ГОСУДАРСТВЕННОЕ УЧРЕЖДЕНИЕ
ВЫСШЕГО ПРОФЕССИОНАЛЬНОГО ОБРАЗОВАНИЯ
«БЕЛОРУССКО-РОССИЙСКИЙ УНИВЕРСИТЕТ»

Кафедра «Электропривод и автоматизация
промышленных установок»

ОСНОВЫ КОМПЬЮТЕРНОГО МОДЕЛИРОВАНИЯ

Методические рекомендации к лабораторным работам для студентов направления подготовки 13.03.02 «Электроэнергетика и электротехника» дневной формы обучения

Могилев 2018

УДК 658.26
ББК 31.19
О 75

Рекомендовано к изданию
учебно-методическим отделом
Белорусско-Российского университета

Одобрено кафедрой «Электропривод и автоматизация промышленных установок» «7» февраля 2018 г., протокол № 7

Составитель О. А. Капитонов

Рецензент канд. техн. наук, доц. А. М. Кургузиков

Методические рекомендации к лабораторным работам для студентов направления подготовки 13.03.02 «Электроэнергетика и электротехника» дневной формы обучения.

Учебно-методическое издание

ОСНОВЫ КОМПЬЮТЕРНОГО МОДЕЛИРОВАНИЯ

Ответственный за выпуск Г. С. Леневский

Технический редактор А. А. Подошевко

Компьютерная верстка Н. П. Полевничая

Подписано в печать . Формат 60×84/16. Бумага офсетная. Гарнитура Таймс.
Печать трафаретная. Усл. печ. л. . Уч.-изд. л. . Тираж 46 экз. Заказ №

Издатель и полиграфическое исполнение:
Государственное учреждение высшего профессионального образования
«Белорусско-Российский университет».
Свидетельство о государственной регистрации издателя,
изготовителя, распространителя печатных изданий
№ 1/156 от 24.01.2014.
Пр. Мира, 43, 212000, Могилев.

Содержание

Введение

Целью преподавания дисциплины «Основы компьютерного моделирования» является обучение студентов основным методам компьютерного моделирования и анализа электромеханических систем, умение студента ориентироваться в современных программных продуктах и применять полученные навыки при решении практических задач.

Курс является необходимым для профиля подготовки «Электрооборудование автомобилей и тракторов», в научной и практической деятельности бакалавра.

Студент, изучивший дисциплину, научится:

– анализировать и принимать решение по применению того или иного программного обеспечения для разработки и исследования моделей электротехнических объектов;

– разрабатывать модели электротехнических объектов;

– составлять модели исследуемых электротехнических объектов;

– записывать программные модели для персонального компьютера;

– проводить вычислительный эксперимент с моделями на персональном компьютере;

– самостоятельно применять современные компьютерные системы для разработки и исследования моделей электротехнических объектов.

Студент, изучивший дисциплину, овладеет:

– навыками работы с программным обеспечением;

– знаниями о перспективах развития компьютерных систем моделирования и анализа электромеханических систем;

– знаниями для разработки и исследования моделей электротехнических объектов.

1 Лабораторная работа № 1. Моделирование динамических электротехнических систем на персональном компьютере

Цель работы: изучить основные методы моделирования переходных процессов в электрических цепях на персональном компьютере.

1.1 Порядок выполнения работы

1.1.1 Получить у преподавателя индивидуальное задание – схему электрическую принципиальную для моделирования.

1.1.2 Составить математическое описание схемы электрической принципиальной при помощи уравнений Кирхгофа.

1.1.3 Разработать компьютерную модель схемы в среде C++ Builder.

1.1.4 Выполнить компьютерное моделирование, получить зависимости переменных состояния модели от времени.

1.1.5 Оформить отчет о лабораторной работе.

1.2 Основные теоретические положения

1.2.1 Основные понятия, термины и определения.

Объект – это все то, на что конкретно направлена деятельность человека.

Научный прогресс в любой области развивается в следующей последовательности:

1) наблюдение;

2) эксперимент;

3) теоретические исследования;

4) организация производственных процессов.

При этом особая роль отводится гипотезам, которые являются предсказаниями, основанными на опытных данных, наблюдении и догадках. Проверка правильности гипотез достигается путем проведения специального эксперимента. Как правило, современная гипотеза строится по аналогии с проверенными на практике научными положениями.

Аналогия – суждение о каком-либо сходстве двух объектов. В простейшем случае она является пропорцией. Аналогия бывает существенной и несущественной.

Важную роль при проведении эксперимента отводится моделям – логическим схемам, заменяющим объект-оригинал при изучении его определенных свойств. Это позволяет значительно упростить эксперимент и более глубоко изучить исследуемые свойства.

Система – совокупность взаимосвязанных элементов, взаимодействующая с окружением как единое целое и обладающая свойствами, которыми не обладают отдельно взятые элементы.

Модель – объект произвольной формы, отражающий существенные для рассматриваемой задачи свойства объекта-оригинала.

Моделирование, согласно [1], – использование системы обработки данных для представления характера изменения выбранных параметров физической или абстрактной системы.

Моделирование – процесс замены объекта-оригинала моделью с целью получения информации о некоторых изучаемых свойствах оригинала путем проведения эксперимента. Оно занимает промежуточное положение между аналитическим исследованием и натурным экспериментом. Теория моделирования изучает методы построения моделей и проведения с их помощью экспериментов.

Объект моделирования – объект или явление, который выступает предметом исследования.

Эмуляция – использование системы обработки данных для имитации другой системы обработки данных таким образом, что имитирующая система получает те же данные, выполняет те же операции и достигает тех же результатов, что и имитируемая система. Примечание: эмуляция обычно выполняется с помощью аппаратных или программно-аппаратных средств.

Используется следующая классификация моделей по способу представления:

1) концептуальная модель – это идеальный образ, зависящий не только от объективных свойств, но и субъективных знаний и других факторов разработчика модели;

2) символьные (знаковые) модели – это дальнейшее развитие концептуальной модели. Отражают исследуемые свойства с помощью символов. Дополнительно подразделяются на:

– лингвистические (вербальные) модели – это описание свойств на некотором естественном языке, например, техническое задание, пояснительная записка и т. д.;

– графические модели. Отражают свойства модели в графической форме и бывают двух типов: портретные (иконические), отражающие графическими средствами реальные свойства объекта (чертежи, планы местности, схемы), и условные, графически отображающие непосредственно ненаблюдаемые свойства (графики, диаграммы);

– математические модели – описание исследуемых свойств и параметров модели средствами математики;

3) материальные модели – реализация в материальной форме концептуальной или символьной модели, позволяющая производить эксперимент. Их дополнительно классифицируют на:

– геометрические модели. Используются для исследования геометрических свойств объектов и систем без учета внутренней природы (макеты, муляжи и т. п.);

– физические модели. Используются для исследования свойств исследуемого объекта с сохранением его физической природы или внутреннего состава (гидродинамические модели судов, аэродинамические модели, химические лабораторные установки);

– аналоговые модели. В отличие от физических позволяют производить исследования без сохранения физической природы внутренних процессов объекта, заменяя ее по аналогии более удобной для исследования (например, в АВМ все различные процессы моделируются электрическими сигналами);

– программные (имитационные) модели. Позволяют производить вычислительный эксперимент с помощью компьютеров.

При моделировании исследуемого объекта или явления наиболее часто используется системный подход.

Система – это совокупность взаимосвязанных элементов, взаимодействующая с окружением как единое целое и обладающая свойствами, которыми не обладают отдельно взятые элементы.

Когда объект моделирования рассматривается как система, то характеризуется следующим:

– набором отдельных составных элементов, выполняющих определенные функции;

– структурой – способ организации элементов в системе с помощью внутренних связей;

– окружающей средой, которая взаимодействует с системой через ее входы и выходы;

– историей развития системы.

В зависимости от исследуемых свойств системы различают два основных типа моделей:

1) функциональные, отражающие основные свойства функционирования системы в окружающей среде как единого целого без рассмотрения ее внутренней структуры;

2) структурные – когда исследуется особенность строения системы и взаимодействие элементов внутри системы.

Параметр – величина, характеризующая свойства и режим работы исследуемого объекта. Различают внутренние и внешние параметры систем. Первые характеризуют свойства исследуемого объекта, а вторые – свойства его элементов. Внешние параметры – это параметры внешней среды, влияющие на функционирование исследуемого объекта.

По степени детализации исследуемых свойств системы различают три иерархических уровня абстрагирования моделей:

1) метауровень – модели верхнего уровня абстракции;

2) макроуровень – средний уровень абстрагирования;

3) микроуровень – нижний уровень абстракции.

