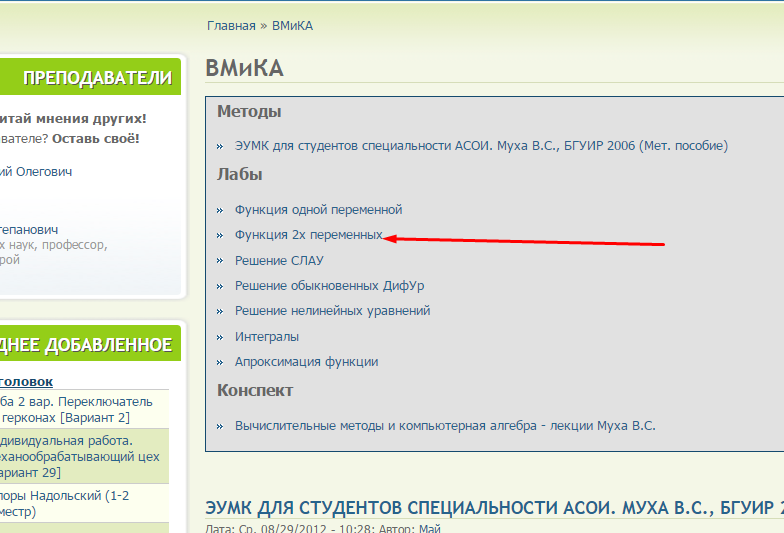
Здесь находятся уже реализованные варианты нужно подставить данные. http://bsuir-helper.ru/predmet/vmika



Выполните первые 4 лабораторные работы из "Лабораторного практикума" по данной дисциплине. Две лабораторные работы выполняются в счет одной контрольной работы. **Вам выдается вариант 50 по таблице 1.0.** Отчет по каждой работе присылайте в виде отдельной папки, содержащей программы (m-файлы), отвечающие на каждый пункт «порядка выполнения работы». Имена файлов должны содержать номер пункта задания, например, файл с именем prog4\_3\_1.m решает задачи пункта 4.3.1 задания по лабораторной работе 4. Если работа не зачтена, т.е. по ней имеются замечания, то работа присылается повторно целиком, включая и зачтенные вопросы (новая папка). Перед проверкой старая папка заменяется новой.

Таблица 1.0

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Последние две цифры зачетки | Вариант работы 1 по таблицам 1.1, 1.2 | Вариант работы 2 по таблице 2.1 | Вариант работы 3 по таблице 1.1 | Вариант работы 4 по таблице 1.1 |
| [ 50]+ | '1.05a' | '2.13б' | '3.14б' | '4.05б' |

## Лабораторная работа № 1. Работа в системе Matlab

**1.1. Цель работы**

* + 1. Ознакомление с системой Matlab, приобретение навыков работы.
    2. Ознакомление с языком программирования Matlab.
    3. Приобретение навыков программирования на языке Matlab.

**1.2. Порядок выполнения работы**

**1.2.1.** Ознакомиться с системой Matlab, ее запуском и работой в ней по методическому пособию "Введение в Matlab" [1].

**1.2.2.** Разработать m-файл-сценарий для вывода в графическое окно графика функции одной переменной с помощью программы **plot**. Функцию взять из табл. 1.1 в соответствии с номером своей бригады и кодом подгруппы (а или б). Функцию оформить в виде m-файла-функции.

**1.2.3.** Разработать m-файл-сценарий для вывода в одно графическое окно контурных графиков двух функций двух переменных на уровне ,  с помощью программ **meshgrid** и **contour**. Функции взять из табл. 1.2 в соответствии с номером своей бригады и кодом подгруппы. Функции оформить в виде m-файлов-функций.

**1.2.4.** Разработать m-файл-сценарий для вывода в графическое окно графика функции двух переменных с помощью программ **meshgrid**, **mesh** и **meshс** для одной из функций табл. 1.2 в соответствии с номером своей бригады и кодом подгруппы. Функцию оформить в виде m-файла-функции.

Таблица 1.1

Функции одной переменной для индивидуальных заданий

|  |  |  |  |
| --- | --- | --- | --- |
| № вари-анта | Функция | № вари-анта | Функция |
| 5а |  |  |  |

Таблица 1.2

Функции двух переменных для индивидуальных заданий

|  |  |  |  |
| --- | --- | --- | --- |
| № вари-анта | Функции | № вари-анта | Функции |
| 1 | 2 | 3 | 4 |
| 1а |  | 1б |  |
| 2а |  | 2б |  |
| 3а |  | 3б |  |
| 4а |  | 4б |  |
| 5а |  | 5б |  |
| 6а |  | 6б |  |
| 7а |  | 7б |  |
| 8а |  | 8б |  |
| 9а |  | 9б |  |
| 10а |  | 10б |  |
| 11а |  | 11б |  |
| 12а |  | 12б |  |
| 13а |  | 13б |  |
| 14а |  | 14б |  |
| 15а |  | 15б |  |

## Лабораторная работа № 2. Решение систем линейных алгебраических уравнений

**2.1. Цель работы**

* + 1. Изучение методов решения систем линейных алгебраических уравнений (СЛАУ).
    2. Приобретение навыков программирования методов Гаусса и Гаусса–Зейделя.
    3. Приобретение навыков использования стандартных средств системы Matlab для решения СЛАУ.

