



Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования

«Уфимский государственный нефтяной технический университет»

Филиал ФГБОУ ВО УГНТУ в г. Салавате

Кафедра «Электрооборудование и автоматика промышленных предприятий»

КОМПЬЮТЕРНЫЕ ТЕХНОЛОГИИ

Учебно-методическое пособие
к выполнению практических, лабораторных
и самостоятельных работ

Уфа
2018

Учебно-методическое пособие предназначено для студентов-бакалавров направления подготовки 13.03.02 «Электроэнергетика и электротехника», профиль «Электрооборудование и электрохозяйство предприятий, организаций и учреждений».

Материал содержит задания для выполнения самостоятельных, практических и лабораторных работ студентами заочной и очной форм обучения, по дисциплине «Компьютерные технологии». Учебно-методическое пособие способствует приобретению студентам навыков самостоятельной работы с научной литературой.

Публикуется в авторской редакции.

Составитель: Сафин Э.М., к.ф.-м.н., доцент каф. ЭАПП

Рецензенты: Баширов М.Г., д.-р техн. наук проф. каф. ЭАПП
Хисматуллин А.С., к.ф.-м. наук, доцент. каф. ЭАПП

СОДЕРЖАНИЕ

ВВЕДЕНИЕ	4
1 Самостоятельная работа обучающихся	5
1.1 Задания для самостоятельной работы	5
1.2 Вопросы для письменного и устного опроса	5
1.3 Методические указания по ведению конспекта лекций.....	7
2 Задания для выполнения практических работ.....	9
2.1 Практическая работа № 1. Знакомство с MATLAB. Основные объекты языка MATLAB.	9
2.2 Практическая работа № 2. Операции с матрицами.	14
2.3 Практическая работа № 3. Типы массивов.....	18
2.4 Практическая работа № 4. Средства графики.	21
3 Задания для выполнения лабораторных работ.....	26
3.1 Лабораторная работа № 1 Дифференциальные уравнения и математическое моделирование.....	26
3.2 Лабораторная работа № 2 Основы моделирования электрических цепей в MATLAB/SIMULINK (scilab xcos).....	40
3.3 Лабораторная работа № 3 Представление данных. Вычисление статистических параметров.	57
3.4 Лабораторная работа № 4 Автоматизация выполнения расчетов при обработке результатов испытаний.....	62
4 Вопросы к зачету по курсу «Компьютерные технологии».....	65
СПИСОК РЕКОМЕНДУЕМОЙ ДЛЯ ИЗУЧЕНИЯ ЛИТЕРАТУРЫ.....	69
РЕКОМЕНДУЕМЫЕ ИНФОРМАЦИОННЫЕ СПРАВОЧНЫЕ СИСТЕМЫ	71

ВВЕДЕНИЕ

«Компьютерные технологии», является вариативной дисциплиной подготовки студентов по направлению 13.03.02 «Электроэнергетика и электротехника». Изучение курса формирует у студента способности: применять соответствующий физико-математический аппарат, методы анализа и моделирования, теоретического и экспериментального исследования при решении профессиональных задач; использовать методы анализа и моделирования электрических цепей; обрабатывать результаты экспериментов.

Задания для практических и лабораторных работ построены на следующих источниках [1–4]. Все виды работ проводятся в программе MATLAB, а в случае необходимости в её мощнейшем аналоге Scilab (имеющем лицензию CeCILL), это особенно легко сделать, поскольку утилита входящая в пакет последней, позволяет конвертировать документы MATLAB в Scilab.

1 Самостоятельная работа обучающихся

1.1 Задания для самостоятельной работы

Самостоятельная работа студентов заключается в подготовке к практическим и лабораторным занятиям по нижеприведенным темам. Так же необходимо выполнить все практические и лабораторные работы из методического пособия в программе Scilab.

1.2 Вопросы для письменного и устного опроса

1. Для чего предназначена система MATLAB?
2. Назовите окна интерфейса MATLAB и поясните их назначение.
3. Дайте определение следующим понятиям: текущая сессия, режим прямых вычислений.
4. Назовите базовые объекты языка MATLAB.
5. Дайте определение команды.
6. Дайте определение константы.
7. Какие типы констант используются в MATLAB?
8. Как вводятся комплексные константы?
9. Какие форматы вывода констант используются в MATLAB?
10. Дайте определение формы E и нормализованной формы E.
11. Какие константы называют стандартными?
12. Дайте определение переменной и поясните, с помощью какого оператора ей присваивается значение.
13. Дайте определение массива.
14. Чем характеризуется массив?
15. Дайте расшифровку названия " MATLAB " и поясните его смысл.
16. Как вектор и скаляр воспринимаются в MATLAB?
17. Чему равна нижняя граница индексов матрицы в MATLAB?
18. Как вводятся матрица, вектор и скаляр?

19. Чему соответствует простая переменная в MATLAB?
20. Дайте определение выражения в MATLAB.
21. Какие типы выражений используются в MATLAB?
22. Дайте определение арифметического и логического выражений.
23. В какое графическое окно выводится график по умолчанию?
24. Как вывести график в новое графическое окно?
25. Как вывести несколько графиков на одних координатных осях?
26. Как удалить графики перед выводом нового графика в то же графическое окно?
27. Как вывести несколько независимых графиков в одном графическом окне с его разбиением на отдельные поля?
28. Какие средства оформления графиков используются в MATLAB?
29. Какие средства предусмотрены для установки типа, цвета и толщины линий?
30. Какие средства предусмотрены для установки вида, размера и цвета маркеров?
31. Какая функция используется для построения двумерных графиков в линейном масштабе с линейной интерполяцией между соседними значениями?
32. Какая функция используется для построения графика последовательности чисел?
33. Какие функции используются для построения графиков в полулогарифмическом и логарифмическом масштабах?
34. Какая функция используется для построения гистограмм?
35. В чем заключается подготовка перед построением трехмерного графика?
36. Какие функции используются для построения трехмерных графиков?
37. Как выбрать палитру цветов при построении трехмерного графика?
38. Как вывести шкалу цветов на поле трехмерного графика?
39. Для чего предназначен режим программирования?
40. Что такое М-файл?
41. Какие разновидности М-файлов создаются в режиме программирования?
42. Как вывести листинг М-файла?
43. Что такое script-файл и как к нему обратиться в режиме прямых вычислений?
44. Что такое function-файл и как к нему обратиться в режиме прямых вычислений и в script-файле?

45. Каков формат описания function-файла?
46. Какие переменные function-файла называют формальными и фактическими?
47. Какие переменные сохраняются в Workspace после выполнения script-файла?
48. Какие переменные сохраняются в Workspace после выполнения function-файла?
49. Какие переменные называют локальными и глобальными?
50. В каком окне создаются script- и function-файлы?
51. Как организовать ввод данных с клавиатуры в режиме программирования?
52. Как организовать вывод данных в окно **Command Window** в режиме программирования?
53. Как вывести на одной строке значение численной переменной одновременно с текстом?
54. В каких случаях целесообразно предусмотреть паузу?
55. Как сохранить М-файл в требуемой папке?
56. Как сохранить путь к данной папке?
57. Поясните назначение и формат оператора if.
58. Поясните назначение и формат оператора switch.
59. Поясните назначение и формат оператора for.
60. Поясните назначение и формат оператора while.
61. Как выполнить принудительный выход из цикла? Какой части программы передается управление в этом случае?

1.3 Методические указания по ведению конспекта лекций

Для ведения конспекта лекций желательно использовать тетрадь большого формата. Поскольку для преподавания дисциплины используется презентация, которая в электронном виде предоставляется студентам, графическую часть лекции целесообразно располагать в соответствующих местах текста (например, вклеивать в конспект). Если какая-то лекция пропускается, лучше сразу переписать ее или, по крайней мере, оставить для нее чистые листы. Хороший конспект может в дальнейшем помочь при изучении других предметов, а также в процессе выполнения выпускной квалификационной работы и в практической деятельности

после окончания университета. В тетради следует отвести большие поля (примерно $\frac{1}{3}$ ширины листа). Их используют при изучении материала лекции, которую дополняют сведениями из рекомендованной литературы, решением задач и ответами на вопросы, которые преподаватель задает во время лекции. Не оставляйте без внимания непонятные места, найдите ответы сами в литературе или проконсультируйтесь с преподавателем и дополните свой конспект. Перед каждой лекцией необходимо проработать предыдущую, это будет помогать освоить новый материал и закрепить прошлые знания. Очень полезно рассказывать материал заинтересованному слушателю (товарищу по учебе), который не только внимательно выслушает, но и задаст массу вопросов. Такой подход к изучению предмета трудно переоценить, он позволяет сформировать устойчивые знания и развить логическое мышление.

Записи по практическим занятиям целесообразно продолжать делать в лекционной тетради.

2 Задания для выполнения практических работ

2.1 Практическая работа № 1. Знакомство с MATLAB. Основные объекты языка MATLAB.

Цель занятия: познакомиться с назначением и интерфейсом системы MATLAB и овладеть начальными навыками работы в режиме прямых вычислений.

Задание на практическую работу включает в себя следующие пункты:

Содержание работы связано с изучением режима прямых вычислений и базовых объектов языка MATLAB.

Запуск системы MATLAB и знакомство с ее интерфейсом. Пояснить, какие окна образуют интерфейс MATLAB.

Знакомство со справочной системой MATLAB в формате HTML. Пояснить, как обратиться к справочной системе.

Ввод комментария в окне Command Window. Ввести наименование практической работы. Пояснить, какой символ используется для ввода комментария.

Знакомство с командами языка MATLAB.

Выполнить команду:

- help general.

Пояснить назначение и формат команды help.

О каких объектах языка MATLAB будет выдана справка?

Очистка окна Command Window.

Пояснить, какая команда используется.

Ввод вещественных констант.

Ввести следующие константы в обычной форме и форме E без символа в конце строки:

0

0,000

0,814

–0,814

$8,14 \cdot 10^{-7}$

0,814578942

0,9999999999

0,0000814765178

8145,7

-8145,577777777

$0,814557 \cdot 10^5$

Пояснить:

- смысл символа «;» в конце строки;
- какой переменной присваиваются значения вводимых констант;
- в каком случае при вводе констант целесообразно использовать форму E;
- в каком формате выводятся константы по умолчанию;
- как вывести указанные константы с максимальным количеством значащих цифр в дробной части;
- какое количество значащих цифр в дробной части будет максимальным;
- какие форматы предусмотрены для вывода вещественных констант;
- какую форму E называют нормализованной.

Ввод комплексных констант.

Ввести следующие константы без символа " в конце строки:

0,057+0,5j

0,057+0,5i

1200000,5+561

1200000,57857+56i

12,5+561

12,5+0,0000561

-0,99999991

0i

$17+10^{-5}i$

$15 \cdot 10^{-5}i$

Пояснить:

- в какой форме вводятся комплексные константы;
- в какой форме вводятся их вещественные и мнимые части;
- в каком формате выводятся комплексные константы по умолчанию;
- какой формат целесообразно выбрать для вывода указанных констант;
- какая из констант списка будет воспринята как вещественная.

Ввод логических констант.

Ввести константы true и false без символа "; " в конце строки.

Пояснить, какие значения будут выведены и какой переменной присвоены.

Ввод символьных констант.

Ввести константы:

- ФИО;
- наименование лабораторной работы.

Пояснить, как вводятся и выводятся символьные константы.

Ввод векторов.

Ввести векторы – строки и столбцы – со следующими элементами:

–0,9; 125; 0; 5+3i; 12i;

–0,9; 125; 0; 5; 12;

1; 2; 4; 5; 12.

Пояснить:

- какие символы используются при вводе векторов;
- как в MATLAB воспринимаются скаляры и векторы.

Ввод матрицы.

Ввести матрицы 3x3 и 3x2 с произвольными элементами.

Пояснить, что называют размером и порядком матрицы.

Ввод переменных.

Присвоить произвольные значения простой переменной, вектору и матрице.

Пояснить, как выбираются имена переменных и как переменные воспринимаются в MATLAB.

Знакомство с особенностями ввода комплексных переменных.

Присвоить переменной i значение 5.

Присвоить переменной f значение комплексной константы $5+3i$, которую ввести двумя способами: без символа умножения в мнимой части; с символом умножения.

Пояснить:

- в каком из этих случаев и почему возникает ошибка;
- как предотвратить возникновение ошибок в подобных случаях.

Знакомство со стандартными функциями с комплексным аргументом.

Присвоить переменной произвольное комплексное значение.

Вычислить модуль, аргумент, вещественную и мнимую части переменной.

Присвоить другой переменной значение комплексно сопряженной константы.

Пояснить, какие стандартные функции для этого используются.

