Министерство образования Республики Беларусь
Учреждение образования
«МОГИЛЕВСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
ПРОДОВОЛЬСТВИЯ»

Кафедра автоматизации технологических процессов и производств

КОМПЬЮТЕРНОЕ МОДЕЛИРОВАНИЕ ТЕХНОЛОГИЧЕСКИХ ПРОЦЕССОВ И ОБОРУДОВАНИЯ ПИЩЕВОЙ ПРОМЫШЛЕННОСТИ

Лабораторный практикум

для студентов специальности
1 – 40 05 01 Информационные системы и технологии (по направлениям)

Направления специальности:

1 – 40 05 01-11 Информационные системы и технологии (в пищевой промышленности)

Часть 1

Могилев 2017

УДК 65.011.66

На заседании кафедры автоматизации технологических процессов и производств

Протокол №10 от 04.03.2011

Одобрена и рекомендована к утверждению УМС по специальности

1– 53 01 01 Автоматизация технологических процессов и производств (по направлениям)

протокол № _ от ___________ г.

Председатель УМСС

к.т.н., доцент М.М. Кожевников

Составитель:
Е.А. Колюкович

Рецензент
кандидат технических наук, доцент МГУП
Е.Л. Волынская

Лабораторный практикум предназначен для использования студентами для студентов специальности 1 – 40 05 01 Информационные системы и технологии (по направлениям), направление специальности 1 – 40 05 01-11 Информационные системы и технологии (в пищевой промышленности) дневной формы обучения при выполнении лабораторных работ по дисциплине «Компьютерное моделирование технологических процессов и оборудования пищевой промышленности».

Приведены теоретические сведения, методические указания к самостоятельной подготовке и проведению работ, а также вопросы для самопроверки.

Содержание

Общие положения ... 4
Основные сведения по программному продукту Matlab. ... 4
Лабораторная работа №1. Решение систем линейных алгебраических уравнений ... 5
Лабораторная работа №2. Решение систем нелинейных уравнений ... 9
Лабораторная работа №3. Анализ экспериментальных данных ... 13
Лабораторная работа №4. Численное решение интегральных уравнений ... 16
Лабораторная работа №5. Численное решение дифференциальных уравнений (задачи Коши) ... 20
Лабораторная работа №6. Численное решение системы дифференциальных уравнений(задачи Коши) ... 24
Лабораторная работа №7. Составление статической математической модели смесителя потоков ... 28
Лабораторная работа №8. Составление статической математической модели теплообменника ... 31
Список рекомендуемой литературы ... 34
Приложение А – Исходные даные ... 35
Приложение Б – Исходные даные ... 37
Приложение В – Исходные даные ... 39
Приложение Г – Исходные даные ... 41
Приложение Д – Исходные даные ... 43
Приложение Е – Исходные даные ... 44
Приложение Ж – Исходные даные ... 48

Общие положения

Лабораторный практикум по курсу «Компьютерное моделирование технологических процессов и оборудования пищевой промышленности» проводится в соответствии с предусмотренным планом количеством часов и графиком, составленным для каждой учебной группы. Перечень лабораторных работ для каждой специальности утвержден в учебных программах.

На первом занятии студенты проходят инструктаж по технике безопасности при выполнении лабораторных работ, о чем делается запись в соответствующем журнале.

К началу лабораторного занятия студент обязан ознакомиться с содержанием и методикой выполнения предстоящей работы, как по настоящим методическим указаниям, так и по рекомендуемым литературным источникам. Особое внимание следует обратить на синтаксис и ключевые слова программного продукта Matlab, разработку алгоритма решений поставленных задач, правильность написания функций и подпрограмм. Студент, не подготовленный к занятию, к работе не допускается.

Основные сведения по программному продукту Matlab.

Среди бурно развивающихся систем компьютерной математики (СКМ), в первую очередь ориентированных на численные расчеты, особо выделяется матричная математическая система Matlab.

Эффективность Matlab обусловлена, прежде всего, ее ориентацией на матричные вычисления с программной эмуляцией параллельных вычислений и упрощенными средствами задания циклов. Последние версии системы поддерживают 64-разрядные микропроцессоры и многоядерные микропроцессоры, что обеспечивает высочайшие показатели по скорости вычислений и скорости математического имитационного моделирования.

В Matlab удачно реализованы средства работы с многомерными массивами, большими и разреженными матрицами и многими типами данных. Система прошла многолетний путь развития от узко специализированного матричного программного модуля, используемого только на больших ЭВМ, до универсальной интегрированной СКМ, ориентированной на массовые персональные компьютеры. Matlab имеет мощные средства диалога, графики и комплексной визуализации вычислений.

Система Matlab предлагается разработчиками (корпорация The MathWorks, Inc.) как лидирующий на рынке, в первую очередь на предприятиях военно-промышленного комплекса, в энергетике, в аэрокосмической отрасли и в автомобилестроении язык программирования высокого уровня для технических вычислений, расширяемый большим числом пакетов прикладных программ – расширений. Самым известным из них стало расширение Simulink, обеспечивающее блочное имитационное моделирование различных систем и устройств.

Лабораторная работа №1. Решение систем линейных алгебраических уравнений

Цель работы: изучение методов решения систем линейных алгебраических уравнений (СЛАУ) с помощью программного продукта Matlab.

1.1 Теоретические сведения

Будем рассматривать только **определенные (нормальные) системы**, в которых количество неизвестных соответствует количеству уравнений.

СЛАУ называется система, которая содержит алгебраические функции с неизвестными в первой степени во всех уравнениях системы или СЛАУ это система, состоящая только из линейных алгебраических уравнений.

Пусть дана система n линейных алгебраических уравнений с n неизвестными

$$\begin{cases} a_{11}\cdot x_1+a_{12}\cdot x_2+\ldots+a_{1n}\cdot x_n=b_1 \\ a_{21}\cdot x_1+a_{22}\cdot x_2+\ldots+a_{2n}\cdot x_n=b_2 \\ \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \\ a_{n1}\cdot x_1+a_{n2}\cdot x_2+\ldots+a_{nn}\cdot x_n=b_n. \end{cases} \qquad (1.1)$$

Совокупность чисел $\xi 1$, $\xi 2$, …,ξn, обращающих систему (1.1) в тождество, называется решением этой системы.

Способы решения систем линейных алгебраических уравнений можно разделить на две группы:

1) точные методы, представляющие собой конечные алгоритмы для вычисления корней системы. Таковы, например, правило Крамера, метод Гаусса, метод главных элементов, метод квадратных корней и другие;

2) итерационные методы, позволяющие получить корни системы с заданной точностью путем сходящихся бесконечных процессов. К числу их относятся метод итерации, метод Зейделя, метод релаксации и другие.

Вследствие неизбежных округлений результаты даже точных методов являются приближенными, причем оценка погрешностей корней в общем случае затруднительна. При использовании итерационных методов, сверх того, добавляется погрешность метода.

Рассмотрим в качестве примера СЛАУ вида:

$$\begin{cases} 1\cdot x_1+2\cdot x_2-3\cdot x_3=10 \\ 4\cdot x_1+5\cdot x_2+6\cdot x_3=20 \\ 7\cdot x_1-8\cdot x_2-9\cdot x_3=30. \end{cases} \qquad (1.2)$$

1.2 Решение в Matlab

В Matlab решение СЛАУ можно найти различными способами, используя разнообразные встроенные функции, дающие улучшенные результаты при определенных условиях. В каждой функции возможен ввод дополнительных параметров, таких как: точность вычислений, количество итераций и др. Рассмотрим некоторые функции вычисления СЛАУ с базовыми параметрами.

В Matlab вычисления СЛАУ производятся следующими способами

1) Метод матричных операций

Для того что бы ввести матрицу А необходимо в рабочем пространстве в командной строке ввести ее имя «А», знак присваивания «=» и в квадратных скобках ввести элементы матрицы, отделяя каждый элемент матрицы «пробелом», а каждую строку «;». После ввода матрицы необходимо нажать «Ввод». Matlab автоматически осуществит вывод матрицы А.

```
>> A=[1 2 -3; 4 5 6; 7 -8 -9]
```

Рисунок 1.1 – Ввод матрицы А

```
A =

   1   2  -3
   4   5   6
   7  -8  -9
```

Рисунок 1.2 – Вывод матрицы А

Далее вводится вектор В и расчетная формула.

```
>> A=[1 2 -3; 4 5 6; 7 -8 -9]

A =

   1   2  -3
   4   5   6
   7  -8  -9

>> B=[10; 20; 30]

B =

  10
  20
  30
```

```
>> X=(A^-1)*B

X =

   4.6667
   1.3333
  -0.8889
```

Рисунок 1.3 – Метод матричных операций в Matlab

Вместо «А^-1» можно использовать функцию «inv», которая инвертирует матрицу А.

2) Итерационный метод наименьших квадратов.

Для решения используется встроенная функция lsqr.

Функция lsqr(A, B) – возвращает точное решение X СЛАУ A·X=B, если оно существует, в противном случае возвращает решение по итерационному методу наименьших квадратов. Матрица коэффициентов A должна быть прямоугольной размера m×n, а вектор-столбец правых частей уравнений B должен иметь размер m. Условие m>=n может быть и необязательным. Функция lsqr начинает итерации от начальной оценки, по умолчанию представляющей собой вектор длиной n, состоящий из нулей. Итерации производятся или до сходимости к решению, или до появления ошибки, или до достижения максимального числа итераций (по умолчанию равного min(20,m, n) – либо 20, либо числу уравнений, либо числу неизвестных).

```
>> A=[1 2 -3; 4 5 6; 7 -8 -9]

A =

   1   2  -3
   4   5   6
   7  -8  -9

>> B=[10; 20; 30]

B =

  10
  20
  30
```

```
>> lsqr(A,B)
lsqr converged at iteration 3 to a solution with relative residual 1.6e-015.

ans =

   4.6667
   1.3333
  -0.8889
```

Рисунок 1.4 – Решение СЛАУ итерационным методом с помощью функции lsqr

3) Итерационный двунаправленный метод сопряженных градиентов.

Решение СЛАУ как с обычной, так и с разреженной матрицей возможно также известным двунаправленным методом сопряженных градиентов. Он реализован функцией bicg.

Функция bicg(A, B) – возвращает решение X СЛАУ A·X=B. Матрица коэффициентов A должна быть квадратной размера n×n, а вектор-столбец правых частей уравнений B должен иметь длину n. Функция bicg начинает итерации от начальной оценки, по умолчанию представляющей собой вектор длиной n, состоящий из нулей. Итерации производятся или до сходимости к решению, или до появления ошибки, или до достижения максимального числа итераций (по умолчанию равно min(20,n) – либо 20, либо числу уравнений).

Благодаря использованию двунаправленного метода сопряженных градиентов bicg сходится за меньшее число итераций, чем lsqr, но требует квадратную матрицу A, отбрасывая информацию, содержащуюся в дополнительных уравнениях, в то время как lsqr работает и с прямоугольной матрицей.

```
>> A=[1 2 -3; 4 5 6; 7 -8 -9]

A =

    1    2   -3
    4    5    6
    7   -8   -9

>> B=[10; 20; 30]

B =

   10
   20
   30
```

```
>> bicg(A,B)
bicg converged at iteration 3 to a solution with relative residual 1.4e-014.

ans =

    4.6667
    1.3333
   -0.8889
```

Рисунок 1.5 – Решение СЛАУ итерационным методом с помощью функции bicg

Как видим результаты вычислений СЛАУ в Matcad и Matlab различными методами дают идентичные результаты.

1.3 Задание на выполнение работы

1) Внимательно ознакомиться с теоретическими сведениями лабораторной работы.

2) Выбрать свой вариант исходных данных. Исходные данные представлены в таблице А.1 приложения А. Номер варианта – это порядковый номер в журнале группы.

3) Подготовить алгоритм выполнения лабораторной работы.

4) Выполнить расчеты с использованием программного продукта Matlab.

5) Подготовить дополнительные информационные материалы таблицы, графики и др., если это необходимо.

6) Результаты оформляются в виде отчета по выполнению лабораторной работы. Краткий отчет содержит: цель работы, исходные данные, скриншоты или листинг программы расчетов, дополнительные информационные материалы, краткий анализ результатов работы.

1.4 Контрольные вопросы

1) Что такое линейное уравнение?

2) Что такое СЛАУ?

3) Какие методы используются для нахождения решения СЛАУ?

4) Какие функции используются для решения СЛАУ в Matlab?

Лабораторная работа №2. Решение систем нелинейных уравнений

Цель работы: изучение методов решения систем нелинейных уравнений (СНУ) с помощью программного продукту Matlab.

2.1 Теоретические сведения

Будем рассматривать только **определенные (нормальные) системы**, в которых количество неизвестных соответствует количеству уравнений.

СНУ называется система, которая содержит трансцендентные (показательные, тригонометрические, логарифмические и др.) функции с неизвестными или алгебраические (целые, рациональные, иррациональные) функции, с неизвестными в степени отличной от единицы, хотя бы в одном уравнении системы или СНУ это система, в которой хотя бы одно уравнение является нелинейным.

