

Министерство науки и высшего образования Российской Федерации

Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Оренбургский государственный университет»

ВЫЧИСЛИТЕЛЬНАЯ МАТЕМАТИКА И ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ

Учебное пособие

Рекомендовано ученым советом федерального государственного бюджетного образовательного учреждения высшего образования «Оренбургский государственный университет» для обучающихся по образовательным программам высшего образования по направлениям подготовки 01.04.02 Прикладная математика и информатика, 02.04.02 Фундаментальная информатика и информационные технологии и 09.04.02 Информационные системы и технологии

Оренбург
2021

УДК 519.6:004(075.8)
ББК 22.19я73+32.97я73
В 94

Рецензент – доцент, кандидат технических наук Д.В. Горбачев
Авторы: И.П. Болодурина, Л.М. Анциферова, Д.И. Парфенов,
Л.С. Гришина

В 94 Вычислительная математика и информационные технологии: учебное пособие / И.П. Болодурина, Л.М. Анциферова, Д.И. Парфенов, Л.С. Гришина; Оренбургский гос. ун-т. – Оренбург: ОГУ, 2021 – 132 с.

ISBN 978-5-7410-2572-7

В учебном пособии представлены основные понятия, определения и подходы, используемые в современных задачах вычислительной математики. Описаны методы и алгоритмы численного решения задач, возникающих при принятии решений в социальных, экономических и технических системах. Кроме того, рассмотренные численные алгоритмы позволяют совершенствовать методы получения и обработки информации в различных прикладных задачах. Учебное пособие включает примеры применения численных методов для моделирования эпидемий с различными ограничениями, а также моделирование распространения атак в сетях. Разработанные программные средства позволяют обеспечивать надежность и своевременную диагностику функционирования вычислительных машин, комплексов и компьютерных сетей.

Учебное пособие предназначено для обучающихся по образовательным программам высшего образования по направлению подготовки 01.04.02 Прикладная математика и информатика, 02.04.02 Фундаментальная информатика и информационные технологии и 09.04.02 Информационные системы и технологии, а также для других направлений и специальностей, изучающих численное моделирование реальных процессов и явлений.

УДК 519.6:004(075.8)
ББК 22.19я73+32.97я73

ISBN 978-5-7410-2572-7

© Болодурина И.П.,
Анциферова Л.М.,
Парфенов Д.И.
Гришина Л.С., 2021
© ОГУ, 2021

Содержание

Введение.....	6
1 Математические модели и численные методы	7
1.1 Основные понятия теории моделирования	7
1.2 Этапы численного моделирования.....	8
1.3 Элементарная теория погрешностей.....	9
2 Аналитическое приближение табличных функций.....	13
2.1 Понятие о приближении функций.....	13
2.2 Многочлен Лагранжа.....	18
2.3 Интерполирование сплайнами.....	19
3 Численное дифференцирование	23
3.1 Аппроксимация производных.....	23
3.2 Погрешность численного дифференцирования.....	25
3.3 Применение интерполяционных формул	27
3.4 Метод неопределенных коэффициентов	31
4 Численное интегрирование	33
4.1 Теоретические основы численного интегрирования.....	33
4.2 Методы прямоугольников.....	36
4.3 Метод трапеций.....	39
4.4 Метод Симпсона	42
5 Численные методы решения уравнений и систем.....	46
5.1 Общие теоретические положения о численном решении уравнений	46
5.1.1 Метод хорд.....	48
5.1.2 Метод касательных	51
5.1.3 Метод секущих.....	53
5.1.3 Метод простых итераций	54
5.2 Численные методы решения систем линейных уравнений	55
5.2.1 Метод исключения Гаусса	55
5.2.2 Метод LU разложения матриц.....	59
5.2.3 Метод простой итерации.....	64

5.2.4	Метод Зейделя	68
5.3	Численные методы решения систем нелинейных уравнений	69
6	Численные методы решения дифференциальных уравнений и систем	72
6.1	Численные методы решения обыкновенных дифференциальных уравнений	72
6.1.1	Одношаговые методы решения ОДУ	74
6.1.1.1	Метод Эйлера	74
6.1.1.2	Модифицированный метод Эйлера	77
6.1.1.3	Методы Рунге-Кутта	79
6.1.2	Многошаговые методы решения ОДУ	80
6.1.2.1	Метод Адамса	80
6.1.2.2	Метод Гира	82
6.1.3	Нейросетевой подход к решению ОДУ	84
6.2	Численные методы решения систем дифференциальных уравнений	86
6.2.1	Метод Эйлера	87
6.2.2	Модифицированный метод Эйлера	88
6.2.3	Метод Рунге-Кутта 4-го порядка	88
6.2.4	Нейросетевой подход к решению систем ДУ	89
7	Численные методы решения краевых задач для дифференциальных уравнений в частных производных	91
7.1	Методы решения задач математической физики, описываемых ДУ в частных производных	92
7.2	Метод конечных разностей для ДУ в частных производных	94
7.3	Начальные и граничные условия	95
7.4	Уравнение колебаний	96
7.5	Уравнение теплопроводности	98
7.6	Уравнение Пуассона	101
8	Применение методов вычислительной математики в различных научных областях	104
8.1	Моделирование эпидемий	104
8.1.1	Распространение гриппа	104
8.1.1.1	Метод Эйлера для системы дифференциальных уравнений	108

8.1.1.2 Программирование численной схемы: частный случай	109
8.1.2 Численное моделирование вспышки	111
8.1.3 Общая постановка задачи решения систем ДУ для SIR	112
8.1.3.1 Программирование численной схемы: общий случай	112
8.1.4 Моделирование иммунитета с ограничением по времени	114
8.1.5 Моделирование вакцинации населения	116
8.2 Моделирование распространения вирусных атак в сетях	118
8.2.1 Актуальность проведения исследований в области обеспечения безопасности IoT	119
8.2.2 Математическая модель распределенной атаки типа «отказ в обслуживании» в сети IoT	122
8.2.3 Численное моделирование распределенной атаки в сети IoT	126
Список использованных источников	130

Введение

Учебное пособие «Вычислительная математика и информационные технологии» предназначено для обучающихся по образовательным программам высшего образования по направлению подготовки 01.04.02 Прикладная математика и информатика, 02.04.02 Фундаментальная информатика и информационные технологии и 09.04.02 Информационные системы и технологии, а также для других направлений и специальностей, изучающих численное моделирование реальных процессов и явлений.

Изучение представленного теоретического материала способствует формированию компетенций, позволяющих проводить поэтапно математическое моделирование объектов, описываемых дифференциальными, разностными и алгебраическими уравнениями, а также применять методы и алгоритмы исследования этих моделей с учетом их возможной реализации на ЭВМ. Учебное пособие затрагивает вопросы применения методов и приёмов решения задач аппроксимации функций, численного интегрирования и дифференцирования, линейной алгебры, решения нелинейных уравнений и систем, дифференциальных уравнений, а также уравнений в частных производных.

Представленные методы и алгоритмы численного решения задач могут быть использованы при принятии решений в социальных, экономических и технических системах. Учебное пособие включает примеры применения численных методов для моделирования эпидемий с различными ограничениями, а также моделирование распространения атак в сетях. Разработанные программные средства позволяют обеспечивать надежность и своевременную диагностику функционирования вычислительных машин, комплексов и компьютерных сетей.

При разработке материала пособия использовался опубликованный теоретический материал и опыт преподавания учебных дисциплин, основанных на численном решении задач в различных областях науки и практической деятельности, а также личный методический опыт авторов.

1 Математические модели и численные методы

На настоящий момент применение методов вычислительной математики прочно вошло в различные прикладные области. Так как в процессе разработки новых технологических процессов и средств их оптимизации, инструментов решения различных экономических, физических, технических и др. типов задач, а также при принятии управленческих решений используют математическое моделирование, то проведение численных экспериментов является актуальной задачей. При этом прикладная задача должна быть представлена формальной математической постановкой, т.е. для реального объекта, процесса или системы должна быть построена его математическая модель. В рамках данной главы описываются основные понятия компьютерного моделирования, общие этапы проведения вычислительного эксперимента, а также отличие полученного приближенного решения от точного.

1.1 Основные понятия теории моделирования

Построение моделей реальных объектов является инструментом выявления свойств и закономерностей поведения данного объекта. В связи с этим *целью моделирования* являются получение, обработка, представление и использование информации об объектах, которые взаимодействуют между собой и внешней средой.

В основе теории моделирования лежит теория подобия. Ясно, что достигнуть абсолютного подобия при проведении моделирования невозможно. Однако, чтобы модель лучше отображала исследуемый процесс, следует включать как можно большее количество информативных свойств объектов.

Математическое моделирование - это инструмент исследования реального объекта, процесса или системы путем построения математической модели, более удобной для экспериментального исследования с помощью ЭВМ.

Математическая модель является приближенным представлением реальных объектов, процессов или систем, выраженным в математических терминах и сохраняющим существенные черты оригинала. Математические модели в количественной форме, с помощью логико-математических конструкций, описывают основные свойства объекта, процесса или системы, его параметры, внутренние и внешние связи.

В общем случае *математическая модель* может быть представлена в виде системы:

$$\Phi(X, Y, Z, t) = 0, \quad (1.1)$$

где $X = \{x_1, \dots, x_n\}$ - вектор входных переменных, $Y = \{y_1, \dots, y_n\}$ - вектор выходных переменных, $Z = \{z_1, \dots, z_n\}$ - вектор внешних воздействий, t - координата времени.

Таким образом, построение математической модели заключается в формировании математического аппарата, который позволяет описать качественно и количественно взаимосвязи исследуемых процессов и явлений. Как правило, несущественные факторы на данном этапе моделирования должны быть исключены. Целью данного этапа является постановка математической задачи, решение которой с необходимой точностью выражает результаты, интересующие специалиста.

1.2 Этапы численного моделирования

Решение различных прикладных задач с применением методов вычислительной математики можно описать следующими **этапами**.

Этап 1. Описание постановки задачи в исследуемой области. На данном этапе необходимо сформулировать цель исследования, зафиксировать объекты, процессы или системы, участвующие в моделировании.

Этап 2. Формирование математической модели. На данном этапе происходит выбор подходящей или построение новой математической модели, адекватно отражающей исследуемые процессы и явления и соответствующей цели проведения численного моделирования.

Этап 3. Выбор численного метода. Данный этап предназначен для выбора наиболее эффективных (возможно более одного) численного метода решения, позволяющего в дальнейшем построить вычислительный алгоритм.

Этап 4. Разработка алгоритма решения задачи. В соответствии с выбранным численным методом составляется алгоритм решения задачи в виде последовательности логических и арифметических операций.

Этап 5. Реализация алгоритма в виде программного обеспечения. Составление программы численного решения поставленной задачи может осуществляться на любом из известных языков программирования высокого уровня.

Этап 6. Отладка программного обеспечения. Реализованный алгоритм должен пройти тестирование и разработчику необходимо внести исправление найденных ошибок.

Этап 7. Анализ численных результатов. Результаты численного эксперимента, полученные с помощью разработанного ПО анализируются, при возможности сравниваются с экспериментальными данными и оформляется соответствующая научно-техническая документация.

Этап 8. Внедрение полученных результатов.

В результате проведения численного моделирования специалист получает приближенное решение поставленной задачи, поэтому наибольшее внимание следует уделять погрешности получаемого решения.

1.3 Элементарная теория погрешностей

Основными источниками погрешности численных результатов являются исходные данные, математическая модель, приближенный метод и округления

при вычислениях. В рамках данного учебного пособия опущено рассмотрение погрешностей математических моделей, которые могут быть связаны с исключением влияния некоторых физических процессов.

Как правило, исходные данные могут содержать неточности, если измерения проводились путем эксперимента. Кроме того, исходя из необходимого порядка точности вычислений в каждой отдельной научной области, можно выявить границы точностей. Например, для физических процессов точность измерения может достигать до 10^{-12} , для астрономических вычислений до 10^{-6} , для технических задач допустимы измерения в пределах 1–10%.

При наличии *погрешности исходных данных* δx численный результат содержит *неустранимую* погрешность решения $\delta y = A(x + \delta x) - A(x)$. В этом случае, необходимо применять дополнительные методы оценки порогового значения данной погрешности во избежание сильного расхождения численного результата с точным решением.

Погрешность метода аппроксимации или **методическая погрешность** появляется в результате замены точных операторов и исходных значений на приближенные. Примером появления данного типа погрешности является замена производной функции на некоторую разность, интеграла функции на сумму конечного числа точек значений функции и т.п. Данные подстановки в таких задачах обусловлены тем, что при стремлении шага разбиения к нулю, погрешность решения также стремится к нулю. В связи с данным свойством при применении каких-либо численных подходов производят теоретическую и практическую оценку точности.

Главной проблемой в установлении порога точности для численного решения является отношение погрешности метода и неустранимой погрешности. Как правило, методическая погрешность должна быть в 2-4 раза меньше. Связано это с тем, что большое значение погрешности метода понижает точность решения, а маленькое значение погрешности метода значительно повышает требуемые ресурсы для вычислений.

Погрешность округления связана с форматом представления чисел в ЭВМ и хранимым количеством разрядов числа при проведении операций. Данная погрешность накапливается в процессе вычислений.

Пусть значение a является точным решением задачи, а решение a^* – приближенным решением. Тогда справедливы следующие определения.

Абсолютной погрешностью приближенного значения a^* называется модуль разности $|a - a^*|$.

Верхней границей погрешности $\Delta(a^*) = \Delta_{a^*}$ (оценкой) приближенного значения a^* называется значение, которое удовлетворяет неравенству $|a - a^*| < \Delta(a^*)$.

Относительной погрешностью приближенного решения a^* называется отношения модуля разности точного решения и приближенного к значению приближенного решения $\frac{|a - a^*|}{a^*}$.

Верхней границей погрешности $\delta(a^*) = \delta_{a^*}$ (оценкой) приближенного значения a^* называется значение, которое, при $a^* \neq 0$, удовлетворяет неравенству $\frac{|a - a^*|}{a^*} < \delta(a^*)$.

При решении любых прикладных задач необходимо соблюдать правило: необходимо задавать верхние границы абсолютной или относительной погрешности так, чтобы погрешность округления была существенно меньше других погрешностей и соответствовала потребностям научной области.

Задача $y = A(x)$ называется *корректно поставленной*, если решение y существует, единственно и устойчиво для любых исходных данных x .

Рассмотрим данное определение относительно влияния погрешности: если исходные данные содержат погрешность, то их можно представить в виде $x + \delta x$ и решение соответственно $y + \delta y = A(x + \delta x)$. Таким образом, неустранимая погрешность имеет вид $\delta y = A(x + \delta x) - A(x)$.

Задача $y = A(x)$ называется *устойчивой* по входным данным, если ее решение непрерывно зависит от них, т.е. при $\|\delta x\| \rightarrow 0$ выполняется $\|\delta y\| \rightarrow 0$.

Не устойчивые задачи ставят в соответствие небольшой погрешности входных данных δx сравнительно большую погрешность решения δy . Однако, даже в случае устойчивости задачи приближенный метод решения может оказаться неустойчивым, что впоследствии приведет к увеличению погрешности.

2 Аналитическое приближение табличных функций

Вычислительные эксперименты, охватывающие огромное множество объектов моделирования, как правило, проводятся в достаточно сложных условиях с ограниченным временем, а полученные результаты необходимо с хорошей точностью обобщать или конкретизировать. В рамках данной главы представлены методы аппроксимации табличных функций и данных экспериментов.

2.1 Понятие о приближении функций

Постановка задачи. Пусть величина y является функцией аргумента x . Это означает, что любому x_i из области определения поставлено в соответствие значение $y_i, i=1, \dots, n$. Вместе с тем, на практике часто неизвестна явная связь между y и x , т.е. невозможно записать эту связь в виде некоторой зависимости $y = f(x)$. В некоторых случаях даже при известной зависимости $y = f(x)$ она настолько громоздка (например, содержит трудно вычисляемые выражения, сложные интегралы и т.п.), что ее использование в практических расчетах затруднительно.

Наиболее распространенным и практически важным случаем, когда вид связи между y и x неизвестен, является задание этой связи в виде некоторой таблицы $\{x_i, y_i\}, i=1, \dots, n$. Это означает, что дискретному множеству значений аргумента $\{x_i\}$ поставлено в соответствие множество значений функции $\{y_i\}, i=1, \dots, n$. Данные значения являются либо результатами расчетов, либо экспериментальными данными. На практике могут понадобиться значения величины y и в других точках, отличных от узлов $\{x_i\}$. Однако получить эти значения можно лишь путем сложных расчетов или проведением дорогостоящих экспериментов.

Таким образом, с точки зрения экономии времени и средств приходим к необходимости использования имеющихся табличных данных для приближенного вычисления искомого параметра y при любом значении (из некоторой области) определяющего параметра x , поскольку точная связь $y = f(x)$ неизвестна.

Задача о приближении (аппроксимации) функций: искомую функцию $f(x)$ требуется приближенно заменить (аппроксимировать) некоторой функцией $\varphi(x)$ так, чтобы отклонение (в некотором смысле) $\varphi(x)$ от $f(x)$ в заданной области были наименьшим. Функция $\varphi(x)$ при этом называется аппроксимирующей.

Наиболее распространенным подходом к аппроксимации функций основан на представлении функций многочленом:

$$\varphi(x) = a_0 + a_1 \cdot x + a_2 \cdot x^2 + \dots + a_m \cdot x^m. \quad (2.1)$$

При этом коэффициенты a_i подбираются так, чтобы достичь наименьшего отклонения многочлена от данной функции.

Если приближение строится на заданном дискретном множестве точек $\{x_i\}$, то аппроксимация называется **точечной**. К ней относятся интерполирование, среднее квадратическое приближение и др.

При построении приближения на непрерывном множестве точек (например, на отрезке $[a, b]$) аппроксимация называется **непрерывной** (или **интегральной**).

Точечная аппроксимация. Одним из основных типов точечной аппроксимации является интерполирование: для заданной функции $y = f(x)$ строится многочлен (2.1), принимающий в заданных точках x_i те же значения y_i , что и функция $f(x)$, т.е.

$$\varphi(x_i) = y_i, \quad i = 0, 1, \dots, n. \quad (2.2)$$

При этом предполагается, что среди значений x_i нет одинаковых, т.е. $x_i \neq x_j$ при $i \neq j$. Точки x_i называются **узлами интерполяции**, а многочлен $\varphi(x)$ – **интерполяционным многочленом**.

Таким образом, близость интерполяционного многочлена к заданной функции состоит в том, что их значения совпадают на заданной системе точек (рисунок 2.1).

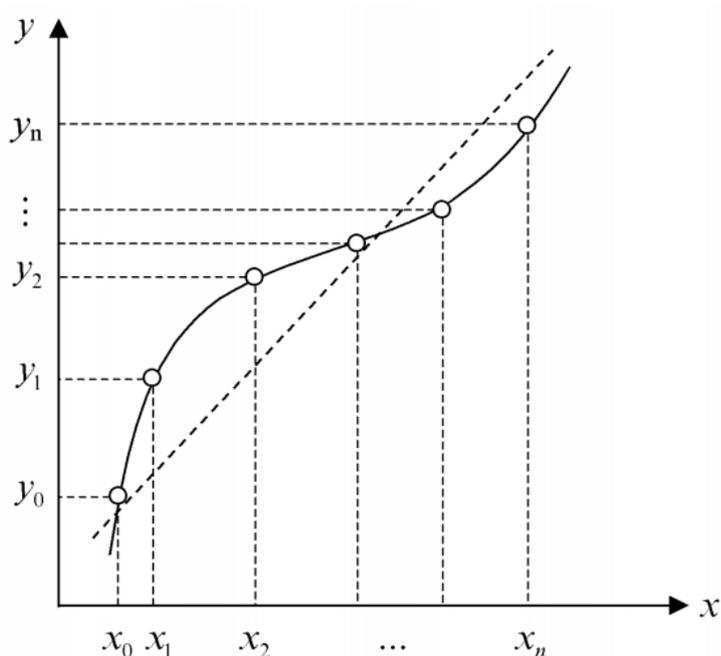


Рисунок 2.1 – Пример аппроксимации функции на отрезке

Максимальная степень интерполяционного многочлена $m = n$; в этом случае говорят о **глобальной интерполяции**, поскольку один многочлен используется для интерполяции функции $f(x)$ на всем рассматриваемом интервале изменения аргумента x :

$$\varphi(x) = a_0 + a_1 \cdot x + a_2 \cdot x^2 + \dots + a_n \cdot x^n. \quad (2.3)$$

Коэффициенты a_i многочлена (2.3) находятся из системы уравнений (2.2). Можно показать, что при $x_i \neq x_j$ ($i \neq j$) данная система имеет единственное решение.

Интерполяционные многочлены могут строиться отдельно для разных частей рассматриваемого интервала изменения x . В этом случае имеем **кусочную** (или **локальную**) **интерполяцию**.

Как правило, интерполяционные многочлены используются для аппроксимации функции в промежуточных точках между крайними узлами интерполяции, т.е. при $x_0 < x < x_n$. Однако иногда они используются для **экстраполяции** – приближенного вычисления функции вне рассматриваемого отрезка ($x < x_0, x > x_n$).

Заметим, что при интерполировании основным условием является прохождение графика интерполяционного многочлена через данные значения функции в узлах интерполяции. Однако в ряде случаев выполнение этого условия затруднительно или даже нецелесообразно.

Например, при большом количестве узлов интерполяции получается высокая степень многочлена (2.3) в случае глобальной интерполяции, т.е. когда необходимо получить один интерполяционный многочлен для всего интервала изменения аргумента. Кроме того, табличные данные могли быть получены путем измерения и содержать ошибки, для устранения которых необходимо сгладить полученные значения (рисунок 2.1).

Одним из видов приближения является **среднеквадратическое приближение** функции с помощью многочлена (2.1) при $m \leq n$.

Мерой отклонения многочлена $\varphi(x)$ от заданной функции $f(x)$ на множестве точек $\{x_i, y_i\}$, $i = 0, 1, \dots, n$ при среднеквадратическом приближении является величина S , равная сумме квадратов разностей между значениями многочлена и функции в данных точках:

$$S = \sum_{i=0}^n [\varphi(x_i) - y_i]^2. \quad (2.4)$$

Идея *метода наименьших квадратов* заключается в построении аппроксимирующего многочлена и подборе коэффициентов a_0, a_1, \dots, a_m так, чтобы величина S была наименьшей.

Равномерное приближение. Во многих случаях, например при обработке экспериментальных данных, среднеквадратическое приближение вполне приемлемо, поскольку оно сглаживает некоторые неточности функции $f(x)$ и дает достаточное количество информации. Иногда при построении приближения ставится более жесткое условие: во всех точках некоторого отрезка $[a, b]$ отклонение многочлена $\varphi(x)$ от функции $f(x)$ по абсолютной величине должно быть меньше заданной величины $\varepsilon > 0$:

$$|f(x) - \varphi(x)| < \varepsilon, \quad a \leq x \leq b.$$

В этом случае многочлен $\varphi(x)$ называют *равномерно аппроксимирующим* функцию $f(x)$ с погрешностью ε на отрезке $[a, b]$.

Абсолютным отклонением Δ многочлена $\varphi(x)$ от функции $f(x)$ на отрезке $[a, b]$ называют максимальное значение абсолютной величины разности между ними на данном отрезке:

$$\Delta = \max_{a \leq x \leq b} |f(x) - \varphi(x)| \quad (2.5)$$

По аналогии можно ввести понятие *среднеквадратического отклонения* $\Delta = \sqrt{S/n}$ при среднеквадратичном приближении функций.

Существует также понятие *наилучшего приближения функции* $f(x)$ многочленом $\varphi(x)$ степени m . В этом случае коэффициенты многочлена $\varphi(x)$ следует подбирать так, чтобы на заданном отрезке $[a, b]$ величина абсолютного

отклонения (2.5) была минимальной. Многочлен $\varphi(x)$ называется *многочленом наилучшего равномерного приближения*.

2.2 Многочлен Лагранжа

Рассмотрим более подробно глобальную интерполяцию, при условии, что интерполяционный многочлен должен проходить через все заданные точки.

Запишем многочлен в виде:

$$\varphi(x) = a_0 + a_1 \cdot x + a_2 \cdot x^2 + \dots + a_n \cdot x^n. \quad (2.6)$$

Из условий равенства значений данного многочлена в узлах x_i соответствующим заданным табличным значениям y_i получим следующую систему уравнений для нахождения коэффициентов a_0, a_1, \dots, a_n :

$$\begin{aligned} a_0 + a_1 \cdot x_0 + a_2 \cdot x_0^2 + \dots + a_n \cdot x_0^n &= y_0, \\ a_0 + a_1 \cdot x_1 + a_2 \cdot x_1^2 + \dots + a_n \cdot x_1^n &= y_1, \\ &\dots \\ a_0 + a_1 \cdot x_n + a_2 \cdot x_n^2 + \dots + a_n \cdot x_n^n &= y_n. \end{aligned} \quad (2.7)$$

Найдем коэффициенты интерполяционного многочлена (2.6), решив уравнения (2.7). Данный подход к построению интерполяционного многочлена требует значительного объема вычислений при большом количестве узлов, рассмотрим метод Лагранжа построения интерполяционных многочленов.

Будем искать многочлен L в виде линейной комбинации многочленов $l_i(x)$ степени n :

$$L_n(x) = y_0 \cdot l_0(x) + y_1 \cdot l_1(x) + y_2 \cdot l_2(x) + \dots + y_n \cdot l_n(x) = \sum_{i=0}^n y_i \cdot l_i(x). \quad (2.8)$$

Потребуем, чтобы каждый многочлен $l_i(x)$ обращался в нуль во всех узлах интерполяции, за исключением одного (i -го), где он должен равняться единице, т.е.

$$l_i(x_j) = \begin{cases} 1, & \text{при } i = j, \\ 0, & \text{при } i \neq j. \end{cases}$$

Данным условиям удовлетворяют соответственно многочлены вида

$$\begin{aligned} l_0(x) &= \frac{(x-x_1)(x-x_2)\dots(x-x_n)}{(x_0-x_1)(x_0-x_2)\dots(x_0-x_n)}, \\ l_1(x) &= \frac{(x-x_0)(x-x_2)\dots(x-x_n)}{(x_1-x_0)(x_1-x_2)\dots(x_1-x_n)}, \\ &\dots \\ l_i(x) &= \frac{(x-x_0)\dots(x-x_{i-1})(x-x_{i+1})\dots(x-x_n)}{(x_i-x_0)\dots(x_i-x_{i-1})(x_i-x_{i+1})\dots(x_i-x_n)}, \\ &\dots \end{aligned} \tag{2.9}$$

Подставляя в (2.8) выражения (2.9) получим:

$$L_n(x) = \sum_{i=0}^n y_i \cdot \frac{(x-x_0)\dots(x-x_{i-1})(x-x_{i+1})\dots(x-x_n)}{(x_i-x_0)\dots(x_i-x_{i-1})(x_i-x_{i+1})\dots(x_i-x_n)} = \sum_{i=0}^n y_i \prod_{\substack{j=0 \\ j \neq i}}^n \frac{x-x_j}{x_i-x_j}. \tag{2.10}$$

Формула (2.10) называется **интерполяционным многочленом Лагранжа**. Можно показать, что данный многочлен является единственным.

2.3 Интерполирование сплайнами

Интерполирование многочленом Лагранжа на отрезке $[a, b]$ с использованием большого количества узлов интерполяции часто приводит к плохому приближению, что объясняется накоплением погрешностей в процессе вычислений – явление «волнистости». Для снижения больших погрешностей, отрезок разбивают на подотрезки и на каждом из них приближенно заменяют

функцию $f(x)$ многочленом невысокой степени (так называемая кусочно-полиномиальная интерполяция).

Одним из способов интерполяции на всем отрезке является интерполирование с помощью сплайн-функций. **Сплайн-функцией** или **сплайном** называют кусочно-полиномиальную функцию, определенную на отрезке $[a, b]$ и имеющую на этом отрезке некоторое число непрерывных производных.

Для построения кубического интерполяционного сплайна можно использовать различные подходы. Рассмотрим один из наиболее распространенных вариантов интерполяции кубическими сплайнами. Используя законы упругости, можно установить, что недеформируемая линейка между соседними узлами проходит по линии, удовлетворяющей уравнению

$$\varphi^{IV}(x) = 0. \quad (2.11)$$

Если в качестве функции $\varphi(x)$ выбрать полином, то в соответствии с уравнение (2.11) степень полинома должна быть не выше третьей и данный полином называют **кубическим сплайном**, который на интервале $x \in [x_{i-1}, x_i] \subseteq [a, b]$ записывают в виде

$$\varphi_i(x) = s_i(x) = a_i + b_i(x - x_{i-1}) + c_i(x - x_{i-1})^2 + d_i(x - x_{i-1})^3, \quad (2.12)$$

где a_i, b_i, c_i, d_i - коэффициенты сплайна, определяемые из дополнительных условий; i - номер сплайна.

Преимущество сплайнов перед обычной интерполяцией является сходимость и устойчивость процесса вычислений.

Интерполяционным кубическим сплайном, соответствующим данной функции $f(x)$ и данным узлам, называется функция $s(x)$, удовлетворяющая следующим условиям:

а) на каждом подотрезке $[x_{i-1}, x_i]$, $i=1, \dots, n$ функция $s(x)$ является многочленом третьей степени (2.12);

б) значения сплайнов $s(x_i)$ и аппроксимируемой функции $y_i = f(x_i)$, в узлах равны (условия Лагранжа):

$$s(x_{i-1}) = y_{i-1}, \quad s(x_i) = y_i. \quad (2.13)$$

в) функция $s(x)$, а также ее первая и вторая производные непрерывны на отрезке $[a, b]$

$$s'(x_i) = s'(x_{i+1}), \quad (2.14)$$

$$s''(x_i) = s''(x_{i+1}). \quad (2.15)$$

Кроме перечисленных условий необходимо задать условия на концах, т.е. в точках x_0 и x_n . В общем случае эти условия зависят от конкретной задачи. Довольно часто используется условия свободных концов сплайнов. Если линейка не закреплена в точках вне интервала $[x_0 = a, x_n = b]$, то соответствует полиному первой степени. Следовательно, исходя из условий непрерывности вторых производных сплайнов на концах интервала, получим:

$$s_1''(x_0) = 0, \quad (2.16)$$

$$s_n''(x_n) = 0. \quad (2.17)$$

Из (2.16)-(2.17) получим метод определения кубических сплайнов. Условия (2.13) в узлах x_{i-1} и x_i , после подстановки i -го сплайна имеют вид:

$$a_i = y_{i-1}, \quad a_i + b_i h_i + c_i h_i^2 + d_i h_i^3 = y_i, \quad (2.18)$$

где $h_i = x_i - x_{i-1}$, $i = 1, \dots, n$.