Сложность модели возрастает при переходе от верхнего уровня к нижнему.

1.2.2 Математическое описание технических систем на метауровне.

Метауровень является верхним, наиболее общим уровнем абстрагирования модели. При этом в большинстве случаев вся исследуемая система рассматривается в качестве единого «черного ящика», без учета внутренней структуры. Метауровень позволяет исследовать только функциональные модели систем при их взаимодействии с внешней средой. Модели метауровня обычно используют для описания внутренних подсистем отдельных функционально законченных компонентов внутри модели макроуровня.

Модели метауровня также используются при анализе результатов эксперимента и построении экспериментальных (факторных) моделей систем. Модель метауровня может в равной степени быть как динамической (если определяется в составе модели независимая переменная времени), так и статической.

Для описания непрерывных динамических моделей метауровня наиболее удобно использовать преобразования Лапласа и вид передаточной функции системы, которая определяет отношение изображения выходного сигнала к изображению входного сигнала исследуемого канала системы. В общем случае передаточная функция имеет вид отношения двух полиномов:

$$W(p)=\frac{Y(p)}{X(p)}=\frac{a_m p^m + a_{m\text{-}1}p^{m\text{-}1}+\ldots+a_1 p+a_0}{b_m p^m + b_{m\text{-}1}p^{m\text{-}1}+\ldots+b_1 p+b_0}, \tag{1.1}$$

где $Y(p)$ – изображение выходного сигнала;

$X(p)$ – изображение входного сигнала;

p – оператор Лапласа (аналог производной).

Использование оператора Лапласа позволяет заменить операции взятия производной и неопределенного интеграла соответственно математическими операциями умножения и деления.

Для статических непрерывных систем используется описание на основе системы линейных или нелинейных уравнений, связывающих зависимости выходных координат системы от величины внешних воздействий.

1.2.3 Порядок составления математической модели коммутации электрической системы.

При составлении математического описания электрической системы последовательно проводится выполнение следующих этапов:

– составляется список переменных состояния электрической схемы: значения токов и напряжений участков электрической цепи;

– на основании 1-го и 2-го законов Кирхгоффа составляются уравнения, описывающие переменные состояния для послекоммутационной схемы;

– система уравнений преобразуется в вид обыкновенных дифференциальных уравнений в нормальной форме Коши;

– определяются начальные условия переменных состояния левых частей дифференциальных уравнений на основе докоммутационной электрической схемы.

Содержание отчета

Отчет о выполнении лабораторной работы должен содержать следующее.

1 Титульный лист.

2 Цель работы.

3 Описание хода выполнения работы по пп. 1.1.1–1.1.5.

4 Вывод.

Отчет оформляется на персональном компьютере в текстовом редакторе.

Контрольные вопросы

1 Дайте определение понятий «наблюдение», «эксперимент».
2 Что такое модель?
3 Какие уровни абстрагирования моделей вы можете выделить?
4 Изложите порядок составления математической модели коммутации электрической системы.

2 Лабораторная работа № 2. Изучение методов численного интегрирования дифференциальных уравнений

Цель работы: изучить основные методы численного интегрирования дифференциальных уравнений.

2.1 Порядок выполнения работы

2.1.1 Получить у преподавателя индивидуальное задание – дифференциальное уравнение, численный метод интегрирования и интервал, на котором необходимо провести интегрирование.
2.1.2 Разработать программу в среде C++ Builder для решения индивидуального задания.
2.1.3 Методом компьютерного моделирования получить искомую в дифференциальном уравнении функцию в табличном виде.
2.1.4 Оформить отчет о лабораторной работе.

2.2 Основные теоретические положения

2.2.1 Общая характеристика численных методов интегрирования.
Решением обыкновенного дифференциального уравнения (ОДУ) в нормальной форме Коши

$$dy / dx = \varphi(x, y) \tag{2.1}$$

является функция $y = f(x)$, удовлетворяющая начальным условиям – точке $y = f(x)$. При использовании численного метода интегрирования решение уравнения получается в виде массива точек (x_i, y_i), описывающих кривую, проходящую через точку начальных условий (x_i, y_i).

Различают три группы численных методов решения дифференциальных уравнений:

1) явные методы численного интегрирования;

2) неявные методы численного интегрирования;

3) методы прогноза и коррекции.

Явные методы численного интегрирования основаны на использовании формулы разложения в ряд Тейлора в окрестностях точки (x_n, y_n) для расчета каждого последующего значения искомой функции на базе данных о ее предыдущих значениях

$$y_{n+1} = y_n + hf'(x_n, y_n) + f''(x_n, y_n)h_2/2 + \ldots + f(n)(x_n, y_n)h_n/n!,$$

$$x_{n+1} = x_n + h, \tag{2.2}$$

где x_n, y_n – значения аргумента и функции на предыдущем шаге интегрирования;

x_{n+1}, y_{n+1} – значения аргумента и функции на последующем шаге интегрирования;

h – величина шага интегрирования;

$f(n)(x_n, y_n)$ – производная n-й степени.

В зависимости от степени производной, учитываемой численным методом, определяется степень точности метода. Шаг интегрирования может быть постоянным на всем расчетном интервале аргумента х или изменяться в зависимости от величины ошибки определения функции. Различают следующие явные методы интегрирования: одношаговые, многошаговые.

2.2.2 Метод Эйлера.

Является наиболее простым и имеет первый порядок точности, поэтому на практике используется редко. Последующее значение функции реализуется при помощи выражения

$$y_{n+1} = y_n + h\varphi(x_n, y_n). \tag{2.3}$$

Схема алгоритма данного метода с учетом вышеприведенного описания исходных данных и результатов расчета показана на рисунке 2.1.

2.2.3 Метод Рунге-Кутта четвертого порядка точности.

Последующее значение функции определяется при помощи выражения

$$y_{n+1} = y_n + (k1 + 2k2 + 2k3 + k4)/6, \tag{2.4}$$

где коэффициенты вычисляются по формулам

$$k1 = h\varphi(x_n, y_n);$$

$$k2 = h\varphi\left(x_n + \frac{h}{2}, y_n + \frac{k1}{2}\right);$$

$$k3 = h\varphi\left(x_n + \frac{h}{2},\ y_n + \frac{k2}{2}\right);$$

$$k4 = h\varphi\left(x_n + h,\ y_n + k3\right). \tag{2.5}$$

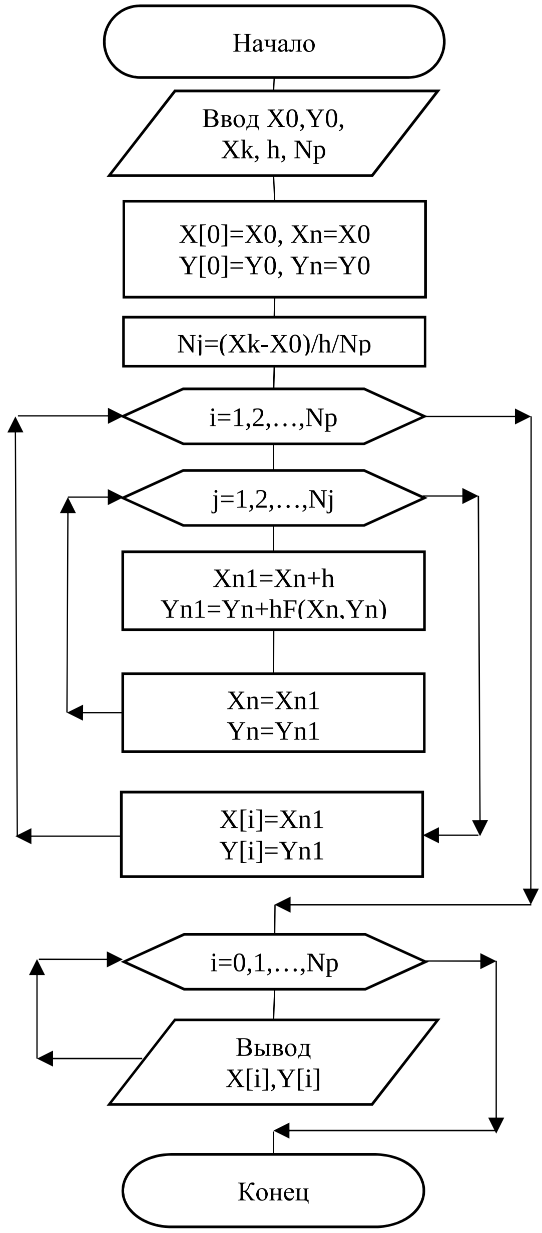

Рисунок 2.1 – Алгоритм метода Эйлера

2.2.4 Метод Рунге-Кутта пятого порядка точности. Основан на использовании следующего соотношения:

$$y_{n+1} = y_n + \left(23k1 + 125k3 - 81k5 + 125k6\right)/192, \tag{2.6}$$

где коэффициенты рассчитываются по выражениям

$$k1 = h\varphi(x_n,\ y_n);$$
$$k2 = h\varphi(x_n +\ h/3,\ y_n + k1/3);$$
$$k3 = h\varphi(x_n + 2h/5,\ y_n + (4k1 + 6\cdot k2\ )/25);$$
$$k4 = h\varphi(x_n + h,\ y_n + (k1 - 12k2 + 15k3)/4);$$
$$k5 = h\varphi(x_n + 2h/3,\ y_n + (6k1 + 90k2 - 50k3 + 8k4)/81);$$
$$k6 = h\varphi(x_n + 4h/5,\ y_n + (6k1 + 26k2 + 10k3 + 8k4)/75). \qquad (2.7)$$

2.2.5 Многошаговые методы численного интегрирования.