Пусть дана система n нелинейных уравнений с n неизвестными:

$$f_i(x_1,x_2,\ldots,x_n),\ i=0,1,\ldots.n, \tag{2.1}$$

где f_i – некоторые действительные функции неизвестных $x_1,x_2,\ldots,x_n$, причем одна или несколько или все являются нелинейными относительно одного или нескольких или всех неизвестных $x_1,x_2,\ldots,x_n$.

Совокупность чисел ξ1, ξ2, …,ξn, обращающих систему (2.1) в тождество, называется решением этой системы.

В общем случае решение СНУ не может быть получено в аналитическом виде; решение, как правило, ищется численными методами, для которых анализ сходимости оказывается значительно сложнее по сравнению с линейными системами. Кроме того, следует отметить, что сходимость методов существенно зависит от задания начального приближения. Поэтому при выборе начального приближения необходимо использовать максимум имеющейся информации об искомом решении.

Рассмотрим в качестве примера СНУ вида:

$$\begin{cases} \cos(x_1) - x_2 = 0 \\ x_1 + \sin(x_2) = 0 \end{cases} \tag{2.2}$$

2.2 Решение в Matlab

Для лучшего представления решения СНУ необходимо построить совмещенный график двух уравнений. График функции строиться командой plot(X,Y).

Синтаксис функции:

plot(X,Y), 2.3

где X – вектор координат x;

Y – вектор координат y.

Для того что бы построить на одном графике две функции необходимо использовать команду hold on. Команда hold on включает режим сохранения текущего графика и свойств объекта axes, так что последующие команды приведут к добавлению новых графиков в графическом окне. Команда hold off выключает режим сохранения графика.

```
X1=[-5:0.1:5];
X2=cos(X1);
plot(X1,X2)
hold on
X2=[-5:0.1:5];
X1=-sin(X2);
plot(X1,X2)
```

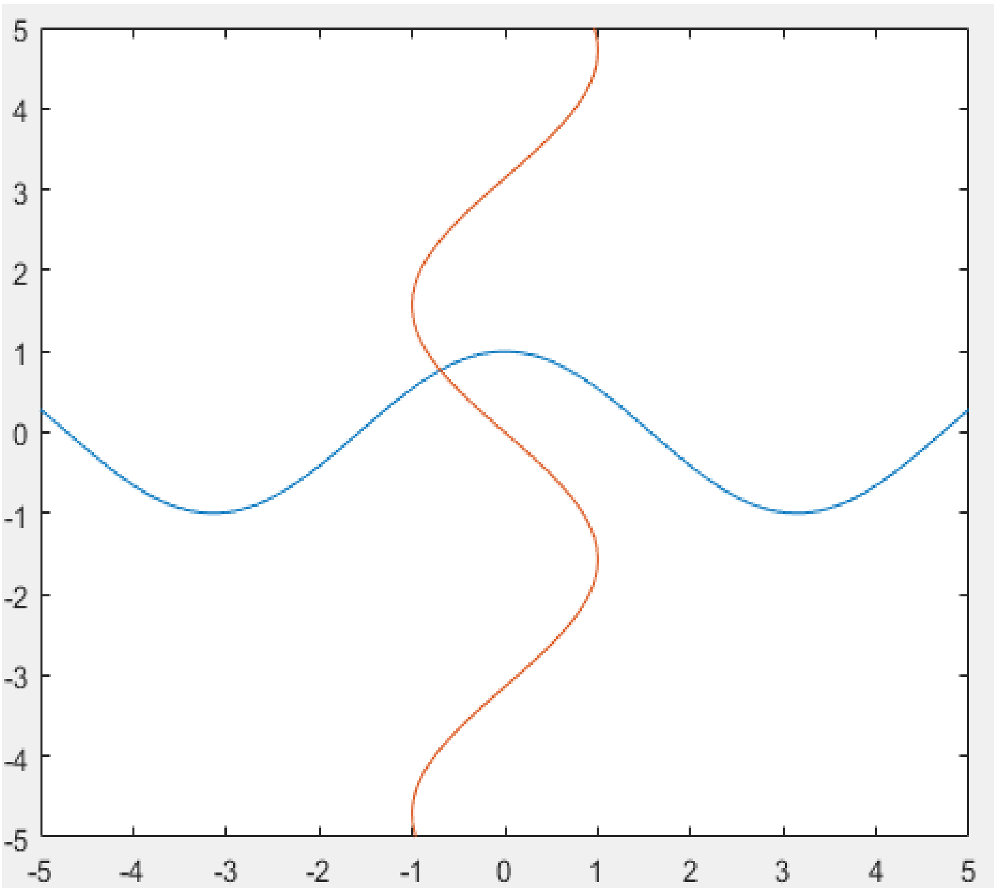

Рисунок 2.1 – Скрипт построения графиков и графики СНУ

Для того, чтобы решить СНУ в Matlab в первую очередь необходимо создать m-файл, содержащий уравнения системы и сохранить его.

Рассмотрим в качестве примера систему 2.2.

Создадим новый m-файл (расположение: File/New/Function). Введем систему нелинейных уравнений F, записанных как функции переменных x1 и x2.

Так как Matlab оперирует матрицами, то необходимо соблюдать определенный синтаксис. Переменные заносятся в вектор-столбец x, первая переменная находится в первом элементе вектор-столбца, а вторая во втором элементе. Обращение к элементам осуществляется так «имя вектор-столбца (номер элемента вектор-столбца)». Поэтому переменную x1 надо записать как x(1), а переменную x2, как x(2).

Сохраним под именем myf. Теперь эту функцию можно вызывать в рабочем окне программы.

```
myf.m    Untitled.m    Untitled3.m

function [F] = myf(x)
F=[cos(x(1))-x(2);
x(1)+sin(x(2))];
end
```

Рисунок 2.2 – Создание m-файла с исходной СНУ

1) Итерационный метод наименьших квадратов.

Решение таких задач возможно и с помощью функции fsolve из пакета Optimization Toolbox, которая решает систему нелинейных уравнений вида f(x)=0 методом наименьших квадратов, ищет не только точки пересечения, но и точки касания.

Синтаксис функции:

[x,fval]=fsolve(@myf,x0,options), 2.4

где x – вывод значений неизвестных x1, x2;

fval – вывод значений СНУ;

(@myf – вызов пользовательской функции myf;

x0 – вектор начальный условий;

options – параметры вывода результатов расчета.

Тогда решение этой системы уравнений с выводом хода решения осуществляется следующим образом:

```
>> [x,fval]=fsolve(@myf, [-5 -5],optimset('display','iter'))
```

Рисунок 2.3 – Вызов функции fsolve с параметрами

Как видим, результаты вычислений СНУ в Matcad и Matlab дают практически идентичные результаты.

```
                                        Norm of      First-order   Trust-region
 Iteration  Func-count     f(x)          step         optimality    radius
     0          3         44.2474                         9.11               1
     1          6         25.7511               1         6.61               1
     2          9         12.1243             2.5         1.45             2.5
     3         12         6.96811             2.5         2.56             2.5
     4         15         1.70971          2.7175         1.37            6.25
     5         18         0.61752         1.28921        0.785            6.25
     6         21       0.0380458        0.678164          0.2            6.25
     7         24     3.91887e-05        0.154356      0.00618            6.25
     8         27     6.89321e-11      0.00533573     7.81e-06            6.25
     9         30     1.99468e-22     7.08955e-06     1.41e-11            6.25

Equation solved.

fsolve completed because the vector of function values is near zero
as measured by the default value of the function tolerance, and
the problem appears regular as measured by the gradient.

<stopping criteria details>

x =

   -0.6948    0.7682

fval =

   1.0e-10 *

   -0.1284
   -0.0589
```

Рисунок 2.4 – Результат работы функции fsolve с параметрами

2.3 Задание на выполнение работы

1) Внимательно ознакомиться с теоретическими сведениями лабораторной работы.

2) Выбрать свой вариант исходных данных. Исходные данные представлены в таблице Б.1 приложения Б. Номер варианта – это порядковый номер в журнале группы.

3) Подготовить алгоритм выполнения лабораторной работы.

4) Выполнить расчеты с использованием программного продукта Matlab.

5) Подготовить дополнительные информационные материалы таблицы, графики и др., если это необходимо.

6) Результаты оформляются в виде отчета по выполнению лабораторной работы. Краткий отчет содержит: цель работы, исходные данные, скриншоты или листинг программы расчетов, дополнительные информационные материалы, краткий анализ результатов работы.

2.4 Контрольные вопросы

1) Что такое нелинейное уравнение?

2) Что такое СНУ?

3) Какие методы используются для нахождения решения СНУ?

4) Какие функции используются для решения СНУ в Matlab?

Лабораторная работа №3. Анализ экспериментальных данных

Цель работы: изучение интерполяции и аппроксимации в программном продукте Matlab.

3.1 Теоретические сведения

При проведении различных экспериментов обычно требуется массив экспериментальных данных представить в виде функции, которую можно использовать в дальнейших расчетах. Если кривая, описываемая этой функцией, должна проходить через все экспериментальные точки, операция получения промежуточных точек и расчетной функции называется интерполяцией. Основная задача интерполяции - оценить значение представляемой данными зависимости в промежутках между ее узловыми точками. Для этого используются подходящие функции, значения которых в узловых точках совпадают с координатами этих точек. Например, при линейной интерполяции зависимости y(x) узловые точки просто соединяются друг с другом отрезками прямых, и считается, что искомые промежуточные точки расположены на этих отрезках.

Для повышения точности интерполяции применяют параболы (квадратичная интерполяция) или полиномы более высокой степени (полиномиальная интерполяция).

График аппроксимирующей функции может не проходить через узловые точки, но приближать их с некоторой (по возможности, малой) среднеквадратической погрешностью. Это характерно для регрессии - реализации метода наименьших квадратов (МНК).

Одна из наиболее известных аппроксимаций - полиномиальная аппроксимация. В Matlab определены функции аппроксимации данных полиномами по методу наименьших квадратов - полиномиальной регрессии. Это достаточно универсальный вид аппроксимации. Например, при степени полинома 1 имеем линейную регрессию, при степени полинома 2 - квадратичную и т.д.

Если необходимо уменьшить разброс данных или исключить некоторую систематическую погрешность, например, в виде наложенных колебаний, используют сглаживание данных или фильтрацию спектра колебаний данных.

3.2 Решение в Matlab

Рассмотрим интерполяцию в Matlab. Под интерполяцией обычно подразумевают вычисление таблично заданных значений функции A(t) в промежутках между узловыми точками с координатами x_i Линейная, квадратичная и полиномиальная интерполяции реализуются как частные случаи полиномиальной аппроксимации.

Для одномерной табличной интерполяции используется функция interpl.

yi = interpl(x, Y, xi) возвращает вектор yi, содержащий элементы, соответствующие элементам xi и полученные интерполяцией векторов x и Y. Вектор x определяет точки, в которых задано значение Y.

yi = interpl (x, Y, xi, method) позволяет с помощью параметра method задать метод интерполяции:

- ' nearest' - ступенчатая интерполяция;
- ' linear' - линейная интерполяция (принята по умолчанию);
- ' spline ' - кубическая сплайн-интерполяция;
- ' cubic ' или ' pchip ' - интерполяция многочленами Эрмита;
- ' vbcubic ' - кубическая интерполяция Matlab 5.

Для решения необходимо ввести исходные данные, затем ввести функцию interpl с параметрами и построить график.

```
x=[0 1 2 3 4 5 6];
y=[4.1 2.4 3 4.3 3.6 5.2 5.9];
xi=(0:0.1:6);
yi=interp1(x,y,xi);
plot(x,y,'o',xi,yi);
hold on;
yi2=interp1(x,y,xi,'spline');
plot(xi,yi2)
```

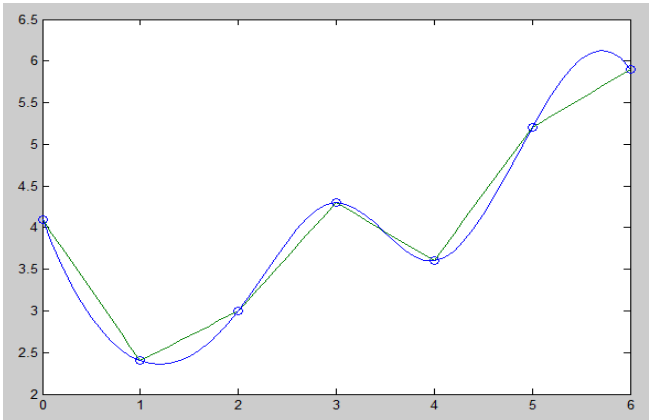

Рисунок 3.1 – Линейная интерполяция (зеленая линия) и кубическая сплайн-интерполяция (синяя линия)

Рассмотрим регрессию в Matlab. В системе Matlab определены функции аппроксимации данных полиномами по методу наименьших квадратов - полиномиальной регрессии. Это достаточно универсальный вид аппроксимации. Например, при степени полинома 1 мы имеем линейную регрессию, при степени полинома 2 – квадратичную и т. д. Полиномиальную регрессию реализует функция:

polyfit(x,y,n) возвращает вектор коэффициентов полинома p(x) степени n, который с наименьшей среднеквадратичной погрешностью аппроксимирует функцию y(x). Результатом является вектор-строка длиной n+1, содержащий коэффициенты полинома в порядке уменьшения степеней. Если x и y равно n+1, то реализуется обычная полиномиальная аппроксимация, при которой график полинома точно проходит через узловые точки с координатами (x, y), хранящиеся в векторах x и y. В противном случае точного совпадения графика с узловыми точками не наблюдается.