Продифференцируем дважды функцию (2.12) и подставляя в (2.14) и (2.15) получим:

$$b_i + 2c_i h_i + 3d_i h_i^2 = b_{i+1}, \quad (2.19)$$

$$c_i + 4d_i h_i = c_{i+1}. \quad (2.20)$$

Из граничных условий (2.16)-(2.17) на основании выражения второй производной получим:

$$c_1 = 0, \quad (2.21)$$

$$c_n + 3d_n h_n = 0. \quad (2.22)$$

Соотношения (2.18)-(2.22) являются полной системой линейных алгебраических уравнений относительно коэффициентов сплайнов a_i, b_i, c_i, d_i . Решение данной системы можно осуществить с помощью метода прогонки, который состоит в последовательном исключении неизвестных.

3 Численное дифференцирование

Множество физических процессов при математическом моделировании можно описать с помощью производных функций. Однако их исследование может быть затруднено из-за проблем с поиском и представлением этих моделей в аналитическом виде. Иногда данные об объекте могут быть получены исключительно экспериментально и представлены в виде таблиц. Многообразие численных методов дифференцирования позволяет решать данные проблемы. В рамках главы 3 рассмотрены наиболее распространенные подходы к приближенному вычислению производных таблично заданных функций.

3.1 Аппроксимация производных

Определение. Производная функции $y = f(x)$ это предел отношения приращения функции Δy к приращению аргумента Δx при $\Delta x \rightarrow 0$:

$$y' = f'(x) = \lim_{\Delta x \rightarrow 0} \frac{\Delta y}{\Delta x}, \quad \Delta y = \Delta f(x + \Delta x) - f(x). \quad (3.1)$$

Как правило, для вычисления значений производной функции применяют таблицу производных и правил, не анализируя соотношение (1). При этом если исходная функция $y = f(x)$ задана в дискретном виде (таблицей значений на некотором промежутке), то при конечном значении шага Δx используют **аппроксимацию (приближение) производной с помощью отношения конечных разностей**:

$$y' \approx \frac{\Delta y}{\Delta x}. \quad (3.2)$$

Рассмотрим задачу вычисления производной функции $y = f(x)$, которая задана таблицей значений $\{x_i, y_i\}, i = 1, \dots, n$. Отметим, что **шаг** по сетке $h = x_i - x_{i-1} = \text{const}, i = 1, \dots, n$.

Дифференцирование методом *левых разностей*:

$$\Delta y_1 = y_1 - y_0, \quad \Delta x = h, \quad y_1' \approx \frac{y_1 - y_0}{h}. \quad (3.3)$$

Дифференцирование методом *правых разностей*:

$$\Delta y_1 = y_2 - y_1, \quad \Delta x = h, \quad y_1' \approx \frac{y_2 - y_1}{h}. \quad (3.4)$$

Дифференцирование методом *центральных разностей*:

$$\Delta y_1 = y_2 - y_0, \quad \Delta x = 2h, \quad y_1' \approx \frac{y_2 - y_0}{2h}. \quad (3.5)$$

Таким образом, используя соотношение (3.2) возможно найти приближение производной функции любого порядка. Например, вторая производная имеет вид:

$$y_1'' = (y_1')' \approx \frac{y_2' - y_1'}{h} = \frac{\frac{y_2 - y_1}{h} - \frac{y_1 - y_0}{h}}{h} = \frac{y_2 - 2y_1 + y_0}{h^2}. \quad (3.6)$$

Наиболее важным вопросом при проведении численного дифференцирования является точность результатов. Далее покажем, что для наиболее эффективной аппроксимации производных функций необходимо использовать больше значений узлов, однако соотношение (3.2) это не предусматривает.

3.2 Погрешность численного дифференцирования

Рассмотрим аппроксимацию функции $f(x)$ некоторой функцией $\varphi(x)$, т.е. справедливо соотношение:

$$f(x) = \varphi(x) + R(x). \quad (3.7)$$

Отметим, что аппроксимирующая функция $\varphi(x)$ может являться интерполяционной функцией или частичной суммой ряда, а $R(x)$ соответственно остаточным членом интерполяционной формулы или ряда и называется **погрешностью аппроксимации**.

Таким образом, дифференцируя соотношение (3.7) заданное число раз, получим необходимое значение производной:

$$f'(x) = \varphi'(x) + R'(x), \quad f''(x) = \varphi''(x) + R''(x), \quad \dots \quad (3.8)$$

Тогда аппроксимационным значением k -ой производной функции $f(x)$ является значение производной функции $\varphi(x)$, т.е. $f^{(k)}(x) \approx \varphi^{(k)}(x)$.

Погрешностью аппроксимации производной называется значение отклонения приближенного значения производной от реального значения:

$$R^{(k)}(x) = f^{(k)}(x) - \varphi^{(k)}(x), \quad (3.9)$$

При этом, погрешность численного дифференцирования зависит от шага h и имеет вид $O(h^k)$, где k - порядок погрешности. Продемонстрируем прямую оценку погрешности на примере ряда Тейлора:

$$f(x + \Delta x) = f(x) + f'(x)\Delta x + \frac{f''(x)\Delta x^2}{2!} + \frac{f^{(3)}(x)\Delta x^3}{3!} + \frac{f^{(4)}(x)\Delta x^4}{4!} + \dots$$

Пусть для функции $f(x)$ определены значения $f(x_i) = y_i$, $i = 0, 1, \dots, n$. Ряд Тейлора при $x = x_1$, $\Delta x = -h$ с точностью до $O(h^2)$ имеет вид:

$$f(x_1 - h) = f(x_0) = f(x_1) - f'(x_1)h + \frac{f''(x_1)h^2}{2!}.$$

Данное соотношение эквивалентно записи:

$$y_0 = y_1 - y_1' h + O(h^2),$$

или

$$y_1' = \frac{y_1 - y_0}{h} + O(h).$$

Последнее выражение является аппроксимацией производной первого порядка. Применим ряд Тейлора для оценки погрешности аппроксимаций (3.5) и (3.6). При $\Delta x = -h$ и $\Delta x = h$ получим соответственно:

$$y_0 = y_1 - y_1' h + \frac{y_1'' h^2}{2!} - \frac{y_1^{(3)} h^3}{3!} + \frac{y_1^{(4)} h^4}{4!} + \dots \quad (3.10)$$

$$y_2 = y_1 + y_1' h + \frac{y_1'' h^2}{2!} + \frac{y_1^{(3)} h^3}{3!} + \frac{y_1^{(4)} h^4}{4!} + \dots$$

Из разности уравнений (3.10) получим соотношение эквивалентное центральным разностям:

$$y_1' = \frac{y_2 - y_0}{2h} + O(h^2). \quad (3.11)$$

Из суммы уравнений (3.10) получим соотношение эквивалентное производной второго порядка:

$$y_1'' = \frac{y_2 - 2y_1 + y_0}{2h} - O(h^2). \quad (3.12)$$

Отметим, что в целом анализ погрешностей остаточного члена довольно сложная задача и более подробные сведения можно найти в теории разностных схем.

Кроме рассмотренного вида погрешностей, также существуют проблемы, связанные с недостаточно точными вычислениями значений y_i , а также погрешностями округлений при расчетах в ЭВМ. При этом с уменьшением шага по сетке h погрешность округления возрастает, в отличие от погрешности аппроксимации. В связи с этим, итоговая погрешность решения численного дифференцирования может снижаться при уменьшении шага только до некоторого предельного значения, а последующее уменьшение шага не повысит точности значений.

Оптимальная тонность может быть достигнута за счет регуляризации процедуры численного дифференцирования. Простейшим способом регуляризации является такой выбор шага h , при котором справедливо неравенство $|f(x+h) - f(x)| > \varepsilon$, где $\varepsilon > 0$ - некоторое заданное достаточно малое значение. Данный подход при численном дифференцировании исключает вычитает близких по величине значений, что, как правило, приводит к возрастанию погрешности. Следующий способ регуляризации – использование аппроксимирующих функций (например, многочленов) для сглаживания табличных значений.

3.3 Применение интерполяционных формул

В случае если для аналитически заданной функции затруднительно найти производную из-за ее сложности или же выражение производной имеет неудобную форму для применения, используются численные

дифференцирование. Применение интерполяционных многочленов один из наиболее распространенных подходов к решению.

Предположим, что для функции $f(x)$ необходимо найти производную в некоторой точке отрезка $[a, b]$, а $\varphi(x)$ - соответствующий интерполяционный многочлен на $[a, b]$. Получим значение $f'(x)$ как значение производной соответствующего интерполяционного многочлена, т.е.

$$f'(x) \approx \varphi'(x). \quad (3.13)$$

Для производных функции $f(x)$ более высокого порядка введем аналогичные обозначения. Далее рассмотрим основные подходы к численному дифференцированию с помощью многочленов.

Аппроксимация производной каноническим многочленом. Пусть функция $f(x)$, заданная в табличном виде с постоянным шагом $h = x_i - x_{i-1}$, ($i = 1, 2, \dots, n$), может быть аппроксимирована каноническим многочленом:

$$P_n(x) = a_0 + a_1x + a_2x^2 + \dots + a_nx^n = \sum_{i=0}^n a_i x^i.$$

Коэффициенты многочлена a_0, a_1, \dots, a_n можно найти из СЛАУ, получаемой из условий Лагранжа $P_n(x_i) = y_i$.

Для получения вида производных функций первого и второго порядка, продифференцируем канонический многочлен по x :

$$\begin{aligned} P_n'(x) &= \sum_{i=0}^n i \cdot a_i \cdot x^{i-1}, \\ P_n''(x) &= \sum_{i=0}^n i \cdot (i-1) \cdot a_i \cdot x^{i-2}. \end{aligned} \quad (3.14)$$

Отметим, что вычисление значений полинома и его производных при заданных значениях переменной x более эффективно проводить по схеме Горнера.

Аппроксимация производной многочленом Лагранжа. Пусть функция $f(x)$, заданная в табличном виде с постоянным шагом $h = x_i - x_{i-1}$, ($i = 1, 2, \dots, n$), может быть аппроксимирована многочленом Лагранжа:

$$L_n(x) = \sum_{i=0}^n y_i \cdot l_i, \quad (3.15)$$

где

$$l_i = \prod_{\substack{j=0, \\ i \neq j}}^n \frac{(x - x_j)}{(x_i - x_j)}. \quad (3.16)$$

Вычислим производную функции, продифференцировав многочлен (3.15) по x :

$$L_n'(x) = \sum_{i=0}^n \frac{1}{\prod_{\substack{j=0, \\ i \neq j}}^n (x_i - x_j)} \sum_{\substack{j=0, \\ i \neq j}}^n \frac{\prod_{k=0, \\ k \neq i}^n (x - x_j)}{(x - x_k)} \cdot y_i. \quad (3.17)$$

или

$$L_n'(x) = \sum_{i=0}^n \frac{\prod_{\substack{j=0, \\ i \neq j}}^n (x - x_j)}{\prod_{\substack{j=0, \\ i \neq j}}^n (x_i - x_j)} \sum_{\substack{k=0, \\ k \neq i}}^n \frac{(x - x_k)}{(x - x_k)} \cdot y_i. \quad (3.18)$$

Формула (3.18) имеет сложный вид для программирования. В связи с этим, определение 1-ой и 2-ой производных эффективнее выполнять одновременно с вычислением многочлена Лагранжа:

$$l_i(x) = \frac{(x-x_0)\dots(x-x_{i-1})(x-x_{i+1})\dots(x-x_n)}{(x_i-x_0)\dots(x_i-x_{i-1})(x_i-x_{i+1})\dots(x_i-x_n)}$$

в виде:

$$l := l \cdot \frac{(x-x_j)}{(x_i-x_j)}. \quad (3.19)$$

Реализовав цикл вычислений для формулы (3.19) при $j \neq i$ и начиная с $l := 1$, получим значение многочлена Лагранжа. Соответствующие рекуррентные формулы для 1-ой и 2-ой производных получим, продифференцировав уравнение (3.19):

$$l' := \frac{1}{x_i - x_j} \cdot [l'(x - x_j) + l], \quad (3.20)$$

$$l'' := \frac{1}{x_i - x_j} \cdot [l''(x - x_j) + 2l']. \quad (3.21)$$

Аппроксимация производной многочленом Ньютона. Многочлен Ньютона имеет следующий вид:

$$P_n(x) = a_0 + \sum_{i=0}^n a_i \prod_{j=0}^{i-1} (x - x_j). \quad (3.22)$$

Производная первого порядка многочлена Ньютона:

$$P'_n(x) = \sum_{i=0}^n a_i \sum_{j=0}^{i-1} \prod_{\substack{k=0 \\ k \neq j}}^{i-1} (x - x_k). \quad (3.23)$$

Коэффициенты многочлена a_0, a_1, \dots, a_n можно найти из СЛАУ, получаемой из условий Лагранжа $P_n(x_i) = y_i$.

3.4 Метод неопределенных коэффициентов

Альтернативным способом вычисления производной функции является **метод неопределенных коэффициентов**, который заключается в определении производной k -го порядка в заданной точке $x = x_1$ в виде линейной комбинации заданных значений функции в узлах x_0, x_1, \dots, x_n :

$$y_i^{(k)} = a_0 y_0 + a_1 y_1 + \dots + a_n y_n. \quad (3.24)$$

Как правило, соотношение (3.24) выполняется для многочленов $y = 1, y = x - x_1, \dots, y = (x - x_i)^n$. Осуществляя последовательную подстановку данных выражений в (3.24), получаем СЛАУ порядка $(n + 1)$ для определения неизвестных коэффициентов a_0, a_1, \dots, a_n .

Пример. Необходимо определить функцию производной y_1' для 4-х равноудаленных узлов.

Соотношение (3.24) примет вид:

$$y_1' = a_0 y_0 + a_1 y_1 + a_2 y_2 + a_3 y_3. \quad (3.25)$$

Подставим следующие многочлены:

$$y = 1, y = x - x_0, y = (x - x_0)^2, y = (x - x_0)^3. \quad (3.26)$$

Производные данных многочленов будут иметь вид:

$$y' = 0, y' = 1, y' = 2(x - x_0), y' = 3(x - x_0)^2. \quad (3.27)$$

Соотношения (3.26) и (3.27) подставим соответственно в правую и левую части уравнения (3.25) при $x = x_1$:

$$\begin{aligned} 0 &= a_0 \cdot 1 + a_1 \cdot 1 + a_2 \cdot 1 + a_3 \cdot 1, \\ 1 &= a_0(x_0 - x_0) + a_1(x_1 - x_0) + a_2(x_2 - x_0) + a_3(x_3 - x_0), \\ 2(x_1 - x_0) &= a_0(x_0 - x_0)^2 + a_1(x_1 - x_0)^2 + a_2(x_2 - x_0)^2 + a_3(x_3 - x_0)^2, \\ 3(x_1 - x_0)^2 &= a_0(x_0 - x_0)^3 + a_1(x_1 - x_0)^3 + a_2(x_2 - x_0)^3 + a_3(x_3 - x_0)^3. \end{aligned}$$

Данная система эквивалентна следующей СЛАУ:

$$\begin{aligned} a_0 + a_1 + a_2 + a_3 &= 0, \\ a_1 h + a_2 2h + a_3 3h &= 1, \\ a_1 h + a_2 4h + a_3 9h &= 2, \\ a_1 h + a_2 8h + a_3 27h &= 3. \end{aligned}$$

Решение СЛАУ имеет вид:

$$a_0 = -\frac{1}{3h}, \quad a_1 = -\frac{1}{2h}, \quad a_2 = -\frac{1}{h}, \quad a_3 = -\frac{1}{6h}.$$

После подстановки данных значений коэффициентов в (3.25), получим значение производной:

$$y_1' = \frac{1}{6h}(-2y_0 - 3y_1 - 6y_2 - y_3). \quad (3.28)$$

4 Численное интегрирование

Широкое применение интегралов в различных научных областях показывают важность их исследования. Использование определенных интегралов во многом облегчает решение прикладных задач геометрии, физики и других наук. В рамках главы 4 рассмотрены наиболее распространенные подходы к приближенному вычислению интегралов функций: методы прямоугольников, метод трапеций и квадратурная формула Симпсона.

4.1 Теоретические основы численного интегрирования

Пусть функция $y = f(x)$ определена на отрезке $[a, b]$. Разобьем отрезок $[a, b]$ на n элементарных отрезков вида $[x_{i-1}, x_i], (i = 1, \dots, n)$, где $x_0 = a, x_n = b$. Зафиксируем произвольную точку $\xi_i (x_{i-1} \leq \xi_i \leq x_i)$ на каждом отрезке и вычислим произведение s_i значений функций в точке $f(\xi_i)$ на длину элементарного отрезка $\Delta x_i = x_i - x_{i-1}$:

$$s_i = f(\xi_i) \cdot \Delta x_i. \quad (4.1)$$

Общая сумма всех произведений S_n называется *интегральной суммой* и имеет вид:

$$S_n = \sum_{i=1}^n s_i = \sum_{i=1}^n f(\xi_i) \cdot \Delta x_i. \quad (4.2)$$

Определение. *Определенным интегралом* от функции $f(x)$ на отрезке $[a, b]$ называется предел интегральной суммы при неограниченном увеличении числа точек разбиения; при этом длина наибольшего из элементарных отрезков стремится к нулю:

$$\int_a^b f(x)dx = \lim_{\max \Delta x \rightarrow 0} \sum_{i=1}^n f(\xi_i) \cdot \Delta x_i. \quad (4.3)$$

Геометрический смысл данных определений для случая $f(x) > 0$ показан на рисунке 4.1. Каждая точка M_i соответствует значению $(\xi_i, f(\xi_i))$. Отметим, что соотношения (4.1) для $i=1, 2, \dots, n$ геометрически представляют площади элементарных прямоугольников, а значение интегральной суммы (4.2) описывает площадь многоступенчатой фигуры, образуемой данными прямоугольниками. При неограниченном возрастании количества точек разбиения и стремлении к нулю элементов Δx_i верхняя граница фигуры (ломаная) переходит в непрерывную кривую $y = f(x)$. Связи с этими свойствами, площадь фигуры, вычисляемую как определенный интеграл (4.3), называют **криволинейной трапецией**.

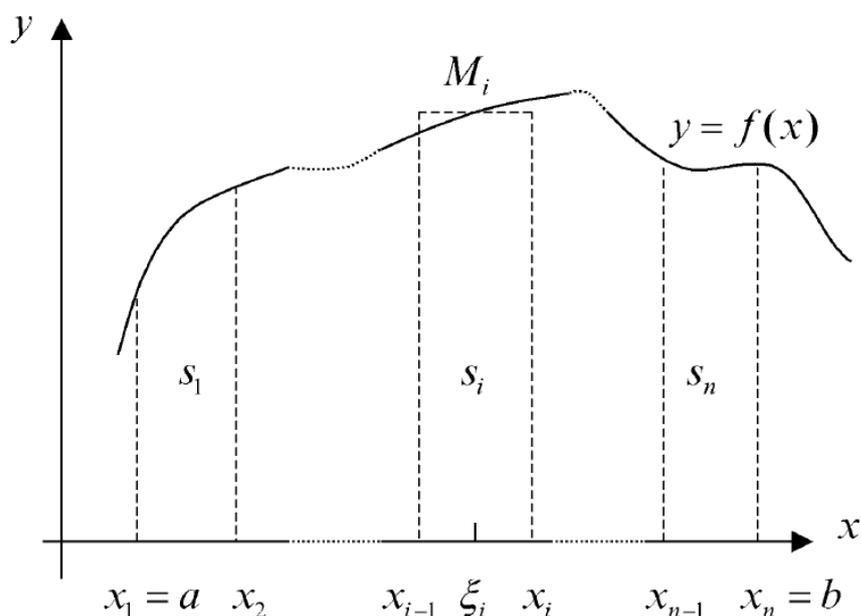


Рисунок 4.1 – Геометрический смысл определенного интеграла функции

Как правило, при задании подынтегральной функции в аналитическом виде, вычисление определенного интеграла возможно осуществить

непосредственно с помощью вычисления неопределенного интеграла по **формуле Ньютона-Лейбница**:

$$\int_a^b f(x)dx = F(x)\Big|_a^b = F(b) - F(a). \quad (4.4)$$

Формула (4.4) описывает значение определенного интеграла через приращение первообразной $F(x)$ на заданном отрезке интегрирования.

Возможные причины ограничений использования формулы Ньютона-Лейбница:

- невозможность выразить первообразную функцию через элементарные;
- функция задана не аналитически, а таблично.

На помощь аналитическим методам вычисления определенных интегралов в этих случаях приходят численные методы интегрирования. Основная идея этих методов состоит в аппроксимации подынтегральной функции более простыми выражениями (например, многочленами).

Один из наиболее распространенных подходов, решающих первую проблему при вычислении интегралов – **представление подынтегральной функции в виде степенного ряда** (ряда Тейлора), которое сводит определение интеграла сложной функции к интегрированию многочлена из нескольких первых членов ряда (количество членов ряда определяется исходя из необходимой точности вычислений).

Однако **методы численного интегрирования** являются более универсальными и позволяют решать обе проблемы при интегрировании. Основная идея численных методов основана на аппроксимации подынтегральной функции с помощью интерполяционных многочленов. В рамках данной главы рассмотрим кусочную (локальную) интерполяцию, что позволит определенный интеграл заменить приближенно интегральной суммой (4.2). Методы численного интегрирования отличаются способами

вычисления интегральной суммы: методы прямоугольников, трапеций, парабол, сплайнов и др.

Актуальность применения численных методов интегрирования подтверждается широким кругом решаемых прикладных задач: вычисление работы переменной силы, определение площади фигур и другие.

4.2 Методы прямоугольников

Основная идея *метода прямоугольников* заключается в прямой замене определенного интеграла на интегральную сумму (4.2), поэтому данный метод является наиболее простым. В качестве точек ξ_i могут выступать *левые* $\xi_i = x_{i-1}$ или *правые* $\xi_i = x_i$ границы элементарных отрезков, а соответствующие формулы метода прямоугольников имеют вид:

$$\int_a^b f(x)dx \approx h_1y_0 + h_2y_1 + h_3y_2 + \dots + h_ny_{n-1} = \sum_{i=1}^n h_iy_{i-1}, \quad (4.5)$$

$$\int_a^b f(x)dx \approx h_1y_1 + h_2y_2 + h_3y_3 + \dots + h_ny_n = \sum_{i=1}^n h_iy_i, \quad (4.6)$$

где $f(x_i) = y_i$, а $\Delta x_i = h_i$.

Наиболее распространенным и точным среди методов прямоугольников является вид формулы, использующий в качестве точек ξ_i серединные точки элементарных отрезков (в *полуцелых узлах*), называемый *методом средних прямоугольников*:

$$\int_a^b f(x)dx \approx \sum_{i=1}^n h_i \cdot f(x_{i-1/2}) = \sum_{i=1}^n h_i y_{i-1/2}, \quad (4.7)$$

$$x_{i-1/2} = \frac{x_{i-1} + x_i}{2} = x_{i-1} + \frac{h_i}{2}, \quad i = 1, 2, \dots, n.$$

Погрешность методов прямоугольников.

Рассмотрим интервал $[x_i, x_i + h]$ и определим на нем значение интеграла методом средних прямоугольников:

$$\int_{x_i}^{x_i+h} f(x)dx = h \cdot f(\bar{x}) + R, \quad (4.8)$$

где $\bar{x} = x_i + h/2, R = J_{\text{точн}} - J_{\text{прибл}}$.

Для оценки погрешности R разложим в ряд Тейлора подынтегральную функцию $f(x)$ вблизи средней точки \bar{x} :

$$f(x) = f(\bar{x}) + (x - \bar{x}) \cdot f'(\bar{x}) + \frac{(x - \bar{x})^2}{2!} \cdot f''(\bar{x}) + \frac{(x - \bar{x})^3}{3!} \cdot f'''(\bar{x}) + \dots \quad (4.9)$$

Соответствующий определенный интеграл (4.8) при подстановке данной функции $f(x)$ примет вид:

$$J_{\text{точн}} = h \cdot f(\bar{x}) + \frac{(x - \bar{x})^2}{2} \Big|_{x_i}^{x_i+h} \cdot f'(\bar{x}) + \frac{(x - \bar{x})^3}{2 \cdot 3} \Big|_{x_i}^{x_i+h} \cdot f''(\bar{x}) + \dots =$$

$$= h \cdot f(\bar{x}) + \frac{h^3}{24} \cdot f'''(\bar{x}) + \dots \quad (4.10)$$

Тогда, при достаточно малом значении шага интегрирования h , главный вклад в погрешность R вносит первое слагаемое – главный член погрешности R_{0i} вычисления определенного интеграла на отрезке $[x_i, x_i + h]$:

$$R_{0i} = \frac{h^3}{24} \cdot f'''(\bar{x}_i). \quad (4.11)$$

После суммирования погрешностей на каждом интервале $[x_i, x_i + h]$, получим главный член полной погрешности для интеграла на всем интервале $[x_0, x_n]$:

$$R_0 = \sum_{i=1}^n R_{0i} = \frac{h^2}{24} \sum_{i=1}^n h \cdot f'''(\bar{x}_i) = \frac{h^2}{24} \int_{x_0}^{x_n} f'''(x) dx. \quad (4.12)$$

Отметим, что соотношение (4.12) является теоретической оценкой погрешности интеграла метода средних прямоугольников. Данная оценка является априорной и не удобной для практического вычисления погрешности. Однако формула (4.12) эффективна при установлении структуры главного члена погрешности. При этом степень шага h , пропорциональная R_0 , называется порядком метода интегрирования. Следовательно, метод средних прямоугольников имеет второй порядок.

Осуществим априорную оценку метода левых прямоугольников. Для этого разложим в ряд Тейлора подынтегральную функцию $f(x)$ вблизи точки $x = x_i$:

$$f(x) = f(x_i) + (x - x_i) \cdot f'(x_i) + \frac{(x - x_i)^2}{2!} \cdot f''(x_i) + \frac{(x - x_i)^3}{3!} \cdot f'''(x_i) + \dots \quad (4.13)$$

Соответствующий определенный интеграл для (4.13) примет вид:

$$\begin{aligned} J_{\text{точн}} &= h \cdot f(x_i) + \frac{(x - x_i)^2}{2} \Big|_{x_i}^{x_i+h} \cdot f'(x_i) + \frac{(x - x_i)^3}{2 \cdot 3} \Big|_{x_i}^{x_i+h} \cdot f''(x_i) + \dots = \\ &= h \cdot f(x_i) + \frac{h^2}{2} \cdot f''(x_i) + \dots \end{aligned} \quad (4.14)$$

Тогда, при достаточно малом значении шага интегрирования h , главный вклад в погрешность R вносит первое слагаемое – главный член погрешности R_{0i} вычисления определенного интеграла на отрезке $[x_i, x_i + h]$:

$$R_{0i} = \frac{h^2}{2} \cdot f'(x). \quad (4.15)$$

После суммирования погрешностей на каждом интервале $[x_i, x_i + h]$, получим главный член полной погрешности для интеграла на всем интервале $[x_0, x_n]$:

$$R_0 = \sum_{i=1}^n R_{0i} = \frac{h}{2} \int_{x_0}^{x_n} f'(x) dx. \quad (4.16)$$

Исходя из оценки (4.16), метод левых прямоугольников имеет первый порядок и погрешность данного метода больше метода средних. Как правило, для большинства функций выполняется следующее соотношение:

$$\left| \int_{x_0}^{x_n} f'(x) dx \right| \geq \left| \int_{x_0}^{x_n} f''(x) dx \right|. \quad (4.17)$$

При этом, в случае если подынтегральная функция определена из эксперимента на дискретном наборе точек, то невозможно применить метод средних прямоугольников из-за отсутствия информации о средних точках и необходимо использовать другие подходы.

4.3 Метод трапеций

Основная идея *метода трапеций* заключается в линейной интерполяции, т.е. график функции $[x_i, x_i + h]$ представляется в виде ломаной, соединяющей точки (x_i, y_i) . Тогда площадь криволинейной трапеции состоит из площадей

элементарных прямолинейных трапеций (рисунок 4.2), которые равны половине произведения суммы оснований на высоту.

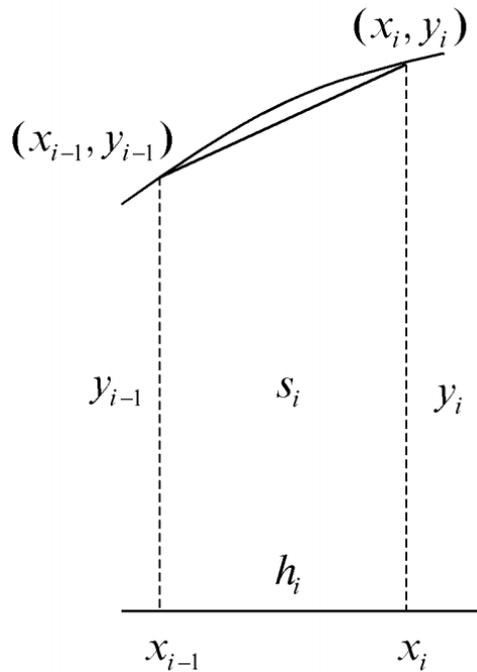


Рисунок 4.2 – Геометрический смысл метода трапеций

Таким образом, формула трапеций для численного интегрирования имеет вид:

$$\int_a^b f(x)dx = \frac{1}{2} \sum_{i=1}^n h_i \cdot (y_{i-1} + y_i). \quad (4.18)$$

Рассмотрим применение данного подхода при численном интегрировании с постоянным шагом, т.е. $h_i = h = const$. Тогда формулы прямоугольников и трапеций примут вид:

$$\int_a^b f(x)dx = h \sum_{i=1}^n f(x_{i-1/2}), \quad (4.19)$$

$$\int_a^b f(x)dx = h \left(\frac{y_0 + y_n}{2} + \sum_{i=1}^{n-1} y_i \right).$$

Погрешность метода трапеций.

