систему относительно $c_2 \div c_5$:

$$\begin{vmatrix} 0{,}18 & 0{,}04 & 0 & 0 \\ 0{,}04 & 0{,}2 & 0{,}06 & 0 \\ 0 & 0{,}06 & 0{,}18 & 0{,}03 \\ 0 & 0 & 0{,}03 & 0{,}1 \end{vmatrix} \times \begin{vmatrix} c_2 \\ c_3 \\ c_4 \\ c_5 \end{vmatrix} = \begin{vmatrix} 0{,}159 \\ -0{,}185 \\ -0{,}170 \\ -1{,}50 \end{vmatrix}.$$

Найдя c_i по (38), находят d_i и затем по (39) – b_i.

5.6. Сглаживание результатов экспериментов

В случае невозможности обеспечения чистоты эксперимента, при получении табличных значений функции, нужно иметь в виду ошибки этих данных. Интерполирование усугубляет эти ошибки. В этом случае для аппроксимации прибегают к построению эмпирических формул, как моделей приближенных функциональных зависимостей. График эмпирических зависимостей не проходит через точки $\{x_i, y_i\}$. В результате экспериментальные данные как бы сглаживаются посредством подбора эмпирических формул.

Построение эмпирических формул состоит из 2-х этапов:

1) построение их общего вида;

2) определение наилучших значений содержащихся в них параметров.

1. Общий вид определяется из физических соображений. Если характер зависимостей неизвестен, то формулы выбираются произвольно, сообразуясь с их простотой. Сначала они выбираются из геометрических соображений среди простейших функций.

2. Если эмпирические формулы подобраны, то они представляются в общем виде:

$$y = \varphi(x, a_0, a_1, ..., a_m); \qquad (41)$$

φ – известная функция; a_i – неизвестные коэффициенты, которые и подбираются для лучшего приближения.

Тогда отклонение (*невязка*) определяется:

$$\varepsilon_i = \varphi(x_i, a_0, a_1, ..., a_m) - y_i; \qquad i = \overline{0, n}. \qquad (42)$$

Задача нахождения a_i сводится к минимизации ε_i. Существует несколько способов: метод выбранных точек, метод средних, метод наименьших квадратов.

1. Метод выбранных точек

В системе координат *XOY* наносится система точек и проводится простейшая плавная кривая или прямая. На проведенной прямой набирается система точек, число которых должно быть равно числу неизвестных коэффициентов в эмпирической формуле. Координаты (x^0_j, y^0_j) старательно измеряются и используются для записи условия прохождения через них прямой.

Из следующей системы находят a_i:

$$\varphi(x_j^0, a_0, a_1, ..., a_m) = y_j^0 \; ; \qquad j = \overline{0, m} \, .$$

2. Метод средних

В данном случае параметры a_i для соотношения (41) находятся из условия:

$$\sum_{i=0}^{n} \varepsilon_i = \sum_{i=0}^{n} [\varphi(x_i, a_0, a_1, ..., a_m) - y_i] = 0 \, . \tag{43}$$

Условно равенство (43) разбивают на систему, состоящую из (m+1) уравнений:

$$\begin{cases} \varepsilon_0 + \varepsilon_1 + \varepsilon_2 = 0; \\ \varepsilon_3 + \varepsilon_4 + \varepsilon_5 + \varepsilon_6 = 0; \\ \quad \ldots \\ \varepsilon_{n-1} + \varepsilon_n = 0. \end{cases} \tag{44}$$

Решают систему (44) и находят коэффициенты a_i.

3. Метод наименьших квадратов

В данном случае речь идет о среднеквадратичном приближении аппроксимируемой функции посредством многочлена:

$$\varphi(x) = a_0 + a_1 x + a_2 x^2 + ... + a_m x^m , \tag{45}$$

при этом $m \le n$; случай $m = n$ соответствует интерполяции. На практике, как правило, m = 1,2,3. Мерой отклонения $\varphi(x)$ от $f(x)$ на множестве точек (x_i, y_i), (i=0,1,…,n), в данном случаи является соотношение по невязке

$$S = \sum_{i=0}^{n} \varepsilon_i^2 = \sum_{i=0}^{n} [\varphi(x_i, a_0, a_1, ..., a_m) - y_i]^2 \, . \tag{46}$$

Параметры $\bar{a}$, как независимые переменные, находятся из условия минимума функции $S = S(a_0, a_1, \ldots, a_{n-1})$.

Система уравнений

$$\left\{ \frac{\partial S}{\partial a_0} = 0, \quad \frac{\partial S}{\partial a_1} = 0, \quad \ldots, \quad \frac{\partial S}{\partial a_m} = 0; \right. \tag{47}$$

трактуется следующим образом

$$\min_{\bar{a}} \sum_{i=1}^{n} [y_i - \varphi(x_i, \bar{a})]^2 = \min_{\bar{a}} \sum_{i=1}^{n} \delta_i^2 = \min_{\bar{a}} \delta(\bar{a}) \, . \tag{48}$$

Из системы (47) определяются параметры a_0, a_1, …, a_m. В этом и состоит метод наименьших квадратов (МНК).

5.7. Вычисление многочленов

Из вышеизложенного очевидно, что при аппроксимации очень часто приходиться вычислять значения многочленов вида:

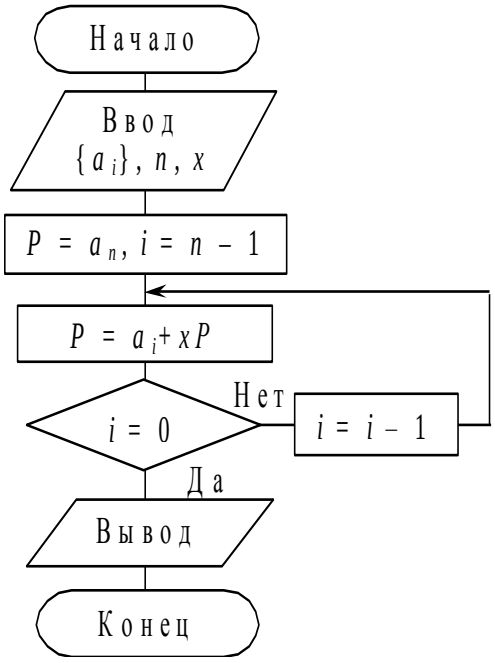
$$P(x) = a_0 + a_1x + a_2x^2 + \ldots + a_nx^n \ . \qquad (54)$$

Если считать в лоб, то нужно $(n^2+n/2)$ умножений, n сложений и плюс округления при этих операциях. Поэтому для вычисления используют схему Горнера.

$$P(x) = a_0 + x(a_1 + x(a_2 + \ldots + x(a_{n-1} + xa_n) \ \ldots \ )) \ . \qquad (55)$$

Здесь требуется n умножений и n сложений.

Алгоритм реализации (54) согласно (55):



Раздел 6. Численное интегрирование

6.1. Постановка задачи

6.1.1. Понятие численного интегрирования

Во многих научных и технических задачах интегрирование функций является важной составной частью математического моделирования площадей и объемов, значений работы, произведенной некоторыми силами и многие другие технические задачи. Напомним, что геометрический смысл простейшего определенного интеграла

$$I = \int_a^b f(x)dx, \tag{1}$$

от $f(x) \geq 0$, как известно, состоит в том, что значение величины I – это площадь, ограниченная кривой $y = f(x)$, осью абсцисс и прямыми $x = a$, $x = b$

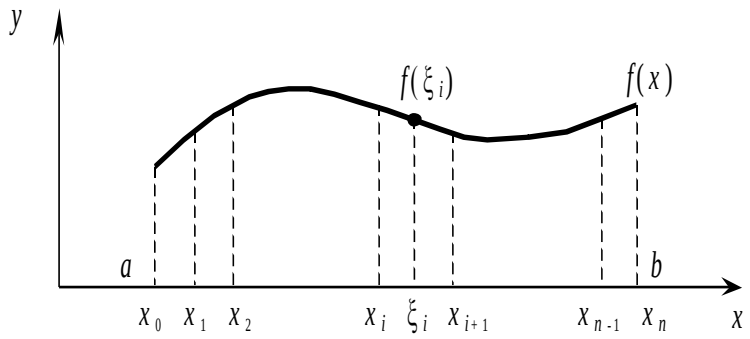


Рис. 6.1

Во многих случаях, когда функция $f(x)$ в (1) задана в аналитическом виде, определенный интеграл вычисляется непосредственно с помощью неопределенного интеграла (посредством первообразной) по формуле Ньютона-Лейбница:

$$\int_a^b f(x)dx = F(x)\Big|_a^b = F(b) - F(a). \tag{2}$$

Однако формулой (2) на практике можно воспользоваться не всегда, а именно:

– когда вид $f(x)$ не допускает непосредственного интегрирования, т.е. первообразная $F(x)$ не выражается в элементарных функциях;

– если значения $f(x)$ заданы в табличной форме.

Универсальным подходом для решения поставленной задачи является использование методов ***численного интегрирования***, основанных на аппроксимации подынтегральной функции с помощью интерполяционных многочленов различных степеней.

Следует подчеркнуть, что основная идея численного интегрирования заложена уже в определении известного интеграла Римана от $f(x)$, формально записанного в виде (1). Напомним суть этого определения.

Пусть вещественная функция $f(x)$ определена и ограничена на интервале $[a, b]$. Разобьем его на n произвольных частичных интервалов $[x_i,\ x_{i+1}]$, $0 \le i \le n-1$, $x_0 = a$, $x_n = b$.

Выберем в каждом частичном интервале произвольную точку ξ, $x_i \le \xi \le x_{i+1}$ и составим, так называемую, *интегральную сумму* (рис. 6.1).

$$S = \sum_{i=0}^{n-1} f(\xi_i)(x_{i+1} - x_i). \tag{3}$$

Если предел S при стремлении длины наибольшего частичного интервала к нулю существует для произвольных ξ_i, то его называют ***интегралом Римана*** от $f(x)$:

$$I = \lim_{\max|x_{i+1}-x_i|\to 0} S. \tag{4}$$

Тогда сумма (3) и дает простейший пример численного интегрирования. А ее верхняя S_2 и нижняя S_1 суммы определяют величину погрешности S, а именно:

$$\begin{cases} |I - S| \le S_2 - S_1; \\ S_1 = \sum_{i=0}^{n-1} m_i (x_{i+1} - x_i); \quad m_i = \min\limits_{x_i \le x \le x_{i+1}} f(x); \\ S_2 = \sum_{i=0}^{n-1} M_i (x_{i+1} - x_i); \quad M_i = \max\limits_{x_i \le x \le x_{i+1}} f(x); \end{cases} \tag{5}$$

Существующие на практике формулы численного интегрирования, по существу, отличаются от (3) только явным указанием способов:

1) выбора x_i, ξ_i;

2) ускорения сходимости в (4);

3) оценки погрешности посредством дополнительной информации о поведении $f(x)$ (например, что $f(x) \in C^2[a,b]$).

В качестве рабочего инструмента численного интегрирования вводится понятие ***квадратурной формулы*** для (1). Для этого обобщим понятие интегральной суммы (3). Точки ξ_i (рис. 6.1), в которых вычисляются значения $f(x)$ называются ***узлами***, а коэффициенты $(x_{i+1} - x_i)$ в (3) заменяют некоторыми числами q_i, не зависящими от $f(x)$, называемыми ***весами***. Формула (3) заменяется следующей:

$$I = \sum_{i=0}^{n-1} q_i f(\xi_i), \tag{6}$$

где $a \le \xi_i \le b$.

Очевидно, что интеграл (1) согласно (5) следует записать в виде:

$$\int_a^b f(x)dx \approx \sum_{i=0}^{n-1} q_i f(\xi_i) + R. \tag{7}$$

Формула (7) и называется *квадратурной формулой*, а R в (7) – погрешностью квадратурной формулы. При наличии альтернативы при выборе численных методов интегрирования следует заметить, что каждая конкретная квадратурная формула считается заданной, если указано, как выбирать ξ_i, соответствующие веса q_i, а также методика оценки погрешности R для определенных классов функций.

6.1.2. Понятие точной квадратурной формулы

Для некоторых классов функций можно записать квадратурные формулы с погрешностью $R \equiv 0$ сразу для всего класса. Такие квадратурные формулы называются ***точными***. Для иллюстрации этого рассмотрим

$$f(x) = P_m(x) = a_0 + a_1 x + ... + a_m x^m$$

на интервале $[a,b]$. Определим на $[a,b]$ произвольные попарно различные узлы ξ_i, $0 \le i \le m$. Искомое точное соотношение для данной функции $f(x)$ будет иметь вид согласно (7):

$$\int_a^b P_m(x)dx = \sum_{i=0}^{m} q_i P_m(\xi_i). \qquad (8)$$

Полином $P_m(x)$ в левой части (8) можно записать в виде интерполяционного многочлена:

$$P_m(x) = \sum_{i=0}^{m} P_m(\xi_i) \frac{(x-\xi_0)...(x-\xi_{i-1})(x-\xi_{i+1})...(x-\xi_m)}{(\xi_i-\xi_0)...(\xi_i-\xi_{i-1})(\xi_i-\xi_{i+1})...(\xi_i-\xi_m)}.$$

Тогда условие (8) позволяет найти значения для весов q_i при $0 \le i \le m$

$$q_i = \int_a^b \frac{(x-\xi_0)...(x-\xi_{i-1})(x-\xi_{i+1})...(x-\xi_m)}{(\xi_i-\xi_0)...(\xi_i-\xi_{i-1})(\xi_i-\xi_{i+1})...(\xi_i-\xi_m)} dx, \qquad (9)$$

Если взять произвольные различные узлы ξ_i на $[a,b]$ и вычислить (9), то соотношение (8) имеет место, т.е. $I = \sum_{i=0}^{m} q_i P_m(\xi_i)$ является точной.

Следует заметить, что формула (8) может оказаться точной для полиномов степени большей, чем m. Это достигают специальным выбором узлов ξ_i на отрезке $[a, b]$, $0 \le i \le m$, что построено Гауссом для полиномов степени $2m + 1$. Практический смысл точных квадратурных формул появляется для таких классов $f(x)$, которые могут быть хорошо аппроксимированными полиномами на интервале $[a,b]$.

Тогда применяя точную формулу к $f(x)$, есть надежда получить малую погрешность R в (7) для рассматриваемого класса функций.