Данная функция работает в связке с функцией polyval (p, x), она возвращает значения полинома p, вычисленные в точках, заданных в массиве x. Полином p – вектор, элементы которого являются коэффициентами полинома в порядке уменьшения степеней, x может быть матрицей или вектором. В любом случае функция polyval вычисляет значения полинома p для каждого элемента x.

Хотя можно обойтись и без нее, используя элементы вектора p, возвращаемого функцией polyfit.

Для решения необходимо ввести исходные данные, вызвать функцию polyfit и polyval с параметрами, построить график.

```
x=[0 1 2 3 4 5 6];
y=[4.1 2.4 3 4.3 3.6 5.2 5.9];
p=polyfit(x,y,1);
f=polyval(p,x);
plot(x,y,'o',x,f);
hold on;
p1=polyfit(x,y,3);
f1=polyval(p1,x);
plot(x,f1)
```

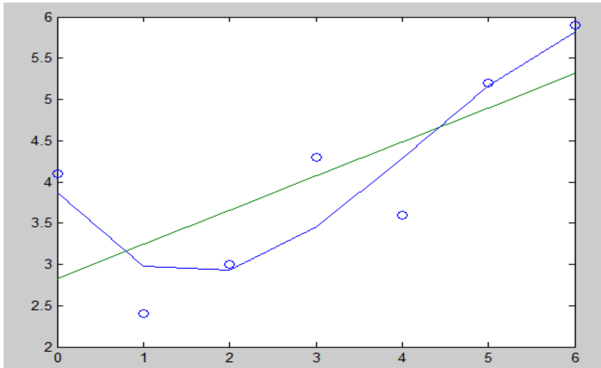

Рисунок 3.2 – Использование линейной регрессии и полиномиальной регрессии третьего порядка

3.3 Задание на выполнение работы

1) Внимательно ознакомиться с теоретическими сведениями лабораторной работы.

2) Выбрать свой вариант исходных данных. Исходные данные представлены в таблице В.1 приложения В. Номер варианта – это порядковый номер в журнале группы.

3) Подготовить алгоритм выполнения лабораторной работы.

4) Выполнить расчеты с использованием программного продукта Matlab.

5) Подготовить дополнительные информационные материалы таблицы, графики и др., если это необходимо.

6) Результаты оформляются в виде отчета по выполнению лабораторной работы. Краткий отчет содержит: цель работы, исходные данные, скриншоты или листинг программы расчетов, дополнительные информационные материалы, краткий анализ результатов работы.

3.4 Контрольные вопросы

1) Что такое интерполяция?
2) Что такое аппроксимация?
3) Что такое регрессия?
4) В чем разница между интерполяцией и регрессией?
5) Что такое сплайн-интерполяция?

Лабораторная работа №4. Численное решение интегральных уравнений

Цель работы: изучение методов численного решения интегральных уравнений с помощью программного продукта Matlab.

4.1 Теоретические сведения

Численное интегрирование традиционно является одной из важнейших сфер применения математического аппарата. Программный продукт Matlab позволяет численно решать определенные интегралы и символьно вычислять неопределенные интегралы.

Численное интегрирование заключается в приближенном вычислении определенного интеграла вида:

$$\int_a^b f(\mathrm{x})\,\mathrm{dx} \tag{4.1}$$

одним из многочисленных численных методов.

В основе способов вычисления интегралов численными методами лежит представление определенного интеграла как некоторой суммы:

$$I = \int_a^b f(\mathrm{x})\,\mathrm{dx} = \sum_{i=1}^{n} I_i = \sum_{i=1}^{n} \int_{x_{i-1}}^{x_i} f(\mathrm{x})\,\mathrm{dx} \tag{4.2}$$

где $[x_{i-1}, x_i]$ – n замкнутых интервалов, на которые разбивается отрезок интегрирования [a, b], причем $a=x_0$ и b=x.

Задача приближенного интегрирования по формуле (4.2) практически сводится к эффективному разбиению отрезка [a, b] на соответствующие интервалы $[x_{i-1}, x_i]$ и правильному описанию вида подынтегральной функции f(x).

В простейших случаях для интерполяции подынтегральной функции на интервале используются многочлены:

$P_0(x) = a_0$(метод прямоугольников) (4.3)

$P_1(x) = a_0+a_1\cdot x$ (метод трапеций) (4.4)

$P_2(x) = a_a + a_1\cdot x + a_n\cdot x^n$ (метод парабол) (4.5)

В действительности, вид подынтегральной функции обычно значительно сложнее, и приведенные простые функции могут удовлетворительно применяться только на небольших интервалах $[x_{i-1}, x_i]$. Отсюда следует, что для достижения требуемой точности надо задаваться как можно меньшим шагом интегрирования. Однако это может привести к существенному увеличению вычислительных затрат, что также нежелательно.

4.2 Решение в Matlab

Рассмотрим вычисление определенного интеграла. В Matlab существеует более десяти функций решения различных интегральных уравнений. Для одинарных интегральный уравнений используются: метод трапеций, метод Симпсона и т.д.

Так как метод трапеций обеспечивает невысокую точность при заданном числе шагов или дает слишком большое число шагов при вычислениях с заданной погрешностью поэтому его расматривать не будем.

Рассмотрим функции которые осуществляют интегрирование и двойное интегрирование, используя более точную квадратурную формулу Симпсона или метод Гаусса-Лобатто.

Квадратура – численный метод нахождения площади под графиком функции f(x), то есть вычисление определенного интеграла вида (4.1).

В приведенных ниже формулах подынтегральное выражение fun обычно задается или в прямых апострофах, или в форме handle-функции.

Функции quad и quadl используют два различных алгоритма квадратуры для вычисления определенного интеграла. Функция quad выполняет интегрирование по методу низкого порядка, используя рекурсивное правило Симпсона. Но она может быть более эффективной при негладких подынтегральных функциях или при низкой требуемой точности вычислений.

Функция quad (fun, a, b) возвращает численное значение определенного интеграла от заданной функции @fun на отрезке [a, b]. Используется адаптивный метод Симпсона. Также можно задавать в качестве параметра необходимую точность решения, например, quad (fun, a, b, tol) возвращает численное значение определенного интеграла с заданной относительной погрешностью tol.

В версии Matlab 2016 введена функция integral. Синтаксис функции:

q = integral(fun,xmin,xmax) (4.6)

где fun – подинтегральное выражение, описанное ранее;

xmin – начальный предел интегрирования;

xmax – конечный предел интегрирования.

Подинтегральная функция описывается следующим образом:

fun=@(x)f(x),

где fun – имя подинтегральной функции;

@(x) – оператор функции и переменная функции, относительно которой будет производиться интегрирование;

f(x) –подинтегральная функция.

В качестве пределов можно использовать не только числовые значения, но и бесконечность «Inf».

Пример представлен на рисунке 4.1.

```
>> quad('sin(x)',0,pi)

ans =

    2.0000

>> fun=@(x)log(x).*(x+1);
>> integral(fun,0,2)

ans =

   -0.2274
```

Рисунок 4.1 – Численное решение определенного интеграла

Функция quadl (квадратура Лобатто) использует адаптивное правило квадратуры Гаусса-Лобатто очень высокого порядка.

Пример представлен на рисунке 4.2.

```
>> fun=@(x)log(x).*(x+1);
>> quadl(fun,0,2)

ans =

   -0.2274
```

Рисунок 4.2 – Численное решение определенного интеграла

Рассмотрим решение неопределенных интегралов в Matlab. Для выполнения аналитического интегрирования нужно использовать пакет расширения Symbolic Math Toolbox. Пакет прикладных программ Symbolic Math Toolbox дает системе Matlab принципиально новые возможности решения задач в символьном (аналитическом) виде, включая реализацию точной арифметики произвольной разрядности. Обеспечивает выполнение символьного дифференцирования и интегрирования, вычисление сумм и произведений, разложение в ряды Тейлора и Маклорена, операции со степенными многочленами (полиномами), вычисление корней полиномов, решение в аналитическом виде нелинейных уравнений, всевозможные символьные преобразования, подстановки и многое другое.

Для работы с пакетом надо задать неопределенные символьные переменные. Они объявляются с помощью команды syms, например:

$$>> \text{syms x.} \tag{4.7}$$

После объявления символьной переменной необходимо вызвать функцию символьного вычисления интеграла int(). Если единственным аргументом указано символьное выражение с одной символьной переменной, то в результате возвращается первообразная для выражения (если эту первообразную удается найти). Если в выражении несколько символьных переменных, интеграл вычисляется по переменной, наиболее близкой в алфавитном списке к букве x. Чтобы явно указать переменную интегрирования, ее передают вторым аргументом функции int (). Пример приведен на рисунке 4.3.

```
>> syms x
>> int(sin(x)*x^2)

ans =

2*x*sin(x) - cos(x)*(x^2 - 2)
```

Рисунок 4.3 – Символьное вычисление неопределенного интеграла

4.3 Задание на выполнение работы

1) Внимательно ознакомиться с теоретическими сведениями лабораторной работы.

2) Выбрать свой вариант исходных данных. Исходные данные представлены в таблице Г.1 приложения Г. Номер варианта – это порядковый номер в журнале группы.

3) Подготовить алгоритм выполнения лабораторной работы.

4) Выполнить расчеты с использованием программного продукта Matlab.

5) Подготовить дополнительные информационные материалы таблицы, графики и др., если это необходимо.

6) Результаты оформляются в виде отчета по выполнению лабораторной работы. Краткий отчет содержит: цель работы, исходные данные, скриншоты или листинг программы расчетов, дополнительные информационные материалы, краткий анализ результатов работы.

4.4 Контрольные вопросы

1) Какие методы численного интегрирования вы знаете?

2) Что такое определенный интеграл?

3) Что такое неопределенный интеграл?

4) Какие действия выполняет функция syms?

5) Как выполнить аналитическое вычисление интегрального уравнения в Matlab?

Лабораторная работа №5. Численное решение дифференциальных уравнений (задачи Коши)

Цель работы: изучение методов численного решения дифференциальных уравнений (задачи Коши) с помощью программного продукта Matlab.

5.1 Теоретические сведения

Задача численного дифференцирования возникает в тех случаях, когда дифференцируемая функция f(x) имеет сложное аналитическое выражение для непосредственного определения производных или же задана таблично. В этих случаях вместо функции f(x) используют соответствующую интерполирующую зависимость. Если интерполирующая зависимость с достаточной степенью точности совпадает с заданной функцией, то считают, что производные от этих зависимостей отличаются незначительно.

Обыкновенным дифференциальным уравнением (ОДУ) n-го порядка называется выражение:

$$y^{(n)} = f(x, y, y', y'', ..., y^{(n-1)}), \tag{5.1}$$

устанавливающее взаимосвязь между одной независимой переменной x, неизвестной (искомой) функцией y и ее производными y', y'',..., $y^{(n-1)}$.

Решением дифференциального уравнения является некоторая функция y(x), которая при подстановке в уравнение обращает его в тождество. Существует множество решений (частных решений) дифференциального уравнения (5.1), которые могут быть объединены и записаны в виде общего решения:

$$y = y(x, C_1, C_2, ..., C_n) \tag{5.2}$$

где $C_1, C_2, ..., C_n$ – произвольные постоянные.

Для выделения одного частного решения уравнения (5.1) необходимо задать n условий, которые единственным образом определяли бы постоянные $C_1, C_2, ..., C_n$. Если эти условия заданы в одной точке x_0, и представляют собой совокупность значений искомой функции y(x) и всех ее производных до (n-1)-го порядка включительно в этой точке:

$$\begin{cases} y(x_0) = y_0 \\ y'(x_0) = y'_0 \\ .... \\ y^{(n-1)}(x_0) = y_0^{(n-1)} \end{cases} \tag{5.3}$$

то задача интегрирования дифференциального уравнения (5.1) называется задачей Коши, а условия (5.3) называются начальными условиями.

5.2 Решение в Matlab

Для решения систем ОДУ в Matlab реализованы различные численные методы. Их реализации названы решателями ОДУ.

Решатели реализуют следующие методы решения обычных дифференциальных равнений и систем дифференциальных уравнений:

1) ode45 – одношаговые явные методы Рунге-Кутта 4-го и 5-го порядков в модификации Дорманда и Принца. Это классический метод, рекомендуемый для начальной пробы решения. Во многих случаях он дает хорошие результаты – если система решаемых уравнений нежесткая.