Ввод арифметических выражений.

Присвоить переменным a, b и c значения произвольных вещественных констант, не равных нулю.

Вычислить значения переменных d и e по следующим формулам:

$$d = a + b \sin(\pi/a + b/c - \cos \pi);$$

$$e = a^2 + \sqrt{|b|} + \sqrt[3]{c} + \frac{d+ac}{b}.$$

Пояснить:

- приоритет выполнения операций в арифметических выражениях;
- что является результатом вычисления арифметического выражения.

Ввод логических выражений.

Используя переменные предыдущего пункта, записать логическое выражение с использованием операций отношения и вычислить его значение.

Добавить в данное выражение логические операции и вычислить значение нового логического выражения.

Пояснить:

- приоритет выполнения операций в логических выражениях;
- что является результатом вычисления логического выражения.

Знакомство со стандартными переменными.

Ввести арифметические выражения, которым по умолчанию будут присвоены константы Nan и Inf.

Пояснить назначение данных констант.

Знакомство со стандартными функциями округления.

Выполнить следующие вычисления:

`floor([8.2 8.5 8.7 -8.2 -8.5 -8.7])`

`ceil ([8.2 8.5 8.7 -8.2 -8.5 -8.7])`

`convergent([8.2 8.5 8.7 -8.2 -8.5 -8.7])`

`nearest([8.2 8.5 8.7 -8.2 -8.5 -8.7])`

`round([8.2 8.5 8.7 -8.2 -8.5 -8.7])`

`fix([8.2 8.5 8.7 -8.2 -8.5 -8.7])`

Привести и пояснить полученные результаты.

Знакомство со стандартными функциями преобразования систем счисления.

Записать произвольное целое десятичное число и преобразовать его в шестнадцатеричное и двоичное.

Выполнить обратные преобразования.

Пояснить, какие стандартные функции использовались для преобразования.

Сохранение переменных на диске.

Присвоить переменным A, B и C произвольные значения и сохранить их в текущей папке в файле с произвольным именем.

Пояснить:

- какая команда используется для сохранения данных;
- как выбирается имя файла данных;
- какое расширение имеют файлы данных.

Знакомство с рабочим пространством памяти Workspace.

Выполнить следующие действия:

- очистить и проверить содержимое Workspace;
- загрузить сохраненный файл данных и вывести значение переменных A, B, C в окне Command Window;
- проверить содержимое Workspace;

- удалить из Workspace переменную *a* и проверить содержимое Workspace.

Пояснить назначение Workspace и выполняемые команды.

Завершение работы MATLAB.

2.2 Практическая работа № 2. Операции с матрицами.

Цель занятия: овладеть навыками матричной обработки данных в MATLAB.

Содержание работы связано с изучением типовых операций с матрицами в MATLAB в режиме прямых вычислений.

Задание на практическую работу включает в себя следующие пункты:

Определение длины вектора и размера матрицы.

Ввести:

- произвольную матрицу *A*;
- пустую матрицу *Z*;
- вектор *B* в виде регулярной сетки с начальным элементом $-\pi$, конечным π и шагом $\pi/32$.

Определить размеры матриц и длину вектора.

Пояснить:

- что такое длина вектора и размер матрицы и как они определяются в MATLAB;
- с какой целью и как вводится пустая матрица и каков ее размер;
- как вводится регулярная сетка; в каком случае допускается не указывать шаг изменения значения переменной.

Генерирование типовых матриц.

Сгенерировать следующие квадратные матрицы 3-го порядка:

- нулевую матрицу *C*;
- матрицу единиц *D*;
- единичную матрицу *D1*;
- матрицу Теплица *T* с произвольными вещественными элементами первого

столбца;

- матрицу E случайных чисел, распределенных по равномерному закону;
- матрицу F случайных чисел, распределенных по нормальному закону.

Пояснить:

- как выполняется генерация указанных матриц в MATLAB;
- что собой представляют матрицы: нулевая, единиц, единичная и Теплица.

Выделение элементов матрицы.

В матрице F выделить:

- второй элемент третьей строки;
- вектор диагональных элементов;
- первую строку;
- третий столбец;
- подматрицу с номерами строк 2:3 и номерами столбцов 1:3.

Пояснить, как происходит выделение подматриц.

Преобразование матриц.

Произвести с матрицей F следующие преобразования:

- выполнить горизонтальную конкатенацию матрицы F с матрицей C;
- выполнить вертикальную конкатенацию матрицы F с матрицей D;
- сформировать квадратную матрицу G 6-го порядка посредством копирования матрицы F.

Пояснить, как выполняются указанные преобразования.

Поэлементные операции с матрицами.

Для всех элементов матрицы F выполнить операцию возведения в квадрат и умножения на 2.

Пояснить, какие символы арифметических операций использованы.

Сложение и вычитание матриц.

Ввести квадратные матрицы A и B 3-го порядка с произвольными вещественными элементами.

Выполнить операции сложения и вычитания матриц A и B и присвоить результаты переменным C1 и C2.

Пояснить:

- что собой представляют переменные C1 и C2;
- является ли операция сложения (вычитания) матриц коммутативной.

Умножение матриц.

Ввести матрицы A и B с произвольными вещественными элементами. Размеры матриц выбрать так, чтобы для этих матриц была возможна операция умножения.

Выполнить умножение матрицы A на матрицу B и присвоить результат переменной C.

Пояснить:

- как должны быть согласованы размеры матриц A и B, чтобы для них была возможна операция умножения;

- что собой представляет переменная C;
- является ли операция умножения матриц коммутативной.

Транспонирование и эрмитово сопряжение матриц.

Выполнить следующие операции:

- транспонировать матрицу F;
- сформировать квадратную матрицу P 3-го порядка с произвольными комплексными элементами;
- транспонировать матрицу P;
- сформировать матрицу R, эрмитово сопряженную с матрицей p;
- сформировать матрицу RI с комплексно сопряженными элементами относительно матрицы p.

Пояснить, как указанные операции выполняются в MATLAB.

Обращение матриц.

Выполнить следующие операции:

- вычислить определитель матрицы F;
- сформировать матрицу F1, обратную к матрице F;
- найти произведение матриц F и F1 и присвоить результат переменной F2.

Пояснить:

- для какой матрицы возможна операция обращения;

- какая функция служит для вычисления определителя матрицы;
- что собой представляет переменная F2 и с какой целью она вычислена.

Решить СЛАУ

$$\begin{cases} x_1 + 2x_2 + 3x_3 = 14; \\ 2x_1 - x_2 - 5x_3 = -15; \\ x_1 - x_2 - x_3 = -4. \end{cases}$$

и проверить правильность решения.

Пояснить, какая операция матричного деления используется и почему.

Вычисление норм матрицы и вектора.

Для матрицы F вычислить нормы.

Для вектора $x = \text{rand}(1,100)$ вычислить нормы.

Пояснить смысл указанных норм и способ их вычисления в MATLAB.

Операции с матрицами в задачах математической статистики.

Для матрицы F вычислить:

- максимальные и минимальные элементы столбцов;
- сумму и произведение элементов столбцов;
- средние значения элементов столбцов;
- СКО элементов столбцов с помощью функции `std (F,1)`;
- дисперсию элементов столбцов с помощью функции `var (F,1)`.

Пояснить:

- какие функции MATLAB использованы в каждом из этих случаев (кроме `std` и `var`);
- как проверить правильность результатов согласно определению среднего значения, СКО и дисперсии.

2.3 Практическая работа № 3. Типы массивов.

Цель работы: изучить основные типы массивов, используемых в MATLAB, и овладеть навыками их формирования.

Содержание работы связано с изучением типов массивов в MATLAB в режиме прямых вычислений. Содержание работы связано с изучением типов массивов в MATLAB в режиме прямых вычислений.

Задание на практическую работу включает в себя следующие пункты:

Знакомство с матрицами числового и логического типов.

Ввести матрицу A с элементами:

$$A = \begin{bmatrix} 127,1 & -127,1 & 127,7 \\ -127,7 & 0 & 128,4 \\ -128,4 & 255,7 & 255,1 \end{bmatrix}$$

и выполнить с ней следующие действия:

- преобразовать в матрицу B целых 8-битовых чисел со знаком;
- преобразовать в матрицу C целых 8-битовых чисел без знака;
- преобразовать в матрицу D логического типа;
- определить тип матриц A, B, C, D.

Пояснить:

- как преобразовать матрицу A в матрицы B, C и D;
- по какому правилу формируются элементы матрицы B;
- по какому правилу формируются элементы матрицы C;
- по какому правилу формируются элементы матрицы D;
- как определяется тип матрицы.

Операции с матрицами числового типа.

Выполнить следующие операции:

- вычислить значения синуса всех элементов матриц A, B и C;
- вычислить суммы матриц A и B, B и C;
- проверить, являются ли элементы матрицы A числами, большими единицы, и

определить вид и тип результата;

- выполнить логическую операцию "И" с матрицами В, С и определить вид и тип результата.

Сделать выводы по результатам выполнения операций.

Знакомство с матрицами символьного типа.

Выполнить следующие действия:

- сформировать трехэлементный вектор-строку x символьного типа с ФИО студента;
- сформировать трехэлементный вектор-столбец y символьного типа с ФИО студента;
- сформировать матрицу F символьного типа 2x2 с элементами:

$$F = \begin{bmatrix} \text{КИХ} & R=15 \\ \text{FIR} & \text{Order}=15 \end{bmatrix};$$

- преобразовать матрицу A в матрицу G символьного типа;
- определить типы матриц A и G.

Пояснить:

- как обеспечить не слитный вывод элементов вектора-строки;
- как обеспечить автоматическое добавление пробелов при выводе элементов столбца;
- как преобразовать матрицу числового типа в матрицу символьного типа.

Знакомство с массивами записей (структурами).

Сформировать массив записей (структуру) с именем Filter для описания четырех фильтров.

Каждая запись должна содержать три поля со следующими именами и их значениями:

- Type (тип избирательности) — lowpass, highpass, bandpass, stopband;

- Order (порядок фильтра) — 10, 20, 30, 40;
- Poles (наличие полюсов) — true, false, false, true.

Выполнить следующие действия с массивом Filter:

- вывести список полей;
- вывести значение 1-й записи;
- вывести значения поля Type во всех записях;
- вывести значение поля Type в 3-й записи;
- удалить поле Poles.

Пояснить:

- с какой целью создается массив записей;
- что собой представляет запись и значение записи;
- каков размер массива записей Filter.

Знакомство с матрицами ячеек.

Создать матрицу ячеек S 3x3, элементами которой являются сформированные ранее массивы:

$$S = \begin{bmatrix} A & B & C \\ D & F & G \\ X & Y & \text{Filter} \end{bmatrix}.$$

Выполнить следующие действия:

- последовательно вывести элементы матрицы S и определить их тип;
- вывести графическое представление матрицы S.

Пояснить:

- из каких элементов создается матрица ячеек;
 - как эти элементы вводятся;
 - как выводится графическое представление матрицы ячеек;
- что оно собой представляет.

2.4 Практическая работа № 4. Средства графики.

Цель занятия: изучить графический инструмент MATLAB и овладеть навыками построения двумерных и трехмерных графиков.

Содержание работы связано с изучением инструментария MATLAB для построения, оформления и управления свойствами двумерных и трехмерных графиков.

Задание на практическую работу включает в себя следующие пункты:

Построение двумерного графика. Для аргумента x , заданного на интервале $x \in [0; 8\pi]$ с шагом $\Delta x = \pi/8$

- вычислить функцию $y_1 = \sin x$ и вывести ее график в линейном масштабе с линейной интерполяцией между соседними значениями.

- выполнить следующие действия по оформлению графика:

- 1 нанести координатную сетку;

- 2 обозначить ось абсцисс.

Пояснить:

- какая функция используется для вывода графика;

- в какое окно выводится график;

- какие функции используются для нанесения координатной сетки и обозначения оси абсцисс.

Построение нескольких двумерных графиков на одних координатных осях.

В том же окне вывести графики функций:

$$y_2 = \frac{\sin x}{x};$$

$$y_3 = 0,5 \cos x;$$

для которых аргумент x задан на интервале $x \in [0; 8\pi]$.

При выводе графиков выбрать различный цвет линий для функций y_1 , y_2 и y_3 .

Выполнить следующие действия по оформлению графика:

- обозначить ось ординат как axis y ;

- ввести заголовок графика в виде Functions $y_1 y_2 y_3$;

- разместить легенду для графиков функций:

$$y_1 = \sin(x); y_2 = \sin(x)/x; y_3 = 0.5\cos(x).$$

Пояснить:

- какая команда обеспечивает вывод нескольких графиков на одних координатных осях;

- какая функция используется для вывода графиков;

- сохраняется ли координатная сетка и обозначение оси абсцисс при выводе следующих графиков в то же окно;

- какие функции используются для обозначения оси ординат, вывода заголовка и размещения легенды.