Рассмотрим подынтегральную функцию $f(x)$ на интервале $[x_i, x_i + h]$ и заменим ее полиномом первой степени $P_1(x)$. Тогда приближенное значение интеграла определяется площадью трапеции:

$$\int_{x_i}^{x_i+h} f(x)dx = h \cdot [f(x_i) + f(x_i + h)] / 2 + R. \quad (4.20)$$

Для оценки априорной погрешности R разложим в ряд Тейлора подынтегральную функцию $f(x)$ вблизи точки x_i :

$$f(x) = f(x_i) + (x - x_i) \cdot f'(x_i) + \frac{(x - x_i)^2}{2!} \cdot f''(x_i) + \frac{(x - x_i)^3}{3!} \cdot f'''(x_i) + \dots \quad (4.21)$$

Тогда определенный интеграл будет иметь вид:

$$\int_{x_i}^{x_i+h} f(x)dx = h \cdot f(x_i) + \frac{h^2}{2} \cdot f'(x_i) - \frac{h^3}{3 \cdot 2} \cdot f''(x_i) + \dots \quad (4.22)$$

Вычислим значение (4.21) в точке $(x_i + h)$:

$$f(x_i + h) = f(x_i) + h \cdot f'(x_i) + \frac{h^2}{2!} \cdot f''(x_i) + \dots \quad (4.23)$$

Тогда верно следующее соотношение:

$$h \cdot f'(x_i) = f(x_i + h) - f(x_i) - \frac{h^2}{2!} \cdot f''(x_i) - \dots \quad (4.24)$$

Подставим выражение (4.24) в (4.22):

$$\int_{x_i}^{x_i+h} f(x)dx = \frac{h}{2} \cdot [f(x_i) + f(x_i + h)] - \frac{h^3}{12} \cdot f''(x_i) + \dots$$

Тогда, при достаточно малом значении шага интегрирования h , главный член погрешности R_{0i} вычисления определенного интеграла методом трапеций на отрезке $[x_i, x_i + h]$:

$$R_{0i} = -\frac{h^3}{12} \cdot f''(x_i). \quad (4.25)$$

После суммирования погрешностей на каждом интервале $[x_i, x_i + h]$, получим главный член полной погрешности для интеграла на всем интервале $[x_0, x_n]$:

$$R_0 = \sum_{i=1}^n R_{0i} = -\frac{h^2}{12} \int_{x_0}^{x_n} f''(x) dx. \quad (4.26)$$

Исходя из оценки (4.26), метод трапеций имеет второй порядок.

4.4 Метод Симпсона

Рассмотрим *метод Симпсона*, основанный на квадратичной интерполяции. Для проведения численного интегрирования на отрезках $[x_0, x_2], [x_2, x_4], \dots, [x_i, x_{i+2}], \dots, [x_{n-2}, x_n]$ заменим подынтегральную функцию $f(x)$ интерполяционным полиномом $P_2(x)$ второй степени (параболой):

$$\int_{x_i}^{x_{i+2}} f(x) dx = \int_{x_i}^{x_{i+2}} P_2(x) dx + R,$$

Таким образом, задача состоит в отыскании значений коэффициентов квадратных трехчленов вида:

$$f(x) \approx P_2(x) = a_i x^2 + b_i x + c, \quad x_{i-1} \leq x \leq x_{i+1}.$$

Все коэффициенты полинома вычисляются из равенства табличным значениям функции y_i в точках x_i .

Запишем полином $P_2(x)$ для трех узлов x_0, x_1, x_2 через формулу Ньютона:

$$\begin{aligned} P_2(x) &= a_0 + a_1(x - x_0) + a_2(x - x_0)(x - x_1) = \\ &= y_0 + y_{01}(x - x_0) + y_{012}(x - x_0)(x - x_1), \end{aligned} \quad (4.27)$$

где $a_0 = y_0, a_1 = y_{01} = \frac{y_0 - y_1}{x_0 - x_1} = \frac{y_0 - y_1}{h}$ и $a_2 = y_{012} = \frac{y_{01} - y_{02}}{x_1 - x_2} = \frac{y_0 - 2y_1 + y_2}{2h^2}$ -

разделенные разности 1-го и 2-го порядка.

Пусть $z = x - x_0$, тогда $x - x_1 = z - h$ и выражение (4.27) имеет вид:

$$P_2(z) = a_0 + a_1z + a_2(z^2 - zh). \quad (4.28)$$

Значение определенного интеграла для квадратичного полинома (4.28):

$$\begin{aligned} \int_{x_0}^{x_2} P_2(x) dx &= \int_{x_0}^{x_2} P_2(z) dz = \int_0^{2h} [a_0 + a_1z + a_2(z^2 - zh)] dz = \\ &= \left[a_0z + a_1 \frac{z^2}{2} + \frac{2}{3} a_2 \left(\frac{z^3}{3} - \frac{z^2}{2} h \right) \right]_0^{2h} = 2a_0h + 2a_1h^2 + \frac{2}{3} a_2h^3 = \\ &= \frac{h}{3} (6a_0 + 6a_1h + 2a_2h^2) = \\ &= \frac{1}{3} \left(6y_0 + 6h \frac{y_1 - y_0}{h} + 2h^2 \frac{y_0 - 2y_1 + y_2}{2h^2} \right) = \frac{h}{3} (y_0 + 4y_1 + y_2). \end{aligned} \quad (4.29)$$

Соотношение (4.29) называют **квадратурной формулой Симпсона** (или **формулой параболы**).

Если рассмотреть интерполяционный многочлен Лагранжа второй степени, то аналогично рассуждениям выше можно показать, что элементарная площадь s_i отрезка $[x_{i-1}, x_{i+1}]$ может быть вычислена по формуле:

$$s_i = \frac{h}{3}(y_{i-1} + 4y_i + y_{i+1}). \quad (4.30)$$

Общая площадь фигуры будет иметь вид:

$$S = \sum_{i=1}^{n/2} s_i = \frac{h}{3}(y_0 + 4y_1 + 2y_2 + 4y_3 + 2y_4 + \dots + 2y_{n-2} + 4y_{n-1} + y_n). \quad (4.31)$$

Таким образом, приближенное значение определенного интеграла определено:

$$\int_a^b f(x)dx \approx \frac{h}{3}(y_0 + 4y_1 + 2y_2 + 4y_3 + 2y_4 + \dots + 2y_{n-2} + 4y_{n-1} + y_n). \quad (4.32)$$

Для достижения заданной точности, как правило, используют численный метод «с пересчетом», который заключается в увеличении количества точек n вдвое и сравнении различий в значениях интеграла.

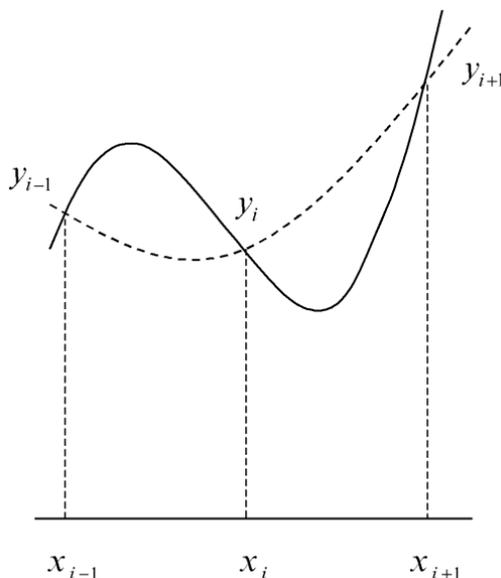


Рисунок 4.3 – Геометрический смысл метода Симпсона

Для оценки погрешности численного интегрирования формулой Симпсона разложим функцию $f(x)$ в ряд Тейлора около точки x_1 на отрезке $[x_0, x_2]$, получим:

$$\int_{x_0}^{x_2} f(x) dx \approx 2h \cdot f(x_1) + \frac{h^3}{3} \cdot f''(x_1) + \frac{h^5}{3 \cdot 4 \cdot 5} f^{IV}(x_1) + O(h^7). \quad (4.33)$$

Соотношение (4.33) эквивалентно:

$$\int_{x_0}^{x_2} f(x) dx \approx (f_0 + 4f_1 + f_2) \cdot h / 3 - \frac{2h^5}{180} f^{IV}(x_1) + O(h^7). \quad (4.34)$$

Тогда главный член погрешности имеет вид:

$$R_{0i} = -\frac{2h^5}{180} f^{IV}(x_1). \quad (4.35)$$

Аналогично применим формулу Симпсона для каждого подинтервала и для достаточно малого шага h на основании метода средних прямоугольников получим:

$$\int_{i=0}^n 2h \cdot f^{IV}(x_{2i+1}) \cong \int_{x_0}^{x_n} f^{IV}(x) dx,$$

а полная погрешность имеет вид:

$$R_0 = -\frac{h^4}{180} \int_{x_0}^{x_n} f^{IV}(x) dx. \quad (4.36)$$

Таким образом, квадратурная формула Симпсона имеет четвертый порядок точности.

5 Численные методы решения уравнений и систем

Физические и технические процессы часто можно представить с помощью алгебраических и трансцендентных уравнений, что может представлять собой самостоятельную задачу или являться частью более сложных задач. В обоих случаях практическая ценность метода в значительной мере определяется быстротой и эффективностью полученного решения.

Задачи, сводящиеся к решению алгебраических и трансцендентных уравнений, можно классифицировать по числу уравнений и в зависимости от предлагаемого характера и числа решений. Одно уравнение будем называть линейным, алгебраическим или трансцендентным в зависимости от того, имеет ли оно одно решение, n решений или неопределенное число решений. Систему уравнений будем называть линейной или нелинейной в зависимости от математической природы входящих в нее уравнений.

5.1 Общие теоретические положения о численном решении уравнений

При решении общетехнических и профессиональных задач довольно часто на практике сталкиваются с задачами решения уравнений. Однако найти точное решение уравнений во всех случаях невозможно и тогда на помощь приходит приближенные методы решений. В настоящий момент существует большое разнообразие численных методов решения линейных и нелинейных уравнений, которые позволяют отыскать решение и найти корень с любой наперед заданной точностью.

Пусть необходимо решить уравнение вида $f(x)=0$, при этом функция $f(x)$ определена и непрерывна на заданном интервале $[a,b]$. Также положим, что первая производная $f'(x)$ и вторая производная $f''(x)$ на заданном отрезке непрерывны. Отметим, что геометрически корни уравнения являются точками

пересечения графика функции $y = f(x)$ с осью OX , т.к. представляют нули функции $y = f(x)$ (рисунок 5.1).

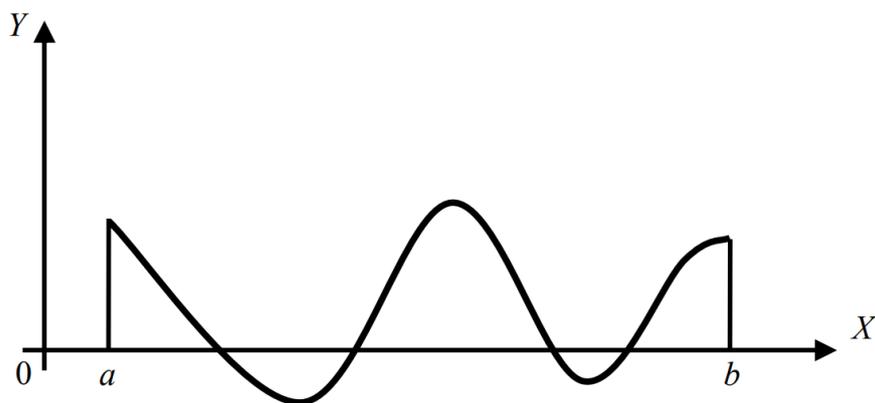


Рисунок 5.1 – Геометрический смысл решения уравнения

Разделим решение уравнения на две подзадачи, позволяющих определить искомые действительные корни:

1. **Отделение (изоляция) корня**, т.е. нахождение подинтервала $[a, b]$ из области определения функции $f(x)$. При этом, необходимо, чтобы на найденном интервале находился один и только один корень уравнения $f(x) = 0$.

2. **Вычисление (уточнение) корня** с заданной точностью.

Рассмотрим два элементарных *свойства*, на которых основано отделение корней уравнения:

- Выражение $f(a)f(b) < 0$ имеет место в том случае, если на различных концах интервала $[a, b]$ функция имеет разные знаки. Тогда внутри интервала $[a, b]$ существует минимум один корень уравнения $f(x) = 0$. Стоит отметить, что данное условие распространяется на случай, когда существует более 1 корня. Например, на рисунке 5.1 $f(a) > 0$, $f(b) < 0$, т.е. $f(a)f(b) < 0$, а интервал $[a, b]$ содержит три корня.

- Производная функции $f'(x)$ не изменяет знака на интервале $[a, b]$, если функция $f(x)$ монотонна, т.е. либо возрастающая, либо убывающая.

Методы отделения корней - аналитический и графический. Наиболее простой способ геометрического отделения корней уравнения $f(x) = 0$ состоит

в построении графика функции $y = f(x)$ и приближенном определении точек пересечения функции с осью ОХ.

Метод аналитического отделения корня состоит в определении интервалов монотонности функции $f(x)$, где производная равна нулю $f'(x) = 0$, а также вычислении значения функции $f(x)$ на границах найденных интервалов для поиска интервала с разными знаками $f(x)$ на концах. Возможен случай, когда интервал, удовлетворяющий данным условиям, не найден. В этом случае уравнение $f(x) = 0$ не имеет корня или корни являются границами интервалов монотонности, т.е. кратными корнями.

Приближенным значением решения уравнения является любая точка отрезка $c \in [a, b]$. Кроме того, разность точного значения x^* и приближенного значения не превышает величину $b - a$, т.е. $|x^* - c| < b - a$. Для повышения точности определения корня, необходимо изменять границы интервала $[a, b]$ так, чтобы уменьшить его длину и обеспечить выполнение условий существования корня. Для поиска численного решения уравнений и построения более точного интервала существует множество методов последовательного приближения. Данные методы основаны на поэтапном (итерационном) сжатии начального отрезка в новый отрезок, длина которого настолько мала, что ей можно пренебречь.

5.1.1 Метод хорд

Основная идея численного решения уравнения методом хорд заключается в построении на отрезке $[a, b]$ хорды АВ, которая соединяет границы дуги функции $y = f(x)$, а приближенным значением решения выбирается c , которое является абсцисса точки пересечения хорды АВ с осью ОХ (рисунок 5.2).

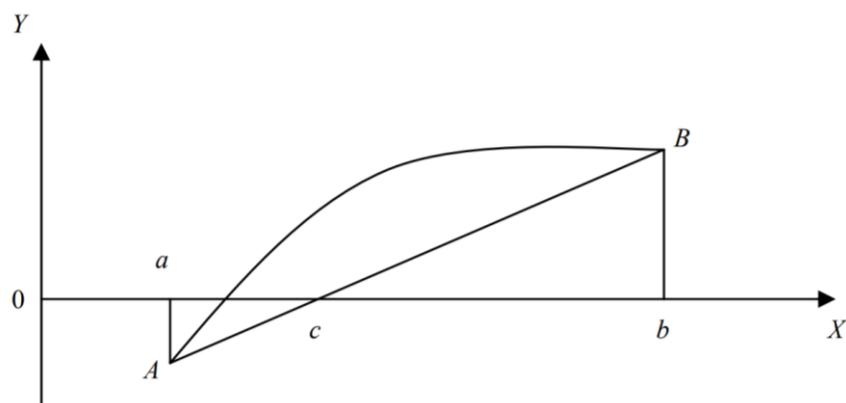


Рисунок 5.2 – Геометрический смысл решения уравнения

Вычислим значение c через уравнение хорды, которая является прямой и проходит через точки $A = (a, f(a))$, $B = (b, f(b))$:

$$\frac{x-a}{b-a} = \frac{y-f(a)}{f(b)-f(a)}. \quad (5.1)$$

Пусть $y = 0$, а $x = c$, тогда уравнение (5.1) примет вид:

$$\frac{c-a}{b-a} = \frac{f(a)}{f(b)-f(a)}. \quad (5.2)$$

Преобразовав выражение (5.2) двумя различными подходами, получим следующие соотношения:

$$c = a - \frac{f(a)(b-a)}{f(b)-f(a)}; \quad c = b - \frac{f(b)(b-a)}{f(b)-f(a)}. \quad (5.3)$$

Ясно, что значение c является первым приближением к корню уравнения, которое обозначим как $x_1 = c$. В случае, если производная функции $f'(x)$ не изменяет знак на интервале $[a, b]$, то приближение x_1 разделит интервал $[a, b]$ на два отрезка $[a, x_1]$ и $[x_1, b]$, а истинный корень уравнения будет принадлежать одному из них. Тогда для поиска интервала, содержащего корень, необходимо сравнить знаки $f(a)$, $f(b)$ и $f(x_1)$.

На рисунке 5.3 показаны все возможные варианты изменения функции $f(x)$ на интервале. Ясно, что при $f'(x) \cdot f''(x) > 0$, истинный корень содержит отрезок $[x_1, b]$, а при $f'(x) \cdot f''(x) < 0$, истинный корень содержит отрезок $[a, x_1]$.

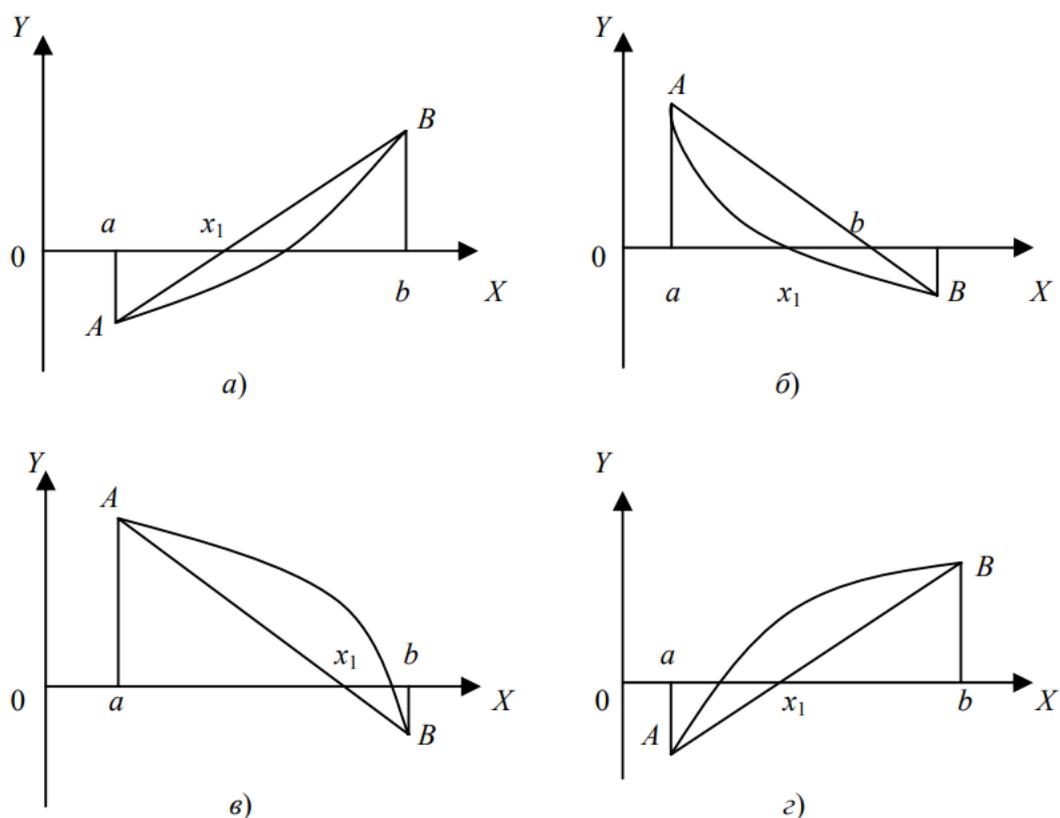


Рисунок 5.3 – Варианты изменения значений функции на интервале

По аналогии на выбранном интервале определим число x_2 , далее x_3, x_4 , и т.д. получим итерационные формулы:

- если $f'(x) \cdot f''(x) > 0$:

$$x_{i+1} = x_i - \frac{f(x_i)(b - x_i)}{f(b) - f(x_i)}; \quad (5.4)$$

- если $f'(x) \cdot f''(x) < 0$:

$$x_{i+1} = x_i - \frac{f(x_i)(x_i - a)}{f(x_i) - f(a)}. \quad (5.5)$$

Итерационный процесс вычисления нового приближения останавливается, если погрешность приближения x_i становится меньше или равна заданного значения. Как правило, для оценки погрешности задается достаточно малое значение $\varepsilon > 0$ и останавливают вычисления, если разность между двумя последними приближениями становится меньше ε , т.е. $|x_{i-1} - x_i| < \varepsilon$. При выполнении данного условия значение x_i выбирают за приближенное значение решения уравнения $f(x) = 0$.

5.1.2 Метод касательных

Основная идея численного решения уравнения методом касательных (Ньютона) заключается в построении касательной на одном из концов дуги АВ графика функции $y = f(x)$ и выборе значения c как абсциссы точки пересечения касательной с осью ОХ (рисунок 5.4).

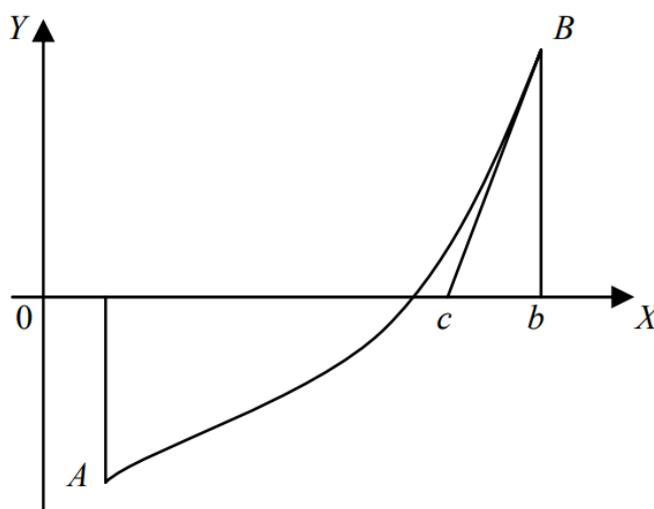


Рисунок 5.4 – Геометрическое представление метода касательных

Составим уравнение касательной к кривой $y = f(x)$ в некоторой точке $(t, f(t))$: $y - f(t) = f'(t)(x - t)$. Согласно методу Ньютона соответствующие уравнения касательных в точках А и В имеют вид $y - f(a) = f'(a)(x - a)$, $y - f(b) = f'(b)(x - b)$. Аналогично предыдущему методу, определим $y = 0$ и

$x = c$, тогда абсцисса точки пересечения касательной с осью OX будет иметь следующий вид:

$$c = a - \frac{f(a)}{f'(a)}; \quad c = b - \frac{f(b)}{f'(b)}. \quad (5.6)$$

Значение c - первое приближением к корню, т.е. $c = x_1$. Так как точка $(x_1, 0)$ расположена с выпуклой стороны кривой $y = f(x)$, то x_1 разобьет интервал $[a, b]$ на два подинтервала $[a, x_1]$ и $[x_1, b]$, а корень уравнения будет принадлежать одному из них. Если $f'(x) \cdot f''(x) > 0$, то корень содержится внутри отрезка $[a, x_1]$, т.е. касательную необходимо провести к точке В, а при $f'(x) \cdot f''(x) < 0$ корень содержится внутри отрезка $[x_1, b]$, т.е. касательную необходимо провести к точке А. Определив следующий отрезок и приближение, повторим итерацию с построением касательной в точке $(x_i, f(x_i))$ и получим последующие приближения по формуле

$$x_{i+1} = x_i - \frac{f(x_i)}{f'(x_i)}. \quad (5.7)$$

Итерационный процесс длится до тех пор, пока разность между последними приближениями больше заданного значения порога погрешности $\varepsilon > 0$, т.е. $|x_{i-1} - x_i| < \varepsilon$. Главный недостаток метода касательных – необходимость вычисления значения производной функции $f'(x)$, что неудобно в случае, если функция $f(x)$ не задана аналитически. Как правило, это касается задач с построением функции $f(x)$ на основе дифференциального уравнения или их системы. Решением данной проблемы является применение метода конечных разностей для выражения производной.

5.1.3 Метод секущих

Основная идея численного решения уравнения методом секущих заключается в замене производной функции соотношением конечных разностей в окрестности рассматриваемой точки, т.е. использования выражения

$$\frac{df(x)}{dx} = \frac{f(x) - f(x+h)}{x - (x+h)} = \frac{f(x) - f(x+h)}{h}, \quad (5.8)$$

где h – некоторая малая величина (шаг).

На рисунке 5.5. представлено геометрическое представление данного подхода. В этом случае формула (5.8) дает возможность провести через рассматриваемую точку секущую.

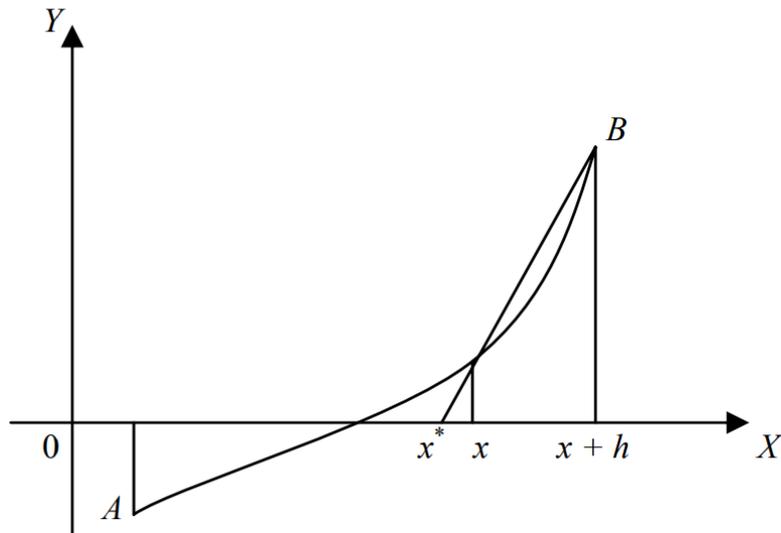


Рисунок 5.5 – Геометрическое представление метода секущих

Итерационная формула аналогична методу касательных и имеет вид:

$$x_{i+1} = x_i - \frac{f(x_i) \cdot h}{f(x_i) - f(x_i - h)}. \quad (5.9)$$

При использовании этого метода следует уменьшать величину h по мере приближения к корню.

5.1.3 Метод простых итераций

Основная идея численного решения уравнения методом простых итераций заключается в построении уравнения $x = g(x)$ из уравнения $f(x) = 0$ путем суммирования к обоим частям уравнения x и введения замены $g(x) = x + f(x)$. Тогда корень уравнения $x = g(x)$ совпадает с корнем уравнения $f(x) = 0$.

Пусть $[a, b]$ – отрезок, содержащий корень x^* , т.е. $x^* = g(x^*)$. Выберем произвольную точку $x_0 \in [a, b]$ и вычислим значение функции $g(x)$:

$$x_1 = g(x_0). \quad (5.10)$$

Тогда на основе значения x_1 можно найти x_2 , а потом x_3, x_4 , и т.д. и получить итерационную формулу:

$$x_{i+1} = g(x_i). \quad (5.11)$$

Если полученная таким образом последовательность x_i сходится, то она сходится к корню x^* , т.е. $\lim_{i \rightarrow \infty} x_i = x^*$ и за конечное число итераций можно получить приближенное значение корня x^* с заданной точностью ε , т.е. $|x_{i-1} - x_i| < \varepsilon$. Однако описанный итерационный процесс не всегда сходится.

Рассмотрим геометрический смысл процесса и его сходимость. Корень уравнения $x = g(x)$, это точка пересечения прямой $y = x$ и графика функции $y = g(x)$ (рисунок 5.6).

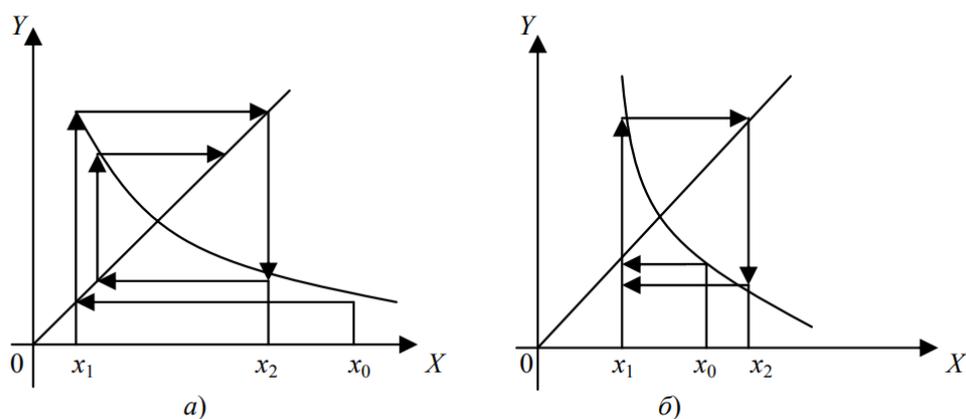


Рисунок 5.6 – Геометрическое представление метода простых итераций

Абсцисса x_1 получена пересечением прямых $y = g(x_0)$ и $y = x$. Абсцисса x_2 получается пересечением прямых $y = g(x_1)$ и $y = x$ и т.д.

На рис. 5.6, а видно, что последовательность x_i сходится к x^* , а на рисунке 5.6, б – расходится. Сходимость процесса зависит от угла наклона линии $x = g(x)$, т.е. от значения $g'(x)$. Если $|g'(x)| < 1$, $x \in [a, b]$, то процесс сходится, при $|g'(x)| > 1$, $x \in [a, b]$ процесс расходится и при $|g'(x)| = 1$, $x \in [a, b]$ процесс может как сходиться, так и расходиться.

5.2 Численные методы решения систем линейных уравнений

Методы решения систем уравнений делятся на точные (прямые) и приближенные (итерационные). Прямые методы позволяют в предположении отсутствия ошибок округления получить точное решение задачи за конечное число арифметических действий. Итерационные методы основаны на использовании повторяющегося процесса и позволяют получить решение в результате последовательных приближений. Рассмотрим наиболее распространенные методы решения систем линейных уравнений.

5.2.1 Метод исключения Гаусса

Рассмотрим систему линейных алгебраических уравнений вида:

$$\begin{cases} a_{11}x_1 + a_{12}x_2 + \dots + a_{1m}x_m = b_1, \\ a_{21}x_1 + a_{22}x_2 + \dots + a_{2m}x_m = b_2, \\ \dots \\ a_{n1}x_1 + a_{n2}x_2 + \dots + a_{nm}x_m = b_n. \end{cases} \quad (5.12)$$

Идея *метода Гаусса* или *метода последовательных исключений* заключается в том, что неизвестные $x_j, j=1, \dots, n-1$ последовательно исключаются из соответствующих уравнений системы (5.12), в результате чего она преобразуется к эквивалентной системе с треугольной матрицей, решение которой находится элементарно.