2) ode23 – одношаговые явные методы Рунге-Кутта 2-го и 4-го порядков в модификации Богацки и Шампина. При умеренной жесткости системы ОДУ и низких требованиях к точности этот метод может дать выигрыш в скорости решения.

3) ode113 – многошаговый метод Адамса-Башворта-Мултона переменного порядка класса предиктор-корректор. Это адаптивный метод, который может обеспечить высокую точность решения.

4) ode15s – многошаговый метод переменного порядка (от 1 до 5, по умолчанию 5), использующий формулы численного «дифференцирования назад». Это адаптивный метод, его стоит применять, если решатель ode45 не обеспечивает решения и система дифференциальных уравнений жесткая.

5) ode23s – одношаговый метод, использующий модифицированную формулу Розенброка 2-го порядка. Может обеспечить высокую скорость вычислений при низкой точности решения жесткой системы дифференциальных уравнений.

Существуют и другие решатели направленные на вычисление ОДУ.

Перейдем к описанию синтаксиса функций для решения дифференциальных уравнений (под именем solver подразумевается любая из представленных выше функций).

$$[T, Y] = solver (@F\ tspan, y_0) \tag{5.4}$$

где @F – дифференциальное уравнение;

tspan – вектор, определяющий интервал интегрирования;

y_0 – начальное условие.

Данная функция интегрирует дифференциальные уравнения вида y’=F(t,y) на интервале tspan с начальными условиями y_0. Каждая строка в массиве решений Y соответствует значению времени, возвращаемому в векторе-столбце T.

Пример представлен на рисунке 5.1.

Произведем вычисления ОДУ с помощью функции ode113.

Пример представлен на рисунке 5.2.

```
>> [T,Y]=ode45(@(x,y)(y-x.^2),1:0.3:2.2,0);
>> plot(T,Y)
```

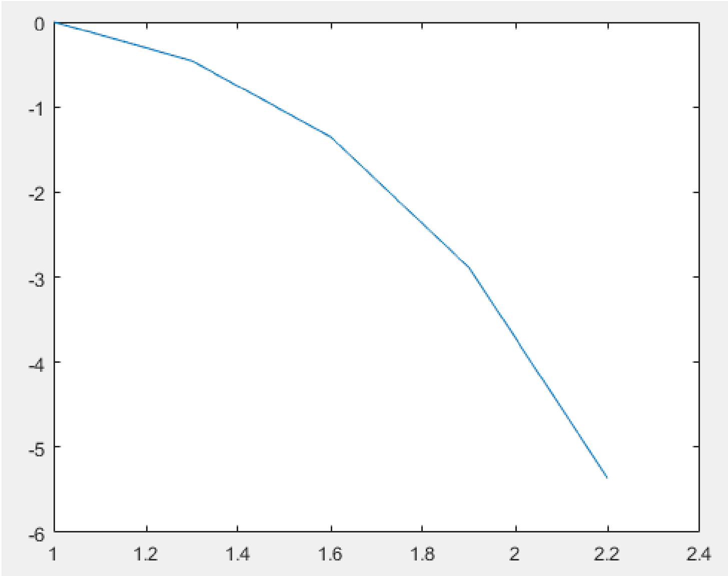

Рисунок 5.1 – Численное решение ОДУ функцией ode45

```
>> [T,Y]=ode113(@(x,y)(y-x.^2),1:0.3:2.2,0);
>> plot(T,Y)
```

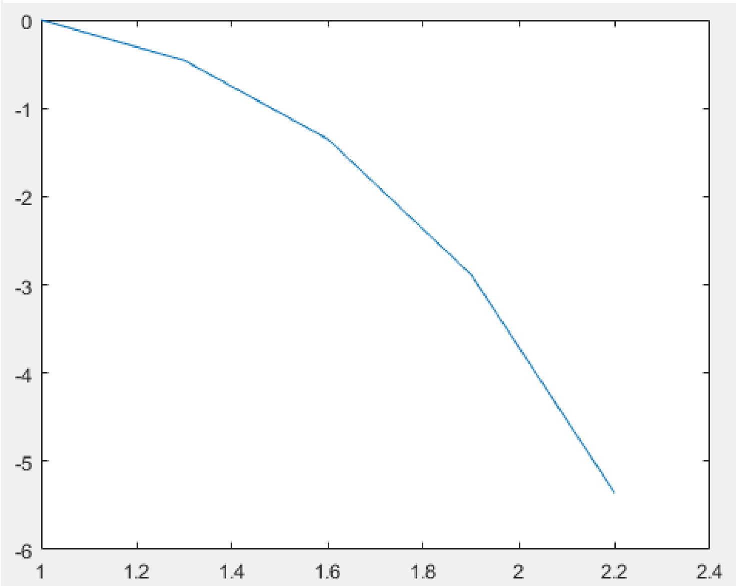

Рисунок 5.2 – Численное решение ОДУ функцией ode113

5.3 Задание на выполнение работы

1) Внимательно ознакомиться с теоретическими сведениями лабораторной работы.

2) Выбрать свой вариант исходных данных. Исходные данные представлены в таблице Д.1 приложения Д. Номер варианта – это порядковый номер в журнале группы.

3) Подготовить алгоритм выполнения лабораторной работы.

4) Выполнить расчеты с использованием программного продукта

Matlab.

5) Подготовить дополнительные информационные материалы таблицы, графики и др., если это необходимо.

6) Результаты оформляются в виде отчета по выполнению лабораторной работы. Краткий отчет содержит: цель работы, исходные данные, скриншоты или листинг программы расчетов, дополнительные информационные материалы, краткий анализ результатов работы.

5.4 Контрольные вопросы

1) Что такое задача Коши?

2) Что такое стандартная форма записи ОДУ?

3) Какие функции решения ОДУ в Matlab вы знаете?

4) Что такое жесткое ОДУ?

Лабораторная работа №6. Численное решение системы дифференциальных уравнений(задачи Коши)

Цель работы: изучение методов численного решения системы дифференциальных уравнений (задачи Коши) с помощью программного продукта Matlab.

6.1 Теоретические сведения

Рассмотрим систему дифференциальных уравнений первого порядка:

$$\frac{dy_i^{(n)}}{dx} = f_i(\mathrm{x}, \mathrm{y}_1(\mathrm{x}), ..., \mathrm{y}_n(\mathrm{x})), i = 1,..,\mathrm{n} \quad (6.1)$$

с начальными условиями

$$y_i(\mathrm{x}_0) = \mathrm{y}_{i0}, i = 1,..,\mathrm{n} \quad (6.2)$$

В результате ее решения могут быть получены функции со стационарными точками решения, так называемые решения устойчивые в «большом», когда после некоторого значения независимой переменной x, например x_5, величина искомой функции-решения становится константой, т.е.

$$y_i(\mathrm{x}) = const, i = 1,..,\mathrm{n} \quad (6.2)$$

и соответственно:

$$\frac{dy_i}{dx}(\mathrm{x}) = 0, i = 1,..,\mathrm{n} \quad (6.2)$$

При всех $x>x_5$.

Таким образом, устойчивые решения дифференциальных уравнений (кривая 1 на рисунке 6.1) представляют собой функции, состоящие из двух частей:

1) области стационарности (I), в которой они не меняются и их значения равны соответствующим стационарным точкам решения;

2) области нестационарности (II) (так называемые области «переходного процесса»), в которой значения функций изменяются с изменением величины независимой переменной x.

Неустойчивые решения (кривая 2 на рисунке 6.1) не содержат стационарных точек решения и всегда изменяются с изменением независимой переменной x.

Очевидно, что больший интерес представляют устойчивые решения дифференциальных уравнений (решения, устойчивые «в большом»), которые с изменением независимой переменной x достигают областей стационарности. Эти области соответствуют областям нормальной эксплуатации большей части технологических процессов в непрерывных режимах.

Поэтому анализ обусловленности задачи решения систем дифференциальных уравнений (устойчивости «в малом»), в большинстве случаев, выполняется для систем с устойчивыми «в большом» решениями и в двух областях решений – области стационарности (чаще всего) и области нестационарности (реже). По существу, в этом случае речь идет о влиянии незначительных возмущений (поэтому говорят об устойчивости в «малом») на результаты решения задачи. Если результаты решения (искомые функции) изменяются также незначительно, то задача решения обыкновенного дифференциального уравнения считается хорошо обусловленной, в противном случае – плохо обусловленной.

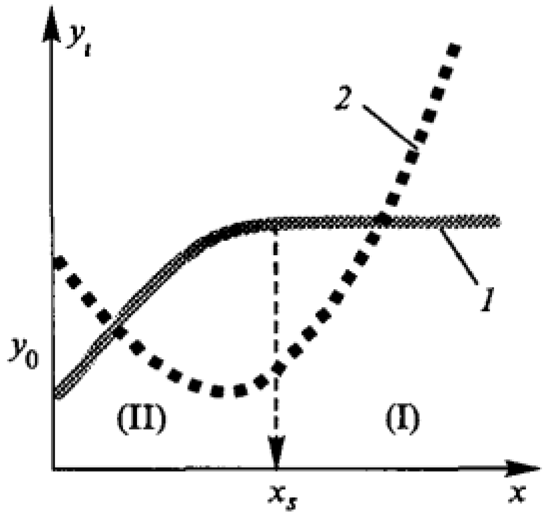

Рисунок 6.1 – Графическое изображение устойчивых и неустойчивых в «большом» решений обыкновенных дифференциальных уравнений

6.2 Решение в Matlab

В Matlab для решения СОДУ используются те же самые решатели что и для решения ОДУ. Сами решатели представлены в пункте 5.2. лабораторной работы №5.

Перейдем к описанию синтаксиса функций для решения систем дифференциальных уравнений (под именем solver подразумевается любая из представленных выше функций).

$$[T, Y] = solver (@F\ tspan, y_0) \quad (6.3)$$

где @F – система дифференциальное уравнение;

tspan – вектор, определяющий интервал интегрирования;

y_0 – начальное условие.

Данная функция интегрирует систему дифференциальных уравнений вида y'=F(t,y) на интервале tspan с начальными условиями y_0. Каждая строка в массиве решений Y соответствует значению времени, возвращаемому в векторе-столбце T.

Расмотрим применение решателя ОДУ ode15s на ставшем классическим примере – решении нелинейного дифференциального уравнения второго порядка (уравнения Ван-дер-Поля), записанного в виде системы из двух дифференциальных уравнений:

$$\begin{cases} \dfrac{dy_1}{du} = y_2 \\ \dfrac{dy_2}{du} = m \cdot (1 - y_1{}^2) \cdot y_2 - y_1 \end{cases}, \text{ при начальных условиях } \begin{cases} y_1(0) = 0 \\ y_2(0) = 1 \end{cases}, \qquad (6.4)$$

Это уравнение описывает колебания в нелинейной системе второго порядка, например в LC-генераторе на электронной лампе или полевом транзисторе, и является классическим примером математического моделирования этих устройств. Поведение системы Ван-дер-Поля существенно зависит от параметра m который задает степень влияния нелинейности на возникновение и развитие колебаний. При больших m представленная система ОДУ является жесткой. Возьмем значение m=100.

Перед решением нужно записать систему дифференциальных уравнений в виде ODE-функции. Для этого в главном меню выберем File/New/M-File и введем данные, как на рисунке 6.2.

```
function dydt = vdp100(t,y)
dydt = zeros(2,1); dydt(1) = y(2);
dydt(2) = 100*(1 - y(1)^2)*y(2) - y(1);
```

Рисунок 6.2 – Создание ODE-функции

Тогда решение решателем odel5s и сопровождающий его график (рисунок 6.3) можно получить, используя следующие команды:

```
>> [T,Y]=ode15s(@vdp100,[0 30],[2 0]); plot(T,Y)
>> hold on; gtext('y1'), gtext('y2')
```

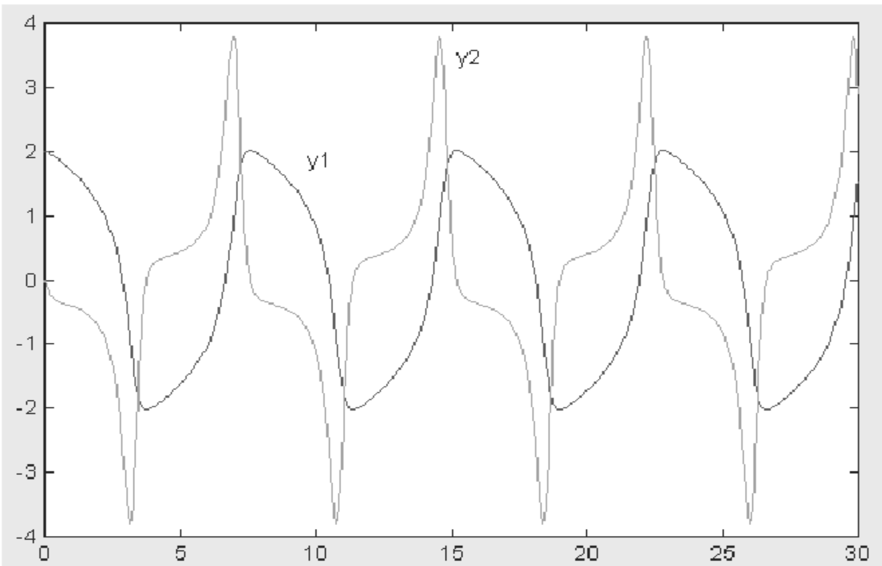

Рисунок 6.3 – Численное решение СОДУ функцией ode15s

Последние команды позволяют с помощью мыши нанести па графики решений y_1=y(1) и y_2=y(2) помечающие их надписи.

