

МИНИСТЕРСТВО ОБРАЗОВАНИЯ РЕСПУБЛИКИ БЕЛАРУСЬ
УЧРЕЖДЕНИЕ ОБРАЗОВАНИЯ
«БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ ТРАНСПОРТА»

Кафедра «Прикладная математика»

И. Н. КРАВЧЕНЯ

**ПРИМЕНЕНИЕ ЧИСЛЕННЫХ МЕТОДОВ
ПРИ РЕШЕНИИ ЗАДАЧ
ВОДОСНАБЖЕНИЯ И ВОДООТВЕДЕНИЯ**

**Учебно-методическое пособие для студентов
строительного факультета специальности «Водоснабжение,
водоотведение и охрана водных ресурсов»**

Гомель 2009

УДК 519.6: 628 (075.8)
ББК 22.19
К77

Рецензенты – зав. кафедрой «Математические проблемы управления» д-р техн. наук, профессор *И. В. Максимей* (УО «ГГУ им. Ф. Скорины»); зав. кафедрой «Экологии и рационального использования ресурсов» канд. техн. наук, доцент *Р. Н. Вострова* (УО «БелГУТ»).

Кравченя, И. Н.

К77 Применение численных методов при решении задач водоснабжения и водоотведения : учеб.-метод. пособие для студентов строительного факультета специальности «Водоснабжение, водоотведение и охрана водных ресурсов» / И. Н. Кравченя ; М-во образования Респ. Беларусь, Белорус. гос. ун-т трансп. – Гомель : БелГУТ, 2009. – 94 с.
ISBN 978-985-468-628-8

Изложены основные численные методы решения задач алгебры, численного интегрирования, аппроксимации функций и обыкновенных дифференциальных уравнений. Каждый метод иллюстрирован примерами решения задач гидравлики, водоснабжения и водоотведения с использованием системы компьютерной математики MathCAD.

Предназначено для студентов строительного факультета специальности «Водоснабжение, водоотведение и охрана водных ресурсов».

УДК 519.6 : 628.1 (075.8)
ББК 22.19

ISBN 978-985-468-628-8

© Кравченя И. Н., 2009

© Оформление. УО «БелГУТ», 2009

ОГЛАВЛЕНИЕ

Введение. Математическое моделирование и вычислительный эксперимент	6
1 Решение систем линейных алгебраических уравнений	11
1.1 Основные понятия линейной алгебры. Линейные системы.....	11
1.2 О методах решения линейных систем	12
1.3 Прямые методы решения СЛАУ	14
1.3.1 Метод Гаусса	14
1.3.2 Метод прогонки.....	15
1.3.3 Обращение матрицы	16
1.4 Итерационные методы решения СЛАУ	18
1.4.1 Метод простой итерации.....	18
1.4.2 Метод Зейделя.....	20
1.5 Решение СЛАУ в MathCAD.....	22
2 Собственные значения и собственные векторы матриц	25
2.1 Основные определения и свойства собственных значений и векторов	25
2.2 Прямые методы.....	27
2.3 Итерационные методы решения частичной проблемы собственных значений	28
2.4 Нахождение собственных значений и собственных векторов матриц в MathCAD	34
3 Аппроксимация функций	36
3.1 Понятие о приближении функций обобщёнными полиномами	36
3.1.1 Постановка задачи.....	36
3.1.2 Точечная аппроксимация (интерполирование).....	37
3.1.3 Интерполяционный многочлен Лагранжа.....	40
3.1.4 Интерполяционный многочлен Ньютона.....	42
3.1.5. Сплаины. Аппроксимация данных кубическими сплайнами	44
3.1.6. Построение интерполяционных полиномов в MathCAD.....	45
3.1.7 Определение критической глубины при построении кривой свободной поверхности потока с использованием MathCAD	48
3.2 Подбор эмпирических формул	50
3.2.1 Характер опытных данных. Эмпирические формулы	50
3.2.2 Определение параметров эмпирической зависимости. Метод наименьших квадратов.....	52
3.3 Исследование влияния активных илов на приживаемость саженцев сосны	59
4 Численное интегрирование	61
4.1 Задача численного интегрирования. Квадратурные формулы	61
4.2 Метод трапеций	62

4.3 Метод Симпсона	64
4.4 Определения суточной производительности насосной станции второго подъема.....	64
4.5 Обзор методов численного интегрирования	67
4.6 Вычисление определенных интегралов методом Монте-Карло	68
5 Обыкновенные дифференциальные уравнения	70
5.1 Теоретические сведения из курса дифференциальных уравнений.....	70
5.2 Методы решения обыкновенных дифференциальных уравнений	72
5.3 Метод Эйлера.....	74
5.3 Метод Рунге-Кутты	75
5.4 Решение обыкновенных дифференциальных уравнений в MathCAD	75
5.5 Построение кривой свободной поверхности потока неравномерного установившегося движения воды в открытых призматических руслах.....	80
Приложение А. Варианты заданий к лабораторным работам по численным методам.....	84
Приложение Б. Учебная программа по дисциплине «Численные методы».....	98
Список литературы.....	94

ВВЕДЕНИЕ. МАТЕМАТИЧЕСКОЕ МОДЕЛИРОВАНИЕ И ВЫЧИСЛИТЕЛЬНЫЙ ЭКСПЕРИМЕНТ

Эффективное решение прикладных задач гидравлики, водоснабжения и водоотведения невозможно без применения вычислительной техники.

Схема вычислительного эксперимента. В настоящее время выработалась технология исследования сложных проблем, основанная на построении и анализе с помощью ЭВМ математических моделей изучаемого объекта. Такой метод исследования называют *вычислительным экспериментом*. Схема вычислительного эксперимента представлена на рисунке I.

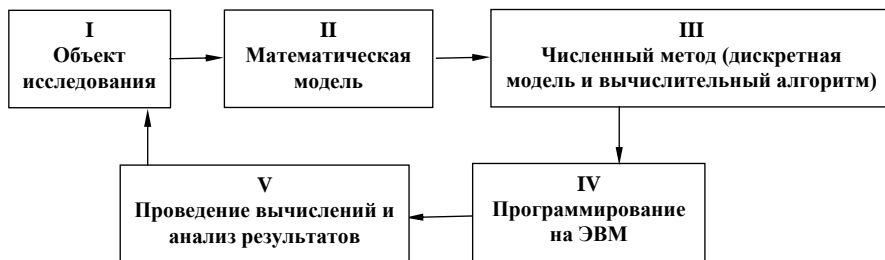


Рисунок I – Схема вычислительного эксперимента

На этапе I формулируются основные законы, описывающие данный объект исследования, строится соответствующая *математическая модель* (II), представляющая обычно запись этих законов в форме системы уравнений (алгебраических, дифференциальных, интегральных и т. д.). После того как задача сформулирована в математической форме, необходимо найти ее решение. Только в исключительных случаях удается найти решение в явном виде, например в виде ряда. Именно на этом этапе требуется привлечение ЭВМ и, как следствие, развитие численных методов (III).

Под *численным методом* понимается такая интерпретация математической модели («дискретная модель»), которая доступна для реализации на ЭВМ. Например, если математическая модель представляет

собой дифференциальное уравнение, то численным методом может быть аппроксимирующее его разностное уравнение совместно с алгоритмом, позволяющим отыскать решение этого разностного уравнения. Результатом реализации численного метода на ЭВМ является число или таблица чисел.

Чтобы реализовать численный метод, необходимо составить программу (IV) или воспользоваться готовой. После отладки программы следует этап проведения вычислений и анализа результатов (V). Полученные результаты изучаются с точки зрения их соответствия исследуемому явлению и при необходимости вносятся исправления в численный метод и уточняется математическая модель.

Метод математического моделирования и вычислительный эксперимент соединяют в себе преимущества традиционных теоретических и экспериментальных методов исследования.

Основные источники и типы погрешностей. Процесс исследования исходного объекта методом математического моделирования и вычислительного эксперимента носит приближенный характер, потому что на каждом этапе вносятся те или иные погрешности. Так, построение математической модели связано с упрощением исходного процесса, недостаточно точным заданием коэффициентов уравнения и других входных данных. По отношению к численному методу, реализующему данную математическую модель, указанные погрешности являются *неустраняемыми*, поскольку они неизбежны в рамках данной модели.

При переходе от математической модели к численному методу возникают погрешности, называемые *погрешностями метода*. Они связаны с тем, что всякий численный метод воспроизводит исходную математическую модель приближенно. Наиболее типичными погрешностями метода являются *погрешность дискретизации* и *погрешность округления*.

Поясним причины возникновения таких погрешностей.

Обычно построение численного метода для заданной математической модели разбивается на *два этапа*:

- а) формулировка дискретной задачи,
- б) разработка вычислительного алгоритма, позволяющего отыскать решение дискретной задачи.

Например, если исходная математическая задача сформулирована в виде системы ДУ, то для численного решения необходимо заменить ее системой конечного, может быть, очень большого числа линейных или разностных алгебраических уравнений. В этом случае говорят, что проведена *дискретизация исходной математической задачи*. Решение дискретизированной задачи отличается от решения исходной задачи. Разность соответствующих решений и называется *погрешностью дискретизации*.

Дискретная модель представляет собой систему большого числа алгебраических уравнений. Невозможно найти решение такой системы точно и в явном виде. Поэтому приходится использовать тот или иной численный алгоритм решения системы алгебраических уравнений. Входные данные этой системы задаются в ЭВМ не точно, а с округлением. В процессе работы алгоритма погрешности округления обычно накапливаются, и в результате решение, полученное на ЭВМ, будет отличаться от точного решения дискретизированной задачи. Результирующая погрешность называется *погрешностью округления* (иногда ее называют *вычислительной погрешностью*). Величина этой погрешности определяется двумя факторами: точностью представления вещественных чисел в ЭВМ и чувствительностью данного алгоритма к погрешностям округления.

Различают **два вида погрешностей**: абсолютную и относительную.

Абсолютная погрешность некоторого числа равна разности между его истинным значением и приближенным значением, полученным в результате вычисления или измерения. *Относительная погрешность* – это отношение абсолютной погрешности к приближенному значению числа.

Таким образом, если a – приближенное значение числа x , то выражения для абсолютной и относительной погрешностей будут выглядеть соответственно следующим образом: $\Delta x = x - a$ и $\delta x = \Delta x / a$.

К сожалению, истинное значение величины x обычно неизвестно. Поэтому приведенные выражения для погрешностей практически не могут быть использованы. Имеется лишь приближенное значение a и нужно найти его *предельную погрешность* Δa , являющуюся верхней оценкой модуля абсолютной погрешности, т. е. $|\Delta x| \leq a$. В дальнейшем значение Δa принимается в качестве абсолютной погрешности приближенного числа a . В этом случае истинное значение x находится в интервале $(a - \Delta a; a + \Delta a)$.

Можно выделить *две группы требований к численным методам*. Первая группа связана с адекватностью дискретной модели исходной математической задаче, вторая – с возможностью реализации данной модели на ЭВМ, т. е. с возможностью получить на ЭВМ решение (препятствия – ограниченный объем оперативной памяти ЭВМ и ресурсы времени счета).

К первой группе относятся такие требования, как сходимость численного метода, устойчивость и корректность.

Устойчивость. Алгоритм называется *устойчивым*, если в процессе его работы вычислительные погрешности возрастают незначительно, и *неустойчивым* – в противоположном случае. При использовании неустойчивых вычислительных алгоритмов накопление погрешностей округления приводит в процессе счета к переполнению арифметического устройства ЭВМ.

Пусть в результате решения задачи по исходному значению величины x находится значение искомой величины y . Если исходная величина имеет абсолютную погрешность Δx , то решение имеет погрешность Δy . Задача называется *устойчивой* по исходному параметру x , если решение y непрерывно от него зависит, т. е. малое приращение исходной величины Δx приводит к малому приращению искомой величины Δy . Другими словами, малые погрешности в исходной величине приводят к малым погрешностям в решении.

Отсутствие устойчивости означает, что даже незначительные погрешности в исходных данных приводят к большим погрешностям в решении или даже к неверному результату. О неустойчивых задачах также говорят, что они *чувствительны* к погрешностям исходных данных.

Иногда бывает, что теоретически задача устойчива, но тем не менее чувствительна к погрешностям исходным данным.

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

Рассмотренная выше неустойчивая задача является некорректно поставленной. Применять для решения таких задач численные методы, как правило, нецелесообразно, поскольку возникающие в расчетах погрешности округления будут сильно возрастать в ходе вычислений, что приведет к значительному искажению результатов. Вместе с тем отметим, что в настоящее время развиты методы решения некоторых некорректных задач. Это в основном так называемые *методы регуляризации*. Они основываются на замене исходной задачи корректно поставленной задачей. Последняя содержит некоторый параметр, при стремлении которого к нулю решение этой задачи переходит в решение исходной задачи.

Неустойчивость методов. Иногда при решении корректно поставленной задачи может оказаться неустойчивым метод ее решения. В частности, по этой причине при вычислении синуса большого аргумента был получен результат, не имеющий смысла.

Численный алгоритм (метод) называется *корректным* в случае существования и единственности численного решения при любых значениях исходных данных, а также в случае устойчивости этого решения относительно погрешностей исходных данных.

Понятие сходимости. При анализе точности вычислительного процесса одним из важнейших критериев является *сходимость* численного метода. Она означает близость получаемого численного решения задачи к истинному решению.

Рассмотрим понятие *сходимости итерационного процесса*. Этот процесс состоит в том, что для решения некоторой задачи и нахождения искомого значения определяемого параметра (например, корня нелинейного

уравнения) применяют метод последовательных приближений. В результате многократного повторения этого процесса (или *итераций*) получаем последовательность значений $x_1, x_2, \dots, x_n, \dots$. Говорят, что эта последовательность сходится к точному решению $x = a$, если при неограниченном возрастании числа итераций предел этой последовательности существует и равен a : $\lim_{n \rightarrow \infty} x_n = a$. В этом случае имеем сходящийся численный метод.

Другой подход к понятию сходимости используется в методах дискретизации. Эти методы заключаются в замене задачи с непрерывными параметрами на задачу, в которой значения функций вычисляются в фиксированных точках. Это относится, в частности, к численному интегрированию, решению дифференциальных уравнений и т. п. Здесь под *сходимостью метода* понимается стремление значений решения дискретной модели задачи к соответствующим значениям решения исходной задачи при стремлении к нулю параметра дискретизации (например, шага интегрирования).

Таким образом, для получения решения задачи с необходимой точностью ее постановка должна быть корректной, а используемый численный метод должен обладать устойчивостью (корректностью) и сходимостью.

1 РЕШЕНИЕ СИСТЕМ ЛИНЕЙНЫХ АЛГЕБРАИЧЕСКИХ УРАВНЕНИЙ

К численным методам линейной алгебры относятся численные методы решения систем линейных алгебраических уравнений (СЛАУ), вычисление определителя, обратной матрицы, нахождение собственных значений и собственных векторов матриц.

К решению СЛАУ сводятся некоторые практические задачи водоснабжения. Например, при моделировании работы скважинных водозаборов для каждого узла расчетной схемы составляется система линейных уравнений исходя из условий баланса расходов воды.

1.1 Основные понятия линейной алгебры. Линейные системы

Запишем систему n линейных алгебраических уравнений с n неизвестными:

$$\begin{aligned} a_{11}x_1 + a_{12}x_2 + \dots + a_{1n}x_n &= b_1; \\ a_{21}x_1 + a_{22}x_2 + \dots + a_{2n}x_n &= b_2; \\ \dots & \\ a_{n1}x_1 + a_{n2}x_2 + \dots + a_{nn}x_n &= b_n. \end{aligned}$$

Совокупность коэффициентов этой системы запишем в виде матрицы:

$$A = \begin{pmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & a_{22} & \dots & a_{2n} \\ \dots & \dots & \dots & \dots \\ a_{n1} & a_{n2} & \dots & a_{nn} \end{pmatrix}.$$

Данная матрица n^2 элементов, состоящая из n строк и n столбцов, называется *квадратной матрицей* порядка n . Если подобная матрица содержит mn элементов, расположенных в m строках и n столбцах, то она называется *прямоугольной матрицей*.

Можно записать в векторно-матричном виде: $AX = b$, где X и b – вектор-столбец неизвестных и вектор-столбец правых частей соответственно:

$$X = \begin{pmatrix} x_1 \\ x_2 \\ \dots \\ x_n \end{pmatrix}; \quad b = \begin{pmatrix} b_1 \\ b_2 \\ \dots \\ b_n \end{pmatrix}.$$

Необходимым и достаточным условием существования единственного решения системы линейных уравнений является условие $D \neq 0$. В случае равенства нулю определителя системы матрица называется *вырожденной*; при этом система линейных уравнений либо не имеет решения, либо имеет их бесконечное множество.

Замечание – На практике, особенно при вычислениях на компьютере, когда происходят округление или отбрасывание младших разрядов чисел, далеко не всегда удается получить точное равенство определителя нулю ($D \approx 0$). Таким образом, малые погрешности вычислений или исходных данных могут привести к существенным погрешностям в решении. Такие системы уравнений называются *плохо обусловленными*. Заметим, что условие $D \approx 0$ является необходимым для плохой обусловленности системы линейных уравнений, но недостаточным.

1.2 О методах решения линейных систем

Методы решения систем линейных уравнений делятся на две группы – прямые и итерационные.

Прямые методы используют конечные соотношения (формулы) для вычисления неизвестных, позволяющие получить решение после выполнения заранее известного числа операций. Эти методы сравнительно *просты* и наиболее *универсальны*, т. е. пригодны для решения широкого класса линейных систем. К ним относятся: правило Крамера нахождения решения с помощью определителей, метод Гаусса (метод исключений), метод прогонки, метод обратной матрицы и др.

Прямые методы имеют и ряд *недостатков*:

- требуют хранения в оперативной памяти компьютера сразу всей матрицы, и при больших значениях n расходуется много места в памяти;
- не учитывают структуру матрицы при большом числе нулевых элементов в разреженных матрицах. Эти элементы занимают место в памяти машины, и над ними проводятся арифметические действия (исключение –

метод прогонки);

- накапливание погрешностей в процессе решения, поскольку вычисления на любом этапе используют результаты предыдущих операций.

Последнее особенно опасно для больших систем, когда резко возрастает общее число операций, а также для плохо обусловленных систем, весьма чувствительных к погрешностям. В связи с этим прямые методы используют обычно для не слишком больших ($n < 1000$) систем с плотно заполненной матрицей и не близким к нулю определителем.

Замечание – Прямые методы решения линейных систем иногда называют *точными*, поскольку решение выражается в виде точных формул через коэффициенты системы. Однако точное решение может быть получено лишь при точном выполнении вычислений (и, разумеется, при точных значениях коэффициентов системы). На практике же при использовании компьютеров вычисления проводятся с погрешностями. Поэтому неизбежны погрешности и в окончательных результатах.

Итерационные методы – это методы последовательных приближений. В них необходимо задать некоторое приближенное решение – *начальное приближение*. После этого с помощью некоторого алгоритма проводится один цикл вычислений, называемый *итерацией*. В результате итерации находят новое приближение. Итерации проводятся до получения решения с требуемой точностью. Алгоритмы решения СЛАУ с использованием итерационных методов обычно более сложные по сравнению с прямыми методами. Объем вычислений заранее определить трудно.

Тем не менее итерационные методы в ряде случаев *предпочтительнее*, т. к. обладают следующими *достоинствами*:

- требуют хранения в памяти машины не всей матрицы системы, а лишь нескольких векторов с n компонентами;

- иногда элементы матрицы можно совсем не хранить, а вычислять их по мере необходимости;

- погрешности окончательных результатов при использовании итерационных методов не накапливаются, поскольку точность вычислений в каждой итерации определяется лишь результатами предыдущей итерации и практически не зависит от ранее выполненных вычислений.

Эти достоинства итерационных методов делают их особенно полезными в случае большого числа уравнений, а также плохо обусловленных систем. Следует отметить, что при этом сходимость итераций может быть очень медленной; поэтому ищут эффективные пути ее ускорения.

Итерационные методы могут использоваться для уточнения решений, полученных с помощью прямых методов. Такие смешанные алгоритмы

обычно довольно эффективны, особенно для плохо обусловленных систем. В последнем случае могут также применяться методы регуляризации.

1.3 Прямые методы решения СЛАУ

1.3.1 Метод Гаусса

Метод Гаусса основан на приведении матрицы системы к треугольному виду. Это достигается последовательным исключением неизвестных из уравнений системы. Сначала с помощью первого уравнения исключается x_1 из всех последующих уравнений системы. Затем с помощью второго уравнения исключается x_2 из третьего и всех последующих уравнений. Этот процесс, называемый *прямым ходом метода Гаусса*, продолжается до тех пор, пока в левой части последнего (n -го) уравнения не останется лишь один член с неизвестным x_n , т. е. матрица системы будет приведена к треугольному виду.

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

Описанные процедуры применимы лишь для систем с невырожденной матрицей ($D \neq 0$). В противном случае (при условии, что вычисления проводятся точно) с помощью метода Гаусса можно ответить на вопрос, имеет ли система бесконечное множество решений или не имеет ни одного. Однако эти случаи мы в дальнейшем рассматривать не будем, предполагая, что матрица системы невырожденная.

Пример 1.1. Решение СЛАУ методом Гаусса:

$$\begin{aligned}10 x_1 - 7 x_2 &= 7; \\ -3 x_1 + 3 x_2 + 6 x_3 &= 4; \\ 5 x_1 - x_2 + 5 x_3 &= 6.\end{aligned}$$

Исключим x_1 из второго и третьего уравнений. Для этого сначала умножим первое уравнение на 0,3 и результат прибавим ко второму, а затем умножим первое же уравнение на $-0,5$ и результат прибавим к третьему. Получим:

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

Умножим второе уравнение на 25 и результат сложим с третьим уравнением. Получим систему в треугольном виде:

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

На этом заканчивается прямой ход метода Гаусса.

Обратный ход состоит в последовательном вычислении x_3 , x_2 , x_1 соответственно из третьего, второго, первого уравнений. $x^* = (0; -1; 1)$ – решение.

Погрешность решения СЛАУ методом Гаусса. Запишем систему в матричном виде: $AX = b$. Решение этой системы можно представить в виде $X = A^{-1}b$. Однако вычисленное по методу Гаусса решение X^* отличается от этого решения из-за погрешностей округлений, связанных с ограниченностью разрядной сетки машины.

Существуют две величины, характеризующие степень отклонения полученного решения от точного. Одна из них – *погрешность* Δx , равная разности этих значений; другая – *невязка* r , равная разности между левой и правой частями уравнений при подстановке в них решения:

$$\Delta X = X - X^*, \quad r = AX^* - b.$$

Можно показать, что если одна из этих величин равна нулю, то и другая должна равняться нулю. Однако из малости одной не следует малость другой. При $\Delta X \approx 0$ обычно $r \approx 0$, но обратное утверждение справедливо не всегда. В частности, для плохо обусловленных систем при $r \approx 0$ погрешность решения может быть большой.

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

1.3.2 Метод прогонки

Метод прогонки является модификацией метода Гаусса для частного случая разреженных систем – системы уравнений с *трехдиагональной матрицей*. Такие системы получаются при моделировании некоторых инженерных задач, а также при численном решении краевых задач для дифференциальных уравнений.

Запишем систему уравнений в виде

$$b_1 x_1 + c_1 x_2 = d_1;$$

$$\begin{aligned}
a_2 x_1 + b_2 x_2 + c_2 x_3 &= d_2; \\
a_3 x_2 + b_3 x_3 + c_3 x_4 &= d_3; \\
&\dots\dots\dots \\
a_{n-1} x_{n-2} + b_{n-1} x_{n-1} + c_{n-1} x_n &= d_{n-1}; \\
a_n x_{n-1} + b_n x_n &= d_n.
\end{aligned}
\tag{1.1}$$

На главной диагонали матрицы этой системы стоят элементы b_1, b_2, \dots, b_n , над ней – c_1, c_2, \dots, c_n , под ней – a_1, a_2, \dots, a_n . При этом обычно все коэффициенты b_i не равны нулю, остальные элементы матрицы равны нулю.

Метод прогонки состоит из двух этапов – *прямой прогонки* (аналога прямого хода метода Гаусса) и *обратной прогонки* (аналога обратного хода метода Гаусса). Прямая прогонка состоит в вычислении прогоночных коэффициентов A_i, B_i , с помощью которых каждое неизвестное x_i выражается через x_{i+1} :

$$x_i = A_i x_{i+1} + B_i, \quad i = 1, 2, \dots, n-1. \tag{1.2}$$

Вычислим прогоночные коэффициенты:

$$A_1 = -\frac{c_1}{b_1}, \quad B_1 = \frac{d_1}{b_1}; \tag{1.3}$$

для любого номера i :

$$A_i = \frac{-c_i}{e_i}, \quad B_i = \frac{d_i - a_i B_{i-1}}{e_i}, \quad e_i = a_i A_{i-1} + b_i, \quad i = 2, 3, \dots, n-1. \tag{1.4}$$

Обратная прогонка состоит в последовательном вычислении неизвестных x_i . Сначала нужно найти x_n :

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

Далее, используя формулы (1.2) и вычисленные ранее по формулам (1.3), (1.4) прогоночные коэффициенты, последовательно вычисляем все неизвестные $x_{n-1}, x_{n-2}, \dots, x_1$.

1.3.3 Обращение матрицы

Непосредственное нахождение определителя требует большого объема вычислений. Вместе с тем легко вычисляется определитель треугольной матрицы: он равен произведению ее диагональных элементов.

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

$$\det A = (-1)^k \prod_{i=1}^n a_{ii} \cdot \quad (1.5)$$

Здесь диагональные элементы a_{ii} берутся из преобразованной (а не исходной) матрицы. Через k обозначено число перестановок строк (или столбцов) матрицы при ее приведении к треугольному виду (для получения ненулевого или максимального по модулю ведущего элемента на каждом этапе исключения). Благодаря методу исключения можно вычислять определители 1000-го и большего порядков, и объем вычислений значительно меньше, чем в приведенных ранее оценках.