Рассмотрим данный подход для случая $m=n=3$. При этом значении n система (5.12) примет вид:

$$\begin{cases} a_{11}x_1 + a_{12}x_2 + a_{13}x_3 = b_1, \\ a_{21}x_1 + a_{22}x_2 + a_{23}x_3 = b_2, \\ a_{31}x_1 + a_{32}x_2 + a_{33}x_3 = b_3. \end{cases} \quad (5.13)$$

Введем для коэффициентов матрицы $A = \{a_{ij}\}, i=1, \dots, n, j=1, \dots, m$ и $B = \{b_j\}, j=1, \dots, m$ верхний индекс, для описания итерации, т.е. $a_{ij} := a_{ij}^{(0)}$, $b_j := b_j^{(0)}$. Тогда

$$\begin{cases} a_{11}^{(0)}x_1 + a_{12}^{(0)}x_2 + a_{13}^{(0)}x_3 = b_1^{(0)}, \\ a_{21}^{(0)}x_1 + a_{22}^{(0)}x_2 + a_{23}^{(0)}x_3 = b_2^{(0)}, \\ a_{31}^{(0)}x_1 + a_{32}^{(0)}x_2 + a_{33}^{(0)}x_3 = b_3^{(0)}. \end{cases} \quad (5.14)$$

Первый шаг. Предположим, что $a_{11} = a_{11}^{(0)} \neq 0$ и поделим первое уравнение системы на этот коэффициент. Затем умножим полученное уравнение на $a_{21}^{(0)}$ и результат вычтем из второго уравнения

$$\left(a_{21}^{(0)} - \frac{a_{21}^{(0)}}{a_{11}^{(0)}} a_{11}^{(0)} \right) x_1 + \left(a_{22}^{(0)} - \frac{a_{21}^{(0)}}{a_{11}^{(0)}} a_{12}^{(0)} \right) x_2 + \left(a_{23}^{(0)} - \frac{a_{21}^{(0)}}{a_{11}^{(0)}} a_{13}^{(0)} \right) x_3 = b_2^{(0)} - \frac{a_{21}^{(0)}}{a_{11}^{(0)}} b_1^{(0)}.$$

Коэффициент при x_1 в преобразованном втором уравнении равен нулю, поэтому система (5.14) после данного преобразования примет вид:

$$\begin{cases} a_{11}^{(0)} x_1 + a_{12}^{(0)} x_2 + a_{13}^{(0)} x_3 = b_1^{(0)}, \\ a_{22}^{(1)} x_2 + a_{23}^{(1)} x_3 = b_2^{(1)}, \\ a_{31}^{(0)} x_1 + a_{32}^{(0)} x_2 + a_{33}^{(0)} x_3 = b_3^{(0)}. \end{cases} \quad (5.15)$$

где $a_{2j}^{(1)} = a_{2j}^{(0)} - l_{21} a_{1j}^{(0)}$, $l_{21} = \frac{a_{21}^{(0)}}{a_{11}^{(0)}}$, $b_2^{(1)} = b_2^{(0)} - l_{21} b_1^{(0)}$.

Далее преобразованное первое уравнение с единичным коэффициентом при x_1 умножим на $a_{31}^{(0)}$ и вычтем из третьего уравнения

$$\left(a_{31}^{(0)} - \frac{a_{31}^{(0)}}{a_{11}^{(0)}} a_{11}^{(0)} \right) x_1 + \left(a_{32}^{(0)} - \frac{a_{31}^{(0)}}{a_{11}^{(0)}} a_{12}^{(0)} \right) x_2 + \left(a_{33}^{(0)} - \frac{a_{31}^{(0)}}{a_{11}^{(0)}} a_{13}^{(0)} \right) x_3 = b_3^{(0)} - \frac{a_{31}^{(0)}}{a_{11}^{(0)}} b_1^{(0)}.$$

Коэффициент при x_1 равен нулю, а исходная система преобразуется к виду:

$$\begin{cases} a_{11}^{(0)} x_1 + a_{12}^{(0)} x_2 + a_{13}^{(0)} x_3 = b_1^{(0)}, \\ a_{22}^{(1)} x_2 + a_{23}^{(1)} x_3 = b_2^{(1)}, \\ a_{32}^{(1)} x_2 + a_{33}^{(1)} x_3 = b_3^{(1)}. \end{cases} \quad (5.16)$$

где $a_{3j}^{(1)} = a_{3j}^{(0)} - l_{31} a_{1j}^{(0)}$, $l_{31} = \frac{a_{31}^{(0)}}{a_{11}^{(0)}}$, $b_3^{(1)} = b_3^{(0)} - l_{31} b_1^{(0)}$.

Второй шаг. Матрица с элементами $a_{ij}^{(1)}$ из (5.16) называется **ведущей подматрицей** (второго) шага. Предположим, что $a_{22}^{(1)} \neq 0$, и поделим второе

уравнение этой системы на $a_{22}^{(1)}$. Домножая полученное уравнение на $a_{32}^{(1)}$ и вычитая из третьего уравнения, приходим к системе с треугольной матрицей

$$\begin{cases} a_{11}^{(0)}x_1 + a_{12}^{(0)}x_2 + a_{13}^{(0)}x_3 = b_1^{(0)}, \\ a_{22}^{(1)}x_2 + a_{23}^{(1)}x_3 = b_2^{(1)}, \\ a_{33}^{(2)}x_3 = b_3^{(2)}. \end{cases} \quad (5.17)$$

где $a_{33}^{(2)} = a_{33}^{(1)} - l_{32}a_{22}^{(1)}$, $l_{32} = \frac{a_{32}^{(1)}}{a_{22}^{(1)}}$, $b_3^{(2)} = b_3^{(1)} - l_{32}b_2^{(1)}$.

В общем случае процедура исключения аналогична описанной выше и состоит из $(n-1)$ шагов. Коэффициенты окончательной треугольной системы

$$\begin{aligned} a_{11}^{(0)}x_1 + a_{12}^{(0)}x_2 + a_{13}^{(0)}x_3 + \dots + a_{1n}^{(0)}x_n &= b_1^{(0)}, \\ a_{22}^{(1)}x_2 + a_{23}^{(1)}x_3 + \dots + a_{2n}^{(1)}x_n &= b_2^{(1)}, \\ &\dots \\ a_{ij}^{(i-1)}x_i + \dots + a_{in}^{(i-1)}x_n &= b_i^{(i-1)}, \\ &\dots \\ a_{nn}^{(n-1)}x_n &= b_n^{(n-1)}. \end{aligned} \quad (5.18)$$

и всех промежуточных, равно как и их правые части на k -ом шаге, $k=1,2,\dots,n-1$ вычисляются по формуле:

$$a_{ij}^{(k)} = a_{ij}^{(k-1)} - l_{ik}a_{kj}^{(k-1)}, \quad i, j = k+1, \dots, n, \quad (5.19)$$

$$b_i^{(k)} = b_i^{(k-1)} - l_{ik}b_k^{(k-1)}, \quad i = k+1, \dots, n, \quad (5.20)$$

$$l_{ik} = \frac{a_{ik}^{(k-1)}}{a_{kk}^{(k-1)}}, \quad i = k+1, \dots, n, \quad (5.21)$$

причем $a_{ij} := a_{ij}^{(0)}$, $b_j := b_j^{(0)}$.

Вычисления по формулам (5.19)-(5.21) называют *прямым ходом метода Гаусса*. После этого неизвестные $x_j, j=1, \dots, n-1$ последовательно, начиная с x_n , находятся из (5.18) по формулам

$$x_i = \left[b_i^{(i-1)} - \sum_{j=i+1}^n a_{ij}^{(i-1)} x_j \right] / a_{ii}^{(i-1)}, \quad i = n, \dots, 1. \quad (5.22)$$

Вычисления по этим формулам называют *обратным ходом метода Гаусса* или *обратной подстановкой*.

Замечание 1. В формулах (5.18) при преобразовании системы (5.12) первое уравнение осталось без изменений. Аналогично можно использовать другой вариант исключения, когда первое уравнение (5.12) делится на a_{11} , а вместо (5.18) получается система с единичными коэффициентами x_j при в j -ом уравнении.

Замечание 2. Вычисления по формулам (5.19)-(5.22) возможны лишь в случае, когда все числа

$$a_{ii}^{(i-1)} \neq 0, \quad i = 1, \dots, n. \quad (5.23)$$

Необходимые и достаточные условия (5.23) устанавливаются далее при доказательстве теоремы 2.

5.2.2 Метод LU разложения матриц

Покажем, что метод Гаусса эквивалентен разложению матрицы A системы (5.12) в произведение нижней L и верхней U треугольных матриц с последующим решением вспомогательных систем с данными матрицами.

Рассмотрим данный подход на примере матрицы третьего порядка из (5.13) и введем в рассмотрение матрицы вида

$$E_{21} = \begin{bmatrix} 1 & 0 & 0 \\ -l_{21} & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}, \quad E_{31} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ -l_{31} & 0 & 1 \end{bmatrix}, \quad L_2^{-1} := E_{32} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & -l_{32} & 1 \end{bmatrix},$$

где l_{ik} определяются соотношениями (5.21). Ясно, что переход от системы (5.14) к системе (5.15) может быть осуществлен путем умножения матрицы $A^{(0)}$ системы (5.14) и ее правой части $b^{(0)}$ на матрицу E_{21} , а от (5.15) к (5.16) – умножением соответствующей матрицы и вектора на матрицу E_{31} . Кроме того, выполняется

$$L_1^{-1} := E_{31}E_{21} = E_{21}E_{31} = \begin{bmatrix} 1 & 0 & 0 \\ -l_{21} & 1 & 0 \\ -l_{31} & 0 & 1 \end{bmatrix}$$

и, следовательно, переход к (5.16) осуществляется умножением матрицы системы (5.14) и ее правой части на L_1^{-1} .

Аналогично рассуждая, получим

$$A^{(2)} = L_2^{-1}A^{(1)} = L_2^{-1}L_1^{-1}A^{(0)},$$

где $A^{(k)}$ - матрицы, получаемые на k -ом шаге и, следовательно,

$$A = A^{(0)} = L_1L_2A^{(2)}. \quad (5.24)$$

Рассмотрим операцию умножения матриц:

$$L := L_1L_2 = \begin{bmatrix} 1 & 0 & 0 \\ l_{21} & 1 & 0 \\ l_{31} & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & l_{32} & 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ l_{21} & 1 & 0 \\ l_{31} & l_{32} & 1 \end{bmatrix}.$$

При этом, матрица $A^{(2)}$ из (1.24) есть верхняя треугольная матрица. Обозначая ее через U и принимая во внимание выше сказанное, получаем разложение

$$A = LU, \quad (5.25)$$

где L - нижняя треугольная матрица с единичной главной диагональю.

Все вышесказанное остается справедливым и в общем случае матрицы A порядка n . Тогда матрицы L и U из (5.25) имеют вид

$$L = \begin{bmatrix} 1 & 0 & 0 & \dots & 0 \\ l_{21} & 1 & 0 & \dots & 0 \\ l_{31} & l_{32} & 1 & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots \\ l_{n1} & l_{n2} & l_{n3} & \dots & 1 \end{bmatrix}, \quad U = \begin{bmatrix} u_{11} & u_{12} & u_{13} & \dots & u_{1n} \\ 0 & u_{22} & u_{23} & \dots & u_{2n} \\ 0 & 0 & u_{33} & \dots & u_{3n} \\ \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & 0 & \dots & u_{nn} \end{bmatrix}, \quad (5.26)$$

где элементы l_{ik} матрицы L определяются соотношениями (5.21), а элементы $u_{kj} := a_{kj}^{(k-1)}$ матрицы U - по формулам (5.19).

На основании разложения (5.25) систему (5.12) в виде:

$$Ax = LUx = Ly = b, \quad Ux = y,$$

после чего решение системы (5.12) разделяется на решение двух систем с треугольными матрицами

$$Ly = b \quad \text{и} \quad Ux = y. \quad (5.27)$$

Решение первой из этих систем заменяет преобразование вектора правой части системы (5.12) по формулам (5.20) прямого хода метода Гаусса. Решение второй системы определяется формулами (5.22) обратной подстановки, где $b_i^{(i-1)} = y_i$, а $a_{ij}^{(i-1)} = u_{ij}$.

Замечание 3. Соотношения (5.19) содержат формулы для $u_{kj} = a_{kj}^{(k-1)}$ и промежуточные значения, которые необходимо вычислить и хранить. Преобразуем данные формулы к виду, при котором хранение промежуточных вычислений не требуется.

Пусть матрицы L и U имеют вид (5.26), т.е. их элементы

$$l_{ik} = 0 \quad \text{при} \quad k > i, \quad (5.28)$$

$$u_{kj} = 0 \quad \text{при} \quad k > j. \quad (5.29)$$

Так как $LU = A$, то по правилу умножения матриц получим

$$a_{ij} = \sum_{k=1}^n l_{ik} u_{kj}. \quad (5.30)$$

Преобразуем данное выражение двумя способами. В силу (5.26), (5.28)

$$\sum_{k=1}^n l_{ik} u_{kj} = \sum_{k=1}^{i-1} l_{ik} u_{kj} + l_{ii}^{=1} u_{ij} + \sum_{k=i+1}^n l_{ik}^{=0} u_{kj} = \sum_{k=1}^{i-1} l_{ik} u_{kj} + u_{ij},$$

а в силу (5.29)

$$\sum_{k=1}^n l_{ik} u_{kj} = \sum_{k=1}^{j-1} l_{ik} u_{kj} + l_{ij} u_{jj} + \sum_{k=i+1}^n l_{ik} u_{kj}^{=0} = \sum_{k=1}^{j-1} l_{ik} u_{kj} + l_{ij} u_{jj}.$$

На основе данного соотношения и уравнения (5.30) получим

$$\begin{aligned} u_{ij} &= \left[a_{ij} - \sum_{k=1}^{i-1} l_{ik} u_{kj} \right], \quad i = 1, \dots, n; \quad j = i, \dots, n; \\ l_{ij} &= \frac{1}{u_{jj}} \left[a_{ij} - \sum_{k=1}^{j-1} l_{ik} u_{kj} \right], \quad j = 1, \dots, n; \quad i = j+1, \dots, n. \end{aligned} \quad (5.31)$$

Очевидно, что численно реализовать формулы (5.31) возможно только в том случае, если все $u_{jj} = a_{jj}^{(j-1)}$ отличны от нуля.

Замечание 4. Соотношения (5.31) построены таким образом, что нельзя сначала вычислить все u_{ij} , а затем l_{ij} или наоборот. Для решения поставленной задачи вычисления необходимо выполнять итерационно в соответствии с порядком:

$$\begin{aligned} u_{1j} &= a_{1j} \quad j = 1, \dots, n; \\ l_{i1} &= \frac{a_{i1}}{u_{11}}, \quad i = 2, \dots, n. \\ u_{2j} &= a_{2j} - l_{21}u_{1j}, \quad j = 2, \dots, n; \\ l_{i2} &= (a_{i2} - l_{i1}u_{12}) / u_{22}, \quad i = 3, \dots, n; \end{aligned} \tag{5.32}$$

и т.д., чередуя вычисления строк матрицы U и столбцов матрицы L .

После построения матриц L и U решение систем (5.27) с треугольными матрицами находятся по формулам

$$y_i = b_i - \sum_{k=1}^{i-1} l_{ik} y_k, \quad i = 1, \dots, n; \tag{5.33}$$

(вычисления ведутся сверху вниз) – прямая подстановка,

$$x_k = \frac{1}{u_{kk}} \left[y_k - \sum_{j=k+1}^n u_{kj} x_j \right], \quad k = n, n-1, \dots, 1 \tag{5.34}$$

(вычисления ведутся снизу вверх) – обратная подстановка.

Одной из важнейших характеристик любого численного метода является его трудоемкость, под которой обычно понимают число арифметических действий, необходимых для нахождения искомого решения.

Заметим, что для вычислений по формулам (5.31) и получения треугольного разложения требуется

$$\begin{aligned}
Q &= \sum_{i=1}^n \sum_{j=i}^n (i-1) + \sum_{j=1}^n \sum_{i=j+1}^n j = \sum_{i=1}^n [(i-1)(n-i+1) + i(n-i)] = \\
&= 2 \sum_{i=1}^n [(n+1)i - i^2] - n(n+1) = n(n+1)^2 - \frac{n(n+1)(2n+1)}{3} - n(n+1) = \\
&= \frac{n(n^2-1)}{3} = \frac{n^3}{3} + O(n) \approx \frac{n^3}{3}
\end{aligned}$$

действий умножения и деления.

Для вычисления по формулам (5.33) и (5.34) имеем соответственно

$$\bar{q} = \sum_{i=1}^n (i-1) = \frac{n(n-1)}{2} \quad \text{и} \quad \tilde{c} = \sum_{k=1}^n (k+1) = \frac{n(n+1)}{2},$$

то есть общее число действий для решения системы по формулам (5.33), (5.34) есть

$$q = \bar{q} + \tilde{c} \tag{5.35}$$

5.2.3 Метод простой итерации

Рассмотрим линейную систему из n уравнений с n неизвестными вида:

$$\begin{cases} a_{11}x_1 + a_{12}x_2 + \dots + a_{1n}x_n = b_1, \\ a_{21}x_1 + a_{22}x_2 + \dots + a_{2n}x_n = b_2, \\ \dots \\ a_{n1}x_1 + a_{n2}x_2 + \dots + a_{nn}x_n = b_n. \end{cases} \tag{5.36}$$

В векторно-матричном виде система (5.36) примет форму

$$Ax = b, \tag{5.37}$$

которую можно привести к удобному для интегрирования виду

$$x = Cx + f, \quad (5.38)$$

где $C_{n \times n}$ – некоторая матрица; f - векторный столбец.

Итерационный процесс состоит в задании некоторого начального приближения

$$x^{(0)} = \begin{pmatrix} x_1^{(0)} \\ x_2^{(0)} \\ \dots \\ x_n^{(0)} \end{pmatrix}$$

и подставляя его в правую часть уравнения (5.38), получим следующее приближение:

$$x^{(1)} = Cx^{(0)} + f,$$

и

$$x^{(k+1)} = Cx^{(k)} + f, \quad k = 0, 1, 2, \dots \quad (5.39)$$

В развернутой записи система (5.39) имеет вид:

$$\begin{cases} x_1^{(k+1)} = c_{11}x_1^{(k)} + c_{12}x_2^{(k)} + \dots + c_{1n}x_n^{(k)} + f_1, \\ \dots \\ x_n^{(k+1)} = c_{n1}x_1^{(k)} + c_{n2}x_2^{(k)} + \dots + c_{nn}x_n^{(k)} + f_n. \end{cases} \quad (5.40)$$

Последовательно выполняя итерации получим приближения $x^{(1)}, x^{(2)}, \dots, x^{(k)}, \dots$. Рассмотрим более подробно вопрос о сходимости данной последовательности.

Условия сходимости итерационного процесса.

Введем для описания сходимости последовательности, следующие обозначения.

Обобщением понятия *длины* m -мерного вектора x является *норма* вектора $\|x\|$, из множества способов введения которой в численных методах наиболее употребительными являются следующие нормы:

$$\|x\|_1 = \sum_{i=1}^m |x_i|, \quad \|x\|_2 = \sqrt{\sum_{i=1}^m |x_i|^2}, \quad \|x\|_\infty = \max_{1 \leq i \leq m} |x_i|. \quad (5.41)$$

Первая и вторая нормы являются частными случаями (при $p=1$ и $p=2$) общей *нормы Минковского*:

$$\|x\|_p = \left(\sum_{i=1}^m |x_i|^p \right)^{1/p}, \quad p \geq 1,$$

а последняя норма получается при предельном переходе к $p \rightarrow \infty$. Заметим, что норма $\|x\|_2$ называется *евклидовой*.

Нормой матрицы A называется значение

$$\|A\| = \max_{x \neq 0} \frac{\|Ax\|}{\|x\|}. \quad (5.42)$$

Соответствующие нормы матрицы имеют вид:

$$\|A\|_1 = \max_{1 \leq j \leq m} \sum_{i=1}^m |a_{ij}|, \quad \|A\|_2 = \sqrt{\sum_{i=1}^m \sum_{j=1}^m |a_{ij}|^2}, \quad \|A\|_\infty = \max_{1 \leq i \leq m} \sum_{j=1}^m |a_{ij}|. \quad (5.43)$$

Теорема о сходимости итерационного процесса.

Пусть выполнено условие

$$\|C\| < 1. \quad (5.44)$$

Тогда:

- 1) решение x системы (5.36) существует и единственно;
- 2) при *произвольном* векторе начального приближения $x^{(0)}$ метод простой итерации сходится к точному решению системы \bar{x}

$$\lim_{k \rightarrow \infty} x^{(k)} = \bar{x},$$

и справедлива следующая оценка абсолютной погрешности k -го приближения

$$\Delta = \|x^{(k)} - \bar{x}\| \leq \frac{q}{1-q} \|x^{(k)} - x^{(k-1)}\|, \quad (5.45)$$

где в качестве коэффициента сжатия $q = \|C\|$ можно использовать *любую* норму матрицы (5.43), удовлетворяющую условию (5.44).

Замечание 1. Из сходимости итераций по одной из норм следует и сходимость по другой норме, т. е., например, если $\|x^{(k)} - \bar{x}\|_1 \rightarrow 0$ при $k \rightarrow \infty$, то и $\|x^{(k)} - \bar{x}\|_2 \rightarrow 0$ при $k \rightarrow \infty$, и наоборот.

Замечание 2. Приведем *критерий достижения требуемой точности ε* : вычисления можно остановить, когда абсолютная погрешность Δ , оцениваемая по формуле (5.45), станет меньше ε . На практике необходимо проверять выполнение условия для **каждой** компоненты вектора k -го приближения

$$\frac{q}{1-q} \|x_i^{(k)} - x_i^{(k-1)}\| < \varepsilon, \quad i = 1, 2, \dots, n. \quad (5.46)$$

Замечание 3. Простейший способ преобразования системы (5.37) к виду (5.38), удобному для итераций, состоит в том, чтобы в каждом уравнении (5.36) выразить соответствующую неизвестную через все остальные:

$$\begin{cases} x_1 = \frac{1}{a_{11}}(b_1 - a_{12}x_2 - a_{13}x_3 - \dots - a_{1n}x_n), \\ x_2 = \frac{1}{a_{22}}(b_2 - a_{21}x_1 - a_{23}x_3 - \dots - a_{2n}x_n), \\ \dots \\ x_n = \frac{1}{a_{nn}}(b_n - a_{n1}x_1 - a_{n2}x_2 - \dots - a_{n,n-1}x_{n-1}). \end{cases} \quad (5.47)$$

Данный способ можно применять только в случае, если диагональные элементы матрицы A отличны от нуля.

5.2.4 Метод Зейделя

Метод простой итерации для решения системы уравнений базируется на вычисления вектора $(k+1)$ -й итерации с использованием компонент вектора предыдущей k -й итерации. *Метод Зейделя* является модификацией метода *простой итерации* с несколько усложненным итерационным процессом: при вычислении вектора $(k+1)$ -й итерации используются не только компоненты вектора k -й итерации, но и *все вычисленные* к данному моменту компоненты вектора текущей $(k+1)$ -й итерации:

$$\begin{aligned} x_1^{(k+1)} &= c_{11}x_1^{(k)} + c_{12}x_2^{(k)} + c_{13}x_3^{(k)} + \dots + c_{1n}x_n^{(k)} + f_1, \\ x_2^{(k+1)} &= c_{21}x_1^{(k+1)} + c_{22}x_2^{(k)} + c_{23}x_3^{(k)} + \dots + c_{2n}x_n^{(k)} + f_2, \\ x_3^{(k+1)} &= c_{31}x_1^{(k+1)} + c_{32}x_2^{(k+1)} + c_{33}x_3^{(k)} + \dots + c_{3n}x_n^{(k)} + f_3, \\ x_4^{(k+1)} &= c_{41}x_1^{(k+1)} + c_{42}x_2^{(k+1)} + c_{43}x_3^{(k+1)} + \dots + c_{4n}x_n^{(k)} + f_4, \\ &\dots \dots \\ x_n^{(k+1)} &= c_{n1}x_1^{(k+1)} + c_{n2}x_2^{(k+1)} + c_{n3}x_3^{(k+1)} + \dots + c_{nn}x_n^{(k)} + f_n. \end{aligned} \quad (5.48)$$

В сходящемся итерационном процессе каждое следующее приближение становится точнее, поэтому метод Зейделя обычно демонстрирует более быструю сходимость, чем метод простой итерации. Условия сходимости для метода Зейделя аналогичны методу простой итерации.

5.3 Численные методы решения систем нелинейных уравнений

Решение систем нелинейных и (или) трансцендентных уравнений является гораздо более сложной задачей, чем решение линейных систем. Наибольшую трудность представляет этап отделения корней. Для системы с двумя неизвестными можно использовать геометрические построения, но для системы с $n > 2$ неизвестными такой подход становится неприменимым.

Описанные далее итерационные методы уточнения корней основаны на том, что задача отделения корней уже решена и имеется достаточно малая область изоляции, в которой находится подлежащий уточнению корень. Для простоты анализа будем рассматривать нелинейную систему из двух уравнений с двумя неизвестными

$$\begin{cases} f_1(x, y) = 0, \\ f_2(x, y) = 0. \end{cases} \quad (5.49)$$

Предположим, что начальные приближения корней $x^{(0)}, y^{(0)}$ в области изоляции получены. Преобразуем систему (5.49) к виду, удобному для применения итерационных методов:

$$\begin{cases} x = F_1(x, y), \\ y = F_2(x, y). \end{cases} \quad (5.50)$$

Подставляя в правые части уравнений (5.50) вместо x, y начальные приближения, получим первые приближения к корням

$$\begin{cases} x^{(1)} = F_1(x^{(0)}, y^{(0)}), \\ y^{(1)} = F_2(x^{(0)}, y^{(0)}). \end{cases} \quad (5.51)$$

Аналогично вычисляются последующие приближения для *метода простой итерации*:

$$\begin{cases} x^{(k+1)} = F_1(x^{(k)}, y^{(k)}), \\ y^{(k+1)} = F_2(x^{(k)}, y^{(k)}). \end{cases} \quad (5.52)$$

Для метода Зейделя итерационные формулы примут вид:

$$\begin{cases} x^{(k+1)} = F_1(x^{(k)}, y^{(k)}), \\ y^{(k+1)} = F_2(x^{(k+1)}, y^{(k)}). \end{cases} \quad (5.53)$$

Рассмотрим сходимость данного итерационного процесса. Введем вектор неизвестных значений

$$X = \begin{pmatrix} x \\ y \end{pmatrix}$$

и вектор-функцию

$$F(x, y) = F(X) = \begin{pmatrix} F_1(x, y) \\ F_2(x, y) \end{pmatrix} = \begin{pmatrix} F_1(X) \\ F_2(X) \end{pmatrix}.$$

Система (5.50) примет вид:

$$X = F(X).$$

Введем матрицу Якоби $F'(X)$, составленную из частных производных по аргументам,

$$F'(X) = \begin{pmatrix} \frac{\partial F_1(X)}{\partial x} & \frac{\partial F_1(X)}{\partial y} \\ \frac{\partial F_2(X)}{\partial x} & \frac{\partial F_2(X)}{\partial y} \end{pmatrix}.$$

Теорема о сходимости итерационного процесса. Пусть в некоторой окрестности решения \bar{X} системы (56) функции $F_1(X)$, $F_2(X)$ дифференцируемы и выполнено условие

$$\|F'(X)\| = \left\| \begin{pmatrix} \frac{\partial F_1(X)}{\partial x} & \frac{\partial F_1(X)}{\partial y} \\ \frac{\partial F_2(X)}{\partial x} & \frac{\partial F_2(X)}{\partial y} \end{pmatrix} \right\| \leq q < 1, \quad (5.54)$$

где константа q - коэффициент сжатия. В этом случае независимо от выбора начального приближения $X^{(0)}$ из данной окрестности:

- 1) итерационные последовательности $\{x^{(i)}\}$ и $\{y^{(i)}\}$ не выходят из этой окрестности и сходятся соответственно к \bar{x} и \bar{y} - решению системы;
- 2) справедлива оценка абсолютной погрешности k -го приближения

$$\Delta = \|X^{(k)} - \bar{X}\| \leq \frac{q}{1-q} \|X^{(k)} - X^{(k-1)}\|, \quad (5.55)$$

которая используется в критерии достижения требуемой точности: как только оценка (5.55) становится меньше ε - требуемой точности, итерационный процесс останавливается.

6 Численные методы решения дифференциальных уравнений и систем

Множество физических, экономических, технических и других систем и процессов в них можно описывать через динамические уравнения, передающие характер их изменения. Из курса классической математики известно, что скорость изменения некоторого процесса можно выразить через производную, а уравнения содержащие производные любого порядка называются *дифференциальными*. Поиск решения подобных задач в аналитическом виде, как правило, затруднен, а решение и вовсе может отсутствовать. В рамках данной главы рассмотрены основные численные методы решения дифференциальных уравнений и их систем.

6.1 Численные методы решения обыкновенных дифференциальных уравнений

Дифференциальные уравнения, содержащие одну независимую переменную и ее производную, называются *обыкновенными*.

Формально *обыкновенными дифференциальными уравнениями* (ОДУ) называются такие уравнения, которые содержат одну или несколько производных от искомой функции $y = y(x)$, их можно представить в виде:

$$F(x, y, y', \dots, y^{(n)}) = 0, \quad (6.1)$$

где x - независимая переменная.

При этом, n - высший порядок производной, входящей в уравнение, называется порядком ДУ. Если из общего вида (6.1) можно выразить старшую производную в явном виде:

$$y^{(n)} = f(x, y, y', \dots, y^{(n-1)}), \quad (6.2)$$

то уравнение называется *разрешенным относительно старшей производной*.

Если ДУ (6.1) включает только первую степень неизвестной функции и ее производной, то оно называется *линейным дифференциальным уравнением*.

Любая функция $y = \varphi(x)$, которая после подстановки в ДУ (6.1) превращает его в тождество, называется *решением*.

Отметим, что *общее решение* ОДУ n -го порядка содержит n произвольных постоянных и имеет вид:

$$y = \varphi(x, c_1, c_2, \dots, c_n). \quad (6.3)$$

Если произвольным постоянным из (6.3) задать конкретные значения (например, в соответствии с граничными условиями в задаче), то получившаяся функция будет называться *частным решением*.

Существуют различные способы определения дополнительных условий для задания частного решения ДУ: *краевая задача* и *задача Коши*. Примером таких условий могут выступать значения неизвестной функции или её производной в некоторых точках.

Если дополнительные условия заданы в более чем одной точке, то данная задача называется *краевой задачей*, а условия - *граничными*.

Если подобные дополнительные условия заданы в одной и той же точке $x = x_0$, то данная задача называется *задачей Коши*, а условия - *начальными*.

Рассмотрим два класса методов решения задачи Коши: *одношаговые методы* и *методы прогноза и коррекции* (многошаговые).