6.3 Задание на выполнение работы

1) Внимательно ознакомиться с теоретическими сведениями лабораторной работы.

2) Выбрать свой вариант исходных данных. Исходные данные представлены в таблице Е.1 приложения Е. Номер варианта – это порядковый номер в журнале группы.

3) Подготовить алгоритм выполнения лабораторной работы.

4) Выполнить расчеты с использованием программного продукта Matlab.

5) Подготовить дополнительные информационные материалы таблицы, графики и др., если это необходимо.

6) Результаты оформляются в виде отчета по выполнению лабораторной работы. Краткий отчет содержит: цель работы, исходные данные, скриншоты или листинг программы расчетов, дополнительные информационные материалы, краткий анализ результатов работы.

6.4 Контрольные вопросы

1) Что такое задача Коши?

2) Что такое стандартная форма записи СОДУ?

3) Какие функции решения СОДУ в Matlab вы знаете?

4) Что такое жесткое СОДУ?

Лабораторная работа №7. Составление статической математической модели смесителя потоков

Цель работы Составление статической математической модели смесителя потоков и расчет ее с помощью программного продукта Matlab.

7.1 Теоретические сведения

При составлении математического описания смесителя потоков (рисунок 7.1) воспользуемся следующими допущениями:

1) структура потока в аппарате соответствует режиму идеального смещения;

2) режим смешения в аппарате – установившийся.

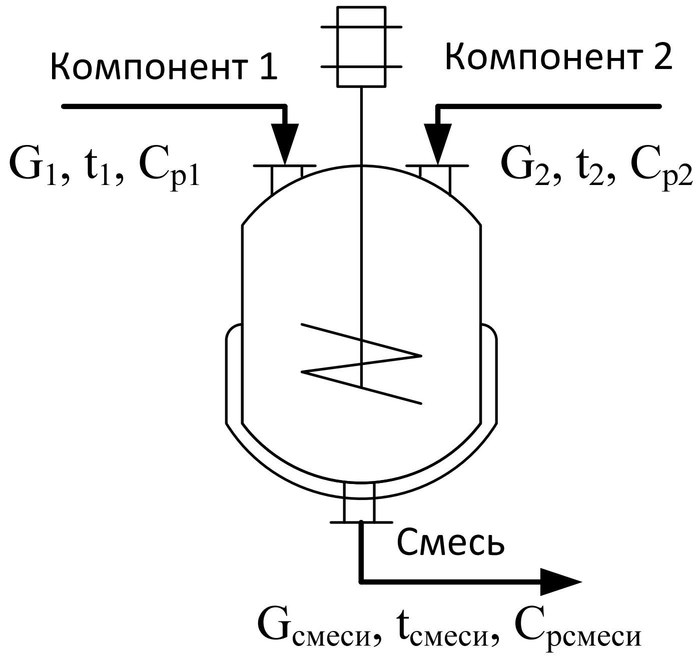

Рисунок 7.1 – Смеситель потоков

На рисунке 7.1 обозначены буквами G_i, t_i, C_{pi}, – расход, температура и теплоемкость i-го технологического потока. В нашем случае i=l, 2, а теплоемкости компонентов i-го потока рассчитываются при температуре этого потока.

Общее уравнение материального баланса имеет вид

$$G_{смеси} = \sum G_i \text{ , i= 1, 2, … N,} \tag{7.1}$$

где $G_{смеси}$ – расход выходного потока;

G_i – расходы компонентов смеси;

N – число веществ в потоке.

Уравнение теплового баланса имеет вид

$$Q_{смеси} = \sum Q_i - Q_{потерь}, \quad (7.2)$$

где $Q_{смеси}$ – теплота смеси;

Q_i – теплота компонентов смеси;

$Q_{потерь}$ – теплота потерь в окружающую среду (можно принять равной 5%).

Теплота компонентов определяется:

$$Q_i = G_i \cdot C_{pi} \cdot t_i \quad (7.3)$$

С учетом 7.3, 7.2 уравнение теплового баланса примет вид

$$G_{смеси} \cdot C_{pсмеси} \cdot t_{смеси} = 0{,}95 \cdot \sum G_i \cdot C_{pi} \cdot t_i\ , \quad (7.4)$$

где $C_{pсмеси}$, $t_{смеси}$ – удельная теплоемкость и температура смеси;

C_{pi}, t_i, – удельные теплоемкости и температуры компонентов смеси.

Выразим температуру из 7.4

$$t_{смеси} = \frac{0{,}95 \cdot \sum G_i \cdot C_{pi} \cdot t_i}{G_{смеси} \cdot C_{pсмеси}} \quad (7.5)$$

Температурная зависимость удельной теплоемкости смеси определяется в виде

$$C_{pсмеси} = \sum C_{pi} \cdot \frac{G_i}{G_{смеси}}\ , \quad (7.6)$$

Перепишем уравнение 7.6 с учетом 7.7 для двухкомпонентной смеси

$$t_{смеси} = 0.95 \cdot \frac{G_1 \cdot C_{p1} \cdot t_1 + G_2 \cdot C_{p2} \cdot t_2}{G_1 \cdot C_{p1} + G_2 \cdot C_{p2}}. \quad (7.7)$$

7.2 Решение в Matlab

Для решения в Matlab необходимо создать новый скрипт. Внести необходимые формулы и переменные и выполнить скрипт. Также необходимо построить графики зависимости температуры от расходов G_1 и G_2.

7.3 Задание на выполнение работы

1) Внимательно ознакомиться с теоретическими сведениями лабораторной работы.

2) Выбрать свой вариант исходных данных. Исходные данные представлены в таблице Ж.1 приложения Ж. Номер варианта – это порядковый номер в журнале группы.

3) Подготовить алгоритм выполнения лабораторной работы.

4) Выполнить расчеты с использованием программного продукта Matlab

5) Построить в одних координатах графики зависимости температуры смеси от расходов компонентов.

6) Результаты оформляются в виде отчета по выполнению лабораторной работы. Краткий отчет содержит: цель работы, исходные данные, скриншоты или листинг программы расчетов, дополнительные информаци-

онные материалы, краткий анализ результатов работы.

7.4 Контрольные вопросы

1) Какие допущения используются для построения математических моделей?

2) Что такое статическая математическая модель?

3) Что такое смеситель потоков?

4) Что такое математическая модель?

Лабораторная работа №8. Составление статической математической модели теплообменника

Цель работы Составление статической математической модели теплообменника и расчет ее с помощью программного продукта Matlab.

8.1 Теоретические сведения

При построении математического описания теплообменника (рисунок 8.1) применяются следующие допущения:

- рассматривается стационарный режим;
- теплоотдача не сопровождается изменением агрегатного состояния теплоносителей;
- потери тепла не учитываются;
- схема движения теплоносителей – противоточная;
- коэффициенты теплоотдачи в трубном и межтрубном пространствах рассчитываются при начальных температурах теплоносителей;
- теплоноситель, отдающий теплоту, направляется в трубы, а теплоноситель, воспринимающий теплоту – в межтрубное пространство.

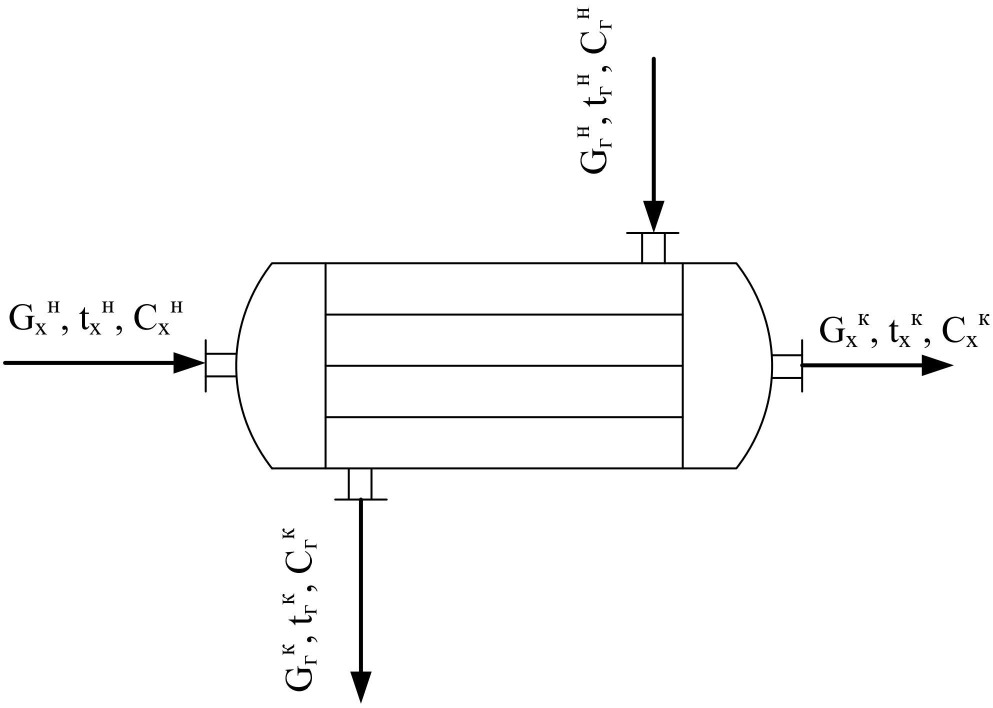

Рисунок 8.1 – Теплообменник

На рисунке 8.1 обозначены буквами $G_{х(г)}$, $t_{х(г)}$, $C_{х(г)}$ – расход, температура и теплоемкость холодного(горячего) потока. Причем $G^{н(к)}$, $t^{н(к)}$, $C^{н(к)}$ начальные(конечные) состояние потоков.

Так как теплообменник не изменяет состава материальных потоков, то

$$G_х^н=G_х^к,\ G_г^н= G_г^к. \tag{8.1}$$

Количество теплоты, переданное через площадь теплообмена в секунду в противоточном теплообменнике, равно

$$Q=k_т\cdot\frac{\Delta t_2-\Delta t_1}{\ln\left(\frac{\Delta t_2}{\Delta t_1}\right)}\cdot F, \tag{8.2}$$

где $k_т$ – коэффициент теплопередачи;

F – площадь поверхности теплообмена;

$\Delta t_1 = t_г^к-t_х^н$;

$\Delta t_2 = t_г^н-t_х^к$.

Количество теплоты, отданное горячим теплоносителем

$$Q=w_г\cdot(t_г^н-t_г^к), \tag{8.3}$$

воспринятое холодным теплоносителем

$$Q=w_х\cdot(t_х^к-t_х^н), \tag{8.4}$$

При этом водяные эквиваленты равны:

$$w_г= G_г\cdot C_г,\ w_х= G_х\cdot C_х. \tag{8.5}$$

Уравнение теплового баланса теплообменника имеет вид

$$w_г\cdot(t_г^н-t_г^к)= w_х\cdot(t_х^к-t_х^н). \tag{8.6}$$

Из уравнения (8.6) находим

$$t_г^к=t_х^к-\frac{t_х^к-t_х^н}{\rho}, \tag{8.7}$$

где $\rho=\frac{w_г}{w_х}$.

Так как потерями теплоты пренебрегаем, то

$$w_г\cdot(t_г^н-t_г^к)=k_т\cdot F\cdot\frac{(t_г^н-t_х^к)-(t_г^к-t_х^н)}{\ln\left(\frac{t_г^н-t_х^к}{t_г^к-t_х^н}\right)} \tag{8.8}$$

Подставив (8.7) в (8.8) и выполнив простейшие преобразования, получим

$$t_х^к=t_х^н+(t_г^н-t_х^н)\cdot\rho\cdot\frac{\exp(\beta\cdot(1-\rho))-1}{\exp(\beta\cdot(1-\rho))-\rho}, \tag{8.9}$$

где $\beta=k_г\cdot\frac{F}{w_г}$.

Коэффициент теплопередачи рассчитывается по формуле

$$k_т=\frac{1}{\frac{1}{\alpha_т}+\sum r_{ст}+\frac{1}{\alpha_м}}=\frac{1}{\frac{1}{\alpha_т}+\frac{\delta}{\lambda}+\frac{1}{\alpha_м}}, \tag{8.10}$$

где $\alpha_т$, $\alpha_м$ – коэффициенты теплоотдачи в трубном и межтрубном пространстве;

$\sum r_{ст}$ – сумма термических сопротивлений стенки;

δ – толщина стенки внутренней трубы;

λ – коэффициент теплопроводности.