**2.2. Теоретические положения**

**2.2.1.** Постановка задачи

Системой линейных алгебраических уравнений (СЛАУ) называется следующая система равенств

 (2.1)

которая при некоторых значениях переменных  превращается в систему тождеств. Решить данную систему – это значит по известным коэффициентам системы , и правым частям , найти значения переменных , при которых эти равенства превращаются в тождества.

Часто систему (2.1) записывают в векторно-матричной форме. Для этого вводят в рассмотрение квадратную ()-матрицу коэффициентов системы , и векторы-столбцы неизвестных , , и правой части системы , ,

, , .

Тогда система уравнений (2.1) записывается в виде

. (2.2)

Эта запись совпадает по форме с линейным уравнением , решением которого является . Аналогичную простую формулу можно записать и для решения векторно-матричного уравнения (2.2). Если определитель , то система имеет единственное решение

, (2.3)

где  – матрица, обратная матрице . Известно также правило Крамера для решения СЛАУ (2.1), в соответствии, с которым неизвестные определяются по формуле:

, (2.4)

где  – определитель матрицы , а ** – определитель матрицы , в которую вместо коэффициентов  при  подставлены свободные члены .

Однако решение СЛАУ с помощью обратной матрицы (2.3) или с помощью правила Крамера (2.4) является достаточно трудоемким. Известны более рациональные численные методы решения СЛАУ, рассмотренные ниже. Вообще все методы решения СЛАУ можно разделить на конечные и итерационные. Конечные методы позволяют получить решение с определенной точностью за известное заранее конечное число операций. В итерационных методах число операций заранее не определено. Оно зависит от точности, с которой необходимо получить решение. К конечным методам решения СЛАУ относится метод исключения Гаусса, а к итерационным – метод Гаусса–Зейделя.

**2.2.2.** Метод Гаусса для решения СЛАУ

Метод Гаусса в решении СЛАУ (2.1) состоит из двух этапов: исключение переменных (прямой ход) и нахождение решения (обратный ход).

*Прямой ход* состоит из  шагов. На первом шаге исключается неизвестная  из всех уравнений, начиная со второго. На втором шаге исключается  из всех уравнений, начиная с третьего. На -м шаге исключается , из всех уравнений, начиная  уравнения. На последнем ()-м шаге исключается  из последнего уравнения. В результате выполнения прямого хода мы получаем систему уравнений с так называемой верхней треугольной матрицей коэффициентов.

*Обратный ход* позволяет последовательно получить неизвестные системы уравнений. Сначала определяют  из последнего -го уравнения. Затем это значение подставляют в ()-е уравнение и определяют , и т. д., до определения  из первого уравнения.

Опишем более подробно шаги прямого хода. На первом шаге -е уравнение начиная с  преобразуется следующим образом. Вводится коэффициент

,

и из -го уравнения вычитается 1-е уравнение, умноженное на этот коэффициент. Результирующее уравнение записывается на место -го. В результате из -го уравнения исключается переменная . После этого шага система уравнений примет следующий вид:



где – коэффициенты, полученные на первом шаге прямого хода. Они определяются следующими выражениями:

,

.

На втором шаге -е уравнение начиная с  преобразуется следующим образом. Вводится коэффициент

,

и из -го уравнения вычитается 2-е уравнение, умноженное на этот коэффициент. Результирующее уравнение записывается на место -го. В результате из -го уравнения исключается переменная . После второго шага система уравнений примет следующий вид:



где – коэффициенты, полученные на втором шаге прямого хода. Они определяются выражениями

,

.

Вообще, на -м шаге -е уравнение начиная с  преобразуется следующим образом. Вводится коэффициент

, (2.5)

и из -го уравнения вычитается -е уравнение, умноженное на этот коэффициент. Результирующее уравнение записывается на место -го. В результате из -го уравнения исключается переменная . Коэффициенты системы уравнений на -м шаге пересчитываются по формулам:

; (2.6)

, (2.7)

.

При  происходит исключение  из последнего уравнения и окончательная верхняя треугольная система записывается следующим образом:



Теперь выполняется *обратный ход*. Видно, что из последнего уравнения можно сразу определить ,

.

Подставляя это значение в предпоследнее уравнение, находим ,

.

Для нахождения любой переменной  применяется формула

.