Найдем обратную матрицу A^{-1} . Обозначим ее элементы через z_{ij} . Запишем равенство $AA^{-1} = E$ в виде

$$\begin{pmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & a_{22} & \dots & a_{2n} \\ \dots & \dots & \dots & \dots \\ a_{n1} & a_{n2} & \dots & a_{nn} \end{pmatrix} \cdot \begin{pmatrix} z_{11} & z_{12} & \dots & z_{1n} \\ z_{21} & z_{22} & \dots & z_{2n} \\ \dots & \dots & \dots & \dots \\ z_{n1} & z_{n2} & \dots & z_{nn} \end{pmatrix} = \begin{pmatrix} 1 & 0 & \dots & 0 \\ 0 & 1 & \dots & 0 \\ \dots & \dots & \dots & \dots \\ 0 & 0 & \dots & 1 \end{pmatrix} \quad (1.6)$$

Отсюда следует, что

$$A z_j = e_j, \quad j = 1, 2, \dots, n,$$

где z_j и e_j — j -е столбцы матриц A^{-1} и E соответственно.

Таким образом, для нахождения j -го столбца обратной матрицы нужно решить систему уравнений (1.6). Решив n таких систем для $j = 1, 2, \dots, n$, найдем все столбцы z_j и, следовательно, саму обратную матрицу.

Поскольку при разных j матрица A системы (1.6) не меняется, исключение неизвестных при использовании метода Гаусса (прямой ход) проводится только один раз, причем сразу для всех правых частей — столбцов e_j . Затем для каждой из систем (1.6) делается обратный ход с соответствующей преобразованной правой частью.

Оценки показывают, что это весьма экономичный способ обращения матрицы. Он требует примерно лишь в три раза больше действий, чем при решении одной системы уравнений.

1.4 Итерационные методы решения СЛАУ

1.4.1 Метод простой итерации

Найдем решение системы линейных алгебраических уравнений. Запишем исходную систему уравнений в векторно-матричном виде:

$$AX = b.$$

Приведем исходную систему к стандартному виду (при этом предполагается, что все элементы a_{ii} отличны от нуля.):

$$x_i = -\sum_{j=1}^{i-1} \frac{a_{ij}}{a_{ii}} x_j - \sum_{j=i+1}^n \frac{a_{ij}}{a_{ii}} x_j + \frac{b_i}{a_{ii}}, \quad i = \overline{1, n}, \quad (1.7)$$

или в матричной форме

$$X = BX + f. \quad (1.8)$$

Получившаяся система эквивалентна исходной системе и служит основой для построения метода простой итерации.

Выберем некоторое начальное приближение $x_i^{(0)}$ $i = \overline{1, n}$ и подставим его в правую часть системы (1.7):

$$x_i^{(1)} = -\sum_{j=1}^{i-1} \frac{a_{ij}}{a_{ii}} x_j^{(0)} - \sum_{j=i+1}^n \frac{a_{ij}}{a_{ii}} x_j^{(0)} + \frac{b_i}{a_{ii}}, \quad i = \overline{1, n}. \quad (1.8^*)$$

Полученное решение $x_i^{(1)}$ $i = \overline{1, n}$ будем рассматривать в качестве следующего (первого) приближения к решению. Аналогично, по известному k -му приближению можно найти $(k+1)$ -е приближение:

$$x_i^{(k+1)} = -\sum_{j=1}^{i-1} \frac{a_{ij}}{a_{ii}} x_j^{(k)} - \sum_{j=i+1}^n \frac{a_{ij}}{a_{ii}} x_j^{(k)} + \frac{b_i}{a_{ii}}, \quad i = \overline{1, n}, \quad (1.9)$$

или

$$x_i^{(k+1)} = Bx_i^{(k)} + f, \quad i = \overline{1, n}. \quad (1.10)$$

Окончание итераций определяется либо заданием максимального числа итераций n_0 , либо условием

$$\max_{1 \leq i \leq n} |x_i^{(k+1)} - x_i^{(k)}| \leq \varepsilon, \quad (1.11)$$

где $\varepsilon > 0$ – заданное число.

Придем теорему (без доказательства), дающую необходимое и достаточное условие сходимости метода простой итерации.

Теорема 1.1. Пусть система (1.8) имеет единственное решение. Метод простой итерации (1.10) сходится при любом начальном приближении $X^{(0)}$ тогда и только тогда, когда все собственные числа матрицы B по модулю меньше единицы.

Однако на практике теоремой 1.1 воспользоваться непросто, так как требуется знание границ собственных значений матрицы B .

Для практических расчетов для оценки погрешности k -го приближения используют следующую формулу:

$$\|X^* - X^{(k)}\| \leq \frac{\|B\|^{k+1}}{1 - \|B\|} \|f\|. \quad (1.12)$$

Не проводя вычислений, формула (1.12) позволяет, используя норму матриц $\|B\|$ и $\|f\|$, по заданной точности ε определить число итераций:

$$k = \frac{1}{\ln(\|B\|)} \cdot \ln \left(\frac{\varepsilon (1 - \|B\|)}{\|f\|} \right). \quad (1.13)$$

Пример 1.2. Методом итераций решить систему линейных уравнений с точностью до 0,001:

$$\begin{cases} x_1 = 0,13x_1 + 0,22x_2 - 0,33x_3 + 0,07x_4 + 0,11; \\ x_2 = 0,45x_1 - 0,23x_3 + 0,07x_4 - 0,33; \\ x_3 = 0,11x_1 - 0,08x_2 + 0,18x_4 + 0,85; \\ x_4 = 0,08x_1 + 0,09x_2 + 0,33x_3 + 0,21x_4 - 1,7. \end{cases}$$

Так как норма матрицы, состоящей из коэффициентов при неизвестных в правых частях уравнений ($\|B\|_1 = \max_i \sum_{j=1}^n |b_{ij}|$),

$$\|B\|_1 = \max_i \{0,13 + 0,22 + 0,33 + 0,07; 0,45 + 0,23 + 0,07;$$

$$0,11 + 0,08 + 0,18; 0,08 + 0,09 + 0,33 + 0,21\} = \max\{0,75; 0,75; 0,37; 0,68\} = 0,75 < 1,$$

то итерационный процесс сходится.

В качестве начального приближения выберем свободные члены правых частей уравнений:

$$x_1^{(0)} = 0,11, \quad x_2^{(0)} = -0,33, \quad x_3^{(0)} = 0,85, \quad x_4^{(0)} = -1,7.$$

Последующие приближения получим следующим образом.

Первая итерация:

$$\begin{cases} x_1^{(1)} = 0,13 \cdot 0,11 + 0,22 \cdot (-0,33) - 0,33 \cdot 0,85 + 0,07 \cdot (-1,7) + 0,11 = -0,3478; \\ x_2^{(1)} = 0,45 \cdot 0,11 - 0,23 \cdot 0,85 + 0,07 \cdot (-1,7) - 0,33 = -0,595; \\ x_3^{(1)} = 0,11 \cdot 0,11 - 0,08 \cdot (-0,33) + 0,18 \cdot (-1,7) + 0,85 = 0,5825; \\ x_4^{(1)} = 0,08 \cdot 0,11 + 0,09 \cdot (-0,33) + 0,33 \cdot 0,85 + 0,21 \cdot (-1,7) - 1,7 = -1,7974. \end{cases}$$

Вторая итерация:

$$\begin{cases} x_1^{(2)} = 0,13 \cdot (-0,3478) + 0,22 \cdot (-0,595) - 0,33 \cdot 0,5825 + 0,07 \cdot (-1,7974) + 0,11 = -0,38416; \\ x_2^{(2)} = 0,45 \cdot (-0,3478) - 0,23 \cdot 0,5825 + 0,07 \cdot (-1,7974) - 0,33 = -0,7463; \\ x_3^{(2)} = 0,11 \cdot (-0,3478) - 0,08 \cdot (-0,595) + 0,18 \cdot (-1,7974) + 0,85 = 0,53581; \\ x_4^{(2)} = 0,08 \cdot (-0,3478) + 0,09 \cdot (-0,595) + 0,33 \cdot 0,5825 + 0,21 \cdot (-1,7974) - 1,7 = -1,9666. \end{cases}$$

Процесс вычисления продолжаем до тех пор, пока не достигнем требуемой точности. Результаты вычисления представим в таблице.

Таблица 2.1 – Результаты расчетов

k	$x_1^{(k)}$	$x_2^{(k)}$	$x_3^{(k)}$	$x_4^{(k)}$
0	0,11000	-0,33000	0,85000	-1,7000
1	-0,34780	-0,59500	0,58250	-1,79740
2	-0,38416	-0,74630	0,53581	-1,9666
3	-0,41861	-0,76377	0,51346	-2,03407
4	-0,42427	-0,77885	0,49892	-2,05994
5	-0,42534	-0,77987	0,49485	-2,07198
6	-0,42521	-0,78026	0,49265	-2,07603
7	-0,42483	-0,77997	0,49196	-2,07763
8	-0,42460	-0,77976	0,49169	-2,07814

Сходимость в тысячных долях имеет место уже на восьмом шаге.

Решение СЛАУ: $x_1 \approx -0,425$; $x_2 \approx -0,780$; $x_3 \approx 1,492$; $x_4 \approx -2,078$.

1.4.2 Метод Зейделя

Одним из самых распространенных итерационных методов, отличающийся простотой и легкостью программирования, является *метод Зейделя*. Отличие метода Зейделя от метода простой итерации состоит в том,

что при вычислении $(k + 1)$ -го приближения полученные компоненты вектора $x^{(k+1)}$ сразу же используются в вычислениях:

$$x_i^{k+1} = -\sum_{j=1}^{i-1} \frac{a_{ij}}{a_{ii}} x_j^{k+1} - \sum_{j=i+1}^n \frac{a_{ij}}{a_{ii}} x_j^k + \frac{b_i}{a_{ii}}, \quad i = 1, 2, \dots, n. \quad (1.14)$$

В некоторых случаях метод Зейделя дает более быструю сходимость по сравнению с методом простой итерации.

Для сходимости итерационного процесса (1.14) достаточно, чтобы модули диагональных коэффициентов для каждого уравнения системы были не меньше сумм модулей всех остальных коэффициентов (преобладание диагональных элементов):

$$\left| a_{ii} \right| \geq \sum_{j \neq i} \left| a_{ij} \right|, \quad i = 1, 2, \dots, n. \quad (1.15)$$

При этом хотя бы для одного уравнения неравенство должно выполняться строго. Эти условия являются достаточными для сходимости метода, но они не являются необходимыми, т. е. для некоторых систем итерации сходятся и при нарушении условий (1.15).

Пример 1.3. Методом Зейделя решить с точностью 0,001 систему линейных уравнений:

$$\begin{cases} x_1 = 0,13x_1 + 0,22x_2 - 0,33x_3 + 0,07x_4 + 0,11; \\ x_2 = 0,45x_1 - 0,23x_3 + 0,07x_4 - 0,33; \\ x_3 = 0,11x_1 - 0,08x_2 + 0,18x_4 + 0,85; \\ x_4 = 0,08x_1 + 0,09x_2 + 0,33x_3 + 0,21x_4 - 1,7. \end{cases}$$

Так как норма матрицы, состоящей из коэффициентов при неизвестных в правых частях уравнений ($\|B\|_1 = \max_i \sum_{j=1}^n |b_{ij}|$),

$$\|B\|_1 = \max_i \{0,13 + 0,22 + 0,33 + 0,07; 0,45 + 0,23 + 0,07;$$

$$0,11 + 0,08 + 0,18; 0,08 + 0,09 + 0,33 + 0,21\} = \max\{0,75; 0,75; 0,37; 0,68\} = 0,75 < 1,$$

то итерационный процесс сходится.

В качестве начального приближения выберем свободные члены правых частей уравнений:

$$x_1^{(0)} = 0,11, \quad x_2^{(0)} = -0,33, \quad x_3^{(0)} = 0,85, \quad x_4^{(0)} = -1,7.$$

Последующие приближения получим следующим образом.

Первая итерация:

$$\begin{cases} x_1^{(1)} = 0,13 \cdot 0,11 + 0,22 \cdot (-0,33) - 0,33 \cdot 0,85 + 0,07 \cdot (-1,7) + 0,11 = -0,3478; \\ x_2^{(1)} = 0,45 \cdot 0,3478 - 0,23 \cdot 0,85 + 0,07 \cdot (-1,7) - 0,33 = -0,80101; \\ x_3^{(1)} = 0,11 \cdot 0,3478 - 0,08 \cdot (-0,80101) + 0,18 \cdot (-1,7) + 0,85 = 0,569823; \\ x_4^{(1)} = 0,08 \cdot 0,3478 + 0,09 \cdot (-0,80101) + 0,33 \cdot 0,569823 + 0,21 \cdot (-1,7) - 1,7 = -1,96887. \end{cases}$$

Вторая итерация:

$$\begin{cases} x_1^{(2)} = 0,13 \cdot (-0,3478) + 0,22 \cdot (-0,595) - 0,33 \cdot 0,5825 + 0,07 \cdot (-1,7974) + 0,11 = -0,4373; \\ x_2^{(2)} = 0,45 \cdot (-0,4373) - 0,23 \cdot 0,5825 + 0,07 \cdot (-1,7974) - 0,33 = -0,79566; \\ x_3^{(2)} = 0,11 \cdot (-0,4373) - 0,08 \cdot (-0,79566) + 0,18 \cdot (-1,7974) + 0,85 = 0,511153; \\ x_4^{(2)} = 0,08 \cdot (-0,4373) + 0,09 \cdot (-0,79566) + 0,33 \cdot 0,511153 + 0,21 \cdot (-1,7974) - 1,7 = -2,05138. \end{cases}$$

Процесс вычисления продолжаем до тех пор, пока не достигнем требуемой точности. Результаты вычислений представим в таблице:

Таблица 2.1 – Результаты расчетов

k	$x_1^{(k)}$	$x_2^{(k)}$	$x_3^{(k)}$	$x_4^{(k)}$
0	0,11000	-0,33000	0,85000	-1,7000
1	-0,34780	-0,80101	0,56982	-1,96887
2	-0,43730	-0,79566	0,51115	-2,05138
3	-0,43417	-0,78654	0,49592	-2,07266
4	-0,42822	-0,78185	0,49237	-2,07740
5	-0,42557	-0,78017	0,49167	-2,07826
6	-0,42469	-0,77967	0,49157	-2,07836

Сходимость в тысячных долях имеет место на шестом шаге.

Решение СЛАУ: $x_1 \approx -0,425$; $x_2 \approx -0,780$; $x_3 \approx 1,492$; $x_4 \approx -2,078$.

1.5 Решение СЛАУ в MathCAD

В системе компьютерной математики MathCAD СЛАУ можно решить как в более наглядной, так и в более удобной для записи форме.

Для первого способа следует использовать вычислительный блок, состоящий из трех частей, идущих последовательно друг за другом:

- Given – ключевое слово;
- система, записанная логическими операторами;

□ $\text{Find}(x_1, x_2, \dots, x_m)$ – встроенная функция для решения системы относительно переменных x_1, x_2, \dots, x_m .

Блок `Given/Find` использует для поиска решения итерационные методы, поэтому требуется задать начальные значения для всех x_1, x_2, \dots, x_m . Сделать это необходимо до ключевого слова `Given`.

Приведем пример решения СЛАУ в MathCAD с помощью вычислительного блока:

$$\begin{aligned}
 &x1 := 0 \quad x2 := 0 \quad x3 := 0 \quad x4 := 0 \\
 &\text{Given} \\
 &x1 = 0.13 \cdot x1 + 0.22 \cdot x2 - 0.33 \cdot x3 + 0.07 \cdot x4 + 0.11 \\
 &x2 = 0.45 \cdot x1 - 0.23 \cdot x3 + 0.07 \cdot x4 - 0.33 \\
 &x3 = 0.11 \cdot x1 - 0.08 \cdot x2 + 0.18 \cdot x4 + 0.85 \\
 &x4 = 0.08 \cdot x1 + 0.09 \cdot x2 + 0.33 \cdot x3 + 0.21 \cdot x4 - 1.7 \\
 &\text{Find}(x1, x2, x3, x4) = \begin{pmatrix} -0.424 \\ -0.78 \\ 0.492 \\ -2.078 \end{pmatrix}
 \end{aligned}$$

Листинг 1.1 – Решение СЛАУ с помощью вычислительного блока

Для второго способа следует использовать встроенную функцию `lsolve`.

□ $\text{lsolve}(A, b)$ – решение системы линейных уравнений, где:

- A – матрица коэффициентов системы;
- b – вектор правых частей.

Применение функции `lsolve` рассмотрим на примере:

$$\begin{aligned}
 A &:= \begin{pmatrix} -0.87 & 0.22 & -0.33 & 0.07 \\ 0.45 & -1 & -0.23 & 0.07 \\ 0.11 & -0.08 & -1 & 0.18 \\ 0.08 & 0.09 & 0.33 & -0.79 \end{pmatrix} & b &:= \begin{pmatrix} -0.11 \\ 0.33 \\ -0.85 \\ 1.7 \end{pmatrix} \\
 \text{lsolve}(A, b) &= \begin{pmatrix} -0.424 \\ -0.78 \\ 0.492 \\ -2.078 \end{pmatrix}
 \end{aligned}$$

Листинг 1.2 – Решение СЛАУ с помощью встроенной функции

Замечание – Использование блока `Given/Find` дает приближенное решение, поскольку в основе поиска решения лежат итерационные методы, в то время как в функции `lsolve` используется модифицированный метод последовательных исключений Гаусса.

2 СОБСТВЕННЫЕ ЗНАЧЕНИЯ И СОБСТВЕННЫЕ ВЕКТОРЫ МАТРИЦ

2.1 Основные определения и свойства собственных значений и векторов

Рассмотрим квадратную матрицу n -го порядка

$$A = \begin{pmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & a_{22} & \dots & a_{2n} \\ \dots & \dots & \dots & \dots \\ a_{n1} & a_{n2} & \dots & a_{nn} \end{pmatrix}.$$

Вектор $X = (x_1, x_2, \dots, x_n)$ называется *собственным вектором* матрицы A , соответствующим *собственному значению* λ , если он удовлетворяет системе уравнений

$$AX = \lambda X. \quad (2.1)$$

Например, число $\lambda = 2$ является собственным значением матрицы $A = \begin{pmatrix} 1 & 2 \\ 0 & 2 \end{pmatrix}$, поскольку

$$A \cdot \begin{pmatrix} 2 \\ 1 \end{pmatrix} = \begin{pmatrix} 4 \\ 2 \end{pmatrix} = 2 \cdot \begin{pmatrix} 2 \\ 1 \end{pmatrix}.$$

Поскольку при умножении собственного вектора на скаляр он остается собственным вектором той же матрицы, его можно нормировать. В частности, каждую координату собственного вектора можно разделить на максимальную из них или на длину вектора; в последнем случае получится единичный собственный вектор.

Характеристической матрицей C данной матрицы A называется матрица вида

$$C = A - \lambda E = \begin{pmatrix} a_{11} - \lambda & a_{12} & \dots & a_{1n} \\ a_{21} & a_{22} - \lambda & \dots & a_{2n} \\ \dots & \dots & \dots & \dots \\ a_{n1} & a_{n2} & \dots & a_{nn} - \lambda \end{pmatrix}, \quad (2.2)$$

1) прямые, основанные на раскрытии характеристических определителей

$$\det (A - \lambda E) = 0$$

и решении характеристических уравнений;

2) итерационные, основанные на многократном применении итерационного алгоритма, приближающего получаемое в каждом цикле собственное значение к точному;

3) преобразований подобия, использующих свойства подобных матриц (метод вращений, приведение к трехдиагональной матрице).

2.2 Прямые методы

Для нахождения собственных векторов матрицы требуется решить систему линейных алгебраических уравнений, решение которой не единственно. Из линейной алгебры известно, что в этом случае структура общего решения системы имеет следующий вид: одно или несколько неизвестных, называемых *свободными*, могут принимать любые значения, а остальные неизвестные выражаются через свободные. Число свободных неизвестных равно числу уравнений системы, являющихся следствием остальных уравнений. На практике, если свободное неизвестное одно (что часто и бывает), его полагают равным некоторому числу, например единице. После этого остальные неизвестные (компоненты вектора) находятся однозначно из подсистемы линейно независимых уравнений, в которой отброшено уравнение, являющееся следствием остальных. Эта процедура не влияет на результат решения задачи, поскольку, собственные векторы находятся с точностью до постоянного множителя.

Пример 2.1. Вычислить собственные числа и собственные векторы матрицы:

$$A = \begin{pmatrix} 3 & 1 \\ 2 & 4 \end{pmatrix}.$$

Решение. Составим характеристический многочлен:

$$\begin{vmatrix} 3-\lambda & 1 \\ 2 & 4-\lambda \end{vmatrix} = (3-\lambda)(4-\lambda) - 2 = \lambda^2 - 7\lambda + 10.$$

Найдем корни этого многочлена второй степени:

$$\begin{aligned} \lambda^2 - 7\lambda + 10 &= 0, \\ \lambda_1 &= 2, \quad \lambda_2 = 5. \end{aligned}$$

Для нахождения собственных векторов x_1, x_2 , соответствующих собственным значениям λ_1, λ_2 , составим системы уравнений типа (2.3) для каждого из них.

При $\lambda_1 = 2$ получим

$$\begin{pmatrix} 3-2 & 1 \\ 2 & 4-2 \end{pmatrix} \cdot \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix},$$

или, в координатной форме,

$$\begin{aligned} x_1 + x_2 &= 0; \\ 2x_1 + 2x_2 &= 0. \end{aligned}$$

Уравнения линейно зависимы. Поэтому оставляем лишь одно из них.

Из первого уравнения следует, что $x_2 = -x_1$.

Неизвестное x_1 можно считать свободным, полагаем $x_1 = 1$. Тогда $x_2 = -1$, и собственный вектор, соответствующий собственному значению $\lambda_1 = 2$, имеет вид $X_1 = (1; -1)$.

Аналогично находим второй собственный вектор, соответствующий собственному значению $\lambda_2 = 5$:

$$\begin{pmatrix} 3-5 & 1 \\ 2 & 4-5 \end{pmatrix} \cdot \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix};$$
$$\begin{aligned} -2x_1 + x_2 &= 0; \\ 2x_1 - 2x_2 &= 0. \end{aligned}$$

Отсюда $x_1 = 1, x_2 = 2$.

Второй собственный вектор, соответствующий собственному значению $\lambda_2 = 5$, имеет вид $X_2 = (1; 2)$.

2.3 Итерационные методы решения частичной проблемы собственных значений

Часто в практических вычислениях бывают нужны не все собственные значения, а лишь некоторые из них. В этих случаях нецелесообразно решать полную проблему собственных значений.

Для решения *частичной проблемы собственных значений*, состоящей в определении одного или нескольких собственных значений и соответствующих им собственных векторов, обычно используют итерационные методы. Строят такой итерационный процесс, который сходится к одному собственному значению и собственному вектору, причем используемые алгоритмы обычно весьма экономичны.

Построим итерационный процесс, применяя метод итераций к решению системы уравнений

$$AX = \lambda X. \quad (2.7)$$

Модифицируем рассмотренный ранее метод простой итерации. Запишем систему (2.7), введя вспомогательный вектор Y :

$$Y = AX; \quad (2.8)$$

$$\lambda X = Y. \quad (2.9)$$

Пусть $X^{(0)}$ – начальное приближение собственного вектора X , причем собственные векторы на каждой итерации нормированы, так что $|X^{(k)}| = 1$ ($k = 0, 1, \dots$). Используем соотношение (2.8) для вычисления $Y^{(1)}$:

$$Y^{(1)} = AX^{(0)}. \quad (2.10)$$

Соотношение (2.9) используем для вычисления первого приближения $\lambda^{(1)}$, применяя умножение обеих частей равенства скалярно на $X^{(0)}$:

$$\lambda^{(1)} X^{(0)} = Y^{(1)}, \quad \lambda^{(1)} X^{(0)} X^{(0)} = Y^{(1)} X^{(0)},$$

$$\lambda^{(1)} = \frac{Y^{(1)} X^{(0)}}{X^{(0)} X^{(0)}} = Y^{(1)} X^{(0)}.$$

Здесь учтено, что вектор $X^{(0)}$ нормирован, т. е. $X^{(0)} X^{(0)} = 1$. Следующее приближение собственного вектора $X^{(1)}$ можно вычислить, нормируя вектор $Y^{(1)}$.

Окончательно итерационный процесс записывается в виде

$$Y^{(k+1)} = AX^{(k)},$$

$$\lambda^{(k+1)} = Y^{(k+1)} X^{(k)}, \quad (2.11)$$

$$X^{(k+1)} = \frac{Y^{(k+1)}}{|Y^{(k+1)}|}, \quad k = 0, 1, \dots$$

и продолжается до установления постоянных значений λ и X . При этом нужно учесть, что применяя критерии завершения итераций, следует проверять близость векторов $(\text{sign } \lambda^{(k+1)}) X^{(k+1)}$ и $X^{(k+1)}$, а не $X^{(k+1)}$ и $X^{(k)}$.

Можно показать, что, найденное в результате итерационного процесса (2.11) число λ является наибольшим по модулю собственным значением данной матрицы A , а X – соответствующим ему собственным вектором.

Скорость сходимости этого итерационного процесса зависит от удачного выбора начального приближения. Если начальный вектор близок к истинному собственному вектору, то итерации сходятся быстро.

Для решения системы (2.7) можно использовать и другие итерационные методы. В частности, метод Ньютона дает лучшую сходимость, если удачно выбрано начальное приближение $X^{(0)}$. В этом случае бывает достаточно нескольких итераций.

В некоторых задачах нужно искать не наибольшее, а наименьшее по модулю собственное значение матрицы A . В этом случае можно умножить систему (2.7) на обратную матрицу A^{-1} :

$$\lambda A^{-1}X = A^{-1}AX^{(0)}.$$

Отсюда, деля обе части этого равенства на λ и учитывая, что $A^{-1}A = E$, получаем

$$\frac{1}{\lambda}X = A^{-1}X. \quad (2.12)$$

Следовательно, $1/\lambda$ является собственным значением обратной матрицы, и задача (2.12) отличается от ранее рассматриваемой тем, что здесь будет вычисляться наибольшее по модулю собственное значение $1/\lambda$ матрицы A^{-1} , что будет достигнуто при наименьшем по модулю λ . Заметим также, что процесс (2.11) может быть использован для нахождения наименьшего по модулю собственного значения обратной матрицы.