Данные методы используют для определения каждой последующей точки не одно, а несколько значений функции в предыдущих точках интегрирования. В отличие от одношаговых методов многошаговые не обладают свойством самостартования, поэтому перед запуском вычислений необходимо рассчитать требуемое число первых точек искомой функции при помощи одношагового метода, а только потом по этим значениям реализовать процесс интегрирования многошаговым методом.

Исходные данные и результаты расчета при многошаговом методе интегрирования аналогичны данным одношаговых методов. Однако для определения текущих значений точек искомой функции в процессе интегрирования необходимо сформировать два одномерных массива: для аргумента – *MX* и функции – *MY*. В этих массивах первый (нулевой) элемент соответствует текущей искомой n точке, первый – предыдущей n – 1, второй – n – 2 точке, которая стоит впереди текущей на два шага интегрирования, третий – n – 3 точке, отстоящей на три шага интегрирования, и так до последней точки, учитываемой конкретным методом.

2.2.6 Методы Адамса.

Используют следующие выражения для численного интегрирования с различной степенью точности:

$$y_{n+1} = y_n + h\varphi(x_n,\ y_n);$$
$$y_{n+2} = y_{n+1} + 0{,}5h[3\varphi(x_{n+1},\ y_{n+1}) - \varphi(x_n,\ y_n)];$$
$$y_{n+3} = y_{n+2} + h[23\varphi(x_{n+2}, y_{n+2}) - 16\varphi(x_{n+1}, y_{n+1}) + 5\varphi(x_n, y_n)]/12;$$
$$y_{n+4} = y_{n+3} + h[55\varphi(x_{n+3}, y_{n+3}) - 59\varphi(x_{n+2}, y_{n+2}) +$$
$$+ 37\varphi(x_{n+1}, y_{n+1}) - 9\varphi(x_n, y_n)]/24;$$
$$y_{n+5} = y_{n+4} + h[1901\varphi(x_{n+4}, y_{n+4}) - 2774\varphi(x_{n+3}, y_{n+3}) +$$
$$+ 2616\varphi(x_{n+2}, y_{n+2}) - 1274\varphi(x_{n+1}, y_{n+1}) + 251\varphi(x_n, y_n)]/720. \qquad (2.8)$$

Схема алгоритма реализации численного интегрирования по методу Адамса четвертого порядка точности приведена на рисунке 2.2.

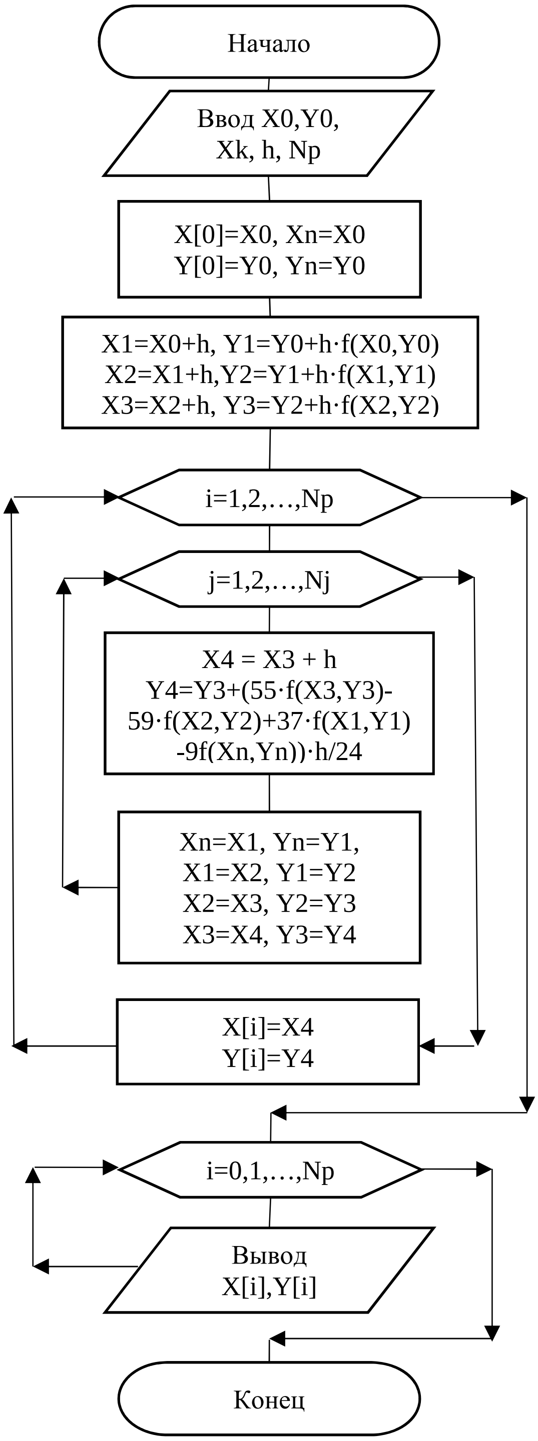

Рисунок 2.2 – Схема алгоритма метода Адамса

2.2.7 Многошаговые методы Нистрема.

Основываются на следующих формулах для численного интегрирования с различной степенью точности:

$$y_{n+2} = y_n + 2h\varphi\left(x_{n+1},\ y_{n+1}\right);$$
$$y_{n+3} = y_{n+1} + h\,[7\varphi\left(x_{n+2}, y_{n+2}\right) - 2\varphi\left(x_{n+1}, y_{n+1}\right) + \varphi\left(x_n, y_n\right)]/3;$$

$$y_{n+4} = y_{n+2} + h\,[8\varphi(x_{n+3}, y_{n+3}) - 5\varphi(x_{n+2}, y_{n+2}) +$$
$$+ 4\varphi(x_{n+1}, y_{n+1}) - \varphi(x_n, y_n)]/3. \quad (2.9)$$

2.2.8 Многошаговые методы Милна.
Реализуются в зависимости от степени точности следующими выражениями:

$$y_{n+4} = y_n + 4h\,[2\varphi(x_{n+3}, y_{n+3}) - \varphi(x_{n+2}, y_{n+2}) + 2\varphi(x_{n+1}, y_{n+1})]/3;$$

$$y_{n+6} = y_n + 0{,}3h\,[11\varphi(x_{n+5}, y_{n+5}) - 14\varphi(x_{n+4}, y_{n+4}) + 26\varphi(x_{n+3}, y_{n+3}) -$$
$$-14\varphi(x_{n+2}, y_{n+2}) + 11\varphi(x_{n+1}, y_{n+1})]. \quad (2.10)$$

Метод Милна по схеме «3/8» описывается выражением

$$y_{n+4} = y_{n+1} + 3h\,[7\varphi(x_{n+3}, y_{n+3}) - 3\varphi(x_{n+2}, y_{n+2}) +$$
$$+ 5\varphi(x_{n+1}, y_{n+1}) - \varphi(x_n, y_n)]/8. \quad (2.11)$$

Содержание отчета

Отчет о выполнении лабораторной работы должен содержать следующее.
1 Титульный лист.
2 Цель работы.
3 Описание хода выполнения работы по пп. 2.1.1–2.1.4.
4 Вывод.

Отчет оформляется на персональном компьютере в текстовом редакторе.

Контрольные вопросы

1 Дайте общую характеристику численных методов интегрирования.
2 Поясните алгоритм метода Эйлера.
3 Поясните алгоритм метода Рунге-Кутта.
4 Поясните алгоритм метода Милна.
5 Поясните отличие явных методов от неявных.

3 Лабораторная работа № 3. Моделирование трансцендентных функций

Цель работы: изучить основные методы поиска корней трансцендентных уравнений.