6.2. Простейшие квадратурные формулы

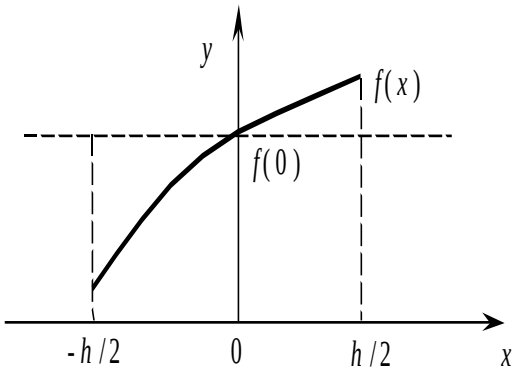
Заметим, что при реализации квадратурных формул (7) в подавляющем большинстве случаев используется равномерная сетка произвольно выбранных

по количеству интерполяционных узлов, что и определяет разные степени используемых интерполяционных многочленов. Чтобы не иметь дело с многочленами высоких степеней обычно интервал интегрирования разбивают на отдельные участки, применяют рабочие формулы невысокого порядка на каждом участке и потом складывают результаты расчета и оценочные погрешности.

Приведем квадратурные формулы для одного интервала $[x_i, x_{i+1}]$, который впоследствии обобщим на весь интервал $[a, b]$ в виде так называемых *составных квадратурных формул*.

6.2.1. Формула прямоугольников

Пусть рассматривается интервал $[-h/2, h/2]$, где $h > 0$.



Предположим, что подынтегральная функция $f(x)$ дважды непрерывно дифференцируема, т.е. $f(x) \in C^2[-h/2, h/2]$. Тогда соотношение (7) запишется в виде:

$$\int_{-h/2}^{h/2} f(x)dx = h \cdot f(0) + R, \qquad (10)$$

здесь взят один узел $\xi = 0$ и соответствующий вес $q = h$.

Полученная квадратурная формула

$$I = h \cdot f(0) \qquad (11)$$

называется *формулой прямоугольников для одного шага* или формулой средних. Такое название определено, так как это есть площадь прямоугольника с высотой $f(0)$ и основанием h. Из рисунка видно, что, уменьшая интервал h при гладкой функции $f(x)$ (т.к. $f(x) \in C^2[-h/2, h/2]$), погрешность $R \to 0$ при $h \to 0$. Доказано, что точность результата для (10) оценивается формулой

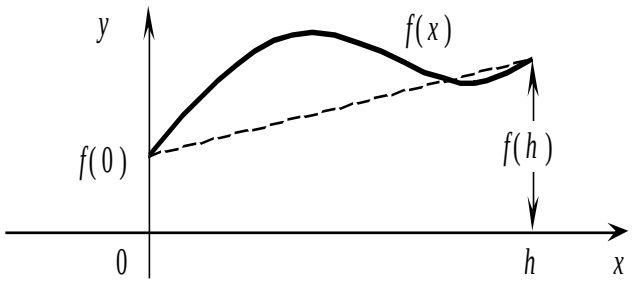
$$R(h, f) = \frac{h^3}{24} f''(\xi), \text{ где } \xi \in [-h/2, h/2].$$

Заметим, что квадратурная формула (11) является точной для полиномов первой степени $P_1(x) = a_0 + a_1 x$, так как $\int_{-h/2}^{h/2} (a_0 + a_1 x)dx = a_0 h$.

Иногда на интервале $[-h/2,\ h/2]$ применяют формулы вида $I=h\cdot f(-h/2)$ и $I=h\cdot f(h/2)$ – формулы правых и левых прямоугольников. Они точны только для полиномов нулевой степени, т.е. констант.

6.2.2. Формула трапеций

Рассмотрим интервал $[0, h]$, $h > 0$



Предположим, что $f(x) \in C^2[0, h]$. Соотношение (7) запишем в виде:

$$\int_0^h f(x)dx = h\frac{f(0)+f(h)}{2} + R, \tag{12}$$

где взяты два узла $\xi_0 = 0$, $\xi_1 = h$ и соответствующие веса $q_0 = q_1 = h/2$.

Получаемая квадратурная формула

$$I = h\frac{f(0)+f(h)}{2}, \tag{13}$$

называется ***формулой трапеций для одного шага***. Название связано с тем, что (13) при положительных значениях $f(0)$, $f(h)$ является площадью трапеции с основаниями $f(0)$, $f(h)$ и высотой h.

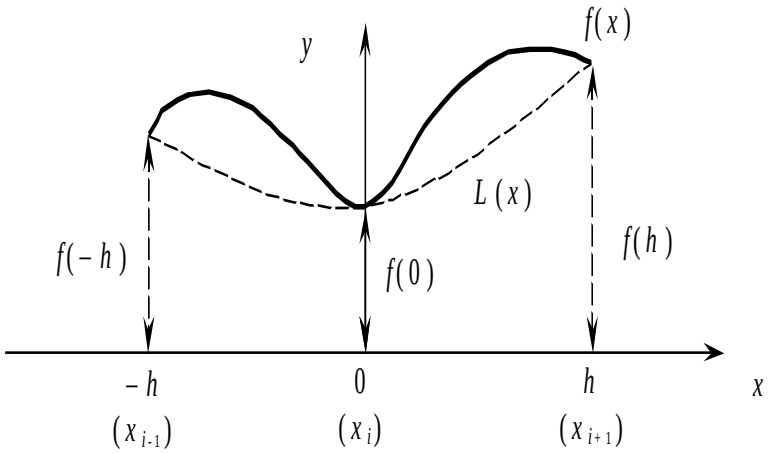
Доказано, что погрешность для (12)

$$R(h, f) = -\frac{h^3}{12} f''(\xi), \tag{14}$$

где ξ – некоторая точка интервала $[0, h]$. Заметим, что (13) так же, как формула прямоугольников точна для полиномов первой степени.

6.2.3. Формула Симпсона

Рассмотрим интервал $[-h, h]$, $h > 0$. Предположим, что $f(x) \in C^4[-h, h]$.



Для соотношения (7) возьмем три узла $\xi_0 = x_{i-1} = -h$, $\xi_1 = x_i = 0$, $\xi_2 = x_{i+1} = h$. Соответствующие им весовые коэффициенты получим из аппроксимации $f(x)$ параболой, построенной на точках $(-h, f(-h))$, $(0, f(0))$, $(h, f(h))$ в виде квадратного многочлена $y = ax^2 + bx + c$. Для получения коэффициентов a, b и c построим многочлен Лагранжа второй степени, проходящий через выбранные точки:

$$P_2(x) = L_n(x) = f(-h)\frac{x(x-h)}{-h(-h-h)} + f(0)\frac{(x+h)(x-h)}{h(-h)} + f(h)\frac{x(x+h)}{h2h}.$$

Вычисляем интеграл:

$$\int_{-h}^{h} P_2(x)dx = \frac{f(-h)}{2h^2}\left(\frac{x^3}{3} - \frac{x^2}{2}h\right)\Big|_{-h}^{h} - \frac{f(0)}{h^2}\left(\frac{x^3}{3} - h^2x\right)\Big|_{-h}^{h} + \\ + \frac{f(h)}{2h^2}\left(\frac{x^3}{3} - \frac{x^2}{2}h\right)\Big|_{-h}^{h} = f(-h)\frac{h}{3} + f(0)\frac{4h}{3} + f(h)\frac{h}{3}. \qquad (16)$$

Тогда соотношение (7) запишется в виде:

$$I = \frac{h}{3}[f(-h) + 4\cdot f(0) + f(h)] + R \qquad (17)$$

и называется формулой ***Симпсона*** (парабол).

Доказано, что погрешность для формулы Симпсона оценивается соотношением:

$$R(h, f) = -\frac{h^5}{90} f^{(IV)}(\xi), \qquad (18)$$

где $\xi \in [-h, h]$.

Из соотношения (18) следует, что квадратурная формула Симпсона точна для полиномов ***третьей*** степени.

Отметим, что при применении простейших квадратурных формул требуются вычисления значения подынтегральных функций $f(x)$:

а) в одной точке – для формулы прямоугольников;

б) в двух точках – для формулы трапеций;

в) в трех точках – для формулы Симпсона.

Однако, несмотря на малый объем вычислений, область практических применений простейших квадратурных формул ограничена лишь малыми интервалами, поскольку при увеличении h погрешность становится значительной, как видно из формул для погрешностей, что и выдвигает необходимость использования т.н. *составных квадратурных формул*.

6.3. Составные квадратурные формулы с постоянным шагом

Итак, если длина интервала $[a, b]$ велика для применения простейших квадратурных формул, то поступают следующим образом:

1) интервал $[a, b]$ разбивают точками x_i, $0 \le i \le n$, на n интервалов по некоторому правилу;

2) на каждом частичном интервале $[x_i, x_{i+1}]$ применяют простейшую квадратурную формулу и находят приближенное значение интеграла

$$\int_{x_i}^{x_{i+1}} f(x)dx \approx I_i; \quad 0 \le i \le n;$$

3) из полученных выражений I_i составляют квадратурную формулу для всего интервала $[a, b]$;

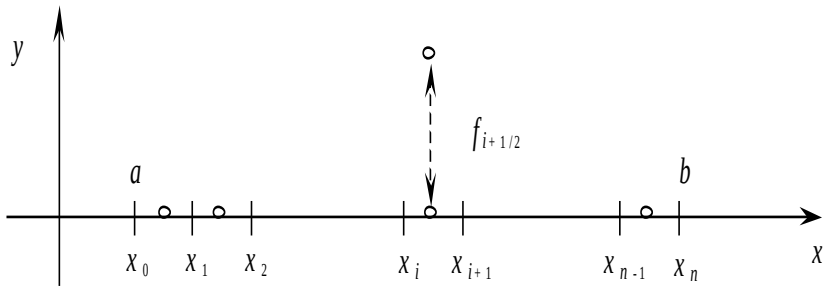
4) абсолютную погрешность R составной формулы находят суммированием R_i.

Для реализации данного алгоритма разобьем интервал $[a, b]$ на частичные интервалы $[x_i, x_{i+1}]$ по следующему правилу: $x_{i+1}–x_i = h$, $0 \le i \le n–1$, $x_0 = a$, $x_n = b$.

Шаг определяется равенством $h = (b – a)/n$.

6.3.1. Составная формула средних

Изобразим рассмотренное правило разбивки



Тогда (10) для каждого интервала будет иметь вид:

$$\int_{x_i}^{x_{i+1}} f(x)dx = hf_{i+1/2} + \frac{h^3}{24} f''(\xi_i), \qquad (19)$$

где $x_i \le \xi_i \le x_{i+1}$, $0 \le i \le n–1$.

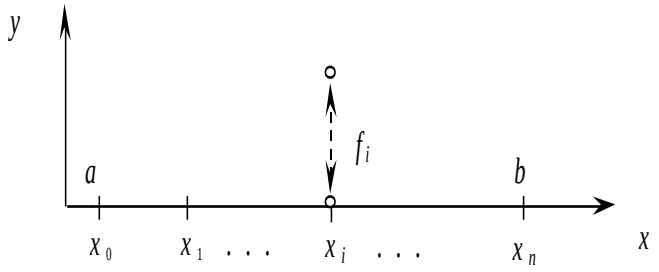
Суммирование по i приводит к составной формуле прямоугольников.

$$\int_a^b f(x)dx = h\sum_{i=0}^{n-1} f_{i+1/2} + R, \qquad (20)$$

где $R = \dfrac{h^2(b-a)}{24}\cdot f''(\xi)$; $\xi \in [a,b]$.

6.3.2. Формула трапеций

Обозначим значение функции $f(x)$ в точках x_i: $f_i = f(x_i)$, $i = \overline{0,n}$.



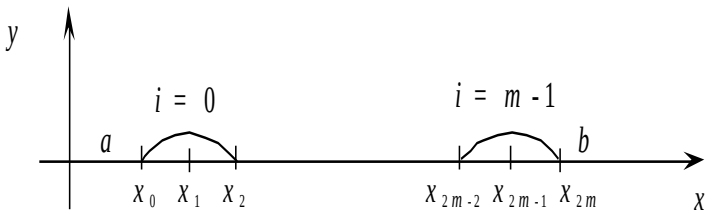
Тогда по аналогии с формулой для прямоугольников из (12) получим составную квадратурную формулу трапеций.

$$\int_a^b f(x)dx = \frac{h}{2}(f_0 + 2\sum_{i=1}^{n-1} f_i + f_n) + R; \qquad R = -\frac{h^2(b-a)}{12}\cdot f''(\xi). \qquad (21)$$

Здесь $\xi \in [a, b]$.

6.3.3. Формула Симпсона

Разобьем интервал $[a, b]$ на четное число частичных интервалов $2m$, где $2m = (b - a)/h$.



Суммируя (17)

$$\int_{x_{2i}}^{x_{2i+2}} f(x)dx = \frac{h}{3}(f_{2i} + 4f_{2i+1} + f_{2i+2}), \quad 0 \le i \le m-1;$$

получим формулу ***Симпсона***

$$\int_a^b f(x)dx = \frac{h}{3}\left(f_0 + 4\sum_{i=1}^{m} f_{2i-1} + 2\sum_{i=1}^{m-1} f_{2i} + f_{2m}\right) + R; \qquad (22)$$

где $R = \dfrac{h^4(b-a)}{180}\cdot f^{IV}(\xi)$, $\xi \in [a, b]$.

Заметим, что в отличие от простейших формул при ***оценке*** их погрешности в составных формулах (20), (21) и (22) нахождение точки $\xi \in [a, b]$ однозначно неопределенно.

На конкретном примере можно оценить точный выбор точки ξ для рассчитанных выше составных формул для интервала $[a, b]$.

Пример. Вычислить интеграл $I = \int_0^1 e^x dx$ с помощью трех квадратурных формул и сравнить ответ с точным значением $I = e - 1 = 1{,}7182818$.