Преобразование подобия можно использовать для упрощения исходной матрицы, а задачу о вычислении ее собственных значений свести к аналогичной задаче для более простой матрицы. Самым лучшим упрощением матрицы A было бы приведение ее к треугольному виду. Тогда матрица (2.2) также имела бы треугольный вид.

Определитель треугольной матрицы равен произведению ее диагональных элементов, поэтому собственные значения треугольной матрицы равны ее диагональным элементам. То же самое относится и к диагональной матрице, которая является частным случаем треугольной.

Некоторые типы матриц удается привести к треугольному виду с помощью преобразования подобия. В частности, симметрическую матрицу можно привести к диагональному виду. На практике часто используется приведение симметрической матрицы к трехдиагональному виду. Процедура вычисления собственных значений для полученной матрицы значительно упрощается по сравнению с задачей для исходной матрицы.

Существует ряд методов, основанных на использовании преобразования подобия, позволяющего привести исходную матрицу к более простой структуре (например, метод вращений).

Пример 2.2. Используя метод итераций, определить первое (максимальное по модулю) и второе собственные значения матрицы и соответствующие им собственные векторы, имеющие первую норму, равную 1, с точностью до $\varepsilon = 0,001$:

$$A = \begin{pmatrix} 3,5 & 1,0 & 2,9 \\ 1,0 & 4,0 & 3,2 \\ 2,5 & 2,8 & 4,3 \end{pmatrix}.$$

Решение. Выберем произвольный ненулевой начальный вектор $Y^{(0)} = \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix}$.

Строим последовательность векторов $Y^{(i)} = AY^{(i-1)}$. Вычисления продолжаем до тех пор, пока расхождение между двумя последовательными приближениями собственных значений не будет превосходить величины ε , т. е.

$$\lambda_1 \approx \frac{y_i^{(k)}}{y_i^{(k-1)}} \quad \text{и} \quad \left| \frac{y_i^{(k)}}{y_i^{(k-1)}} - \frac{y_i^{(k-1)}}{y_i^{(k-2)}} \right| \leq \varepsilon.$$

Первая итерация:

$$Y^{(1)} = AY^{(0)} = \begin{pmatrix} 3,5 & 1,0 & 2,9 \\ 1,0 & 4,0 & 3,2 \\ 2,5 & 2,8 & 4,3 \end{pmatrix} \cdot \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix} = \begin{pmatrix} 3,5 \cdot 1 & 1,0 \cdot 1 & 2,9 \cdot 1 \\ 1,0 \cdot 1 & 4,0 \cdot 1 & 3,2 \cdot 1 \\ 2,5 \cdot 1 & 2,8 \cdot 1 & 4,3 \cdot 1 \end{pmatrix} = \begin{pmatrix} 5,1 \\ 4,4 \\ 6,5 \end{pmatrix}.$$

$$\frac{y_1^{(1)}}{y_1^{(0)}} = \frac{5,1}{1} = 5,1; \quad \frac{y_2^{(1)}}{y_2^{(0)}} = \frac{4,4}{1} = 4,4; \quad \frac{y_3^{(1)}}{y_3^{(0)}} = \frac{6,5}{1} = 6,5.$$

Вторая итерация:

$$Y^{(2)} = AY^{(1)} = \begin{pmatrix} 3,5 & 1,0 & 2,9 \\ 1,0 & 4,0 & 3,2 \\ 2,5 & 2,8 & 4,3 \end{pmatrix} \cdot \begin{pmatrix} 5,1 \\ 4,4 \\ 6,5 \end{pmatrix} = \begin{pmatrix} 3,5 \cdot 5,1 & 1,0 \cdot 4,4 & 2,9 \cdot 6,5 \\ 1,0 \cdot 5,1 & 4,0 \cdot 4,4 & 3,2 \cdot 6,5 \\ 2,5 \cdot 5,1 & 2,8 \cdot 4,4 & 4,3 \cdot 6,5 \end{pmatrix} = \begin{pmatrix} 26,08 \\ 24,12 \\ 37,42 \end{pmatrix}.$$

$$\frac{y_1^{(2)}}{y_1^{(1)}} = \frac{26,08}{5,1} = 5,113725; \quad \frac{y_2^{(2)}}{y_2^{(1)}} = \frac{24,12}{4,4} = 5,481818;$$

$$\frac{y_3^{(2)}}{y_3^{(1)}} = \frac{37,42}{6,5} = 5,756923.$$

Расчёты сведём в таблицу.

Таблица 2.1 – Результаты расчетов

A	1,6	2,3	1,2	$\frac{y_1^{(k)}}{y_1^{(k-1)}}$	$\frac{y_1^{(k)}}{y_1^{(k-1)}}$	$\frac{y_1^{(k)}}{y_1^{(k-1)}}$
	2,3	0,6	1,5			
	1,2	1,5	3,8			
$Y^{(0)}$	1	1	1			
$Y^{(1)}$	5,1	4,4	6,5	5,1	4,4	6,5
$Y^{(2)}$	26,08	24,12	37,42	5,113725	5,481818	5,756923
$Y^{(3)}$	142,108	130,586	209,672	5,448926	5,414013	5,603207
$Y^{(4)}$	779,327	719,708	1163,162	5,484047	5,511372	5,547532
$Y^{(5)}$	4298,046	3969,02	6434,771	5,515074	5,514765	5,532135
$Y^{(6)}$	23727,35	21919,07	35563,31	5,520496	5,52254	5,526742
$Y^{(7)}$	131053,6	121069,3	196492	5,523315	5,523468	5,525132
$Y^{(8)}$	723935,6	668802,9	1085538	5,523966	5,524132	5,524591
$Y^{(9)}$	3999189	3694641	5996971	5,524233	5,524259	5,524423

Для восьмой итерации выполняется соотношение $\left| \frac{y_i^{(8)}}{y_i^{(7)}} - \frac{y_i^{(7)}}{y_i^{(6)}} \right| \leq \varepsilon = 0,001$:

$$\left| \frac{y_1^{(8)}}{y_1^{(7)}} - \frac{y_1^{(7)}}{y_1^{(6)}} \right| = |5,523966 - 5,523315| = 0,000651 < \varepsilon = 0,001;$$

$$\left| \frac{y_2^{(8)}}{y_2^{(7)}} - \frac{y_2^{(7)}}{y_2^{(6)}} \right| = |5,524132 - 5,523468| = 0,000664 < \varepsilon = 0,001;$$

$$\left| \frac{y_3^{(8)}}{y_3^{(7)}} - \frac{y_3^{(7)}}{y_3^{(6)}} \right| = |5,524591 - 5,525132| = 0,00054 < \varepsilon = 0,001;$$

Собственное значение матрицы

$$\lambda_1 = \frac{5,523966 + 5,524132 + 5,524591}{3} = 5,52423 .$$

Ему соответствует собственный вектор $X_1 \approx Y^{(8)} = \begin{pmatrix} 723935,6 \\ 668802,9 \\ 1085538 \end{pmatrix}$.

Чтобы первая норма вектора была равна 1, разделим координаты вектора X_1 на наибольшую из них по абсолютной величине. Тогда окончательно

получим $X_1 = \begin{pmatrix} 0,667 \\ 0,616 \\ 0,1 \end{pmatrix}$.

Второе собственное значение λ_2 определяем из соотношения

$$\lambda_2 \approx \frac{y_i^{(k+1)} - \lambda_1 y_i^{(k)}}{y_i^{(k)} - \lambda_1 y_i^{(k-1)}},$$

где $y_i^{(k-1)}$, $y_i^{(k)}$, $y_i^{(k+1)}$ – одноимённые координаты трёх последовательных векторов; λ_1 – первое собственное значение.

$$\text{В нашем случае возьмём } \lambda_2 = \frac{y_i^{(9)} - \lambda_1 y_i^{(8)}}{y_i^{(8)} - \lambda_1 y_i^{(7)}}.$$

Результаты вычислений второго собственного значения представим в таблице.

Таблица 2.2 – Результаты расчетов

$y_i^{(7)}$	$y_i^{(8)}$	$y_i^{(9)}$	$y_i^{(9)} - \lambda_1 y_i^{(8)}$	$y_i^{(8)} - \lambda_1 y_i^{(7)}$	$\lambda_2 = \frac{y_i^{(9)} - \lambda_1 y_i^{(8)}}{y_i^{(8)} - \lambda_1 y_i^{(7)}}$
131053,6	723935,6	3999189	2,741342	-34,5769	-0,07928
121069,3	668802,9	3694641	19,81679	-11,764	-1,68452
196492	1085538	5996971	210,4135	70,93468	2,966299

Принимаем в качестве второго собственного значения

$$\lambda_2 \approx \frac{-0,07928 - 1,68452 + 2,966299}{3} = 0,400831.$$

Собственный вектор X_2 определяется из равенства $X_2 \approx Y^{(k)} - \lambda_1 Y^{(k)}$. Таким образом, $X_2 \approx Y^{(9)} - \lambda_1 Y^{(8)} = \begin{pmatrix} 2,741342 \\ 19,81679 \\ 210,4135 \end{pmatrix}$. Разделив координаты вектора на

наибольшую из них по абсолютной величине, получим $X_2 = \begin{pmatrix} 0,013 \\ 0,094 \\ 1 \end{pmatrix}$.

Таким, образом, первое собственное значение матрицы $A - \lambda_1 = 5,524$, соответствующий ему вектор – $X_1 = \begin{pmatrix} 0,667 \\ 0,616 \\ 0,1 \end{pmatrix}$. Второе собственное значение –

$\lambda_2 = 0,401$, соответствующий ему вектор – $X_2 = \begin{pmatrix} 0,013 \\ 0,094 \\ 1 \end{pmatrix}$.

Алгоритм определения собственных значений и собственных векторов матрицы в MathCAD приведен в листинге 2.2.

2.4 Нахождение собственных значений и собственных векторов матриц в MathCAD

Для решения задач на собственные векторы и собственные значения в MathCAD встроены функции:

□ `eigenvals(A)` – вычисляет вектор, элементами которого являются собственные значения матрицы A;

□ `eigenvecs(A)` – вычисляет матрицу, содержащую нормированные собственные векторы, соответствующие собственным значениям матрицы A. Причем n-й столбец вычисляемой матрицы соответствует собственному вектору n-го собственного значения, вычисляемого `eigenvals`;

□ `eigenvec(A, λ)` – вычисляет собственный вектор для матрицы A и заданного собственного значения λ, где

- A – квадратная матрица,
- λ – собственное значение матрицы A.

Применение этих функций иллюстрирует следующий пример.

1. Исходные данные:

$$A := \begin{pmatrix} 1.6 & 2.3 & 1.2 \\ 2.3 & 0.6 & 1.5 \\ 1.2 & 1.5 & 3.8 \end{pmatrix} \quad Y_0 := \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix} \quad j := 0..2 \quad i := 0..14$$

2. Результаты:

$$\text{eigenvals}(A) = \begin{pmatrix} -1.293 \\ 1.769 \\ 5.524 \end{pmatrix} \quad \text{eigenvecs}(A) = \begin{pmatrix} 0.595 & -0.634 & 0.494 \\ -0.798 & -0.394 & 0.456 \\ 0.095 & 0.665 & 0.74 \end{pmatrix}$$

$$S := \text{eigenvec}(A, \lambda_1) \quad S = \begin{pmatrix} 0.494 \\ 0.456 \\ 0.74 \end{pmatrix} \quad X_{11} := \frac{S}{\max(|S|)} \quad X_{11} = \begin{pmatrix} 0.667 \\ 0.616 \\ 1 \end{pmatrix}$$

Проверка: $\lambda_1 \cdot X_{11} = A \cdot X_{11}$

$$c01 := A \cdot X_1 \quad c02 := \lambda_1 \cdot X_1 \quad c01 = \begin{pmatrix} 3.684 \\ 3.403 \\ 5.524 \end{pmatrix} \quad c02 = \begin{pmatrix} 3.684 \\ 3.403 \\ 5.524 \end{pmatrix}$$

Листинг 2.1 – Определение собственных значений и собственных векторов матриц с использованием встроенных функций

1 Άλφα ενόιαίρε έ ανήπιααοάεύίρε ιαοδεού :

$$A := \begin{pmatrix} 1.6 & 2.3 & 1.2 \\ 2.3 & 0.6 & 1.5 \\ 1.2 & 1.5 & 3.8 \end{pmatrix} \quad Y_0 := \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix} \quad j := 0..2 \quad i := 0..14$$

2 Άν-εñέáíεγ :

$$Y_{i+1} := A \cdot Y_i \quad \lambda_{i,j} := \frac{(Y_{i+1})_j}{(Y_i)_j}$$

$$\lambda = \begin{pmatrix} 5.1 & 4.4 & 6.5 \\ 5.113725 & 5.481818 & 5.756923 \\ 5.448926 & 5.414013 & 5.603207 \\ 5.484047 & 5.511372 & 5.547532 \\ 5.515074 & 5.514765 & 5.532135 \\ 5.520496 & 5.52254 & 5.526742 \\ 5.523315 & 5.523468 & 5.525132 \\ 5.523966 & 5.524132 & 5.524591 \\ 5.524233 & 5.524259 & 5.524423 \\ 5.524305 & 5.52432 & 5.524368 \\ 5.524331 & 5.524335 & 5.524351 \\ 5.524339 & 5.52434 & 5.524346 \\ 5.524342 & 5.524342 & 5.524344 \\ 5.524343 & 5.524343 & 5.524343 \\ 5.524343 & 5.524343 & 5.524343 \end{pmatrix}$$

3 Ιδñίπδ δάόεүүοαόιά :

$$\lambda_1 := \lambda_{14,0} \quad Y_{\max} := \max(\overrightarrow{Y_{15}}) \quad X_1 := \frac{Y_{15}}{Y_{\max}}$$

$$\lambda_1 = 5.524343 \quad X_1 = \begin{pmatrix} 0.667 \\ 0.616 \\ 1 \end{pmatrix}$$

Листинг 2.2 – Определение собственных значений и собственных векторов матриц непосредственным расчетом

3 АППРОКСИМАЦИЯ ФУНКЦИЙ

Задачи об аппроксимации функций обобщенными полиномами широко используются при решении практических задач гидравлики и водоснабжения. Например, при расчетах критической и нормальной глубины при построении кривой свободной поверхности, при моделировании работы скважинных водозаборов и процессов фильтрации и т. п.

3.1 Понятие о приближении функций обобщёнными полиномами

3.1.1 Постановка задачи

Пусть величина y является функцией аргумента x . Это означает, что любому значению x из области определения поставлено в соответствие значение y . Вместе с тем на практике часто неизвестна явная связь между y и x , т. е. невозможно записать эту связь в виде некоторой зависимости $y = f(x)$. Иногда даже известная зависимость $y = f(x)$ оказывается настолько громоздкой (например, содержит трудно вычисляемые выражения, сложные интегралы и т. п.), что ее использование в практических расчетах требует слишком много времени.

Наиболее распространенным и практически важным случаем, когда вид связи между параметрами x и y неизвестен, является задание этой связи в виде некоторой таблицы $\{x_i, y_i\}$. Это означает, что дискретному множеству значений аргумента x_i поставлено в соответствие множество значений функции $\{y_i\}$ ($i = 0, 1, \dots, n$). Эти значения – либо результаты расчетов, либо экспериментальные данные. На практике могут понадобиться значения величины y и в других точках, отличных от узлов x_i . Однако получить эти значения можно лишь путем очень 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 $y = f(x)$ не известна (либо нам нецелесообразно ею пользоваться).

Этой цели и служит задача о приближении (*аппроксимации*) функций: данную 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 было наименьшим. Функция $\varphi(x)$ при этом называется *аппроксимирующей*.

Аппроксимация рассмотренного выше типа, при которой приближение строится на заданном дискретном множестве точек $\{x_i\}$, называется *точечной*. К ней относятся интерполирование, среднеквадратичное приближение и др.

При построении приближения на непрерывном множестве точек (например, на отрезке) аппроксимация называется *непрерывной* (или *интегральной*). К непрерывной аппроксимации относится, например, равномерное приближение.

Для практики весьма важен случай аппроксимации функции $f(x)$ многочленом:

$$\varphi(x) = P_m(x) = a_0 + a_1 x + a_2 x^2 + \dots + a_m x^m. \quad (3.1)$$

В дальнейшем будем чаще всего рассматривать аппроксимацию такого рода. При этом коэффициенты a_j будем подбирать так, чтобы достичь наименьшего отклонения многочлена от данной функции.

3.1.2 Точечная аппроксимация (интерполирование)

Одним из основных типов *точечной аппроксимации* является *интерполирование*. Оно состоит в следующем: для данной функции $y = f(x)$ строим *интерполирующую функцию* $\varphi(x)$ [например, многочлен (3.1)], принимающую в заданных точках x_i те же значения y_i , что и функция $f(x)$, т. е.

$$\varphi(x_i) = y_i, \quad (i = 0, 1, \dots, n). \quad (3.2)$$

При этом предполагается, что среди значений x_i нет одинаковых, т. е. $x_i \neq x_k$ при $i \neq k$.

Точки x_i называются *узлами интерполяции*.

Таким образом, близость интерполирующей функции (рисунок 3.1) к заданной функции состоит в том, что их значения совпадают на заданной системе точек.

Определитель такой системы в линейной алгебре называется *определителем Вандермонда*. Можно показать, что определитель Вандермонда отличен от нуля, если $x_i \neq x_k$, т. е. если среди узлов интерполяции нет совпадающих. Следовательно, в этом случае система (3.4) имеет единственное решение. Решив систему, можно построить интерполяционный многочлен. Такой метод построения интерполяционного многочлена называется *методом неопределенных коэффициентов*. Заметим вместе с тем, что этот метод требует значительного объема вычислений, особенно при большом числе узлов. Существуют более простые алгоритмы построения интерполяционных многочленов.

Пример 3.1. Для различных условий фильтрации воды в крупнопористых и дисперсных средах установлены зависимости коэффициента фильтрации K_f от размеров и формы материалов, механического состава и прозрачности среды, гидравлических градиентов.

Данные об изменении коэффициента фильтрации воды K_f , м/с, в зависимости от диаметра частиц материала d , см, заданы таблицей:

d , см	5	10	15
K_f , м/с	0,2	0,3	0,33

Построить интерполяционный полином второй степени, описывающий зависимость коэффициента фильтрации K_f от диаметра частиц материала d , методом неопределенных коэффициентов.

Решение. Составим систему уравнений:

$$\begin{cases} a_0 + a_1 \cdot 5 + a_2 \cdot 5^2 = 0,2; \\ a_0 + a_1 \cdot 10 + a_2 \cdot 10^2 = 0,3; \\ a_0 + a_1 \cdot 15 + a_2 \cdot 15^2 = 0,33. \end{cases}$$

Для решения этой системы можно воспользоваться любым известным методом (Крамера, обратной матрицы, Гаусса).

Решим систему с помощью MathCAD (листинг 3.1):

$$a_0 := 0 \quad a_1 := 0 \quad a_2 := 0$$

Given

$$a_0 + 5 \cdot a_1 + 5^2 \cdot a_2 = 0.2$$

$$a_0 + 10 \cdot a_1 + 10^2 \cdot a_2 = 0.3$$

$$a_0 + 15 \cdot a_1 + 15^2 \cdot a_2 = 0.33$$

$$\text{Find}(a_0, a_1, a_2) = \begin{pmatrix} 0.03 \\ 0.041 \\ -1.4 \times 10^{-3} \end{pmatrix}$$

Листинг 3.1 – Решение системы уравнений

Получим следующее решение: $a_0 = 0,03$; $a_1 = 0,041$; $a_2 = -0,0014$.

Интерполяционный полином, описывающий зависимость коэффициента фильтрации K_{Φ} от диаметра частиц материала d , имеет вид

$$y(x) = -0,0014x^2 + 0,041x + 0,03.$$

3.1.3 Интерполяционный многочлен Лагранжа

Рассмотрим случай глобальной интерполяции, т. е. построение интерполяционного многочлена, единого для всего отрезка $[x_0, x_n]$. Для построения такого многочлена достаточно решить систему (3.4), однако этот путь неэффективен.

Будем искать интерполяционный многочлен в виде линейной комбинации многочленов степени n :

$$L(x) = y_0 l_0(x) + y_1 l_1(x) + \dots + y_n l_n(x). \quad (3.5)$$

При этом потребуем, чтобы каждый многочлен $y_i l_i(x)$ обращался в нуль во всех узлах интерполяции, за исключением одного (i -го), где он должен равняться единице. Легко проверить, что этим условиям отвечает многочлен вида

$$l_i(x) = \frac{(x - x_0) \cdots (x - x_{i-1})(x - x_{i+1}) \cdots (x - x_n)}{(x_i - x_0) \cdots (x_i - x_{i-1})(x_i - x_{i+1}) \cdots (x_i - x_n)}. \quad (3.6)$$

Подставив в (3.5) выражения (3.6), находим:

$$L(x) = \sum_{i=0}^n y_i \frac{(x-x_0) \cdots (x-x_{i-1})(x-x_{i+1}) \cdots (x-x_n)}{(x_i-x_0) \cdots (x_i-x_{i-1})(x_i-x_{i+1}) \cdots (x_i-x_n)}. \quad (3.7)$$

Эта формула определяет *интерполяционный многочлен Лагранжа*.

Единственность представления многочлена (3.5) в виде (3.7) следует из единственности решения системы (3.4).

Из формулы (3.7) можно получить выражения для линейной ($n = 1$) и квадратичной ($n = 2$) интерполяций:

$$L(x) = \frac{(x-x_1)}{(x_0-x_1)} y_0 + \frac{(x-x_0)}{(x_1-x_0)} y_1, \quad n = 1;$$

$$L(x) = \frac{(x-x_1)(x-x_2)}{(x_0-x_1)(x_0-x_2)} y_0 + \frac{(x-x_0)(x-x_2)}{(x_1-x_0)(x_1-x_2)} y_1 + \frac{(x-x_0)(x-x_1)}{(x_2-x_0)(x_2-x_1)} y_2, \quad n = 2.$$

Существует несколько обобщений интерполяционного многочлена Лагранжа. Например, довольно широко используются *интерполяционные многочлены Эрмита*. Здесь наряду со значениями функции y_i в узлах x_i задаются значения ее производной y_i' .

Пример 3.2. В крупнопористой каменной наброске плотин, в фильтрующих дамбах и водоприемниках, насыпях, водоочистных префильтрах нарушается линейная зависимость между скоростями фильтрации воды v_ϕ и градиентами напора J . Данные об изменении скорости фильтрации воды v_ϕ , см/с, от градиентов напора J заданы таблицей:

J	0,2	1	2
v_ϕ , см/с	5	18	24

Построить интерполяционный полином второй степени, описывающий зависимость скорости фильтрации воды v_ϕ , см/с, от градиентов напора J .

Решение. Решим задачу, пользуясь интерполяционной формулой Лагранжа.

Подставляя в формулу (для $n = 2$) данные, получим

$$L(x) = \frac{(x-1)(x-2)}{(0,2-1)(0,2-2)} \cdot 5 + \frac{(x-0,2)(x-2)}{(1-0,2)(1-2)} \cdot 18 + \frac{(x-0,2)(x-1)}{(2-0,2)(2-1)} \cdot 24 = -5,69x^2 + 23,08x + 0,61.$$

Интерполяционный полином второй степени, полученный по формуле Лагранжа, описывающий зависимость скорости фильтрации воды v_ϕ , см/с, от градиентов напора J , имеет вид

$$y(x) = -5,69x^2 + 23,08x + 0,61.$$

3.1.4 Интерполяционный многочлен Ньютона

До сих пор не делалось никаких предположений о законе распределения узлов интерполяции. Теперь рассмотрим случай равноотстоящих значений аргумента, т. е. $x_i - x_{i-1} = h = \text{const}$ ($i = 1, 2, \dots, n$). Величина h называется *шагом*.

Введем также понятие *конечных разностей*. Пусть известны значения функции в узлах x_i : $y_i = y = f(x_i)$.

Составим разности значений функции:

$$\Delta y_0 = y_1 - y_0, \quad \Delta y_1 = y_2 - y_1, \quad \dots, \quad \Delta y_n = y_n - y_{n-1}. \quad (3.8)$$

Эти значения называются *первыми разностями* (или *разностями первого порядка*) функции.

Можно составить *вторые разности* функции:

$$\Delta^2 y_0 = \Delta y_1 - \Delta y_0, \quad \Delta^2 y_1 = \Delta y_2 - \Delta y_1, \quad \dots$$

Аналогично составляются разности k -го порядка:

$$\Delta^k y_i = \Delta^{k-1} y_{i+1} - \Delta^{k-1} y_i \quad (i = 1, 2, \dots, n-1).$$

В случае равноотстоящих узлов интерполяции ($x_1 = x_0 + h$, $x_2 = x_0 + 2h$, $x_n = x_0 + nh$) интерполяционный многочлен Ньютона для «интерполирования вперед»

$$N(x) = y_0 + \frac{\Delta y_0}{h}(x-x_0) + \frac{\Delta^2 y_0}{2!h^2}(x-x_0)(x-x_1) + \frac{\Delta^3 y_0}{3!h^3}(x-x_0)(x-x_1)(x-x_2) + \dots + \frac{\Delta^n y_0}{n!h^n}(x-x_0)(x-x_1)\dots(x-x_{n-1}). \quad (3.9)$$

Формула удобна при интерполировании функций для значений x , близких к наименьшему узлу x_0 .

Интерполяционный многочлен Ньютона для «интерполирования назад»:

$$f(x) = y_n + \frac{\Delta y_{n-1}}{h}(x-x_n) + \frac{\Delta^2 y_{n-2}}{2!h^2}(x-x_n)(x-x_{n-1}) + \frac{\Delta^3 y_{n-1}}{3!h^3}(x-x_n)(x-x_{n-1})(x-x_{n-2}) + \dots + \frac{\Delta^n y_{n-1}}{n!h^n}(x-x_n)(x-x_{n-1}) \dots (x-x_1).$$

Формула удобна при интерполировании функций для значений x , близких к наибольшему узлу x_n .