Построение независимых графиков в одном окне с его разбиением на отдельные поля.

В графическом окне с именем Graph2D вывести друг под другом графики функций y_1 , y_2 и y_3 .

Выполнить следующие действия по оформлению графиков:

- нанести координатную сетку;

- обозначить оси абсцисс и ординат;

- ввести заголовки графиков.

Пояснить:

- как создается окно с заданным именем;

- какая функция позволяет строить несколько независимых графиков в одном графическом окне.

Построение графика последовательности чисел.

В окне Sequence1 вывести график значений функции y_2 с нанесением координатной сетки и без закрашивания маркеров.

В новом окне Sequence2 вывести тот же график с нанесением координатной сетки, закрашиванием маркеров и следующей установкой параметров управления:

- толщина линий равна 2;

- размер маркеров равен 6;

- цвет маркеров отличается от цвета линий;

- цвет закрашивания маркеров отличается от цветов линий и маркеров.

Пояснить:

- какая функция используется для вывода графика последовательности чисел;
- какой параметр этой функции отвечает за закрашивание маркеров;
- как устанавливаются параметры управления.

Построение графиков в полулогарифмическом и логарифмическом масштабах. По оси абсцисс x_1 задать диапазон значений $[1; 10^4]$ с помощью функции `logspace`.

Вычислить функцию $y_4 = \sqrt{x}$.

В окне `Logarithms axes` вывести друг под другом графики функции y_4 с нанесением координатной сетки и следующих масштабах по осям:

- логарифмическом — по оси абсцисс; линейном — по оси ординат;
- логарифмическом — по осям абсцисс и ординат.

Пояснить:

- как диапазон значений задается с помощью функции `logspace`;
- какая функция используется для вывода графика в логарифмическом масштабе по оси абсцисс;
- какая функция используется для вывода графика в логарифмическом масштабе по осям абсцисс и ординат.

Построение гистограмм.

В окне `Histogram` вывести гистограмму нормального белого шума — вектора y_5 длиной 1000. Количество интервалов выбрать по умолчанию.

Пояснить:

- какая функция используется для построения гистограммы;
- что отображает гистограмма;
- как гистограмма связана с плотностью вероятности нормального белого шума.

Построение трехмерного графика.

Для аргументов x и y , заданных на одинаковых интервалах:

- $x \in [-\pi; \pi]$ с шагом $\Delta x = \pi/32$,

- $y \in [-\pi; \pi]$ с шагом $\Delta y = \pi/32$,

вычислить функцию

$$z = \sin x + \cos y,$$

и в окне Graph3D вывести ее сетчатый график с автоматическим нанесением координатных сеток.

Выполнить следующие действия по оформлению графика:

- выбрать фиолетово-голубую палитру;
- обозначить оси x , y , z ;
- вывести на поле графика шкалу цветов.

Пояснить:

- с чего начинается построение трехмерного графика; какая функция для этого используется;
- какая функция используется для вывода графика;
- какая функция используется для выбора палитры;
- какая команда используется для вывода шкалы цветов.

Контрольные вопросы:

1. В какое графическое окно выводится график по умолчанию?
2. Как вывести график в новое графическое окно?
3. Как вывести несколько графиков на одних координатных осях?
4. Как удалить графики перед выводом нового графика в то же графическое окно?
5. Как вывести несколько независимых графиков в одном графическом окне с его разбиением на отдельные поля?
6. Какие средства оформления графиков используются в MATLAB?
7. Какие средства предусмотрены для установки типа, цвета и толщины линий?
8. Какие средства предусмотрены для установки вида, размера и цвета маркеров?
9. Какая функция используется для построения двумерных графиков в

линейном масштабе с линейной интерполяцией между соседними значениями?

10. Какая функция используется для построения графика последовательности чисел?

11. Какие функции используются для построения графиков в полулогарифмическом и логарифмическом масштабах?

12. Какая функция используется для построения гистограмм?

13. В чем заключается подготовка перед построением трехмерного графика?

14. Какие функции используются для построения трехмерных графиков?

15. Как выбрать палитру цветов при построении трехмерного графика?

16. Как вывести шкалу цветов на поле трехмерного графика?

3 Задания для выполнения лабораторных работ

3.1 Лабораторная работа № 1 Дифференциальные уравнения и математическое моделирование.

Цель работы: ознакомление с применением обыкновенных дифференциальных уравнений (ОДУ) в решении элементарных задач математического моделирования, а также способами решения ОДУ (задач Коши и краевых задач) в программах научного программирования (на примере MATLAB).

Задание: Разобрать примеры 1-2, выполнить задачи 1 и 2 с учетом вариантов. По результатам выполнения задачи представить научный отчет.

Понятие об обыкновенных дифференциальных уравнениях.

Обыкновенным дифференциальным уравнением (ОДУ) называется уравнение, Математическое моделирование служит для постановки и решения задач реального мира в математических терминах (рисунок 3.1).

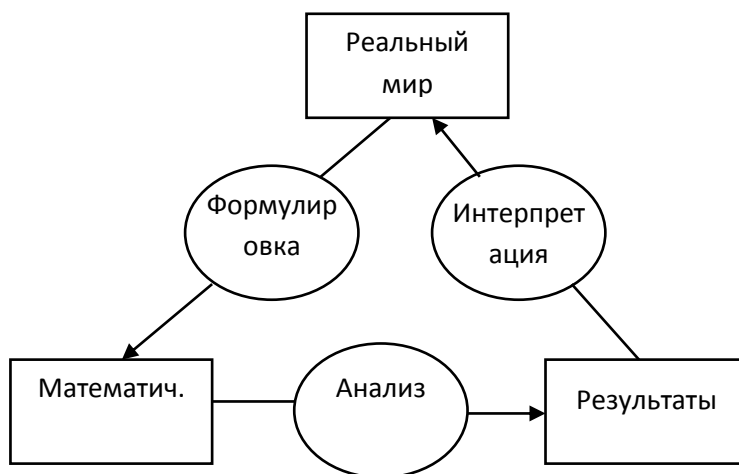


Рисунок 3.1 – Схематический процесс математического моделирования

Обыкновенным дифференциальным уравнением (ОДУ) называется уравнение, связывающее между собой значения независимой переменной x , неизвестной функции $y = f(x)$ и её производных (или дифференциалов): $F(x, y, y', y'', \dots, y^{(n)}) = 0$.

Порядком уравнения называется максимальный порядок n входящей в него производной (или дифференциала).

Решением (интегралом) дифференциального уравнения порядка n называется функция $y(x)$, имеющая на некотором интервале (a, b) производные до порядка n включительно и удовлетворяющая этому уравнению. Процесс решения дифференциального уравнения называется интегрированием

Все дифференциальные уравнения можно разделить на обыкновенные (ОДУ), в которые входят только функции (и их производные) от одного аргумента, и уравнения с частными производными (ДУВЧП), в которых входящие функции зависят от многих переменных.

Геометрический смысл уравнения первого порядка. Уравнение $y' = f(x, y)$ определяет в каждой точке (x, y) области D , в которой задана функция $f(x, y)$, определяет - угловой коэффициент касательной к решению, проходящему через точку (x, y) , т.е. направление, в котором проходит решение через эту точку (рисунок 3.2).

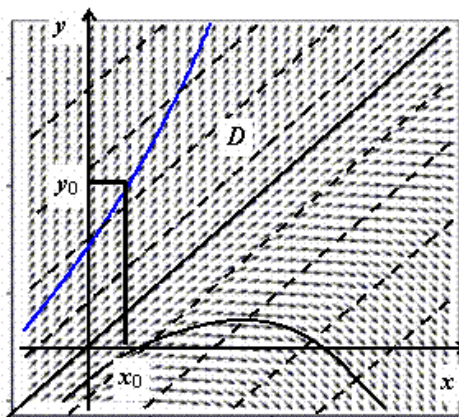


Рисунок 3.2 - Поле решений и общие решения ОДУ: $y' = y - x + 1$

Говорят, что уравнение $y' = f(x, y)$ задаёт в D поле направлений. График любого решения дифференциального уравнения (называемый также интегральной кривой) в любой своей точке касается этого поля, т.е. проходит в направлении, определяемом полем.

Интегрирование дифференциального уравнения геометрически означает

нахождение кривых, у которых направление касательной в каждой точке совпадает с направлением поля. На рисунке справа изображено поле направлений, определяемое уравнением , и три интегральные кривые (три частных решения) этого уравнения. Решение можно провести через любую точку области D; единственное решение можно выделить, если задать точку, через которую проходит интегральная кривая (подсвечено синим на рисунок 3.1): $y|_{x=x_0} = y_0$

Приложения к математическому моделированию и информатике.

Иногда для моделирования производственных процессов в нефтегазовой отрасли промышленности интересны не столько функция $f(x)$, сколько скорость $f'(x)$ и ускорение $f''(x)$ ее изменения. В некоторых случаях только скорость и ускорение бывают известны. В этом случае требуется найти функцию (интеграл).

Пример №1. Постановка модели и решение простейшей задачи. Для рекультивации нефтезагрязненных почв зачастую используются живые организмы, штамм которых вносится в грунт. По условию задачи, колония живых организмов, разлагающих углеводороды, находится в благоприятных условиях, благодаря чему прирост $\alpha > 0$ (рождаемость γ выше, чем смертность σ , $\alpha = \gamma - \sigma$; $\alpha > 0$). Для упрощения, пространство, занимаемое колонией, и ее пищевые ресурсы (т.е. пропитывающие почву углеводороды) будем считать неограниченными. Предположим также, что естественных врагов, питающихся организмами данной колонии, нет. Найти закон изменения численности организмов в зависимости от времени, если при $t = 0$ их число равнялось y_0 .

Решение: будем считать, что скорость изменения численности организмов пропорциональна этой численности и α - коэффициент пропорциональности (рождаемость γ выше, чем смертность σ , $\alpha = \gamma - \sigma$; $\alpha > 0$):

Так как $V = y'$, то численность y организмов в колонии в момент времени t удовлетворяет уравнению: $y' = \alpha \cdot y$.

Отсюда $\frac{dy}{dt} = \alpha \cdot y$.

Разделяем переменные: $\frac{dy}{y} = \alpha \cdot dt$. Начальное условие: при $t=0$ численность особей $x=x_0$.

Интегрируя, получим общее решение:

$$\int_{x_0}^x \frac{dy}{y} = \int_0^t \alpha \cdot dt$$

$$\ln y - \ln y_0 = \ln \frac{y}{y_0} = \alpha t$$

$$y = y_0 \cdot e^{\alpha t}$$

Значит, число организмов в колонии изменяется по законам

$$y = y_0 \cdot e^{\alpha t}$$

Ответ: число организмов в колонии изменяется по законам

$$y = y_0 \cdot e^{\alpha t}$$

Производные и интегралы в MATLAB:

Нахождение интеграла в MATLAB осуществляется с помощью функции `mt()`:

`int(expr,var,a,b);`

Здесь: `expr` - выражение, `var` - переменные выражения, `a` и `b` - нижние и верхние интервалы интегрирования.

Рассмотрим пример решения интеграла выше:

$$\int \alpha \cdot dt = \alpha t$$

```
syms x alpha %define x as symbolic variables int(alpha,x);
```

```
%команда выше выведет
```

```
ans=alpha*x
```

Дифференцирование в MATLAB:

$$f'(x) = \sin(5x)$$

```
%define x as symbolic variables syms x
```

```
fdot = sin(5*x); diff(f)
```

```
%команда выше выведет
```

```
ans =
```

```
5*cos(5*x)
```

Решение ОДУ в MATLAB.

Для решения обыкновенных дифференциальных уравнений в MATLAB применяется функция `ode45()`. С помощью этой функции можно граничные задачи.

Синтаксис команды: `[T y]=ode45(ydot,t,y0)`, здесь: y – вектор (массив- строка) решений дифференциального уравнения, y_0 - начальное значение функции $y(t)$, t_0 – начальное значение аргумента, t – вектор значений времени работы модели, $ydot$ – производная исследуемой функции, T – соответствующее y значение функции.

Выполним постановку, табличное и графическое решение задачи о численности популяции (Пример №1) в MATLAB для определенного коэффициента прироста популяции $\alpha = 0.3$ и на множестве $t = \overline{1,10}$ (т.е. с момента времени 0 до 10).