Возьмем произвольно $h = 0{,}1$. Тогда

$$I_h^{П} = h\sum_{i=0}^{n-1} f_{i+1/2} =$$

$$= 0{,}1(e^{0{,}05}+e^{0{,}15}+e^{0{,}25}+e^{0{,}35}+e^{0{,}45}+e^{0{,}55}+e^{0{,}65}+e^{0{,}75}+e^{0{,}85}+e^{0{,}95}) = 1{,}7176;$$

$$I_h^{T} = \frac{h}{2}\left(f_0 + 2\sum_{i=1}^{n-1} f_i + f_n \right) =$$

$$= 0{,}05[e^{0{,}0} + 2(e^{0{,}1}+e^{0{,}2}+e^{0{,}3}+e^{0{,}4}+e^{0{,}5}+e^{0{,}6}+e^{0{,}7}+e^{0{,}8}+e^{0{,}9}) + e^{1}] = 1{,}7197;$$

$$I_h^{C} = \frac{h}{3}\left(f_0 + 4\sum_{i=1}^{m-1} f_{2i-1} + 2\sum_{i=1}^{m-1} f_{2i} + f_{2m} \right) =$$

$$= 0{,}1/3\cdot[e^{0{,}0} + 4(e^{0{,}1}+e^{0{,}3}+e^{0{,}5}+e^{0{,}7}+e^{0{,}9}) + 2(e^{0{,}2}+e^{0{,}4}+e^{0{,}6}+e^{0{,}8}) + e^{1}] = 1{,}7182828.$$

Точное значение I позволяет определить точки ξ для формул соответствующих погрешностям R в (20), (21), (22).

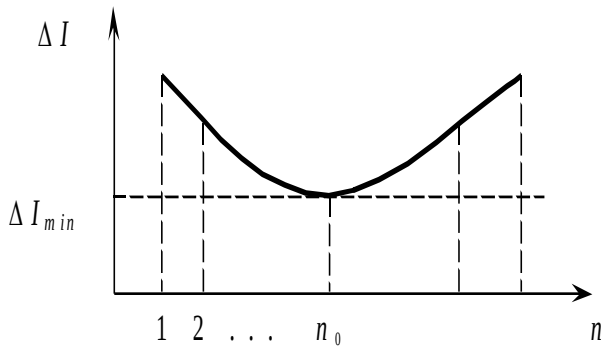
$$I = I_{0{,}1}^{П} + \frac{0{,}1^2}{24} e^{\xi},\ \xi = 0{,}365;$$

$$I = I_{0{,}1}^{T} - \frac{0{,}1^2}{12} e^{\xi},\ \xi = 0{,}532;$$

$$I = I_{0{,}1}^{C} - \frac{0{,}1^4}{180} e^{\xi},\ \xi = 0{,}588.$$

Следовательно, для каждой квадратурной формулы следует выбирать свое ξ с точки зрения оценки точности, что связано с очевидными расчетными трудностями. Утверждение, что повышение точности вычисления интеграла напрямую связано с уменьшением шага h также не совсем верно.

Из практики известно, что, начиная с некоторого (n_0) погрешность вычислений снова начинает увеличиваться по причине округлений малых величин, т.е.



В общем случае погрешность интегрирования может быть представлена в виде:

$$\Delta I = \sum_{i=0}^{n-1}\left(\Delta q_i \max \mid f(\xi_i) \mid + q_i \max \left| \frac{\partial f(\xi_i)}{\partial x} \right| \Delta\xi_i + \mid R \mid \right),$$

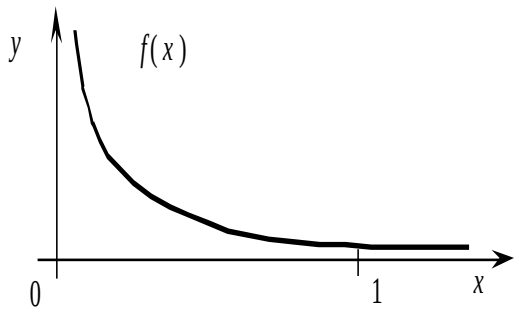
где Δq_i – абсолютная погрешность весов, $\Delta\xi_i$ – абсолютная погрешность узлов, R – погрешность квадратурной формулы.

В связи с вышеизложенным, при вычислении интеграла для выбранной формулы численного интегрирования по заданной точности ε, выбор шага h производится из следующих соображений:

$$\left.\begin{aligned} R_{П} &= \frac{h^2}{24}(b-a)\max_{a\le x\le b} \mid f''(x) \mid = \varepsilon; \\ R_{T} &= \frac{h^2}{12}(b-a)\max_{a\le x\le b} \mid f''(x) \mid = \varepsilon; \\ R_{C} &= \frac{h^4}{180}(b-a)\max_{a\le x\le b} \mid f^{IV}(x) \mid = \varepsilon. \end{aligned}\right\} \qquad (23)$$

Соотношение (23) означает, что шаг h, а, следовательно, и число точек n, в которых вычисляется $f(x)$, определяется значением x с наихудшим поведением $f(x)$ с точки зрения погрешности R.

Однако такое правило разбиения интервала интегрирования может приводить к избыточным вычислениям, если $f(x)$ имеет только частные интервалы с ее «плохим» поведением относительно длины отрезка $[a,b]$.



Для примера рассмотрим подынтегральную функцию типа: $f(x) = e^{-x/\sigma}$ на отрезке [0, 1] с шагом $h = \sigma\sqrt{24\varepsilon}$. Очевидно, что шаг очень мал согласно (23) для обеспечения заданной точности для всего отрезка [0,1], т.е. возникает потребность для устранения избыточных вычислений разбивать интервал $[a, b]$ на частичные интервалы различной длины, которая определяется свойствами $f(x)$ и заданной точностью интегрирования.

Таким образом, возникает задача применения *простейших квадратурных* формул интегрирования с переменным шагом интегрирования на отрезке [*a, b*]. Данная ситуация будет рассмотрена ниже.

6.4. Выбор шага интегрирования для равномерной сетки

Данная задача состоит в выборе шага *h*, обеспечивающего заданную точность ε вычисления интеграла по выбранной формуле численного интегрирования.

Известны два подхода к решению данной задачи:

1) выбор шага по теоретическим оценкам погрешностей (23);

2) по косвенным схемам (эмпирическим оценкам).

6.4.1. Выбор шага интегрирования по теоретическим оценкам погрешностей

Пусть требуется вычислить интеграл с точностью ε. Тогда, используя формулу для *R*, выбирают шаг так, чтобы

$$| R | < \varepsilon/2 .$$

Учитывается также число знаков после запятой, чтобы погрешность округления не превышала ε/2.

Пример. С помощью формулы Симпсона вычислить $I = \int_{\pi/4}^{\pi/2} \frac{\sin x}{x} dx$ с точностью $\varepsilon = 10^{-3}$.

Решение. Выберем шаг *h*.

$$R_c = -\frac{h^4 (b-a)}{180} f^{IV}(\xi); \quad \xi \in [a,b], \quad \text{т.е.} \quad \xi \in [\pi/4, \pi/2] ;$$

Согласно соотношений (23), получим

$$\frac{h^4 (b-a)}{180} \max_{[a,b]} \left| f^{IV}(x) \right| < 0{,}5 \cdot 10^{-3}.$$

Вычислим $f^{IV}(x)$

$$f^{IV}(x) = \frac{\sin x}{x} + 4\frac{\cos x}{x^2} - 12\frac{\sin x}{x^3} - 24\frac{\cos x}{x^4} + 24\frac{\sin x}{x^5} . \qquad (24)$$

Оценим $| f^{IV}|$ на отрезке [π/4, π/2]. Воспользуемся величинами из (24) $\frac{\sin x}{x}\left(1 - \frac{12}{x^2} + \frac{24}{x^4}\right)$ и $\frac{4\cos x}{x^2}\left(\frac{6}{x^2} - 1\right)$. Они положительные и убывают, следовательно, их максимальное значение в точке $x = \pi/4$.

При этом $\left|f^{(IV)}(x)\right| \le \frac{\sin x}{x}\left(1-\frac{12}{x^2}+\frac{24}{x^4}\right)+\frac{4\cos x}{x^2}\left(\frac{6}{x^2}-1\right) < 81$. Таким образом, $R \le \frac{h^4 \cdot \pi/4}{180} \cdot 81 < 0{,}5\cdot 10^{-3}$; $h^4 < 14\cdot 10^{-4}$; $h \le 0{,}19$.

С другой стороны для данного метода h выбирается с учетом того, чтобы $[\pi/4, \pi/2]$ делился на четное число отрезков. Этим двум требованиям отвечает $h=\pi/24 = 0{,}13 < 0{,}19$, при котором $n = \frac{b-a}{h} = 6$. Тогда, чтобы погрешность округления не превысила $0{,}5\cdot 10^{-3}$ достаточно вычисления выполнить с 4 знаками после запятой.

Составим таблицу $y = \frac{\sin x}{x}$, с $h = \pi/24 = 7°\ 30' = 0{,}1309$

i	x_i^0	x_i	$\sin x$	y_0, y_6	y_{2m}	y_{2m-1}
0	45° 00′	0,7854	0,7071	0,9003		
1	52° 30′	0,9163	0,7934			0,8659
2	60° 00′	1,0472	0,8660		0,8270	
3	67° 30′	1,1781	0,9239			0,7843
4	75° 00′	1,3090	0,9659		0,7379	
5	82° 30′	1,4399	0,9914			0,6885
6	90° 00′	1,5708	1,0000	0,6366		
Сумма				1,5369	1,5649	2,3386

Для $n = 6$ по формуле Симпсона

$$\int_{\pi/4}^{\pi/2} \frac{\sin x}{x} = \frac{h}{3}\left[(y_0 + y_6) + 4(y_1 + y_3 + y_5) + 2(y_2 + y_4)\right] = 0{,}6118 \approx 0{,}612 .$$

6.4.2. Выбор шага интегрирования по эмпирическим схемам

1. Двойной пересчет

В связи с тем, что вычисления максимального значения по абсолютной величине k-ой производной приводят к громоздкости расчетов, на практике прибегают к искусственным приемам достижения заданной точности. А именно, определенный интеграл вычисляют по какой-либо квадратурной формуле дважды с шагом h и $h/2$, что удваивает число n.

Определяют:

если $|I_n - I_{2n}| < \varepsilon$, то $I = I_{2n}$;

если $|I_n - I_{2n}| > \varepsilon$, то берут шаг $h/4$; (25)

если $|I_{2n} - I_{4n}| < \varepsilon$, то $I = I_{4n}$.

В качестве начального шага h можно рекомендовать $h = \sqrt[m]{\varepsilon}$, где m=2 для формул среднего и трапеций, m=4 – для Симпсона.

2. Схема Эйткина

На практике для повышения точности численного интегрирования широко используется *схема Эйткина*. Рассмотрим ее смысл.

Расчет проводиться три раза с h_1, h_2, h_3, при этом соотношение между ними $\frac{h_2}{h_1} = \frac{h_3}{h_2} = q$. Получают три значения I_1, I_2, I_3.

Производится уточнение по эмпирической формуле:

$$I = I_1 - \frac{(I_1 - I_2)^2}{I_1 - 2I_2 + I_3}. \tag{26}$$

Порядок точности $\rho = \frac{1}{\ln g} \ln \frac{I_3 - I_2}{I_2 - I_1}$.

3. Правило Рунге

Это наиболее популярное практическое правило, разработанное в предположении, что $f(x) \in C^4[a, b]$ для квадратурных формул прямоугольников и трапеций, $f(x) \in C^6[a, b]$ – для формулы Симпсона. В этом случае можно показать, что погрешности $R(h, f)$ имеют следующие представления при $h \to 0$:

$$R^{П} = \left(\frac{1}{24} \int_a^b f''(x)dx \right) h^2 + O(h^4); \tag{27}$$

$$R^{T} = \left(-\frac{1}{12} \int_a^b f''(x)dx \right) h^2 + O(h^4);$$

$$R^{C} = \left(-\frac{1}{180} \int_a^b f^{IV}(x)dx \right) h^4 + O(h^6).$$

Суть его также состоит в том, чтобы, организовав вычисления двух значений интеграла по двум семействам узлов, сравнивают результаты вычислений с оценкой погрешности. Объединив (27) можно получить рабочую формулу:

$$I = I_{h/2} + \frac{I_{h/2} - I_h}{2k - 1} + O(h^{k+m}), \quad h \to 0; \tag{28}$$

где $k = 2$, $m = 2$ – для прямоугольников и трапеций;
$k = 4$, $m = 2$ – для формулы Симпсона.

4. Другие оценки погрешности

1. Приближенной оценкой погрешности могут быть:

$\Delta \approx \frac{1}{3} |I_n - I_{2n}|$ – для трапеции и прямоугольника;

$\Delta \approx \frac{1}{15}|I_n - I_{2n}|$ – для формулы Симпсона.

2. Следует заметить, что эмпирические формулы (25), (26), (28) предполагают и автоматическое изменение шага интегрирования h. Для этой цели имеется другая схема расчета, заключающаяся в следующем.

Анализ составных формул $I_П$, I_T, I_C показывают, что точное значение интеграла находится между $I_{Ср.П}$ (формула средних) и I_T, при этом имеет место соотношение:

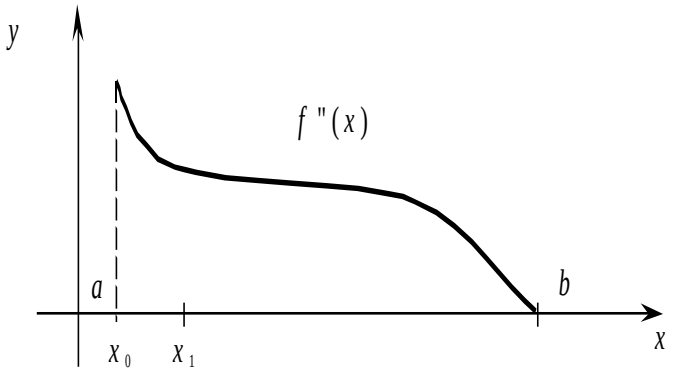
$$I_C = (2 \cdot I_{Ср.П} + I_T)/3. \qquad (29)$$

Соотношение (29) используется и для контроля погрешности вычисления. Если $|I_C - I_T| \geq \varepsilon$, то шаг уменьшают вдвое и расчет повторяют. Если точность достигнута, то окончательное значение интеграла и получают по формуле (29).

6.5. Составные квадратурные формулы с переменным шагом

Проиллюстрируем решение данной проблемы на примере квадратурной формулы прямоугольников.

Пусть $f(x) \in C^2[a, b]$ с дополнительным ограничением: $f''(x)$ – монотонная знакоопределенная функция на $[a, b]$. Для определенности возьмем $f''(x)$ – монотонно убывающую положительную функцию.