Пример 3.3. При биологической очистке воды в биореакторе с трехслойной капроновой насадкой важным показателем работы биореактора является эффективность снижения перманганатной окисляемости (ПО), характеризующей наличие легкоокисляемой органики.

Данные об изменении перманганатной окисляемости в зависимости от времени фильтрации заданы таблицей:

Время фильтрации, t	2	4	6	8
Перманганатная окисляемость	15	22	20	10

Для исследования эффективности очистки воды построить интерполяционный многочлен Ньютона.

Решение. Интерполяционный многочлен Ньютона, используемый в случае равноотстоящих узлов, для данной задачи имеет вид

$$N(x) = y_0 + \frac{\Delta y_0}{h}(x - x_0) + \frac{\Delta^2 y_0}{2!h^2}(x - x_0)(x - x_1) + \frac{\Delta^3 y_0}{3!h^3}(x - x_0)(x - x_1)(x - x_2) \cdot$$

Шаг $h = 2$.

Составим таблицу конечных разностей.

Таблица 3.1 – Таблица конечных разностей

i	x_i	y_i	Δy_i	$\Delta^2 y_i$	$\Delta^3 y_i$
0	2	16	$22 - 16 = 7$	$2 - 7 = -9$	$-8 - (-9) = 1$
1	4	22	$20 - 22 = -2$	$10 - (-2) = -8$	
2	6	20	$10 - 20 = -10$		
3	8	10			

Подставив рассчитанные значения в формулу, получаем

$$N(x) = 16 + \frac{7}{2}(x - 2) + \frac{-9}{2! \cdot 2^2}(x - 2)(x - 4) + \frac{1}{3! \cdot 2^3}(x - 2)(x - 4)(x - 6) =$$

$$= 0,0208 x^3 - 1,375 x^2 + 11,167 x - 2.$$

Интерполяционный полином второй степени, полученный по формуле Ньютона, описывающий зависимость перманганатной окисляемости от времени фильтрации, имеет вид

$$y(x) = 0,0208x^3 - 1,375x^2 + 11,167x - 2.$$

Рассмотрели построение интерполяционного многочлена Ньютона для равноотстоящих узлов. Можно построить многочлен Ньютона и для произвольно расположенных узлов, как и в случае многочлена Лагранжа.

Разные способы построения многочленов Лагранжа и Ньютона дают тождественные интерполяционные формулы при заданной таблице значений

функции. Это следует из единственности интерполяционного многочлена заданной степени (при отсутствии совпадающих узлов интерполяции). Многочлен Ньютона удобно применять, например, если количество узлов интерполяции постепенно увеличивается. При этом учет нового узла требует лишь вычисления одного дополнительного слагаемого, в то время как при использовании многочлена Лагранжа требуется пересчитывать все слагаемые.

3.1.5 Сплайны. Аппроксимация данных кубическими сплайнами

Интерполирование многочленом Лагранжа или Ньютона на всем отрезке $[a, b]$ с использованием большого числа узлов интерполяции часто приводит к плохому приближению, что объясняется сильным накоплением погрешностей в процессе вычислений. Кроме того, из-за расходимости процесса интерполяции увеличение числа узлов не обязательно приводит к повышению точности.

Для того чтобы избежать больших погрешностей, весь отрезок $[a, b]$ разбивают на частичные отрезки, и на каждом из частичных отрезков приближенно заменяют функцию $f(x)$ многочленом невысокой степени (так называемая *кусочно-полиномиальная интерполяция*).

Одним из способов интерполирования на всем отрезке является интерполирование с помощью сплайн-функций.

Сплайн-функцией или *сплайном* называется кусочно-полиномиальная функция, определенная на отрезке $[a, b]$ и имеющая на этом отрезке некоторое число непрерывных производных.

Слово «сплайн» (английское spline) означает гибкую линейку, используемую для проведения гладких кривых через заданные точки плоскости.

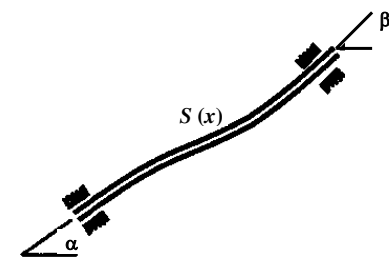


Рисунок 3.2 – Механический сплайн

Механическим аналогом кубического сплайна является гибкий тонкий стержень из упругого материала. Если закрепить его в двух соседних узлах интерполяции с заданными углами наклонов α и β (рисунок 3.2), то между точками закрепления этот стержень (механический сплайн) примет некоторую форму, минимизирующую его

потенциальную энергию.

Преимуществом сплайнов перед обычной интерполяцией является, во-первых, их сходимостъ и, во-вторых, устойчивостъ процесса вычислений.

Чаще всего в вычислительной практике используется случай, когда сплайн определяется с помощью многочленов третьей степени (кубический сплайн).

Построение кубического сплайна. Пусть на отрезке $[a, b]$ задана непрерывная функция $f(x)$. Разобьем отрезок $[a, b]$ точками $a = x_0 < x_1 < \dots < x_n = b$ на n частичных отрезков $[x_{i-1}, x_i]$ и обозначим $f_i = f(x_i)$, $(i = 0, 1, \dots, n)$.

Кубическим сплайном, соответствующим данной функции $f(x)$, называется функция $S(x)$, удовлетворяющая следующим условиям:

1) функция $S(x)$ непрерывна на отрезке $[a, b]$ вместе со всеми своими производными до некоторого порядка $p \leq 3$;

2) на каждом частичном отрезке $[x_{i-1}, x_i]$, $(i = 1, 2, \dots, n)$ функция $S(x)$ является многочленом третьей степени;

3) $S(x_i) = f(x_i)$, $(i = 0, 1, \dots, n)$.

Последнее условие называется *условием интерполирования*, а сплайн, определяемый условиями (1)–(3), называется также *интерполяционным кубическим сплайном*.

Доказано существование и единственность сплайна, определяемого перечисленными выше условиями.

На каждом из отрезков $[x_{i-1}, x_i]$ $(i = 1, 2, \dots, n)$ будем искать функцию $S(x) = S_i(x)$ в виде многочлена третьей степени

$$S_i(x) = a_i + b_i(x - x_i) + \frac{c_i}{2}(x - x_i)^2 + \frac{d_i}{6}(x - x_i)^3, \quad (3.10)$$
$$x_{i-1} \leq x \leq x_i, \quad i = \overline{1, n},$$

где a_i, b_i, c_i и d_i – коэффициенты, подлежащие определению.

Для определения коэффициентов a_i, b_i, c_i и d_i на всех n элементарных отрезках необходимо получить $4n$ уравнений. Полученная система линейных уравнений с трехдиагональной матрицей решается, например, методом прогонки. При этом необходимо задать граничные условия в крайних точках, которые существенно влияют на вид получаемого сплайна.

3.1.6 Построение интерполяционных полиномов в MathCAD

Для построения интерполяционного *многочлена Лагранжа* в MathCAD может быть использована следующая программа:

```

lagran(X, Y, x) :=
  w ← length(X)
  n ← w - 1
  for k ∈ 0.. n
    
$$\text{Lag}_{n,k} \leftarrow \frac{\prod_{j=0}^n \text{if}(j \neq k, x - X_j, 1)}{\prod_{j=0}^n \text{if}(j \neq k, X_k - X_j, 1)}$$

  P ←  $\sum_{k=0}^n Y_k \cdot \text{Lag}_{n,k}$ 
  P

```

Листинг 3.2 – Программа построения интерполяционного многочлена Лагранжа

Пример использования программы приведен в листинге 3.7.

Программа построения интерполяционного *многочлена Ньютона* в MathCAD (листинг 3.3):

```

new(X, Y) :=
  n ← length(X)
  D(0) ← Y
  for j ∈ 1.. n - 1
    for k ∈ j.. n - 1
      
$$D_{k,j} \leftarrow \frac{D_{k,j-1} - D_{k-1,j-1}}{X_k - X_{k-j}}$$

  D

```

$$D := \text{new}(X, Y) \quad P_{\text{new}}(x, D, X, n) := D_{0,0} + \sum_{k=1}^n D_{k,k} \cdot \prod_{j=0}^{k-1} (x - X_j)$$

Листинг 3.3 – Программа построения интерполяционного многочлена Ньютона

Рассмотрим *интерполирование полиномом N -й степени* (т. е. подстановкой значений табличной зависимости в многочлен N -й степени) на примере 3.3:

1 Ααία έπώιαίε είοίαοέε άεγ είοάοίπéεοίαίεγ :

$$N := 3 \quad i := 0..N \quad X := (2 \ 4 \ 6 \ 8)^T \quad Y := (15 \ 22 \ 20 \ 10)^T$$

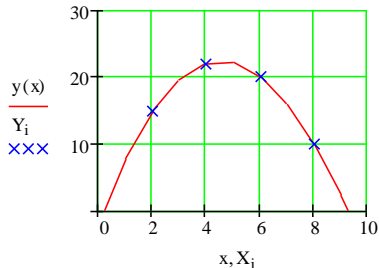
2 Είοάοίπéεοίαίεά πéέίπí N-έ πώιαίε :

$$k := 0..N \quad m := 0..N \quad A_{k,m} := (X_k)^m \quad B_k := Y_k \quad a := A^{-1} \cdot B$$

$$y(x) := \sum_{k=0}^N a_k \cdot x^k \quad x := 0,01..10$$

$$A = \begin{pmatrix} 1 & 2 & 4 & 8 \\ 1 & 4 & 16 & 64 \\ 1 & 6 & 36 & 216 \\ 1 & 8 & 64 & 512 \end{pmatrix} \quad B = \begin{pmatrix} 15 \\ 22 \\ 20 \\ 10 \end{pmatrix} \quad a = \begin{pmatrix} -2 \\ 11.166667 \\ -1.375 \\ 0.020833 \end{pmatrix}$$

Άοάοέε άαέπéηίπéε ίί τó άοάίάίε πώοίε



Листинг 3.4 – Интерполирование полиномом N -й степени

Для *линейной интерполяции* в MathCAD служит встроенная функция

□ $\text{linterp}(x, y, t)$ – функция, аппроксимирующая данные векторов x и y кусочно-линейной зависимостью, где

- x – вектор действительных данных аргумента, элементы которого расположены в порядке возрастания;
- y – вектор действительных данных значений того же размера;
- t – значение аргумента, при котором вычисляется интерполирующая функция.

Рассмотрим построение *сплайнов* в MathCAD с помощью имеющихся встроенных функций.

Для построения интерполяционных полиномов используется функция

□ $\text{interp}(s, x, y, t)$ – функция, аппроксимирующая данные векторов x и y кубическими сплайнами, где

- s – вектор вторых производных, созданный одной из сопутствующих функций cspline , pspline или lspline ;
- x – вектор действительных данных аргумента, элементы которого расположены в порядке возрастания;
- y – вектор действительных данных значений того же размера;
- t – значение аргумента, при котором вычисляется интерполирующая функция.

Перед применением функции `interp` необходимо предварительно определить первый из ее аргументов – векторную переменную `s`. Делается это при помощи одной из трех встроженных функций тех же аргументов (x, y) .

□ `lspline(x, y)` – вектор значений коэффициентов линейного сплайна;

□ `pspline(x, y)` – вектор значений коэффициентов квадратичного сплайна;

□ `cspline(x, y)` – вектор значений коэффициентов кубического сплайна, где

- (x, y) – векторы данных.

Выбор конкретной функции сплайновых коэффициентов влияет на интерполяцию вблизи конечных точек интервала.

Пример сплайн-интерполяции приведен в листингах 3.5 и 3.8.

3.1.7 Определение критической глубины при построении кривой свободной поверхности потока с использованием MathCAD

В трапециевидном призматическом русле построено искусственное сооружение, которое нарушило равномерное движение воды в русле и создало в нем подпор воды.

Для построения кривой свободной поверхности необходимо определить критическую глубину в русле трапециевидного поперечного сечения, если

- расчетный расход $Q = 34 \text{ м}^3/\text{с}$,
- ширина русла по дну $b = 8 \text{ м}$,
- коэффициент заложения откосов $m = 1,2$.

В общем случае критическая глубина определяется из основного уравнения критического состояния потока

$$\frac{\omega_k^3}{B_k^3} = \frac{\alpha Q^2}{g},$$

где ω_k – площадь сечения потока при глубине, равной критической;

B_k – ширина русла по свободной поверхности потока при той же глубине;

α – коэффициент кинетической энергии, обычно принимаемый равным 1,1;

g – ускорение силы тяжести.

Задаваясь значениями глубины $h = 0; 1,0; 1,2; 1,4; 1,6; 1,8$ м, вычисляем соответствующие им площади поперечного сечения, ширины русла по свободной поверхности и определяем соотношения ω^3 / B .

Результаты расчетов сведем в таблицу.

Таблица 3.4 – Результаты расчетов

h , м	$\omega = (b + m h) h$, м ²	ω^3 , м ⁶	$B = b + 2 m h$, м	ω^3 / B , м ⁵
0	0,00	0,00	8,0	0,00
1	9,20	778,69	10,4	74,87
1,2	11,33	1453,65	10,9	133,61
1,4	13,55	2488,92	11,4	219,09
1,6	15,87	3998,48	11,8	337,71
1,8	18,29	6116,44	12,3	496,46
Контрольная критическая глубина h_k				
1,19	11,22	1412,21	10,9	130,09

По данным таблицы строим график зависимости $\omega^3 / B = f(h)$ (листинг 3.5).

Зная величину $\frac{\alpha Q^2}{g} = \frac{1,1 \cdot 34}{9,81} = 130 \text{ м}^5$, находим критическую глубину

$h_k = 1,19$.

Результаты контроля критической глубины $h_k = 1,19$ помещены в таблице 3.4.

1 Άϊϊά έπϊοϊϊϊέ έρϊοδϊαοέè :

N := 5 i := 0.. N

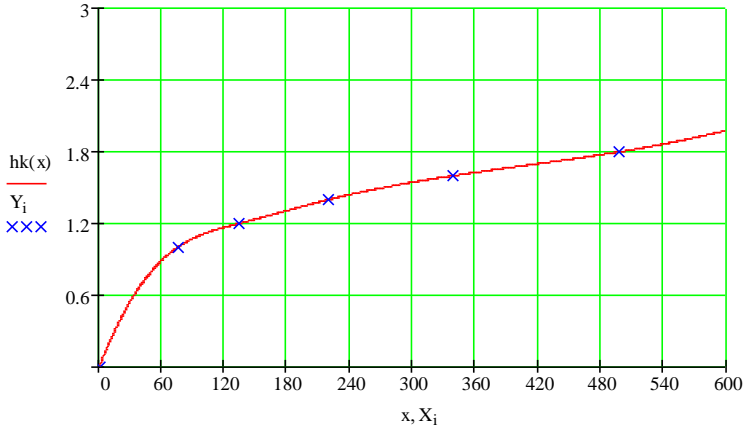
X := (0 74.87 133.61 219.09 337.71 496.46)^T

Y := (0 1 1.2 1.4 1.6 1.8)^T

2 Έϊοάοϊϊέèδϊϊάϊέϊά ñïèàέϊï :

KS := cspline(X, Y) hk(x) := interp(KS, X, Y, x)

3 Άδϊαè÷άñέϊϊά ïδϊάñοϊάέέϊέϊά δάçοέïïοϊά ω³/B = f(h) :



4 ïαοϊæääïέϊά έδèèè÷άñέϊϊά æèοάέïï:

hk(130) = 1.19

Листинг 3.5 – Определение критической глубины

3.2 Подбор эмпирических формул

3.2.1 Характер опытных данных. Эмпирические формулы

При интерполировании функций используется условие равенства значений интерполяционного многочлена и данной функции в известных точках – узлах интерполяции. Это предъявляет высокие требования к точности данных значений функции. В случае обработки опытных данных, полученных в результате наблюдений или измерений, нужно иметь в виду *ошибки* этих данных. Они могут быть вызваны несовершенством измерительного прибора, субъективными причинами, различными случайными факторами и т. д. Ошибки экспериментальных данных можно

условно разбить на три категории по их происхождению и величине: систематические, случайные и грубые.

Таким образом, в экспериментальных данных всегда имеются случайные ошибки. Они, вообще говоря, могут быть уменьшены до сколь угодно малой величины путем многократного повторения опыта. Однако это не всегда целесообразно, поскольку могут потребоваться большие материальные или временные ресурсы. Значительно дешевле и быстрее можно в ряде случаев получить уточненные данные хорошей математической обработкой имеющихся результатов измерений.

В частности, с помощью статистической обработки результатов измерений можно найти закон распределения ошибок измерений, наиболее вероятный диапазон изменения искомой величины (доверительный интервал) и другие параметры. Рассмотрение этих вопросов выходит за рамки данного пособия; их изложение можно найти в некоторых книгах, приведенных в списке литературы. Здесь ограничимся лишь определением связи между исходным параметром x и искомой величиной y на основании результатов измерений.

Эмпирические формулы. Пусть, изучая неизвестную функциональную зависимость между y и x , мы в результате серии экспериментов произвели ряд измерений этих величин и получили таблицу значений

x_0	x_1	...	x_n
y_0	y_1	...	y_n

Задача состоит в том, чтобы найти приближенную зависимость

$$y = f(x), \quad (3.11)$$

значения которой при $x = x_i$, ($i = 0, 1, \dots, n$) мало отличаются от опытных данных y_i . Приближенная функциональная зависимость (3.11), полученная на основании экспериментальных данных, называется *эмпирической формулой*.

Задача построения эмпирической формулы отличается от задачи интерполирования. График эмпирической зависимости не проходит через заданные точки (x_i, y_i) , как в случае интерполяции. Это приводит к тому, что экспериментальные данные в некоторой степени сглаживаются, в то время как интерполяционная формула повторила бы все ошибки, имеющиеся в экспериментальных данных.

Построение эмпирической формулы состоит из двух этапов:

- 1) подбор общего вида этой формулы;
- 2) определение наилучших значений содержащихся в ней параметров.

По экспериментальным точкам строится *диаграмма рассеивания*. Если последовательность точек диаграммы рассеивания группируется в виде некоторой линии, то можно сделать предположение о наличии связи. Затем

проводят выбор формы связи путем сравнения внешнего вида диаграммы рассеивания с имеющимися математическими моделями. (Линия, которая описывает диаграмму рассеивания, называется линией регрессии, а описывающее ее уравнение – уравнением регрессии. Процесс нахождения теоретической линии регрессии называют выравниванием эмпирической линии регрессии.)

Простейшей эмпирической формулой является линейная зависимость

$$y = ax + b. \quad (3.12)$$

Близость экспериментального распределения точек к линейной зависимости легко просматривается после построения графика данной экспериментальной зависимости. Кроме того, эту зависимость можно проверить путем вычисления значений k_i :

$$k_i = \Delta y_i / \Delta x_i, \quad \Delta y_i = y_{i+1} - y_i, \quad \Delta x_i = x_{i+1} - x_i, \quad (3.13)$$

$$i = 0, 1, \dots, n.$$

Если при этом $k_i \approx \text{const}$, то точки (x_i, y_i) расположены приблизительно на одной прямой, и может быть поставлен вопрос о применимости эмпирической формулы (3.12). Точность такой аппроксимации определяется отклонением величин k_i от постоянного значения. В частном случае равноотстоящих точек x_i , (т. е. $\Delta x_i = \text{const}$) достаточно проверить постоянство разностей Δy_i .

Другой простейшей эмпирической формулой является квадратный трехчлен

$$y = ax^2 + bx + c. \quad (3.14)$$

Возможность использования этой формулы для случая равноотстоящих точек x_i можно проверить путем вычисления вторых разностей

$$\Delta^2 y_i = y_{i+2} - 2 y_{i+1} + y_i.$$

При $\Delta^2 y_i \approx \text{const}$ в качестве эмпирической формулы может быть выбрана (3.14).

3.2.2 Определение параметров эмпирической зависимости. Метод наименьших квадратов

Пусть тип эмпирической формулы выбран, и ее можно представить в виде

$$y = \varphi(x, a_0, a_1, \dots, a_m), \quad (3.15)$$

где φ – известная функция;

a_0, a_1, \dots, a_m – неизвестные постоянные параметры.

Задача состоит в том, чтобы определить такие значения этих параметров, при которых эмпирическая формула дает хорошее приближение данной функции, значения которой в точках x_i равны y_i ($i = 0, 1, \dots, n$).

Здесь не ставится условие (как в случае интерполяции) совпадения опытных данных y_i со значениями эмпирической функции (3.15) в точках x . Разность между этими значениями (отклонения) обозначим через ε_i . Тогда

$$\varepsilon_i = \varphi(x_i, a_0, a_1, \dots, a_m) - y_i, \quad i = 0, 1, \dots, n. \quad (3.16)$$

Задача нахождения наилучших значений параметров a_0, a_1, \dots, a_m сводится к некоторой минимизации отклонений ε_i . Существует несколько способов решения этой задачи: метод выбранных точек, метод средних и метод наименьших квадратов.

Метод выбранных точек и метод средних являются сравнительно простыми, однако в ряде случаев получаемые с их помощью аппроксимации не обладают достаточной точностью.

Метод наименьших квадратов. Запишем сумму квадратов отклонений (3.16) для всех точек x_0, x_1, \dots, x_n :

$$S = \sum_{i=0}^n \varepsilon_i^2 = \sum_{i=0}^n [\varphi(x_i, a_0, a_1, \dots, a_m) - y_i]^2 \quad (3.17)$$

Параметры a_0, a_1, \dots, a_m эмпирической формулы (3.16) будем находить из условия минимума функции $S = S(a_0, a_1, \dots, a_m)$. В этом состоит *метод наименьших квадратов*.

В теории вероятностей доказывается, что полученные таким методом значения параметров наиболее вероятны, если отклонения ε_i подчиняются нормальному закону распределения.

Поскольку параметры a_0, a_1, \dots, a_m выступают в роли независимых переменных функции S , то ее минимум найдем, приравнявая нулю частные производные по этим переменным:

$$\frac{\partial S}{\partial a_0} = 0, \quad \frac{\partial S}{\partial a_1} = 0, \quad \dots, \quad \frac{\partial S}{\partial a_m} = 0. \quad (3.18)$$

Полученные соотношения – система нормальных уравнений для определения a_0, a_1, \dots, a_m .

Рассмотрим применение метода наименьших квадратов для широко используемого на практике частного случая, когда функция φ является линейной по неизвестным параметрам a_0, a_1, \dots, a_m :

$$\varphi(x, a_0, a_1, \dots, a_m) = \sum_{j=0}^m a_j \varphi_j(x),$$

где $\varphi_0, \varphi_1, \dots, \varphi_m$ – известные функции x .

Формула (3.17) для определения суммы квадратов отклонений S примет вид

$$S = \sum_{i=0}^n \left[\sum_{j=0}^m a_j \varphi_j(x_i) - y_i \right]^2.$$

Для составления системы (3.18) продифференцируем S по переменным a_k ($k = 0, 1, \dots, m$):

$$\begin{aligned} \frac{\partial S}{\partial a_k} &= \frac{\partial}{\partial a_k} \sum_{i=0}^n \left[\sum_{j=0}^m a_j \varphi_j(x_i) - y_i \right]^2 = \sum_{i=0}^n \frac{\partial}{\partial a_k} \left[\sum_{j=0}^m a_j \varphi_j(x_i) - y_i \right]^2 = \\ &= \sum_{i=0}^n 2 \left[\sum_{j=0}^m a_j \varphi_j(x_i) - y_i \right] \varphi_k(x_i) = 2 \sum_{i=0}^n \varphi_k(x_i) \left[\sum_{j=0}^m a_j \varphi_j(x_i) - y_i \right]. \end{aligned}$$

Приравнявая найденные производные нулю, получим следующую систему уравнений:

$$\sum_{i=0}^n \varphi_k(x_i) \left[\sum_{j=0}^m a_j \varphi_j(x_i) - y_i \right] = 0, \quad k = \overline{0, m}. \quad (3.19)$$

Система (3.19) является системой нормальных уравнений.

В качестве примера рассмотрим ситуацию, когда есть основания считать, что значения x_i и соответствующие им приближения y_i отражают некую линейную зависимость. Тогда, полагая

$$\varphi(x, a_0, a_1) = a_0 + a_1 x,$$

находим $\frac{\partial \varphi}{\partial a_0} = 1$, $\frac{\partial \varphi}{\partial a_1} = x$, и для вычисления параметров функции φ

составляем систему типа

$$\begin{cases} \sum_{i=1}^n (a_0 + a_1 x_i - y_i) \cdot 1 = 0, \\ \sum_{i=1}^n (a_0 + a_1 x_i - y_i) \cdot x_i = 0. \end{cases}$$

Упрощая ее, приходим к стандартной СЛАУ:

$$\begin{cases} a_0 n + a_1 \sum_{i=1}^n x_i = \sum_{i=1}^n y_i, \\ a_0 \sum_{i=1}^n x_i + a_1 \sum_{i=1}^n x_i^2 = \sum_{i=1}^n x_i y_i. \end{cases}$$

Для параболической зависимости $y = a_0 + a_1 x + a_2 x^2$ система нормальных уравнений будет иметь вид

$$\begin{cases} a_0 n + a_1 \sum_{i=1}^n x_i + a_2 \sum_{i=1}^n x_i^2 = \sum_{i=1}^n y_i, \\ a_0 \sum_{i=1}^n x_i + a_1 \sum_{i=1}^n x_i^2 + a_2 \sum_{i=1}^n x_i^3 = \sum_{i=1}^n x_i y_i, \\ a_0 \sum_{i=1}^n x_i^2 + a_1 \sum_{i=1}^n x_i^3 + a_2 \sum_{i=1}^n x_i^4 = \sum_{i=1}^n x_i^2 y_i. \end{cases}$$

Относительные погрешности полученной аппроксимации в заданных точках оцениваются по формуле

$$\delta y_i = \frac{\varepsilon_i}{y_i} = \frac{\varphi(x_i) - y_i}{y_i}. \quad (3.20)$$

Пример 3.4. Данные об изменении содержания подвижной формы тяжелых металлов (свинца Pb, мг / кг) в растениях (в зеленой массе рапса) от содержания подвижной формы тяжелых металлов (свинца Pb, мг / кг) в почве при изучении способов утилизации осадков сточных вод очистных сооружений г. Гомеля заданы таблицей:

Таблица 3.6 – Содержание свинца (Pb, мг / кг)

Pb в почве (кислотность рН 5,20)	15	30	45	60	80	100	130	150
Pb в рапсе (зеленая масса)	0,2	0,3	0,5	0,8	1,1	1,3	1,7	1,8

Используя метод наименьших квадратов, вывести эмпирическую формулу для функции $y = f(x)$, описывающую содержание подвижной формы свинца в зеленой массе рапса и содержания свинца в почве.