Код MATLAB для примера 1:

```
clear;
```

```
y0=3;
```

```
t0=0;
```

```
alpha=0.3;
```

```
t=0:0.1:10;
```

```
ydot= @(t, y) alpha*y; %derivative function
```

```
[T y]=ode45(ydot,t,y0);
```

```
plot(T, y);
```

Как видно из примера выше, там объявляется функция с помощью оператора `@`. Это так называемый “анонимный” способ объявления функции. Он альтернативен принятому в MATLAB объявлению функции с помощью ключевого

параметра function.

```
function ydot(y, t)
global alpha;
ydot=alpha*y;
```

Эту функцию необходимо объявить в отдельном файле ydot.m. Ключевое слово global применяется для указания переменной alpha, объявленной в командном окне.

Пример №2. Решение простой задачи теоретической механики. Для планирования действий по ликвидации последствий возможных чрезвычайных ситуаций (ЧС) на нефтегазовых предприятиях иногда важно знать не только направление, но и скорость движения ветра, которая изменчива и зависит от внешних условий.

Проходя через лес и испытывая сопротивление деревьев, ветер теряет часть своей скорости. На бесконечно малом пути эта потеря пропорциональна скорости в начале этого пути и его длине. Найти скорость ветра, прошедшего в лесу 150 м, зная, что до вступления

в лес начальная скорость ветра $V_0 = 12$ м/с; после прохождения пути $S=1$ м, скорость ветра уменьшилась до величины $V_1 = 11,8$ м/с

Решение: Пусть на расстоянии S от начала леса скорость ветра равна V , потеря скорости на пути dS равна $-dV$ (процесс убывающий). Эта потеря пропорциональна V , и поэтому дифференциальное уравнение процесса примет вид: $-\frac{dV}{dS} = k \cdot V$

Разделяем переменные: $\frac{dV}{V} = -k \cdot dS$

Интегрируя, получим общее решение задачи:

$$V = c \cdot e^{-ks}.$$

Найдем частное решение, используя начальное условие: при $S = 0, V = V_0$
Подставим это условие в последнее уравнение: $V = c \cdot e^0 \Rightarrow c = V_0$.

Закон процесса:

$$V = V_0 \cdot e^{-ks}.$$

Для определения коэффициента пропорциональности k используем

дополнительное условие:

при $S = 1\text{ м}$, $V = V_1 = 11,8\text{ м/с}$.

Откуда: $V_1 = V_0 \cdot e^{-k}$, или: $e^{-k} = \frac{V_1}{V_0} = \frac{11,8}{12} \approx 0,983$.

Подставляя числовые значения в уравнение в последнее уравнение, получим искомую скорость

$$V = 12 \cdot (0,983)^{150} = 12 \cdot 0,0776 \approx 0,92\text{ м/с}.$$

Итак, скорость ветра, углубившегося на 150м в лес, составит 0,93м/с.

Ответ: $V(150) = 0,92\text{ м / с}$.

Для использования программы `ode45()` MATLAB необходимо найти коэффициент пропорциональности k . В коде ниже показано его вычисление.

Код MATLAB для решения примера 2:

```
clear; %memory clean up
V0=12;
k=-log(11.8/12);
%k=0.016;
t=0:1:150;
Vdot= @(S, V) -k*V; %derivative function
[T V]=ode45(Vdot,t,V0); plot(T, V); ylabel('V(S),[m/s]'); xlabel('S,[m]');
title('Changes of wind speed per distance V(S)'); disp ('Wind speed after 150 m = ');
disp (V(150));
```

Вызов программы в MATLAB возвращает графическое решение уравнения (рисунок 3.3) и результат в командной строке.

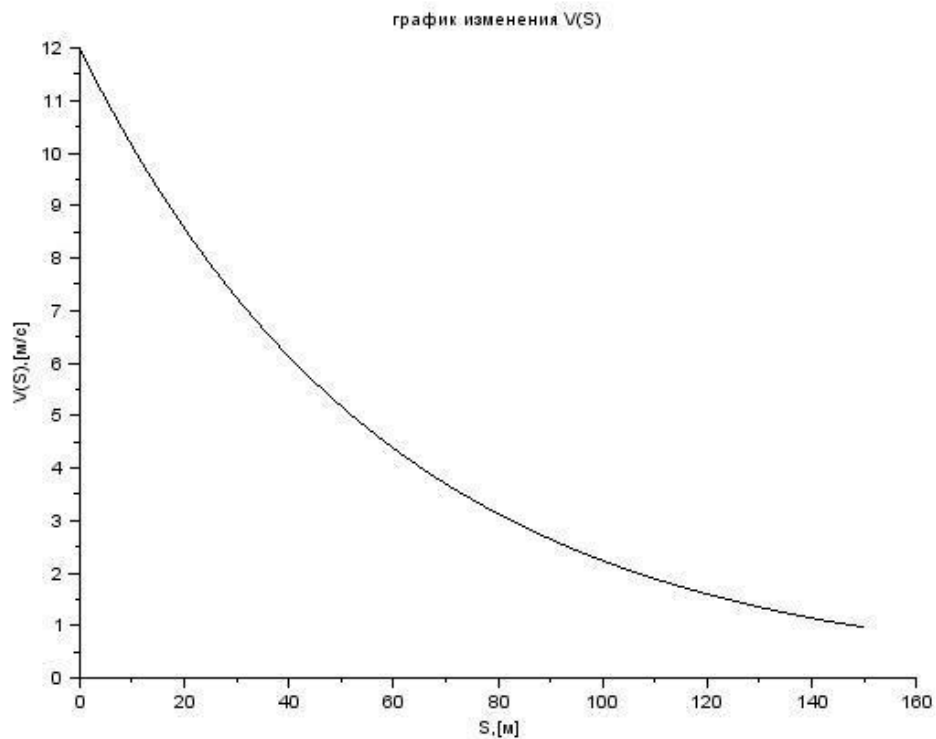


Рисунок 3.3 – График изменения скорости ветра $V(S)$ по мере его продвижения в лесу на расстояние S

Задачи для самостоятельного решения

Задача 1. Разрядка аккумуляторной батареи. Требуется сформулировать задачу разрядки батареи, питающий лампу. Электрическая схема цепи состоит из источника постоянного тока с напряжением U , выключателя и лампы.

Чем дольше горит лампа, тем меньшее напряжение должна вырабатывать батарейка модели и тем слабее должна светить лампочка. В этом и состоит динамика модели: напряжение и связанная с ним яркость свечения лампы зависят от времени.



Рисунок 3.4 – Электрическая схема цепи и электролампы

Уменьшение напряжения батарейки со временем опишем уравнением:

$$U_B(t) = U_0 \cdot e^{-t/T} = U_0 \cdot e^{-\alpha t},$$

где: $U_B(t)$ – зависимость напряжения батарейки от общего времени ее работы;

U_0 – напряжение, создаваемое новой батареейкой;

t – общее время работы батарейки в фонарике;

T – срок службы батарейки, работающей в фонарике. Это характерное время, в течение которого батарейка разрядится почти на две трети, точнее, ее напряжение составит 37% от начального. У многих батареек для фонариков время T составляет 2–3 часа.

Программа, моделирующая фонарик, должна учитывать время работы батарейки и уменьшать ее напряжение в соответствии с формулой выше. Отметим, что при решении этой задачи мы пренебрежем сопротивлением проводов, поэтому напряжение на лампочке будет равно напряжению батарейки.

Для решения задачи нам потребуется представить уравнение разряда батарейки в дифференциальном виде: $\frac{dU}{U} = -\alpha \cdot dt$.

Вариант задачи (Приложение А) содержит: U_0 ; U_n ; – напряжение через время n ; n – время работы до уменьшения напряжения до U_n ;

Требуется:

а) Найдите численное решение уравнения этой задачи, задав характеристики аккумуляторной батареи (см. вариант задачи, Приложение А). Как найти коэффициент пропорциональности α ?

б) По аналогии с предыдущими упражнениями, составьте код MATLAB. Когда разрядится аккумулятор?

в) Постройте, график изменения напряжения $U(t)$ в MATLAB для вставки в отчет.

Задача 2. Зарядка электроконденсатора. Электрический конденсатор изначально не заряжен. Процесс зарядки электрического конденсатора в цепи (рисунок 3.5), к которому в момент времени $t = 0$ приложено напряжение U , описывается ОДУ первого порядка

$$R \frac{dQ}{dt} = V - \frac{Q}{C}$$

с начальным условием $Q(0)=0$.

Требуется:

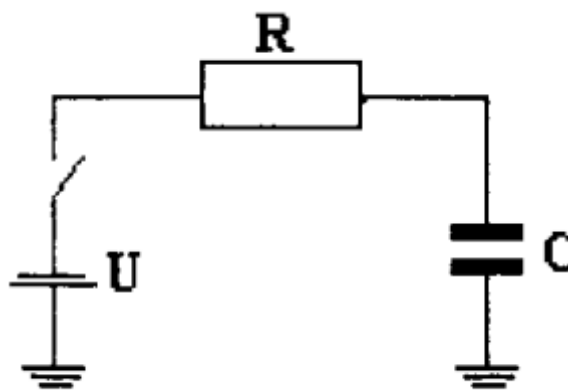


Рисунок 3.5 – Схема зарядки электрического конденсатора в цепи

а) Найдите численное решение уравнения этой задачи, задав следующие характеристики цепи сопротивление R , емкость C , напряжение V (см. вариант

задачи, Приложение Б)

б) Ответьте, на вопросы. Как зависит заряд конденсатора от времени. Заряжается ли он бесконечно, или в какой-то момент времени происходит его насыщение?

в) Постройте, график зарядки конденсатора для вставки в отчет.

Указание. Для увеличения точности решения можно брать время с меньшим шагом. Рекомендуемый шаг времени t наблюдений 0.0001 с, диапазон наблюдений $t = \overline{0,0.05}$ с.

Рекомендация. Для проверки незначительности увеличения заряда конденсатора с дальнейшим течением времени можно использовать код, аналогичный приведенному ниже:

Код MATLAB для обхода вектора значений заряда Q в момент времени t и проверки малости отличий между $Q(i)$ и $Q(i-1)$:

```
//проверка малости изменений пригл. значений вектора Q
for i=2:length(Q)
    i_min=i-1;
    if Q(i)-Q(i_min)<1e-9
        disp('Конденсатор заряжен в момент времени t~');
        disp(t(i));
        disp ('Заряд конденсатора достиг Q = ');
        disp (Q(i));
        break;
    end;
end
```

Требования к отчету

Отчет о выполнении лабораторной работы должен содержать:

- тему и цель лабораторной работы на титульной странице;
- оглавление;
- необходимые теоретические сведения по теме решаемых задач;
- исходные данные и последовательность их обработки (преобразования);

- поэтапный расчет в виде вставок кода Matlab в объектах «Подпись»;
- результаты выполнения кода Matlab в виде вывода в командном окне;
- графики, выполненные в Matlab;
- интерпретацию результатов работы моделей;
- отметку преподавателя о выполнении лабораторной работы.

Варианты исходных данных для решения самостоятельной задачи №1

Вариант	U_0, V	U_n, V	n, мин
1	19	17	8
2	20	18	5
3	21	19	5
4	22	20	8
5	16	14	8
6	16	14	7
7	20	18	7
8	19	17	8
9	18	16	8
10	21	19	5
11	17	15	11
12	19	17	7
13	21	19	12
14	17	15	9
15	16	14	8
16	19	17	10
17	22	20	7
18	16	14	12
19	17	15	10
20	17	15	6
21	20	18	6
22	20	18	12
23	16	14	9
24	16	14	12
25	16	14	12
26	16	14	8
27	20	18	8
28	16	14	12
29	16	14	9
30	15	14	7
31	20	18	11
32	20	18	5
33	17	15	12
34	16	14	7
35	22	20	10
36	21	19	9
37	16	14	12
38	17	15	12
39	18	16	8
40	18	16	6
41	16	14	7
42	15	14	7
43	21	19	8
44	15	14	6
45	22	20	9
46	20	18	9
47	17	15	12
48	15	14	11
49	18	16	6
50	18	16	10

Приложение Б

Варианты исходных данных для решения самостоятельной задачи №2

Вариант	$R, \text{Ом}$	$C, 10^{-n}$	$V, \text{В}$
1	2300	4	5
2	2400	4	5
3	1900	5	10
4	2400	4	11
5	1600	5	7
6	2100	5	5
7	1500	5	7
8	2400	5	8
9	1700	5	8
10	1700	5	7
11	2000	5	6
12	1800	6	9
13	1700	4	6
14	1500	5	8
15	2500	6	9
16	1800	5	10
17	1800	6	12
18	2000	4	12
19	1500	5	12
20	2200	4	5
21	2200	4	6
22	1500	5	5
23	2500	5	9
24	2400	4	11
25	2300	4	8
26	2300	4	6
27	2100	4	10
28	2300	5	10
29	2500	6	12
30	2300	4	8
31	2000	4	12
32	1500	6	5
33	2300	5	9
34	2500	4	6
35	2200	6	5
36	1800	6	10
37	2400	5	8
38	1900	4	5
39	1500	4	12
40	1800	5	6
41	2400	6	5
42	2000	6	6
43	2300	5	5
44	1600	4	11
45	1900	5	11
46	2500	5	9
47	1800	6	5
48	2100	5	7
49	2200	4	7
50	2200	6	9

3.2 Лабораторная работа № 2 Основы моделирования электрических цепей в MATLAB/SIMULINK (scilab xcos)

Цель работы заключается в приобретении умения работать с программой визуального моделирования SIMULINK (Xcos) , навыков составления и использования имитационных моделей.