Положим $x_0 = a$. Определим наибольшее значение x_1 из условия (23), т.е. чтобы погрешность для

$$\int_{x_0}^{x_1} f(x)dx = (x_1 - x_0) f(\frac{x_0 + x_1}{2}) + R_П \; ; \; R_П = \frac{(x_1 - x_0)^3}{24} \cdot f''(\xi) = \varepsilon; \; x_0 \leq \xi \leq x_1 ; \quad (30)$$

не превышала заданной величины ε. Очевидно, что для этого достаточно решить (24) относительно x_1.

Имеем $x_1 = \left(\frac{24\varepsilon}{f''(x_0)} \right)^{1/3} + x_0$.

Следующие интервалы определяются аналогично.

Из рисунка видно, что длина последующих интервалов будет возрастать. Общая формула их определения такова:

$$x_{i+1} = \left(\frac{24\varepsilon}{f''(x_i)} \right)^{1/3} + x_i ; \qquad 0 \le i \le k . \tag{31}$$

Количество интервалов k неизвестно, т.к. оно определяется как точностью ε, так и поведением $f''(x)$ на интервале $[a,b]$. Однако верхняя оценка для k может быть легко определена по длине наименьшего частичного интервала:

$$k \le \left[\frac{(b-a)(f''(x_0))^{1/3}}{(24\varepsilon)^{1/3}} \right].$$

Суммируя (30) получим составную квадратурную формулу прямоугольников с переменным шагом:

$$\int_a^b f(x)dx = (24\varepsilon)^{1/3} \sum_{i=0}^{k} \frac{f\left(x_1 + \frac{1}{2}\left(\frac{24\varepsilon}{f''(x_i)} \right)^{1/3} \right)}{(f''(x_i))^{1/3}} + R ;$$

где x_i определяется рекуррентно формулами (31). Для погрешности R имеет место оценка $|R| \le k\varepsilon$.

В общем случае для произвольной функции $f(x)$, если $f''(x)$ – монотонно возрастающая положительная функция, то частичные интервалы определяются справа налево, т.е. от b к a. Для отрицательной производной $f''(x)$ и монотонно возрастающей – слева направо от a к b, для убывающей – справа налево от b к a.

В качестве иллюстрации рассмотрим интегрирование $f(x) = e^{-x/\sigma}$, $\sigma = 10^{-2}$ с точностью $\varepsilon = 10^{-4}$ на каждом частичном интервале, принадлежащем отрезку [0;1]. По (31) определим границы интервалов:

$x_0 = 0{,}0000$; $x_1 = 0{,}0062$; $x_2 = 0{,}0138$; $x_3 = 0{,}0237$; $x_4 = 0{,}0374$;

$x_5 = 0{,}0590$; $x_6 = 0{,}1030$; $x_7 = 0{,}2990$; $x_8 = 1{,}0000$.

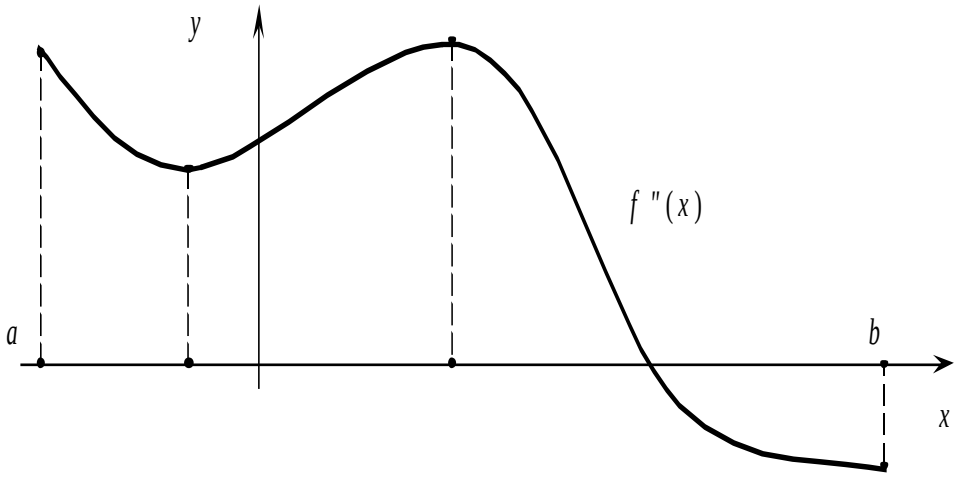
Общая погрешность имеет оценку $R \le 8 \cdot 10^{-4}$. Такую погрешность посредством формулы прямоугольников с h = const можно получить, если выбирать шаг h на всем интервале из условия $\frac{h^2}{24}(1-0)\max\limits_{0 \le x \le 1}(e^{-x/\sigma})'' = R$, на 721-м частичном интервале:

$$K = \frac{1}{\sigma\sqrt{24R}} \approx 721 .$$

В общем случае, если $f''(x)$ на всем интервале $[a,b]$ не удовлетворяет принятому дополнительному ограничению, то

– сначала следует интервал $[a,b]$ разбить на частичные интервалы, на которых $f''(x)$ монотонна и знакоопределена;

– затем на каждом из них построить составную квадратурную формулу с переменным шагом по приведенным выше формулам.



Аналогичные рассуждения имеют место и для формулы Симпсона с соблюдением монотонности $f^{IV}(x)$.

Однако следует заметить, что переход к переменному шагу h не всегда оправдан из-за необходимости вычислять $f''(x)$ и определять ее монотонность и знакоопределенность. Это бывает оправданным только при серийных расчетах.

6.6. Квадратурные формулы наивысшей алгебраической точности (формула Гаусса)

Рассмотренные выше квадратурные формулы прямоугольников, трапеций и Симпсона применяются для интегрирования функций $f(x)$ невысокой степени гладкости (не выше $f(x) \in C^2[a, b]$). Для данного класса функций они просты и удобны. И как показано выше, для повышения точности результатов, как один из подходов, всегда стремятся отрезок интегрирования разбивать на достаточно большее число частей. Однако практикой доказано, что для класса функций высокой степени гладкости ($f(x) \in C^k[a,b]$, $k>2$) точность этих квадратурных формул не повышается с ростом k, т.е. имеет место так называемое явление насыщения численного метода. Для такого класса функций разработаны другие квадратурные формулы такого же типа, что и раньше $\int_a^b f(x)dx = \sum_{i=0}^{n-1} q_i f(\xi_i) + R$, но посредством их структурного реформирования путем подбора в них $(2n+1)$ параметров: n узлов x_i, n коэффициентов q_i и самого числа n.

Все эти параметры выбираются так, чтобы квадратурная сумма возможно меньше отличалась от точного значения интеграла для всех функций f из некоторого класса. Используя математический аппарат в виде, так называемых, полиномов Лежандра, построенных на отрезке [–1, 1] получаем рабочую квадратурную формулу Гаусса:

$$\int_{-1}^{1} f(x)dx = \sum_{i=1}^{n} q_i f(\xi_i) + R, \qquad (33)$$

которая является точной ($R = 0$) для всех полиномов степени $N = 2n – 1$.

Корни вспомогательного полинома Лежандра расположены симметрично относительно нуля, соответствующие веса совпадают и они всегда положительные.

Для практических целей искомые коэффициенты q_i и абсциссы ξ_i для произвольных n табулированы для формулы (33).

n	ξ_i	q_i
. . . 4	 $-\xi_1 = \xi_4 = 0{,}861136312$ $-\xi_2 = \xi_3 = 0{,}339981044$	 $q_1 = q_4 = 0{,}347854845$ $q_2 = q_3 = 0{,}652145155$
5	$-\xi_1 = \xi_5 = 0{,}906179846$ $-\xi_2 = \xi_4 = 0{,}538469310$ $\xi_3 = 0$	$q_1 = q_5 = 0{,}236926885$ $q_2 = q_4 = 0{,}478628670$ $q_3 = 0{,}568888889$
. . .		

При вычислении интеграла $\int_a^b f(t)dt$ следует сделать замену переменной интегрирования $t = x(b – a)/2 + (a + b)/2$. Тогда

$$\int_a^b f(t)dt = \frac{b-a}{2}\sum_{k=1}^{n} q_k f(t_k) + R, \qquad (34)$$

где $t_k = x_k(b – a)/2 + (b + a)/2$, x_k – узлы формулы (33) на отрезке $[–1;1]$ и q_k – соответствующие им коэффициенты, взятые из таблицы.

Пример. По формуле Гаусса при $n = 5$ вычислить $I = \int_0^1 \frac{dx}{1+x^2}$.

Решение. Сделаем замену переменной $x = 1/2+ t \cdot 1/2$, тогда

$$I = 2\int_{-1}^{1} \frac{dt}{4+(t+1)^2}.$$

Составим таблицу значений подынтегральной функции.

i	ξ_i	$f(\xi_i)$	q_i
1	–0,9061179846	0,24945107	0,236926885
2	–0,538469310	0,23735995	0,478628670
3	0	0,2	0,568888889
4	0,538469310	0,15706211	0,478628670
5	0,906179846	0,13100114	0,236926885

По формуле Гаусса (33) определим:

$I = 2[q_1 f(\xi_1) + q_2 f(\xi_2) + ... + q_3 f(\xi_3)] = 0{,}78539816$;

$I_{\text{Точное}} = \pi/4 = 0{,}785398163\ldots$ метод Симпсона с шагом $h = 0{,}1$ даст погрешность в шестом разряде.

Раздел 7. Численное дифференцирование

7.1. Постановка задачи

К численному дифференцированию (ЧД) прибегают тогда, когда приходиться вычислять производные для функций, заданных таблично или когда непосредственное дифференцирование $y = f(x)$ затруднительно. Формулы для расчета $\dfrac{d^m f(x)}{dx^m}$ в точке x области определения функции получают посредством аппроксимации оператора дифференцирования интерполяционными многочленами как локальной, так и глобальной интерполяции. А именно, к исследуемой точке x берутся несколько близких к ней узлов $x_1, x_2, \ldots, x_n$ ($n \geq m+1$), называемых шаблоном. Вычисляются значения $y_i = f(x_i)$ в узлах шаблона и строится интерполяционный многочлен

$$y = f(x) \approx \varphi(x; \overline{a}) = P_{n-1}(x).$$

Тогда $\dfrac{d^m f}{dx^m} \approx d^m P_{n-1} / dx^m$.

Для получения рабочих формул с точки зрения упрощения их реализации интерполирование производится на равномерной сетке, и производные обычно находятся в узлах x_i с соответствующей оценкой их погрешностей. При $n = m+1$ формулы ЧД не зависят от положения точки x внутри шаблона, т.к. m-я производная от полинома m-й степени есть константа. Такие формулы называются простейшими формулами ЧД.

7.2. Аппроксимация производных посредством локальной интерполяции

В случае табличного задания функции производную находят, опираясь на формулу

$$y' = f'(x) = \lim_{\Delta x \to 0} \frac{\Delta y}{\Delta x}; \qquad \Delta y = f(x + \Delta x) - f(x);$$

полагая

$$y' \approx \frac{\Delta y}{\Delta x}. \tag{1}$$

Это соотношение называется аппроксимацией производной с помощью отношения конечных разностей. При заданных значениях таблицы $\{x_i, y_i\}$, $i = \overline{0, n}$ и шаге расположения интерполяционных узлов h = const в зависимости от способа вычисления конечных разностей для i-го узла имеют место следующие алгоритмы вычисления (1). Пусть $i = 1$.

1. Формула левых разностей

$$\Delta y_1 = y_1 - y_0; \quad \Delta x = h;$$

$$y'_1 = \frac{y_1 - y_0}{h} \ . \qquad (2)$$

2. Формула правых разностей

$$\Delta y_1 = y_2 - y_1; \quad \Delta x = h;$$

$$y'_1 = \frac{y_2 - y_1}{h} . \qquad (3)$$

3. Формула центральных разностей

$$\Delta y_1 = y_2 - y_0; \quad \Delta x = 2h;$$

$$y'_1 = \frac{y_2 - y_0}{2h} . \qquad (4)$$

Используя соотношения (2), (3), (4) последовательно можно получить выражения для вычисления производных высших порядков. К примеру, используя (3), получим:

$$y''_1 = (y'_1)' = (y'_2 - y'_1)/h = ((y_2 - y_1)/h - (y_1 - y_0)/h)/h = \frac{y_2 - 2y + y_0}{h^2} . \qquad (5)$$

Открытым остается вопрос точности.

7.3. Погрешность численного дифференцирования

Аппроксимируя исследуемую функцию, ее представляют в виде:

$$f(x) = \varphi(x) + R(x) . \qquad (6)$$

В качестве $\varphi(x)$ можно принять либо интерполяционную функцию, либо частичную сумму ряда. Тогда погрешность аппроксимации $R(x)$ определяется остаточным членом ряда или $P_{n-1}(x)$. Дифференцируя (6) необходимое число раз находим:

$$f'(x) = \varphi'(x) + R'(x); \quad f''(x) = \varphi''(x) + R''(x) \ \text{ и т.д.}$$

Тогда погрешность аппроксимации $R^{(k)}(x) = f^{(k)}(x) - \varphi^{(k)}(x)$ при численном дифференцировании функции, заданной таблицей с шагом h зависит от h, и ее записывают в виде $O(h^k)$. Показатель степени k называют порядком погрешности аппроксимации производной. При этом предполагается, что $|h| < 1$.

Оценку погрешности формул (2) – (5) можно проиллюстрировать с помощью ряда Тейлора.

Пусть дважды непрерывно дифференцируемая функция $f(x)$ задана таблицей значений.

x	x_0	x_1	x_2	…	x_n
y	y_0	y_1	y_2	…	y_n

Где $y_i = f(x_i)$, $i = \overline{0,n}$. Пусть далее узлы равностоящие, $h = (x_n - x_0)/n$, $x_i = x_0 + ih$, $h = x_{i+1} - x_i$, $i = \overline{0,n-1}$.

Ряд Тейлора в общем виде:

$$f(x+\Delta x) = f(x) + f'(x)\Delta x + \frac{f''(x)}{2!}\Delta x^2 + \frac{f'''(x)}{3!}\Delta x^3 + \ldots \qquad (7)$$

Запишем (7) при $x = x_1,\ \Delta x = -h$ с точностью до h^1:

$$y_0 = y_1 - y'_1 h + O(h^2).$$

Тогда $y'_1 = \frac{y_1 - y_0}{h} + O(h)$.