По виду диаграммы рассеяния (рисунок 3.3) и предметной постановке задачи можно сделать предположение о линейной зависимости между величиной содержания подвижной формы свинца в зеленой массе рапса и величиной содержания свинца в почве.

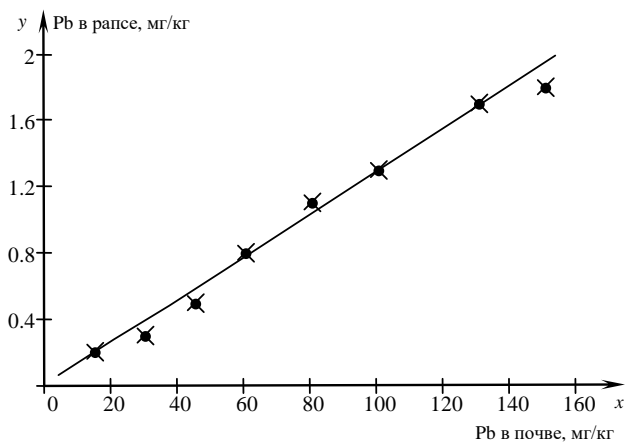


Рисунок 3.3 – Диаграмма рассеяния содержания свинца в почве и зеленой массе рапса

В качестве эмпирической формулы для аппроксимации функции $y = f(x)$ примем линейную функцию

$$y = a_0 + a_1x.$$

Для вычисления параметров a_0 и a_1 составим систему нормальных уравнений:

$$\begin{cases} a_0n + a_1 \sum_{i=1}^n x_i = \sum_{i=1}^n y_i, \\ a_0 \sum_{i=1}^n x_i + a_1 \sum_{i=1}^n x_i^2 = \sum_{i=1}^n x_i y_i. \end{cases}$$

Результаты вычислений представим в таблице.

Таблица 3.7 – Результаты промежуточных вычислений

N	x_i	y_i	x_i^2	$x_i \cdot y_i$
1	15	0,2	225	3,0
2	30	0,3	900	9,0
3	45	0,5	2025	22,5
4	60	0,8	3600	48,0
5	80	1,1	6400	88,0
6	100	1,3	10000	130,0
7	130	1,7	16900	221,0
8	150	1,8	22500	270,0
Σ	610	7,7	62550	791,5

После подстановки значений система примет вид

$$\begin{cases} 8a_0 + 610 a_1 = 7,7; \\ 610 a_0 + 62550 a_1 = 791,5. \end{cases}$$

Решив полученную систему уравнений известными методами или с помощью MathCAD (листинг 3.6), получим следующее решение: $a_0 = -0,009$; $a_1 = 0,013$.

$$a0 := 0 \quad a1 := 0$$

Given

$$8 \cdot a0 + 610 \cdot a1 = 7.7$$

$$610 \cdot a0 + 62550 \cdot a1 = 791.5$$

$$\text{Find}(a0, a1) = \begin{pmatrix} -9.197 \times 10^{-3} \\ 0.013 \end{pmatrix}$$

Листинг 3.6 – Решение системы нормальных уравнений

Таким образом, получаем следующую аппроксимацию функции, заданной в табличном виде:

$$y(x) = -0,009 + 0,013x.$$

На рисунке 3.4 построен график найденной эмпирической зависимости.

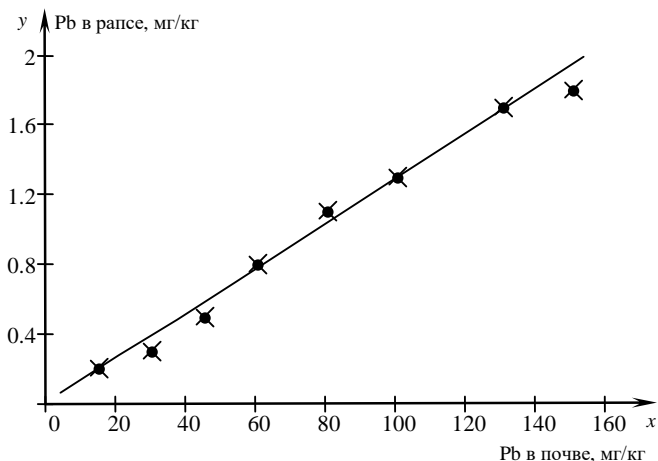


Рисунок 3.4 – График зависимости содержания свинца в зеленой массе рапса и содержания свинца в почве

По формуле (3.20) оценим относительные погрешности полученной аппроксимации в заданных точках. Результаты вычислений представим в виде таблицы.

Таблица 3.7 – Относительные погрешности аппроксимации

x_i	$\varphi(x_i)$	y_i	ε_i	δy_i
15	0,186	0,2	-0,014	-0,070
30	0,381	0,3	0,081	0,270
45	0,576	0,5	0,076	0,152
60	0,771	0,8	-0,029	-0,036
80	1,031	1,1	-0,069	-0,063
100	1,291	1,3	-0,009	-0,007
130	1,681	1,7	-0,019	-0,011
150	1,941	1,8	0,141	0,078
610	0,186	0,2	-0,014	-0,070

С увеличением уровня загрязнения почв тяжелыми металлами происходит накопление их в сельскохозяйственных культурах. Таким образом, при увеличении содержания свинца в почве на 10 мг происходит увеличение содержания его в зеленой массе рапса на

$$y(10) = -0,009 + 0,013 \cdot 10 = 0,121 \text{ мг.}$$

3.3 Исследование влияния активных илов на приживаемость саженцев сосны

При выращивании посадочного материала и лесных культур требуется внесение минеральных удобрений. Одним из источников могут выступать осадки сточных вод (ОСВ) и активные илы, получаемые в процессе промышленной и хозяйственно-бытовой деятельности. Осадки сточных вод довольно широко используются в лесном хозяйстве, при озеленении городов, для рекультивации земель и т. д.

Однако основным лимитирующим фактором при внесении в почву ОСВ является попадание в почву солей тяжелых металлов, что приводит к исчезновению многих видов трав и почвенных мхов, уменьшению вегетативного разрастания трав. Еще одной проблемой, связанной с внесением тяжелых металлов в почву вместе с ОСВ, является миграция их в нижележащие слои почвы и, в связи с этим, опасность загрязнения грунтовых вод и водоемов тяжелыми металлами и другими элементами, содержащимися в ОСВ.

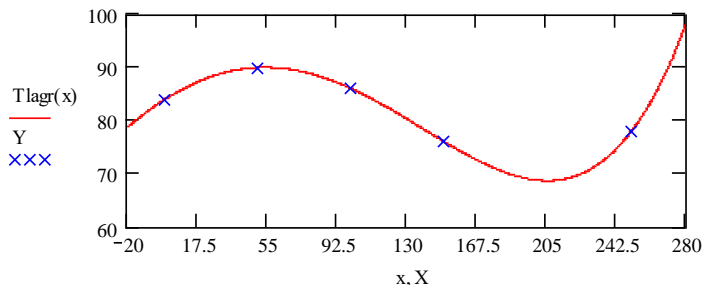
Рассмотрим пример использования полиномиальной аппроксимации для анализа влияния активных илов ОАО «Мозырский НПЗ» на процессы развития лесных культур сосны. В листинге 3.7 приведен пример построение интерполяционного многочлена Лагранжа в MathCAD при исследовании влияния различных доз илов на приживаемость саженцев сосны. При исследовании использовали интерполирование кубическими сплайнами и аппроксимацию полиномом второй степени (листинг 3.8).

1. Даны координаты точек:

$$X := (0 \ 50 \ 100 \ 150 \ 250)^T \quad Y := (84 \ 90 \ 86 \ 76 \ 78)^T$$

2. Построить интерполяционный многочлен Лагранжа:

$$T_{\text{lagr}}(x) := \text{lagran}(X, Y, x) \quad T_{\text{lagr}}(130) = 80.264$$



Листинг 3.7 – Построение интерполяционного многочлена Лагранжа

1 Άνά εποίησέ είσοδιαόε :

2 Απόειπείαόυ πείπι 2-έ ποάιέ (ίΕ) :

$$N := 4 \quad i := 0..N$$

$$X := \begin{pmatrix} 0 \\ 50 \\ 100 \\ 150 \\ 250 \end{pmatrix} \quad Y := \begin{pmatrix} 84 \\ 90 \\ 86 \\ 76 \\ 78 \end{pmatrix}$$

$$A := \begin{bmatrix} \sum_i (X_i)^0 & \sum_i (X_i)^1 & \sum_i (X_i)^2 \\ \sum_i (X_i)^1 & \sum_i (X_i)^2 & \sum_i (X_i)^3 \\ \sum_i (X_i)^2 & \sum_i (X_i)^3 & \sum_i (X_i)^4 \end{bmatrix} \quad B := \begin{bmatrix} \sum_i (X_i)^0 \cdot Y_i \\ \sum_i (X_i)^1 \cdot Y_i \\ \sum_i (X_i)^2 \cdot Y_i \end{bmatrix}$$

$$A = \begin{pmatrix} 5 & 550 & 97500 \\ 550 & 97500 & 2.0125 \times 10^7 \\ 97500 & 2.0125 \times 10^7 & 4.51875 \times 10^9 \end{pmatrix}$$

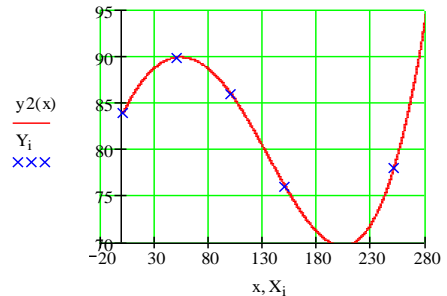
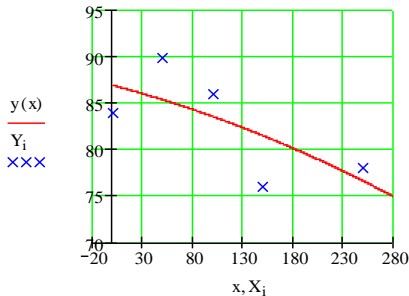
$$B = \begin{pmatrix} 414 \\ 44000 \\ 7.67 \times 10^6 \end{pmatrix} \quad a := A^{-1} \cdot B \quad a = \begin{pmatrix} 86.945508 \\ -0.028601 \\ -0.000051 \end{pmatrix}$$

$$y(x) := a_0 + a_1 \cdot x + a_2 \cdot x^2 \quad x := 0, 0.1.. 300$$

3 Είσαόπείεόιαίά πείέπι :

$$KS := \text{cspline}(X, Y) \quad y2(x) := \text{interp}(KS, X, Y, x)$$

4 Άδαόε-άπείά ίδαήποαάείά δάόέυαόιά (1 - άπείπείαόε; 2 - είσαόπείεόιαίά πείέπι) :



5 Άύάιά άιά-άίεέ όίόέε :

$$y(25) = 86.198454 \quad y(200) = 79.175258 \quad y2(70) = 89.455059 \quad y2(130) = 80.181647$$

Листинг 3.8 – Аппроксимация полиномом второй степени и интерполирование сплайнами

Таким образом, проведенные исследования показали, что внесение илов в почву в дозах 50–250 т/га отрицательно сказывается на процессах формирования растений сосны. В то же время использование илов в виде компоста в дозах 30–150 т/га оказало положительное воздействие на всхожесть семян сосны, выход стандартного посадочного материала и его параметры.

4 ЧИСЛЕННОЕ ИНТЕГРИРОВАНИЕ

4.1 Задача численного интегрирования. Квадратурные формулы

Во многих случаях, когда подынтегральная функция задана в аналитическом виде, определенный интеграл удастся вычислить непосредственно с помощью неопределенного интеграла (вернее, первообразной) по *формуле Ньютона – Лейбница*. Она состоит в том, что определенный интеграл равен приращению первообразной $F(x)$ на отрезке интегрирования:

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

Однако на практике этой формулой часто нельзя воспользоваться по двум основным причинам:

1) вид функции $f(x)$ не допускает непосредственного интегрирования т. е. первообразную нельзя выразить в элементарных функциях;

2) значения функции $f(x)$ заданы только на фиксированном конечном множестве точек x_i , т. е. функция задана в виде таблицы.

В этих случаях используются приближенные методы интегрирования. Они основаны на аппроксимации подынтегральной функции некоторыми более простыми выражениями, например многочленами.

Одним из таких способов, который может быть использован для вычисления интегралов в первом случае, является *представление подынтегральной функции в виде степенного ряда* (ряда Тейлора). Это позволяет свести вычисление интеграла от сложной функции к интегрированию многочлена представляющего первые несколько членов ряда.

Более универсальными методами, которые пригодны для обоих случаев, являются *методы численного интегрирования*, основанные на аппроксимации подынтегральной функции с помощью интерполяционных многочленов. Такая аппроксимация позволяет приближенно заменить определенный интеграл конечной суммой

$$\int_a^b f(x)dx \approx \sum_{i=0}^n \alpha_i y_i, \quad (4.1)$$

где y_i – значения функции в узлах интерполяции;

α_i – числовые коэффициенты.

Соотношение (4.1) называется *квадратурной формулой*, а его правая часть – *квадратурной суммой*. В зависимости от способа ее вычисления получаются разные методы численного интегрирования (квадратурные формулы) – методы прямоугольников, трапеций, парабол, сплайнов и т. д.

Квадратурную сумму можно вычислить как $\sum_{i=0}^n \alpha_i y_i = \sum_{i=0}^n \sigma_i$, где σ_i –

приближенное значение площади элементарной криволинейной трапеции, соответствующей элементарному отрезку $[x_{i-1}, x_i]$.

В дальнейшем при вычислении квадратурной суммы будем аппроксимировать подынтегральную функцию, используя кусочную (локальную) интерполяцию.

Следует отметить, что к вычислению определенного интеграла сводятся многие практические задачи вычисления площади фигур, определения работы переменной силы и т. д. Решение задач с использованием кратных интегралов также обычно может быть в конечном итоге сведено к вычислению определенных интегралов.

4.2 Метод трапеций

Метод трапеций использует линейную интерполяцию, т. е. график функции $y=f(x)$ представляется в виде ломаной, соединяющей точки (x_i, y_i) . В этом случае площадь всей фигуры приближенно складывается из площадей элементарных прямолинейных трапеций (рисунок 4.1)

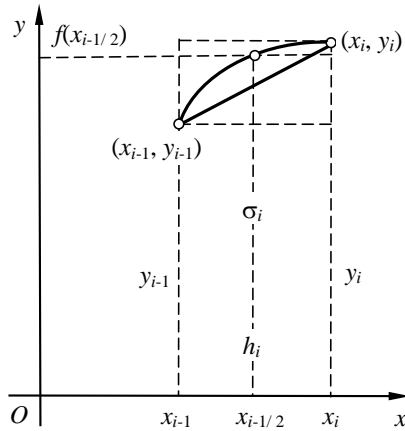


Рисунок 4.1 – Графическая интерпретация метода трапеций

Площадь каждой такой трапеции равна произведению полусуммы оснований на высоту:

$$\sigma_i = \frac{y_{i-1} + y_i}{2} h_i, \quad i = \overline{1, n}.$$

Складывая все эти равенства, получаем формулу трапеций для численного интегрирования:

$$\int_a^b f(x) dx \approx \frac{1}{2} \sum_{i=1}^n h_i (y_{i-1} + y_i). \quad (4.2)$$

Важным частным случаем формулы является ее применение при численном интегрировании с постоянным шагом $h_i = h = \text{const}$ ($i = 1, \dots, n$). Формула трапеций в этом случае принимает вид

$$\int_a^b f(x) dx \approx h \left(\frac{y_0 + y_n}{2} + \sum_{i=1}^{n-1} y_i \right). \quad (4.3)$$

Если $f''(x)$ непрерывна на $[a; b]$, то по остаточному члену формулы трапеций можно получить оценку погрешности:

$$R_n(f) \leq \frac{(b-a)^3 M_2}{12n^2} \leq \varepsilon, \quad (4.4)$$

где $M_2 = \max_{a \leq t \leq b} |f''(x)|$.

Погрешность численного интегрирования определяется шагом разбиения. Уменьшая этот шаг, можно добиться большей точности. Правда,

увеличивать число точек не всегда возможно. Если функция задана в табличном виде, приходится, как правило, ограничиваться данным множеством точек. Повышение точности может быть в этом случае достигнуто за счет повышения степени используемых интерполяционных многочленов.

Рассмотрим один из таких способа численного интегрирования: использование квадратичной интерполяции (метод Симпсона) (еще возможно интерполирование с помощью сплайнов).

4.3 Метод Симпсона

Разобьем отрезок интегрирования $[a, b]$ на четное число n равных частей с шагом h . На каждом отрезке $[x_0, x_2]$, $[x_2, x_4]$, ..., $[x_{i-1}, x_{i+1}]$, ..., $[x_{n-2}, x_n]$ подынтегральную функцию $f(x)$ заменим интерполяционным многочленом второй степени.

Формула Симпсона (или формула парабол) будет иметь вид

$$\int_a^b f(x) dx \approx \frac{h}{3} [y_0 + 4(y_1 + y_3 + \dots + y_{n-1}) + 2(y_2 + y_4 + \dots + y_{n-2}) + y_n]. \quad (4.5)$$

Для оценки погрешности используется неравенство

$$|R_n(f)| \leq \frac{(b-a)^5 M_4}{180n^4} \leq \varepsilon, \quad (4.6)$$

где $M_4 = \max_{a \leq x \leq b} |f^{IV}(x)|$.

В случае, когда вычисление производной и её максимального значения на рассматриваемом отрезке затруднено, для оценки точности вычислений можно пользоваться *правилом двойного пересчета*.

Правило двойного пересчета заключается в следующем: интеграл по выбранной квадратурной формуле вычислять дважды: сначала для n узлов, затем – для $2n$ узлов (т. е. удваивают число узлов, уменьшая вдвое шаг разбиения), после чего сравнивают $|J_n - J_{2n}| < \varepsilon$, где ε – допустимая погрешность. Если неравенство выполняется, то $J \approx J_{2n}$. Если не выполняется, то число разбиений удваивают и продолжают процедуру до достижения требуемой точности.

4.4 Определения суточной производительности насосной станции второго подъема

Рассмотрим систему: резервуар чистой воды (РЧВ) – насосная станция второго подъема (НС) – распределительная сеть водоснабжения (РСВ).

Вычислить суточную производительность (тыс. м³ / сут) насосной станции второго подъема, содержащей два насоса и работающей ступенчато в системе: РЧВ – НС – РСВ.

Суточная производительность насосной станция второго подъема определяется следующим образом:

$$W = K_r k_{\text{ступ}} \int_{t_0}^{t_k} P_t(t) dt ,$$

где K_r – коэффициент неравномерности водопотребления (1,3–1,58),

$k_{\text{ступ}}$ – количество ступеней подачи воды ($k_{\text{ступ}} = 1$ или 2),

t_0 – начальный момент времени ($t_0 = 0$ ч),

t_k – конечный момент времени ($t_k = 24$ ч),

$P_t(t)$ – функция подачи насосами воды.

Решение. Вычислим интеграл $\int_0^{24} P_t(t) dt = \int_0^{24} \frac{t dt}{\sqrt{t^2 + 1}}$ с точностью до

$\varepsilon = 0,01$, используя формулу Симпсона.

Разобьем отрезок интегрирования $[0, 24]$ на четное число $n = 10$ равных частей с шагом $h = (b - a) / n = (24 - 0) / 10 = 2,4$.

Составим таблицу значений $y_k = \frac{t_k}{\sqrt{t_k^2 + 1}}$.

Таблица 4.1 – Результаты промежуточных вычислений для $n = 10$

k	t_k	$\frac{t_k}{\sqrt{t_k^2 + 1}}$	y_0, y_8	y_k (k – нечётное)	y_k (k – чётное)
0	0,0	0,000000	0,000000		
1	2,4	0,923077		0,923077	
2	4,8	0,978980			0,978980
3	7,2	0,990492		0,990492	
4	9,6	0,994618			0,994618
5	12,0	0,996546		0,996546	
6	14,4	0,997597			0,997597
7	16,8	0,998233		0,998233	
8	19,2	0,998646			0,998646
9	21,6	0,998930		0,998930	
10	24,0	0,999133	0,999133		
Σ			0,999133	4,907278	3,969843

По формуле Симпсона получим:

$$\int_0^{24} \frac{tdt}{\sqrt{t^2+1}} \approx \frac{h}{3} [y_0 + 4(y_1 + y_3 + y_5 + y_7) + 2(y_2 + y_4) + y_8] =$$

$$= \frac{2,4}{3} (0 + 4 \cdot 4,907278 + 2 \cdot 3,969843 + 0,999133) = 23,02074.$$

Для оценки точности вычислений пользуемся правилом двойного пересчета. Интеграл вычислим для $2n = 20$ узлов (таблица 4.2).

Таблица 4.2 – Результаты промежуточных вычислений для $n = 20$

k	t_k	$\frac{t_k}{\sqrt{t_k^2+1}}$	y_0, y_8	y_k (k – нечётное)	y_k (k – чётное)
1	2	3	4	5	6
0	0,0	0,000000	0,000000		
1	1,2	0,768221		0,768221	
2	2,4	0,923077			0,923077
3	3,6	0,963518		0,963518	
4	4,8	0,978980			0,978980

Окончание таблицы 4.2

k	t_k	$\frac{t_k}{\sqrt{t_k^2+1}}$	y_0, y_8	y_k (k – нечётное)	y_k (k – чётное)
5	6,0	0,986394		0,986394	
6	7,2	0,990492			0,990492
7	8,4	0,992988		0,992988	
8	9,6	0,994618			0,994618
9	10,8	0,995741		0,995741	
10	12,0	0,996546			0,996546
	13,2	0,997143		0,997143	
	14,4	0,997597			0,997597
	15,6	0,997952		0,997952	
	16,8	0,998233			0,998233
	18,0	0,998460		0,998460	
	19,2	0,998646			
	20,4	0,998801		0,998801	0,998801
	21,6	0,998930			
	22,8	0,999040		0,999040	0,999040
	24,0	0,999133	0,999133		
Σ			0,999133	9,698257	8,877385

По формуле Симпсона получим:

$$\int_0^{24} \frac{tdt}{\sqrt{t^2+1}} \approx \frac{h}{3} [y_0 + 4(y_1 + y_3 + y_5 + y_7) + 2(y_2 + y_4) + y_8] =$$

$$= \frac{1,4}{3} (0 + 4 \cdot 9,698257 + 2 \cdot 8,877389 + 0,999133) = 23,01877.$$

Оценим погрешность вычислений:

$$|J_{10} - J_{20}| = |23,02074 - 23,01877| = 0,001973 < \varepsilon = 0,01$$

Определим суточную производительность насосной станции второго подъема:

$$W = 1,45 \cdot 2 \int_0^{24} \frac{tdt}{\sqrt{t^2+1}} = 1,45 \cdot 2 \cdot 23,01877 \approx 66,754 \text{ тыс. м}^3 / \text{сут.}$$

Интегрирование в MathCAD реализовано в виде вычислительного оператора. Оператор интегрирования устроен по принципу «как пишется, так и вводится»:

$$\int_0^{24} \frac{t}{\sqrt{t^2+1}} dt = 23.021$$

Листинг 4.1 – Вычисление интеграла

4.5 Обзор методов численного интегрирования

Кроме рассмотренных выше методов численного интегрирования существует ряд других. Дадим краткий обзор некоторых из них.

Формулы Ньютона – Котеса получаются путем замены подынтегральной функции интерполяционным многочленом Лагранжа с разбиением отрезка интегрирования на n равных частей. Получающиеся формулы используют значения подынтегральной функции в узлах интерполяции и являются точными для всех многочленов некоторой степени, зависящей от числа узлов. Точность формул растет с увеличением степени интерполяционного многочлена.

Метод Гаусса не предполагает разбиения отрезка интегрирования на равные промежутки. Формулы численного интегрирования интерполяционного типа ищутся такими, чтобы они обладали наивысшим порядком точности при заданном числе узлов. Узлы и коэффициенты формул численного интегрирования находятся из условий обращения в нуль их остаточных членов для всех многочленов максимально высокой степени.

Формула Эрмита, являющаяся частным случаем формул Гаусса, использует многочлены Чебышева для вычисления интегралов вида

$$\int_{-1}^1 \frac{f(x)dx}{\sqrt{1-x^2}}.$$

Получающаяся формула характерна тем, что все коэффициенты α_i в выражении (4.1) равны.

Метод Маркова состоит в том, что при выводе формул Гаусса вводятся дополнительные предположения о совпадении точек разбиения отрезка по крайней мере с одним из его концов.

Формула Эйлера использует значения не только подынтегральной функции в точках разбиения, но и ее производных до некоторого порядка на границах отрезка.

4.6 Вычисление определенных интегралов методом Монте-Карло

Во многих задачах исходные данные носят случайный характер, поэтому для их решения должен применяться статистико-вероятностный подход. На основе таких подходов построен ряд численных методов, которые учитывают случайный характер вычисляемых или измеряемых величин. К ним принадлежит и *метод статистических испытаний*, называемый также *методом Монте-Карло*, который применяется к решению некоторых задач вычислительной математики, в том числе и для вычисления интегралов.