*Замечание.* В процессе решения СЛАУ легко может быть получен определитель системы . Он равен произведению диагональных элементов матрицы верхней треугольной системы:

.

Метод исключения Гаусса требует приблизительно  ячеек памяти и выполнения приблизительно  арифметических операций.

Метод Гаусса реализуется по схеме, приведенной на рис. 2.1., в случае, когда блок № 5 “Выбор главного элемента” пропускается.

**2.2.3.** Метод Гаусса с выбором главного элемента

Основные вычисления в методе исключений Гаусса выполняются по формулам (2.6), (2.7). Эти формулы позволяют проследить накопление погрешностей в процессе вычислений.

Обозначим  относительную погрешность, содержащуюся в коэффициенте ,  – относительные погрешности округления соответственно при делении, умножении и вычитании. Тогда для относительной погрешности  вычисления коэффициента  можно получить следующее выражение:

,

откуда получаем формулу для оценки абсолютной погрешности :

.

Если предположить, что погрешности ,  не превышают некоторой величины , то получим следующую формулу для оценки погрешности:

.

Из последнего выражения видно, что погрешность вычисления коэффициента  в основном определяется первым слагаемым в скобках этого выражения и уменьшается с уменьшением . Чтобы  было по возможности меньшим, необходимо чтобы  было по возможности большим. Поэтому перед выполнением шага исключения каждой переменной желательно переставить уравнения системы так, чтобы

,

потому что тогда

.

Если в методе Гаусса выполняется такая перестановка, то метод называется методом Гаусса с выбором главного элемента. Этот метод имеет меньшую погрешность при определении решения СЛАУ. Метод Гаусса с выбором главного элемента реализуется с помощью схемы, приведенной на рис. 2.1. Схема блока № 5 “Выбор главного элемента” приведена на рис. 2.2.

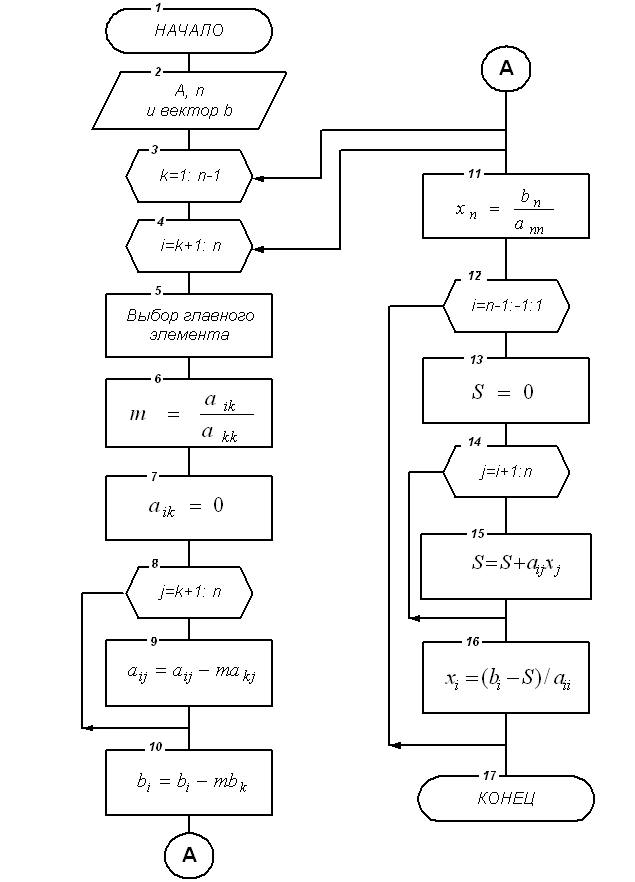


Рис. 2.1.



Рис. 2.2.

**2.2.4.** Метод Гаусса–Зейделя

Метод Гаусса–Зейделя – это итерационный метод решения задачи, или метод последовательных приближений. Дадим описание этого метода. Пусть решается система уравнений (2.1). Выразим из 1-го уравнения , из 2-го уравнения  и т.д. В результате получим

,

,

…………………………………………….

,

или вообще для любого 

. (2.8)

Предположим, что на некоторой -й итерации мы получили решение . Используем известные к моменту расчета  значения других переменных в правой части уравнения (2.8) для расчета значения  в левой части. В результате мы получим следующую рекуррентную формулу, которая и составляет метод Гаусса–Зейделя:

.

Расчеты по последней формуле продолжаются при  до тех пор, пока не будет выполняться условие

,

или условие

,

где , причем  – требуемая абсолютная погрешность нахождения решения СЛАУ, а  – требуемая относительная погрешность.

Итерационные методы могут сходиться к решению или не сходиться. То же самое относится и к методу Гаусса–Зейделя. Метод Гаусса–Зейделя сходится, если выполняются условия



для всех , кроме одного, для которого выполняется менее жесткое условие

.