Это выражение совпадает с (2) и является аппроксимацией первого порядка ($k = 1$). Тогда для произвольного узла $y'_i = \frac{y_i - y_{i-1}}{h}, \quad i = \overline{1, n-1}$.

А по всему отрезку $[a,b]$, где $h = (b-a)/n$ для $f'(x)$ погрешность не превысит величины $R = \frac{h}{2}\max\limits_{a \le x \le b} |f''(x)|$.

Полагая для (7) $\Delta x = h$, можно получить этот результат и для соотношения (3). Для оценки погрешности для (4) и (5) воспользуемся рядом Тейлора, полагая $\Delta x = -h$ и $\Delta x = h$ соответственно получим:

$$y_0 = y_1 - y'_1 h + \frac{y''_1}{2!}h^2 - \frac{y'''_1}{3!}h^3 + O(h^4); \qquad (8)$$

$$y_2 = y_1 + y'_1 h + \frac{y''_1}{2!}h^2 + \frac{y'''_1}{3!}h^3 + O(h^4);$$

в предположении, что $f(x)$ трижды непрерывно дифференцируемая функция.

Вычитая из второго равенства первое, получаем:

$$y'_1 = \frac{y_2 - y_0}{2h} + O(h^2), \qquad \text{здесь } k = 2.$$

Для произвольного узла:

$$y'_i = \frac{y_{i+1} - y_{i-1}}{2h} + O(h^2), \quad i = \overline{1, n-1}.$$

На основании (7) по всему отрезку погрешность аппроксимации не превзойдет величины:

$$R_1 \le \frac{h^2}{6}\max_{a \le x \le b} \left| f'''(x) \right|.$$

Складывая равенства (8) найдем:

$$y''_1 = \frac{y_0 - 2y_1 + y_2}{h^2} + O(h^2), \qquad k = 2.$$

Для отрезка $[x_{i-1}, x_{i+1}]$ получим:

$$y''_i = \frac{y_{i-1} - 2y_i + y_{i+1}}{h^2}, \quad i = \overline{1, n-1}.$$

А погрешность на отрезке [*a*, *b*] для второй производной оценивается соотношением:

$$R_2 \le \frac{h^2}{12} \max_{a \le x \le b} \left| f^{(IV)}(x) \right|.$$

Следует отметить, что, вообще говоря, приближенное дифференцирование представляет собой операцию менее точную, чем интерполирование. Считают, что при численном дифференцировании функции $y = f(x)$, заданной табличо, имеют место два типа погрешностей:

а) погрешности усечения, которые вызываются заменой функции $y = f(x)$ интерполяционным многочленом $P_n(x)$;

б) погрешности округления, которые вызываются неточным заданием исходных значений y_i.

При этом известно, что с уменьшением шага численного дифференцирования погрешность округления возрастает, а погрешность же усечения, как правило, убывает. Поэтому при вычислениях по формулам численного дифференцирования стоит задача и оптимального выбора шага *h*.

7.4. Аппроксимация производных посредством глобальной интерполяции

7.4.1. Аппроксимация посредством многочлена Ньютона

Предположим, что функция $f(x)$, заданная в виде таблицы с постоянным шагом $h = x_i - x_{i-1}$ ($i = 1,2,\ldots, n$) может быть аппроксимирована интерполяционным многочленом Ньютона:

$$y \approx N(x_0 + th) = y_0 + t\Delta y_0 + \frac{t(t-1)}{2!}\Delta^2 y_0 + \ldots + \frac{t(t-1)\ldots(t-n+1)}{n!}\Delta^n y_0, t = \frac{x - x_0}{h}. \quad (9)$$

Дифференцируя (9) по переменной *x* как функцию сложную:

$$\frac{dN}{dx} = \frac{dN}{dt} \cdot \frac{dt}{dx} = \frac{1}{h} \cdot \frac{dN}{dt}$$

можно получить формулы для получения производных любого порядка:

$$y' \approx \frac{1}{h}\left(\Delta y_0 + \frac{2t-1}{2!}\Delta^2 y_0 + \frac{3t^2 - 6t + 2}{3!}\Delta^3 y_0 + \frac{4t^3 - 18t^2 + 22t - 6}{4!}\Delta^4 y_0 + \ldots. \right);$$

$$y'' \approx \frac{1}{h^2}\left(\Delta^2 y_0 + \frac{6t-6}{3!}\Delta^3 y_0 + \frac{12t^2 - 36t + 22}{4!}\Delta^4 y_0 + \right.$$

$$\left. + \frac{20t^3 - 120t^2 + 210t - 100}{5!}\Delta^5 y_0 + \ldots \right); \quad (10)$$

Следует заметить, что точность ЧД для выбранного *x* будет существенно зависеть от значений функции во многих узлах, что не предусмотрено в соотношениях (2) – (4).

Пример. Для функции заданной таблично

x	y	Δy	$\Delta^2 y$	$\Delta^3 y$	$\Delta^4 y$	$\Delta^5 y$
0	1,2833					
0,1	1,8107	0,5274	0,0325	0,0047	0,0002	0,0000
0,2	2,3606	0,5599	0,0372	0,0049	0,0002	
0,3	2,9577	0,5971	0,0421	0,0051		
0,4	3,5969	0,6392	0,0472			
0,5	4,2833	0,6864				

вычислить в точке $x = 0{,}1$ первую $f'(x)$ и вторую $f''(x)$ производные. Здесь h=0,1; $t = (0{,}1 - 0)/0{,}1 = 1$. Предварительно вычислим конечные разности для (10).

Используя формулы (10), находим:

$$y' \approx 10\cdot (0{,}5274+((2\cdot 1-1)/2)\cdot 0{,}0325+0{,}0047\cdot(3\cdot 1-6\cdot 1+2)/6+0{,}0002\cdot(4\cdot 1- -18\cdot 1+22\cdot 1-6)/24) = 5{,}436;$$

$$y'' \approx 100\cdot (0{,}0325+0{,}0047\cdot(6\cdot 1-6)/6+0{,}0002\cdot(12-36+22)/24) = 3{,}25.$$

Замечание. В расчетной практике численного дифференцирования интерполяционные многочлены Ньютона, Гаусса, Стирлинга и Бесселя используются в несколько иной форме, так как формулы ЧД применяют для нахождения производных в равностоящих узлах $x_i = x_0 + ih$ ($i = 0, \pm 1, \pm 2, \ldots$), то любую точку сетки можно принять за начальную и формулы ЧД записывают для точки x_0. А это равносильно подстановке в них $t = (x - x_0)/h = 0$. Тогда дифференцирование многочленов приводит к следующим формулам.

По Ньютону:

$$y'_0 = f'(x_0) = \frac{1}{h}\left(\Delta y_0 - \frac{1}{2}\Delta^2 y_0 + \frac{1}{3}\Delta^3 y_0 - \ldots + (-1)^{n-1}\frac{1}{n}\Delta^n y_0\right); \qquad (a)$$

$$y''_0 = f'(x_0) = \frac{1}{h^2}\left(\Delta^2 y_0 - \Delta^3 y_0 + \frac{11}{12}\Delta^4 y_0 - \frac{5}{6}\Delta^5 y_0 + \ldots\right);$$

$$y'_0 = f'(x_0) = \frac{1}{h}\left(\Delta y_{-1} - \frac{1}{2}\Delta^2 y_{-2} + \frac{1}{3}\Delta^3 y_{-3} + \ldots + \frac{1}{n}\Delta^n y_{-n}\right); \qquad (б)$$

$$y''_0 = f''(x_0) = \frac{1}{h^2}\left(\Delta^2 y_{-2} - \Delta^3 y_{-3} + \frac{11}{12}\Delta^4 y_{-4} + \frac{5}{6}\Delta^5 y_5 + \ldots\right).$$

Формулы (*а*) применяются для начальных строк таблиц, а (*б*) – для последних строк таблицы. Тогда по Стирлингу:

$$y'_0 = f'(x_0) \approx \frac{1}{h}\left(\frac{\Delta y_{-1} + \Delta y_0}{2} - \frac{1}{6}\frac{\Delta^3 y_{-2} + \Delta^3 y_{-1}}{2} + \frac{1}{30}\frac{\Delta^5 y_{-3} + \Delta^5 y_{-2}}{2} + \ldots.\right); \qquad (c)$$

$$y''_0 = f''(x_0) \approx \frac{1}{h^2}\left(\Delta^2 y_{-1} - \frac{1}{12}\Delta^4 y_{-2} + \frac{1}{90}\Delta^6 y_{-4} + \ldots..\right).$$

Формулы (*с*) – для дифференцирования в середине таблицы.

Пример. Использование формул (*a*) и (*c*) для функции $y = \mathrm{sh}2x$ с $h = 0{,}05$. Найти y' и y'' в точках $x = 0{,}00$ и $x = 0{,}1$. Возьмем расчетную таблицу для $y = f(x)$ в виде:

x	$y = f(x)$	Δy	$\Delta^2 y$	$\Delta^3 y$	$\Delta^4 y$	$\Delta^5 y$
0,00	0,0000					
		10017				
0,05	0,10017		100			
		10117		101		
0,10	0,20134		201		3	
		10318		104		0
0,15	0,30452		305		3	
		10623		107		
0,20	0,41075		412			
		11035				
0,25	0,52110					

Решение. Воспользуемся формулами ЧД на основе интерполяционных многочленов. Составим таблицу конечных разностей. Она продолжилась до разностей 4-го порядка, т.к. дальше получится «0».

Для точки $x = 0{,}0$ используем формулы (*a*), считая $x_0 = 0{,}0$:

$$y'|_{x=0,0} \approx \frac{1}{h}\left(\Delta y_0 - \frac{1}{2}\Delta^2 y_0 + \frac{1}{3}\Delta^3 y_0 - \frac{1}{4}\Delta^4 y_0\right) =$$

$$= 20 \cdot (0{,}10017 - 0{,}00050 + 0{,}0034 - 0{,}00001) = 2{,}0000;$$

$$y''|_{x=0,0} \approx \frac{1}{h^2}\left(\Delta^2 y_0 - \Delta^3 y_0 + \frac{11}{12}\Delta^4 y_0\right) =$$

$$= 400 \cdot (0{,}00100 - 0{,}00101 + 0{,}00003) = 0{,}008.$$

Для точки $x = 0{,}1$ используем формулы (*c*), считая $x_0 = 0{,}1$:

$$y'|_{x=0,1} \approx \frac{1}{h}\left(\frac{\Delta y_{-1} + \Delta y_0}{2} - \frac{1}{6}\frac{\Delta^3 y_{-2} + \Delta^3 y_{-1}}{2}\right) =$$

$$= 20 \cdot (0{,}10217 - 0{,}00017) = 2{,}0400;$$

$$y''|_{x=0,1} \approx \frac{1}{h^2}\left(\Delta^2 y_{-1} - \frac{1}{12}\Delta^4 y_{-2}\right) = 400 \cdot (0{,}00201 - 0{,}00000) = 0{,}804.$$

Для сравнения приведем точные значения первой и второй производных функции $y = \mathrm{sh}2x$:

$y' = 2\mathrm{ch}2x$: для $x = 0{,}0$: $y' = 2$; а для $x = 0{,}1$: $y' = 2{,}0401$;

$y'' = 4\mathrm{sh}2x$: для $x = 0{,}0$: $y'' = 0$; а для $x = 0{,}1$: $y'' = 0{,}8052$.

Интерполяционный многочлен (9) и его интерпретации (Стирлинга, Гаусса) для вычисления производной в середине и в конце отрезка определения $f(x)$ дают выражение для производной через конечные разности $\Delta^k y$ $(k = 1,2,...)$.

Однако на практике выгоднее иногда выражать значения производных непосредственно через значения y_i.

Ответ на этот вопрос дает интерполяционный многочлен Лагранжа для равномерной сетки интерполяционных узлов.

7.4.2. Вычисление производных на основании многочлена Лагранжа

Запишем интерполяционный многочлен Лагранжа $L(x)$ и его остаточный член $R_L(x)$ для случая трех узлов интерполяции ($n = 2$), но с учетом, что $x_i - x_{i-1} = h = \text{const}$ ($i = 1,2, ..., n$):

$$L(x) = \frac{1}{2h^2}[(x - x_1)(x - x_2)y_0 - 2(x - x_0)(x - x_2)y_1 + (x - x_0)(x - x_1)y_2];$$

$$R_L(x) = \frac{y'''_*}{3!}(x - x_0)(x - x_1)(x - x_2).$$

Найдем их производные:

$$L'(x) = \frac{1}{2h^2}[(2x - x_1 - x_2)y_0 - 2(2x - x_0 - x_2)y_1 + (2x - x_0 - x_1)y_2];$$

$$R'_L(x) = \frac{y'''_*}{3!}[(x - x_1)(x - x_2) + (x - x_0)(x - x_2) + (x - x_0)(x - x_1)].$$

Здесь y'''_* – значение производной в некоторой внутренней точке $x_* \in [x_0, x_n]$.

Запишем выражение для производной y'_0 при $x = x_0$:

$$y'_0 = L'(x_0) + R'_L(x_0) = \frac{1}{2h^2}[(2x_0 - x_1 - x_2)y_0 - 2(2x_0 - x_0 - x_2)y_1 +$$

$$+ (2x_0 - x_0 - x_1)y_2] + \frac{y'''_*}{3!}[(x_0 - x_1)(x_0 - x_2) + (x_0 - x_0)(x_0 - x_2) + (x_0 - x_0)(x_0 - x_1)] =$$

$$= \frac{1}{2h}(-3y_0 + 4y_1 - y_2) + \frac{h^2}{3}y'''_*.$$

Аналогично можно получить значения y'_1, y'_2 при $x = x_1$, $x = x_2$.