Метод Монте-Карло состоит в том, что рассматривается некоторая случайная величина ξ , математическое ожидание которой равно искомой величине x :

$$M \xi = x.$$

Проводится серия n независимых испытаний, в результате которых получается (генерируется) последовательность n случайных чисел $\xi_1, \xi_2, \dots, \xi_n$ (выборка), имеющих то же распределение, что и ξ , и по совокупности этих значений находится выборочное среднее $\bar{\xi}$, которое является статистической оценкой $M\xi$. Искомая величина x полагается приближенно равной этой оценке:

$$x \approx \bar{\xi} = 1/n (\xi_1 + \xi_2 + \dots + \xi_n).$$

Пусть η – равномерно распределенная па отрезке $[0,1]$ случайная величина. Это означает, что ее плотность распределения задается соотношением

$$p_{\eta}(x) = \begin{cases} 1, & \text{если } 0 \leq x \leq 1, \\ 0 & \text{в противном случае.} \end{cases}$$

Тогда любая функция $\xi = f(\eta)$ также будет случайной величиной, и ее математическое ожидание

$$M\xi = \int_{-\infty}^{+\infty} f(x)p_{\eta}(x)dx = \int_0^1 f(x)dx.$$

Следовательно, читая это равенство в обратном порядке, приходим к выводу, что интеграл может быть вычислен как оценка математического ожидания некоторой случайной величины ξ , которая является функцией случайной величины η с равномерным законом распределения, причем оценка $M\xi$ определяется независимыми реализациями η_i случайной величины:

$$\int_0^1 f(x)dx \approx \bar{\xi} = \frac{1}{n} \sum_{i=1}^n f(\eta_i).$$

Аналогично могут быть вычислены и кратные интегралы.

5 ОБЫКНОВЕННЫЕ ДИФФЕРЕНЦИАЛЬНЫЕ УРАВНЕНИЯ

Инженеру-исследователю постоянно приходится в своей деятельности сталкиваться с дифференциальными уравнениями. Многие задачи гидравлики сводятся к дифференциальным уравнениям. В связи с этим решение дифференциальных уравнений является одной из важнейших математических задач. В вычислительной математике изучаются численные методы решения дифференциальных уравнений, которые особенно эффективны в сочетании с использованием вычислительной техники.

5.1 Теоретические сведения из курса дифференциальных уравнений

В зависимости от числа независимых переменных дифференциальные уравнения делятся на две существенно различные категории: обыкновенные дифференциальные уравнения, содержащие одну независимую переменную, и уравнения с частными производными, содержащие несколько независимых переменных.

Обыкновенными дифференциальными уравнениями (ОДУ) называются такие уравнения, которые содержат одну или несколько производных от искомой функции $y = y(x)$. Их можно записать в виде

$$F(x, y, y', \dots, y^{(n)}) = 0, \quad (5.1)$$

где x – независимая переменная.

Наивысший порядок n входящей в уравнение (5.1) производной называется *порядком дифференциального уравнения*.

В ряде случаев из общей записи дифференциального уравнения (5.1) удастся выразить старшую производную в явном виде. Например,

$$y' = f(x, y), \quad y'' = f(x, y, y'). \quad (5.2)$$

Такая форма записи называется *уравнением, разрешенным относительно старшей производной*.

Линейным дифференциальным уравнением называется уравнение, линейное относительно искомой функции и ее производных. Например, $y' - x^2y = \sin x$ – линейное уравнение первого порядка.

Решением дифференциального уравнения (5.1) называется всякая n раз дифференцируемая функция $y = \varphi(x)$, которая после ее подстановки в уравнение превращает его в тождество.

Общее решение обыкновенного дифференциального уравнения n -го порядка (5.1) содержит n произвольных постоянных C_1, C_2, \dots, C_n :

$$y = \varphi(x, C_1, C_2, \dots, C_n), \quad (5.3)$$

где (5.3) является решением уравнения (5.1) при любых значениях C_1, C_2, \dots, C_n , а любое решение уравнения (5.1) можно представить в виде (5.3) при некоторых C_1, C_2, \dots, C_n .

Частное решение дифференциального уравнения получается из общего, если произвольным постоянным придать определенные значения.

Для уравнения первого порядка общее решение зависит от одной произвольной постоянной.

Если постоянная принимает определенное значение $C = C_0$, то получается частное решение

$$y = \varphi(x, C_0).$$

Дадим геометрическую интерпретацию дифференциального уравнения первого порядка (5.2). Поскольку производная y' характеризует наклон касательной к графику решения $y = y(x)$ (*интегральной кривой*) в данной точке, то при $y' = k = \text{const}$ из (5.2) получим $f(x, y) = k$ – уравнение линии постоянного наклона, называемой *изоклиной*. Меняя k , получаем семейство изоклин.

Приведем геометрическую интерпретацию общего решения (5.3). Это решение описывает бесконечное семейство интегральных кривых с параметром C , а частному решению соответствует одна кривая из этого семейства. При некоторых дополнительных предположениях через каждую точку (x_0, y_0) проходит одна и только одна интегральная кривая. Это утверждение вытекает из следующей теоремы.

Теорема Коши. Если правая часть $f(x, y)$ уравнения (5.2) и ее частная производная $f'_y(x, y)$ определены и непрерывны в некоторой области G изменения переменных x, y , то для всякой внутренней точки (x_0, y_0) этой области данное уравнение имеет единственное решение, принимающее заданное значение $y = y_0$ при $x = x_0$.

Для уравнений высших порядков геометрическая интерпретация более сложная. Через каждую точку в области решения уравнения при $n > 1$ проходит не одна интегральная кривая. Поэтому если для выделения

некоторого частного решения уравнения первого порядка достаточно задать координаты (x_0, y_0) произвольной точки на данной интегральной кривой, то для уравнений высших порядков этого недостаточно. Здесь правило следующее: для выделения частного решения из общего нужно задавать столько дополнительных условий, сколько произвольных постоянных в общем решении, т. е. каков порядок уравнения. Следовательно, для уравнения второго порядка нужно задать два дополнительных условия, благодаря которым можно найти значения двух произвольных постоянных.

В зависимости от способа задания дополнительных условий для получения частного решения дифференциального уравнения существуют два различных типа задач: *задача Коши* и *краевая задача*. В качестве дополнительных условий могут задаваться значения искомой функции и ее производных при некоторых значениях независимой переменной, т. е. в некоторых точках.

Если эти условия задаются в одной точке, то такая задача называется *задачей Коши*. Дополнительные условия в задаче Коши называются *начальными условиями*, а точка $x = x_0$, в которой они задаются, – *начальной точкой*.

Для уравнения первого порядка дополнительное условие одно, поэтому в этом случае может быть сформулирована только задача Коши: для заданных x_0, y_0 найти такое решение $y = y(x)$ уравнения (5.2), что $y(x_0) = y_0$. Таким образом, теорема Коши дает достаточные условия существования и единственности решения задачи Коши.

Если же для уравнения порядка $n > 1$ дополнительные условия задаются в более чем одной точке, т. е. при разных значениях независимой переменной, то такая задача называется *краевой*. Сами дополнительные условия называются при этом *граничными* (или *краевыми*) *условиями*. На практике обычно граничные условия задаются в двух точках: $x = a$ и $x = b$, являющихся границами отрезка, на котором рассматривается дифференциальное уравнение.

5.2 Методы решения обыкновенных дифференциальных уравнений

Методы решения обыкновенных дифференциальных уравнений можно разбить на следующие группы: графические, аналитические, приближенные и численные.

Графические методы используют геометрические построения. В частности, одним из них является *метод изоклин* для решения дифференциальных уравнений первого порядка вида (5.2). Он основан на

геометрическом определении интегральных кривых по заранее построенному полю направлений, определенному изоклинами.

С некоторыми *аналитическими методами* мы знакомы по курсу дифференциальных уравнений. Для ряда уравнений первого порядка (с разделяющимися переменными, однородных, линейных и др.), а также для некоторых типов уравнений высших порядков (например, линейных с постоянными коэффициентами) удастся получить решения в виде формул путем аналитических преобразований.

Приближенные методы используют различные упрощения самих уравнений путем обоснованного отбрасывания некоторых содержащихся в них членов, а также специальным выбором классов искомых функций. Например, в некоторых инженерных задачах удастся представить решение в виде суммы двух составляющих, первое из которых определяет основное решение, а второе – малая добавка (*возмущение*), квадратом которой можно пренебречь. На этом основаны различные методы линеаризации. В приближенных методах также широко используется разложение решения в ряд по некоторому малому параметру, содержащемуся в данной задаче. К данной группе методов относятся и асимптотические методы, с помощью которых получают решения, описывающие предельную картину рассматриваемого явления.

Численные методы решения дифференциальных уравнений в настоящее время являются основным инструментом при исследовании научно-технических задач, описываемых дифференциальными уравнениями. При этом необходимо подчеркнуть, что данные методы особенно эффективны в сочетании с использованием современных компьютеров.

Наиболее распространенным и универсальным численным методом решения дифференциальных уравнений является *метод конечных разностей*. Его сущность состоит в следующем. Область непрерывного изменения аргумента (например, отрезок) заменяется дискретным множеством точек, называемых *узлами*. Эти узлы составляют *разностную сетку*. Искомая функция непрерывного аргумента приближенно заменяется функцией дискретного аргумента на заданной сетке. Эта функция называется *сеточной*. Исходное дифференциальное уравнение заменяется разностным уравнением относительно сеточной функции. При этом для входящих в уравнение производных используются соответствующие конечно-разностные соотношения. Такая замена дифференциального уравнения разностным называется его *аппроксимацией* на сетке (или *разностной аппроксимацией*). Таким образом, решение дифференциального уравнения сводится к отысканию значений сеточной функции в узлах сетки.

Обоснованность замены дифференциального уравнения разностным, точность получаемых решений, устойчивость метода – важнейшие вопросы, которые требуют тщательного изучения.

5.3 Метод Эйлера

Пусть требуется решить задачу Коши: найти решение дифференциального уравнения

$$y'(x) = f(x, y), \quad (5.4)$$

удовлетворяющее начальному условию $y(x) = y_0$.

При численном решении уравнения (5.4) задача ставится так: в точках $x_0, x_1, x_2, \dots, x_n$ найти приближения y_k ($k = 0, 1, 2, \dots, n$) для значений точного решения $y(x_k)$. Разность $\Delta x_k = x_{k+1} - x_k$ называется шагом сетки.

Во многих случаях величину Δx принимают постоянной h , тогда

$$x_k = x_0 + kh, \quad (k = 0, 1, 2, \dots, n). \quad (5.5)$$

Метод Эйлера основан на непосредственной замене производной разностным отношением по приближенной формуле $\frac{\Delta y}{\Delta x} = f(x, y)$, где

$$\Delta y = y(x+h) - y(x), \quad \Delta x = (x+h) - x = h.$$

Приближенное значение y_k в точке $x_k = x_0 + kh$, вычисляется по формуле

$$y_{k+1} = y_k + hf(x_k, y_k) \quad (k = 0, 1, 2, \dots, n). \quad (5.6)$$

Пример. Методом Эйлера найти значения решения дифференциального уравнения $y' = 2x - y$, удовлетворяющего условию $y(1) = 1$ в пяти точках отрезка $[1; 1,5]$, приняв $h = 0,1$.

По формуле (5.5) находим точки $x_0 = 1,0$; $x_1 = 1,1$; $x_2 = 1,2$; $x_3 = 1,3$; $x_4 = 1,4$; $x_5 = 1,5$. Значения искомой функции $y = y(x)$, удовлетворяющей условию данной задачи Коши, вычисляем по формуле (5.6). Результаты вычислений занесем в таблицу.

Таблица 5.1 – Результаты вычислений

k	y_k	x_k	$2x_k$	$f(x, y) = 2x_k - y_k$	$hf(x, y) = 0.1(2x_k - y_k)$	$y_{k+1} = hf(x_k, y_k) + y_k$
0	1,0000	1,0	2,0	1,0000	0,1000	1,1000
1	1,1000	1,1	2,2	1,1000	0,1100	1,2100
2	1,2100	1,2	2,4	1,1900	0,1190	1,3290
3	1,3290	1,3	2,6	1,2710	0,1271	1,4564
4	1,4561	1,4	2,8	1,3439	0,1344	1,5905

5	1,5905	1,5	3,0	1,4095	0,1410	1,7315
---	--------	-----	-----	--------	--------	--------

5.3 Метод Рунге-Кутты

Пусть требуется найти численное решение дифференциального уравнения $y'(x) = f(x, y)$, удовлетворяющее начальному условию $y(x_0) = y_0$.

Идея метода Рунге-Кутты состоит в представлении разности

$$\Delta y = y(x+h) - y(x) \quad (5.7)$$

в виде суммы поправок k_j с коэффициентами p_j :

$$y_{1,i+1} = y_{1,i} + \Delta y_{1,i}; \quad y_{2,i+1} = y_{2,i} + \Delta y_{2,i};$$

$$\Delta y_{1,i} = \frac{1}{6}(k_{1,1}^{(i)} + 2k_{2,1}^{(i)} + 2k_{3,1}^{(i)} + k_{4,1}^{(i)});$$

$$\Delta y_{2,i} = \frac{1}{6}(k_{1,2}^{(i)} + 2k_{2,2}^{(i)} + 2k_{3,2}^{(i)} + k_{4,2}^{(i)}),$$

$$k_{1,1}^{(i)} = hf_1(t_i, y_{1,i}, y_{2,i}); \quad k_{1,2}^{(i)} = hf_2(t_i, y_{1,i}, y_{2,i}),$$

$$k_{2,1}^{(i)} = hf_1\left(t_i + \frac{h}{2}, y_{1,i} + \frac{k_{1,1}^{(i)}}{2}, y_{2,i} + \frac{k_{1,2}^{(i)}}{2}\right);$$

$$k_{2,2}^{(i)} = hf_2\left(t_i + \frac{h}{2}, y_{1,i} + \frac{k_{1,1}^{(i)}}{2}, y_{2,i} + \frac{k_{1,2}^{(i)}}{2}\right),$$

$$k_{3,1}^{(i)} = hf_1\left(t_i + \frac{h}{2}, y_{1,i} + \frac{k_{2,1}^{(i)}}{2}, y_{2,i} + \frac{k_{2,2}^{(i)}}{2}\right);$$

$$k_{3,2}^{(i)} = hf_2\left(t_i + \frac{h}{2}, y_{1,i} + \frac{k_{2,1}^{(i)}}{2}, y_{2,i} + \frac{k_{2,2}^{(i)}}{2}\right);$$

$$k_{4,1}^{(i)} = hf_1(t_i + h, y_{1,i} + k_{3,1}^{(i)}, y_{2,i} + k_{3,2}^{(i)}); \quad k_{4,2}^{(i)} = hf_2(t_i + h, y_{1,i} + k_{3,1}^{(i)}, y_{2,i} + k_{3,2}^{(i)}).$$

5.4 Решение обыкновенных дифференциальных уравнений в MathCAD

Для решения ОДУ в MathCAD имеется выбор: либо использовать вычислительный блок `Given/Odesolve`, либо встроенные функции. Первый путь предпочтительнее из соображений наглядности представления задачи и результатов, а второй дает пользователю больше рычагов воздействия на параметры численного метода. Рассмотрим последовательно оба варианта решения.

Вычислительный блок `Given/Odesolve` для решения ОДУ, реализующий метод Рунге-Кутты, состоит из трех частей:

- `Given` – ключевое слово;
- ОДУ и начальное условие, записанное с помощью логических операторов, причем начальное условие должно быть в форме $y(t_0) = b$;
- `Odesolve(t, t1)` – встроенная функция для решения ОДУ относительно переменной t на интервале (t_0, t_1) .

Для решения задачи Коши для ОДУ и для системы ОДУ в MathCAD имеются следующие функции: `rkfixed`, `Bulstoer` и `Rkadapt`.

Наиболее часто встречается функция `rkfixed`:

□ `rkfixed(y0, t0, t1, M, D)` – решение задачи Коши для системы ОДУ методом Рунге-Кутты четвертого порядка с фиксированным шагом, где

- y_0 – вектор начальных условий;
- (t_0, t_1) – интервал интегрирования;
- M – число шагов интегрирования;
- $D(t, y)$ – векторная функция, задающая систему ОДУ.

Встроенная функция `Rkadapt`:

□ `Rkadapt(y0, t0, t1, M, D)` – решение задачи Коши для системы ОДУ методом Рунге-Кутты четвертого порядка с переменным шагом, где

- y_0 – вектор начальных условий;
- (t_0, t_1) – интервал интегрирования;
- M – число шагов интегрирования;
- $D(t, y)$ – векторная функция, задающая систему ОДУ.

Встроенная функция `Bulstoer`:

□ `Bulstoer(y0, t0, t1, M, D)` – решение задачи Коши для системы ОДУ методом Булирша-Штера, где

- y_0 – вектор начальных условий;
- (t_0, t_1) – интервал интегрирования;
- M – число шагов интегрирования;
- $D(t, y)$ – векторная функция, задающая систему ОДУ.

Применение этих функций проиллюстрируем на примере.

Пример. Уравнение свободных колебаний механической системы с одной степенью свободы с учётом сил трения имеет вид

$$\frac{d^2y}{dt^2} + k \frac{dy}{dt} + \omega^2 y = 0,$$

где k – коэффициент трения, ω^2 – квадрат частоты колебаний. Найти перемещения системы на отрезке времени $t \in \left[0; \frac{2\pi}{\omega}\right]$, удовлетворяющие начальным условиям $y(0) = 0$; $y'(0) = 1,5$; $k = 1$; $\omega^2 = 9$.

Расчёты выполнить для числа разбиений $n = 8$ и по результатам вычислений построить графики функций $y(t)$, $y'(t)$.

Решение. Уравнение свободных колебаний механической системы имеет вид

$$\frac{d^2y}{dt^2} + 1,5 \frac{dy}{dt} + 9y = 0.$$

Пусть $y = y_1$ и $\frac{dy}{dt} = y_2$. Тогда

$$\frac{dy_1}{dt} = y_2; \quad \frac{d^2y_1}{dt^2} = \frac{d}{dt} \left(\frac{dy_1}{dt} \right) = \frac{dy_2}{dt} = -1,5 \frac{dy_1}{dt} - 9y_1 = -1,5y_2 - 9y_1.$$

Иначе можно записать:

$$\begin{cases} \frac{dy_1}{dt} = y_2 = f_1(t, y_1, y_2); \\ \frac{d^2y_1}{dt^2} = \frac{dy_2}{dt} = -1,5y_2 - 9y_1 = f_2(t, y_1, y_2). \end{cases}$$

Получили систему обыкновенных дифференциальных уравнений второго порядка с начальными условиями $y_1(0) = 0$; $y_1'(0) = y_2(0) = 1$, которую решим в MathCAD.

1 $\frac{d^2}{dt^2} \left(y(t) + k \frac{d}{dt} y(t) \right) + \omega^2 \cdot y(t) = 0$

$$\frac{d^2}{dt^2} \left(y(t) + k \frac{d}{dt} y(t) \right) + \omega^2 \cdot y(t) = 0$$

$$y(0) = 0 \quad \frac{d}{dt} y(0) = 1 \quad 0 \leq t \leq \frac{2 \cdot \pi}{\omega} = \frac{2 \cdot \pi}{\sqrt{\omega^2 = \omega^2}}$$

2 $\frac{d}{dt} \left(k \cdot Y_1 + \omega^2 \cdot Y_0 \right)$

$k := 1.5$ $\omega^2 := 9$ k - $\frac{d}{dt} \left(k \cdot Y_1 + \omega^2 \cdot Y_0 \right)$; ω^2 - $\frac{d}{dt} \left(k \cdot Y_1 + \omega^2 \cdot Y_0 \right)$

$$D(x, Y) := \begin{bmatrix} Y_1 \\ -(k \cdot Y_1 + \omega^2 \cdot Y_0) \end{bmatrix} \quad \text{Äeöðáðáíøèæüíúé ðáðáðíð} \quad (\text{Çáíá-áíèá: áòíðíé áðáóíáíó äíèæáí áúòü áæèòíðíí äëý íáèçááñóíúð çíá-áííéé óóíéòèèè})$$

$$ic := \begin{pmatrix} 0 \\ 1 \end{pmatrix} \quad \text{I} \alpha \text{-} \alpha \text{e} \text{u} \text{i} \text{u} \alpha \text{ } \text{o} \text{n} \text{e} \text{i} \alpha \text{e} \text{y} \text{ } \text{c} \alpha \alpha \alpha \text{-} \text{e}$$

$$t0 := 0 \quad t1 := \frac{2 \cdot \pi}{\sqrt{\omega 2}} \quad \text{A} \alpha \text{i} \alpha \text{e} \text{o} \text{u} \text{ } \text{e} \text{i} \text{o} \alpha \text{d} \alpha \alpha \text{e} \alpha \text{ } \text{d} \alpha \alpha \text{i} \alpha \text{e} \text{y}$$

$$N1 := 100 \quad \text{E} \text{i} \text{e} \text{-} \alpha \text{n} \text{o} \alpha \text{i} \text{ } \text{c} \text{i} \alpha \text{-} \alpha \text{i} \text{e} \text{e} \text{ } \text{d} \alpha \alpha \text{i} \alpha \text{e} \text{y} \text{ } \text{i} \alpha \text{ } \text{e} \text{i} \text{o} \alpha \text{d} \alpha \alpha \text{e} \alpha \text{ } [t0; t1]:$$

$$N1 - \alpha \text{e} \text{y} \text{ } \alpha \text{d} \alpha \text{o} \text{e} \text{-} \alpha \text{n} \text{e} \text{i} \alpha \text{i} \text{ } \text{i} \text{d} \alpha \alpha \text{n} \text{o} \alpha \alpha \text{e} \alpha \text{i} \alpha \text{e} \text{y} \text{ } \text{d} \alpha \alpha \text{i} \alpha \text{e} \text{y};$$

$$N2 := 8 \quad N2 - \alpha \text{e} \text{y} \text{ } \text{o} \alpha \alpha \text{e} \text{-} \text{i} \text{i} \alpha \text{i} \text{ } \text{i} \text{d} \alpha \alpha \text{n} \text{o} \alpha \alpha \text{e} \alpha \text{i} \alpha \text{e} \text{y} \text{ } \text{d} \alpha \alpha \text{i} \alpha \text{e} \text{y}.$$

3 I \alpha \text{o} \text{d} \alpha \text{o} \alpha \text{ } \text{d} \alpha \alpha \text{i} \alpha \text{e} \text{e}, \text{i} \text{i} \text{o} \text{-} \alpha \text{i} \alpha \text{y} \text{ } \text{i} \alpha \text{o} \text{i} \alpha \text{i} \text{ } \text{D} \text{o} \text{i} \alpha \text{-} \text{E} \text{o} \text{o} \text{o}:

$$S1 := \text{rkfixed}(ic, t0, t1, N1, D) \quad S2 := \text{rkfixed}(ic, t0, t1, N2, D)$$

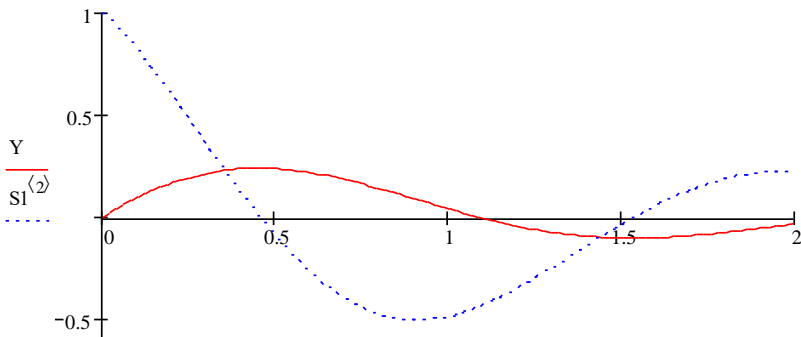
4 I \text{o} \text{i} \text{n} \text{i} \text{o} \text{d} \text{ } \text{o} \alpha \alpha \text{e} \text{e} \text{o} \text{u} \text{ } \text{d} \alpha \alpha \text{i} \alpha \text{e} \text{e}: \quad (\text{i} \alpha \text{d} \alpha \text{u} \text{e} \text{ } \text{n} \text{o} \text{i} \text{e} \alpha \alpha \text{o} - t, \text{ } \alpha \text{o} \text{i} \text{d} \text{i} \text{e} - y(t), \text{ } \text{o} \text{d} \alpha \text{o} \text{e} \text{e} - y'(t)

$$S2 = \begin{pmatrix} 0 & 0 & 1 \\ 0.261799 & 0.194833 & 0.451589 \\ 0.523599 & 0.232909 & -0.137707 \\ 0.785398 & 0.146417 & -0.470591 \\ 1.047198 & 0.017224 & -0.469255 \\ 1.308997 & -0.078615 & -0.242112 \\ 1.570796 & -0.105648 & 0.028516 \\ 1.832596 & -0.073029 & 0.198131 \\ 2.094395 & -0.01572 & 0.21753 \end{pmatrix}$$

5 A \text{d} \alpha \text{o} \text{e} \text{-} \alpha \text{n} \text{e} \alpha \text{y} \text{ } \text{e} \text{i} \text{o} \alpha \text{d} \text{i} \text{o} \alpha \text{o} \text{e} \text{y} \text{ } \text{d} \alpha \alpha \text{i} \alpha \text{e} \text{y} \text{ } \text{c} \alpha \alpha \alpha \text{-} \text{e}:

$$X := S1^{(0)} \quad \text{C} \text{i} \alpha \text{-} \alpha \text{i} \alpha \text{e} \text{y} \text{ } \text{i} \alpha \text{c} \alpha \alpha \text{e} \text{n} \text{e} \text{i} \alpha \text{e} \text{ } \text{i} \alpha \text{d} \alpha \text{i} \alpha \text{i} \alpha \text{i} \alpha \text{e} \text{ } t$$

$$Y := S1^{(1)} \quad Y1 := S1^{(2)} \quad \text{C} \text{i} \alpha \text{-} \alpha \text{i} \alpha \text{e} \text{y} \text{ } \text{o} \text{o} \text{i} \text{e} \text{o} \text{e} \text{e} \text{ } y(t) \text{ } \text{e} \text{ } \alpha, \text{ } \text{i} \text{o} \text{i} \text{e} \text{c} \alpha \text{i} \alpha \text{i} \alpha \text{i} \alpha \text{e} \text{ } y'(t)$$



X

Листинг 5.1 – Решение ОДУ

5.5 Построение кривой свободной поверхности потока неравномерного установившегося движения воды в открытых призматических руслах

На начальной стадии проектирования гидротехнического сооружения (плотины, шлюза) на открытых трапециевидальных руслах необходимо установить законы изменения глубины и скорости жидкости вдоль потока. При этом обычно рассматривают задачу о построении кривой свободной поверхности потока. Получив такую кривую, можно в любом сечении русла найти глубину потока и среднюю скорость движения воды.