Для *одношаговых методов* характерно следующее – для вычисления следующей точки для функции $y = f(x)$ необходимо вычислить значение точки на предыдущем шаге. К группе таких методов относятся методы Эйлера и методы Рунге-Кутты, которые рассмотрим в рамках данной главы.

Для *многошаговых методов* вычисление следующей точки для функции $y = f(x)$ требует вычисления значений точек на нескольких предыдущих шагах. К группе таких методов относятся методы Адамса, Хемминга, Милна и Гира.

Методы решения ОДУ делятся на *графические*, *аналитические*, *приближенные* и *численные*. В рамках данной главы рассмотрим численные методы решения ДУ, которые можно применять во множестве прикладных задач научно-технического характера и, как правило, упрощают поиск решения. Стоит отметить, что данные методы рекомендуют применять на быстродействующих ЭВМ с достаточным объемом оперативной памяти для получения более точных результатов.

Среди численных методов решения ДУ *метод конечных разностей* является наиболее распространенным и универсальным. Область определения искомой функции (непрерывного изменения аргумента) заменяется на дискретное множество точек (*узлов*), которые составляют *разностную сетку*. При этом, искомая функция непрерывного аргумента приближенно заменяется функцией дискретного аргумента на заданной сетке - *сеточной функцией*. Тогда исходное ДУ можно заменить разностным уравнением относительно сеточной функции, производные которой выражены разностными соотношениями. Данная замена ДУ разностным уравнением называется *аппроксимацией* на сетке и соответствует задаче отыскания значений сеточной функции на узлах.

6.1.1 Одношаговые методы решения ОДУ

Рассмотрим группу одношаговых численных методов решения обыкновенных дифференциальных уравнений.

6.1.1.1 Метод Эйлера

Метод Эйлера – простейший метод решения задачи Коши, который позволяет интегрировать ДУ 1-го порядка. Точность данного метода небольшая и на практике данный метод используют при крайней необходимости.

Рассмотри ДУ первого порядка

$$y' = f(x, y), \quad (6.4)$$

относительно независимой переменной x , искомой функции $y = y(x)$ и ее производной $y' = y'(x)$.

Начальные условия заданы следующим образом:

$$y_0 = y(x_0). \quad (6.5)$$

Решим задачу Коши (6.4)-(6.5) методом Эйлера, который основан на разложении неизвестной функции $y(x)$ в ряд Тейлора в окрестностях узлов $x = x_i$, ($i = 0, 1, \dots, n$). Данное разложение в точке $(x_0 + \Delta x)$ будет иметь вид:

$$y(x_0 + \Delta x) = y(x_0) + \Delta x \cdot y'(x_0) + \frac{\Delta x^2}{2!} \cdot y''(x_0) + \frac{\Delta x^3}{3!} \cdot y'''(x_0) + \dots \quad (6.6)$$

В разложении (6.6) отбросим слагаемые с производными второго порядка и выше, получим следующее приближение:

$$y(x_0 + \Delta x) \approx y(x_0) + \Delta x \cdot y'(x_0) + O(h^2). \quad (6.7)$$

Так как $x_0 + \Delta x = x_1$, то подставив начальное условие в выражение (6.7) получим:

$$y(x_0 + \Delta x) = y(x_1) = y(x_0) + \Delta x \cdot f(x_0, y_0).$$

Считая шаг сетки постоянным $\Delta x = h = const$, получим

$$y(x_1) = y(x_0) + h \cdot f(x_0, y_0).$$

Продолжая данный итерационный процесс, получим значения для любой точки сетки:

$$\begin{aligned}
y(x_2) &= y(x_1) + h \cdot f(x_1, y_1), \\
y(x_3) &= y(x_2) + h \cdot f(x_2, y_2), \\
&\dots \\
y(x_i) &= y(x_{i-1}) + h \cdot f(x_{i-1}, y_{i-1}).
\end{aligned}
\tag{6.8}$$

Таким образом, итерационный процесс можно представить в виде:

$$y(x_n) = y(x_{n-1}) + h \cdot f(x_{n-1}, y_{n-1}). \tag{6.9}$$

Выражение (6.9) позволяет вычислить приближенное значение искомой функции при малом смещении h от начальной точки с ошибкой порядка h^2 .

Графическая интерпретация метода Эйлера.

Графическая интерпретация метода Эйлера (рис.6.1) описывает взаимосвязь приближенного значения функции с касательной к кривой в точке, т.е.

$$\begin{aligned}
y_1 - y_0 &= \operatorname{tg}\alpha \cdot (x_1 - x_0) = y'(x_0) \cdot (x_1 - x_0); \\
y_1 &= y_0 + h \cdot y'(x_0).
\end{aligned}$$

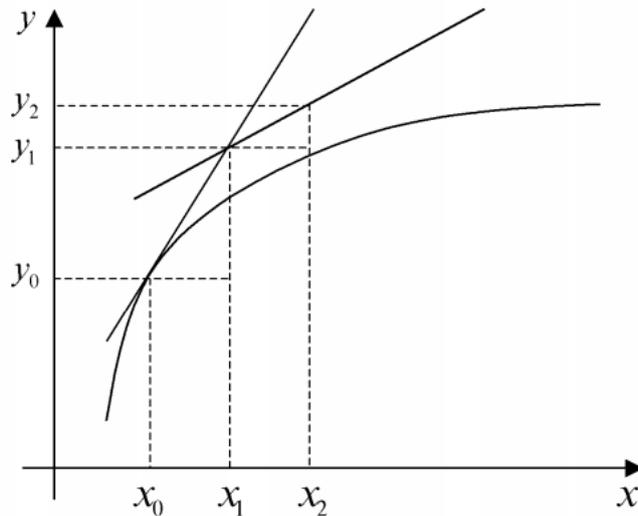


Рисунок 6.1 – Геометрическое представление метода Эйлера

Заметим, что метод Эйлера имеет низкую точность, а погрешность последующих шагов увеличивается (накапливается). В связи с этим, для получения более точных результатов необходимо рассмотреть другие подходы.

6.1.1.2 Модифицированный метод Эйлера

Модифицированный метод Эйлера – метод решения задачи Коши с точностью второго порядка, требующий относительно предыдущего метода дополнительных затрат времени ЭВМ.

Рассмотрим разложение функции в ряд Тейлора:

$$y(x_0 + \Delta x) = y(x_0) + \Delta x \cdot y'(x_0) + \frac{\Delta x^2}{2!} \cdot y''(x_0) + \frac{\Delta x^3}{3!} \cdot y'''(x_0) + \dots \quad (6.10)$$

Повысим точность вычислений, отбросив слагаемые с производными третьего порядка и выше. Однако для вычисления второй производной воспользуемся ее аппроксимацией конечной разностью:

$$y''(x_0) = \frac{y'(x_0 + \Delta x) - y'(x_0)}{\Delta x} = \frac{y'(x_0 + h) - y'(x_0)}{h}. \quad (6.11)$$

Подставим выражение (6.11) в ряд Тейлора, учитывая $\Delta x = h$, получим

$$\begin{aligned} y(x_0 + \Delta x) &= y(x_0) + h \cdot y'(x_0) + \frac{h^2}{2!} \cdot \left[\frac{y'(x_0 + h) - y'(x_0)}{h} \right] = \\ &= y(x_0) + h \cdot y'(x_0) + \frac{h}{2} \cdot y'(x_0 + h) - \frac{h}{2} \cdot y'(x_0) = y(x_0) + \frac{h}{2} \cdot [y'(x_0 + h) + y'(x_0)] = \\ &= y(x_0) + \frac{h}{2} \cdot [y'(x_1) + y'(x_0)]. \end{aligned}$$

Продолжая данный итерационный процесс для сколь угодно большого количества шагов, получим соотношение:

$$y_{n+1} = y_n + \frac{h}{2} \cdot [y'_{n+1} + y'_n]. \quad (6.11)$$

Погрешность данного метода имеет порядок h^3 , ее можно повысить с сохранением большего числа членов ряда Тейлора. Заметим, что в модифицированном методе Эйлера сначала необходимо определить значение функции в следующей точке по методу Эйлера:

$$y^*(x_{n+1}) = y(x_n) + \frac{h}{2} \cdot f(x_n, y(x_n)) \text{ или } y_{n+1}^* = y_n + \frac{h}{2} \cdot f(x_n, y_n).$$

Данное значение применяют для вычисления приближенного значения производной. Найдем более точное значение y_{n+1} как среднее между значением производной и ее значением в начале интервала:

$$y_{n+1} = y_n + \frac{h}{2} \cdot [f(x_n, y_n) + y_{n+1}^*].$$

Таким образом, имеем:

$$\begin{aligned} y_1 &= y_0 + h \cdot \operatorname{tg} \alpha_{cp} = y_0 + h \cdot \frac{y_2 - y_0}{2h} = y_0 + h \cdot \frac{y_2 - y_1^* + y_1^* - y_0}{2h} = \\ &= y_0 + h \cdot \frac{(y_2 - y_1^*) + (y_1^* - y_0)}{2h} = y_0 + \frac{h}{2} \cdot \left[\frac{y_2 - y_1^*}{h} + \frac{y_1^* - y_0}{h} \right] = \\ &= y_0 + \frac{h}{2} \cdot [\operatorname{tg} \alpha_1 + \operatorname{tg} \alpha_2] = y_0 + \frac{h}{2} \cdot (y'_1 + y'_0). \end{aligned}$$

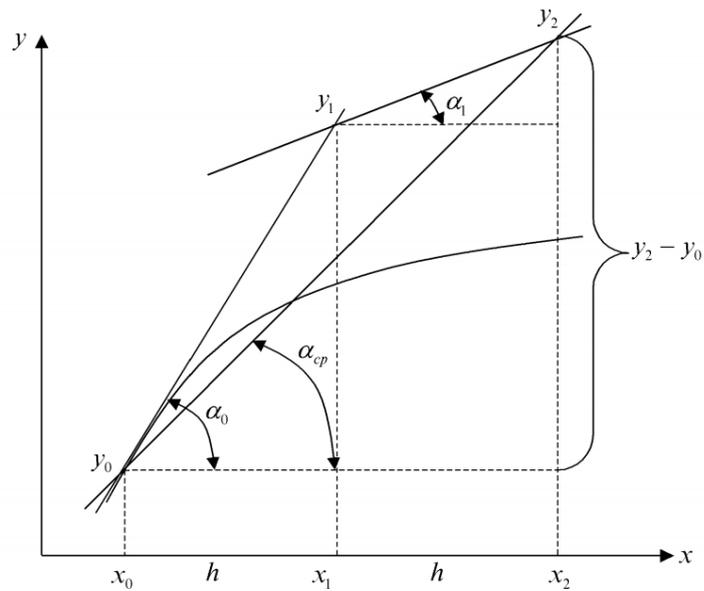


Рисунок 6.2 – Геометрическое представление модифицированного метода Эйлера

6.1.1.3 Методы Рунге-Кутты

Как обсуждалось ранее, при сохранении большего количества слагаемых в приближенной аппроксимации функций рядом Тейлора увеличивается точность. Однако для вычисления n -го члена ряда необходимо определить производную n -го порядка. Для модифицированного метода Эйлера вычисление 2-й производной достаточно было определить наклоны кривой на концах рассматриваемого отрезка. Таким образом, для вычисления 3-й производной необходимо определить значения 2-й производной как минимум в 2-х точках, т.е. дополнительно вычислить наклон кривой в некоторой точке интервала h между точками x_n и x_{n+1} . Для определения производных более высокого порядка потребуется еще больше дополнительных вычислений внутри интервала.

Рассматриваемые методы Рунге-Кутты позволяют рассчитывать координаты внутренних точек и формируют семейство методов решения ДУ 1-го порядка. Наиболее распространенным среди них является метод,

сохраняющий члены ряда с ошибкой h^5 . Итерационные формулы при этом имеют вид:

$$y_{n+1} = y_n + \frac{k_0 + 2k_1 + 2k_2 + k_3}{6}, \quad (6.12)$$

где

$$\begin{aligned} k_0 &= h \cdot f(x_n, y_n), \\ k_1 &= h \cdot f\left(x_n + \frac{1}{2}h, y_n + \frac{1}{2}k_0\right), \\ k_2 &= h \cdot f\left(x_n + \frac{1}{2}h, y_n + \frac{1}{2}k_1\right), \\ k_3 &= h \cdot f(x_n + h, y_n + k_2). \end{aligned}$$

Заметим, что метод Эйлера и его модификация – методы Рунге-Кутты первого и второго порядка соответственно. Более высокая точность метода Рунге-Кутты позволяет смягчить недостаток большого объема вычислений. Отметим, что величину шага интегрирования h необходимо выбирать исходя из максимально допустимой ошибки на шаге.

6.1.2 Многошаговые методы решения ОДУ

Рассмотрим некоторые многошаговые (многоточечные) численные методы решения обыкновенных дифференциальных уравнений. Данная группа алгоритмов основывается на представлении правых частей ОДУ или интегральных кривых $y_k(x)$ интерполяционными полиномами.

6.1.2.1 Метод Адамса

Метод Адамса – четырехточечный метод решения задачи Коши, аппроксимирующий интерполяционными полиномами правые части ОДУ.

Рассмотрим значения искомым функций y_1, y_2, y_3 заданного ДУ в точках x_1, x_2, x_3 любым из выше рассмотренных методов. При этом зависимость правой части ДУ $f(x, y)$ на интегральной кривой с некоторыми начальными условиями

будет является функцией одного аргумента x : $f(x) = f(x, y(x))$, значения которой обозначим в соответствующих точках как f_0, f_1, f_2, f_3 . Аппроксимируем функцию $f(x)$ интерполяционным полиномом Ньютона в окрестности рассматриваемых точек:

$$f(x) = f_0 + f_{01}(x - x_0) + f_{012}(x - x_0)(x - x_1) + f_{0123}(x - x_0)(x - x_1)(x - x_2), \quad (6.13)$$

где $f_{01}, f_{012}, f_{0123}$ – разделенные разности.

Разложим в ряд Тейлора вблизи x_3 искомое решение в $x_4 = x_3 + h$:

$$y_4 = y_3 + hf(x_3) + \frac{h^2}{2!} f'(x_3) + \frac{h^3}{3!} f''(x_3) + \frac{h^4}{4!} f'''(x_3) + O(h^5), \quad (6.14)$$

где $f'(x_3), f''(x_3), f'''(x_3)$ – производные по x от правой части ДУ в x_3 .

Производные функции $f(x)$ имеют вид:

$$\begin{aligned} f'(x) &= f_{01} + f_{012}(2x - x_0 - x_1) + f_{0123}[(x - x_0)(x - x_1) + \\ &\quad + (x - x_0)(x - x_2) + (x - x_1)(x - x_2)], \\ f''(x) &= 2f_{012} + 2f_{0123}(3x - x_0 - x_1 - x_2), \\ f'''(x) &= 6f_{0123}. \end{aligned}$$

Рассмотрим равномерную сетку с постоянным шагом, тогда разностные формулы производных в точке $x = x_3$ примут вид:

$$\begin{aligned} f'(x_3) &= \frac{-2f_0 + 9f_1 - 18f_2 + 11f_3}{6h}, \\ f''(x_3) &= \frac{-f_0 + 4f_1 - 5f_2 + 2f_3}{h^2}, \\ f'''(x_3) &= \frac{-f_0 + 3f_1 - 3f_2 + f_3}{h^3}. \end{aligned} \quad (6.15)$$

Подставим соотношения (6.15) в выражение (6.14) и получим экстраполяционную формулу Адамса:

$$y_4 = y_3 + hf(x_3) + \frac{(55f_3 - 59f_2 + 37f_1 - 9f_0)h}{24} + O(h^5). \quad (6.16)$$

Соотношение (6.16) показывает, что локальная погрешность имеет 5 порядок, а глобальная погрешность имеет 4-ый порядок. Заметим, что увеличивая количество слагаемых ряда (6.14) можно увеличить порядок разностных схем Адамса.

Формула Адамса (6.16) имеет остаточный член $\frac{251}{750}h^5 f^{IV}(x)$, который имеет достаточно большой коэффициент и объясняет расположение точки x_4 вне интервала. Таким образом, данный подход является экстраполяцией и его погрешность больше методов интерполяции, что подтверждается и другими исследованиями.

6.1.2.2 Метод Гира

Метод Гира – неявные алгоритмы решения задачи Коши, эффективно определяющие решение «жестких» уравнений, для которых при быстро затухающих возмущениях происходит медленное изменение решений.

Построим для задачи Коши (6.4)-(6.5) методом Рунге-Кутты любого порядка решения y_1, y_2, y_3 в точках x_1, x_2, x_3 . Аппроксимируем искомую функцию $y(x)$ интерполяционным полиномом Ньютона 4-й степени:

$$y(x) = y_0 + y_{01}(x - x_0) + y_{012}(x - x_0)(x - x_1) + y_{0123}(x - x_0)(x - x_1)(x - x_2) + y_{01234}(x - x_0)(x - x_1)(x - x_2)(x - x_3), \quad (6.17)$$

где $y_{01}, y_{012}, y_{0123}, y_{01234}$ – разделенные разности.

Производная функции $y(x)$ имеют вид:

$$\begin{aligned}
y'(x) = & y_{01} + y_{012}(2x - x_0 - x_1) + y_{0123}[(x - x_0)(x - x_1) + (x - x_0)(x - x_2) + \\
& + (x - x_1)(x - x_2)] + y_{01234}[(x - x_0)(x - x_1)(x - x_2) + (x - x_0)(x - x_1)(x - x_3) + \\
& + (x - x_0)(x - x_2)(x - x_3) + (x - x_1)(x - x_2)(x - x_3)].
\end{aligned} \quad (6.18)$$

Разделенные разности на равномерной сетке с постоянным шагом можно представить через значения аппроксимируемой функции:

$$\begin{aligned}
y_{01} &= \frac{y_1 - y_0}{h}, \\
y_{012} &= \frac{y_2 - 2y_1 + y_0}{2h^2}, \\
y_{0123} &= \frac{y_3 - 3y_2 + 3y_1 - y_0}{6h^3}, \\
y_{01234} &= \frac{y_4 - 4y_3 + 6y_2 - 4y_1 + y_0}{24h^4},
\end{aligned} \quad (6.19)$$

где $h = x_{i+1} - x_i$.

Подставим соотношения (6.19) в выражение (6.18) при $x = x_4$ и получим

$$y'(x_4) = \frac{3y_0 - 16y_1 + 36y_2 - 48y_3 + 25y_4}{12h}. \quad (6.20)$$

Учитывая, что $y'(x_4) = f(x_4, y_4)$, получим неявную схему Гира 4-го порядка

$$y_4 = \frac{3[4h \cdot f(x_4, y_4) - y_0] + 16y_1 - 36y_2 + 48y_3}{25}. \quad (6.21)$$

По аналогии, изменяя количество точек(узлов) на которых строится схема Гира можно получить разностные формулы различного порядка. На практике для вычисления y_4 в (6.21), как правило, рекомендуют применять метод Ньютона решения нелинейных алгебраических уравнений. Данный метод требует задать начальное приближение к значению y_4 .

Если в выражение (6.18) подставить $x = x_3$ и приравнять к правой части уравнения (6.4) при $x = x_3$, то получим, что начальное приближение можно взять как

$$y_4 = 4h \cdot f(x_3, y_3) - \frac{y_0 - 10y_3}{3} - 2y_1 + 6y_2. \quad (6.22)$$

6.1.3 Нейросетевой подход к решению ОДУ

Рассмотрим применение нейросетевого подхода для решения ОДУ первого порядка, которые имеют вид:

$$\begin{cases} \frac{dy}{dx} = f(x, y), \\ u(0) = A. \end{cases} \quad (6.23)$$

где $x \in [0, 1]$.

Согласно теореме Колмогорова любую функцию многих переменных можно представить в виде суперпозиции непрерывных функций одного переменного и операции сложения. В связи с этим, решение ОДУ (6.23) будем искать в виде:

$$\tilde{y} = x \cdot N(x, \vec{p}), \quad (6.24)$$

где $N(x, \vec{p})$ - значение выхода нейронной сети обратной связи с некоторым входным сигналом x и параметрами \vec{p} .

Входные сигналы x , проходя через нейроны сети и домножаясь на их весовые коэффициенты \vec{w} , в итоге формируют единый взвешенный признак $\Sigma_i = \sum_j w_{ij}^{(0)} x_j$. Взвешенный признак Σ_i далее преобразуется с помощью функции активации (некоторый нелинейный преобразователь) и формирует выход ИНС для нейрона на следующем слое $\sigma(\Sigma_i) = \sigma\left(\sum_j w_{ij}^{(0)} x_j\right)$ (рис. 6.3). На выходном

слое значения нейронов домножаются на весовые коэффициенты следующего слоя $w_i^{(1)}$, а ответом (выходом ИНС) считается полученная взвешенная сумма.

Таким образом, функция $N(x, \vec{p})$ имеет вид:

$$N(x, \vec{p}) = \sum_{i=1}^m w_i^{(1)} \sigma \left(\sum_{j=1}^n w_{ij}^{(0)} x_j \right), \quad (6.25)$$

где m – количество нейронов на входном слое, а n – количество нейронов на выходном слое.

Задача состоит в подборе оптимальных параметров ИНС таким образом, чтобы аппроксимируемая функция (6.24) с некоторой погрешностью удовлетворяла системе (6.23). Под параметрами ИНС \vec{p} понимается количество нейронов входного и выходного слоя, их весовые коэффициенты, а также используемая функция активации.

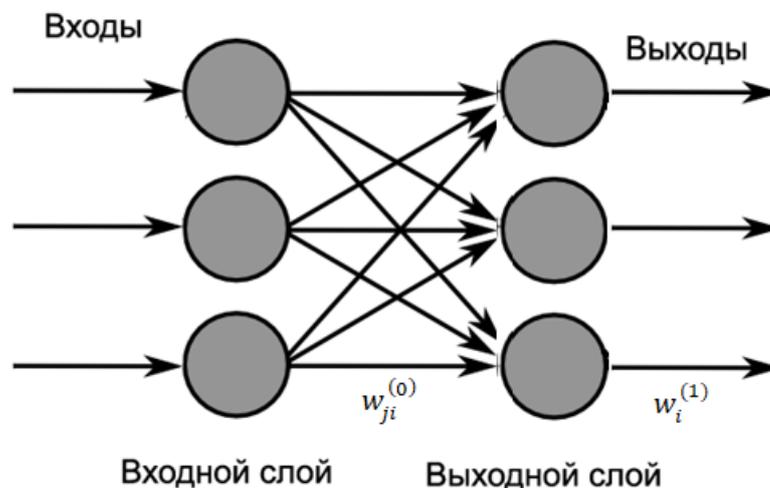


Рисунок 6.3 – Структура ИНС для решения ОДУ

Подбор оптимальных весовых коэффициентов ИНС можно осуществить с помощью алгоритма обратного распространения ошибки, для которой необходимо составить соответствующую задачу оптимизации. Критерий качества на основе метода наименьших квадратов представим в виде:

$$E[\vec{p}] = \sum_k \left\{ \frac{d\tilde{y}(x_k)}{dx} - J(x_k, \tilde{y}(x_k)) \right\}^2 \rightarrow \min, \quad (6.26)$$

где $\{x_k\}$ – заданные точки разбиения области определения.

Таким образом, обновление весовых коэффициентов ИНС проводят в направлении антиградиента функции ошибок (6.26) и алгоритм обратного распространения ошибки схематично можно представить согласно рисунку 6.4.



Рисунок 6.4 – Алгоритм оптимизации ИНС для решения ОДУ

Таким образом, представленный алгоритм нейросетевого подхода позволит получить решение задачи Коши для ОДУ 1-го порядка в виде функциональной зависимости (6.24).

6.2 Численные методы решения систем дифференциальных уравнений

Обобщим рассмотренные алгоритмы численного решения ОДУ для систем дифференциальных уравнений.

Системой дифференциальных уравнений называется система вида:

$$\left\{ \begin{array}{l} \frac{dy_1}{dx} = f_1(x, y_1, \dots, y_n) \\ \frac{dy_2}{dx} = f_2(x, y_1, \dots, y_n) \\ \dots \\ \frac{dy_n}{dx} = f_n(x, y_1, \dots, y_n) \end{array} \right. \quad (6.27)$$

или

$$\frac{dy_i}{dx} = f_i(x, y_1, \dots, y_n), \quad i = 1, \dots, n, \quad (6.28)$$

относительно независимой переменной x , искомым функций $y_i, i = 1, \dots, n$ и их производных $y'_i, i = 1, \dots, n$.

Начальные условия заданы следующим образом:

$$y_i(x_0) = y_{i0}, \quad i = 1, \dots, n. \quad (6.29)$$

Решением системы дифференциальных уравнений называются функции $y_i(x)$, при подстановке которых в систему (6.27) получаются тождества.

Рассмотрим обобщение рассмотренных методов решения ОДУ – метода Эйлера и его модификации, а также метода Рунге-Кутты для решения задачи Коши (6.28)-(6.29).

6.2.1 Метод Эйлера

Рассмотрим сетку с постоянным шагом h , для каждой точки $x_j, j = 1, \dots, m$ которой необходимо вычислить значение искомым функций.

Разностная схема Эйлера примет вид:

$$\begin{cases} y_{1j+1} = y_{1j} + h \cdot f_1(x_j, y_{1j}, \dots, y_{nj}), \\ y_{2j+1} = y_{2j} + h \cdot f_2(x_j, y_{1j}, \dots, y_{nj}), \\ \dots \\ y_{nj+1} = y_{nj} + h \cdot f_n(x_j, y_{1j}, \dots, y_{nj}). \end{cases} \quad (6.30)$$

или

$$y_{ij+1} = y_{ij} + h \cdot f_i(x_j, y_{1j}, \dots, y_{nj}), \quad i = 1, \dots, n. \quad (6.31)$$

где с.

6.2.2 Модифицированный метод Эйлера

Рассмотрим сетку с постоянным шагом h , для каждой точки с которой необходимо вычислить значение искомым функций.

Разностная схема модифицированного метода Эйлера примет вид:

$$\begin{aligned} y_{ij+1} &= y_{ij} + (k_{i1} + k_{i2}) / 2, \quad i = 1, \dots, n. \\ k_{i1} &= h \cdot f_i(x_j, y_{1j}, \dots, y_{nj}), \\ k_{i2} &= h \cdot f_i(x_j + h, y_{1j} + k_{i1}, \dots, y_{nj} + k_{i1}). \end{aligned} \quad (6.32)$$

где $y_{ij} = y_i(x_j)$.

6.2.3 Метод Рунге-Кутты 4-го порядка

По аналогии для решения задачи Коши (6.28)-(6.29) рассмотрим разностную схему метода Рунге-Кутты 4-го порядка:

$$y_{ij+1} = y_{ij} + \frac{k_{i0} + 2k_{i1} + 2k_{i2} + k_{i3}}{6}, \quad (6.33)$$

где

$$\begin{aligned}
k_{i0} &= h \cdot f_i(x_j, y_{1j}, \dots, y_{nj}), \\
k_{i1} &= h \cdot f_i(x_j + \frac{1}{2}h, y_{1j} + \frac{1}{2}k_{i0}, \dots, y_{nj} + \frac{1}{2}k_{n0}), \\
k_{i2} &= h \cdot f_i(x_j + \frac{1}{2}h, y_{1j} + \frac{1}{2}k_{i1}, \dots, y_{nj} + \frac{1}{2}k_{n1}), \\
k_{i3} &= h \cdot f_i(x_j + h, y_{1j} + k_{i1}, \dots, y_{nj} + k_{n2}).
\end{aligned}$$

Замечание. Для численного решения дифференциальных уравнений и их систем более высокого порядка необходимо путем замены переменных составить соответствующую систему ДУ 1-го порядка (с большим количеством неизвестных функций) и решить их методами, описанными в рамках этой главы.

6.2.4 Нейросетевой подход к решению систем ДУ

Рассмотрим нейросетевой подход к решению задачи Коши для систем дифференциальных уравнений 1-го порядка. Пусть задана система (6.28)-(6.29), по аналогии пунктом 6.1.3 решение системы ДУ представим в следующем виде:

$$\begin{cases}
\tilde{y}_1(x) = y_{10} + (x - x_0) \cdot N_1(x, \vec{p}_1), \\
\tilde{y}_2(x) = y_{20} + (x - x_0) \cdot N_2(x, \vec{p}_2), \\
\dots \\
\tilde{y}_n(x) = y_{n0} + (x - x_0) \cdot N_n(x, \vec{p}_n),
\end{cases} \quad (6.34)$$

где $N_1(x, \vec{p}_1), N_2(x, \vec{p}_2), \dots, N_n(x, \vec{p}_n)$ - значение выходов нейронных сетей обратной связи с некоторым входным сигналом x и параметрами \vec{p} .

Согласно МНК, при подстановке нейросетевого решения в квадрат разницы между левой и правой частью ДУ, получим следующую задачу оптимизации:

$$\left\{ \begin{array}{l} E_1[\vec{p}] = \sum_k \left\{ \frac{\tau}{dx} - J_1(x_k, \tilde{\dots}, \tilde{\dots}) \right\}^2 \min, \\ E_2[\vec{p}] = \sum_k \left\{ \frac{\tau}{dx} - J_2(x_k, \tilde{\dots}, \tilde{\dots}) \right\}^2 \min, \\ \dots \\ E_n[\vec{p}] = \sum_k \left\{ \frac{\tau}{dx} - J_n(x_k, \tilde{\dots}, \tilde{\dots}) \right\}^2 \min, \end{array} \right. \quad (6.35)$$

где $\{x_k\}$ – заданные точки разбиения области определения.

Таким образом, обновление весовых коэффициентов ИНС проводят в направлении антиградиента функций ошибок (6.35) и алгоритм обратного распространения ошибки схематично можно представить согласно рисунку 6.5.