Для упрощения расчетов примем коэффициент теплопередачи равным 3800 $Вт/м^2·К$.

Необходимые данные для составления математической модели теплообменника $d_{вн}$=0,021 м; $d_{нар}$=0,025 м; длина труб L=1.25 м; $n_т$= 20 шт.

8.2 Решение в Matlab

Для решения в Matlab необходимо создать новый скрипт. Внести необходимые формулы и переменные и выполнить скрипт. Также необходимо построить графики зависимости температуры от расходов G_1 и G_2.

8.3 Задание на выполнение работы

1) Внимательно ознакомиться с теоретическими сведениями лабораторной работы.

2) Выбрать свой вариант исходных данных. Исходные данные представлены в таблице Ж.1 приложения Ж. Номер варианта – это порядковый номер в журнале группы.

3) Подготовить алгоритм выполнения лабораторной работы.

4) Выполнить расчеты с использованием программного продукта Matlab

5) Построить в одних координатах графики зависимости температуры смеси от расходов компонентов.

6) Результаты оформляются в виде отчета по выполнению лабораторной работы. Краткий отчет содержит: цель работы, исходные данные, скриншоты или листинг программы расчетов, дополнительные информационные материалы, краткий анализ результатов работы.

8.4 Контрольные вопросы

1) Какие допущения используются для построения математических моделей?

2) Что такое статическая математическая модель?

3) Что такое кожухотрубный теплообменник?

4) Что такое математическая модель?

Список рекомендуемой литературы

1 Гартман Т.Н. Основы компьютерного моделирования химик-технологичсеких процессов / Т.Н. Гартман, Д.В. Клушин. – М.: ИКЦ Академкнига, 2006. – 416 с.

2 Дьяконов В.П. Matlab. Полный самоучитель / В.П. Дьяконов. – М.: ДМК пресс, 2012. – 768 с.

Приложение А – Исходные даные

Таблица А.1 – Исходны даные для лабораторной работы №1

Вариант	Задание	Вариант	Задание
1	2	3	4
1	$\begin{cases} 5\cdot x_1+1\cdot x_2+1\cdot x_3=10 \\ 1\cdot x_1+3\cdot x_2-1\cdot x_3=4 \\ -3\cdot x_1+2\cdot x_2+4\cdot x_3=6 \end{cases}$	16	$\begin{cases} -4\cdot x_1+2\cdot x_2+1\cdot x_3=-5 \\ -1\cdot x_1+5\cdot x_2+1\cdot x_3=-5 \\ 2\cdot x_1+0\cdot x_2+3\cdot x_3=5 \end{cases}$
2	$\begin{cases} 2\cdot x_1+1\cdot x_2-1\cdot x_3=-3 \\ -1\cdot x_1+3\cdot x_2+1\cdot x_3=2 \\ 1\cdot x_1-1\cdot x_2+4\cdot x_3=3 \end{cases}$	17	$\begin{cases} 5\cdot x_1-1\cdot x_2+1\cdot x_3=-3 \\ -1\cdot x_1+5\cdot x_2+1\cdot x_3=-1 \\ 2\cdot x_1+1\cdot x_2+4\cdot x_3=1 \end{cases}$
3	$\begin{cases} 2\cdot x_1+0\cdot x_2-1\cdot x_3=1 \\ 1\cdot x_1-3\cdot x_2+1\cdot x_3=2 \\ 1\cdot x_1+1\cdot x_2+4\cdot x_3=4 \end{cases}$	18	$\begin{cases} 4\cdot x_1+2\cdot x_2+1\cdot x_3=3 \\ 3\cdot x_1+5\cdot x_2-1\cdot x_3=4 \\ -2\cdot x_1+1\cdot x_2-4\cdot x_3=6 \end{cases}$
4	$\begin{cases} 5\cdot x_1+1\cdot x_2-1\cdot x_3=-5 \\ -1\cdot x_1+3\cdot x_2+1\cdot x_3=5 \\ 1\cdot x_1-2\cdot x_2+4\cdot x_3=1 \end{cases}$	19	$\begin{cases} 4\cdot x_1+1\cdot x_2+2\cdot x_3=-6 \\ 0\cdot x_1+3\cdot x_2+1\cdot x_3=-1 \\ 1\cdot x_1+2\cdot x_2+4\cdot x_3=-5 \end{cases}$
5	$\begin{cases} 3\cdot x_1+1\cdot x_2-1\cdot x_3=-1 \\ -2\cdot x_1+4\cdot x_2+1\cdot x_3=5 \\ 1\cdot x_1+1\cdot x_2+3\cdot x_3=-3 \end{cases}$	20	$\begin{cases} 3\cdot x_1+1\cdot x_2+1\cdot x_3=6 \\ 0\cdot x_1-2\cdot x_2+1\cdot x_3=-3 \\ 2\cdot x_1-1\cdot x_2+4\cdot x_3=-1 \end{cases}$
6	$\begin{cases} 3\cdot x_1+1\cdot x_2-1\cdot x_3=6 \\ 2\cdot x_1+4\cdot x_2+1\cdot x_3=9 \\ 1\cdot x_1-1\cdot x_2+3\cdot x_3=4 \end{cases}$	21	$\begin{cases} 2\cdot x_1-1\cdot x_2+0\cdot x_3=-5 \\ 2\cdot x_1+5\cdot x_2+1\cdot x_3=2 \\ 2\cdot x_1+1\cdot x_2-4\cdot x_3=-7 \end{cases}$
7	$\begin{cases} 2\cdot x_1-1\cdot x_2+0\cdot x_3=-2 \\ 2\cdot x_1+5\cdot x_2-2\cdot x_3=-4 \\ 1\cdot x_1-1\cdot x_2+3\cdot x_3=2 \end{cases}$	22	$\begin{cases} 5\cdot x_1+0\cdot x_2+1\cdot x_3=11 \\ 1\cdot x_1-3\cdot x_2+1\cdot x_3=3 \\ 3\cdot x_1-1\cdot x_2+5\cdot x_3=11 \end{cases}$
8	$\begin{cases} 3\cdot x_1-1\cdot x_2+1\cdot x_3=1 \\ 0\cdot x_1+2\cdot x_2-1\cdot x_3=3 \\ -1\cdot x_1+1\cdot x_2+5\cdot x_3=-5 \end{cases}$	23	$\begin{cases} 1\cdot x_1+1\cdot x_2+1\cdot x_3=12 \\ 2\cdot x_1-7\cdot x_2-3\cdot x_3=16 \\ 8\cdot x_1+4\cdot x_2-6\cdot x_3=13 \end{cases}$
9	$\begin{cases} 4\cdot x_1+1\cdot x_2-1\cdot x_3=7 \\ 2\cdot x_1+3\cdot x_2+0\cdot x_3=7 \\ 1\cdot x_1-1\cdot x_2+5\cdot x_3=11 \end{cases}$	24	$\begin{cases} -11\cdot x_1+5\cdot x_2+6\cdot x_3=19 \\ 5\cdot x_1+2\cdot x_2-3\cdot x_3=16 \\ -4\cdot x_1+4\cdot x_2-1\cdot x_3=14 \end{cases}$

Продолжение таблицы А.1

1	2	3	4
10	$\begin{cases} 2\cdot x_1+0\cdot x_2-1\cdot x_3=1 \\ 1\cdot x_1-4\cdot x_2+2\cdot x_3=-5 \\ 1\cdot x_1+1\cdot x_2+3\cdot x_3=6 \end{cases}$	25	$\begin{cases} -1\cdot x_1+15\cdot x_2-7\cdot x_3=21 \\ 6\cdot x_1-2\cdot x_2+7\cdot x_3=26 \\ 1\cdot x_1+4\cdot x_2-11\cdot x_3=34 \end{cases}$
11	$\begin{cases} 2\cdot x_1-1\cdot x_2-0\cdot x_3=3 \\ 0\cdot x_1+5\cdot x_2+2\cdot x_3=7 \\ 1\cdot x_1-1\cdot x_2+3\cdot x_3=4 \end{cases}$	26	$\begin{cases} 3\cdot x_1+6\cdot x_2+2\cdot x_3=9 \\ 3\cdot x_1+2\cdot x_2-4\cdot x_3=7 \\ -2\cdot x_1+4\cdot x_2+1\cdot x_3=5 \end{cases}$
12	$\begin{cases} 3\cdot x_1+1\cdot x_2-1\cdot x_3=-2 \\ 1\cdot x_1+5\cdot x_2-1\cdot x_3=8 \\ 2\cdot x_1-0\cdot x_2+3\cdot x_3=1 \end{cases}$	27	$\begin{cases} -7\cdot x_1+1\cdot x_2+4\cdot x_3=1 \\ 3\cdot x_1-9\cdot x_2+8\cdot x_3=3 \\ 5\cdot x_1-3\cdot x_2 2\cdot x_3=15 \end{cases}$
13	$\begin{cases} 3\cdot x_1+0\cdot x_2-1\cdot x_3=-4 \\ 2\cdot x_1-5\cdot x_2+1\cdot x_3=9 \\ 2\cdot x_1-2\cdot x_2+6\cdot x_3=8 \end{cases}$	28	$\begin{cases} 1\cdot x_1+12\cdot x_2-14\cdot x_3=25 \\ 1\cdot x_1+9\cdot x_2-18\cdot x_3=32 \\ 5\cdot x_1-13\cdot x_2-12\cdot x_3=19 \end{cases}$
14	$\begin{cases} 3\cdot x_1+0\cdot x_2-1\cdot x_3=7 \\ 2\cdot x_1-5\cdot x_2+1\cdot x_3=-2 \\ 2\cdot x_1-2\cdot x_2+5\cdot x_3=1 \end{cases}$	29	$\begin{cases} 3\cdot x_1-2\cdot x_2+7\cdot x_3=2 \\ -2\cdot x_1+2\cdot x_2-1\cdot x_3=7 \\ -1\cdot x_1+3\cdot x_2+7\cdot x_3=9 \end{cases}$
15	$\begin{cases} 3\cdot x_1-1\cdot x_2+1\cdot x_3=0 \\ 3\cdot x_1+5\cdot x_2+1\cdot x_3=12 \\ -1\cdot x_1-2\cdot x_2+4\cdot x_3=-1 \end{cases}$	30	$\begin{cases} 3\cdot x_1+2\cdot x_2+4\cdot x_3=12 \\ 2\cdot x_1-5\cdot x_2-0\cdot x_3=21 \\ 9\cdot x_1-2\cdot x_2-1\cdot x_3=9 \end{cases}$

Приложение Б – Исходные даные

Таблица Б.1 – Исходны даные для лабораторной работы №2

Вариант	Задание	Вариант	Задание
1	2	3	4
1	$\begin{cases}\sin(x-1)=1{,}3-y\\ x-\sin(y+1)=0{,}8\end{cases}$	16	$\begin{cases}\sin(x+1)-y=1\\ 2x+\cos y=2\end{cases}$
2	$\begin{cases}\sin y+2x=2\\ \cos(x-1)+y=0{,}7\end{cases}$	17	$\begin{cases}\sin(x+2)-y=1{,}5\\ x+\cos(y-2)=-0{,}5\end{cases}$
3	$\begin{cases}\cos(x-1)+y=0{,}5\\ x-\cos y=3\end{cases}$	18	$\begin{cases}\cos(x-1)+y=0{,}8\\ x-\cos y=2\end{cases}$
4	$\begin{cases}\cos y+x=1{,}5\\ 2y-\sin(x-0{,}5)=1\end{cases}$	19	$\begin{cases}\sin(y+1)-x=1{,}2\\ 2y+\cos x=2\end{cases}$
5	$\begin{cases}\sin(x+1)-y=1{,}2\\ 2x+\cos y=2\end{cases}$	20	$\begin{cases}\sin(x-1)=1{,}3+y\\ x+\sin(y+1)=0{,}8\end{cases}$
6	$\begin{cases}\sin(y+0{,}5)-x=1\\ \cos(x-2)+y=0\end{cases}$	21	$\begin{cases}\cos(y-1)+x=0{,}5\\ y-\cos x=3\end{cases}$
7	$\begin{cases}\sin x+2y=2\\ \cos(y-1)+x=0{,}72\end{cases}$	22	$\begin{cases}\cos(x+1)-y=0{,}5\\ x+\cos y=3\end{cases}$
8	$\begin{cases}\cos(y+0{,}5)+x=0{,}8\\ \sin x-2y=1{,}6\end{cases}$	23	$\begin{cases}\sin(x+1)+y=1{,}2\\ 2x-\cos y=2\end{cases}$
9	$\begin{cases}\cos x+y=1{,}5\\ 2x-\sin(y-0{,}5)=1\end{cases}$	24	$\begin{cases}x-\cos(y+1)=0\\ y+2\sin x=-0{,}4\end{cases}$
10	$\begin{cases}\sin(y-1)+x=1{,}3\\ y-\sin(x+1)=0{,}8\end{cases}$	25	$\begin{cases}\sin x-2y=2\\ \cos(y+1)+x=0{,}72\end{cases}$
11	$\begin{cases}\sin(x+0{,}5)-y=1\\ \cos(y-2)+x=0\end{cases}$	26	$\begin{cases}\cos(y-0{,}5)+x=2\\ \sin x+2y=1\end{cases}$