Данное условие является достаточным, но не необходимым. Возможны случаи невыполнения данного условия при сходимости метода.

Содержательный смысл приведенных условий сходимости поясним на примере системы двух уравнений с двумя неизвестными

.

Достаточные условия сходимости для этой системы имеют вид

 и ,

или

 и .

Данные условия означают, что диагональные элементы системы уравнений по абсолютной величине превышают недиагональные. Такой же смысл имеют и условия сходимости для системы из  уравнений. В связи с этим часто для обеспечения сходимости метода Гаусса–Зейделя бывает достаточно поменять местами уравнения системы. Схема алгоритма для решения системы линейных уравнений итерационным методом Гаусса–Зейделя представлена на рис. 2.3.



Рис. 2.3.

**2.2.5.** Средства Matlab для решения СЛАУ

Для работы с матрицами в Matlab применяются общепринятые для матричной алгебры символы: +(плюс) или -(минус) для сложения или вычитания матриц, \* (звездочка) для умножения матриц. Для возведения матрицы в степень используется символ ^, для транспонирования матрицы – символ ' (кавычка).

Например, для получения обратной матрицы можно использовать запись

. (2.9)

Это дает возможность получить решение СЛАУ с помощью обратной матрицы в виде (2.3). Для этого достаточно записать команду

. (2.10)

Кроме того, в Matlab имеется функция  для обращения квадратной матрицы . С помощью этой функции можно найти обратную матрицу, записав вместо (2.9) команду

,

или решить СЛАУ, записав вместо (2.10) команду

.

Однако решение СЛАУ с помощью обратной матрицы связано с большим объемом вычислений. Более рациональным является использование операций матричного деления:

/ (наклонная черта или slash) для правого деления,

\ (обратная наклонная черта или backslash) для левого деления.

Запись  означает левое деление матрицы  на матрицу . По смыслу это то же, что и , однако расчеты выполняются по-другому. Запись



означает решение СЛАУ (2.2) методом исключения Гаусса.

Запись  означает правое деление матрицы  на матрицу . По смыслу это то же, что и , однако расчеты выполняются по-другому. Более точно,  (см. левое деление). Запись



дает решение уравнения  методом исключения Гаусса.

**2.3. Порядок выполнения работы**

**2.3.1.** Написать m-файл-сценарий для решения СЛАУ -го порядка (2.1) методом Гаусса без выбора главного элемента. Работу программы продемонстрировать на системе уравнений, выбранной из табл. 2.1 в соответствии с номером своей бригады и кодом подгруппы (а или б). Правильность решения подтвердить путем использования средств Matlab.

**2.3.2.** Написать m-файл-сценарий для решения СЛАУ -го порядка методом Гаусса с выбором главного элемента. Работу программы продемонстрировать на той же системе уравнений. Правильность решения подтвердить путем использования средств Matlab.

**2.3.3.** Написать m-файл-сценарий для решения СЛАУ -го порядка методом Гаусса–Зейделя. Работу программы продемонстрировать на той же системе уравнений. Правильность решения подтвердить путем использования средств Matlab.

Таблица 2.1

Системы уравнений для индивидуальных заданий

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| № вари-анта | | Матрица системы | | Вектор  правой части | |
| 1 | | 2 | | 3 | |
| 13б | | -5 5 -2  -2 3 -3  0 2 2 | | -28  -12  -8 |

## Лабораторная работа № 3. Аппроксимация функций

|  |  |
| --- | --- |
| № вари-анта | Функция |

|  |  |
| --- | --- |
| 14б |  |

**3.1. Цель работы**

**3.1.1.** Ознакомление с задачей аппроксимации функций одной переменной, изучение задачи интерполирования функций.

**3.1.2.** Приобретение навыков программирования интерполяционных формул.

**3.1.3.** Приобретение навыков использования стандартных средств системы Matlab для интерполирования функций.

**3.2. Теоретические положения**

**3.2.1.** Понятие аппроксимации функций

Аппроксимация функции  – это замена этой функции другой более простой функцией , близкой к  в некотором смысле. Критерий близости функций  и  определяет способ аппроксимации.

Если расстояние  между функциями  и  на некотором отрезке  действительной прямой определить выражением

,

то аппроксимация функции  по критерию минимума такого расстояния  будет называться аппроксимацией с минимальной интегральной квадратичной погрешностью.

Если критерий близости функций  и  состоит в том, чтобы  и  совпадали в дискретном ряде точек  отрезка , то такой способ аппроксимации функции  называется *интерполированием* функции .

Если расстояние  между функциями  и  на некотором отрезке  действительной прямой определить выражением

,

то аппроксимация функции  по критерию минимума такого расстояния  будет называться аппроксимацией по методу наименьших квадратов.