Итак, для случая трех узлов ($n = 2$) рабочие формулы имеют следующий вид:

$$\begin{aligned} y'_0 &= \frac{1}{2h}(-3y_0 + 4y_1 - y_2) + \frac{h^2}{3}y'''_* \\ y'_1 &= \frac{1}{2h}(y_2 - y_0) - \frac{h^2}{6}y'''_* \\ y'_2 &= \frac{1}{2h}(y_0 - 4y_1 + 3y_2) + \frac{h^2}{3}y'''_* \end{aligned} \qquad (11)$$

В справочных пособиях приведены формулы Лагранжа для n = 3,4, …. Так для случая четырех узлов (n = 3):

$$y'_0 = \frac{1}{6h}\left(-11y_0 + 18y_1 - 9y_2 + 2y_3\right) - \frac{h^3}{4}y^{IV}{}_*$$

$$y'_1 = \frac{1}{6h}\left(-2y_0 - 3y_1 + 6y_2 - y_3\right) - \frac{h^3}{12}y^{IV}{}_*$$

$$y'_2 = \frac{1}{6h}\left(y_0 - 6y_1 + 3y_2 + 2y_3\right) - \frac{h^3}{12}y^{IV}{}_* \qquad (12)$$

$$y'_3 = \frac{1}{6h}\left(-2y_0 + 9y_1 - 18y_2 + 11y_3\right) - \frac{h^3}{4}y^{IV}{}_*$$

Анализируя (11) и (12) можно утверждать, что, используя значения функции в (n+1) узлах, получают аппроксимацию n-го порядка точности для производной. Эти формулы можно использовать не только для узлов x_0, x_1, x_2, …, но и для любых узлов $x = x_i$, x_{i+1}, x_{i+2}, … с соответствующей заменой индексов в (11) и (12). С помощью многочлена Лагранжа получены аппроксимации и для старших производных.

Таким образом, при n = 3:

$$y''_0 = \frac{1}{h^2}(2y_0 - 5y_1 + 4y_2 - y_3) + O(h^2)_*;$$

$$y''_1 = \frac{1}{h^2}(y_0 - 2y_1 + y_2) + O(h^2)_*;$$

$$y''_2 = \frac{1}{h^2}(y_1 - 2y_2 + y_3) + O(h^2)_*;$$

$$y''_3 = \frac{1}{h^2}(-y_0 + 4y_1 - 5y_2 + 2y_3) + O(h^2)_*; \qquad \text{и т.д.}$$

Аналогичные формулы можно получить и для случая произвольной сетки расположения узлов. Однако в этом случае имеют место неизбежные громоздкие выражения для расчетов производных.

При возникшей необходимости таких расчетов целесообразнее применять искусственный прием, так называемый метод неопределенных коэффициентов.

7.5. Метод неопределенных коэффициентов

В основном используется для случая произвольного расположения интерполяционных узлов. В данном случае искомое выражение k-ой производной в некоторой точке $x = x_i$ представляется в виде линейной комбинации заданных значений функции $y_j = f(x_j)$, в узлах $j = \overline{0,n}$:

$$y_i^{(k)} = c_0 y_0 + c_1 y_1 + ... + c_n y_n, \qquad i = \overline{1,n}\ . \qquad (13)$$

Предполагается, что это соотношение выполняется точно, если $y = f(x)$ является многочленом степени не выше n, т.е. если она может быть представлена в виде:

$$y = b_0 + b_1(x - x_j) + ... + b_n(x - x_j)^n,\ j = \overline{0,n}.$$

Отсюда следует, что соотношение (13) должно выполняться точно для многочленов $y = 1$, $y = x - x_j$, $y = (x - x_j)^2$, $y = (x - x_j)^n$. Производные от них соответственно равны:

$$y' = 0; \qquad y' = 1; \qquad y' = 2(x - x_j), \qquad \ldots, \qquad y' = n(x - x_j)^{n-1}.$$

Подставляя эти выражения в левую и правую части (13), получают систему линейных алгебраических уравнений $(n + 1)$-го порядка для вычисления значений $c_0, c_1, \ldots, c_n$.

Пример. Найти выражение для производной y'_1 в случае четырех узлов (n=3), h = const. Запишем (13) в виде:

$$y'_1 = c_0 y_0 + c_1 y_1 + c_2 y_2 + c_3 y_3.$$

Используем многочлены:

$$y = 1; \qquad y = x - x_0; \quad y = (x - x_0)^2; \qquad y = (x - x_0)^3; \qquad (14)$$

$$y' = 0; \qquad y' = 1; \qquad y' = 2(x - x_0); \qquad y = 3(x - x_0)^2. \qquad (15)$$

Подставим (14) и (15) в искомое уравнение при $x = x_1$

$$0 = c_0 \cdot 1 + c_1 \cdot 1 + c_2 \cdot 1 + c_3 \cdot 1;$$

$$1 = c_0(x_0 - x_0) + c_1(x_1 - x_0) + c_2(x_2 - x_0) + c_3(x_3 - x_0);$$

$$2(x_1 - x_0) = c_0(x_0 - x_0)^2 + c_1(x_1 - x_0)^2 + c_2(x_2 - x_0)^2 + c_3(x_3 - x_0)^2;$$

$$3(x_1 - x_0)^2 = c_0(x_0 - x_0)^3 + c_1(x_1 - x_0)^3 + c_2(x_2 - x_0)^3 + c_3(x_3 - x_0)^3.$$

Получаем после преобразования:

$$\begin{cases} c_0 + c_1 + c_2 + c_3 = 0; \\ hc_1 + 2hc_2 + 3hc_3 = 1; \\ hc_1 + 4hc_2 + 9hc_3 = 2; \\ hc_1 + 8hc_2 + 27hc_3 = 3. \end{cases}$$

Решение полученной системы алгебраических уравнений дает следующие значения:

$$c_0 = -\frac{1}{3h}; \qquad c_1 = -\frac{1}{2h}; \qquad c_2 = \frac{1}{h}; \qquad c_3 = -\frac{1}{6h};$$

$$y'_1 = \frac{1}{6h}\left(-2y_0 - 3y_1 + 6y_2 - y_3\right).$$

Это тождественно соотношению (12) для y'_1, только без указания теоретической погрешности.

7.6. Улучшение аппроксимации при численном дифференцировании

Из рассмотренных выше конечноразностных соотношений для определения производных видно, что порядок их точности прямо пропорционален числу узлов интерполяции. Однако с увеличением числа интерполяционных точек увеличивается объем вычислений, усложняется оценка их точности. Для устранения этого разработан простой и эффективный способ уточнения решения при конечном числе узлов при конечно-разностном подходе – ***метод Рунге-Ромберга***.

Пусть $F(x)$ производная, подлежащая аппроксимации, а $f(x,h)$ – ее конечно-разностная аппроксимация на равномерной сетке с шагом h. Тогда остаточный член аппроксимации можно записать в следующем виде:

$$R = h^{\rho}\varphi(x) + O(h^{\rho+1}),$$

где первый член является главной частью погрешности. Значение производной примет вид

$$F(x) = f(x,h) + h^{\rho}\varphi(x) + O(h^{\rho+1}). \qquad (16)$$

Запишем (16) в той же точке, но с другим шагом $h_1 = kh$, тогда:

$$F(x) = f(x,kh) + (kh)^{\rho}\varphi(x) + O[(kh)^{\rho+1}]. \qquad (17)$$

Приравнивая правые части (16) и (17) находим выражения для определения главного члена погрешности.

$$h^{\rho}\varphi(x) \approx \frac{f(x,h) - f(x,kh)}{k^{\rho} - 1} + O(h^{\rho+1}). \qquad (18)$$

Подставляя (18) в (16) получим рабочую формулу:

$$F(x) = f(x,h) + \frac{f(x,h) - f(x,kh)}{k^{\rho} - 1} + O(h^{\rho+1}). \qquad (19)$$

Данная формула позволяет по результатам двух расчетов значений производной с шагом h и kh повысить порядок точности от h^{ρ} до $h^{\rho+1}$.

Пример. Вычислить производную от $y = x^3$ для $x = 1$. Очевидно, что ее точное значение $y(1) = 3$. Составим таблицу значений этой функции в окрестности заданной точки ($x = 1$):

x	0,8	0,9	1,0
y	0,512	0,729	1,0

Воспользуемся аппроксимацией с помощью левых разностей с порядком $\rho = 1$. Примем $h_1 = 0{,}1$; $h_2 = 0{,}2$; т.е. $k = 2$

$$f(x, h) = y'(1;\ 0{,}1) = \frac{y(1) - y(0{,}9)}{0{,}1} = 2{,}71;$$

$$f(x, kh) = y'(1;\ 0{,}2) = \frac{y(1) - y(0{,}8)}{0{,}2} = 2{,}44;$$

Тогда

$$F(x) = y'(1) = 2{,}71 + \frac{2{,}71 - 2{,}44}{2^1 - 1} = 2{,}98.$$

Есть подходы к решению данной задачи для общего случая, когда для уточнения решения используется h_1, h_2,…, h_g шагов. Для этого необходимо, чтобы исходная функция имела производные высших порядков.

Замечания

1. Как видно из выше изложенного, что порядок точности по полученным формулам для численного дифференцирования по отношению к шагу сетки равен числу узлов интерполяции минус порядок производной. Поэтому минимальное число узлов интерполяции, необходимое для вычисления m-ой производной должен быть равным m+1.

2. Из практических соображений рекомендуется использовать для расчетов 4–6 интерполяционных узла. Тогда при хорошо составленной сетке хорошая точность достигается при вычислении первой или второй производных, удовлетворительная точность достигается для 3 и 4 производных. Для более высоких порядков производных данная сетка не применима.

3. С ростом порядка m обычно резко падает точность численного дифференцирования, и поэтому эти формулы для вычисления производных выше второго порядка используются редко.

Раздел 8. Обыкновенные дифференциальные уравнения

8.1. Постановка задачи

Различные задачи во многих областях науки и техники при их математическом моделировании сводятся к дифференциальным уравнениям

Дифференциальными уравнениями называются такие уравнения, которые, кроме неизвестных функций одного или нескольких независимых переменных, содержат также и их производные. Дифференциальные уравнения (ДУ) называют *обыкновенными* (ОДУ), если неизвестные функции являются функциями одного переменного, в противном случае ДУ называться уравнениями *в частных производных*.

Соотношение

$$F(x, y, y', y'', \ldots y^{(n)}) = 0, \qquad (1)$$

связывающее переменную x, неизвестную функцию $Y = Y(x)$ и ее производные до порядка (n) включительно, называют ОДУ n-го порядка.

Решить уравнение (1) – значит найти функциональную зависимость $Y=Y(x)$ превращающую ее в тождество.

Для практической реализации из общей записи ДУ (1) стараются выразить старшую производную, так для $n = 1$ соотношение (1) примет вид:

$$Y' = f(x, Y); \qquad (2)$$

$$Y'' = f(x, Y, Y'), \text{ если } n = 2.$$

Общее решение уравнения (1) имеет вид:

$$Y = Y(x, c_1, c_2, c_3, \ldots, c_n),$$

где c_i – произвольные постоянные.

Если из каких-то условий задать c_i, то получают частное искомое решение:

$$Y = Y(x, c_{10}, c_{20}, c_{30}, \ldots, c_{n0}). \qquad (3)$$

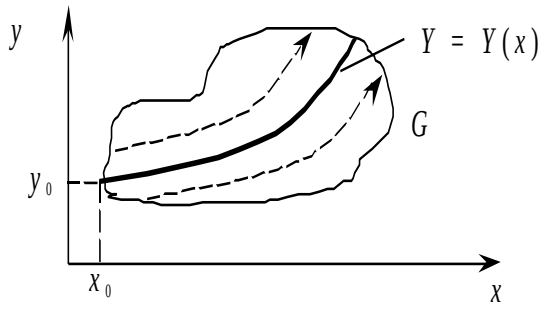
В зависимости от способа задания этих условий различают две задачи для ОДУ:

1) задача Коши;

2) краевая задача.

В качестве дополнительных условий могут задаваться значения искомой функции или ее производных. Если условия задаются в одной точке отрезка определения $Y(x) \in [a, b]$ и, как правило, в его начале $x = x_0 = a$ – это задача Коши с начальной точкой. Если дополнительные условия задаются в точках $x = a$ и $x = b$ – это краевая задача с граничными условиями. Общим решением для ДУ первого порядка будет $Y = f(x, c)$, частным решением будет $Y = f(x, c_0)$.

Дадим геометрическую интерпретацию ДУ первого порядка из (2). Его решение можно изобразить в виде семейства кривых на плоскости X_0Y:



Пусть неявная функция $f(x,y)$, правая часть уравнения (2), определена и непрерывна в области G этой плоскости. В каждой точке плоскости G функция $f(x,y)$ задает некоторое направление. В целом это будет поле направлений. Для общего решения требуется найти все интегральные кривые, касательные к которым в каждой точке совпадают с направлением поля. А решение $Y = Y(x)$ являться частным решением соответствующим какой-то постоянной. Через каждую точку из области решения проходит одна интегральная кривая. Для ДУ при $n > 1$ через каждую точку проходит не одна интегральная кривая и нужно n дополнительных условий, т.е. для уравнений высших порядков геометрическая интерпретация их решений более сложная. А найти общее решение в ана-

литическом виде удается даже для ДУ первого порядка только в редких случаях. Частное решение тоже приходиться искать приблизительно.

Методы решения ОДУ можно разбить на следующие группы:

– графические;

– аналитические;

– приближенные аналитические;

– численные.

Первые три метода рассмотрены в курсе дифференциальных уравнений.

Численные методы являются основным инструментом при решении научно-технических задач посредством ЭВМ.

Наиболее распространенными численными методами решения дифференциальных уравнений является ***методы конечных разностей***, сущность которых состоит в замене области непрерывного изменения аргумента (например, отрезок) дискретным множеством точек, называемых узлами. Эти узлы составляют разностную сетку. На ней искомая функция непрерывного аргумента заменяется приближенной функцией дискретного аргумента, т.е. решение ДУ сводиться к отысканию значений сеточной функции в узлах сетки. Численные методы – это алгоритмы вычисления приближенных значений искомой функции. Решение получается в виде таблицы. Они не позволяют найти общее решение в принципе.

С их помощью можно определить лишь частное решение, но они применимы к широким классам уравнений и всем типам задач. Следует заметить, что как во всех задачах о приближениях здесь также требуются исследования на корректность и точность решений. Это подробно рассматривается в специальной литературе. Рассматриваемые ниже численные методы предполагают изначальное обеспечение этих двух компонент искомого решения

8.2. Задача Коши для ОДУ

В зависимости от вида ДУ (1) задача Коши формируется следующим образом.