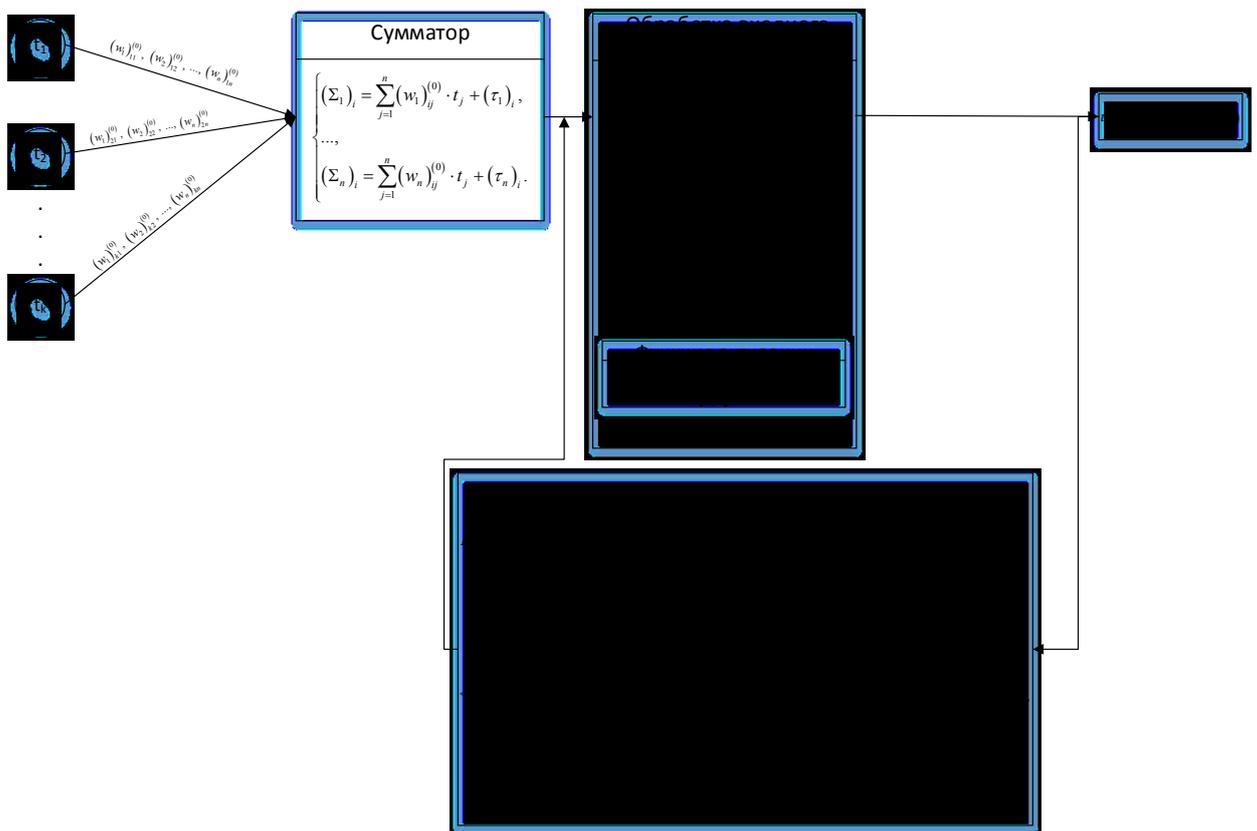


Рисунок 6.5 – Алгоритм оптимизации ИНС для решения системы ДУ

Таким образом, представленный алгоритм нейросетевого подхода позволит получить решение задачи Коши для системы ДУ 1-го порядка в виде функциональной зависимости (6.34).

7 Численные методы решения краевых задач для дифференциальных уравнений в частных производных

Наиболее сложные процессы, зависящие от большого количества факторов, как правило, могут также зависеть от частных производных различного порядка. В связи с этим, граничные условия этих процессов усложняются и, следовательно, необходимы альтернативные методы построения численных схем решения данных задач. В рамках данной главы рассмотрены основные численные методы решения дифференциальных уравнений в частных производных.

Дифференциальным уравнением в частных производных (ДУ в ЧП) называется уравнение вида:

$$F(x_1, \dots, x_n, u, \frac{\partial u}{\partial x_1}, \dots, \frac{\partial u}{\partial x_n}, \dots, \frac{\partial^k u}{\partial x_1^{k_1} \dots \partial x_n^{k_n}}) = 0, \quad (7.1)$$

где $k = \sum_{i=1}^n k_i$ – порядок старшей производной (*порядок уравнения*).

Решением ДУ в ЧП вида (7.1) называется функция $u(x_1, \dots, x_n)$, при подстановке которой уравнение превращается в тождество.

Наиболее распространенный тип уравнений, встречающийся при моделировании физических процессов – *линейное ДУ в ЧП второго порядка*:

$$A \frac{\partial^2 u}{\partial x^2} + B \frac{\partial^2 u}{\partial x \partial y} + C \frac{\partial^2 u}{\partial y^2} + D \frac{\partial u}{\partial x} + E \frac{\partial u}{\partial y} + Gu = f(x, y), \quad (7.2)$$

где $A = A(x, y)$, $B = B(x, y)$, $C = C(x, y)$, $D = D(x, y)$, $E = E(x, y)$, $G = G(x, y)$.

Наиболее распространенными видами ДУ в ЧП описывается множество уравнений математической физики. Например, ДУ в ЧП 2-го порядка применяют при математическом моделировании процессов в областях:

- механика (колебания пластин, мембран),

- физика (электромагнитные колебания, диффузия),
- гидродинамика и газодинамика (различные виды течений).

Существует несколько подходов к классификации типов ДУвЧП. В рамках данной главы рассмотрим численное решение ДУвЧП, наиболее распространенных в области моделирования физических процессов.

7.1 Методы решения задач математической физики, описываемых ДУ в частных производных

Классификацию дифференциальных уравнений в частных производных можно рассматривать исходя из их математической природы или физического смысла соответствующих задач. С математической точки зрения различают эллиптические, параболические и т.п. типы ДУ в ЧП. Если рассматривать прикладные физические задачи, то классификация будет еще более широкой – уравнения диффузии, волновое уравнение и т.п.

Таблица 7.1 – Простейшие ДУ в ЧП

Тип уравнения	Математическое представление	Примеры прикладных задач
Уравнение Лапласа	$\Delta f = 0$	Стационарные тепловые поля. Установившееся течение жидкости.
Уравнение Пуассона	$\Delta f = -k$	Теплопередача с внутренними источниками тепла.
Уравнение диффузии	$\Delta f = \frac{1}{h^2} \frac{\partial f}{\partial t}$	Нестационарная теплопроводность.
Волновое уравнение	$\Delta f = \frac{1}{C^2} \frac{\partial^2 f}{\partial t^2}$	Распространение звуковых волн.
Бигармоническое уравнение	$\Delta^2 f = F(x, y)$	Деформация пластин.

В таблице 7.1 представлены простейшие дифференциальные уравнения в частных производных с использованием следующих обозначений:

$$\Delta f = \frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2}, \quad \Delta^2 f = \frac{\partial^4 f}{\partial x^4} + \frac{\partial^4 f}{\partial x^2 \partial y^2} + \frac{\partial^4 f}{\partial y^4}.$$

На практике выделяют три основных подхода к решению задач математической физики, которые включают в себя большое разнообразие математических подходов.

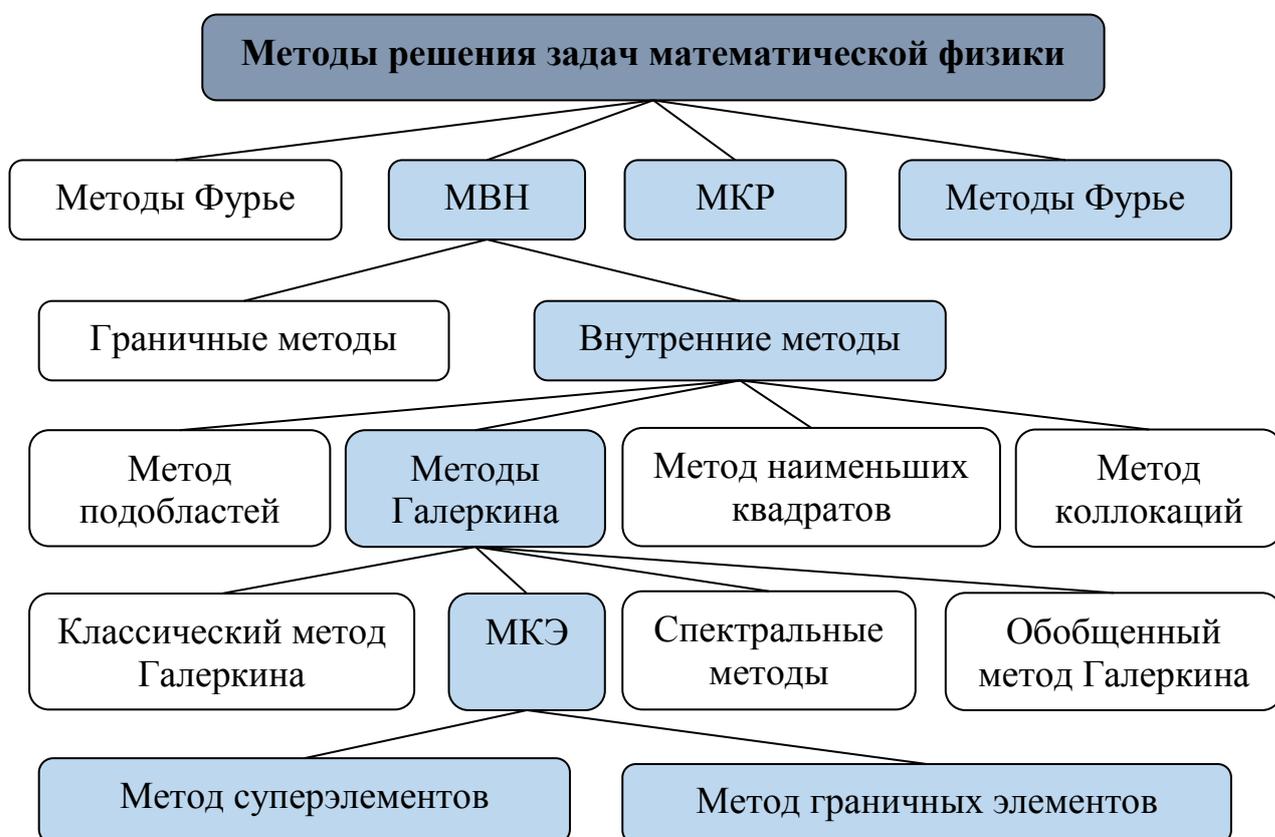


Рисунок 7.1 – Классификация методов решения задач математической физики

Отметим, что помимо известных *методов взвешенных невязок* (МВН) большую актуальность получили различные сеточные или *конечно-разностные методы* (МКР), а также *вероятностные методы*.

Развитие возможностей вычислительной техники позволило получить всевозможные модификации различных численных алгоритмов и повысить точность результатов, скорость работы и расширить области их применения.

7.2 Метод конечных разностей для ДУ в частных производных

Рассмотрим идею конечно-разностной аппроксимации производных для численного решения дифференциальных уравнений в частных производных. Непрерывную область определения независимых переменных x, y ДУ в ЧП вида:

$$f(x, y, U, U_x, U_y, U_{xx}, U_{yy}, U_{xy}, \dots) = 0 \quad (7.3)$$

представим дискретной сеткой точек (узлов) с постоянным достаточно малым шагом h .

Заменим в узлах сетки производные в уравнении (7.3) конечно-разностными соотношениями и получим алгебраическое (трансцендентное) уравнение вида

$$f_h(x, y, U, U_x, U_y, U_{xx}, U_{yy}, U_{xy}, \dots) = 0, \quad (7.4)$$

которое называется *сеточным* или *разностным* уравнением.

При неограниченном дроблении сетки $h \rightarrow 0$ для любой достаточно гладкой функции выполняется следующее свойство:

$$f_h(U) \rightarrow f(U). \quad (7.5)$$

Локальной погрешностью (или *погрешностью аппроксимации*) называется абсолютное отклонение сеточного уравнения от непрерывного ДУ в ЧП:

$$|f_h(U) - f(U)|. \quad (7.6)$$

Значение данной погрешности можно вычислить с помощью разложения в окрестности некоторой точки (узла сетки) достаточно гладкой функции в ряд Тейлора.

Таким образом, *построение численного решения ДУ в ЧП* можно описать 3 этапами:

1. Определение в области решения *равномерной сетки «узловых» точек*, которые позволяют задать граничные условия и удовлетворяют характеру задачи.

2. Используя координатное представление ДУ в ЧП и конечно-разностное представление производных, формируется *разностное уравнение*. Данная форма позволяет оценить взаимосвязи между узлами сетки и использовать их для описания.

3. Для каждого узла сетки записываются *разностные уравнения*, формирующие алгебраическую систему уравнений, которую можно решить любым из рассмотренных выше методов.

Сложность осуществления данных этапов численного решения обуславливается многообразием типов и размеров сеток, видов уравнений, методов конечно-разностной аппроксимации, а также алгоритмов решения полученных систем уравнений. Наиболее популярными типами сеток на данный момент являются прямоугольные, полярные, треугольные и скошенные. В связи с этим, построение соответствующих разностных уравнений является обширной исследовательской задачей.

7.3 Начальные и граничные условия

Начальные условия определяют значения искомой функции (и ее производных до $(k-1)$ -го порядка) в начальный момент времени. Например, для

волнового уравнения (см. таблицу 7.1) в начальный момент может быть зафиксировано положение струны, а также задана скорость движения ее точек.

Граничные условия определяют значения искомой функции (и ее производных до $(k-1)$ -го порядка) на границе области определения в пространстве. Например, для волнового уравнения может быть зафиксировано положение струны на границе (граничных точках).

Типы граничных условий:

1. заданы значения искомой функции на границе (*задача Дирихле*);
2. заданы значения производной искомой функции по нормали к границе (*задача Неймана*);
3. задана линейная комбинация значений функции и ее производной по нормали к границе (*задача Робена*).

Рассмотрим различные виды ДУ в ЧП и применим к ним метод конечных разностей, описанный выше.

7.4 Уравнение колебаний

Рассмотрим малые колебания однородной струны, описываемые ДУ в ЧП вида:

$$\frac{\partial^2 u}{\partial t^2} = \alpha^2 \frac{\partial^2 u}{\partial x^2} + f(x, t), \quad 0 < x < 1, \quad 0 < t \leq T, \quad (7.7)$$

С граничными условиями вида:

$$\begin{aligned} u(x, 0) = u_0(x), \quad u'_t(x, 0) = \bar{u}_0(x), \quad 0 < x < 1, \\ u(0, t) = \mu_1(t), \quad u(1, t) = \mu_2(t), \quad 0 < t \leq T. \end{aligned} \quad (7.8)$$

Определим на непрерывной области дискретную равномерную сетку «узловых» точек вида:

$$\omega_{h\tau} = \{(x_i, t_j) : x_i = ih, 0 \leq i \leq n, h = 1/n, t_j = j\tau, 0 \leq j \leq m, \tau = T/m\}. \quad (7.9)$$

Для построения разностной схемы воспользуемся следующей аппроксимацией производных функции второго порядка:

$$\left(\frac{\partial^2 u}{\partial t^2}\right)_{ij} \approx \frac{u_i^{j+1} - 2u_i^j + u_i^{j-1}}{\tau^2}, \quad \left(\frac{\partial^2 u}{\partial x^2}\right)_{ij} \approx \frac{u_{i+1}^j - 2u_i^j + u_{i-1}^j}{h^2}. \quad (7.10)$$

Разностные уравнения для ДУ в ЧП (7.7) примут вид:

$$\frac{v_i^{j+1} - 2v_i^j + v_i^{j-1}}{\tau^2} = \alpha^2 \frac{v_{i+1}^j - 2v_i^j + v_{i-1}^j}{h^2} + f_i^j. \quad (7.10)$$

Граничные условия в дискретном виде:

$$v_0^j = \mu_1(t_j), \quad v_N^j = \mu_2(t_j), \quad 0 \leq j \leq L. \quad (7.11)$$

Начальные условия в дискретном виде:

$$v_i^0 = u_0(x_i), \quad 1 \leq i \leq n-1. \quad (7.12)$$

Аппроксимацию для данной задачи можно осуществлять как первого порядка:

$$(v_i^1 - v_i^0) / \tau = \bar{u}_0(x_i), \quad 1 \leq i \leq n-1, \quad (7.13)$$

так и второго порядка:

$$\frac{v_i^1 - v_i^0}{\tau} = \bar{u}_0(x_i) + \frac{\tau}{2} \left(\alpha^2 \frac{v_{i+1}^0 - 2v_i^0 + v_{i-1}^0}{h^2} + f_i^0 \right), \quad 1 \leq i \leq n-1. \quad (7.14)$$

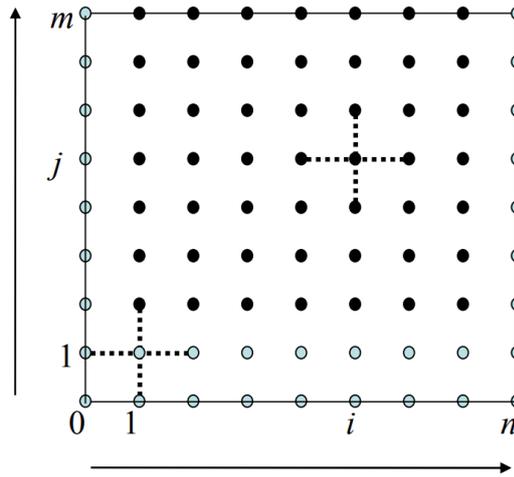


Рисунок 7.1 – Геометрическое представление явной схемы

Численное решение строится по слоям в явной форме, поэтому данная схема называется явной:

- 0-й слой: $v_i^0 = u_0(x_i)$;
- 1-й слой: $v_i^1 = v_i^0 + \tau \bar{u}_0(x_i) + \frac{\tau^2}{2} \left(\frac{v_{i+1}^0 - 2v_i^0 + v_{i-1}^0}{h^2} + f_i^0 \right)$;
- j-й слой: $v_i^{j+1} = 2v_i^j - v_i^{j-1} + \left(\frac{\alpha\tau}{2} \right)^2 \left(v_{i+1}^j - 2v_i^j + v_{i-1}^j \right) + \tau^2 f_i^j$.

Тогда погрешность численного решения будет иметь вид:

$$\max_{i,j} |u_i^j - v_i^j| \leq M(h^2 + \tau^2). \quad (7.15)$$

Для обеспечения устойчивости численного решения необходимо выполнить **условие Куранта**:

$$\alpha\tau < h. \quad (7.16)$$

7.5 Уравнение теплопроводности

Рассмотрим распространение тепла в одномерном стержне, описываемое ДУ в ЧП вида:

$$\frac{\partial u}{\partial t} = \frac{\partial^2 u}{\partial x^2} + f(x, t), \quad 0 < x < 1, \quad 0 < t \leq T, \quad (7.17)$$

С граничными условиями:

$$\begin{aligned} u(x, 0) &= u_0(x), \quad 0 < x < 1, \\ u(0, t) &= u_1(t), \quad u(1, t) = u_2(t), \quad 0 < t \leq T. \end{aligned} \quad (7.18)$$

Определим на непрерывной области дискретную равномерную сетку D :

$$\omega_{h\tau} = \{(x_i, t_j) : x_i = ih, \quad 0 \leq i \leq n, \quad h = 1/n, \quad t_j = j\tau, \quad 0 \leq j \leq m, \quad \tau = T/m\}. \quad (7.19)$$

Вспользуемся аппроксимацией производных функции

- 1-го порядка по τ : $\left(\frac{\partial u}{\partial t}\right)_{ij} \approx \frac{u_i^{j+1} - u_i^j}{\tau}$;
- 2-го порядка по h : $\left(\frac{\partial^2 u}{\partial x^2}\right)_{ij} \approx \frac{u_{i+1}^j - 2u_i^j + u_{i-1}^j}{h^2}$.

Разностные уравнения для ДУ в ЧП (7.17) для **явной схемы**:

$$\frac{v_i^{j+1} - v_i^j}{\tau} = \frac{v_{i+1}^j - 2v_i^j + v_{i-1}^j}{h^2} + f_i^j. \quad (7.20)$$

Граничные условия в дискретном виде:

$$\begin{aligned} v_i^0 &= u_0(x_i), \quad 0 \leq i \leq n, \\ v_0^j &= u_1(t_j), \quad v_n^j = u_2(t_j), \quad 0 \leq j \leq m. \end{aligned} \quad (7.21)$$

Тогда, численное решение строится по явной схеме следующим образом:

- 0-й слой: $v_i^0 = u_0(x_i)$;
- j-й слой: $v_i^{j+1} = \left(1 - \frac{2\tau}{h^2}\right)v_i^j + \frac{\tau}{h^2}(v_{i+1}^j + v_{i-1}^j) + \tau f_i^j$.

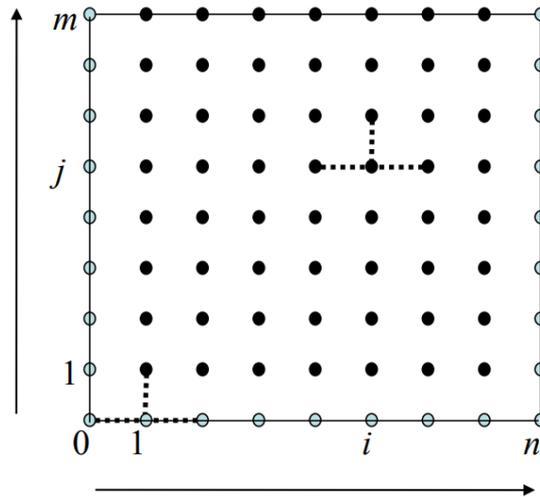


Рисунок 7.2 – Геометрическое представление явной схемы

Погрешность численного решения будет иметь вид:

$$\max_{i,j} |u_i^j - v_i^j| \leq M(h^2 + \tau). \quad (7.22)$$

Для обеспечения *устойчивости* численного решения необходимо выполнить условие:

$$\tau \leq \frac{h^2}{2}. \quad (7.23)$$

Рассмотрим разностные уравнения для ДУ в ЧП (7.17) для *неявной* *схемы*:

$$\frac{v_i^{j+1} - v_i^j}{\tau} = \frac{v_{i+1}^{j+1} - 2v_i^{j+1} + v_{i-1}^{j+1}}{h^2} + f_i^{j+1}. \quad (7.24)$$

Дискретные граничные условия имеют вид:

$$\begin{aligned} v_i^0 &= u_0(x_i), \quad 0 \leq i \leq n, \\ v_0^j &= u_1(t_j), \quad v_n^j = u_2(t_j), \quad 0 \leq j \leq m. \end{aligned} \quad (7.25)$$

Тогда, численное решение строится по неявной схеме следующим

образом:

– 0-й слой: $v_i^0 = u_0(x_i)$;

$$v_0^{j+1} = u_1(t_{j+1}), \quad v_n^{j+1} = u_2(t_{j+1}),$$

– (j+1)-й слой: $\frac{\tau}{h^2} v_{i-1}^{j+1} - \left(1 + \frac{2\tau}{h^2}\right) v_i^{j+1} + \frac{\tau}{h^2} v_{i+1}^{j+1} = v_i^j + \tau f_i^{j+1}$.

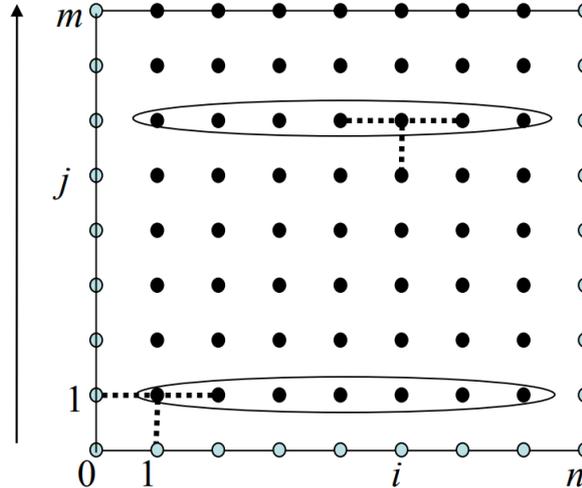


Рисунок 7.3 – Геометрическое представление неявной схемы

Погрешность численного решения будет иметь вид:

$$\max_{i,j} |u_i^j - v_i^j| \leq M(h^2 + \tau). \quad (7.26)$$

Данная неявная схема *абсолютно устойчива* при любых τ и h .

7.6 Уравнение Пуассона

Рассмотрим изгиб упругой пластины, описываемый уравнением Пуассона вида:

$$\Delta u(x, y) = -f(x, t), \quad (x, y) \in D \quad (7.27)$$

С граничными условиями:

$$u(x, y) = \psi(x, y), \quad (x, y) \in \Gamma(D). \quad (7.28)$$

Определим на непрерывной области дискретную равномерную сетку D :

$$D_{h\tau} = \{(x_i, y_j) : x_i = ih, y_j = jk, 0 \leq i \leq n, 0 \leq j \leq m, h = 1/n, k = 1/m\}. \quad (7.29)$$

Воспользуемся аппроксимацией производных функции

- 2-го порядка по x : $\left(\frac{\partial^2 u}{\partial x^2}\right)_{ij} \approx \frac{u_{i-1}^j - 2u_i^j + u_{i+1}^j}{h^2}$;
- 2-го порядка по y : $\left(\frac{\partial^2 u}{\partial y^2}\right)_{ij} \approx \frac{u_i^{j-1} - 2u_i^j + u_i^{j+1}}{k^2}$.

Разностные уравнения для ДУ в ЧП (7.27) имеют вид:

$$\begin{cases} \frac{v_{i-1}^j - 2v_i^j + v_{i+1}^j}{h^2} + \frac{v_i^{j-1} - 2v_i^j + v_i^{j+1}}{k^2} = -f_i^j, \\ v_i^j = \psi(x_i, y_j), \quad (x_i, y_j) \in \Gamma(D). \end{cases} \quad (7.29)$$

Таким образом, разностная схема (7.29) представляет собой систему линейных уравнений относительно v_i^j , $i = 1, n, j = 1, m$.

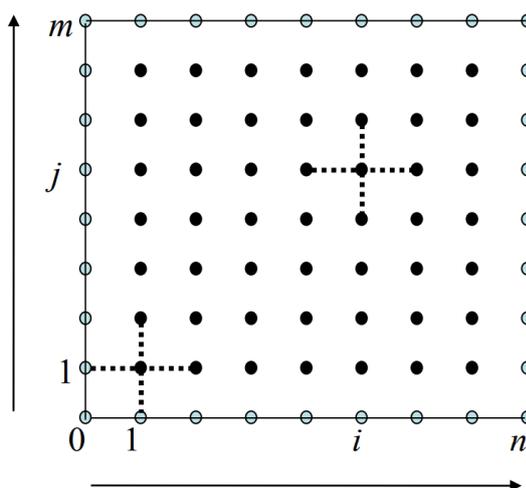


Рисунок 7.4 – Геометрическое представление разностной схемы

Погрешность численного решения будет иметь вид:

$$\max_{i,j} |u_i^j - v_i^j| \leq M(h^2 + k^2). \quad (7.30)$$

Таким образом, в рамках данной главы рассмотрен метод конечных разностей для дифференциальных уравнений в частных производных, которыми могут быть представлены различные процессы математической физики. Однако существует большое множество методов решения, не охваченных в данном учебном пособии и требующих дополнительных исследований.

8 Применение методов вычислительной математики в различных научных областях

Методы вычислительной математики могут быть применены в различных научных областях для решения прикладных задач в экономике, биологии, физике и других. В рамках данной главы рассмотрим частные примеры использования модели отсека для моделирования эпидемий с различными ограничениями, а также моделирование распространения атак в сетях.

8.1 Моделирование эпидемий

Наша цель в этом разделе - показать подробно, как можно применить методы вычислительной математики и программирования для исследования распространения болезней. Математическая модель представляет собой систему трех дифференциальных уравнений с тремя неизвестными функциями. Для построения модели использованы основные принципы взаимодействия незараженных, восприимчивых и болеющих людей без специальных знаний о болезнях.

8.1.1 Распространение гриппа

Представим небольшую школу-интернат, которая представляет собой маленькое и закрытое общество. Внезапно один или несколько учеников заболевают гриппом. Можно ожидать, что грипп может довольно эффективно распространиться или исчезнуть. Вопрос в том, сколько учеников и сотрудников школы пострадают. Некоторые довольно простые математические вычисления могут помочь нам понять динамику распространения болезни.

Пусть математическая функция $S(t)$ показывает, сколько людей в момент времени t имеют возможность заразиться. Здесь t может, например, считать часы или дни. Эти люди составляют категорию под названием «восприимчивые», обозначенную как S . Другая категория - I , состоит из инфицированных людей. Пусть функция $I(t)$ показывает, сколько людей

находится в категории I в момент времени t . Считается, что человек, выздоровевший от болезни, обретает иммунитет. Также существует небольшая вероятность того, что зараженный умрет.

Индивидуум перемещается из категории I в категорию, которую называется «невосприимчивые» и обозначается R . Функция $R(t)$ показывает количество людей в категории R в момент времени t . Те, кто попадает в категорию R , не могут покинуть эту категорию.

Подводя итог, можно сказать, что распространение этой болезни - это, по сути, динамика перехода людей из категории S в категорию I , а затем в категорию R (рис. 8.1).



Рисунок 8.1 – Схема распространения гриппа

Построим математическую модель, описывающую обмен между категориями. Основная идея состоит в том, чтобы описать изменения, происходящие в течение небольшого промежутка времени, обозначаемого Δt .

Данную модель болезни часто называют *модель отсека*, где величины перетасовываются между отсеками (категориями) в соответствии с некоторыми правилами. Правила выражают *изменения* в малом временном интервале Δt и из этих изменений при $\Delta t \rightarrow 0$ можно выразить производные. Полученные уравнения затем переходят от разностных уравнений (с конечным Δt) к дифференциальным уравнениям ($\Delta t \rightarrow 0$).

Введем равномерную сетку по времени $t_n = n \cdot \Delta t$, $n = 0, \dots, N_t$ и найдем значения S в точках сетки. Численное приближение к S в момент времени t_n обозначается S_n . Аналогичным образом мы ищем неизвестные значения $I(t)$ и

$R(t)$ в узлах сетки и вводим аналогичные обозначения I_n и R_n для приближений к точным значениям $I(t_n)$ и $R(t_n)$.

В промежутке времени Δt некоторые люди будут инфицированы, поэтому S будет уменьшаться. Кроме того, в этом временном интервале появятся новые инфицированные $\beta\Delta tSI$, где β - параметр, отражающий насколько легко люди заразятся в течение временного интервала единичной длины. Если потери в S равны $\beta\Delta tSI$, то изменение S представимо в виде:

$$S^{n+1} - S^n = -\beta\Delta tS^n I^n. \quad (8.1)$$

Разделив на Δt и положив $\Delta t \rightarrow 0$, получим левосторонний подход к $S'(t_n)$ и следующее дифференциальное уравнение:

$$S'(t) = -\beta S(t)I(t). \quad (8.2)$$

Разберем более подробно слагаемое $\beta\Delta tSI$: пусть имеется S восприимчивых и I зараженных людей, которые могут встретиться и составить пары $S \cdot I$. Предположим, что в течение интервала времени T действительно происходят m фактических попарных встреч среди n теоретически возможных пар людей из категорий S и I . Вероятность того, что люди встретятся парами в течение времени T , равна (согласно эмпирическому определению вероятности частоты) m/n . Выразим вероятность μ в единицу времени: $\mu = m/(nT)$.