Продолжение таблицы Б.1

1	2	3	4
12	$\begin{cases} 2x - \cos(y+1) = 0 \\ y + \sin x = -0{,}4 \end{cases}$	27	$\begin{cases} \cos x + 2y = 1{,}5 \\ x - \sin(y - 0{,}5) = 1 \end{cases}$
13	$\begin{cases} 2y - \cos(x+1) = 0 \\ x + \sin y = -0{,}4 \end{cases}$	28	$\begin{cases} \sin(x+1) - 2y = 3 \\ x + \cos y = 2 \end{cases}$
14	$\begin{cases} \cos(y + 0{,}5) - x = 2 \\ \sin x - 2y = 1 \end{cases}$	29	$\begin{cases} \sin(x + 0{,}5) - y = 2 \\ \cos(y - 2) - x = 1 \end{cases}$
15	$\begin{cases} \cos(x + 0{,}5) - y = 2 \\ \sin y - 2x = 1 \end{cases}$	30	$\begin{cases} \cos(x - 1) + y = 0{,}8 \\ x + 4\cos y = 2 \end{cases}$

Приложение В – Исходные даные

Таблица В.1 – Исходны даные для лабораторной работы №3

Вариант 1		Вариант 2		Вариант 3		Вариант 4		Вариант 5		Вариант 6	
x	y	x	y	x	y	x	y	x	y	x	y
0,1	-5,40	0,98	2,22	0,46	-0,19	0,37	-0,33	0,50	-0,10	0,71	0,95
0,3	-2,00	1,27	2,68	0,75	1,64	0,62	0,40	0,75	1,89	0,92	2,51
0,5	0,01	1,48	3,76	0,95	1,98	0,90	2,17	0,97	2,52	1,14	2,47
0,7	1,33	1,73	3,50	1,19	3,30	1,11	2,89	1,17	3,06	1,35	3,47
0,9	1,99	2,03	4,38	1,46	3,52	1,35	2,73	1,45	3,27	1,56	3,95
1,1	2,25	2,28	4,99	1,69	3,76	1,58	3,98	1,67	3,80	1,76	3,64
1,3	2,46	2,56	5,52	1,96	4,36	1,79	4,17	1,94	3,88	2,03	4,02
1,5	3,16	2,78	5,82	2,25	5,06	2,09	4,91	2,14	4,14	2,23	5,11
1,7	3,37	2,98	5,12	2,45	4,57	2,34	4,95	2,37	4,92	2,45	5,27
1,9	4,35	3,21	6,29	2,74	4,85	2,63	5,15	2,66	5,19	2,70	4,92
Вариант 7		Вариант 8		Вариант 9		Вариант 10		Вариант 11		Вариант 12	
x	y	x	y	x	y	x	y	x	y	x	y
0,62	1,30	0,32	0,83	0,37	1,45	0,47	1,22	0,43	0,89	0,62	1,36
0,92	2,08	0,56	1,80	0,63	1,52	0,76	1,69	0,67	1,87	0,87	1,77
1,20	2,73	0,86	2,32	0,91	2,16	0,96	1,84	0,91	1,92	1,15	2,50
1,43	2,61	1,06	2,31	1,21	3,02	1,17	2,80	1,17	2,27	1,41	3,04
1,69	3,11	1,29	2,36	1,49	2,82	1,38	3,00	1,43	2,92	1,64	3,20
1,95	3,56	1,54	2,58	1,74	3,50	1,66	3,13	1,71	3,12	1,91	3,22
2,25	3,39	1,82	3,02	2,01	3,66	1,93	3,30	1,94	3,47	2,18	3,79
2,49	3,47	2,03	3,21	2,25	3,77	2,20	3,33	2,22	3,62	2,45	3,83
2,76	3,29	2,29	3,26	2,54	3,39	2,48	3,19	2,48	3,25	2,71	3,66
2,97	3,72	2,58	3,93	2,83	4,00	2,74	3,36	2,74	3,73	2,96	3,88
Вариант 13		Вариант 14		Вариант 15		Вариант 16		Вариант 17		Вариант 18	
x	y	x	y	x	y	x	y	x	y	x	y
0,33	3,28	0,37	3,74	0,40	3,41	1,11	8,69	1,37	8,65	1,46	9,32
0,61	3,75	0,64	4,71	0,62	3,42	1,37	9,35	1,58	12,79	1,74	16,06
0,90	4,55	0,87	5,21	0,86	4,96	1,66	13,69	1,83	14,51	2,02	12,00
1,16	6,22	1,16	4,41	1,09	4,54	1,93	13,73	2,11	15,08	2,26	14,12
1,37	5,46	1,40	7,10	1,39	5,51	2,22	22,08	2,33	17,32	2,49	21,79
1,59	7,00	1,68	8,41	1,60	7,37	2,43	17,73	2,60	21,02	2,71	22,10
1,83	6,07	1,97	8,43	1,87	8,77	2,64	20,71	2,89	22,21	2,92	30,57
2,10	7,97	2,24	8,48	2,15	9,17	2,86	38,74	3,18	34,62	3,19	34,05
2,37	15,59	2,45	11,66	2,44	14,54	3,10	47,12	3,47	36,28	3,48	57,22
2,65	18,23	2,71	19,31	2,65	14,86	3,37	49,43	3,74	54,69	3,77	81,53

Продолжение таблицы В.1

Вариант 19		Вариант 20		Вариант 21		Вариант 22		Вариант 23		Вариант 24	
x	y	x	y	x	y	x	y	x	y	x	y
1,19	2,94	1,22	2,80	1,56	2,84	1,50	5,12	2,17	4,38	2,17	0,29
1,48	2,27	1,43	2,36	1,82	1,63	1,72	4,10	2,43	3,60	2,43	-0,56
1,77	1,89	1,65	1,63	2,05	1,08	1,93	3,76	2,68	3,53	2,68	-0,68
2,02	1,78	1,95	1,30	2,33	1,07	2,20	3,42	2,89	3,60	2,94	-0,69
2,29	1,25	2,21	1,08	2,60	1,02	2,46	3,11	3,18	3,67	3,15	-0,60
2,59	1,25	2,44	1,15	2,86	1,02	2,75	3,11	3,39	3,83	3,43	-0,49
2,85	1,30	2,67	1,05	3,15	1,19	2,95	3,22	3,61	4,05	3,65	-0,26
3,12	1,37	2,96	1,09	3,44	1,18	3,18	3,13	3,87	3,99	3,88	0,05
3,37	1,55	3,23	1,07	3,70	1,42	3,46	3,73	4,07	3,89	4,14	0,18
3,57	1,37	3,50	1,71	3,97	2,15	3,74	3,83	4,36	4,48	4,38	0,09
Вариант 25		Вариант 26		Вариант 27		Вариант 28		Вариант 29		Вариант 30	
x	y	x	y	x	y	x	y	x	y	x	y
0,58	0,55	0,55	0,22	0,21	0,27	0,03	0,32	-0,15	0,27	0,02	0,21
0,95	0,08	0,94	-0,23	0,58	0,03	0,38	-0,22	0,20	0,47	0,35	0,23
1,28	0,38	1,31	-0,10	0,92	-0,10	0,70	-0,18	0,55	0,20	0,69	0,17
1,61	0,27	1,69	0,20	1,26	-0,08	1,04	-0,06	0,95	0,61	1,01	0,46
1,98	0,74	2,07	1,30	1,61	0,85	1,42	0,31	1,26	0,49	1,34	0,17
2,29	1,04	2,41	1,70	2,00	1,01	1,81	0,98	1,63	0,48	1,69	0,11
2,61	1,43	2,73	2,15	2,33	1,59	2,12	0,63	1,93	0,35	2,00	1,17
2,97	2,81	3,04	2,66	2,63	1,49	2,42	1,55	2,29	1,43	2,36	1,52
3,33	3,18	3,39	3,05	2,94	2,29	2,82	2,44	2,62	1,81	2,74	1,69
3,65	3,84	3,73	4,04	3,32	3,76	3,19	3,45	2,94	2,09	3,07	2,47

Приложение Г – Исходные даные

Таблица Г.1 – Исходны даные для лабораторной работы №4

Вариант	Задание	Вариант	Задание
1	2	3	4
1	$\int_{-2}^{4}(2x^2 - \sqrt{x+3})dx$, $\int(x^2 - 1{,}5\sqrt{x})dx$	16	$\int_{1}^{4}(x^2 - 1{,}5\sqrt{x})dx$, $\int(2x^2 - \sqrt{x+3})dx$
2	$\int_{-3}^{0}(5x^2 + x + 1)dx$, $\int(7\sqrt{x} + 2x^2)dx$	17	$\int_{1}^{4}(7\sqrt{x} + 2x^2)dx$, $\int(5x^2 + x + 1)dx$
3	$\int_{1}^{3}(3x^2 - \sqrt{x})dx$, $\int(7x^2 - 3\sqrt{x+1})dx$	18	$\int_{0}^{3}(7x^2 - 3\sqrt{x+1})dx$, $\int(3x^2 - \sqrt{x})dx$
4	$\int_{1}^{4}(x^3 - \sqrt{x})dx$, $\int(2x^2 - 2 + \sqrt{x})dx$	19	$\int_{2}^{5}(2x^2 - 2 + \sqrt{x})dx$, $\int(x^3 - \sqrt{x})dx$
5	$\int_{1}^{4}(7 + x - 2x^2)dx$, $\int(5x^2 - 1 + \sqrt{x+1})dx$	20	$\int_{0}^{3}(5x^2 - 1 + \sqrt{x+1})dx$, $\int(7 + x - 2x^2)dx$
6	$\int_{1}^{3}(7x^2 - 3\sqrt{x})dx$, $\int(x^2 + 4 + \sqrt{x})dx$	21	$\int_{3}^{6}(x^2 + 4 + \sqrt{x})dx$, $\int(7x^2 - 3\sqrt{x})dx$
7	$\int_{2}^{5}(2x^2 - 2 - \sqrt{x})dx$, $\int(x^2 + 4 + \sqrt{x})dx$	22	$\int_{3}^{6}(x^2 + 4 + \sqrt{x})dx$, $\int(2x^2 - 2 - \sqrt{x})dx$
8	$\int_{1}^{3}(5x^2 + \sqrt{x})dx$, $\int(2x^2 - 1 + \sqrt{x+1})dx$	23	$\int_{0}^{3}(2x^2 - 1 + \sqrt{x+1})dx$, $\int(5x^2 + \sqrt{x})dx$

Продолжение таблицы Г.1

1	2	3	4
9	$\int\limits_{-2}^{2}(x^3+1)dx$, $\int(3x^2+2\sqrt{x+3})dx$	24	$\int\limits_{-2}^{2}(3x^2+2\sqrt{x+3})dx$, $\int(x^3+1)dx$
10	$\int\limits_{1}^{4}(2x^2+1-\sqrt{x})dx$, $\int(x^2+2\sqrt{x+3})dx$	25	$\int\limits_{-2}^{2}(x^2+2\sqrt{x+3})dx$, $\int(2x^2+1-\sqrt{x})dx$
11	$\int\limits_{-2}^{2}(x^2+\sqrt{x+3}-1)dx$, $\int(x^2+2x-1{,}5)dx$	26	$\int\limits_{-3}^{1}(x^2+2x-1{,}5)dx$, $\int(x^2+\sqrt{x+3}-1)dx$
12	$\int\limits_{0}^{2}(x^2+2+\sqrt{x+1})dx$, $\int(3x^2+1+\sqrt{x+4})dx$	27	$\int\limits_{-3}^{0}(3x^2+1+\sqrt{x+4})dx$, $\int(x^2+2+\sqrt{x+1})dx$
13	$\int\limits_{1}^{5}(3x^2-x-1)dx$, $\int(3x^2+5+\sqrt{x+1})dx$	28	$\int\limits_{0}^{3}(3x^2+5+\sqrt{x+1})dx$, $\int(3x^2-x-1)dx$
14	$\int\limits_{-1}^{3}(x^3+2)dx$, $\int(7x+x^2-\sqrt{x})dx$	29	$\int\limits_{1}^{4}(7x+x^2-\sqrt{x})dx$, $\int(x^3+2)dx$
15	$\int\limits_{-2}^{2}(2x^2+1-\sqrt{x+4})dx$, $\int(x^2-3\sqrt{x+1})dx$	30	$\int\limits_{0}^{3}(x^2-3\sqrt{x+1})dx$, $\int(2x^2+1-\sqrt{x+4})dx$