**3.2.2.** Постановка задачи интерполирования функций

Задача интерполирования функции  на некотором отрезке  формулируется следующим образом. На отрезке  задано  точек , которые называют узлами. Обычно считают, что первая и последняя точки совпадают с концами отрезка : , . Известны значения  функции  в этих точках, . Требуется заменить эту функцию некоторой другой функцией  таким образом, чтобы значения обеих функций совпадали в узлах, т.е. чтобы выполнялись равенства

.

Искомой неизвестной в данной задаче является функция .

Сформулированную задачу иногда интерпретируют следующим образом. Некоторая функция  задана на отрезке  таблицей своих значений

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  |  |  |  | … |  |
|  |  |  |  | … |  |

и требуется найти способ определения значений этой функции в любых других точках отрезка .

При формулировке задачи интерполирования обычно предполагают, что аппроксимирующая функция  задана с точностью до  параметров , т.е. в виде . Тогда нам необходимо отыскать неизвестные параметры  исходя из заданных равенств

. (3.1)

Эти равенства можно рассматривать как систему  уравнений относительно неизвестных коэффициентов .

Чаще всего функцию  представляют в виде полинома -й степени

. (3.2)

Тогда система уравнений (3.1) принимает следующий вид:

 (3.3)

Определитель этой системы имеет вид



и называется определителем Вандермонда на системе точек . Доказано, что если точки  попарно различны, что предполагается при постановке задачи, то определитель Вандермонда не равен нулю. Тогда система уравнений (3.3) имеет единственное решение, т.е. существует единственный полином (3.2) степени , коэффициенты которого удовлетворяют системе уравнений (3.3). Этот полином называется интерполяционным полиномом для функции .

**3.2.3.** Интерполяционный полином Лагранжа

Интерполяционный полином может быть представлен в различных формах. Одной из них является форма Лагранжа. Полином Лагранжа имеет следующий вид:

,

или в компактной форме

. (3.4)

Этот полином легко может быть получен путем следующих рассуждений. Рассмотрим так называемый полином влияния -го узла:



. (3.5)

Понятно, что это полином -й степени. Из выражения (3.5) видно, что он равен нулю в любых точках, кроме , а в точке  равен 1. Вид этого полинома приведен на рис. 3.1. Произведение  есть полином -й степени, который во всех узлах, кроме , равен нулю, а в узле  равен . Просуммировав произведения  по всем  (по всем узлам ), мы получим полином (3.4), удовлетворяющий постановке задачи интерполирования. Следовательно, построенный полином будет интерполяционным.

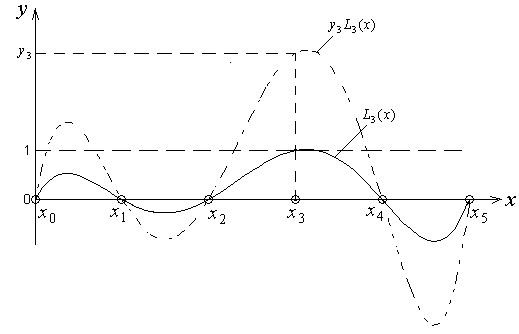


Рис. 3.1.

Полином Лагранжа можно записать в другом виде. Для этого рассмотрим полином

. (3.6)

Это полином -й степени со старшим коэффициентом, равным 1, и обращающийся в нуль во всех узлах. Легко показать, что производная от этого полинома в точке  имеет вид

.

Тогда полином влияния -го узла (3.5) можно записать в виде

,

а сам полином Лагранжа (3.4) – в виде

. (3.7)

**3.2.4.** Погрешность интерполирования

Погрешность интерполяционного полинома Лагранжа оценивается выражением

,

где  – максимальное по модулю значение -й производной функции  на отрезке интерполирования ,  – полином (3.6).

**3.2.5.** Наилучший выбор узлов интерполирования

Чаще всего узлы интерполирования располагают на отрезке интерполирования с равномерным шагом. Вместе с тем, выбирая неравномерную сетку узлов, можно увеличить точность интерполирования. Задача о наилучшем выборе узлов интерполирования была решена Чебышевым. Наилучшие узлы интерполирования выбираются равными корням так называемого "полинома, наименее отклоняющегося от нуля" на отрезке интерполирования . Полином, наименее отклоняющийся от нуля на отрезке , был найден Чебышевым и назван его именем. Полином Чебышева  определяется следующим выражением:

.

Следовательно, наилучшие узлы интерполирования  на отрезке  выбираются из условия



и имеют следующие значения:

. (3.8)

Наилучшие узлы интерполирования на произвольном отрезке  находятся по формуле

, (3.9)

где  – узлы (3.8).

**3.2.6.** Средства Matlab для интерполирования функций

Для интерполирования функций в Matlab можно использовать функции **polyfit** и **polyval**.