Учитывая вероятность μ , ожидаемое количество встреч за временной интервал возможных пар людей $S \cdot I$ составляет μSI . В течение временного интервала Δt будет ожидаемое число встреч между восприимчивыми и инфицированными людьми в $\mu\Delta tSI$, так что вирус может распространиться. Лишь часть встреч $\mu\Delta tSI$ эффективны в том смысле, что восприимчивый действительно заражается. Подсчитав, что m человек заразятся за n таких попарных встреч, то вероятность заражения можно представить как $p = m/n$. Ожидаемое количество людей в категории S , которые за промежуток времени

Δt заразятся вирусом $p\mu\Delta tSI$. Вводя новую константу $\beta = p\mu$, приходим к слагаемому $\beta\Delta tSI$.

Значение β должно быть известно, чтобы предсказывать будущее с помощью модели болезни. Одна из возможностей - оценить p и μ , исходя из их значений в приведенном выше выводе. В качестве альтернативы, мы можем наблюдать «эксперимент», в котором в какой-то момент времени есть S_0 - восприимчивые и I_0 - инфицированные. В течение промежутка времени T заразилось N восприимчивых. Используя (8.1) в качестве грубого приближения того, как S развивалось за время T , получим:

$$N = \beta TS_0 I_0 \Rightarrow \beta = \frac{N}{TS_0 I_0}. \quad (8.3)$$

Заметим, что «потеря» в категории S является соответствующим «выигрышем» в категории I . Таким образом,

$$I^{n+1} - I^n = \beta\Delta t S^n I^n. \quad (8.4)$$

При этом «потери» в категории I возникают при выздоровлении людей. Предположим, что m из n человек выздоравливают за период времени T . Тогда $\gamma = m/(nT)$ - вероятность того, что один человек выздоровеет за единственный интервал времени. Тогда (в среднем) инфицированные $\gamma\Delta t I$ выздоравливают за интервал времени Δt . Данная величина представляет собой «убыток» по категории I и «выигрыш» по категории R . Таким образом, можно записать общее изменение в категории I как

$$I^{n+1} - I^n = \beta\Delta t S^n I^n - \gamma\Delta t I^n. \quad (8.5)$$

Изменения в категории R - переходящие из категории I :

$$R^{n+1} - R^n = \gamma \Delta t I^n. \quad (8.6)$$

В категории R нет потерь - люди либо выздоравливают и имеют иммунитет, либо мертвы.

Разделив на Δt в (8.5) и (8.6) и положив $\Delta t \rightarrow 0$, получим соответствующие дифференциальные уравнения

$$I'(t) = \beta S(t)I(t) - \gamma I(t), \quad (8.7)$$

$$R'(t) = \gamma I(t). \quad (8.8)$$

Таким образом, получим следующую систему разностных уравнений:

$$\begin{cases} S^{n+1} = S^n - \beta \Delta t S^n I^n \\ I^{n+1} = I^n + \beta \Delta t S^n I^n - \gamma \Delta t I^n \\ R^{n+1} = R^n + \gamma \Delta t I^n \end{cases} \quad (8.9)$$

Соответствующая система дифференциальных уравнений имеет вид:

$$\begin{cases} S'(t) = -\beta S(t)I(t). \\ I'(t) = \beta S(t)I(t) - \gamma I(t), \\ R'(t) = \gamma I(t). \end{cases} \quad (8.10)$$

Данная модель (8.10) известна как *модель SIR*. Входные данные для модели - параметры β и γ , а также начальные условия $S(0) = S_0, I(0) = I_0$ и $R(0) = R_0$.

8.1.1.1 Метод Эйлера для системы дифференциальных уравнений

Применим методом Эйлера для численного решения системы (8.10) на сетке по времени с дискретными величинами $S_n, I_n, R_n, n=0, \dots, N_t$. Тогда в точке t_n имеем:

$$\begin{cases} S'(t_n) = -\beta S(t_n)I(t_n), \\ I'(t_n) = \beta S(t_n)I(t_n) - \gamma I(t_n), \\ R'(t_n) = \gamma I(t_n). \end{cases} \quad (8.11)$$

для $n=0, \dots, N_t$. Использование прямых конечных разностей для производных приводит к следующему приближению:

$$\begin{cases} \frac{S^{n+1} - S^n}{\Delta t} = -\beta S^n I^n, \\ \frac{I^{n+1} - I^n}{\Delta t} = \beta S^n I^n - \gamma I^n, \\ \frac{R^{n+1} - R^n}{\Delta t} = \gamma I^n. \end{cases} \quad (8.12)$$

Ясно, что использование других численных методов, кроме прямой схемы Эйлера, приведет к несколько другим разностным уравнениям.

8.1.1.2 Программирование численной схемы: частный случай

Реализация численной схемы (8.12) в следующей программе [SIR1.m](#):

```
% Time unit: 1 h
beta = 10/(40*8*24);
gamma = 3/(15*24);
dt = 0.1; % 6 min
D = 30; % Simulate for D days
N_t = floor(D*24/dt); % Corresponding no of hours

t = linspace(0, N_t*dt, N_t+1);
S = zeros(N_t+1, 1);
I = zeros(N_t+1, 1);
R = zeros(N_t+1, 1);

% Initial condition
S(1) = 50;
I(1) = 1;
R(1) = 0;

% Step equations forward in time
for n = 1:N_t
    S(n+1) = S(n) - dt*beta*S(n)*I(n);
    I(n+1) = I(n) + dt*beta*S(n)*I(n) - dt*gamma*I(n);
    R(n+1) = R(n) + dt*gamma*I(n);
end
```

```

plot(t, S, t, I, t, R);
legend('S', 'I', 'R', 'Location', 'northwest');
xlabel('hours');
print('tmp', '-dpdf'); print('tmp', '-dpng');

```

Данная программа реализована для исследования распространения гриппа в частной школе-интернате, и аргументы в пользу конкретных вариантов β и γ заключаются в следующем. В некоторой школе, где болезнь уже распространилась, было замечено, что в начале дня было 40 восприимчивых и 8 инфицированных, а через 24 часа их было 30 и 18 соответственно. Используя 1 час в качестве единицы времени и получаем $\beta = 10 / (40 \cdot 8 \cdot 24)$. Было замечено, что из 15 инфицированных 3 выздоровели в течение дня, что дало $\gamma = 3 / (15 \cdot 24)$. Применение этих параметров к новому случаю, когда один инфицированный изначально и 50 восприимчивых, дает графики на рисунке 8.2.

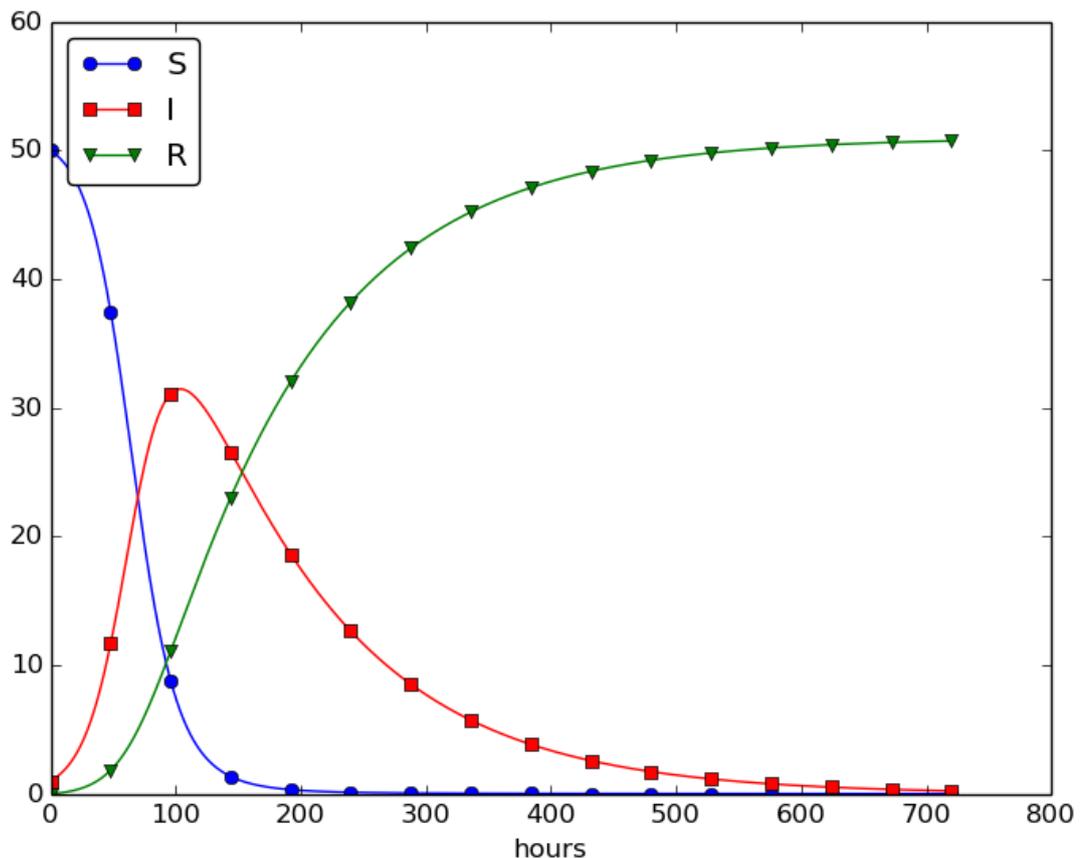


Рисунок 8.2 – Естественная эволюция гриппа в школе-интернате

Заметим, что S уменьшается с ростом I и R . Примерно через 30 дней все заболели и снова выздоровели.

Проведем эксперимент с β и γ , чтобы увидеть, получим ли вспышку болезни или нет. Представьте, что кампания «Мойте руки» была успешной и в другой школе в этом случае наблюдалось снижение β в 5 раз. При этом более низком β болезнь распространяется очень медленно, поэтому с моделируем процесс в течение 60 дней (рис. 8.3).

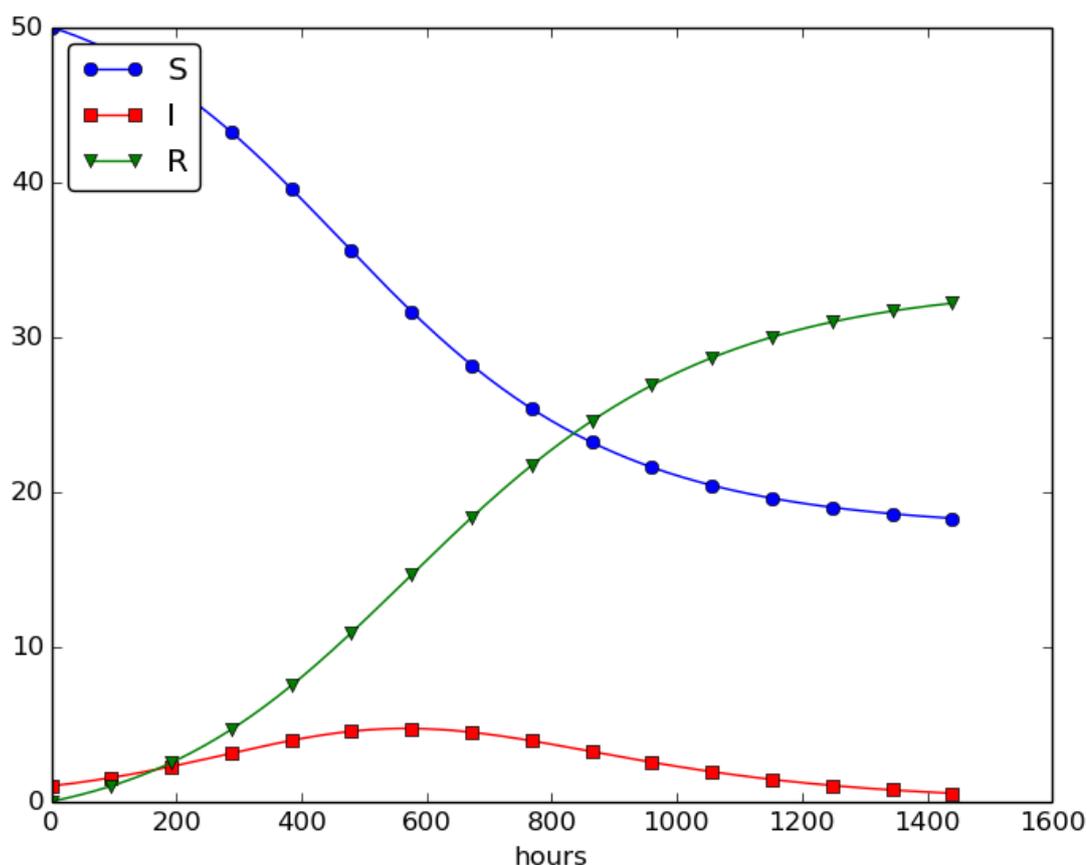


Рисунок 8.3 – Небольшая вспышка гриппа в школе-интернате

8.1.2 Численное моделирование вспышки

Исходя из дифференциального уравнения для I , заметим, что для увеличения I необходимо, чтобы $\beta SI - \gamma I > 0$. В начале моделирования, данное условие означает, что $\beta S(0)I(0) - \gamma I(0) > 0$ или

$$\frac{\beta S(0)}{\gamma} > 1 \quad (8.13)$$

необходимо для увеличения числа инфицированных и ускорения распространения болезни. Если запустить программу [SIR1.m](#) с меньшим β таким образом, чтобы было нарушено (8.13), то можно заметить, что вспышки не будет.

8.1.3 Общая постановка задачи решения систем ДУ для SIR

Запишем систему дифференциальных уравнений модели SIR в общем виде векторной системы ОДУ:

$$u' = f(u, t), \quad (8.14)$$

где u и f - вектора функций ($u = (S(t), I(t), R(t))$, $f(u, t) = (-\beta SI, \beta SI - \gamma I, \gamma I)$).

Поскольку $u' = (S'(t), I'(t), R'(t))$, получаем, что $u' = f(u, t)$ подразумевает

$$\begin{cases} S' = -\beta SI, \\ I' = \beta SI - \gamma I, \\ R' = \gamma I. \end{cases}$$

Обобщенное краткое обозначение $u' = f(u, t)$ удобно, поскольку дает возможность вывести численные схемы и реализовать ПО для этой абстрактной системы, а в конкретном приложении просто идентифицировать формулы в векторе f , реализовать их и вызвать функции, которые решают систему ДУ.

8.1.3.1 Программирование численной схемы: общий случай

В системе Matlab шаг Эйлера вперед имеет вид:

$$u^{n+1} = u^n + \Delta t \cdot f(u, t), \quad (8.15)$$

будучи скалярным или векторным уравнением, может быть закодирован как:

```
u(n+1, :) = u(n, :) + dt*f(u(n, :), t(n))
```

как в скалярном, так и в векторном случае.

Функция для векторного ODE файла `ode_system_FE.m` записана

следующим образом:

```
function [u, t] = ode_FE(f, U_0, dt, T)
    N_t = floor(T/dt);
    u = zeros(N_t+1, length(U_0));
    t = linspace(0, N_t*dt, length(u));
    u(1,:) = U_0;      % Initial values
    t(1) = 0;
    for n = 1:N_t
        u(n+1,:) = u(n,:) + dt*f(u(n,:), t(n));
    end
end
```

Давайте покажем, как предыдущая модель SIR может быть решена с помощью общего метода `ode_FE`, который может решить любое векторное ОДУ в файле `demo_SIR.m`:

```
function demo_SIR()
    % Test case using an SIR model

    dt = 0.1;          % 6 min
    D = 30;            % Simulate for D days
    N_t = floor(D*24/dt); % Corresponding no of hours
    T = dt*N_t;       % End time
    U_0 = [50 1 0];

    f_handle = @f;

    [u, t] = ode_FE(f_handle, U_0, dt, T);

    S = u(:,1);
    I = u(:,2);
    R = u(:,3);
    plot(t, S, 'b-', t, I, 'r-', t, R, 'g-');
    legend('S', 'I', 'R');
    xlabel('hours');
    % Consistency check:
    N = S(1) + I(1) + R(1);
    eps = 1E-12; % Tolerance for comparing real numbers
    for n = 1:length(S)
        err = abs(S(n) + I(n) + R(n) - N);
        if (err > eps)
            error('demo_SIR: error=%g', err);
        end
    end
end

function result = f(u,t)
    beta = 10/(40*8*24);
    gamma = 3/(15*24);

    S = u(1); I = u(2); R = u(3);
    result = [-beta*S*I beta*S*I - gamma*I gamma*I];
end
```

8.1.4 Моделирование иммунитета с ограничением по времени

Предположим теперь, что иммунитет после болезни сохраняется только в течение определенного периода времени. Это означает, что существует переход из состояния R в состояние S (рис. 8.4).



Рисунок 8.4 – Схема распространения гриппа при иммунитете с ограничением по времени

Моделирование потери иммунитета очень похоже на моделирование выздоровления от болезни: количество людей, теряющих иммунитет, пропорционально количеству выздоровевших пациентов и продолжительности временного интервала Δt . Следовательно, можно записать потерю в категории R как $-v\Delta tR$ за время Δt , где v^{-1} - типичное время, необходимое для потери иммунитета. Потеря $R(t)$ - это выигрыш в $S(t)$. Таким образом, численная схема данной модели примет вид:

$$\begin{cases} S^{n+1} = S^n - \beta\Delta t S^n I^n + v\Delta t R^n, \\ I^{n+1} = I^n + \beta\Delta t S^n I^n - \gamma\Delta t I^n, \\ R^{n+1} = R^n + \gamma\Delta t I^n - v\Delta t R^n. \end{cases} \quad (8.16)$$

Разделив на Δt и положив $\Delta t \rightarrow 0$, получим систему ДУ:

$$\begin{cases} S' = -\beta SI + vR, \\ I' = \beta SI - \gamma I, \\ R' = \gamma I - vR. \end{cases} \quad (8.17)$$

Система (8.17) может быть решена теми же методами, которые продемонстрирована для исходной модели SIR. В этом случае, в программу необходимо внести изменение:

```

for n = 1:N_t
    S(n+1) = S(n) - dt*beta*S(n)*I(n) + dt*nu*R(n)
    I(n+1) = I(n) + dt*beta*S(n)*I(n) - dt*gamma*I(n)
    R(n+1) = R(n) + dt*gamma*I(n) - dt*nu*R(n)
end

```

Установка ν^{-1} на 50 дней, уменьшение β в 4 раза по сравнению с предыдущим примером ($\beta = 0,00033$) и моделирование в течение 300 дней дает колебательное поведение в категориях, как показано на рисунке 8.5, включая потерю иммунитета.

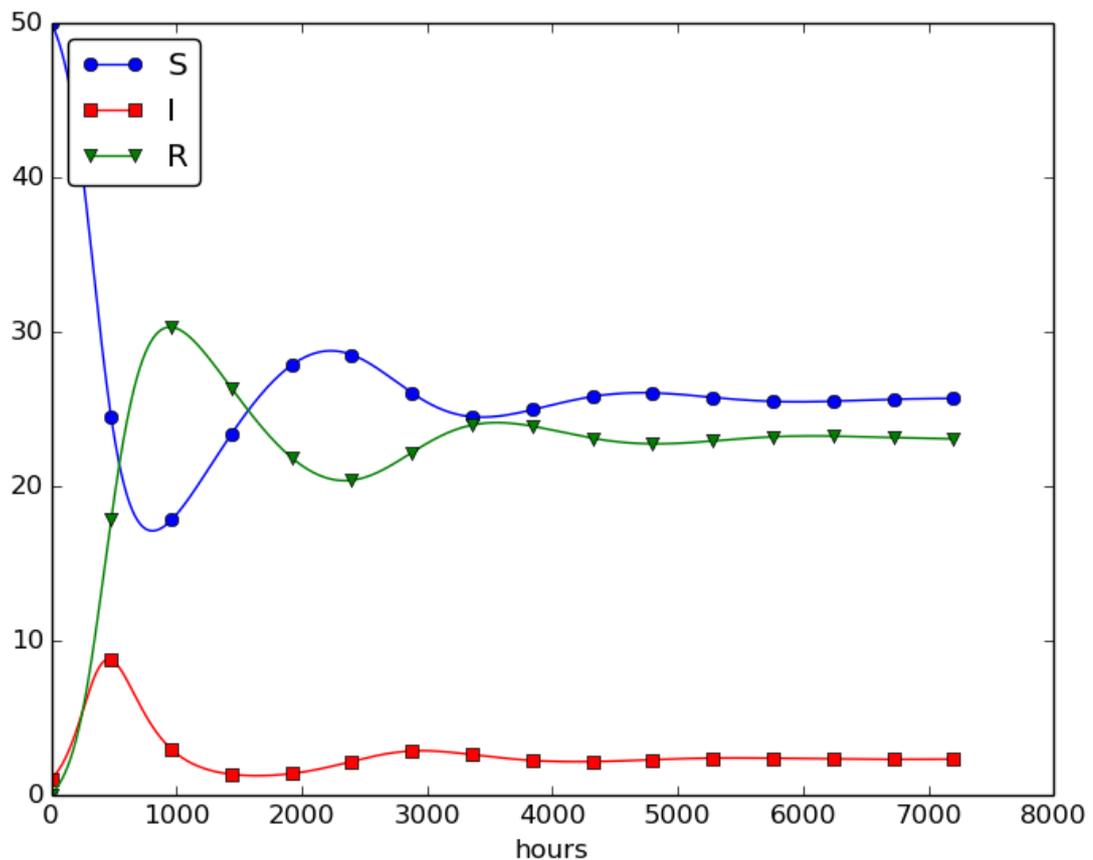


Рисунок 8.5 – Моделирование болезни с потерей иммунитета

Исследуем влияние параметров на распространение болезни. Например, если сделать болезнь немного более эффективной (увеличить β до 0,00043) и

увеличить среднее время потери иммунитета до 90 дней, это приведет к другим колебаниям на рисунке 8.6.

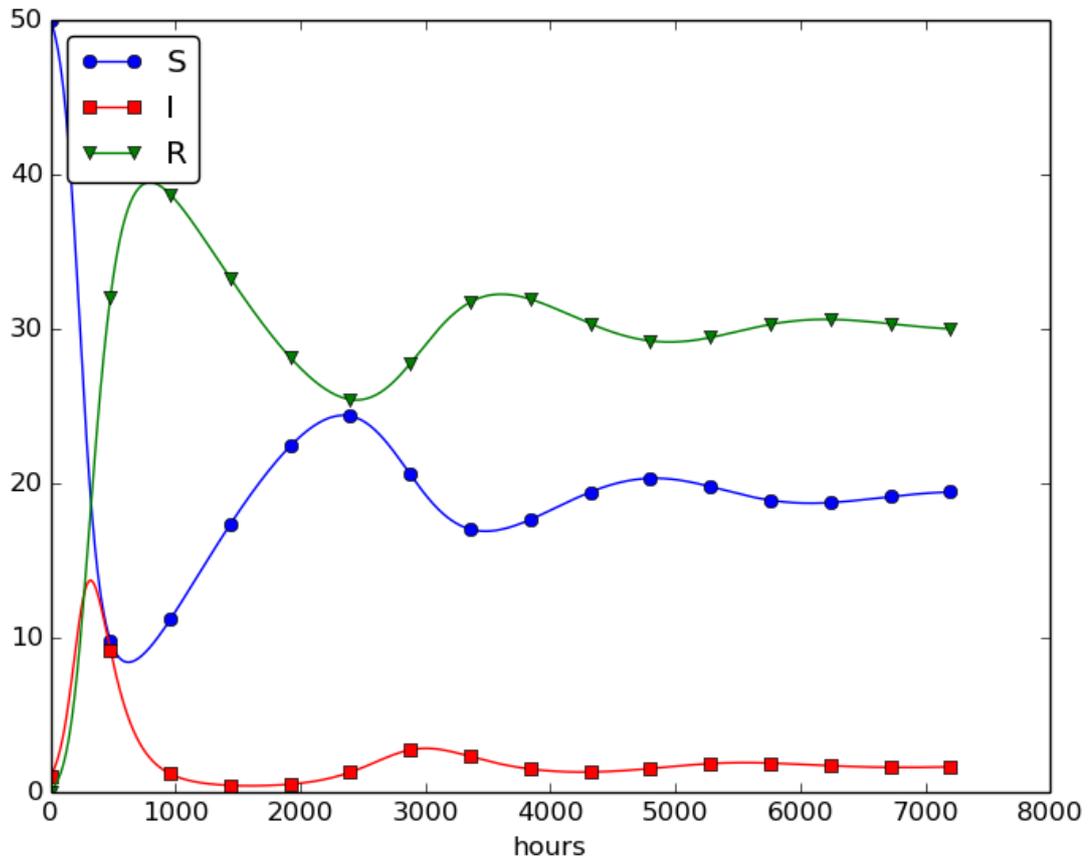


Рисунок 8.6 – Моделирование болезни с измененными параметрами

8.1.5 Моделирование вакцинации населения

Расширим модель, включив в нее вакцинацию населения. С этой целью может быть полезно отслеживать тех, кто вакцинирован, и тех, кто не вакцинирован. Таким образом, введем четвертую категорию V , для тех, кто прошел успешную вакцинацию. Кроме того, предположим, что в интервале времени Δt часть $p\Delta t S$ категории S подлежит успешной вакцинации. Это означает, что $p\Delta t S$ люди уходят из категории S в категорию V за время Δt . Поскольку вакцинированные не могут заразиться, это не влияет на категории I или R . Визуализируем категории и движение между ними на рисунке 8.7.

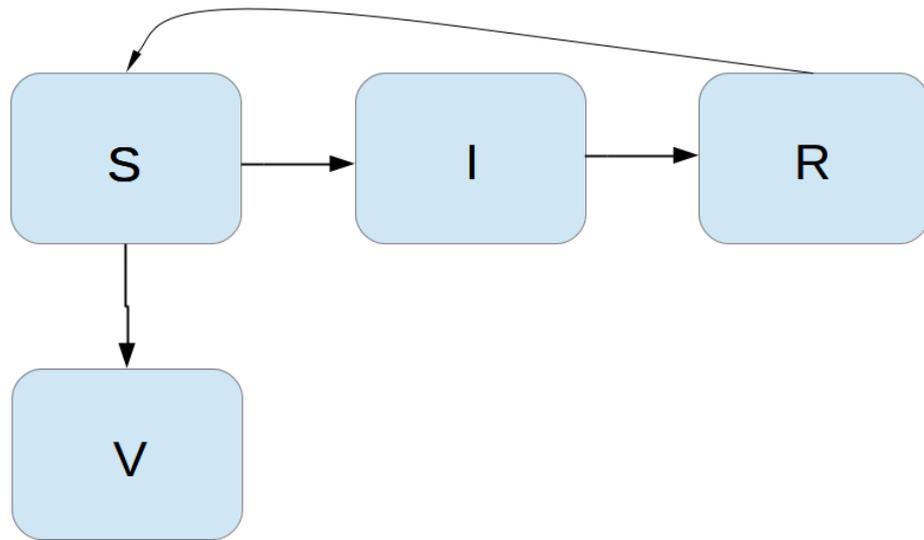


Рисунок 8.7 – Схема распространения гриппа при вакцинации населения

Новые расширенные дифференциальные уравнения с величиной V будут иметь вид:

$$\begin{cases} S' = -\beta SI + \nu R - pS, \\ V' = pS, \\ I' = \beta SI - \gamma I, \\ R' = \gamma I - \nu R. \end{cases} \quad (8.18)$$

Систему (8.18) называют *моделью SIRV*.

Дифференциальное уравнение для V' не представляет трудностей для численного решения. В схеме прямого Эйлера имеет вид:

$$V^{n+1} = V^n + p\Delta t S^n. \quad (8.19)$$

Используя $p = 0,0005$ и $p = 0,0001$ в качестве значений параметра эффективности вакцины, эффект вакцинации показан на рисунке 8.8.

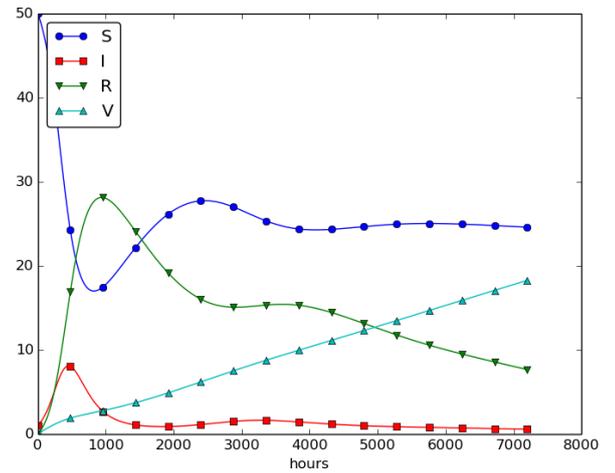
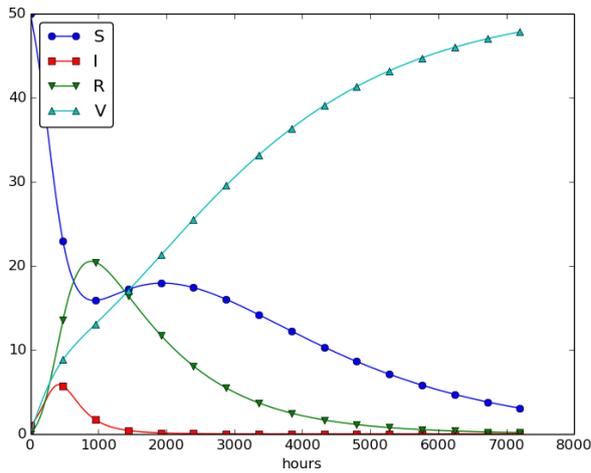


Рисунок 8.8 – Эффект вакцинации: $p = 0.0005$ (слева) и $p = 0.0001$ (справа)

8.2 Моделирование распространения вирусных атак в сетях

Интернет вещей (IoT) открывает возможность объединения различных типов устройств, Интернета и человеческих элементов для обеспечения максимальной взаимосвязанности между ними для достижения полностью связанного мира вещей. Основная адаптация технологии IoT и ее широкое использование также открыли совершенно новую платформу для киберпреступников, которая в основном используется для распределенных атак типа «отказ в обслуживании» (DDoS).

В этом разделе под влиянием внутренних и внешних узлов разработана двойная модель эпидемии, в которой сначала осуществляется атака на устройства IoT, а затем устанавливается распределенная атака вредоносных объектов на основе IoT на целевые ресурсы в сети. Эта модель в основном основана на ботнете Mirai, состоящем из устройств Интернета вещей, который привлек внимание к трем крупным DDoS-атакам в 2016 году. Модель анализируется в точках равновесия, чтобы найти условия для их локальной и глобальной устойчивости. Проанализируем влияние внешних узлов на общую модель и проведем численное моделирование для подтверждения жизнеспособности разработанной модели.