3.1 Порядок выполнения работы

3.1.1 Получить у преподавателя индивидуальное задание – трансцендентную функцию, численный метод поиска нуля функции, интервал поиска.

3.1.2 Составить программу в среде С++ Builder для решения индивидуального задания.

3.1.3 Методом компьютерного моделирования найти значение аргумента, обращающее заданную функцию в ноль.

3.1.4 Оформить отчет о лабораторной работе.

3.2 Основные теоретические положения

3.2.1 Общая характеристика численных методов решения нелинейных уравнений.

Задачей решения нелинейного алгебраического уравнения вида

$$f(x) = 0; \quad y = f(x) \tag{3.1}$$

является поиск его корней. Существует две группы методов решения нелинейных алгебраических уравнений: прямые и итерационные. Первые используют ряд математических выражений, вычисление которых дает сразу точное решение, например, решение квадратного уравнения. Вторая, основная, группа методов реализует циклический процесс постепенного уточнения приблизительного значения, пока не достигается заданная точность.

Методы поиска действительных корней реализуют поиск точки пересечения функции вида $y = f(x)$ с действительной осью на заданном интервале поиска. В ряде случаев, если интервал поиска не задан, то реализуется алгоритм отделения корней, позволяющий выделить интервалы изменения аргумента x, где находится один из действительных корней.

Отделение действительных корней производится с целью определения числа действительных корней и нахождения для каждого из них интервала поиска аргумента, содержащего на концах значения функций различных знаков. Для этого организуется процесс расчета значений функции

$$Y = F(X) \tag{3.2}$$

через равные интервалы аргумента X, начиная с минимально возможного значения до максимального. При этом фиксируются значения начала X_1 и конца X_2 для каждого интервала, если на нем функция изменила знак.

3.2.2 Численный метод дихотомии поиска действительных корней уравнений.

Суть метода заключается в постепенном сужении интервала поиска действительного корня [x_1, x_2] путем определения средней точки

$$x_{cp} = \left(x_2 + x_1\right) / 2 \tag{3.3}$$

и сравнения модуля $|x_{cp}|$ с точностью поиска (рисунок 3.1). Если оно велико, то производится переопределение интервала поиска путем присвоения значения x той границе интервала, значение функции которой имеет одинаковый знак с $f(x_{cp})$. После этого повторяют вычисление до тех пор, пока модуль значения X не станет меньше точности поиска.

Алгоритм данного метода включает следующие операции.

1 Формируется подпрограмма $F(X)$ и вводятся исходные данные: начальное значение а и конечное значение b интервала поиска, точность определения корней E.

2 Определяется среднее значение аргумента на интервале:

$$X_c = \left(X_1 + X_2\right) / 2. \tag{3.4}$$

3 Определяются значения функции (3.2) в точках a, b и X_c:

$$Y_1 = F(a); \quad Y_2 = F(b); \quad Y_c = F(X_c). \tag{3.5}$$

4 Сравнивается величина модуля значения функции в средней точке интервала с заданной точностью и если выполняется условие

$$|Y_c| < E, \tag{3.6}$$

тогда прекращается выполнение итераций и значение X_c считается корнем уравнения для заданной точности; если условие не выполняется, то продолжается выполнение программы.

5 Сравниваются знаки функций Y_1 и Y_c. Если они совпадают, тогда задается новое значение $X_1 = X_c$, иначе $X_2 = X_c$ и возвращается процесс в п. 2 для следующей итерации.

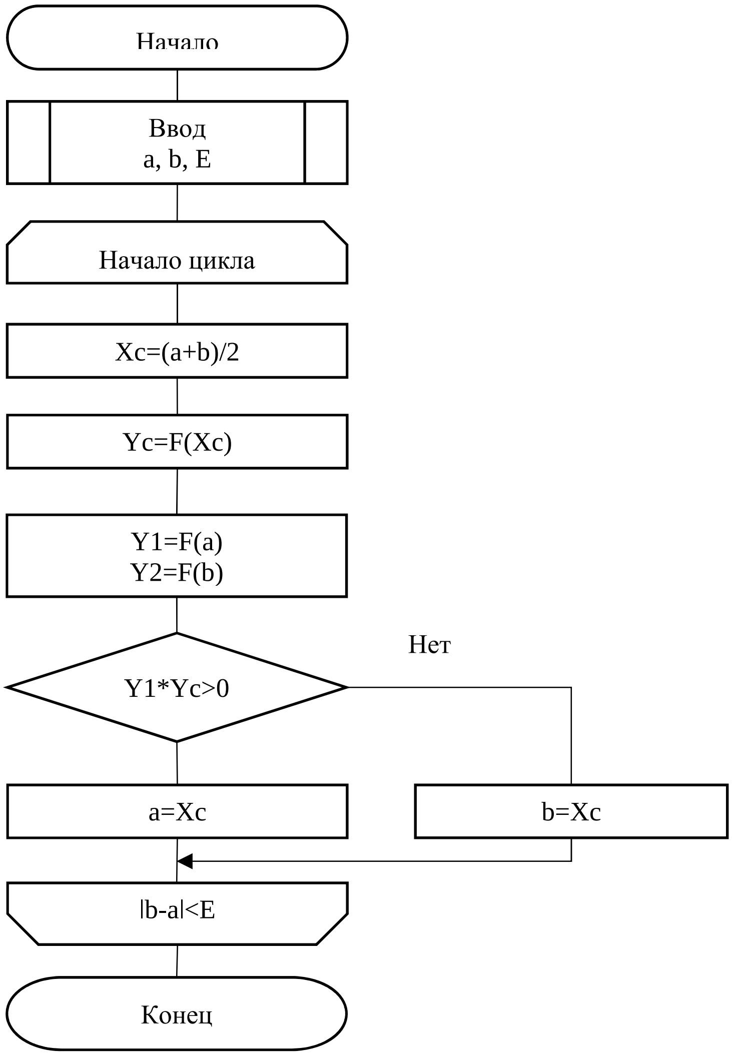

Рисунок 3.1 – Алгоритм дихотомии

3.2.3 Численные методы Ньютона и секущих для поиска действительных корней уравнений.

Метод касательных (Ньютона) не требует поиска интервала аргумента с разными знаками функций на его концах, а только одно начальное значение аргумента, которое выбирается как значение одной из границ (рисунок 3.2). Недостатком метода является то, что необходимо иметь выражение производной для правой части уравнения

$$df(x) = f'(x), \tag{3.7}$$

которое оформляется в виде подпрограммы функции $dF(X)$. Вместо заданного аналитического выражения производной возможно ее определение при помощи численного метода.

Каждое следующее уточненное значение корня при выполнении итераций определяется по формуле

$$x_{n+1} = x_n - f(x_n) / df(x_n), \tag{3.8}$$

где x_n, x_{n+1} – значения аргумента на предыдущем и на текущем шаге итерации.

Для выбора одной из границ интервала следует определить выпуклость функции по формуле второй производной.

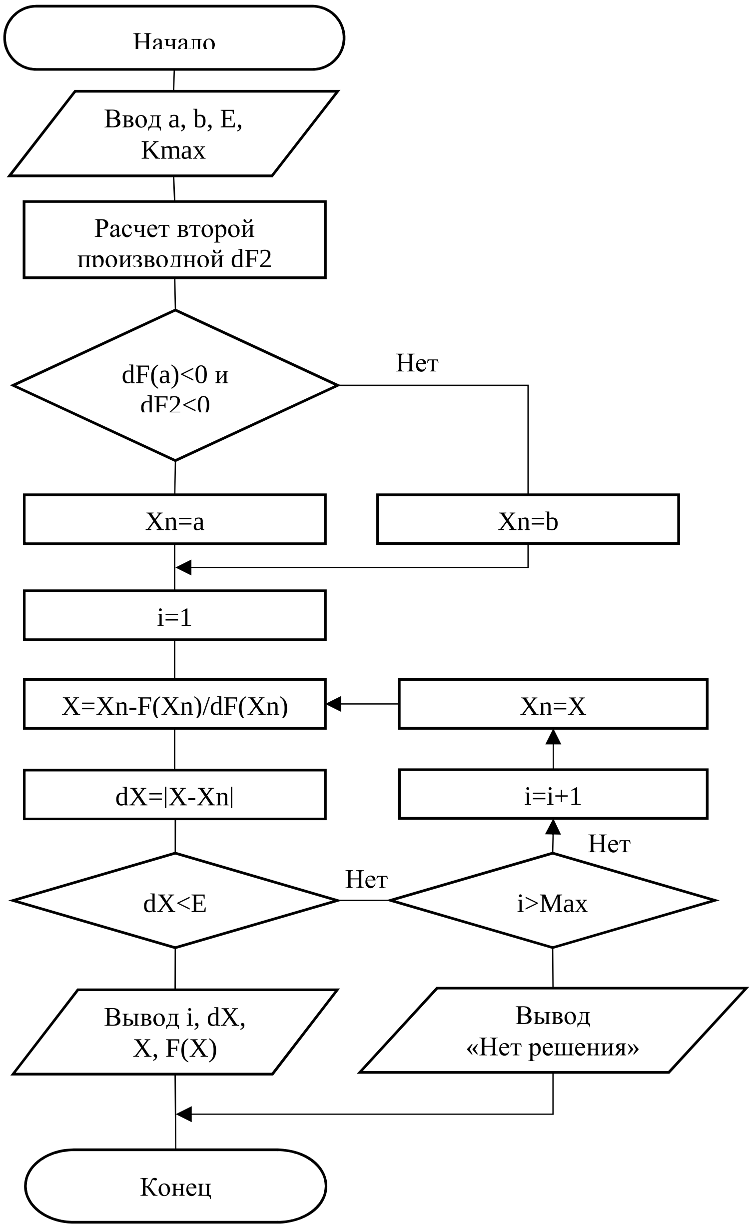

Рисунок 3.2 – Алгоритм метода Ньютона

Алгоритм метода содержит следующие операции:

1) ввод исходных данных: границ интервала, точности определения корня e, максимального числа итераций поиска max;

2) для выбора начальной точки определение второй производной функции по приближенной формуле

$$dY_2 = \frac{f'(b) - f'(a)}{b - a}; \tag{3.9}$$

3) проверка соблюдения одновременно двух условий: $f'(a) < 0$ и $dY_2 < 0$. Если это выполняется, тогда в качестве первой точки принимается граница a, иначе – b;

4) назначение номера текущей итерации $i = 1$;

5) определение следующего значения аргумента по

$$X = X_n - F\left(X_n\right) / dF\left(X_n\right); \tag{3.10}$$

6) сравнивание величины модуля значения функции в точке аргумента x_{n+1} с заданной точностью. Если выполняется условие

$$\left|X - X_n\right| \le E, \tag{3.11}$$

тогда прекращается выполнение итераций и значение X считается корнем уравнения для заданной точности; если условие не выполняется, то продолжается выполнение программы;

7) сравнение значения текущей итерации с максимальным. Если оно превышает, тогда выполнение программы прекращается и выдается сообщение «Нет решения»; если условие не выполняется, то продолжается выполнение программы;

8) увеличение номера итерации;

9) переопределение значения $X_n = X$;

10) передача управления на 5 для следующей итерации. Для параболического вида функций используется ускоренный метод Ньютона-Рафсона, который имеет вид:

$$X_N = X_{N-1} - \frac{Mf(X_{N-1})}{f'(X_{N-1})}, \tag{3.12}$$

где M – коэффициент ускорения, $M > 1$.

Содержание отчета

Отчет о выполнении лабораторной работы должен содержать следующее.

1 Титульный лист.

2 Цель работы.

3 Описание хода выполнения работы по пп. 3.1.1–3.1.4.

4 Вывод.

Отчет оформляется на персональном компьютере в текстовом редакторе.

Контрольные вопросы

1 Дайте общую характеристику численных методов решения нелинейных уравнений.

2 Поясните алгоритм метода Ньютона.

3 Поясните алгоритм метода дихотомии.
4 Поясните алгоритм метода секущих.

4 Лабораторная работа № 4. Интерполяция функциональных зависимостей

Цель работы: изучить основные методы интерполяции табличных функций.

4.1 Порядок выполнения работы

4.1.1 Получить индивидуальное задание у преподавателя, содержащее функцию в табличной форме и набор значений аргумента, для которых необходимо найти значение функции.
4.1.2 Составить программу в среде C++ Builder для решения индивидуального задания.
4.1.3 Получить значения функции для заданных значений аргумента методом интерполяции.
4.1.4 Оформить отчет о лабораторной работе.

4.2 Основные теоретические положения

4.2.1 Интерполяция степенными многочленами.

В данной группе методов интерполяции используется в качестве выражения интерполяционной кривой, заменяющей табличную функцию, степенной ряд общего вида

$$g(x) = a_0 + a_1x + a_2x^2 + a_3x^3 + \ldots + a_mx^m, \tag{4.1}$$

где a_0, a_1, ..., a_m – коэффициенты ряда;
m – степень многочлена.

4.2.2 Линейная интерполяция.

Простейшим видом интерполяции является линейная интерполяция (когда $m = 1$), использующая аппроксимацию исходной кривой $F = Y(X)$ на участке между двумя соседними узлами (X_i, Y_i) и (X_{i+1}, Y_{i+1}) прямой линией, проходящей через эти узлы. При этом уравнение интерполяционной прямой линии между двумя соседними точками имеет вид:

$$g(x) = a_0 + a_1x, \tag{4.2}$$

где a_0, a_1 – коэффициенты интерполяционного многочлена.

При использовании метода линейной интерполяции производится определение на оси абсцисс интервала межу узлами, где лежит искомая точка $[X_a, X_b]$.

Далее значение функции определяется по формуле

$$Y = Y_a + \frac{Y_b - Y_a}{X_b - X_a}(X - X_a). \tag{4.3}$$

Схема алгоритма линейной интерполяции имеет вид рисунка 4.1.

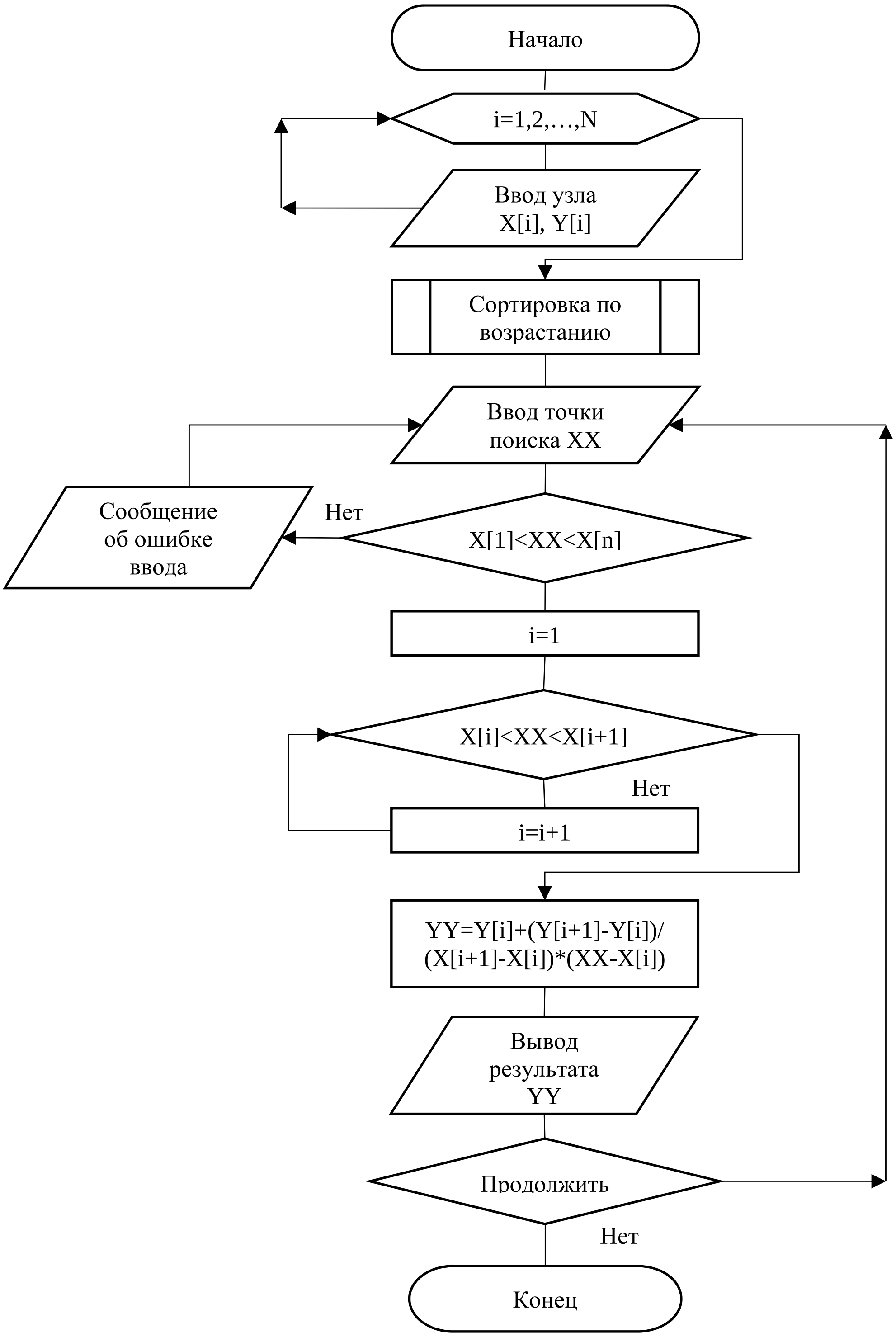

Рисунок 4.1 – Алгоритм линейной интерполяции

4.2.3 Квадратичная интерполяция.