Функция **a=polyfit(x, y, n)** находит массив коэффициентов  длиной  полинома степени  по массивам длиной  узлов  и значений функции в узлах , . Этот полином аппроксимирует функцию  по методу наименьших квадратов. Предполагается, что полином задается в виде

.

Если , то программа возвращает коэффициенты интерполяционного полинома.

Функция **y=polyval(a,x)** возвращает значение  полинома в точке , коэффициенты которого определены в векторе .

Таким образом, для интерполирования функции необходимо последовательно обратиться к функциям **polyfit** и **polyval**.

**3.3. Порядок выполнения работы**

**3.3.1.** Написать m-файл-функцию для интерполирования функции  на отрезке  с помощью полинома Лагранжа (3.4) при равномерной сетке узлов. Входными параметрами m-файла-функции должны быть массив узлов интерполирования, массив значений интерполируемой функции в узлах и аргумент , при котором вычисляется значение полинома Лагранжа (контрольная точка). Выходной параметр m-файла-функции – значение интерполяционного полинома в точке .

**3.3.2.** Использовать написанную m-файл-функцию для интерполирования конкретной функции, взятой из табл. 1.1 лабораторной работы № 1 в соответствии с номером своей бригады и кодом подгруппы. Результаты интерполирования представить в виде графиков интерполируемой функции и интерполяционного полинома в одном графическом окне. Число узлов выбрать равным 3–5. Шаг между узлами интерполирования выбрать равномерным, шаг по переменной  при построении графиков функций выбрать кратным шагу между узлами интерполирования с тем, чтобы можно было наблюдать значения функции и полинома в узлах. Исследовать зависимость погрешности интерполирования от количества узлов интерполирования.

**3.3.3.** Выполнить интерполирование заданной функции с помощью стандартных средств Matlab и сравнить с результатами, полученными по собственным программам.

**3.3.4.** Выполнить интерполирование заданной функции с наилучшим выбором узлов интерполирования по формуле (3.9). Результаты сравнить с результатами, полученными на равномерной сетке узлов.

## Лабораторная работа № 4. Численное интегрирование

|  |  |
| --- | --- |
| № вари-анта | Функция |
| 5б |  |

**4.1. Цель работы**

**4.1.1.** Изучение задачи численного интегрирования функций.

**4.1.2.** Приобретение навыков программирования квадратурных формул.

**4.1.3.** Приобретение навыков использования стандартных средств системы Matlab для численного интегрирования функций.

**4.2. Теоретические положения**

**4.2.1.** Постановка задачи численного интегрирования

Задача численного интегрирования состоит в том, чтобы найти численное значение определенного интеграла

, (4.1)

где  – функция, непрерывная на отрезке интегрирования . Формулы для решения этой задачи называются квадратурными. Квадратурная формула позволяет вместо точного значения интеграла (4.1) найти некоторое его приближенное значение . Разность точного и приближенного значений интеграла называется абсолютной погрешностью квадратурной формулы (или численного метода),

.

Квадратурные формулы используют для вычисления интеграла (4.1) значения функции  в ряде точек отрезка . Рассмотрим различные квадратурные формулы и их погрешности.

**4.2.2.** Методы прямоугольников

Разобьем отрезок интегрирования  на  *частей* точками , как это показано на рис. 4.1. Заменим площадь криволинейной трапеции суммой площадей прямоугольников, построенных на частичных отрезках , как на основаниях. Если высоту -го прямоугольника взять равной значению функции  в левой точке основания прямоугольника, т.е. принять

,

то мы получим квадратурную формулу левых прямоугольников

.

Для равноотстоящих на величину  узлов,

,

формула левых прямоугольников имеет вид

. (4.2)

Если интеграл на -м отрезке  заменить площадью прямоугольника с высотой, равной значению функции  в правой точке основания прямоугольника, т.е. принять

,

то мы получим квадратурную формулу правых прямоугольников

.

Для равноотстоящих на величину  узлов формула правых прямоугольников имеет вид:

. (4.3)

Абсолютная погрешность метода прямоугольников для равномерной сетки значений аргумента оценивается неравенством

,

где  – максимальное по модулю значение первой производной подынтегральной функции  на отрезке интегрирования .

**4.2.3.** Метод трапеций

Заменим площадь криволинейной трапеции суммой площадей трапеций, построенных на частичных отрезках , (см. рис. 4.1),

,

где

.

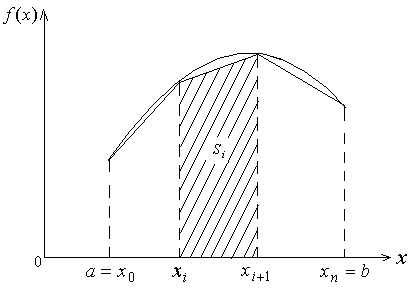


Рис. 4.1.