Задача о построении кривой свободной поверхности потока ставится и решается следующим образом:

– задается характеристика русла водотока:

форма поперечного сечения – симметричная трапеция;

b – ширина русла по низу;

m – коэффициент заложения откосов;

i – уклон русла;

n – коэффициент шероховатости;

Q – расход воды;

– определяется нормальная и критическая глубины потока;

– составляется дифференциальное уравнение неравномерного установившегося движения для участка русла;

– интегрируется составленное дифференциальное уравнение неравномерного установившегося движения, в результате чего получаем уравнение кривой свободной поверхности потока.

Нормальная $h_{\text{норм}}$ и критическая $h_{\text{крит}}$ глубины потока определяются с использованием следующих формул:

$$K(h) = \omega(h) \cdot C(h) \sqrt{R(h) - Q\sqrt{i}};$$

$$F(h) = \frac{\omega^3(h)}{B(h) - \frac{\alpha Q^2}{g}},$$

где $\omega(h) = (b + mh)h$ – площадь поперечного сечения;

$$C(h) = \left(\frac{1}{n}\right) R^y - \text{коэффициент Шези};$$

$$R(h) = \frac{(b + mh)h}{b + 2h\sqrt{1 + m}} - \text{гидравлический радиус};$$

$B(h) = b + mh$ – ширина канала по верху.

Определив значения нормальной и критической глубин, имеем две гидравлические оси, в зависимости от расположения которых существуют несколько видов кривых свободной поверхности.

Кривая свободной поверхности потока определяется как решение дифференциального уравнения неравномерного установившегося движения воды:

$$\frac{dh}{dl} = \frac{i - \frac{Q^2}{C^2 \omega^2 R}}{1 - \frac{\alpha Q^2}{g} \cdot \frac{B}{\omega^3}}.$$

Дифференциальное уравнение может быть решено методом Рунге-Кутты четвертого порядка средствами MathCAD с использованием встроенной функции `rkfixed` (листинг 5.2), что позволяет получить координаты точек кривой свободной поверхности неравномерного установившегося движения воды в открытых призматических руслах.

1 Εἰσαγωγή αἰσθημάτων:

$b := 10$ - ἀριθμὸς ὁρόνων

$n := 0.018$ - ἐπιτόκιο ἐξουσιοδότησης

$m := 1.2$ - ἐπιτόκιο ἐξουσιοδότησης ὑποχρέωσης

$Q := 49$ - ἀριθμὸς ἀξιών

$i := 0.0009$ - ὁἰκονομικὸ ὄψιμα

$\alpha := 1.1$ - ἐπιτόκιο ἐξουσιοδότησης ἔσοδων

$h_0 := 5.31$ - ἀξία ἐξουσιοδότησης ἀεὶ

$N := 1000$ - ἀριθμὸς ὁρόνων

2 Ἐπιτόκιο ἐξουσιοδότησης ἐξουσιοδότησης ἀεὶ ἀξιών

$\omega(h) := (b + m \cdot h) \cdot h$ ἰσοδύναμο ἰσοδύναμο ἰσοδύναμο ἰσοδύναμο

$\chi(h) := b + 2 \cdot h \cdot \sqrt{1 + m^2}$ ἰσοδύναμο ἰσοδύναμο

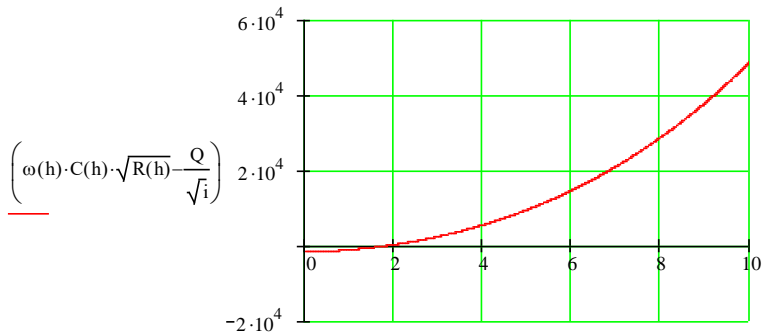
$R(h) := \frac{\omega(h)}{\chi(h)}$ ἰσοδύναμο ἰσοδύναμο ἰσοδύναμο

$y(h) := 2.5 \cdot \sqrt{n} - 0.13 - 0.75 \cdot \sqrt{R(h)} \cdot (n - 0.1)$

$C(h) := \frac{1}{n} \cdot R(h)^{y(h)}$ ἰσοδύναμο ἰσοδύναμο ἰσοδύναμο

ἰσοδύναμο ἰσοδύναμο ἐξουσιοδότησης ἀεὶ ἀξιών:

$K(h) := \omega(h) \cdot C(h) \cdot \sqrt{R(h)} - \frac{Q}{\sqrt{i}}$



h := 0.5 solution := root(K(h), h) solution = 1.832508

B(h) := b + 2 * m * h Øèðèíà èáíàèèà ï ááððó

3 ïñððíáíèá èðèáíé ñáíáíáíé ïñáððíñòè ïòèèá

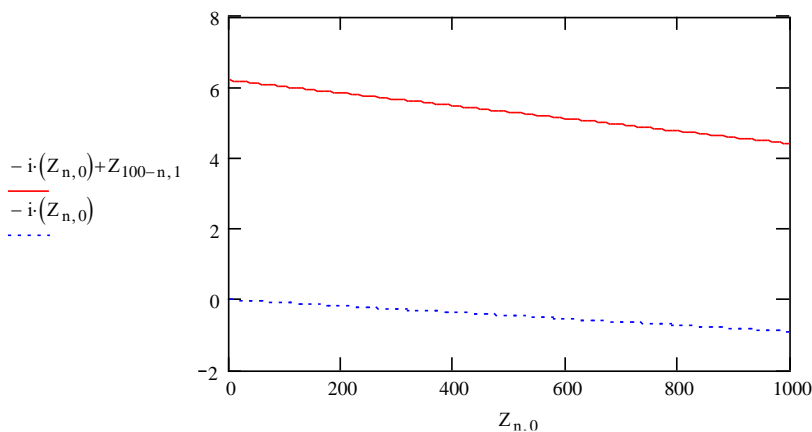
$$D(s, h) := \frac{\omega(h_0)^2 \cdot C(h_0)^2 \cdot R(h_0)}{1 - \frac{\alpha \cdot Q^2 \cdot B(h_0)}{g \cdot \omega(h_0)^3}}$$

Äèðððáíðèèèüííá òðááíáíèá íáðáíñáðííáí
òñðáííèèèðááíñý áèèèáíèý

ic₀ := h₀ Íà-àèüííá òñèíèèá

Z := rkfixed(ic, 0, N, 100, D) Ðáðáíèá àèðððáíðèèèüííáí òðááíáíèý íáðíáíí
Ðóíáá-Èóòòù ÷áðááððíáí ïðýáèá

n := 0.. 100



Листинг 5.2 – Построение кривой свободной поверхности потока

ПРИЛОЖЕНИЕ А
(обязательное)

**Варианты заданий к лабораторным работам
по численным методам**

ЗАДАНИЕ 1. Численные методы решения систем линейных алгебраических уравнений.

1 Методом простой итерации и методом Зейделя найти решение системы линейных уравнений с точностью до 0,001.

2 Найти решение системы линейных уравнений в MathCAD, используя блок Given/Find.

3 Найти решение системы линейных уравнений в MathCAD, используя встроенную функцию lsolve.

$$1. \begin{cases} x_1 = 0,19 x_1 + 0,12 x_2 - 0,62 x_3 + 0,08 x_4 - 1,3; \\ x_2 = 0,03 x_1 + 0,98 x_2 + 0,33 x_3 - 0,07 x_4 + 3,1; \\ x_3 = 0,09 x_1 + 0,13 x_2 + 0,32 x_3 + 0,28 x_4 - 1,7; \\ x_4 = 0,49 x_1 - 0,23 x_2 + 0,08 x_3 + 0,37 x_4 + 1,7. \end{cases}$$

$$2. \begin{cases} x_1 = 0,14 x_2 - 0,35 x_3 + 0,18 x_4 - 1,6; \\ x_2 = 0,18 x_2 + 0,43 x_3 - 0,08 x_4 + 0,33; \\ x_3 = 0,22 x_1 + 0,18 x_2 + 0,21 x_3 + 0,07 x_4 - 0,48; \\ x_4 = 0,18 x_1 + 0,07 x_2 + 0,21 x_3 + 0,04 x_4 - 2,2. \end{cases}$$

$$3. \begin{cases} x_1 = 0,13 x_1 - 0,05 x_2 + 0,22 x_3 - 0,33 x_4 + 0,43; \\ x_2 = 0,22 x_1 + 0,35 x_2 - 0,08 x_3 + 0,07 x_4 - 0,8; \\ x_3 = 0,33 x_1 + 0,13 x_2 - 0,08 x_3 - 0,05 x_4 - 1,8; \\ x_4 = 0,18 x_1 + 0,17 x_2 + 0,29 x_3 + 0,33 x_4 + 1,7. \end{cases}$$

$$4. \begin{cases} x_1 = 0,23 x_1 + 0,22 x_2 - 0,23 x_3 + 0,07 x_4 + 0,34; \\ x_2 = 0,45 x_1 - 0,23 x_3 + 0,07 x_4 - 0,23; \\ x_3 = 0,31 x_1 - 0,08 x_3 + 0,18 x_4 - 0,15; \\ x_4 = 0,08 x_1 + 0,09 x_2 + 0,33 x_3 + 0,21 x_4 - 1,7. \end{cases}$$

5.
$$\begin{cases} x_1 = 0,22x_1 - 0,16x_2 - 0,08x_3 + 0,15x_4 - 0,42; \\ x_2 = 0,16x_1 - 0,23x_2 + 0,11x_3 - 0,21x_4 + 1,43; \\ x_3 = 0,05x_1 - 0,08x_2 + 0,34x_4 - 0,56; \\ x_4 = 0,12x_1 + 0,14x_2 - 0,18x_3 + 0,06x_4 + 1,62. \end{cases}$$
6.
$$\begin{cases} x_1 = 0,38x_2 - 0,13x_3 + 0,62x_4 + 0,31; \\ x_2 = 0,16x_1 - 0,23x_2 + 0,18x_3 + 0,16x_4 - 0,39; \\ x_3 = 0,35x_1 + 0,12x_2 + 0,32x_3 - 0,18x_4 - 0,32; \\ x_4 = 0,25x_1 + 0,21x_2 - 0,16x_3 + 0,03x_4 + 0,63. \end{cases}$$
7.
$$\begin{cases} x_1 = 0,16x_1 + 0,18x_2 + 0,33x_3 + 0,16x_4 + 2,43; \\ x_2 = 0,12x_1 + 0,23x_3 - 0,05x_4 - 1,12; \\ x_3 = 0,16x_1 - 0,08x_2 - 0,12x_4 - 0,56; \\ x_4 = 0,29x_1 + 0,22x_2 - 0,13x_3 + 0,83. \end{cases}$$
8.
$$\begin{cases} x_1 = 0,23x_1 - 0,14x_2 + 0,06x_3 - 0,12x_4 + 1,21; \\ x_2 = 0,32x_1 + 0,32x_3 - 0,18x_4 - 0,72; \\ x_3 = 0,28x_1 - 0,12x_2 + 0,43x_3 + 0,32x_4 + 0,58; \\ x_4 = 0,15x_1 + 0,22x_2 + 0,14x_3 - 0,56. \end{cases}$$
9.
$$\begin{cases} x_1 = 0,14x_1 + 0,23x_2 + 0,18x_3 + 0,17x_4 - 1,42; \\ x_2 = 0,12x_1 - 0,44x_2 + 0,08x_3 + 0,09x_4 - 0,13; \\ x_3 = 0,26x_1 + 0,24x_2 - 0,35x_4 + 1,21; \\ x_4 = 0,23x_1 - 0,08x_2 + 0,05x_3 + 0,25x_4 + 0,65. \end{cases}$$
10.
$$\begin{cases} x_1 = 0,14x_1 + 0,21x_2 + 0,06x_3 - 0,34x_4 + 1,42; \\ x_2 = 0,35x_1 + 0,32x_3 + 0,12x_4 - 0,57; \\ x_3 = 0,15x_1 - 0,17x_2 - 0,05x_4 + 0,48; \\ x_4 = 0,12x_1 - 0,43x_2 + 0,04x_3 - 0,21x_4 - 2,14. \end{cases}$$

11.
$$\begin{cases} x_1 = 0,27x_1 + 0,27x_2 - 0,13x_3 - 0,11x_4 - 1,42; \\ x_2 = 0,13x_1 - 0,12x_2 + 0,09x_3 - 0,06x_4 + 0,48; \\ x_3 = 0,31x_1 + 0,15x_2 - 0,02x_3 + 0,12x_4 - 0,34; \\ x_4 = 0,13x_1 + 0,18x_2 + 0,24x_3 + 0,43x_4 + 2,72. \end{cases}$$
12.
$$\begin{cases} x_1 = 0,45x_1 + 0,05x_2 - 0,08x_3 + 0,14x_4 - 0,48; \\ x_2 = 0,32x_1 - 0,13x_2 - 0,12x_3 + 0,11x_4 + 1,24; \\ x_3 = 0,17x_1 + 0,06x_2 - 0,08x_3 + 0,32x_4 + 1,25; \\ x_4 = 0,21x_1 - 0,16x_2 + 0,36x_3 - 0,88. \end{cases}$$
13.
$$\begin{cases} x_1 = 0,34x_2 - 0,17x_3 + 0,06x_4 + 0,21; \\ x_2 = 0,12x_1 + 0,12x_3 + 0,17x_4 - 1,17; \\ x_3 = 0,37x_1 - 0,38x_2 + 0,21x_3 - 0,81; \\ x_4 = 0,11x_1 + 0,22x_2 + 0,03x_3 + 0,05x_4 + 0,72. \end{cases}$$
14.
$$\begin{cases} x_1 = 0,12x_2 + 0,08x_3 + 0,13x_4 - 0,12; \\ x_2 = 0,07x_1 - 0,38x_2 - 0,05x_3 + 0,41x_4 + 3,8; \\ x_3 = 0,14x_1 + 0,42x_2 + 0,11x_3 - 0,07x_4 - 1,3; \\ x_4 = 0,17x_1 + 0,18x_2 - 0,13x_3 + 0,19x_4 + 0,13. \end{cases}$$
15.
$$\begin{cases} x_1 = 0,23x_1 + 0,27x_2 - 0,22x_3 - 0,28x_4 + 0,21; \\ x_2 = -0,21x_1 - 0,15x_3 + 0,18x_4 - 0,33; \\ x_3 = 0,12x_1 + 0,13x_2 - 0,33x_3 + 0,18x_4 - 0,18; \\ x_4 = 0,33x_1 - 0,05x_2 + 0,06x_3 - 0,28x_4 - 0,17. \end{cases}$$
16.
$$\begin{cases} x_1 = 0,29x_1 - 0,07x_2 + 0,38x_3 - 0,21x_4 - 0,11; \\ x_2 = -0,22x_1 + 0,08x_2 + 0,11x_3 + 0,33x_4 - 0,64; \\ x_3 = -0,21x_1 - 0,07x_2 + 0,09x_3 - 0,11x_4 + 1,51; \\ x_4 = 0,13x_1 - 0,41x_2 - 1,21. \end{cases}$$

$$17. \begin{cases} x_1 = 0,22x_1 + 0,23x_2 + 0,11x_3 + 0,31x_4 + 2,7; \\ x_2 = 0,38x_1 - 0,12x_3 + 0,22x_4 - 1,55; \\ x_3 = 0,11x_1 + 0,23x_2 - 0,51x_4 + 1,2; \\ x_4 = 0,17x_1 - 0,21x_2 + 0,31x_3 - 0,22x_4 - 0,17. \end{cases}$$

$$18. \begin{cases} x_1 = 0,27x_1 - 0,08x_2 + 0,11x_3 - 0,18x_4 - 0,41; \\ x_2 = 0,18x_1 + 0,52x_2 + 0,21x_4 + 1,17; \\ x_3 = 0,33x_1 + 0,31x_2 - 0,21x_4 - 1,02; \\ x_4 = 0,08x_1 - 0,33x_3 + 0,28x_4 - 0,28. \end{cases}$$

$$19. \begin{cases} x_1 = 0,15x_1 - 0,06x_2 - 0,12x_3 + 0,14x_4 - 2,17; \\ x_2 = 0,34x_1 - 0,12x_2 + 0,08x_3 + 0,11x_4 + 1,4; \\ x_3 = 0,54x_1 + 0,08x_2 - 0,46x_3 + 0,14x_4 - 2,1; \\ x_4 = 0,11x_1 + 0,12x_2 - 0,03x_4 - 0,8. \end{cases}$$

$$20. \begin{cases} x_1 = 0,38x_1 - 0,03x_2 - 0,34x_4 - 1,2; \\ x_2 = 0,31x_2 + 0,27x_3 - 0,08x_4 + 0,81; \\ x_3 = 0,33x_1 - 0,07x_3 + 0,21x_4 - 0,92; \\ x_4 = 0,14x_1 + 0,03x_3 + 0,58x_4 + 0,47. \end{cases}$$

$$21. \begin{cases} x_1 = 0,12x_1 - 0,13x_2 + 0,25x_3 - 0,16x_4 + 1,24; \\ x_2 = 0,24x_1 + 0,34x_2 - 0,38x_3 + 0,24x_4 - 0,89; \\ x_3 = 0,33x_1 + 0,03x_2 + 0,16x_3 - 0,32x_4 + 1,15; \\ x_4 = 0,12x_1 - 0,05x_2 + 0,15x_4 - 0,57. \end{cases}$$

$$22. \begin{cases} x_1 = 0,33x_1 - 0,04x_2 + 0,21x_3 - 0,18x_4 - 0,34; \\ x_2 = 0,15x_1 - 0,23x_2 + 0,06x_3 - 0,88; \\ x_3 = 0,26x_1 + 0,34x_2 - 0,11x_3 + 0,62; \\ x_4 = 0,05x_1 - 0,26x_2 + 0,34x_3 - 0,12x_4 - 1,17. \end{cases}$$

$$23. \begin{cases} x_1 = 0,11x_1 + 0,12x_2 - 0,34x_3 - 0,16x_4 - 0,64; \\ x_2 = 0,34x_1 - 0,08x_2 + 0,17x_3 - 0,18x_4 + 1,42; \\ x_3 = 0,36x_1 + 0,34x_2 + 0,15x_3 - 0,31x_4 - 0,42; \\ x_4 = 0,42x_1 - 0,26x_2 - 0,08x_3 + 0,25x_4 + 0,83. \end{cases}$$

$$24. \begin{cases} x_1 = 0,12x_1 - 0,18x_2 + 0,02x_3 + 0,21x_4 + 1,03; \\ x_2 = 0,26x_1 + 0,12x_2 - 0,14x_3 + 0,27x_4 - 0,65; \\ x_3 = 0,37x_1 + 0,27x_2 - 0,02x_3 - 0,24x_4 + 1,23; \\ x_4 = 0,12x_1 + 0,21x_2 - 0,18x_3 + 0,25x_4 - 1,13. \end{cases}$$

$$25. \begin{cases} x_1 = 0,12x_1 - 0,32x_2 + 0,03x_3 + 0,44; \\ x_2 = 0,21x_1 - 0,26x_2 - 0,36x_3 - 0,14x_4 + 1,42; \\ x_3 = 0,12x_1 + 0,08x_2 - 0,14x_3 - 0,24x_4 - 1,83; \\ x_4 = 0,15x_1 - 0,35x_2 - 0,18x_3 - 1,42. \end{cases}$$

$$26. \begin{cases} x_1 = 0,18x_1 - 0,24x_2 - 0,12x_3 + 0,15x_4 - 0,33; \\ x_2 = 0,31x_1 + 0,23x_2 - 0,15x_3 + 0,32x_4 + 0,94; \\ x_3 = 0,15x_1 - 0,12x_2 + 0,14x_3 - 0,18x_4 - 1,16; \\ x_4 = 0,12x_1 + 0,08x_2 + 0,06x_3 + 0,57. \end{cases}$$

$$27. \begin{cases} x_1 = 0,23x_1 + 0,23x_2 - 0,44x_3 - 0,05x_4 + 0,13; \\ x_2 = 0,24x_1 - 0,31x_3 + 0,15x_4 + 0,18; \\ x_3 = 0,06x_1 + 0,15x_2 - 0,23x_4 - 1,44; \\ x_4 = 0,72x_1 - 0,08x_2 - 0,05x_3 + 2,42. \end{cases}$$

$$28. \begin{cases} x_1 = 0,27x_1 + 0,31x_2 - 0,18x_3 + 0,22x_4 + 0,71; \\ x_2 = 0,28x_1 + 0,33x_3 + 0,22x_4 + 0,32; \\ x_3 = 0,32x_1 - 0,18x_2 + 0,05x_3 - 0,19x_4 - 0,80; \\ x_4 = 0,12x_1 + 0,28x_2 - 0,14x_3 + 0,94. \end{cases}$$

$$29. \begin{cases} x_1 = 0,24x_2 + 0,27x_3 - 0,16x_4 + 0,42; \\ x_2 = 0,13x_1 - 0,18x_2 + 0,14x_3 + 0,16x_4 - 0,66; \\ x_3 = 0,23x_1 - 0,22x_2 + 0,16x_3 - 0,35x_4 - 0,08; \\ x_4 = 0,12x_1 + 0,32x_2 - 0,47x_3 + 0,18x_4 + 1,04. \end{cases}$$

$$30. \begin{cases} x_1 = 0,12x_1 + 0,18x_2 + 0,11x_3 - 0,23x_4 - 0,27; \\ x_2 = 0,27x_1 - 0,22x_2 - 0,13x_3 + 0,58; \\ x_3 = 0,14x_1 + 0,32x_2 + 0,15x_3 - 0,07x_4 - 0,18; \\ x_4 = 0,25x_1 - 0,11x_2 + 0,09x_3 - 0,32x_4 + 1,25. \end{cases}$$

ЗАДАНИЕ 2. Нахождение собственных значений и собственных векторов матриц.

1 Используя метод итераций, определить первое (наибольшее по модулю) и второе собственные числа матрицы и соответствующие им собственные векторы, имеющие первую норму, равную единице, с точностью 0,001.

2 Определить собственные значения и собственные вектора матриц в MathCAD, используя встроенные функции.

$$1. \begin{pmatrix} 2,3 & 0,1 & 1,7 \\ 1,6 & 0,5 & 0,4 \\ 0,6 & 0,4 & 1,4 \end{pmatrix}$$

$$2. \begin{pmatrix} 2,7 & 1 & 1,7 \\ 1 & 3,2 & 1,7 \\ 1,7 & 1,7 & 3,7 \end{pmatrix}$$

$$3. \begin{pmatrix} 1,4 & 1,2 & -1,3 \\ 1,2 & 0,9 & 0,4 \\ -1,3 & 0,4 & 0,8 \end{pmatrix}$$

$$4. \begin{pmatrix} 3,2 & 1 & 2,2 \\ 1 & 3,7 & 2,2 \\ 2,2 & 2,2 & 4,2 \end{pmatrix}$$

$$5. \begin{pmatrix} 2,3 & 1,1 & 1,7 \\ 1,1 & 0,5 & 1,4 \\ 0,6 & 0,4 & 0,4 \end{pmatrix}$$

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

$$7. \begin{pmatrix} 1,6 & 1,5 & 1,1 \\ 1,2 & 1,1 & 0,6 \\ 1,1 & 0,6 & 0,8 \end{pmatrix}$$

$$8. \begin{pmatrix} 3,3 & 1 & 2,3 \\ 1 & 3,8 & 2,3 \\ 2,3 & 2,3 & 4,3 \end{pmatrix}$$

$$9. \begin{pmatrix} 2,9 & 1 & 1,9 \\ 1 & 3,4 & 1,9 \\ 1,9 & 1,9 & 3,9 \end{pmatrix}$$

$$10. \begin{pmatrix} 1,3 & 0,1 & -1,7 \\ 1,6 & -0,5 & 0,4 \\ 1,6 & 1,4 & 1,4 \end{pmatrix}$$

$$11. \begin{pmatrix} 3,5 & 1 & 2,5 \\ 1 & 4 & 2,5 \\ 2,5 & 2,5 & 4,5 \end{pmatrix}$$

$$12. \begin{pmatrix} 1,8 & 0,9 & 1 \\ 0,9 & 1,8 & 0,3 \\ 1 & 0,3 & 1,8 \end{pmatrix}$$

13. $\begin{pmatrix} 2,6 & 1,2 & -0,1 \\ 1,2 & 2,1 & 1,6 \\ -0,1 & 1,6 & 0,8 \end{pmatrix}$ 14. $\begin{pmatrix} 0,3 & 0,1 & 0,7 \\ 0,6 & 0,5 & 0,3 \\ 0,6 & 0,3 & 0,4 \end{pmatrix}$ 15. $\begin{pmatrix} 3,4 & 1 & 2,4 \\ 1 & 3,9 & 2,4 \\ 2,4 & 2,4 & 4,4 \end{pmatrix}$
16. $\begin{pmatrix} -3,1 & 1 & -2,1 \\ 1 & -3,6 & 2,1 \\ 2,1 & 2,1 & 4,1 \end{pmatrix}$ 17. $\begin{pmatrix} 2,8 & 1,2 & 0,1 \\ 1,2 & 2,3 & 1,8 \\ 0,1 & 1,8 & 0,8 \end{pmatrix}$ 18. $\begin{pmatrix} 2 & 1,2 & -0,7 \\ 1,2 & 1,5 & 1 \\ -0,7 & 1 & 0,8 \end{pmatrix}$
19. $\begin{pmatrix} 1,6 & 2,3 & -0,5 \\ 2,3 & 2 & 1,2 \\ -0,5 & 1,2 & 0,6 \end{pmatrix}$ 20. $\begin{pmatrix} 2,2 & 1,2 & -0,5 \\ 1,2 & 1,7 & 1,2 \\ -0,5 & 1,2 & 0,8 \end{pmatrix}$ 21. $\begin{pmatrix} 2,4 & 1,2 & 2,5 \\ 1,2 & 3,5 & 1,4 \\ 2,5 & 1,4 & 4,2 \end{pmatrix}$
22. $\begin{pmatrix} 1,3 & -0,1 & 1,7 \\ 1,6 & 0,5 & 2,4 \\ 0,6 & 0,4 & -1,4 \end{pmatrix}$ 23. $\begin{pmatrix} 2,4 & 1 & 1,4 \\ 1 & 2,9 & 1,4 \\ 1,4 & 1,4 & 3,4 \end{pmatrix}$ 24. $\begin{pmatrix} 1,3 & 0,4 & 0,5 \\ 0,4 & 1,3 & 0,3 \\ 0,5 & 0,3 & 1,3 \end{pmatrix}$
25. $\begin{pmatrix} 1,6 & 0,7 & 0,8 \\ 0,7 & 1,6 & 0,3 \\ 0,8 & 0,3 & 1,6 \end{pmatrix}$ 26. $\begin{pmatrix} 1,5 & 0,1 & 1,8 \\ 1,6 & 1,5 & 0,4 \\ 0,6 & 0,4 & 1,4 \end{pmatrix}$ 27. $\begin{pmatrix} 2,5 & 1 & 1,5 \\ 1 & 3 & 1,5 \\ 1,5 & 1,5 & 3,5 \end{pmatrix}$
28. $\begin{pmatrix} 1,4 & 0,5 & 0,6 \\ 0,5 & 1,4 & 0,3 \\ 0,6 & 0,3 & 1,4 \end{pmatrix}$ 29. $\begin{pmatrix} 1,7 & 0,8 & 0,9 \\ 0,8 & 0,7 & 0,3 \\ 0,9 & 0,3 & 1,7 \end{pmatrix}$ 30. $\begin{pmatrix} 2,3 & 1 & 1,3 \\ 1 & 2,8 & 1,3 \\ 1,3 & 1,3 & 3,3 \end{pmatrix}$

ЗАДАНИЕ 3. Построение полиномов таблично заданных функций при исследовании процессов фильтрации.