Программа работы:

- изучить теоретический материал по теме, используя теоретические сведения данного практикума, а также литературу и электронные ресурсы;
- скачать и установить пакет Scilab;
- отладить четыре модели Xcos, представленные в теоретических сведениях к данной работе (рис. 1.8, 1.13, 1.14, 1.15);
- установить и интегрировать с Scilab компилятор C++, необходимый для работы электрических блоков;
- разработать компьютерную модель однофазного неуправляемого выпрямителя по принципиальной схеме предложенного варианта задания;
- исследовать работу модели в двух режимах: без выходного фильтра и с емкостным фильтром на выходе выпрямителя;
- выполнить пункты 1–6 в MATLAB/SIMULINK;
- подготовить и защитить отчет.

Краткие теоретические сведения

Моделирование - это построение и изучение моделей реально существующих объектов, процессов или явлений с целью получения объяснений этих явлений, а также для предсказания явлений, интересующих исследователя. Одним из видов моделирования является имитационное моделирование. Имитационная модель – это логико-математическое описание объекта, которое может быть использовано для экспериментирования на компьютере в целях проектирования, анализа и оценки функционирования объекта. К имитационному моделированию прибегают, когда:

- дорого или невозможно экспериментировать на реальном объекте;
- невозможно построить аналитическую модель: в системе есть время,

причинные связи, последствия нелинейности, стохастические (случайные) переменные;

- необходимо симитировать поведение системы во времени.

Цель имитационного моделирования состоит в воспроизведении поведения исследуемой системы на основе результатов анализа наиболее существенных взаимосвязей между ее элементами, или, другими словами, в разработке симулятора (англ. simulation modeling) исследуемой предметной области для проведения различных экспериментов.

Существует ряд коммерческих и свободно распространяемых компьютерных средств, позволяющих выполнить имитационное моделирование. Одной из наиболее известных бесплатных математических программ является Scilab.

Scilab (читается «сайлэб») – это пакет прикладных математических программ, предоставляющий открытое окружение для инженерных, технических и научных расчетов. Пакет можно скачать с официального сайта <http://www.scilab.org> (рисунок 1.1).

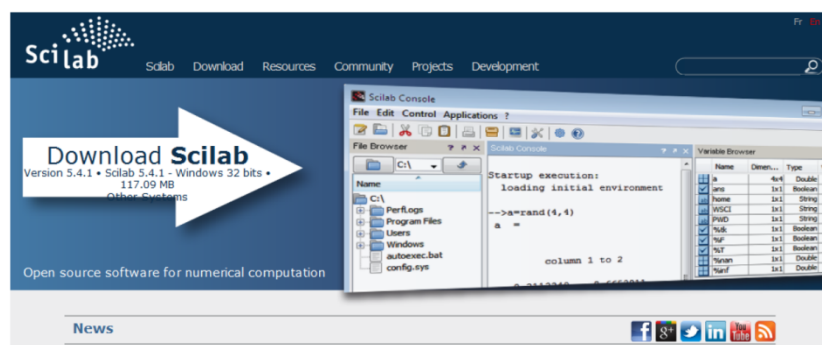


Рисунок 1.1 – Дистрибутив Scilab 5.4.1 для Windows 32 bits на официальном сайте

После установки и запуска пакета Scilab на экране появится главное окно, показанное на рисунок 1.2.

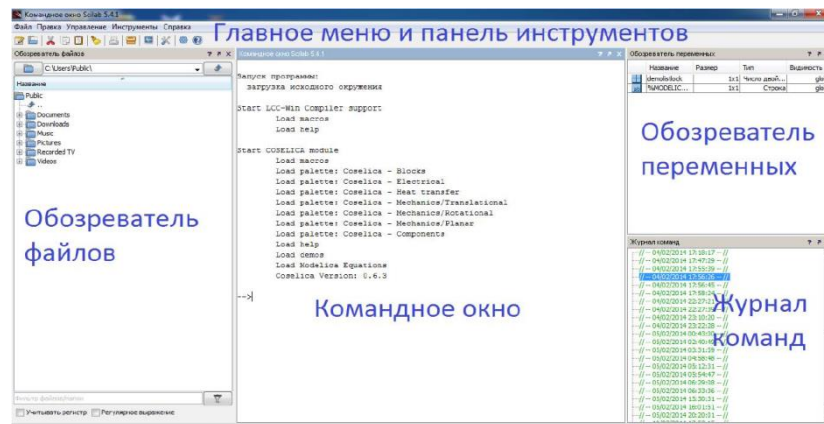


Рисунок 1.2 – Главное окно Scilab

В состав пакета входит программа Xcos – инструмент для редактирования блочных диаграмм и симуляции (аналог Simulink в пакете MATLAB). Программа Xcos разработана на основе другой программы – Scicos. Раньше Scicos входила в состав Scilab вместо Xcos, но сейчас распространяется как самостоятельный продукт, тоже бесплатно. В настоящее время программы очень похожи как визуально, так и по возможностям. В верхней части окна расположено главное меню с панелью инструментов. В рабочей области слева находится обозреватель файлов: в нем можно изменять текущий каталог, просматривать его содержимое и открывать файлы. В центральной части расположено командное окно: здесь пользователь вводит текстовые команды, которые выполняются Scilab. Различные пункты главного меню попросту дублируют многие текстовые команды во избежание необходимости помнить их точное написание. В этом же окне Scilab выводит результат выполнения команд: сообщения об успехе, статистическую информацию, а также весьма важную информацию для имитационного моделирования – текст ошибок, возникающих в процессе работы. С правой стороны рабочей области по умолчанию располагаются обозреватель переменных (сверху) и журнал команд (снизу). В обозревателе будут отображаться переменные среды. Например, при моделировании электронной схемы необходимо сохранить и отобразить временную диаграмму в виде числового массива для последующего математического анализа.

Хорошими вариантами решения может быть сохранение этих чисел в файл на

диске или в переменную в памяти. Во втором случае данные удобно просматривать в окне обозревателя переменных. В журнале команд отображается история ввода команд пользователем.

Для запуска программы Xcos выполните команду главного меню Инструменты → Визуальное моделирование Xcos, как показано на рисунок 1.3.

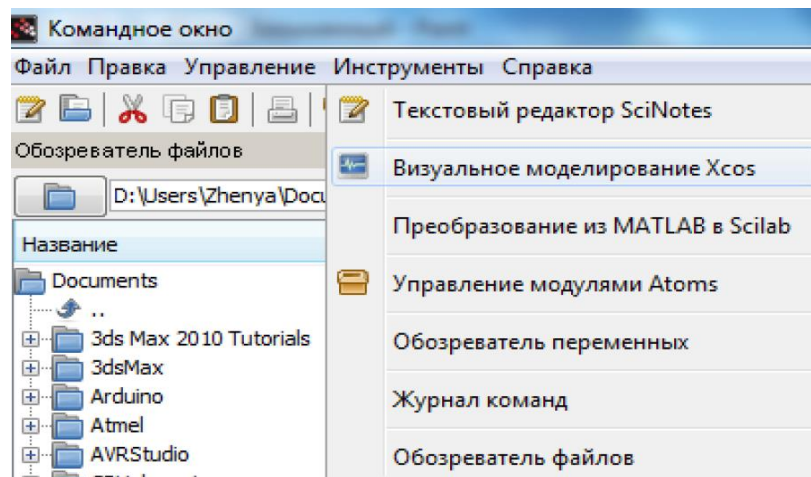


Рисунок 1.3 – Запуск программы Xcos

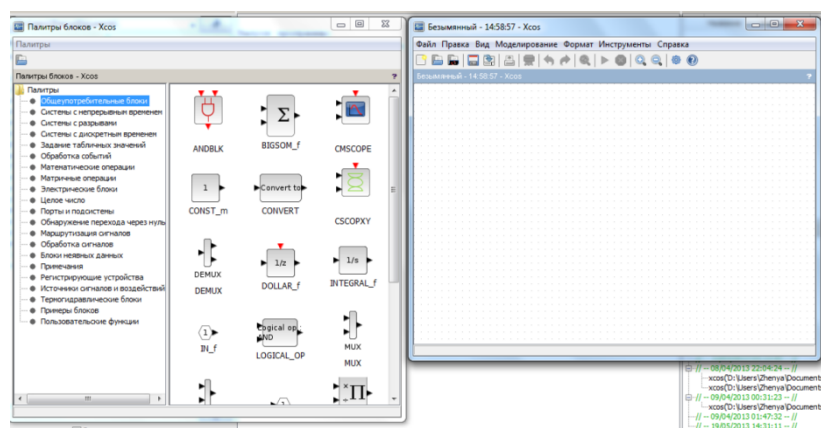


Рисунок 1.4 – Xcos

В версии 5.4.1 появится два окна: главное окно модели и палитра блоков (рисунок 1.4). Если окно с палитрами закрыто, его всегда можно отобразить,

выполнив команду меню (уже главного меню Xcos) Вид → Палитры блоков.

Создание модели в Xcos будет включать следующие шаги:

- размещение блоков в окне модели;
- создание связей между блоками;
- задание параметров блоков;
- настройка параметров моделирования;
- моделирование и отображение результатов.

Рассмотрим их на примере отображения сигнала, форма которого определяется уравнением $y(t) = \sin(t) + \sin(2t)e^{-0.1t}$.

Все блоки программы сгруппированы по категориям в палитры. Блоки, являющиеся источниками сигналов, расположены в палитре «Источники сигналов и воздействий». Блок представляет собой графическое изображение, чаще всего в форме прямоугольника, обозначающее его тип. У блока часто имеются входные и выходные порты. Порты бывают нескольких видов:

- порт явных данных;
- порт неявных данных;
- управляющий порт.

Порт явных данных отображается черным треугольником. Он используется для большинства сигналов, над которыми выполняются математические и логические операции.

Порт неявных данных отображается черным или белым квадратом (на рисунке 1.5 блоки с такими портами отсутствуют). Он применяется, например, в палитре электрических блоков в виде выводов резисторов, конденсаторов и других элементов.

Управляющий порт отображается красным треугольником. Он используется для управления работой модели: включения определенных подсистем в работу модели по времени или управления работой осциллографов и т. д. Очевидно, что напрямую разные типы портов соединить не получится. Также направление треугольника показывает, является порт входным или выходным. Нельзя соединить входной порт с входным портом или выходные порты между собой.