Получим квадратурную формулу трапеций:

.

Для равноотстоящих на величину  узлов формула трапеций имеет вид

. (4.4)

Погрешность метода трапеций оценивается неравенством

,

где ** –** максимальное по модулю значение второй производной подынтегральной функции  на отрезке интегрирования .

**4.2.4.** Метод Симпсона

Разделим точки , разбивающие отрезок интегрирования  на частичные отрезки с равномерным шагом , на тройки точек , ,…, . Для такого разбиения число  необходимо выбрать четным. На отрезке, определяемом -й тройкой точек , , заменим подынтегральную функцию параболой второго порядка , проходящей через точки , , , и заменим точное значение интеграла на этом отрезке интегралом  от полученной параболы. Можно показать, что

.

Приближенное значение интеграла получим как сумму этих частичных интегралов:



. (4.5)

Погрешность метода Симпсона оценивается неравенством

,

где ** –** максимальное по модулю значение четвертой производной подынтегральной функции  на отрезке интегрирования .

**4.2.5.** Интерполяционные квадратурные формулы наивысшей алгебраической степени точности

Рассмотрим вычисление следующего интеграла

, (4.6)

где  – некоторая интегрируемая функция, которая называется весовой,  – функция, которую назовем подынтегральной. Этот интеграл является более общим по сравнению с рассматриваемым ранее интегралом (4.1). Интеграл вида (4.1) мы получим из (4.6) при весовой функции .

Для вычисления интеграла (4.6) применим следующий подход: выберем на отрезке   точек . В отличие от предыдущих методов, не будем вычислять интегралы на частичных отрезках, а заменим подынтегральную функцию на всем отрезке  интерполяционным полиномом (3.7), построенным по узлам . В результате получим следующую квадратурную формулу

, (4.7)

где

, (4.8)

.

Формула (4.7), в которой коэффициенты определяются по выражению (4.8), называется *интерполяционной квадратурной формулой*. Эта формула точна для подынтегральных функций , представляющих собой полиномы до -й степени включительно. В этом случае говорят, что степень точности интерполяционной квадратурной формулы (4.7) равна .

Точность интерполяционной квадратурной формулы (4.7) можно существенно увеличить путем рационального выбора узлов . Рекомендуется выбирать узлы  равными корням полиномов, ортогональных на  с весом . Интерполяционная квадратурная формула (4.7) с таким выбором узлов называется интерполяционной квадратурной формулой наивысшей алгебраической степени точности. Степень точности этой формулы равна . Мы видим, что рациональным выбором узлов мы увеличиваем точность интерполяционной квадратурной формулы более чем в 2 раза. Интерполяционная квадратурная формула (4.7) наивысшей алгебраической степени точности называется также квадратурной формулой Гаусса–Кристоффеля. Коэффициенты  (веса) (4.8) этой формулы называют числами Кристоффеля. Оптимальные узлы  и соответствующие им веса  рассчитываются заранее. Существуют таблицы узлов и весовых коэффициентов для различных весовых функций  [8,9].

Приведем примеры квадратурных формул наивысшей алгебраической степени точности.

4.2.5.1. Интегрирование функции по конечному отрезку

Интеграл с единичной весовой функцией  на конечном отрезке , т.е. интеграл вида

,

линейной заменой переменных



приводится к виду

.

На отрезке  ортогональны с весом  полиномы Лежандра

 .

Узлы  квадратурной формулы в этом случае выбираются равными корням полинома Лежандра . Квадратурная формула имеет вид

. (4.9)

В табл. 4.1 в качестве примера приведены узлы и коэффициенты для этой формулы при использовании двух, трех и четырех узлов.

Таблица 4.1.

Узлы и коэффициенты квадратурной формулы Гаусса–Лежандра

|  |  |  |
| --- | --- | --- |
| Число узлов **()** | Значения узлов | Значения весовых коэффициентов |
| 2 |  |  |
| 3 |  |  |
| 4 |  |  |

4.2.5.2. Интегрирование функции по положительной полуоси

Пусть нужно вычислить интеграл вида

,

т.е. интеграл с весовой функцией . На полуоси  ортогональны с весом  полиномы Лагерра

 .

Узлы  квадратурной формулы в этом случае выбираются равными корням полинома Лагерра . Квадратурная формула имеет вид

. (4.10)

В табл. 4.2 приведены узлы и коэффициенты для этой формулы при использовании двух, трех и четырех узлов.

Таблица 4.2.

Узлы и коэффициенты квадратурной формулы Гаусса–Лагерра

|  |  |  |
| --- | --- | --- |
| Число узлов **()** | Значения узлов | Значения весовых коэффициентов |
| 2 |  |  |
| 3 |  |  |
| 4 |  |  |

4.2.5.3. Интегрирование функции по всей действительной прямой

Пусть нужно вычислить интеграл

,

т.е. интеграл с весовой функцией . На действительной прямой ортогональны с весом  полиномы Эрмита

 .