Для различных условий фильтрации воды в крупнопористых и дисперсных средах установлены зависимости коэффициента фильтрации K_f от размеров и формы материалов, механического состава и прозрачности среды, гидравлических градиентов.

Данные об изменении коэффициента фильтрации воды K_f , м/с, в зависимости от диаметра частиц материала d , см, заданы таблицей.

Построить интерполяционный полином второй степени, описывающий зависимость коэффициента фильтрации K_{ϕ} от диаметра частиц материала d , используя:

- 1) метод неопределенных коэффициентов;
- 2) формулы Лагранжа и Ньютона;
- 3) интерполирование в MathCAD.

1.

d , см	5	10	15
K_{ϕ} , м/с	0,25	0,31	0,4

2.

d , см	3	7	11
K_{ϕ} , м/с	0,11	0,18	0,21

3.

d , см	4	8	12
K_{ϕ} , м/с	0,18	0,20	0,35

4.

d , см	6	12	18
K_{ϕ} , м/с	0,19	0,22	0,26

5.

d , см	2	6	10
K_{ϕ} , м/с	0,05	0,12	0,24

6.

d , см	3	6	11
K_{ϕ} , м/с	0,11	0,18	0,21

7.

d , см	5	10	15
K_{ϕ} , м/с	0,25	0,31	0,4

8.

d , см	3	7	11
K_{ϕ} , м/с	0,11	0,18	0,21

9.

d , см	7	12	17
K_{ϕ} , м/с	0,25	0,35	0,39

10.

d , см	10	15	20
K_{ϕ} , м/с	0,3	0,34	0,38

11.

d , см	5	10	15
K_{ϕ} , м/с	0,19	0,26	0,33

12.

d , см	3	7	11
K_{ϕ} , м/с	0,09	0,13	0,17

13.

d , см	4	8	12
K_{ϕ} , м/с	0,12	0,16	0,23

14.

d , см	6	12	18
K_{ϕ} , м/с	0,15	0,20	0,23

15.

d , см	2	6	10
K_{ϕ} , м/с	0,03	0,08	0,13

16.

d , см	3	6	11
K_{ϕ} , м/с	0,10	0,18	0,24

17.

d , см	5	10	15
K_{ϕ} , м/с	0,16	0,27	0,32

18.

d , см	3	7	11
K_{ϕ} , м/с	0,08	0,15	0,19

19.

d , см	7	12	17
K_{ϕ} , м/с	0,23	0,30	0,35

20.

d , см	10	15	20
K_{ϕ} , м/с	0,24	0,35	0,39

21.

d , см	5	10	15
K_{ϕ} , м/с	0,20	0,30	0,35

22.

d , см	3	7	11
K_{ϕ} , м/с	0,12	0,16	0,22

23.

d , см	4	8	12
K_{ϕ} , м/с	0,17	0,23	0,28

24.

d , см	6	12	18
K_{ϕ} , м/с	0,18	0,24	0,26

25.

d , см	2	6	10
K_{ϕ} , м/с	0,08	0,12	0,20

26.

d , см	3	6	11
K_{ϕ} , м/с	0,10	0,17	0,23

27.

d , см	5	10	15
----------	---	----	----

28.

d , см	3	7	11
----------	---	---	----

K_{ϕ} , м/с	0,17	0,24	0,28
------------------	------	------	------

K_{ϕ} , м/с	0,09	0,16	0,22
------------------	------	------	------

29.

d , см	7	12	17
K_{ϕ} , м/с	0,20	0,25	0,29

30.

d , см	10	15	20
K_{ϕ} , м/с	0,35	0,38	0,40

ЗАДАНИЕ 4. Построение полиномов таблично заданных функций при прогнозировании поведения тяжелых металлов в компонентах агроэкосистем.

Данные об изменении содержания подвижной формы тяжелых металлов (мг/кг) в растениях (зеленой массе) от содержания подвижной формы тяжелых металлов (мг/кг) в почве при изучении способов утилизации осадков сточных вод очистных сооружений заданы таблицей.

1 Вывести эмпирическую зависимость, описывающую содержание подвижной формы тяжелых металлов в зеленой массе и содержания подвижной формы тяжелых металлов в почве, используя метод наименьших квадратов. Построить график полученной эмпирической зависимости. Оценить погрешность аппроксимации.

2 Выполнить аппроксимацию полиномом второй степени и интерполирование сплайнами в MathCAD.

1.

Cd в почве (кислотность pH 4,30)	0,12	0,3	0,5	1,0	1,5	2,0	2,5	2,8	3,0	3,2
Cd в кукурузе	0,22	0,28	0,40	0,85	1,05	1,23	1,30	1,50	1,60	1,8

2.

Cd в почве (кислотность pH 4,30)	0,12	0,4	0,5	0,8	1	1,5	2,0	2,5	3,8	3,0
Cd в кукурузе	0,25	0,3	0,32	0,6	0,8	1,0	1,25	1,37	1,46	1,5

3.

Cd в почве (кислотность pH 5,20)	0,12	0,3	0,5	1,0	1,5	2,0	2,5	3,0	3,5	4,0
Cd в редьке масличной	0,12	0,30	0,45	0,93	1,15	1,32	1,80	2,05	2,25	2,44

4.

Cd в почве (кислотность pH 6,40)	0,12	0,3	0,5	1,0	1,5	2,0	2,5	3,0	3,5	4,0
Cd в редьке масличной	0,09	0,18	0,29	0,39	0,48	0,61	0,84	1,04	1,12	1,26

5.

Cd в почве (кислотность pH 6,40)	0,12	0,3	0,5	1,0	1,5	2,0	2,5	3,0	3,5	4,0
Cd в кукурузе	0,04	0,06	0,09	0,18	0,22	0,31	0,35	0,49	0,55	0,60

6.	Cd в почве (кислотность pH 5,20)	0,12	0,3	0,5	1,0	1,5	2,0	2,5	3,0	3,5	4,0
	Cd в кукурузе	0,06	0,09	0,13	0,26	0,54	0,80	0,89	0,97	1,05	1,12
7.	Cd в почве (кислотность pH 6,40)	0,12	0,3	0,5	1,0	1,5	2,0	2,5	3,0	3,5	4,0
	Cd в тимофеевке	0,05	0,12	0,21	0,35	0,22	0,44	0,50	0,78	1,00	1,24
8.	Cd в почве (кислотность pH 5,20)	0,12	0,3	0,5	1,0	1,5	2,0	2,5	3,0	3,5	4,0
	Cd в тимофеевке	0,06	0,19	0,28	0,55	0,72	0,86	1,02	1,22	1,45	1,69
9.	Pb в почве (кислотность pH 5,20)	30	50	60	70	90	100	110	120	130	150
	Pb в редьке масличной	0,63	0,68	0,73	0,75	0,79	0,82	0,84	0,86	0,88	0,91
10.	Pb в почве (кислотность pH 6,40)	30	50	60	70	90	100	110	120	130	150
	Pb в редьке масличной	0,30	0,37	0,42	0,47	0,52	0,54	0,57	0,60	0,63	0,70
11.	Pb в почве (кислотность pH 6,40)	30	50	60	70	90	100	110	120	130	150
	Pb в кукурузе	0,20	0,31	0,36	0,52	0,69	0,73	0,79	0,80	0,96	1,07
12.	Pb в почве (кислотность pH 5,20)	30	50	60	70	90	100	110	120	130	150
	Pb в кукурузе	0,40	0,82	0,93	1,32	1,72	1,95	2,25	2,40	3,16	3,92
13.	Pb в почве (кислотность pH 6,40)	30	50	60	70	90	100	110	120	130	150
	Pb в тимофеевке	0,38	0,46	0,53	0,67	0,72	0,75	0,79	0,81	0,89	1,01
14.	Pb в почве (кислотность pH 5,20)	30	50	60	70	90	100	110	120	130	150
	Pb в тимофеевке	0,41	0,56	0,62	0,69	0,78	0,89	1,02	1,28	1,45	1,78

15.	Сu в почве (кислотность рН 6,56)	25	40	50	60	75	90	100	120	130	150
	Сu в кукурузе	0,56	0,61	0,67	0,68	0,70	0,79	0,83	0,85	0,87	0,90
16.	Сu в почве (кислотность рН 4,50)	25	40	50	60	75	90	100	120	130	150
	Сu в кукурузе	0,92	0,94	0,95	0,97	0,99	1,09	1,15	1,26	1,32	1,48
17.	Сu в почве (кислотность рН 6,56)	25	40	50	60	75	90	100	120	130	150
	Сu в тимофеевке	1,43	1,54	1,68	1,82	1,97	2,14	2,27	2,32	2,40	2,49
18.	Сu в почве (кислотность рН 4,50)	25	40	50	60	75	90	100	120	130	150
	Сu в тимофеевке	1,79	1,93	2,11	2,20	2,28	2,43	2,61	2,97	3,15	3,43
19.	Zn в почве (кислотность рН 6,56)	75	90	100	110	125	140	150	200	250	300
	Zn в кукурузе	3,52	5,18	6,73	7,75	8,83	9,12	9,67	10,1	12,8	14,2
20.	Zn в почве (кислотность рН 4,50)	75	90	100	110	125	140	150	200	250	300
	Zn в кукурузе	26,3	33,3	39,8	42,7	46,5	64,5	72,9	78,6	80,6	85,5
21.	Zn в почве (кислотность рН 6,56)	75	90	100	110	125	140	150	200	250	300
	Zn в тимофеевке	14,1	15,3	16,6	17,5	18,3	19,5	21,5	25,6	30,1	35,5
22.	Zn в почве (кислотность рН 4,50)	75	90	100	110	125	140	150	200	250	300
	Zn в тимофеевке	24,5	30,3	33,9	40,7	42,5	47,3	49,4	67,6	83,6	99,8
23.	Сг в почве (кислотность рН 6,30)	25	50	75	100	125	150	175	200	250	300
	Сг в кукурузе	0,07	0,09	0,11	0,13	0,15	0,16	0,17	0,18	0,18	0,19

24.	Сг в почве (кислотность рН 5,30)	25	50	75	100	125	150	175	200	250	300
	Сг в кукурузе	0,09	0,11	0,14	0,15	0,17	0,18	0,19	0,21	0,23	0,25
25.	Сг в почве (кислотность рН 4,20)	25	50	75	100	125	150	175	200	250	300
	Сг в кукурузе	0,12	0,18	0,21	0,23	0,26	0,34	0,37	0,40	0,43	0,47
26.	Сг в почве (кислотность рН 6,30)	25	50	75	100	125	150	175	200	250	300
	Сг в тимофеевке	0,05	0,08	0,12	0,14	0,16	0,18	0,18	0,20	0,21	0,22
27.	Сг в почве (кислотность рН 5,30)	25	50	75	100	125	150	175	200	250	300
	Сг в тимофеевке	0,11	0,12	0,17	0,18	0,19	0,20	0,21	0,23	0,26	0,28
28.	Сг в почве (кислотность рН 4,20)	25	50	75	100	125	150	175	200	250	300
	Сг в редьке масличной	0,15	0,20	0,25	0,29	0,31	0,36	0,39	0,44	0,48	0,51
29.	Сг в почве (кислотность рН 6,30)	25	50	75	100	125	150	175	200	250	300
	Сг в редьке масличной	0,14	0,16	0,19	0,28	0,32	0,41	0,45	0,49	0,55	0,60
30.	Сг в почве (кислотность рН 5,30)	25	50	75	100	125	150	175	200	250	300
	Сг в редьке масличной	0,29	0,37	0,42	0,47	0,52	0,54	0,57	0,60	0,63	0,69

ЗАДАНИЕ 5. Определение производительности насосной станции второго подъема.

Рассмотрим систему: резервуар чистой воды (РЧВ) – насосная станция второго подъема (НС) – распределительная сеть водоснабжения (РСВ).

Вычислить производительность (тыс. м³) насосной станции второго подъема, содержащей два насоса и работающей ступенчато в системе: РЧВ – НС – РСВ.

Производительность насосной станция второго подъема определяется

следующим образом:

$$W = K_r k_{\text{ступ}} \int_{t_0}^{t_k} P_t(t) dt ,$$

где K_r – коэффициент неравномерности водопотребления (1,3–1,58),

$k_{\text{ступ}}$ – количество ступеней подачи воды ($k_{\text{ступ}} = 1$ или 2),

t_0 – начальный момент времени ($t_0 = 0$ ч),

t_k – конечный момент времени ($t_k = 24$ ч),

$P_t(t)$ – функция подачи насосами воды.

1	$\int_2^3 \frac{t-0,5}{\sqrt{t^2-1}} dt$	2	$\int_2^4 \frac{t+1}{\sqrt{t^2+1}} dt$	3	$\int_1^3 \frac{0,5t+2}{\sqrt{t^2+1}} dt$
4	$\int_2^3 \frac{t^2+2}{\sqrt{t^2+1}} dt$	5	$\int_0^1 \frac{t^2 dt}{\sqrt{t^2+1}}$	6	$\int_0^1 \frac{t^2+0,5}{\sqrt{t^2+1}} dt$
7	$\int_0^2 \frac{t+2}{\sqrt{t^2+1}} dt$	8	$\int_0^1 \frac{\sqrt{t^2+1}}{t+2} dt$	9	$\int_1^2 \frac{t+1}{\sqrt{t^2+2}} dt$
10	$\int_1^3 \frac{t dt}{\sqrt{t^2+3}}$	11	$\int_0^2 \frac{t dt}{\sqrt{t^2+1,8}}$	12	$\int_0^2 \frac{t^2 dt}{\sqrt{t+1}}$
13	$\int_1^2 \frac{\sqrt{t^2+2}}{t+2} dt$	14	$\int_0^1 \frac{t^2+1,4}{\sqrt{t^2+0,2}} dt$	15	$\int_0^2 \frac{t^2 dt}{\sqrt{t^2+4}}$
16	$\int_1^2 \frac{\sqrt{t^2+1}}{2t+2,5} dt$	17	$\int_1^3 \frac{t dt}{\sqrt{t^2+2}}$	18	$\int_0^1 \frac{t+0,5}{\sqrt{t^2+1,5}} dt$
19	$\int_0^2 \frac{t^2 dt}{\sqrt{t+4}}$	20	$\int_2^3 \frac{t+0,5}{\sqrt{t^2+1}} dt$	21	$\int_0^2 \frac{t dt}{\sqrt{t^2+2,5}}$
22	$\int_0^1 \frac{t^2 dt}{\sqrt{t+2}}$	23	$\int_1^3 \frac{t^2 dt}{\sqrt{t+1,7}}$	24	$\int_0^2 \frac{t dt}{\sqrt{t^2+2,4}}$
25	$\int_1^3 \frac{4-t}{\sqrt{t^2+1}} dt$	26	$\int_1^3 \frac{t^2 dt}{\sqrt{t+1,2}}$	27	$\int_1^3 \frac{t+3}{\sqrt{t^2+1}} dt$

$$28 \int_2^3 \frac{tdt}{\sqrt{t^2+1}}$$

$$29 \int_1^3 \frac{t+2}{\sqrt{t^2+1,4}} dt$$

$$30 \int_1^3 \frac{t+2,2}{\sqrt{t^2+1}} dt$$

ЗАДАЧА 6. Численные методы решения обыкновенных дифференциальных уравнений.

Уравнение свободных колебаний механической системы с одной степенью свободы с учётом сил трения имеет вид

$$\frac{d^2y}{dt^2} + k \frac{dy}{dt} + \omega^2 y = 0,$$

где k – коэффициент трения,

ω^2 – квадрат частоты колебаний.

Найти перемещения системы на $t \in \left[0; \frac{2\pi}{\omega}\right]$, удовлетворяющие

начальным условиям $y(0) = 0$; $y'(0) = 1$.

Расчёты выполнить для числа разбиений $n = 8$ и по результатам расчёта построить графики функций $y(t)$, $y'(t)$.

№ варианта	k	ω^2	№ варианта	k	ω^2
1	0,5	7	16	1,5	6
2	1	6	17	2	8
3	1,5	4	18	2,5	9
4	2	7	19	3	5
5	2,5	8	20	3,5	6
6	3	5	21	4	7
7	3,5	4	22	0,5	8
8	4	7	23	1	9
9	0,5	6	24	2	5
10	1	7	25	2,5	6
11	3	7	26	1,5	5
12	3,5	9	27	2	5
13	4	8	28	2,5	7
14	0,5	9	29	3	7
15	1	6	30	3,5	9

ПРИЛОЖЕНИЕ Б
(обязательное)

Учебная программа по дисциплине
«Численные методы»

- 1 Математическое моделирование и вычислительный эксперимент. Вычислительный алгоритм. Основные источники и типы погрешностей. Устойчивость. Корректность. Сходимость.
- 2 Прямые методы решения систем линейных алгебраических уравнений (СЛАУ): метод Гаусса, метод прогонки. Обращение матрицы. Итерационные методы решения СЛАУ: метод простой итераций, метод Зейделя. Сходимость итерационного процесса. Использование математического пакета MathCAD при решении СЛАУ.
- 3 Итерационные методы решения нелинейных уравнений: метод деления отрезка пополам, метод хорд, метод Ньютона (метод касательных). Сходимость итерационного процесса. Использование MathCAD при решении нелинейных уравнений.
- 4 Нахождение собственных значений и собственных векторов матриц. Методы решения полной проблемы и частичной проблемы собственных значений. Итерационные методы решения частичной проблемы собственных значений. Использование математического пакета MathCAD при нахождении собственных значений и собственных векторов матриц.
- 5 Приближение функций обобщёнными полиномами. Интерполирование. Метод неопределённых коэффициентов. Интерполяционные многочлены Лагранжа и Ньютона, тригонометрические полиномы. Интерполирование сплайнами. Примеры задач по специальности. Использование MathCAD при построении интерполяционных полиномов.
- 6 Подбор эмпирических формул. Определение параметров эмпирической зависимости. Метод наименьших квадратов. Локальное сглаживание данных. Примеры задач по специальности.
- 7 Численное дифференцирование. Аппроксимация производных. Погрешность численного дифференцирования. Использование интерполяционных формул Ньютона, Лагранжа. Дифференцирование в MathCAD.

- 8 Методы численного интегрирования: метод трапеций, метод Симпсона. Оценка погрешности численного интегрирования. Понятие о методе Монте-Карло. Примеры производственных задач. Интегрирование в MathCAD.
- 9 Методы решения дифференциальных уравнений. Численные методы решения задачи Коши для обыкновенных дифференциальных уравнений (ОДУ). Одношаговые и многошаговые методы. Понятие о явных и неявных методах. Метод Эйлера и метод Рунге-Кутты решения задачи Коши для ОДУ. Примеры задач по специальности. Решение ОДУ в MathCAD.
- 10 Уравнения с частными производными. Разностные схемы для дифференциальных уравнений в частных производных. Построение разностных схем. Сходимость, аппроксимация и устойчивость разностной схемы.

СПИСОК ЛИТЕРАТУРЫ

- 1 **Вержицкий, В. М.** Численные методы (линейная алгебра и нелинейные уравнения) : учеб. пособие / В. М. Вержицкий. – М. : Высшая школа, 2000. – 266 с.
- 2 **Вержицкий, В. М.** Численные методы (математический анализ и обыкновенные дифференциальные уравнения) : учеб. пособие / В. М. Вержицкий. – М. : Высшая школа, 2001. – 382 с.
- 3 Водозаборно-очистные сооружения и устройства : учеб. пособие / М. Г. Журба [и др.] ; под ред. М. Г. Журбы. – М. : Астрель, 2003. – 569 с.
- 4 **Воробьев, С. А.** Численные методы решения задач строительства на ЭВМ. Ч.1 / С. А. Воробьев. – Гомель : БелГУТ, 1992. – 47 с.
- 5 **Гуринович, А. Д.** Питьевое водоснабжение из подземных источников: проблемы и решения / А. Д. Гуринович. – М. : Альтекс-А, 2003. – 208 с.
- 6 **Железняков, Г. В.** Гидравлика и гидрология : учеб. пособие / Г. В. Железняков. – М. : Транспорт, 1989. – 376 с.
- 7 **Журба, В. М.** Численные методы (линейная алгебра и нелинейные уравнения) : учеб. пособие / В. М. Журба. – М. : Высшая школа, 2000. – 266 с.
- 8 **Кириянов, Д. В.** Самоучитель MathCAD 12 / Д. В. Кириянов. – СПб. : БХВ-Петербург, 2004. – 576 с.
- 9 **Лазеев, В. М.** Гидравлические расчеты неравномерного установившегося движения воды в открытых призматических руслах : метод. указания / В. М. Лазеев. – Гомель : БелГУТ, 1987. – 31 с.
- 10 **Ракитин, В. И.** Руководство по методам вычислений и приложения MathCAD : учеб. пособие / В. И. Ракитин. – М. : ФИЗМАТЛИТ, 2005. – 264 с.
- 11 Сборник студенческих научных работ. Вып.8 / под ред. В. И. Сенько. – Гомель : БелГУТ, 2002. – 149 с.
- 12 **Семененко, М. Г.** Математическое моделирование в MathCAD / М. Г. Семененко. – М. : Альтекс-А, 2003. – 208 с.
- 13 **Турчак, Л. И.** Основы численных методов : учеб. пособие / Л. И. Турчак, П. В. Плотников. – М. : ФИЗМАТЛИТ, 2002. – 304 с.
- 14 **Формалев, В. Ф.** Численные методы : учеб. пособие / В. Ф. Формалев, Д. Л. Ревизников. – М. : ФИЗМАТЛИТ, 2004. – 400 с.

МИНИСТЕРСТВО ОБРАЗОВАНИЯ РЕСПУБЛИКИ БЕЛАРУСЬ
УЧРЕЖДЕНИЕ ОБРАЗОВАНИЯ
«БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ ТРАНСПОРТА»

Кафедра «Прикладная математика»

И. Н. КРАВЧЕНЯ

ПРИМЕНЕНИЕ ЧИСЛЕННЫХ МЕТОДОВ ПРИ РЕШЕНИИ ЗАДАЧ ВОДОСНАБЖЕНИЯ И ВОДООТВЕДЕНИЯ

Учебно-методическое пособие для студентов
строительного факультета специальности «Водоснабжение,
водоотведение и охрана водных ресурсов»

Одобрено методической комиссией строительного факультета

Гомель 2009

Учебное издание

КРАВЧЕНЯ Ирина Николаевна

**ПРИМЕНЕНИЕ ЧИСЛЕННЫХ МЕТОДОВ ПРИ РЕШЕНИИ ЗАДАЧ
ВОДОСНАБЖЕНИЯ И ВОДООТВЕДЕНИЯ**

Учебно-методическое пособие для студентов
строительного факультета специальности «Водоснабжение,
водоотведение и охрана водных ресурсов»

Редактор И. И. Эвентов
Технический редактор В. Н. Кучерова

Подписано в печать 26.10.2009 г. Формат 60 × 84 ¹/₁₆.
Бумага офсетная. Гарнитура Таймс. Печать на ризографе.
Усл. печ. л. 5,58. Уч.-изд. л. 4,68. Тираж 100 экз.
Зак. № . Изд. № 78

Издатель и полиграфическое исполнение
Белорусский государственный университет транспорта:
ЛИ № 02330/0552508 от 09.07.2009 г.
ЛП № 02330/0494150 от 03.04.2009 г.
246653, г. Гомель, ул. Кирова, 34.