Является более точной по сравнению с линейной интерполяцией. Квадратичная интерполяция использует многочлен второй степени

$$g(x) = a + bx + cx^2. \tag{4.4}$$

При этом аппроксимирующей кривой является квадратичная парабола, которая строится по трем смежным точкам: (x_{i-1}, y_{i-1}), (x_i, y_i) и (x_{i+1}, y_{i+1}).

Коэффициенты многочлена определяются путем решения системы из трех уравнений

$$\begin{aligned} &a_i + b_i x_{i-1} + c_i x_{i-1}^2 = y_{i-1}; \\ &a_i + b_i x_i + c_i x_i^2 = y_i; \\ &a_i + b_i x_{i+1} + c_i x_{i+1}^2 = y_{i+1}, \end{aligned} \tag{4.5}$$

решением которой являются коэффициенты, определяемые по формулам

$$c_i = \frac{1}{x_{i+1} - x_{i-1}} \left(\frac{y_{i+1} - y_i}{x_{i+1} - x_i} - \frac{y_i - y_{i-1}}{x_i - x_{i-1}} \right); \tag{4.6}$$

$$b_i = (y_i - y_{i-1}) / (x_i - x_{i-1}) - c_i (x_i - x_{i-1}); \tag{4.7}$$

$$a_i = y_i - b_i x_i - c_i x_i^2. \tag{4.8}$$

4.2.4 Интерполяция полиномом Лагранжа.

Частным случаем линейной интерполяции является метод Лагранжа (рисунок 4.2). Этот метод позволяет получить выражение интерполяционной функции $G(X)$ в виде полинома Лагранжа, имеющего следующий вид:

$$Y(X_i) = P_n(X_i) = Y_0 B_0(X) + Y_1 B_1(X) + \,..+ Y_n B_n(X), \tag{4.9}$$

где $B_i(X)$ – многочлены степени n, коэффициенты которого находятся при помощи системы $n + 1$ уравнений:

$$Y_i = P_n(X_i). \tag{4.10}$$

Система уравнений для определения $B_i(X)$ имеет вид:

$$\begin{aligned} &Y_0 B_0(X_0) + Y_1 B_1(X_0) + \ldots + Y_n B_n(X_0) = Y_0; \\ &Y_0 B_0(X_1) + Y_1 B_1(X_1) + \ldots + Y_n B_n(X_1) = Y_1; \\ &\ldots \\ &Y_0 B_0(X_n) + Y_1 B_1(X_n) + \ldots + Y_n B_n(X_n) = Y_n. \end{aligned} \tag{4.11}$$

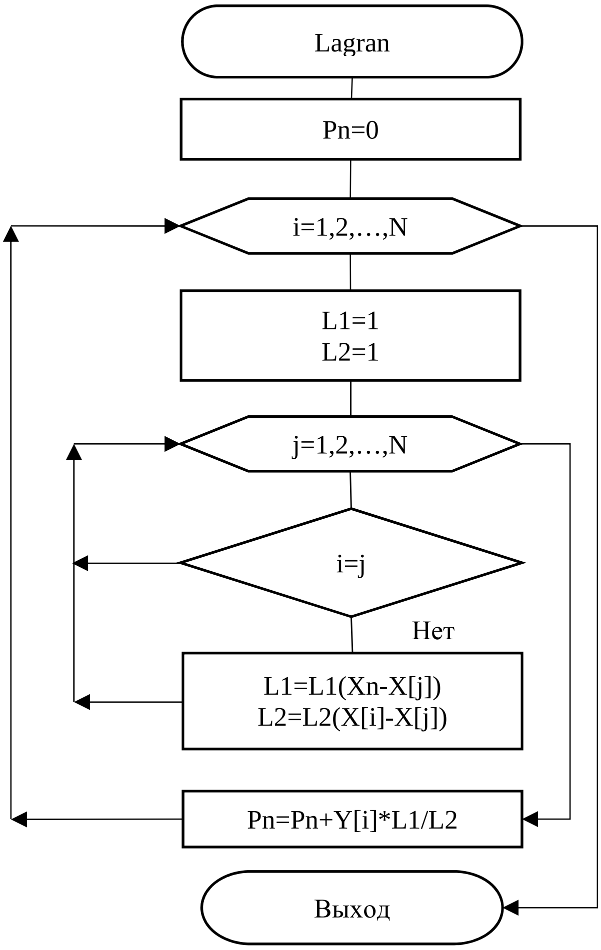

Рисунок 4.2 – Алгоритм интерполяции полиномом Лагранжа

Из условия совпадения значений исходной функции и полинома в узловых точках должно соблюдаться следующее условие:

$$B_j\left(X_i\right)=1 \text{ при } i = j,$$

$$B_j(X_i)=0 \;\; B_j(X_i) = 0 \text{ при } i \neq j. \qquad (4.12)$$

Откуда выражение многочлена записывается как

$$B_j\left(X_i\right)=C_j\left(X-X_0\right)\left(X-X_1\right)\cdot\ldots\cdot\left(X-X_{j-1}\right)\left(X-X_{j+1}\right)\cdot\ldots\cdot\left(X-X_n\right). \qquad (4.13)$$

Вследствие $B_j\left(X_i\right)=1$ можно получить

$$C_j = 1/\left[\left(X_j-X_0\right)\left(X_j-X_1\right)\cdot\ldots\cdot\left(X_j-X_{j-1}\right)\left(X_j-X_{j+1}\right)\cdot\ldots\cdot\left(X_j-X_n\right)\right]. \; (4.14)$$

Общий вид полинома Лагранжа табличной функции

$$P_n(x)=\sum_{i=1}^{N}\left[y_i\frac{\prod\limits_{\substack{j=1\\ i=j}}^{N}(x-x_j)}{\prod\limits_{\substack{j=1\\ i=j}}^{N}(x_i-x_j)}\right]=\sum_{i=0}^{N}Y_i\frac{(X-X_1)\cdot\ldots\cdot(X-X_n)}{(X_i-X_1)\cdot\ldots\cdot(X_i-X_n)}. \qquad (4.15)$$

Содержание отчета

Отчет о выполнении лабораторной работы должен содержать следующее.
1 Титульный лист.
2 Цель работы.
3 Описание хода выполнения работы по пп. 4.1.1–4.1.4.
4 Вывод.

Отчет оформляется на персональном компьютере в текстовом редакторе.

Контрольные вопросы

1 Поясните алгоритм интерполяции степенными многочленами.
2 Поясните алгоритм линейной интерполяции.
3 Поясните алгоритм интерполяции полиномом Лагранжа.
4 Поясните алгоритм квадратичной интерполяции.

5 Лабораторная работа № 5. Аппроксимация функциональных зависимостей

Цель работы: изучить основные методы аппроксимации табличных функций.

5.1 Порядок выполнения работы

5.1.1 Получить индивидуальное задание у преподавателя, содержащее функцию в табличной форме и метод аппроксимации, который необходимо применить.
5.1.2 Составить программу в среде C++ Builder для решения индивидуального задания.
5.1.3 Получить параметры аппроксимирующей функции.
5.1.4 Оформить отчет о лабораторной работе.

5.2 Основные теоретические положения

5.2.1 Линейная аппроксимация.

Линейная аппроксимация для табличной функции имеет вид прямой (5.1). Уравнения коэффициентов A и B получаются путем решения линейной системы двух уравнений вида

$$\begin{aligned} Aa_{11} + Ba_{12} &= b_1; \\ Aa_{21} + Ba_{22} &= b_2, \end{aligned} \tag{5.1}$$

где коэффициенты уравнения определяются по формулам

$$a_{11} = \sum_{i=1}^{K} x_i^2;$$

$$a_{12} = a_{21} = \sum_{i=1}^{K} x_i;$$

$$a_{22} = K;$$

$$b_1 = \sum_{i=1}^{K} x_i y_i;$$

$$b_2 = \sum_{i=1}^{K} y_i. \tag{5.2}$$

Для решения системы можно использовать численный или прямой метод решения системы линейных уравнений. Можно применять прямой метод на основе правила Крамера для системы линейных уравнений. Тогда коэффициенты вычисляются по формулам

$$A = \frac{b_1 a_{22} - b_2 a_{12}}{a_{11} a_{22} - a_{12} a_{21}}; \tag{5.3}$$

$$B = \frac{b_2 a_{11} - b_1 a_{21}}{a_{11} a_{22} - a_{12} a_{21}}. \tag{5.4}$$

На рисунке 5.1 показана схема алгоритма линейной аппроксимации.