Приложение Д – Исходные даные

Таблица Д.1 – Исходны даные для лабораторной работы №5

Вариант	Задание	Вариант	Задание
1	2	3	4
1	$y'=3+2\cdot x^2$, $y(0)=2$, $x\in[0;1]$, $h=0{,}02$	16	$y'=1-x+y$, $y(0)=1$, $x\in[0;2{,}5]$, $h=0{,}05$
2	$y'=y-x^2$, $y(1)=0$, $x\in[1;2{,}2]$, $h=0{,}03$	17	$y'=y^2-5x$, $y(-1)=1$, $x\in[-1;1]$, $h=0{,}04$
3	$y'=1-x^2+y$, $y(1{,}1)=0$, $x\in[1{,}1;1{,}6]$, $h=0{,}01$	18	$y'=x+2\cdot y$, $y(0)=-1$, $x\in[0;2]$, $h=0{,}04$
4	$y'=y-7\cdot x$, $y(3)=3$, $x\in[3;5]$, $h=0{,}05$	19	$y'=x+y+2$, $y(1)=1$, $x\in[1;3]$, $h=0{,}05$
5	$y'=5-y+x^2$, $y(1)=1$, $x\in[1;5]$, $h=0{,}1$	20	$y'=3\cdot x+4\cdot y$, $y(2)=1$, $x\in[2;5]$, $h=0{,}05$
6	$y'=y-2\cdot x^2+3$, $y(0)=4$, $x\in[0;1]$, $h=0{,}02$	21	$y'=3+2\cdot x+y$, $y(0)=2$, $x\in[0;1]$, $h=0{,}02$
7	$y'=4-x^2+2\cdot y$, $y(0)=1$, $x\in[0;1{,}2]$, $h=0{,}03$	22	$y'=2\cdot y-x^2$, $y(1)=0$, $x\in[1;2{,}2]$, $h=0{,}03$
8	$y'=-8+2\cdot x-y$, $y(1)=3$, $x\in[1;3]$, $h=0{,}04$	23	$y'=-x^2+y$, $y(1{,}1)=0$, $x\in[1{,}1;1{,}6]$, $h=0{,}01$
9	$y'=2\cdot y-3\cdot x^2$, $y(4)=0$, $x\in[4{,}6]$, $h=0{,}05$	24	$y'=y-7\cdot x+2$, $y(3)=3$, $x\in[3;5]$, $h=0{,}05$
10	$y'=x^2-2\cdot y$, $y(-1)=1$, $x\in[-1;2]$, $h=0{,}06$	25	$y'=5-y+x^2$, $y(1)=1$, $x\in[1;5]$, $h=0{,}1$
11	$y'=7-x\cdot y$, $y(-2)=0$, $x\in[-2;0]$, $h=0{,}05$	26	$y'=y-2\cdot x+3$, $y(0)=4$, $x\in[0;1]$, $h=0{,}02$
12	$y'=2\cdot x^2+y$, $y(2)=2$, $x\in[2;3{,}5]$, $h=0{,}05$	27	$y'=4-x\cdot 2+2\cdot y$, $y(0)=1$, $x\in[0;1{,}2]$, $h=0{,}03$
13	$y'=5+x-y$, $y(2)=1$, $x\in[2;4]$, $h=0{,}05$	28	$y'=-8+2\cdot x-y$, $y(1)=3$, $x\in[1;3]$, $h=0{,}04$
14	$y'=y+5\cdot x-1$, $y(0)=2$, $x\in[0;3{,}2]$, $h=0{,}08$	29	$y'=2\cdot y-3\cdot x^2$, $y(4)=0$, $x\in[4{,}6]$, $h=0{,}05$
15	$y'=y-5\cdot x+1$, $y(0)=2$, $x\in[0;3{,}2]$, $h=0{,}08$	30	$y'=x^2-2\cdot y$, $y(-1)=1$, $x\in[-1;2]$, $h=0{,}06$

Приложение Е – Исходные даные

Таблица Е.1 – Исходны даные для лабораторной работы №6

Вариант	Задание	Вариант	Задание
1	2	3	4
1	$\frac{d}{dt}x(t)=x(t)-y(t)$ $x(0)=1$ $\frac{d}{dt}y(t)=-4\cdot x(t)+y(t)$ $y(0)=2$	16	$\frac{d}{dt}x(t)=x(t)-2\cdot y(t)$ $x(0)=1$ $\frac{d}{dt}y(t)=3\cdot x(t)-2\cdot y(t)$ $y(0)=2$
2	$\frac{d}{dt}x(t)=x(t)^2-y(t)$ $x(0)=1$ $\frac{d}{dt}y(t)=2\cdot x(t)-y(t)$ $y(0)=1$	17	$\frac{d}{dt}x(t)=y(t)$ $x(0)=1$ $\frac{d}{dt}y(t)=-x(t)-cos(y(t))$ $y(0)=2$
3	$\frac{d}{dt}x(t)=2\cdot y(t)-5\cdot x(t)$ $x(0)=1$ $\frac{d}{dt}y(t)=4\cdot x(t)-y(t)$ $y(0)=0.5$	18	$\frac{d}{dt}x(t)=y(t)$ $x(0)=2$ $\frac{d}{dt}y(t)=2\cdot\left(1-x(t)^2\right)\cdot y(t)-x(t)$ $y(0)=0$
4	$\frac{d}{dt}x(t)=y(t)+sin(t)$ $x(0)=1$ $\frac{d}{dt}y(t)=1\cdot\left(1-x(t)^2\right)\cdot y(t)-2\cdot x(t)$ $y(0)=1$	19	$\frac{d}{dt}x(t)=x(t)+4\cdot y(t)$ $x(0)=1$ $\frac{d}{dt}y(t)=-2\cdot x(t)-y(t)$ $y(0)=2$

Продолжение таблицы Е.1

1	2	3	4
5	$\frac{d}{dt}x(t)=-x(t)+2\cdot y(t)\cdot t$ $x(0)=1$ $\frac{d}{dt}y(t)=-2\cdot x(t)-y(t)$ $y(0)=-1$	20	$\frac{d}{dt}x(t)=-x(t)\cdot\sqrt{t}+y(t)$ $x(1)=1$ $\frac{d}{dt}y(t)=-2\cdot t\cdot x(t)-y(t)$ $y(1)=2$
6	$\frac{d}{dt}x(t)=-x(t)+\frac{y(t)}{t}$ $x(1)=2$ $\frac{d}{dt}y(t)=ln(t)+x(t)-y(t)$ $y(1)=1$	21	$\frac{d}{dt}x(t)=-x(t)+4\cdot y(t)+\sqrt{t}$ $x(1)=1$ $\frac{d}{dt}y(t)=sin(t)\cdot x(t)-8\cdot y(t)$ $y(1)=2$
7	$\frac{d}{dt}x(t)=-x(t)+4\cdot y(t)$ $x(0)=1$ $\frac{d}{dt}y(t)=sin(t)\cdot x(t)-cos(t)\cdot y(t)$ $y(0)=1$	22	$\frac{d}{dt}x(t)=-x(t)+sin(4\cdot t)\cdot y(t)$ $x(0)=2$ $\frac{d}{dt}y(t)=x(t)-cos(2\cdot t)\cdot y(t)$ $y(0)=-1$
8	$\frac{d}{dt}x(t)=\left(t\cdot x(t)-e^{t}\right)\cdot y(t)$ $x(0)=2$ $\frac{d}{dt}y(t)=4\cdot x(t)-0.5\cdot y(t)$ $y(0)=4$	23	$\frac{d}{dt}x(t)=y(t)-e^{1-t}\cdot x(t)$ $x(0)=2$ $\frac{d}{dt}y(t)=sin(x(t))-0.5\cdot y(t)$ $y(0)=1$
9	$\frac{d}{dt}x(t)=y(t)-\sqrt{t-1}\cdot x(t)$ $x(2)=2$ $\frac{d}{dt}y(t)=sin(x(t))-y(t)$ $y(2)=1$	24	$\frac{d}{dt}x(t)=y(t)-sin(x(t))$ $x(0)=0$ $\frac{d}{dt}y(t)=-cos(x(t))-y(t)$ $y(0)=1$

Продолжение таблицы Е.1

1	2	3	4
10	$\frac{d}{dt}x(t)=-x(t)+2^{y(t)}$ $x(0)=0$ $\frac{d}{dt}y(t)=x(t)-t\cdot y(t)$ $y(0)=1$	25	$\frac{d}{dt}x(t)=-x(t)+9^{-y(t)}$ $x(1)=2$ $\frac{d}{dt}y(t)=x(t)-ln(t)\cdot y(t)$ $y(1)=1$
11	$\frac{d}{dt}x(t)=-cos(x(t))+\frac{y(t)}{5\cdot t}$ $x(1)=1$ $\frac{d}{dt}y(t)=x(t)-y(t)$ $y(1)=3$	26	$\frac{d}{dt}x(t)=0.5\cdot x(t)-0.02\cdot y(t)\cdot x(t)$ $x(0)=6$ $\frac{d}{dt}y(t)=-0.5\cdot y(t)+0.02\cdot y(t)\cdot x(t)$ $y(0)=3$
12	$\frac{d}{dt}x(t)=cot(x(t))-y(t)$ $x(0)=1$ $\frac{d}{dt}y(t)=t\cdot x(t)-y(t)$ $y(0)=-1$	27	$\frac{d}{dt}x(t)=x(t)-y(t)$ $x(1)=1$ $\frac{d}{dt}y(t)=e^{t}+x(t)+y(t)$ $y(1)=2$
13	$\frac{d}{dt}x(t)=0.25^{t}+\frac{x(t)}{y(t)}$ $x(1)=1$ $\frac{d}{dt}y(t)=x(t)-t\cdot y(t)$ $y(1)=3$	28	$\frac{d}{dt}x(t)=0.5^{x(t)}+8\cdot sin(y(t))$ $x(0)=1$ $\frac{d}{dt}y(t)=x(t)-y(t)$ $y(0)=3$
14	$\frac{d}{dt}x(t)=0.5^{x(t)}-2\ (y(t))$ $x(1)=1$ $\frac{d}{dt}y(t)=0.8\cdot x(t)-y(t)-ln(t)$ $y(1)=3$	29	$\frac{d}{dt}x(t)=sin\left(\frac{x(t)}{2}\right)-(y(t))$ $x(0)=0.1$ $\frac{d}{dt}y(t)=6\cdot x(t)-2.5\cdot y(t)$ $y(0)=-2$

Продолжение таблицы Г.1

1	2	3	4
15	$\frac{d}{dt}x(t)=\cos\left(\frac{x(t)}{t}\right)-y(t)$ $x(1)=0$ $\frac{d}{dt}y(t)=x(t)-y(t)$ $y(1)=1$	30	$\frac{d}{dt}x(t)=\frac{x(t)}{t}-\sin(y(t))$ $x(1)=0$ $\frac{d}{dt}y(t)=t\cdot x(t)-y(t)$ $y(1)=1$

Приложение Ж – Исходные даные

Таблица Ж.1 – Исходны даные для лабораторных работ №7 и №8

Вариант	Задание					
	Поток 1			Поток 2		
	G_1, кг/ч	C_{p1}, Дж/кг·К	t_1, °С	G_2, кг/ч	C_{p2}, Дж/кг·К	t_2, °С
1	1000	3505	80	300	3171	20
2	800	3964	100	100	2693	10
3	950	3309	80	200	2994	20
4	880	3057	90	250	2515	25
5	820	3892	88	300	2952	30
6	870	3731	82	280	3088	28
7	980	3552	63	480	2915	18
8	800	3617	90	550	2639	22
9	850	3662	92	600	2642	23
10	860	3029	50	1100	2434	12
11	900	3474	60	1200	2751	21
12	920	3306	65	1100	2695	13
13	960	3016	64	1200	2645	18
14	930	3540	72	1000	3098	22
15	990	3505	82	800	3171	23
16	820	3925	88	300	2291	30
17	850	3291	92	600	2721	23
18	1000	3712	80	300	3125	20
19	850	3794	92	600	2332	23
20	920	3669	65	1100	3066	13
21	800	3471	100	100	2608	10
22	900	3290	60	1200	2432	21
23	980	3943	63	480	2603	18
24	950	3214	80	200	2248	20
25	800	3835	90	550	3081	22
26	980	3957	63	480	2666	18
27	880	3757	90	250	2722	25
28	840	3454	75	175	2955	33
29	910	3591	82	225	2523	25
30	820	3925	86	190	2291	16

Учебное издание

КОМПЬЮТЕРНОЕ МОДЕЛИРОВАНИЕ ТЕХНОЛОГИЧЕСКИХ ПРОЦЕССОВ И ОБОРУДОВАНИЯ ПИЩЕВОЙ ПРОМЫШЛЕННОСТИ

Лабораторный практикум

Составитель: Колюкович Евгений Александрович

Редактор А. А. Щербакова
Технический редактор М.О. Хлыстова

Подписано в печать Формат 60х84 1/16
Бумага офсетная. Гарнитура Таймс. Печать трафаретная.
Усл. печ. л. Уч.-изд. л.
Тираж Заказ

Отпечатано на ризографе редакционно-издательского отдела
учреждения образования
«Могилевский государственный университет продовольствия».
212027, Могилев, пр-т Шмидта, 3.
ЛИ № 02330/0131913 от 08.02.2007.