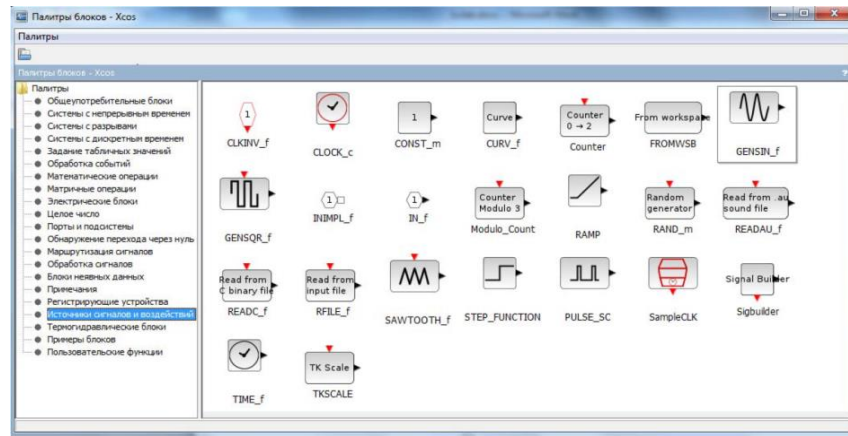


Рисунок 1.5 – Палитра «Источники сигналов и воздействий»

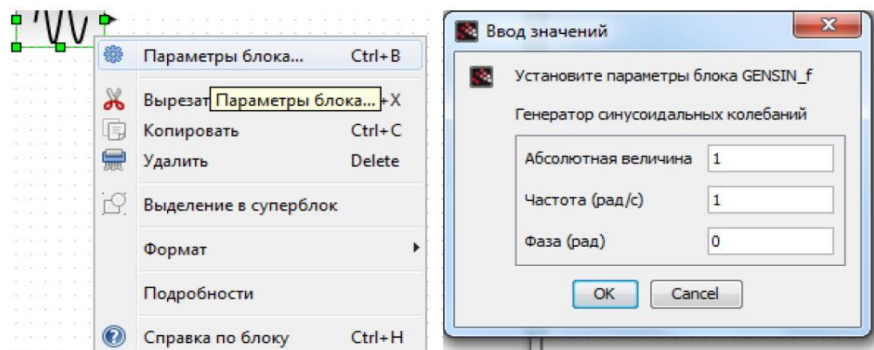


Рисунок 1.6 – Контекстное меню блока и окно параметров GENSIN_f

Источники сигналов обладают выходным портом, через который сигнал определяется типом блока и его параметрами. Интуитивно кажется, что единственным блоком, использование которого поможет смоделировать требуемый сигнал $y(t) = \sin(t) + \sin(2t)e^{-0.1t}$ является генератор синусоидальной функции GENSIN_f. Перенесем его в окно модели (рисунок 1.6). Открыть окно параметров блока можно двойным щелчком мыши или из пункта контекстного меню (открывается правой кнопкой). В появившемся окне можно отредактировать амплитуду, частоту и фазу генерируемого синусоидального сигнала. По умолчанию настройки генератора соответствуют функции $\sin(t)$, амплитуда равна 1, частота 1 рад/с, а начальная фаза отсутствует. Для функции $\sin(2t)$ в окне «Частота (рад/с)» единицу замените на 2.

Рассмотрим порядок реализации экспоненциальной функции блоком из палитры «Источники сигналов и воздействий». Попробуем для начала отобразить несколько периодов функции $\sin(t)$. Для отображения сигналов используются блоки из палитры «Регистрирующие устройства». Вынесите блок одноканального осциллографа CSCOPE (в палитре регистрирующих устройств присутствует 5 видов осциллографов с одинаковыми иконками, но с разными названиями) в окно модели и настройте его параметры (рисунок 1.7).

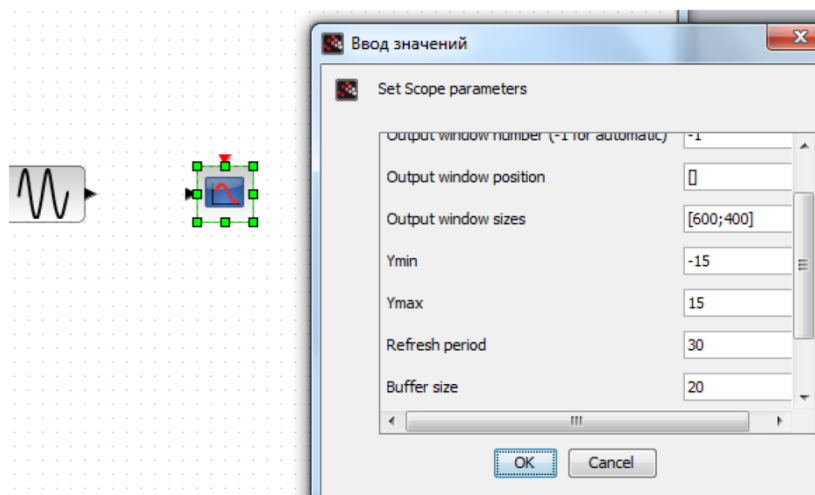


Рисунок 1.7 – Настройка осциллографа

Нас интересуют в первую очередь параметры Y_{min} , Y_{max} и Refresh period – пределы отображения по вертикальной оси и оси времени. Задайте значения -1.5 , 1.5 , 20, так как функция $\sin(t)$ ограничена значениями $-1, 1$. Период составляет 2π , что примерно равно 6.28, и в нашем случае отобразится чуть более трех периодов. Управляющий входной порт необходим для «тактирования» данного виртуального осциллографа. Необходимо подавать импульсы с частотой, с которой требуется отображать точки кривой на осциллографе.

Добавьте в модель блок CLOCK_c, который находится в рассмотренной ранее палитре «Источники сигналов и воздействий». Настройте блок, как показано на рисунок 1.8.

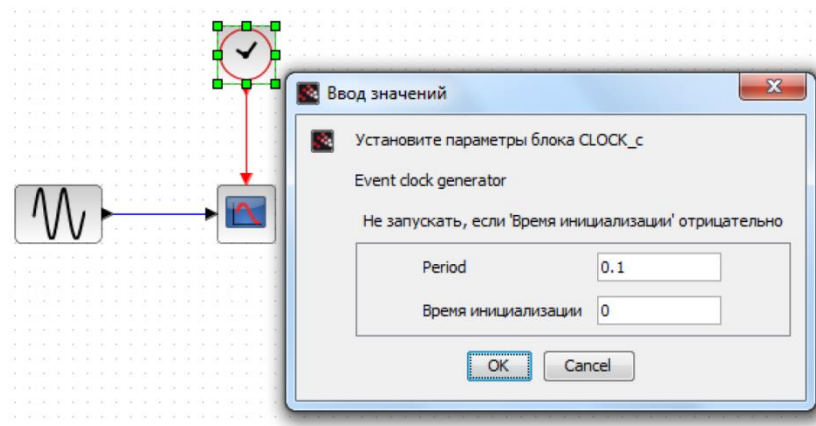


Рисунок 1.8 – Модель в Xcos

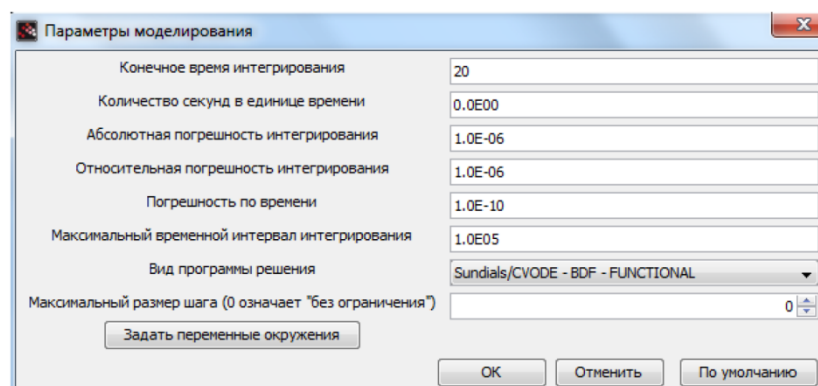


Рисунок 1.9 – Параметры моделирования

Если значение Period будет слишком велико, кривая на осциллографе будет состоять из видимых прямых отрезков, если слишком мало, то визуализация потребует больших вычислительных ресурсов. Пункт «Время инициализации» показывает, когда блок начнет работать, а осциллограф - отображать данные. Он необходим, если требуется пропустить начальное время моделирования. Соедините порты, как показано на рисунок 1.8. Осталось задать конечное время работы модели. Для этого выполните команду меню Моделирование → Установка. В появившемся окне задайте конечное время интегрирования равным 20, оставив остальные параметры по умолчанию (рисунок 1.9). Запустите моделирование, выполнив команду меню Моделирование → Запустить или просто нажав на треугольник на

панели инструментов. Если все выполнено корректно, то результат работы осциллографа будет отображен в появившемся графическом окне (рисунок 1.10).

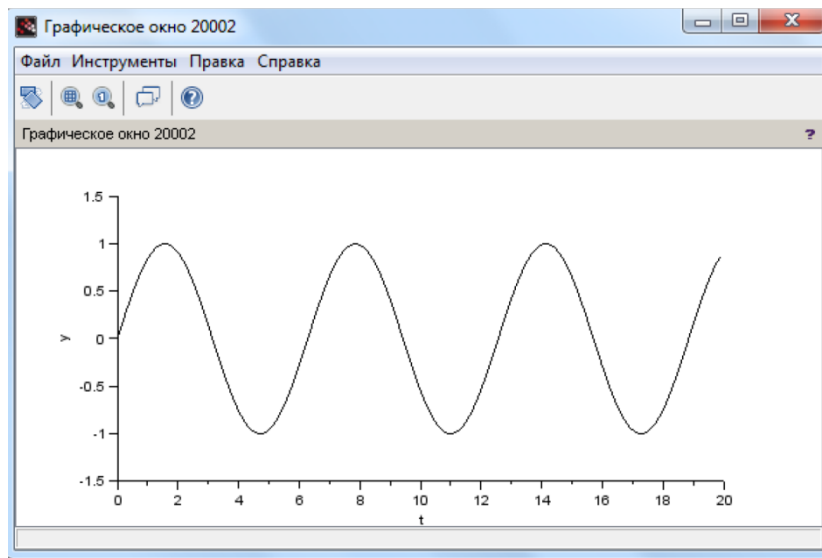


Рисунок 1.10 – Отображение сигнала в осциллографе

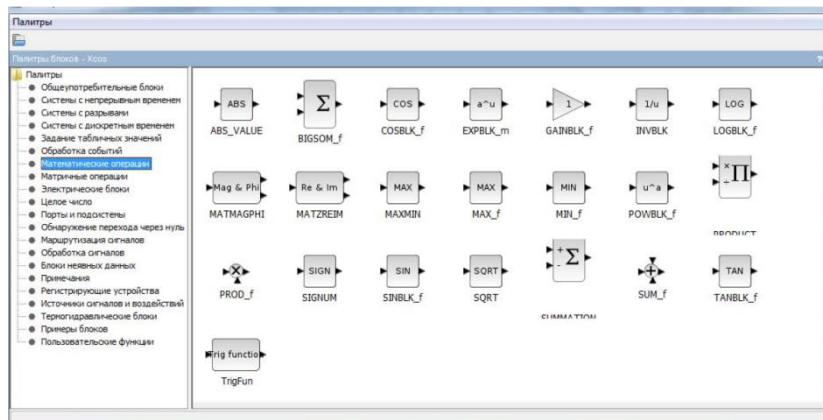


Рисунок 1.11 - Палитра «Математические операции»

Вернемся к функции $y(t) = \sin(t) + \sin(2t)e^{-0.1t}$. Получив от источника сигнала некий базовый сигнал, средствами Xcos можно выполнить с ним огромное количество действий, в том числе математических (рис. 1.11). Чтобы получить функцию $y(t) = \sin(t) + \sin(2t)e^{-0.1t}$, используем блок TIME_f из палитры источников. Умножим данные на -0.1 , используя блок GAINBLK_f (умножает

входную величину на заданный в настройках блока коэффициент). Затем получим e^n , используя математический блок EXPBLK_m (на вход поступает значение показателя степени, основание же задается в настройках, по умолчанию оно равно e). Перемножив синусоидальную функцию с экспоненциальной функцией с помощью блока PRODUCT, получим на осциллографе сигнал, изображенный на рисунок 1.12.

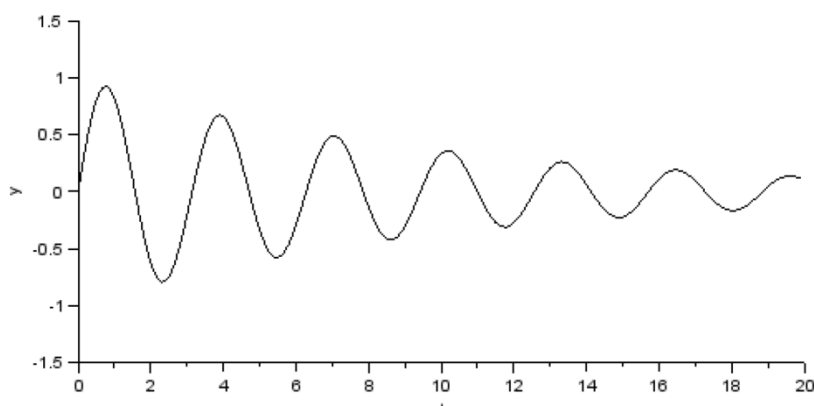


Рисунок 1.12 – Функция $\sin(2t)e^{-0.1t}$

Окончательная модель будет выглядеть, как показано на рисунок 1.13. Блоки PRODUCT и SUMMATION могут выполнять операции умножения и деления или сложения и вычитания в зависимости от настроек, а также для большего числа входов, чем два.

По каждому блоку можно получить справку и пример использования, вызвав соответствующую команду из контекстного меню. Чаще всего тип данных, которыми оперируют те или иные блоки, можно определить по графическому изображению портов. Подавляющее большинство оперирует так называемыми явными данными. Однако Xcos поддерживает несколько палитр с блоками специализированного назначения, например, с термогидравлическими или электрическими блоками. Кроме того, за счет дополнительных модулей можно существенно расширить палитры блоков, ориентированных на определенную область науки и техники. Рассмотрим палитру электрических блоков и простой пример их использования.