5.2.2 Степенная аппроксимация.

Степенная аппроксимация табличной функции производится с помощью базовой функции

$$g(x) = Ax^M, \tag{5.5}$$

где M – известное постоянное значение степени.

Минимум ошибки

$$E(A) = \sum_{i=1}^{K} (Ax_i^M - y_i)^2 \tag{5.6}$$

достигается при решении уравнения

$$E'(A) = 0;$$

$$E'(A) = 2\sum_{i=1}^{K}(Ax_i^M - y_i)(x_i^M). \qquad (5.7)$$

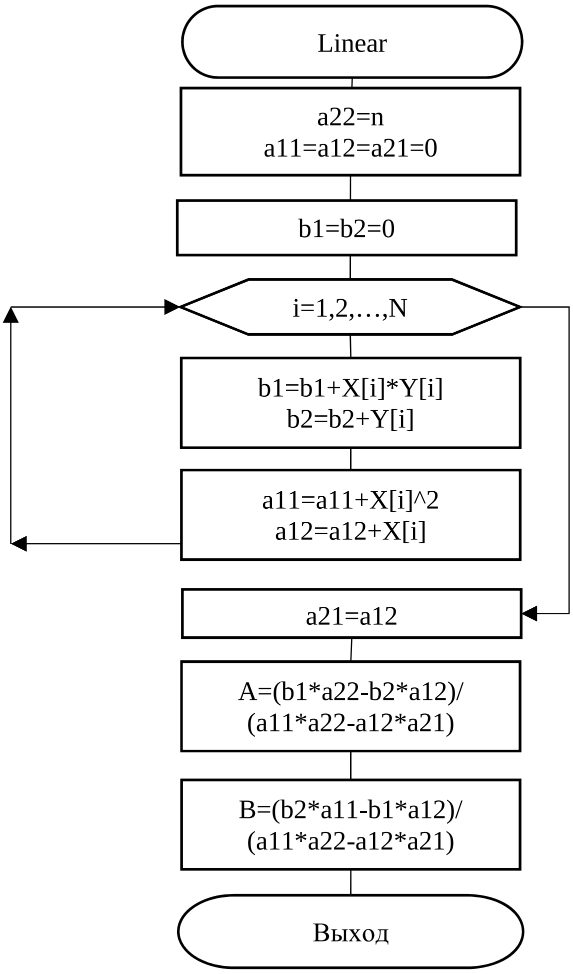

Рисунок 5.1 – Алгоритм линейной аппроксимации

Значение A для заданного M, когда суммарная ошибка минимальна,

$$A = \frac{\sum_{i=1}^{K} x_i^M y_i}{\sum_{i=1}^{K} x_i^{2M}}. \qquad (5.8)$$

Содержание отчета

Отчет о выполнении лабораторной работы должен содержать следующее.
1 Титульный лист.
2 Цель работы.
3 Описание хода выполнения работы по пп. 5.1.1–5.1.4.
4 Вывод.

Контрольные вопросы

1 Поясните назначение методов аппроксимации.
2 Поясните алгоритм линейной аппроксимации.
3 Поясните алгоритм степенной аппроксимации.

6 Лабораторная работа № 6. Оптимизация технических систем

Цель работы: изучить основные методы оптимизации технических систем.

6.1 Порядок выполнения работы

6.1.1 Получить у преподавателя индивидуальное задание – функцию для поиска оптимального значения аргумента.

6.1.2 Разработать программу в среде C++ Builder для решения индивидуального задания.

6.1.3 Методом компьютерного моделирования найти оптимальное значение аргумента заданной функции.

6.1.4 Оформить отчет о лабораторной работе.

6.2 Основные теоретические положения

6.2.1 Оптимизация однопараметрических функций.

В технике довольно часто используются методы одномерного поиска оптимального решения. При этом необходимо определить локальный экстремум однопараметрической (унимодальной) функции на заданном интервале при отсутствии ограничений на область допустимых значений переменной аргумента (независимой переменной).

Поиск локального минимума (максимума) однопараметрической (унимодальной) функции вида

$$y = f(x) \tag{6.1}$$

на заданном интервале $[a,\ b]$ может производиться с помощью следующих методов:

1) метода равномерного поиска;
2) метода деления отрезка пополам (дихотомии);
3) метода золотого сечения;
4) метода чисел Фибоначчи;
5) методов квадратичной оптимизации.

6.2.2 Метод равномерного поиска.

Данный метод относится к пассивным методам (без обучения в процессе поиска). Суть его заключается в постепенном расчете двух массивов значений аргумента X и функции Y от начальной точки a до последней точки b интервала поиска с заданной величиной шага Δx с одновременным поиском минимального (максимального) значения элементов массива функции. Недостаток метода – большой объем вычислений при сравнительно низкой точности. Он может использоваться для первоначального определения локального экстремума.

6.2.3 Метод деления интервала поиска пополам.

Данный метод относится к группе методов с обучением, в которых реализовано постепенное сокращение интервала поиска на каждой итерации (рисунок 6.1) . На каждой итерации происходит сужение интервала поиска вдвое до тех пор, пока размер интервала [ab] не станет меньше, чем заданная точность dX. В теле цикла проверки определяется середина интервала по формуле

$$X_{ср} = (a+b)/2. \qquad (6.2)$$

Затем проверяется значение функции в точках X_1 и X_2, которые определяются на основе введенного значения окрестности е по формулам

$$X_1 = X_{ср} - e;$$
$$X_2 = X_{ср} + e. \qquad (6.3)$$

Далее сравниваются значения функции в точках X_1 и X_2 между собой и делается вывод о том, какую половину исключить из рассмотрения. При поиске минимума если $F(X_1) < F(X_2)$ (для максимума – $F(X_1) > F(X_2)$), тогда $b = X_{ср}$, в противном случае $a = X_{ср}$.

6.2.4 Метод золотого сечения.

Поиск минимума методом золотого сечения на заданном интервале [a, b] является итерационным процессом, когда последовательно производится сужение интервала неопределенности. Для этого интервал дробится на части с помощью промежуточных точек X_1 и X_2, которые выбираются исходя из соотношения золотого сечения по формулам

$$X_1 = a + (b-a)\cdot 0,382; \qquad (6.4)$$

$$X_2 = a + (b-a)\cdot 0,618. \qquad (6.5)$$

Далее сравниваются значения функции в этих точках:

$$Y_1 = f(X_1); \qquad (6.6)$$

$$Y_2 = f(X_2). \qquad (6.7)$$

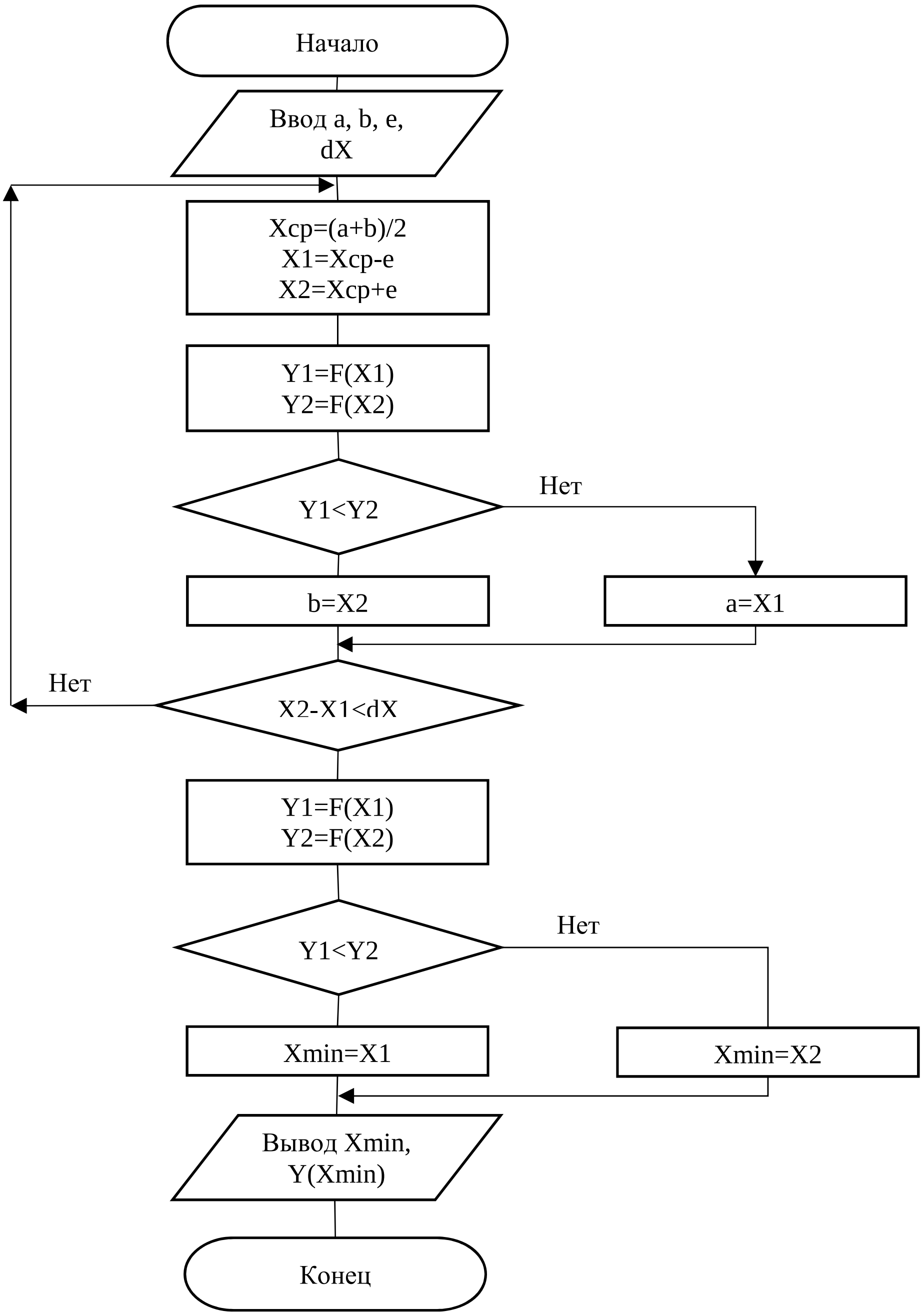

Рисунок 6.1 – Алгоритм минимизации методом деления пополам

Если при поиске минимума $Y_1 < Y_2$, тогда правая граница переносится из точки b в X_2:

$$b = X_2, \tag{6.8}$$

в противном случае переносится левая граница из точки a в точку X_1:

$$a = X_1. \tag{6.9}$$

Далее необходимо повторить итерационный процесс до тех пор, пока ширина интервала поиска не станет меньше заданной точности:

$$\left| b - a \right| < \varepsilon \,. \tag{6.10}$$

После завершения итерационного процесса сравниваются значения функции на границах интервала:

$$Y_a = f\left(a\right); \tag{6.11}$$

$$Y_b = f\left(b\right). \tag{6.12}$$

Если выполняется условие

$$Y_a < Y_b, \tag{6.13}$$

тогда минимум находится в точке *a*:

$$X_{\min} = a. \tag{6.14}$$

В противном случае минимум соответствует точке *b*:

$$X_{\min} = b. \tag{6.15}$$

Содержание отчета

Отчет о выполнении лабораторной работы должен содержать следующее.

1 Титульный лист.

2 Цель работы.

3 Описание хода выполнения работы по пп. 6.1.1–6.1.4.

4 Вывод.

Отчет оформляется на персональном компьютере в текстовом редакторе.

Контрольные вопросы

1 Поясните алгоритм метода равномерного поиска.

2 Поясните алгоритм метода деления интервала поиска пополам.

3 Поясните алгоритм метода золотого сечения.

4 Перечислите основные методы оптимизации однопараметрических функций.

7 Пример выполнения лабораторной работы

Цель работы: изучить основные методы численного интегрирования дифференциальных уравнений.

7.1 Индивидуальное задание

Разработать программу численного интегрирования дифференциального уравнения $y'(x) = y(x) + 2x$ методом Эйлера первого порядка. Произвести интегрирование на интервале $0 < x < 1$.

7.2 Разработка программы в среде C++ Builder

7.2.1 Описание переменных численного интегрирования.

В cpp-файле главной формы описаны следующие глобальные переменные.

```
double X0,  // Начало интервала поиска решения
Xk, // Конец интервала поиска решения
Y0, // Начальное условие
h, // шаг интегрирования
e; // точность интегрирования
int N; // Число сохраняемых расчетных точек
double X[10000], // массив результатов для независимой переменной
Y[10000]; // массив результатов для зависимой переменной
```

7.2.2 Формирование функции дифференциального уравнения.

Дифференциальное уравнение оформлено в виде функции, возвращающей вещественное значение и имеющей два вещественных параметра с передачей данных по значению. Заголовок-прототип функции описан в h-файле главной формы (после описания класса формы), сама функция – в cpp-файле (после описания глобальных переменных).

```
double f(double x, double y)
{
    return 10*(1-y);
} // конец функции f
```

7.2.3 Программирование ввода исходных данных.

Программирование ввода исходных данных выполнено в начале тела функции обработчика события кнопки численного интегрирования. Ввод исходных данных производится путем считывания текстовых значений поля Text соответствующих компонентов с последующим преобразованием в нужный тип с помощью функций StrToFloat или StrToInt. Предварительно в полях text компонентов Edit были заданы значения по умолчанию.

```
X0=StrToFloat(EditX0->Text);
Y0=StrToFloat(EditY0->Text);
Xk=StrToFloat(EditXk->Text);
h=StrToFloat(EditH->Text);
E=StrToFloat(EditE->Text);
N=StrToInt(EditN->Text);
```

7.2.4 Реализация алгоритма численного интегрирования.

Внутри обработчика события нажатия кнопки расчета объявлены локальные переменные.

```
long int Nj; // число точек интегрирования
double Xn=X0, Yn=Y0; // предшествующая точка интегрирования
double Xn1, Yn1; // расчетная точка интегрирования
```

Далее производятся предварительный расчет, сохранение и начальная инициализация параметров интегрирования:

```
Nj=(Xk-X0)/h/N; // расчет числа точек интегрирования на шаг сохранения
if (Nj<1) // Коррекция числа точек
{
   Nj=1;
   h=(Xk-X0)/Nj/N;
}
X[0]=X0;
Y[0]=Y0; // Сохранение начальных значений
```

Выполняется установка параметров компонента ProgressBar:

```
ProgressBar1->Max=N; // Установка максимума полосы расчета
ProgressBar1->Position=0; // Установка текущего значения полосы расчета
```

Алгоритм численного интегрирования дифференциального уравнения реализован следующим образом:

```
for (int i=1; i<N; i++) // цикл сохранения
{
   for(long int j=1; j<=Nj; j++)// цикл интегрирования
  {
      Xn1=Xn+h; // расчет независимой переменной
      Yn1=Yn+h*f(Xn,Yn); // расчет зависимой переменной
      Xn=Xn1; Yn=Yn1; // переопределение текущей точки расчета
   } // конец цикла for j
   X[i]=Xn; Y[i]=Yn; // сохранение текущей точки интегрирования
   ProgressBar1->Position=i; // отображение процесса расчета
}
```

7.3 Результат численного интегрирования заданного дифференциального уравнения

Результат численного интегрирования приведен в таблице 7.1.

Таблица 7.1 – Результат численного интегрирования

X	*Y*
0	0
0,1	0,0103418
0,2	0,0428055
0,3	0,0997176
0,4	0,1836494
0,5	0,2974425
0,6	0,4442376
0,7	0,6275054
0,8	0,8510819
0,9	1,1192063
1	1,4365637

Вывод: в ходе выполнения лабораторной работы были изучены основные методы численного интегрирования дифференциальных уравнений, разработана программа интегрирования дифференциального уравнения методом Эйлера первого порядка на языке C++.

Список литературы

1 Компьютерное моделирование : учебник / В. М. Градов [и др.]. – Москва : КУРС ; ИНФРА-М, 2018. – 264 с.

2 **Королёв, А. Л.** Компьютерное моделирование. Лабораторный практикум / А. Л. Королёв. – 2-е изд. – Москва : БИНОМ ; Лаборатория знаний, 2013. – 296 с. : ил.

3 **Чернышев, А. Ю.** Электропривод переменного тока : учебное пособие / А. Ю. Чернышев, Ю. Н. Дементьев, И. А. Чернышев. – 2-е изд. – Томск : Томск. политехн. ун-т, 2015. – 210 с.

4 **Булавин, Л. А.** Компьютерное моделирование физических систем : учебное пособие / Л. А. Булавин, Н. В. Выгорницкий, Н. И. Лебовка. – Долгопрудный : Интеллект, 2011. – 352 с.