1. Если $n = 1$, то требуется найти $Y = Y(x)$, удовлетворяющую уравнению:

$$\frac{dY}{dX} = f(x, Y) \tag{4}$$

и принимающую при $x = x_0$ заданное значение Y_0:

$$Y(x_0) = Y_0 \, . \tag{5}$$

Для определенности будем считать, что решение нужно получить для значений $x > x_0$. В качестве начального значения может быть произвольное x, но чаще всего принимают $x_0 = 0$, что не влияет на разработку численного метода для (4). Заметим, что все численные методы разработаны для решения ОДУ именно первого порядка.

2. Задача Коши для ОДУ n-го порядка

$$Y^{(n)} = f(x, Y, Y', ..., Y^{(n-1)}); \tag{6}$$

найти $Y = Y(x)$, удовлетворяющую (6) и начальным условиям

$$Y(x_0) = Y_0, \quad Y'(x_0) = Y_0', \ \ldots, \quad Y^{(n-1)}(x_0) = Y_0^{(n-1)}; \tag{7}$$

где $Y_0, Y'_0, ..., Y_0^{(n-1)}$ – есть заданные числа.

3. Задача Коши для системы ДУ:

$$\begin{cases} \dfrac{dY_1}{dx} = f_1(x, Y_1, Y_2, ..., Y_n); \\ \dfrac{dY_2}{dx} = f_2(x, Y_1, Y_2, ..., Y_n); \\ \qquad \ldots \\ \dfrac{dY_n}{dx} = f_n(x, Y_1, Y_2, ..., Y_n). \end{cases} \tag{8}$$

Задача Коши для системы (8) заключается в отыскании $Y_i(x)$ $(i = \overline{1,n})$, удовлетворяющих (8) и начальным условиям:

$$Y_1(x_0) = Y_{10}; \ \ Y_2(x_0) = Y_{20}; \ \ldots; \ \ Y_n(x_0) = Y_{n0}. \tag{9}$$

Численные методы для решения ОДУ (4) и (5) применяются и для решения (8) и (9).

Дифференциальное уравнение n-го порядка (6) может быть приведено к системе (8) путем введения новых неизвестных функций $Y_i(x), i = \overline{1, n-1}$:

$$y_1 = y', y_2 = y'', \ \ldots, y_{n-1} = y^{(n-1)}. \tag{10}$$

Тогда (6) запишется следующим образом

$$\begin{cases} \dfrac{dY}{dx} = Y_1; \\ \dfrac{dY_1}{dx} = Y_2; \\ \qquad \ldots \\ \dfrac{dY_{n-2}}{dx} = Y_{n-1}; \\ \dfrac{dY_{n-1}}{dx} = f(x, Y_1, Y_2, ..., Y_{n-1}). \end{cases}$$

Если удается найти общее решение для (4), (6), или системы (8), то задача Коши сводится к отысканию значений произвольных постоянных. Как правило, она решается приближенно.

8.3. Численные методы решения задачи Коши

Для решения задачи Коши (4) и (5) по технологии разностных методов введем последовательность точек $x_0, x_1, ..., x_n$ и шаги $h_i = x_{i+1} - x_i$ ($i = 0,1,..., n-1$). В каждом узле x_i вместо значений функции $Y(x_i)$ вводятся числа y_i, как результат аппроксимации точного решения $Y(x)$ на данном множестве точек. Функцию y, заданную в виде таблицы $\{x_i, y_i\}$ называют сеточной функцией. Заменяя значение производной в уравнении (4) отношением конечных разностей осуществляем переход от дифференциальной задачи (4), (5) относительно функции $Y(x)$ к разностной задаче относительно сеточной функции

$$y_{i+1} = F(x_i, h_i, y_{i+1}, y_i, ...y_{i-k+1}), \quad i = 1,2,...; \tag{11}$$

$$y_0 = Y_0. \tag{12}$$

Это разностное уравнение в общем виде, а конкретное выражение правой части для (11) зависит от способа аппроксимации производной. Для каждого численного метода получается свой вид уравнения (11).

Если в правой части уравнения (11) отсутствует y_{i+1}, т.е. значение y_{i+1} вычисляется по k предыдущим значениям $y_i, y_{i-1}, ..., y_{i-k+1}$, , то разностная схема называется явной. При этом имеет место k-шаговый метод: $k = 1$ – одношаговый, $k=2$ – двухшаговый и т.д., т.е. в одношаговых методах для вычисления y_{i+1} используется лишь одно найденное значение на предыдущем шаге y_i, в многошаговом – многие из них.

Если y_{i+1} входит в правую часть (11), то это будут неявные методы, реализация которых носит только итерационный характер.

8.3.1. Одношаговые методы решения задачи Коши

Простейшими численными методами для решения задачи Коши для ОДУ являются следующие методы.

1. Метод Эйлера

Этот метод основан на разложении искомой функции $Y(x)$ в ряд Тейлора в окрестностях узлов системы $x = x_i$ ($i = 0, 1, 2, \ldots, n$), в котором отбрасываются все члены, содержащие производные второго и более высоких порядков. Как правило, используется равномерная сетка $\Delta x = x_{i+1} - x_i = h = \text{const}$ ($i = \overline{0,n}$). Разложение запишем в виде

$$Y(x_i + \Delta x) = Y(x_i) + Y'(x_i) \cdot \Delta x_i + O(\Delta x_i^2). \tag{13}$$

Заменяя значение функции $Y(x)$ в узлах сетки x_i значениями сеточной функции и используя уравнение (4), получим

$$Y'(x_i) = f(x_i, Y(x_i)) = f(x_i, y_i).$$

Тогда из (13) получим

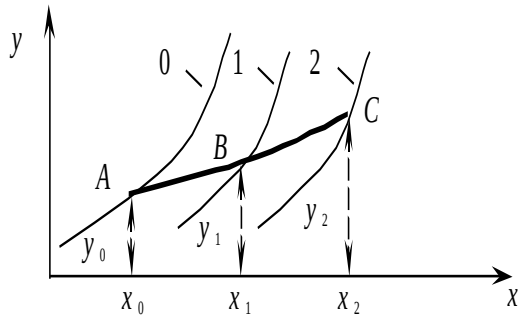
$$y_{i+1} = y_i + h \cdot f(x_i, y_i); \quad i = 0,1,2,..., n-1. \tag{14}$$

При $i = 0$, для узла $x = x_1$: $y_1 = y_0 + h \cdot f(x_0, y_0)$.

Далее по алгоритму (14)

$$y_2 = y_1 + h \cdot f(x_1, y_1);$$
$$\ldots$$
$$y_n = y_{n-1} + h \cdot f(x_{n-1}, y_{n-1}).$$

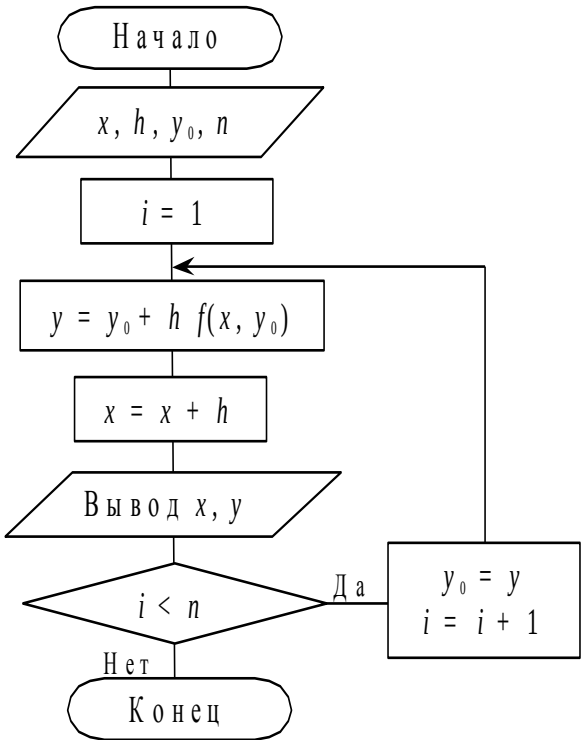
Геометрическая интерпретация имеет вид:



На рисунке линия «0» – точное решение, линии «1» и «2» – приближенные решения.

Искомая интегральная кривая $y(x)$, проходящая через точку (x_0, y_0), заменяется ломаной с вершинами в точках (x_i, y_i). Каждое звено ломаной имеет направление, совпадающее с направлением интегральной кривой (4), которая проходит через точку (x_i, y_i).

Блок-схема алгоритма будет иметь следующий вид:



Вывод полученных результатов выполняется на каждом шаге, но если необходимо сохранить результаты, то следует ввести массив значений $y_0, y_1, \ldots, y_n$.

Локальная погрешность метода Эйлера, как видно из (13), оценивается, как $O(h^2)$. Весь интервал $[a, b]$ разбивается на n частей, тогда общая погрешность

$$n\,O(h^2) = \frac{1}{h}O(h^2) = O(h) \qquad \text{– 1-й порядок.}$$

Для оценки погрешности при машинном расчете пользуются двойным просчетом, т.е. на отрезке $[x_i, x_{i+1}]$ расчет повторяют с шагом $h/2$ и погрешность более точного решения y^*_{i+1} (при шаге $h_i/2$) оценивается как разность $\left|y^*_{i+1} - y_{i+1}\right|$.

2. Метод Эйлера с пересчетом

При данном подходе рекуррентное соотношение (14) видоизменяется, а именно, вместо $f(x_i, y_i)$ берут среднее арифметическое между $f(x_i, y_i)$ и $f(x_{i+1}, y_{i+1})$.

Тогда

$$y_{i+1} = y_i + \frac{h}{2}\left[f(x_i, y_i) + f(x_{i+1}, y_{i+1})\right], \quad i = 0,1,... \tag{15}$$

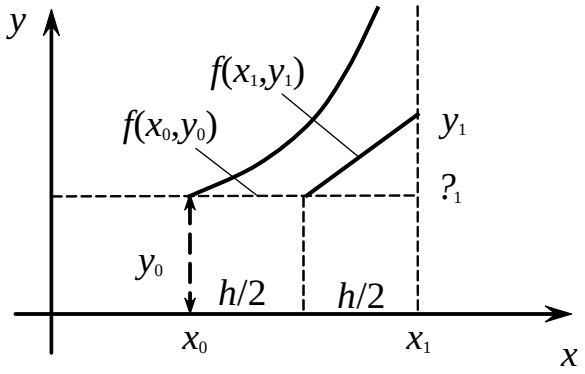
Это неявная схема. Она реализуется в две итерации: сначала находится первое приближение по (14), считая y_i начальной

$$\tilde{y}_{i+1} = y_i + hf(x_i, y_i), \tag{16}$$

затем (16) подставляется в правую часть (15) вместо y_{i+1}

$$y_{i+1} = y_i + \frac{h}{2}\left[f(x_i, y_i) + f(x_{i+1}, \tilde{y}_{1+i})\right], \qquad i = 0,1,... \tag{17}$$

Геометрическая интерпретация метода:



С помощью метода Эйлера с пересчетом можно производить контроль точности, сравнивая y_{i+1} и $\tilde{y}_{i+1}$.

На основании этого можно выбирать шаг. Если величина $|\tilde{y}_{i+1} - y_{i+1}|$ сравнима с заданной точностью ε, то шаг можно увеличивать, если больше, то уменьшать, т.е. имеет место схема двойного просчета с оценкой погрешности по величине

$$\frac{1}{3}\left|y^*_i - y_i\right| \approx \left|y^*_i - y(x_i)\right|,$$

где $y(x_i)$ – точное решение в точке $x = x_i$, а y_i и y^*_i приближенные значения, полученные с шагом h и $h/2$ соответственно

3. Метод Эйлера с последующей итерационной обработкой

Метод Эйлера можно еще более уточнить, применяя итерационную обработку каждого полученного значения y_i. А именно, сначала исходя из первого грубого приближения по (16)

$$y_{i+1}^{(0)} = y_i + h\, f(x_i, y_i) \quad ,$$

строят итерационный процесс согласно (15) по следующей схеме

$$y_{i+1}^{(k)} = y_i + \frac{h}{2}[\, f(x_i, y_i) + f(x_{i+1}, y_{i+1}^{(k-1)}) \,. \qquad (18)$$

Итерации продолжают до тех пор, пока в двух последовательных приближениях y_{i+1}^{k}, $y_{i+1}^{(k+1)}$ не совпадут соответствующие десятичные знаки и полагают $y_{i+1} \approx y_{i+1}^{(k+1)}$. Как правило, при достаточно малом шаге h, итерации сходятся быстро. Если после трех-четырех итераций не произошло совпадение нужного числа десятичных знаков, то шаг расчетов h уменьшается. После такой обработки значения y_i переходят к следующему узлу x_{i+1}.

Пример. По методу Эйлера составить таблицу решения на отрезке [0;1] для уравнения $y' = y - \dfrac{2x}{y}$ с начальным условием $y(0) = 1$, выбрав шаг $h = 0{,}2$.

Результаты вычислений поместим в таблицу, которая заполняется следующим образом:

i	x_i	y_i	Δy_i	Точное $y = \sqrt{2x+1}$
0	0	1,0000	0,2000	1,0000
1	0,2	1,2000	0,1733	1,1832
2	0,4	1,3733	0,1561	1,3416
3	0,6	1,5294	0,1492	1,4832
4	0,8	1,6786	0,1451	1,6124
5	1,0	1,8237		1,7320

В первой строке при $i = 0$ записывается $x_0 = 0$, $y_0 = 1{,}000$ и по ним вычисляется $f(x_0, y_0) = 1$, а затем $\Delta y_0 = hf(x_0, y_0) = 0{,}2$. Тогда по формуле (14) получаем $y_1 = 1 + 0{,}2 = 1{,}2$.

Значения $x_1 = 0{,}2$ и $y_1 = 1{,}2000$ записываются во второй строке при $i = 1$. Используя их можно вычислить

$$f(x_1,y_1) = 0{,}8667;\ \Delta y_1 = hf(x_1,y_1) = 0{,}2 \cdot 0{,}8667 = 0{,}1733.$$

Тогда $y_2 = y_1 + \Delta y_1 = 1{,}2 + 0{,}1733 = 1{,}3733$.

При $i = 2,3,4,5$ вычисления ведутся аналогично. В последнем столбце таблицы для сравнения помещены значения ***точного*** решения.

Из таблицы видно, что абсолютная погрешность для y_5 составляет $\varepsilon = 1{,}8237 - 1{,}7320 = 0{,}0917$, что составляет 5%.