8.2.1 Актуальность проведения исследований в области обеспечения безопасности IoT

Впервые в биологии успешно проанализировано распространение эпидемических заболеваний, таких как чума, оспа, туберкулез, корь, проказа, полиомиелит, малярия, СПИД / ВИЧ, и были достигнуты большие успехи в их искоренении с помощью различных эпидемических моделей. Поскольку компьютерные эпидемии из-за атак вредоносных объектов аналогичны биологическим эпидемиям, возникло использование компьютерных моделей эпидемий. В недавнем прошлом ряд исследователей использовали моделирование эпидемий для анализа атак и защиты вредоносных объектов и их разветвлений в компьютерных сетях, чтобы обеспечить основу для лучшего механизма защиты, помимо решения проблемы атаки.

Недавно были предложены две новые модели эпидемии вредоносных программ, первая из которых основана на распространении двухвирусных вычислений для оценки критериев исчезновения обоих вирусов и выживаемости только одного вируса, а другая основана на SIPS (модель Susceptible-Infected-Patched-Susceptible), которые могут быть распространены по уязвимой сети, чтобы оценить его влияние на распространенность компьютерных вирусов.

Интернет вещей создает новую сетевую парадигму взаимосвязанных объектов с целью улучшить человеческую жизнь с ее повсеместным присутствием. Это расширение Интернета в физическом мире для взаимодействия с физическими системами. Объектом IoT может быть бытовой прибор, медицинское устройство, камера видеонаблюдения, веб-камера, интеллектуальная розетка и почти все, что оснащено датчиками, исполнительными механизмами, блоками питания и встроенными системами, и, что наиболее важно, имеет доступ к сети Интернет. Установлено, что большинство устройств IoT подключены к Интернету через беспроводные сети с использованием таких технологий, как системы радиочастотной

идентификации (RFID) и Wi-Fi, и имеют очень плохие функции безопасности, в основном из-за их низкой мощности и вычислительных возможностей.

В наши дни довольно часто можно увидеть атаки на сеть или сервер, генерируемые тысячами узлов одновременно. Эти типы атак известны как распределенные атаки. Распределенная атака типа «отказ в обслуживании» - это очень популярная распределенная атака, которая сначала устанавливает «армию» вредоносного кода или троянский конь на зараженные узлы различными способами, такими как установка внутри бесплатных игр или медиафайлов или в качестве прикрепления к электронному письму почты. Затем троянский конь создает способ, подобный открытию соединения, для обратной связи со своим хозяином. Наконец, получив команду от хозяина, вся армия начинает массированную атаку.

Согласно отчетам Symantec, атака на веб-сайт BBC первого января 2016 года является крупнейшей в истории DDoS-атакой, достигшей 602 гигабит в секунду (Гбит/с). По данным Gartner, в 2017 году во всем мире использовалось 8,4 миллиарда устройств IoT с поддержкой IP, что на 31 процент больше, чем в 2016 году, и к 2020 году их число достигнет 20,4 миллиарда. Несмотря на растущую популярность Интернета вещей, который имеет огромные перспективы для общественного воздействия, это одна из самых разрушительных технологий из-за их низкой безопасности. Кроме того, уязвимость к DDoS-атакам возрастает с увеличением количества подключений к Интернету критически важных инфраструктур, таких как банковские сети, электрические сети, системы управления воздушным или железнодорожным движением и т. д.

Вредоносные атаки на IP-узел или его сеть также нанесли значительный ущерб отдельным лицам, организациям и странам. Лучшее понимание динамики передачи вредоносных объектов, несомненно, поможет в разработке эффективных стратегий защиты для предотвращения таких вредоносных атак и борьбы с ними. Таким образом, одна из целей данной исследовательской работы - получить точное представление о вредоносных атаках сначала на IP-

устройства IoT, а затем DDoS-атаке на целевые ресурсы в сети с помощью эпидемиологического моделирования. SIR (Susceptible-Infected-Recovered) и SIS (Susceptible-Infected-Susceptible) - две классические модели эпидемии, предложенные Кермаком и Маккендриком для анализа вспышек биологических заболеваний в 1927 и 1932 годах соответственно. Модель SIR в основном применима там, где требуемые люди получают иммунитет против одной и той же атаки, тогда как модель SIS предназначена для тех выздоровевших людей, которые не приобрели иммунитета.

В этом разделе рассматривается модель, которая имеет две части. В первой части достигается моделирование атаки на устройства IoT, которое в основном основано на вышеупомянутой модели SIS вместе с отсеком внешнего узла. Во второй части достигается моделирование DDoS-атаки через этот вредоносный код на целевой ресурс, которое основано на вышеупомянутой модели SIR с только временным иммунитетом вместо постоянного иммунитета. Уязвимые устройства Интернета вещей не только сами по себе, но и сами по себе они также создают серьезную угрозу безопасности любой проводной или беспроводной сетевой инфраструктуры, состоящей из других устройств с выходом в Интернет, таких как компьютер, ноутбук, планшет, смартфон и т. д.

Мобильность - основная функция основной части узлов Интернета вещей в любой беспроводной сети. Из-за этой мобильности очень часто беспроводные узлы подключаются к Интернету, а также отключаются от него. В общем, из-за его мобильности, если IoT-устройство выходит за пределы зоны покрытия, намеренно отключает Wi-Fi или отключается, то можно назвать этот IoT-узел внешним узлом. Даже для проводной сети предположение о полном подключении Интернета несовместимо с ее топологией. Следовательно, в конкретном случае, если узел подключен к Интернету, он известен как внутренний узел, и аналогично, если он отключен от Интернета, он известен как внешний узел. В этом разделе будем рассматривать внешние узлы как узлы IoT, которые отключаются от Интернета только из-за выключения.

Следовательно, здесь внешние узлы - это восстановленные узлы, и повторная перезагрузка делает эти узлы Интернета вещей уязвимыми.

8.2.2 Математическая модель распределенной атаки типа «отказ в обслуживании» в сети IoT

Для рассмотрения математической модели распределенной атаки типа «отказ в обслуживании» сделаем несколько предположений:

1. Каждый атакующий узел (восприимчивый или заразный) отключается от Интернета из-за отключения с постоянной скоростью $\alpha > 0$ для присоединения к внешним атакующим узлам.

2. Поскольку при атаке Mirai было обнаружено, что зараженные устройства Интернета вещей можно очистить путем их перезапуска, предположим, что каждый внешний атакующий узел подключен к Интернету и становится уязвимым только при постоянной скорости $\sigma > 0$.

3. Поскольку наша модель включает жизненную динамику, каждый атакующий узел (восприимчивый, инфекционный или внешний) вымирает с вероятностью $\mu > 0$.

4. μ - это также скорость добавления новых узлов во внешнее узловое отделение.

5. Каждый восприимчивый узел (атакующий или целевой) инфицирован инфекционным атакующим узлом с постоянной скоростью $\beta > 0$.

6. Вылеченный атакующий узел снова становится уязвимым атакующим узлом с постоянной скоростью $\varepsilon_a > 0$.

7. Благодаря эффекту надлежащего лечения каждый инфекционный узел-мишень становится восстановленным узлом-мишенью с постоянной скоростью $\gamma > 0$.

8. Из-за временного иммунитета восстановленный целевой узел снова становится восприимчивым целевым узлом с постоянной скоростью $\varepsilon_r > 0$.

Основываясь на этих гипотезах, рассмотрим модель эпидемии, которая объединяет пять различных аспектов, таких как устройство Интернета вещей,

внутренний или внешний узел, беспроводная сеть, распределенная атака и целевой ресурс, как показано на рисунке 8.9. Обозначения модели представлены в таблице 8.1.

Строение предлагаемой модели учитывает множество факторов. Во-первых, злоумышленники создают множество вредоносного кода, известный как ботнет, нацеливаясь на уязвимые беспроводные узлы атакующего населения. Во-вторых, все множество вредоносного кода коллективно и одновременно проводит массированную атаку на определенную целевую группу.

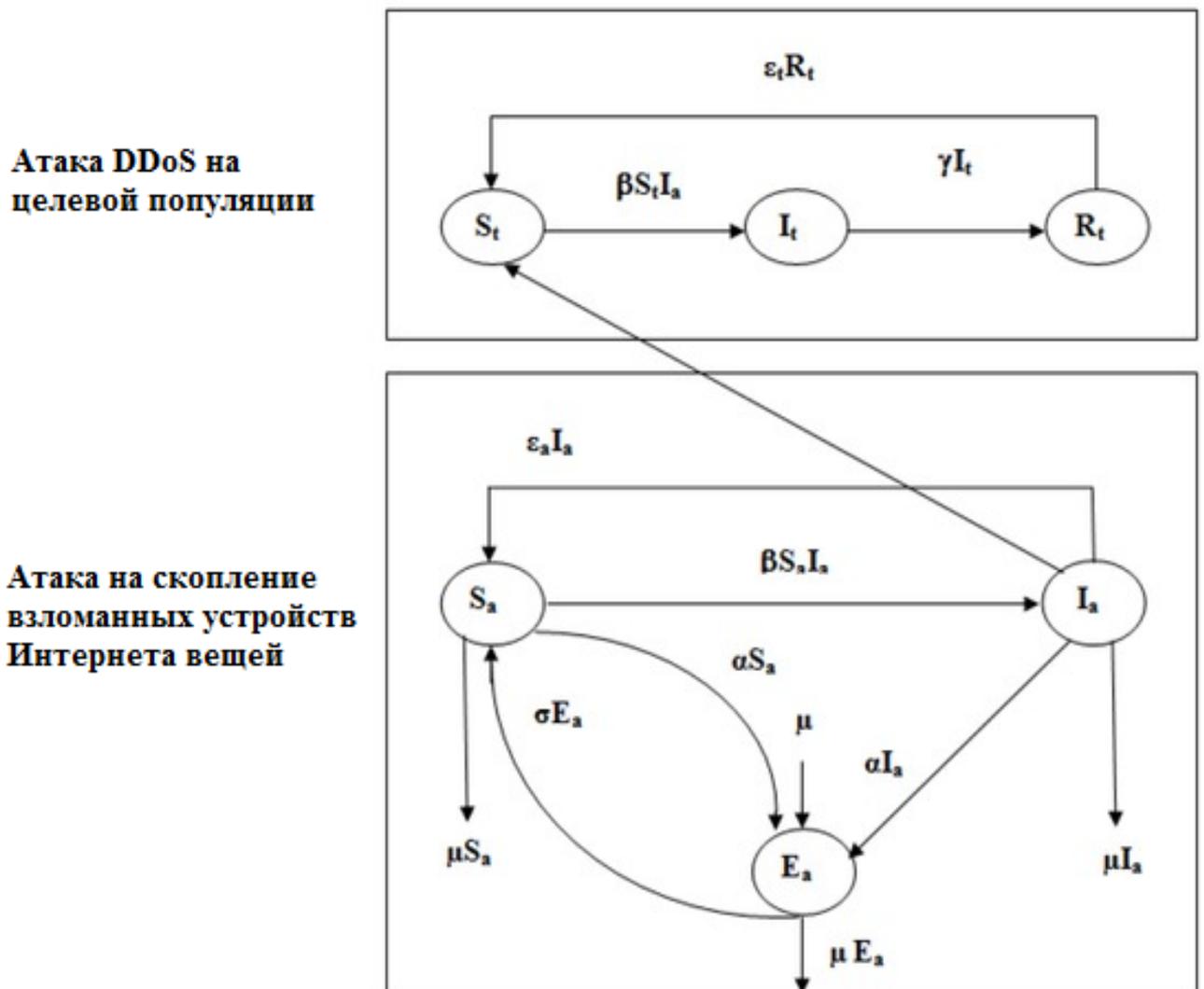


Рисунок 8.9 – Схематическое изображение модели распределенной атаки на целевой ресурс через внутренние и внешние узлы IoT в беспроводной сети

Таблица 8.1 – Обозначения модели

Переменная	Описание
S_t	Восприимчивые целевые узлы
I_t	Инфекционные целевые узлы
R_t	Восстановленные целевые узлы
S_a	Уязвимые атакующие узлы
I_a	Инфекционные атакующие узлы
E_a	Внешние атакующие узлы
β	Скорость контакта на инфекционность
γ	Скорость восстановления инфицированных узлов-мишеней
ε_t	Скорость, с которой восстановленные целевые узлы становятся уязвимыми
ε_a	Скорость, с которой вылеченные атакующие узлы становятся уязвимыми
α	Скорость, с которой атакующие узлы (восприимчивые или заразные) отключаются от Интернета, отключаясь для присоединения к внешним атакующим узлам.
σ	Скорость, с которой внешние атакующие узлы подключаются к Интернету, чтобы стать уязвимыми для атак
μ	Естественная смертность и рождаемость атакующих узлов
R_0	Базовый размер популяции
R_{0a}	Базовая репродуктивная численность атакующей популяции

R_{0t}	Базовая репродуктивная численность целевой группы населения
----------	---

Система ОДУ, которая описывает скорость изменения различных групп (зараженный, восприимчивый и восстановленный) и соответствует вышеупомянутым предположениям формулируется следующим образом:

$$\begin{cases} \frac{dS_t}{dt} = -\beta S_t I_a + \varepsilon_t R_t \\ \frac{dI_t}{dt} = \beta S_t I_a - \gamma I_t \\ \frac{dR_t}{dt} = \gamma I_t - \varepsilon_t R_t \end{cases}, \quad (8.20)$$

где $S_t(t) + I_t(t) + R_t(t) = 1$.

Аналогично для атакующих узлов система обыкновенных дифференциальных уравнений, описывающая скорость изменения разных групп формулируется следующим образом:

$$\begin{cases} \frac{dS_a}{dt} = -\beta S_a I_a - \mu S_a + \varepsilon_a I_a + \sigma E_a - \alpha S_a \\ \frac{dI_a}{dt} = \beta S_a I_a - \mu I_a - \varepsilon_a I_a - \alpha I_a \\ \frac{dE_a}{dt} = \alpha S_a + \alpha I_a - \sigma E_a + \mu - \mu E_a \end{cases}, \quad (8.21)$$

где $S_a(t) + I_a(t) + R_a(t) = 1$.

Система (8.20) и (8.21) сводится к эквивалентной системе обыкновенных дифференциальных уравнений:

$$\begin{cases} \frac{dS_t}{dt} = -\beta S_t I_a + \varepsilon_t (1 - S_t - I_t) \\ \frac{dI_t}{dt} = \beta S_t I_a - \gamma I_t \\ \frac{dI_a}{dt} = \beta (1 - I_a - E_a) I_a - \mu I_a - \varepsilon_a I_a - \alpha I_a \\ \frac{dE_a}{dt} = \alpha (1 - I_a - E_a) + \alpha I_a - \sigma E_a + \mu - \mu E_a \end{cases} \quad (8.22)$$

Область допустимых значений для системы (8.22) может быть задана следующим образом:

$$\psi = \{(S_t, I_t, I_a, E_a) \in R^4 : S_t > 0, I_t \geq 0, I_a \geq 0, E_a \geq 0, S_t + I_t \leq 1, I_a + E_a \leq 1\}. \quad (8.23)$$

Эта допустимая область положительно инвариантна относительно системы (8.22).

8.2.3 Численное моделирование распределенной атаки в сети IoT

Интересной особенностью данной модели является тот факт, что успех или неудача распределенной атаки на целевой ресурс зависит от R_{0a} . Следовательно, моделирование следует проводить либо для $R_{0a} < 1$, либо для $R_{0a} > 1$, в зависимости от ситуации. Численное решение системы (8.22) осуществим с помощью метода Рунге-Кутты 4-го порядка.

Пример 1. Локальная стабильность точки равновесия, свободной от инфекции, была численно смоделирована для сценария, который показан на рисунке 8.10, а соответствующие смоделированные данные для этой неудачной атаки перечислены в таблице 8.2.

В данном примере начальная точка рассматривается как $S_t = 0.97$, $I_t = 0.02$, $R_t = 0.01$, $S_a = 0.55$, $I_a = 0.2$, $E_a = 0.25$ со следующими значениями параметров $\beta = 0.35$, $\varepsilon_t = 0.2$, $\mu = 0.12$, $\varepsilon_a = 0.02$, $\sigma = 0.8$, $\alpha = 0.22$. Значение R_{0a}

равно 0.97, т.е. $R_{0a} < 1$. Очевидно, что точка равновесия E_0 оказывается стабильной.

Таблица 8.2 – Распределение классов узлов для сценария атаки при $R_{0a} < 1$

Время (t)	S_a	I_a	E_a	S_t	I_t	R_t
0	0.55	0.2	0.25	0.97	0.02	0.01
5.38	0.62	0.085	0.29	0.77	0.19	0.04
10.47	0.66	0.04	0.3	0.74	0.2	0.058
15.52	0.58	0.02	0.3	0.76	0.17	0.06
20.18	0.69	0.01	0.3	0.8	0.14	0.06
25.12	0.69	0.01	0.3	0.84	0.11	0.05
30.35	0.7	0.00	0.3	0.87	0.08	0.04
35.45	0.7	0.00	0.3	0.9	0.06	0.03
40.27	0.7	0.00	0.3	0.93	0.05	0.02
45.03	0.7	0.00	0.3	0.95	0.03	0.02
50.08	0.7	0.00	0.3	0.96	0.02	0.01
55.4	0.7	0.00	0.3	0.97	0.02	0.01
60.36	0.7	0.00	0.3	0.98	0.01	0.01
65.41	0.7	0.00	0.3	0.99	0.01	0.00
70.58	0.7	0.00	0.3	0.99	0.01	0.00
75.7	0.7	0.00	0.3	0.99	0.00	0.00
80.21	0.7	0.00	0.3	0.99	0.00	0.00
85.74	0.7	0.00	0.3	1.00	0.00	0.00
90.09	0.7	0.00	0.3	1.00	0.00	0.00
95.02	0.7	0.00	0.3	1.00	0.00	0.00
100	0.7	0.00	0.3	1.00	0.00	0.00

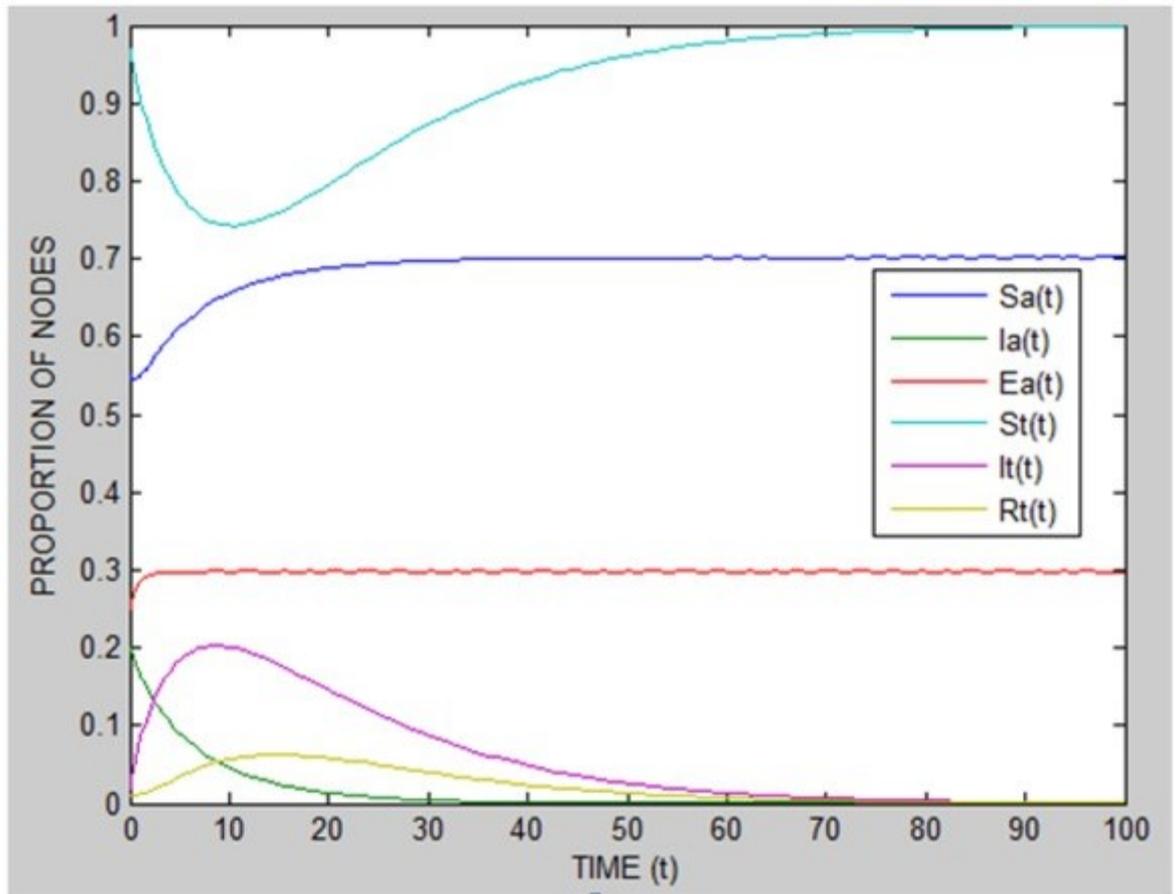


Рисунок 8.10 – Локальная стабильность свободного от инфекции равновесия при $R_{0a} < 1$.

Пример 2. Локальная стабильность точки равновесия была численно смоделирована для сценария, который показан на рисунке 8.11. Здесь начальная точка рассматривается как $S_t = 0.97$, $I_t = 0.02$, $R_t = 0.01$, $S_a = 0.55$, $I_a = 0.2$, $E_a = 0.25$ со следующими значениями параметров $\beta = 0.65$, $\varepsilon_t = 0.2$, $\mu = 0.12$, $\varepsilon_a = 0.05$, $\sigma = 0.5$, $\alpha = 0.1$. Значение R_{0a} равно 2.89, т.е. $R_{0a} > 1$. На рисунке 8.11 видно, что классы I_t и I_a стабилизировались при ненулевых значениях, тем самым показывая стабильность равновесия.

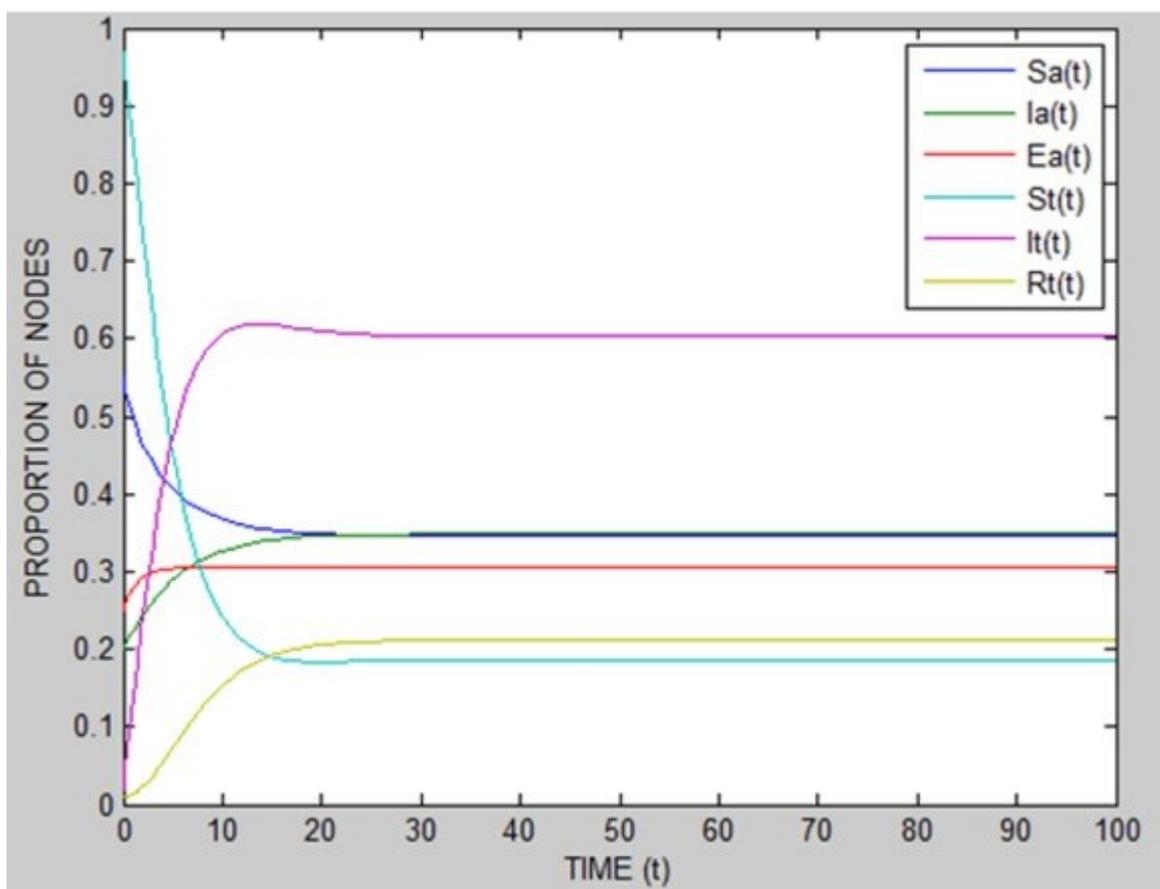


Рисунок 8.11 – Локальная стабильность равновесия при $R_{0a} > 1$.

Таким образом, в данном разделе рассмотрена модель эпидемии DDoS-атак через устройства IoT на целевые ресурсы и проанализирована ее общая динамика. Первая часть этой двойной модели эпидемии на основе IoT разработана для понимания распространения вредоносных атак в беспроводной сети на основе IoT, которая создает множество вредоносного кода, тогда как другая часть модели разработана для понимания DDoS-атаки на целевую сеть с помощью с помощью ранее разработанного ботнета IoT. Данная модель основана на ботнете Mirai, состоящем из устройств Интернета вещей, которые оказались в центре внимания после трех основных DDoS-атак в 2016 году. Получены следующие результаты: (1) точка равновесия E_0 , свободная от инфекций, является локально стабильной, когда $R_{0a} < 1$, и (2) эндемическая точка равновесия E^* локально устойчива, когда $R_{0a} > 1$.

Список использованных источников

- 1 Бахвалов Н.С., Лапин А.В., Чижонков Е.В. Численные методы в задачах и упражнениях. Учебное пособие / Под ред. В.А. Садовниченко. – М.: Высшая школа, 2000.
- 2 Бахвалов Н.С., Жидков Н.П., Кобельков Г.М. Численные методы: учебное пособие для вузов. – СПб: Физматмет, 2000
- 3 Зализняк В.Е. Основы научных вычислений: Введение в численные методы для физиков / Учебное пособие – 296 с.
- 4 Калиткин Н.Н. Численные методы. М.: Наука, 1978. – 512 с.
- 5 Демидович Б.Н., Марон И.А. Основы вычислительной математики. -М.: Наука, 1966.- 664 с.
- 6 Бахвалов Н.С. Численные методы -М.: Наука, 1975. – 632 с.
- 7 Березин Н.С., Жидков Н.П. Методы вычислений. – Т.1. - М.: Наука, 1966. – 464 с.
- 8 Березин Н.С., Жидков Н.П. Методы вычислений. – Т.2. - М.: Физматгиз, 1962.- 640 с.
- 9 Самарский А.А. Теория разностных схем. - М.: Наука, 1983.
- 10 Иванов В.В. Методы вычислений на ЭВМ. Киев: Наукова думка, 1986.
- 11 Тихонов А.Н., Арсенин В.Я. Методы решения некорректных задач. -М.: Наука, 1986, - 288 с.
- 12 Сборник Задач по методам вычислений: Учебное пособие: Для вузов. / Под ред. П.И. Монастырского. - 2-е изд. перераб. и доп. -М.: Физматлит, 1994. -320 с.
- 13 Воробьева Г.Н., Данилова А.Н. Практикум по вычислительной математике. -М.: Высшая школа, 1990.
- 14 Лапчик М.П. Рагулина М.И., Хеннер Е.К. Численные методы: Уч. Пособие для ст. вузов. –М.: Изд. Центр «Академия», 2004. – 384 с.

- 15 Васильев Ф.П. Численные методы решения экстремальных задач: Учебное пособие для вузов - 2-е изд., перераб. и доп. -М.: Наука, Гл. ред. физ.-мат. лит, 1988. -550 с.
- 16 Васильев Ф.П. Методы решения экстремальных задач -М.: Наука, 1981. -400 с.
- 17 Марчук Г.И. Методы вычислительной математики. – М.: Наука, 1980. -536 с.
- 18 Колмогоров А.Н., Фомин С.В. Элементы теории функций и функционального анализа. – М.: Наука, 1976. - 544 с.
- 19 Самарский А.А. Введение в численные методы. – 3-е изд., перераб. – М.: Наука, 1997. - 239 с.
- 20 Тихонов А.Н., Самарский А.А. Уравнения математической физики. – М.: Наука, 1972.
- 21 Шикин Е.В., Плис А.И. Кривые и поверхности на экране компьютера. Руководство по сплайнам для пользователей. – М.: Диалог-МИФИ, 1996 – 240 с.
- 22 Альберг Дж., Нильсон Э., Уолш Дж. Теория сплайнов и их приложения. М.: Наука, 1972.
- 23 Де Бор К. Практическое руководство по сплайнам. - М.: Наука, 1983.
- 24 Foley J.D., van Dam A., Feiner S.K., Hugues J.F. Computer graphics. Principles and practice. Addison-Wesley Pub. Com. 991.
- 25 Боглаев Ю.П. Вычислительная математика и программирование. М.: Высшая школа, 1990.
- 26 Демидович Б.П., Марон И.А., Шувалова Э.З. Численные методы анализа. -М.: Физ.-мат. лит. 1967.
- 27 Хайрер Э., Нерсетт С., Ваннер Г. Решение обыкновенных дифференциальных уравнений. Нежесткие задачи: Пер. с англ. - М.: Мир, 1990. 512 с.

28 Современные численные методы решения обыкновенных дифференциальных уравнений / Под ред. Дж. Холла, Дж. Уатта. М.: Мир, 1979. 312 с.

29 Деккер К., Вервер Я. Устойчивость методов Рунге-Кутты для жестких нелинейных дифференциальных уравнений.- М.: Мир, 1988. 332 с.

30 Олемской И. В. О численном методе интегрирования систем обыкновенных дифференциальных уравнений // Оптимальное управление в механических системах. Л., 1983. С.178-185.

31 Амосов А.А., Дубинский Ю.А., Копченова Н.В. Вычислительные методы для инженеров: Учеб. пособие. – М.: Высш. Шк., 1994. – 544 с.