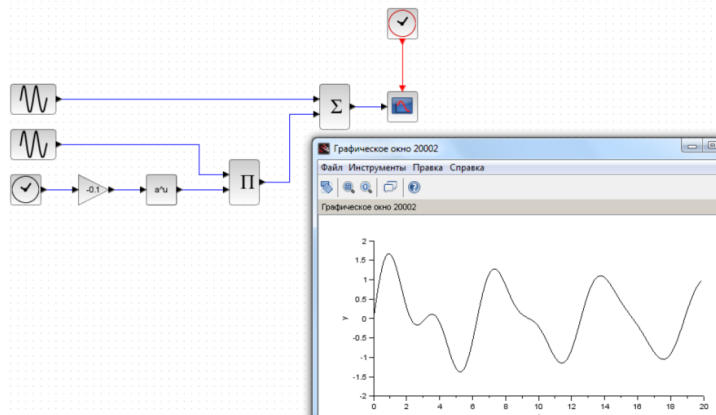


Рисунок 1.13 – Моделирование сигнала формы $y(t) = \sin(t) + \sin(2t)e^{-0.1t}$.

Электрическая цепь состоит из генератора прямоугольных импульсов, резистора, индуктивности и измерительных элементов (рисунок 1.14). Следует обратить внимание на полярности подключения элементов и на способ отображения временных диаграмм напряжения и тока на осциллографе.

В палитре электрических блоков имеются как управляемые, так и неуправляемые источники электрической энергии (тока, напряжения). Для моделирования сигнала произвольной формы можно использовать управляемый источник напряжения, как показано на рисунок 1.14. Управляется блок явными данными на соответствующем порте (треугольник). Если на этот порт поступает значение 1, значит между выводами «+» и «-» источника будет напряжение 1 В прямой полярности. В схеме источник прямоугольного напряжения (настройки по умолчанию) подключен к активно-индуктивной нагрузке (1 Ом, 0.05 Гн), происходит измерение тока, протекающего по цепи. Чтобы отобразить протекающий ток на осциллографе, его нужно виртуально измерить блоком Амперметр. Результатом измерения будут обычные данные, которые можно отобразить на осциллографе или над которыми можно выполнить математические или логические действия. Если через амперметр протекает ток 1 А, то на информационном выходе будет величина 1 и т. д. Чтобы на одном графике отобразить несколько сигналов, их можно объединить в шину блоком MUX.

Следует отметить необходимость наличия блока Земля в схеме, как это требуется во многих других программах схемотехнического моделирования. Без него будет выдана ошибка при попытке моделирования (в случае использования трансформатора, возможно, на обеих сторонах, как на рисунок 1.15).

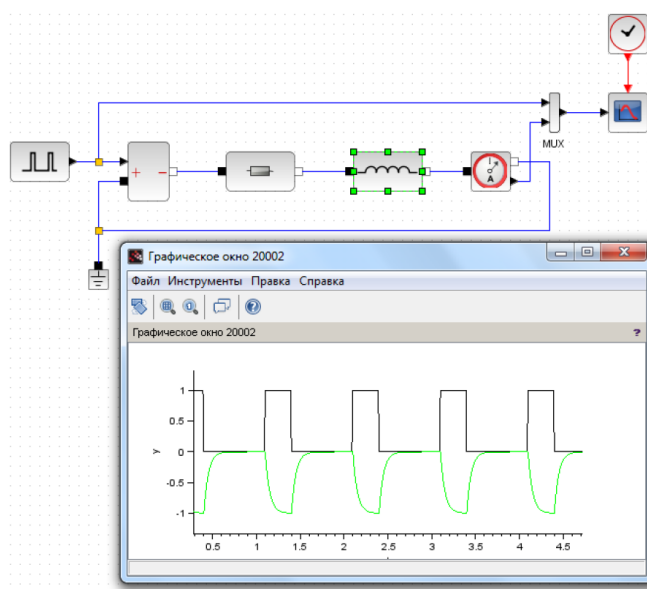


Рисунок 1.14 – Модель электрической схемы

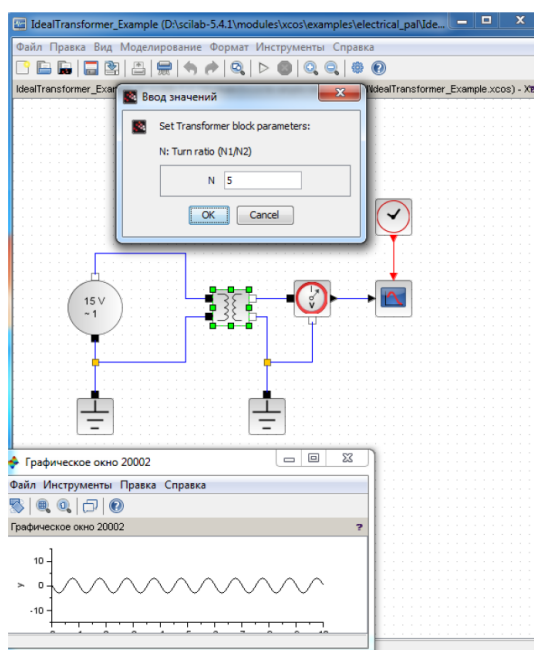


Рисунок 1.15 – Пример использования идеального трансформатора

Подробную информацию о построении модели, о возникающих в процессе

ошибках и предупреждениях, а также рекомендации по оптимизации модели можно увидеть в командном окне Scilab (рисунок 1.16).

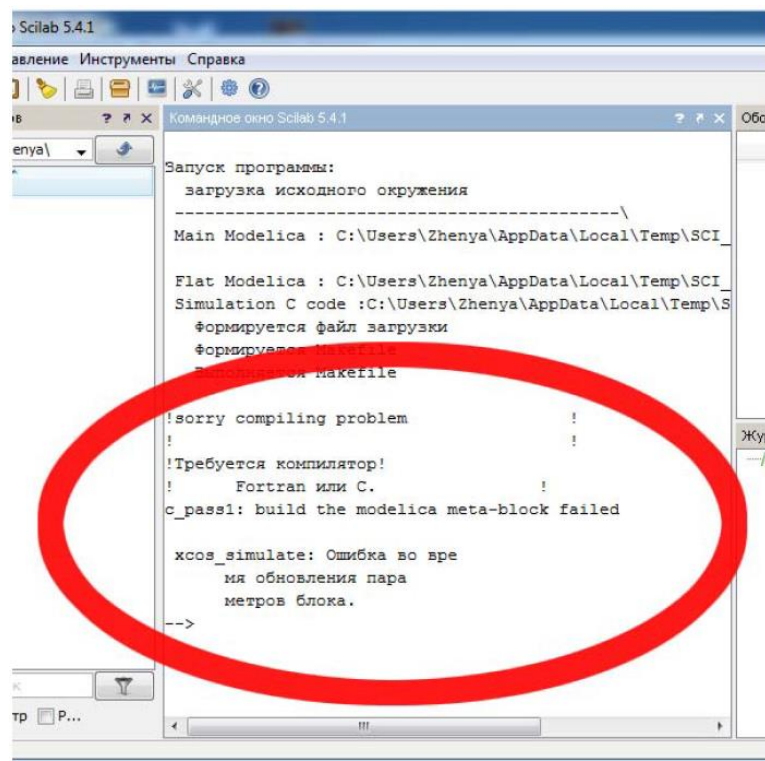


Рисунок 1.16 – Информация о процессе моделирования

Другой потенциальной ошибкой при попытке смоделировать данную схему может быть отсутствие установленного в системе компилятора языка Си. Он требуется для использования электрических блоков. Самым простым решением в этом случае будет использование бесплатного компилятора. Для его установки выполните команду меню Scilab «Инструменты» → Управление модулями Atoms. В появившемся окне выберите категорию Windows Tools и, например, компилятор LCC-windows (поддерживается только 32-битной версией Scilab), как показано на рисунок 1.17.

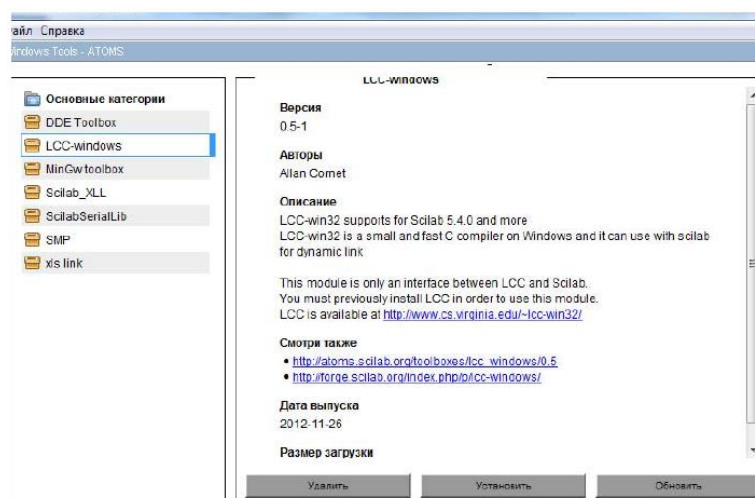


Рисунок 1.17 – Установка модуля поддержки компилятора LCC

В сведениях об устанавливаемом модуле имеются ссылки для скачивания компилятора. Для работы с электрическими блоками необходимо сначала скачать и установить компилятор (см. «LCC is available at...») и затем модуль его поддержки программой Scilab. После скачивания и установки компилятора нажмите кнопку «Установить» для инсталляции модуля связи. После перезапуска Scilab будут скомпилированы необходимые библиотеки, а схема должна успешно моделироваться.

Задание для лабораторной работы.

В программе Xcos составить одну из предложенных схем однофазного неуправляемого выпрямителя (рисунок 1.18). Действующее напряжение сети 220 В, частота 50 Гц, амплитуда выпрямленного напряжения – 12 В. Нагрузка выпрямителя – линейная, активная, сопротивлением 5 Ом. Смоделировать работу схемы, отобразить временные диаграммы без сглаживающего фильтра и с фильтром трех периодов сетевого напряжения, напряжения на вторичной обмотке (на вторичных обмотках) трансформатора, напряжения на диоде и нагрузке, тока нагрузки.

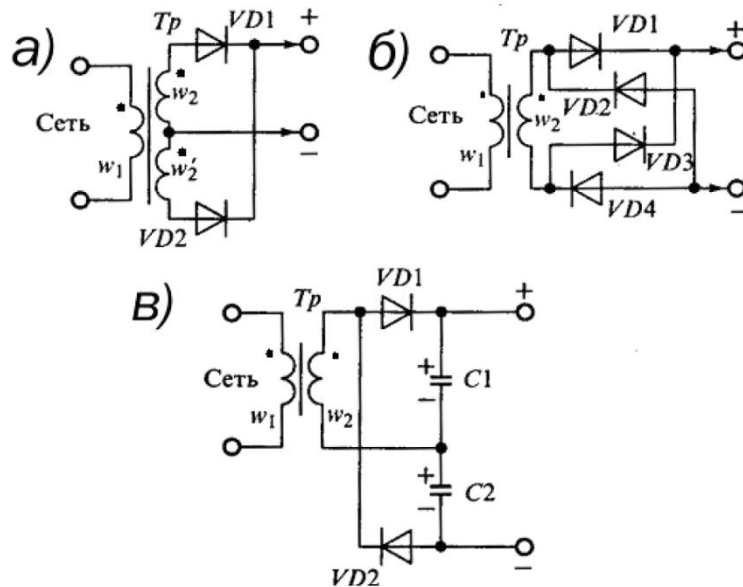


Рисунок 1.18 – Схемы однофазных выпрямителей

Вариант 1

Выпрямитель с нулевой точкой (рисунок 1.18, а)

В этом выпрямителе используются два диода с общей нагрузкой и две одинаковые вторичные обмотки трансформатора или одна со средней точкой. То есть схема представляет собой два простейших однополупериодных выпрямителя, имеющих два разных источника и общую нагрузку. В одном полупериоде переменного напряжения ток в нагрузку проходит с одной половины вторичной обмотки чрез один диод, в другом полупериоде – с другой половины обмотки через другой. Поэтому эта схема выпрямителя имеет в два раза меньше пульсации по сравнению с однополупериодной схемой выпрямления. Емкость сглаживающего конденсатора при одинаковом с однополупериодной схемой коэффициенте пульсаций может быть в два раза меньше. К сожалению, схема подразумевает более сложную конструкцию трансформатора и нерациональное использование в нем меди и стали.