Замечание. Метод Эйлера легко распространяется на системы дифференциальных уравнений и на ДУ высших порядков при их предварительном приведении к системам ДУ первого порядка.

Рассмотрим систему двух уравнений первого порядка

$$\begin{cases} y' = f_1(x, y, z); \\ z' = f_2(x, y, z); \end{cases} \tag{19}$$

с начальными условиями $y(x_0) = y_0$ и $z(x_0) = z_0$.

Тогда приближенные значения $y(x_i) \approx y_i$ и $z(x_i) \approx z_i$ вычисляются по формулам

$$\left.\begin{aligned} y_{i+1} &= y_i + hf_1(x_i, y_i, z_i)\,, \\ z_{i+1} &= z_i + hf_2(x_i, y_i, z_i), \quad i = 0,1,2,... \end{aligned}\right\} \tag{20}$$

Пример. Применяя метод Эйлера, составить на отрезке [1;1,5] таблицу значений решения уравнения

$$y'' + \frac{y'}{x} + y = 0 \tag{21}$$

с начальными условиями $y(1) = 0{,}77$ и $y'(1) = -0{,}44$, выбрав шаг $h = 0{,}1$.

Решение. Заменим уравнение (21) посредством подстановки $y' = z$, $y'' = z'$ системой уравнений первого порядка

$$\begin{cases} y' = z; \\ z' = -\dfrac{z}{x} - y; \end{cases}$$

с начальными условиями $y(1) = 0{,}77$ и $z(1) = -0{,}44$. Таким образом, имеем

$$\begin{cases} f_1(x, y, z) = z; \\ f_2(x, y, z) = -\dfrac{z}{x} - y. \end{cases}$$

Результаты вычисления по формулам (20) записаны в таблице

i	x_i	y_i	ΔY_i	$f_{1i} = z_i$	Δz_i	$f_{2i} = -\dfrac{z_i}{x_i} - z_i$
0	1,0	0,77	–0,044	–0,44	–0,033	–0,33
1	1,1	0,726	–0,047	–0,473	–0,030	–0,296
2	1,2	0,679	–0,050	–0,503	–0,026	–0,260
3	1,3	0,629	–0,053	–0,529	–0,022	–0,222
4	1,4	0,576	–0,055	–0,551		
5	1,5	0,521				

Таблица заполняется следующим образом. Записываем в первой строке $i = 0$, x_0=1,0; $y_0 = 0{,}77$; $z_0 = -0{,}44$.

Далее находим

$$f_{10} = f_1(x_0, y_0, z_0) = z_0 = -0{,}44;$$

$$f_{20} = f_2(x_0, y_0, z_0) = -\frac{z_0}{x_0} - y_0 = -0{,}33.$$

Используя формулы (20) получаем

$$\Delta y_0 = hf_{10} = 0{,}1 \cdot (-0{,}44) = -0{,}044; \qquad y_1 = y_0 + \Delta y_0 = 0{,}726;$$

$$\Delta z_0 = hf_{20} = 0{,}1 \cdot (-0{,}33) = -0{,}033; \qquad z_1 = z_0 + \Delta z_0 = -0{,}473.$$

Таким образом, во второй строке таблицы мы можем записать $i = 1$; $x_1 = 1{,}1$; $y_1 = 0{,}726$; $z_1 = -0{,}473$. По этим значениям находим

$$f_{11} = f_1(x_1, y_1, z_1) = z_1 = -0{,}473;$$

$$f_{21} = f_2(x_1, y_1, z_1) = -\frac{z_1}{x_1} - y_1 = -0{,}296.$$

И, следовательно,

$$\Delta y_1 = hf_{11} = 0{,}1 \cdot (-0{,}47) = -0{,}047; \qquad y_2 = y_1 + \Delta y_1 = 0{,}679;$$

$$\Delta z_1 = hf_{21} = 0{,}1 \cdot (-0{,}30) = -0{,}030; \qquad z_2 = z_1 + \Delta z_1 = -0{,}503.$$

Заполнение таблицы при i =2, 3, 4, 5 производятся аналогично.

4. Метод Рунге-Кутта

На его основе могут быть построены разностные схемы разного порядка точности. Идея его реализации стоит в подгонке ряда Тейлора при разложении искомой функции $y = y(x)$ в окрестностях узлов сетки в плане повышения точности этого разложения, а именно, увеличение числа производных высшего порядка без их непосредственного определения из-за сложности аналитических выражений полных производных по x от функции $f(x,y)$.

Рассмотрим наиболее широко применяемую на практике разностную схему четвертого порядка.

Ее алгоритм состоит в следующем:

$$\left.\begin{aligned} y_{i+1} &= y_i + \Delta Y_i; \\ \Delta Y_i &= \frac{1}{6}(k_1^{(i)} + k_2^{(i)} + k_3^{(i)} + k_4^{(i)}); \end{aligned}\right\} \quad i = 0,1,2,... \qquad (22)$$

где $k_1^{(i)} = hf(x_i, y_i)$; $\qquad k_2^{(i)} = hf(x_i + h/2, y_i + k_1^{(i)}/2)$;

$k_3^{(i)} = hf(x_i + h/2, y_i + k_2^{(i)}/2)$; $k_4^{(i)} = hf(x_i + h,\ y_i + k_3^{(i)})$.

В данной расчетной схеме Рунге-Кутта на каждом шаге вычисления y_i нужно 4-е раза обратиться к правой части уравнения $f(x, y)$, т.е. метод Рунге-Кутта (22) требует бóльшего объема вычислений, однако это окупается повышенной точностью, что позволяет проводить расчет с большим шагом.

Можно показать, что метод Эйлера и его модифицированный вариант является аналогом метода Рунге-Кутта первого и второго порядка, однако для достижения одинаковой точности у них шаг расчета будет значительно меньше.

Для данного метода шаг расчета можно менять при переходе от одной точке к другой. Для контроля правильности выбора шага h рекомендуется вычислять дробь

$$Q = \left| \frac{K_2^{(i)} - K_3^{(i)}}{K_1^{(i)} - K_2^{(i)}} \right|.$$

Величина Q не должна превышать нескольких сотых. В противном случае h следует уменьшать.

Оценка погрешности метода затруднительна. Чаще всего используется грубая оценка погрешности по формуле $\left| y_n^* - y(x_n) \right| \approx \frac{\left| y_n^* - y_n \right|}{15}$, где $y(x_n)$ – значение точного решения уравнения (4) в точке x_n, а y_n^*, y_n – приближенное решение, полученное с шагом $h/2$ и h.

При реализации (на ЭВМ) метода Рунге-Кутта с автоматическим выбором шага, обычно в каждой точке x_i и делают двойной просчет сначала с шагом h, потом с $h/2$. Если полученное y_i при этом различается в пределах допустимой точности, то шаг h для следующей точки x_{i+1} удваивают, в противном случае берут половинный шаг.

В заключении следует отметить, что одношаговые методы Рунге-Кутта успешно могут быть применены к решению систем ДУ первого порядка.

8.3.2. Многошаговые методы решения задачи Коши

В данном случае построение разностных расчетных схем (11) основано на том, что для определения y_{i+1} используются результаты не одного, а k предыдущих шагов y_{i-k+1}, y_{i-k+2},..., y_i в данном случае это k шаговый метод.

Многошаговые методы могут быть построены следующим образом. Исходное уравнение (4) для задачи Коши запишем в виде $dY(x) = f(x,y)dx$. Проинтегрируем обе части этого соотношения на отрезке $[x_i, x_{i+1}]$.

Из левой части получаем

$$\int_{x_i}^{x_{i+1}} dY(x) = Y(x_{i+1}) - Y(x_i) \approx y_{i+1} - y_i, \qquad (23)$$

где y_{i+1}, y_i – сеточные значения искомой функции. Для вычисления интегралов правой части сначала построим интерполяционный многочлен $P_{k-1}(x)$ степени $(k-1)$ для функции $f(x,Y)$ на этом отрезке по значениям $f(x_{i-k+1},Y_{i-k+1})$, $f(x_{i-k+2},Y_{i-k+2})$, ..., $f(x_i,Y_i)$. Тогда

$$\int_{x_i}^{x_{i+1}} f(x,Y)dx \approx \int_{x_i}^{x_{i+1}} P_{k-1}(x)dx \ . \qquad (24)$$

Приравнивая (23) и (24) получает формулу для определения неизвестного значения сеточной функции y_{i+1} в узле x_{i+1}

$$y_{i+1} = y_i + \int_{x_i}^{x_{i+1}} P_{k-1}(x)dx. \qquad (25)$$

На основе (25) можно строить различные многошаговые методы любого порядка точности. Порядок точности при этом зависит от степени $P_{k-1}(x)$, для построения которого используются значения сеточной функции y_i, y_{i-1},..., y_{i-k+1}, вычисленные на k предыдущих узлах.

На практике широко используются следующие многошаговые методы.

1. Семейство методов Адамса

Известны методы Адамса k-го порядка. Простейший из них при $k = 1$ повторяет метод Эйлера первого порядка в точности. Метод четвертого порядка на практике принято называть методом Адамса. Рабочую формулу для него получают следующим образом.

Пусть известные в четырех последовательных узлах ($k = 4$) значение сеточной функции y_{i-3}, y_{i-2}, y_{i-1}, y_i и вычисленные первоначально значения правой части (4) f_{i-3}, f_{i-2}, f_{i-1}, f_i. В качестве интерполяционного многочлена $P_3(x)$ возьмем многочлен Ньютона. В случае h = const конечные разности для правой части в узле x_i будут иметь вид

$$\Delta f_i = f_i - f_{i-1};$$

$$\Delta^2 f_i = f_i - 2 f_{i-1} + f_{i-2};$$

$$\Delta^3 f_i = f_i - 3 f_{i-1} + 3 f_{i-2} - f_{i-3}.$$

Тогда разностная схема метода Адамса запишется в виде

$$y_{i+1} = y_i + hf_i + \frac{h^2}{2}\Delta f_i + \frac{5h^3}{12}\Delta^2 f_i + \frac{3h^4}{8}\Delta^3 f_i. \qquad (26)$$

По сравнению с методом Рунге-Кутта той же точности можно отметить его экономичность, так как (26) предусматривает на каждом шаге только один раз вычисление правой части в соотношении (4). Однако расчет здесь можно начать только с узла x_4. Значения y_1, y_2, y_3 необходимые для вычисления y_4 нужно определять одношаговым методом, что несколько усложняет алгоритм вычисления. Кроме того, метод Адамса не позволяет изменять шаг h в процессе счета, что доступно для одношаговых методов.

2. Многошаговые методы, использующие неявные разностные схемы

На практике они называются методами прогноза и коррекции или (методами предиктор-корректор).

Суть их состоит в том, что на каждом шаге расчета вводятся 2 этапа, использующие многошаговые методы:

а) с помощью явного метода (предиктора) по известным значениям функции в предыдущих узлах находится начальное значение $y_{i+1} = y_{i+1}^{(0)}$ в новом узле;

б) используя неявный метод (корректор) в результате итераций находятся приближения $y_{i+1}^{(1)}, y_{i+1}^{(2)},..., y_{i+1}^{(k)}, y_{i+1}^{(k+1)},...$ Посредством корректора итерации продолжаются до тех пор, пока y_{i+1}^{k} и $y_{i+1}^{(k+1)}$ не совпадут по желаемой точности и затем осуществляется переход к следующей точке сетки, т.е. по рассмотренному выше алгоритму определяется значение y_{i+2}. Одним из вариантов метода прогноза и коррекции является метод на основе метода Адамса четвертого порядка.

Вид разностных соотношений на этапе предиктора

$$y_{i+1} = y_i + \frac{h}{24}(55 f_i + 59 f_{i-1} + 37 f_{i-2} - 9 f_{i-3}); \qquad (27)$$

на этапе корректора

$$y_{i+1} = y_i + \frac{h}{24}(9 f_{i+1} + 19 f_i - 5 f_{i-1} + f_{i-2}). \qquad (28)$$

В (27) и (28) используются не Δf_i (конечные разности), а значения правой части (4), что удобнее для реализации на ЭВМ. Явная схема (27) используется на каждом шаге лишь один раз, а с помощью неявной схемы (28) строится итерационный процесс вычислений y_{i+1}, поскольку это значение входит в правую часть выражения $f_{i+1} = f(x_{i+1}, y_{i+1})$.

В данных формулах, как и в случае метода Адамса, при вычислении y_{i+1} необходимы значения сеточной функции в четырех предыдущих узлах: y_{i-3}, y_{i-2}, y_{i-1}, y_i. Расчет по этому методу может быть начат только со значения y_4.

Необходимые при этом значения y_1, y_2 и y_3 находятся по методу Рунге-Кутта, y_0 задается начальным условием.

Метод Адамса легко распространяется на системы дифференциальных уравнений, а также на дифференциальных уравнений n-го порядка.

3. Повышение точности результатов

Точность можно повысить путем уменьшения значения шага h. Но этот путь ограничен требованием экономичности, поскольку это может потребовать огромного объема вычислений.

На практике часто для повышения точности численного решения без существенного увеличения машинного времени используется *метод Рунге*. Его суть состоит в том, что по одной и той же разностной схеме проводятся повторные расчеты с различными шагами. В соответствие с методом Рунге уточненное значение y_h^* сеточной функции в узлах сетки с шагом h вычисляется по формуле $y_h^* = \frac{2^k y_{h/2} - y_h}{2^k - 1} + O(h^{k+1})$.

Порядок точности этого решения равен (k + 1), хотя используемая разностная схема имеет порядок точности k, т.е. точность повышается на порядок.

Литература

1. Самарский А.А., Гулин А.В. численные методы, физматгиз, Москва, 1989.
2. Березин И.С., Жидков Н.П. метод вычислений, том1. Физматгиз, Москва 1966.
3. Демидович Б.П., Марон И.А. Основы вычислительной математики. Физматгиз, Москва 1968.
4. Демидович Б.П., Марон И.А. основы вычислительной математики. Физматгиз, Москва 1968.
5. Калиткин Н.Н.численные методы. Физматгиз, Масква,1978.
6. Бахвалов Н.С., Жидков Н.П., Кобельков Г.М. численные методы.
7. Юдин Д.Б., Гольдитейн Е.Т. Линейное программирование. Физматгиз, Масква 1969.

– 130 –

– 131 –

– 132 –