Узлы  квадратурной формулы в этом случае выбираются равными корням полинома Эрмита . Квадратурная формула имеет вид

. (4.11)

В табл. 4.3 приведены узлы и коэффициенты для этой формулы при использовании двух, трех и четырех узлов.

Таблица 4.3.

Узлы и коэффициенты квадратурной формулы Гаусса–Эрмита

|  |  |  |
| --- | --- | --- |
| Число узлов **()** | Значения узлов | Значения весовых коэффициентов |
| 2 |  |  |
| 3 |  |  |
| 4 |  |  |

**4.2.6.** Средства Matlab для численного интегрирования

Для численного интегрирования в Matlab можно использовать функции **trapz, cumtrapz, quad,** **quad8**.

**z = trapz(x,y)** вычисляет значение интеграла от функции  методом трапеций по массиву **x** узлов и массиву **y** значений функции  в узлах.

**z1 = trapz(y)** вычисляет значение интеграла от функции  методом трапеций по массиву **y** значений функции  в узлах, используя единичный шаг по аргументу x. Для получения значения интеграла при другом шаге необходимо полученное значение интеграла **z1** умножить на шаг.

**zc = cumtrapz(x,y)** вычисляет накапливающиеся (текущие) значения интеграла от функции  методом трапеций по массиву **x** узлов и массиву **y** значений функции  в узлах.

**zc1 = cumtrapz(y)** вычисляет накапливающиеся (текущие) значения интеграла от функции  методом трапеций по массиву **y** значений функции  в узлах, используя единичный шаг по аргументу x. Для получения значений интеграла при другом шаге необходимо полученные значения интеграла (массив **zc1)** умножить на шаг.

**q = quad('fun',a,b)** возвращает значение **q** определенного интеграла от заданной функции **'fun'** на отрезке **[a,b]**, полученное по адаптивному рекурсивному методу Симпсона с относительной погрешностью 10-3. Подынтегральная функция должна оформляться в виде m-файла-функции **y=fun(x)**. Функция **fun** должна быть построена таким образом, чтобы она могла возвращать массив **y** значений выходной величины, если **x** – массив значений аргумента. Например, если подынтегральная функция имеет вид , то можно оформить следующую m-файл-функцию:

function y=fun(x)

y=x.^2;

**q = quad('fun',a,b,tol)** возвращает значение **q** определенного интеграла, полученное с относительной погрешностью **tol**. Можно также использовать вектор, состоящий из двух элементов **tol=[rel\_tol, abs\_tol]** для определения комбинации относительной и абсолютной погрешностей.

**q = quad('fun',a,b,tol, trace)** возвращает значение **q** определенного интеграла, полученное с относительной погрешностью **tol**, и при значении **tol**, не равном нулю, строит точечный график интегрируемой функции.

**q = quad('fun',a,b,tol, trace, p1,p2,…)** возвращает значение **q** определенного интеграла c использованием указанных выше возможностей, а также параметров **p1,p2,…**, которые напрямую передаются в определенную подынтегральную функцию **y = fun(x,p1,p2,…)**.

Если достигается чрезмерный уровень рекурсии, то функция **quad** возвращает q = Inf, что может означать сингулярность интеграла.

Функция **quad8** выполняет интегрирование по формулам Ньютона–Котеса 8-го порядка. Она имеет те же параметры, что и функция **quad:**

**q = quad8('fun',a,b,tol,trace,p1,p2,…)**

**4.3. Порядок выполнения работы**

**4.3.1.** Написать m-файлы-функции для интегрирования функции  на отрезке  методами левых прямоугольников, правых прямоугольников, трапеций, Симпсона при равномерной сетке узлов. Входными параметрами m-файла-функции должны быть массив узлов интегрирования и массив значений интегрируемой функции в узлах. Выходной параметр m-файл-функции – значение интеграла.

**4.3.2.** Использовать написанные m-файл-функции для интегрирования конкретной функции, взятой из табл. 1.1 лабораторной работы № 1 в соответствии с номером своей бригады и кодом подгруппы. Сравнить результаты интегрирования различными методами при одном и том же шаге интегрирования.

**4.3.3.** Выполнить интегрирование заданной функции с помощью стандартных средств Matlab и сравнить с результатами, полученными по собственным программам.

**4.3.4.** Написать m-файлы-функции для вычисления интегралов вида (4.6) по квадратурным формулам наивысшей алгебраической степени точности Гаусса–Лежандра (4.9), Гаусса–Лагерра (4.10) и Гаусса–Эрмита (4.11), используя для этого 3 или 4 узла. С их помощью вычислить эти интегралы для своей функции .