Вариант 2

Мостовой выпрямитель (рисунок 1.18, б)

В данной схеме выпрямителя используется одна обмотка трансформатора при

выпрямлении обоих полупериодов переменного напряжения.

При выпрямлении положительного полупериода переменного напряжения ток проходит по следующей цепи: верхний вывод вторичной обмотки – диод VD1 – верхний вывод нагрузки – нагрузка – нижний вывод нагрузки – диод VD4 – нижний вывод вторичной обмотки – обмотка.

При выпрямлении отрицательного полупериода переменного напряжения ток проходит по следующей цепи: нижний вывод вторичной обмотки – диод VD3 – верхний вывод нагрузки – нагрузка – нижний вывод нагрузки – диод VD2 – верхний вывод вторичной обмотки – обмотка. Направление тока через нагрузку всегда одинаково.

По сравнению с однополупериодной схемой мостовая схема имеет в два раза меньший уровень пульсаций, более высокий КПД, более рациональное использование трансформатора и уменьшение его расчетной мощности. По сравнению с двухполупериодной схемой мостовая имеет более простую конструкцию трансформатора при таком же уровне пульсаций, а обратное напряжение вентиля может быть значительно ниже, чем в первых двух схемах. Но эти преимущества достигаются за счет увеличения числа диодов и необходимости их шунтирования для выравнивания обратного напряжения на каждом из них. Именно эта схема выпрямителя наиболее часто применяется в самых различных устройствах.

Вариант 3

Схема удвоения напряжения (рисунок 1.18, в)

В этой схеме в одном полупериоде переменного напряжения от вторичной обмотки трансформатора заряжается один конденсатор, а во втором полупериоде от той же обмотки – другой. Поскольку конденсаторы включены последовательно, то результирующее напряжение на обоих конденсаторах и, следовательно, на нагрузке в два раза выше, чем можно получить от той же вторичной обмотки в схеме с однополупериодным выпрямителем. Вторичную обмотку трансформатора можно рассчитывать на значительно меньшее напряжение, следствием чего являются

значительные токи через диоды выпрямителя, а уровень пульсаций значительно выше, чем в схемах двухполупериодных выпрямителей.

Отчет по лабораторной работе должен содержать:

- титульный лист;
- цель работы;
- программу работы;
- задание для лабораторной работы согласно варианту;
- схему имитационной модели выпрямителя;
- подробное текстовое пошаговое описание процесса разработки модели;
- временные диаграммы;
- выводы по результатам моделирования.

Вопросы для самоконтроля

1. Что такое имитационное моделирование?
2. Когда применяется имитационное моделирование?
3. Чем Xcos отличается от Scilab? От MATLAB/SIMULINK?
4. Каковы преимущества и недостатки Xcos в моделировании электронных схем в сравнении с Micro-Cap?
5. Что такое палитры блоков?
6. Какие порты данных применяются в блоках Xcos?
7. Для чего применяется блок CLOCK c?
8. Как изменить цвет кривой, отображаемой в осциллографе?
9. Опишите работу представленных схем выпрямителей.
10. Каким блоком можно смоделировать трансформатор в схеме выпрямителя, представленной на рис. 1.18, а?
11. Как отобразить несколько кривых на одном осциллографе?
12. Как можно изменить масштаб кривой тока для сравнения на осциллографе с кривой напряжения?

3.3 Лабораторная работа № 3 Представление данных. Вычисление статистических параметров.

Цели работы: получение навыков работы с массивами данных в системе MATLAB; вычисление статистических параметров массивов и исследование статистической связи между массивами.

Основные положения

В дальнейшем будут использоваться два вида данных: искусственные статистические данные, генерируемые системой MATLAB, и реальные данные, также извлекаемые из пакета MATLAB.

Статистические данные

Пример 1.1

Данные (по 10 реализаций каждого класса) по трем классам генерируются с помощью функции генерации из нормального закона $R = \text{normrnd}(\mu, \text{SIGMA}, m, n)$, где μ – среднее, SIGMA – среднеквадратичное отклонение (СКО), m, n – количества строк и столбцов матрицы R . Параметры распределений по трем классам назначим так, как показано в таблице 1.1.

Таблица 1.1 – Параметры распределений

Класс	Среднее значение				СКО
	1-й признак	2-й признак	3-й признак	4-й признак	
1-й класс, А	$m = 5$	$m = 4$	$m = 7$	$m = 2$	$\delta = 1$
2-й класс, В	$m = 3$	$m = 5$	$m = 4$	$m = 3$	$\delta = 2$
3-й класс, С	$m = 7$	$m = 6$	$m = 3$	$m = 2$	$\delta = 1$

Могут использоваться и другие генераторы случайных чисел, например

$\text{randn}(n,m)$, в результате которого выдается матрица размерностью $(n \times m)$, в каждом столбце которой генерируются отсчеты нормального распределения с нулевым средним и единичной дисперсией.

1-й класс формируется следующим образом:

<code>all=normrnd(5,1,10,1);</code>	%	генерация	столбца	1-го	признака
<code>al2=normrnd(4,1,10,1);</code>	%	генерация	столбца	2-го	признака
<code>al3=normrnd(7,1,10,1);</code>	%	генерация	столбца	3-го	признака

```
al4=normrnd(2,1,10,1); % генерация столбца 4-го признака al=[a1 a12 a13
a14] % матрица 10 x 4 первого класса
```

Аналогично для 2-го и 3-го классов формируются матрицы a2 и a3.

```
a2l=normrnd(3,2,10,1);
a22=normrnd(5,2,10,1);
a23=normrnd(4,2,10,1);
a24=normrnd(3,2,10,1);
a2=[a21 a22 a23 a24]; % матрица 10 x 4 второго класса
a31=normrnd(7,1,10,1);
a32=normrnd(6,1,10,1);
a33=normrnd(3,1,10,1);
a34=normrnd(2,1,10,1);
a3=[a31 a32 a33 a34]; % матрица 10 x 4 третьего класса
```

Объединив все матрицы в одну a=[a1;a2;a3], получим результирующую матрицу a размерностью 30 x 4, первые 10 строк которой описывают 1-й класс, вторые - 2-й класс и третьи 3-й, причем все объекты описываются по 4-м признакам:

```
a= [ a1; a2 ; a3 ] ;
```

Теперь получим групповую матрицу d, в которой запишем соответствующие наименования классов для каждого наблюдения. Классов всего три: A, B, C.

```
gg ('A'; 'A'; 'A'; 'A'; 'A'; 'A'; 'A'; 'A'; 'A'; 'A'; 'B'; 'B'; 'B'; 'B'; 'B'; 'B'; 'B'; 'B'; 'B'; 'B'; 'B'; 'B';
'C'; 'C'; 'C'; 'C'; 'C'; 'C'; 'C'; 'C'; 'C'; 'C'; 'C'];
```

```
% gg – символьная матрица
```

```
g=cellstr(gg);
```

```
% cellstr – функция, преобразующая символьную матрицу в матрицу строк
```

```
% g – матрица элементов (строк); ее размерность равна 30 x 1;
```

```
% первые 10 строк матрицы g равны 'A', вторые – 'B', третьи 'C'
```

Получение экспериментальных данных из пакета MATLAB («Ирисы Фишера»)

Пример 1.2

Ирисы Фишера – это набор данных для задачи классификации, на примере которого Рональд Фишер (английский ученый, один из основоположников математической генетики) в 1936 г. продемонстрировал работу разработанного им метода дискриминантного анализа. Ирисы Фишера состоят из данных о 150

экземплярах ириса, по 50 экземпляров трёх видов - ириса щетилистого, ириса виргинского и ириса разноцветного. Для каждого экземпляра измерялись четыре характеристики: длина чашелистика; ширина чашелистика; длина лепестка; ширина лепестка. Данные по ирисам Фишера вошли в состав системы MATLAB и могут использоваться для иллюстраций работы алгоритмов классификации. В матрицах системы MATLAB, содержащих статистические данные, строки используются для представления наблюдений, а столбцы - для представления измеренных переменных.

Строка `load fisheriris %` Данные по ирисам Фишера загружает переменную измерений (`meas`) и переменную видов (`species`) в рабочую область MATLAB. Переменная измерений представляет собой числовую матрицу размерностью 150 x 4, в которой представлены 150 наблюдений 4 различных измеренных переменных (по столбцам: длина чашелистика, ширина чашелистика, длина лепестка, ширина лепестка соответственно). Наблюдения в измерениях принадлежат трем различным видам ирисов: щетилистый (`setosa`), разноцветный (`versicolor`) и виргинский (`virginica`). Данные по видам могут быть разделены один от другого с использованием строковой матрицы видов - `species` (150 x 1)

```
setosa_indices = strcmp('setosa',species);
```

```
setosa = meas(setosa_indices,:);
```

Результирующая переменная `setosa` представляет собой матрицу (50 x 4) 50 наблюдений 4 измеренных переменных для ирисов `setosa`. Чтобы получить и отобразить первые 5 наблюдений данных по `setosa`, используем поясняющее индексирование:

```
SetosaObs=setosa(1:5, :)
```

```
SetosaObs=
```

5.1000	3.50	1.	0
4.9000	3.00	1.	0
4.7000	3.20	1.	0
4.6000	3.10	1.	0
5.0000	3.60	1.	0

Эти данные организованы в таблицу с подразумеваемыми названиями колонок «Длина чашелистика», «Ширина чашелистика», «Длина лепестка» и «Ширина лепестка». Подразумеваемые названия строк - это «Наблюдение 1», «Наблюдение 2», «Наблюдение 3» и т. д.

Аналогично, по 50 наблюдений по ирисам разноцветным – *versicolor*, и ирисам виргинским – *virginica*, могут быть выделены из контейнерной переменной измерений (*meas*):

```
versicolor_indices = strcmp('versicolorspecies');  
versicolor = meas(versicolor_indices,:);  
virginica_indices = strcmp('virginicaspecies');  
virginica = meas(virginica_indices,:);
```

Так как множества данных для этих трех видов имеют один и тот же размер, они могут быть переформатированы в один многомерный массив *iris* с размерностью 50 x 4 x 3.

```
iris = cat(3,setosa,versicolor,virginica);
```

Полученный массив *iris* по ирисам представляет собой трехслойную таблицу с одинаковыми подразумеваемыми заголовками строк и столбцов: массивы *setosa*, *versicolor* и *virginica*. Вдоль третьей размерности - подразумеваемые имена слоев «Setosa», «Versicolor» и «Virginica». Чтобы получить и отобразить данные в многомерном массиве, надо использовать поясняющую индексацию, как и для двумерных массивов. Получим первые 5 наблюдений признака длина лепестка (*sepal lengths*) в данных по *setosa*:

```
SetosaSL = iris (1:5,1,1)  
SetosaSL =  
5.1000 4.9000 4.7000 4.6000 5.0000
```

Многомерные массивы обеспечивают естественный путь к организации числовых данных, для которых наблюдения или конструкции экспериментов имеют много размерностей. Если бы, например, данные со структурой ирисов были бы собраны многими наблюдателями в множестве мест в множестве разных дат, то полные данные могли бы быть организованы в единственный многомерный массив

с размерностями для «Наблюдатель», «Местоположение» и «Дата». Подобно этому экспериментальный план для m наблюдений и p p -мерных переменных может быть запомнен как массив $m \times p$.

Задание

Освоить генерацию и ввод матриц данных в системе MATLAB. Освоить способы расчета статистических параметров массива данных.

Освоить основы расчета статистической связи между массивами данных.

Порядок выполнения работы.

Запустите систему MATLAB. В главном окне (в поле Current Directory, расположенном в верхнем правом углу) установите путь к рабочей папке.

Создайте 2 массива из 10 случайных чисел каждый по нормальному закону с параметрами, представленными в таблице 1.2

Таблица 1.2 – Параметры

Массив	V1	V2	V3	V4	V5	V6	V7	V8	V9	V10
1	$m = 2$ $\sigma = 1$	$m = 3$ $\sigma = 1$	$m = 4$ $\sigma = 1$	$m = 5$ $\sigma = 2$	$m = 6$ $\sigma = 2$	$m = 3$ $\sigma = 2$	$m = 5$ $\sigma = 1$	$m = 6$ $\sigma = 1$	$m = 7$ $\sigma = 2$	$m = 8$ $\sigma = 3$
2	$m = 6$ $\sigma = 2$	$m = 8$ $\sigma = 4$	$m = 7$ $\sigma = 3$	$m = 2$ $\sigma = 1$	$m = 2$ $\sigma = 1$	$m = 7$ $\sigma = 4$	$m = 2$ $\sigma = 1$	$m = 7$ $\sigma = 1$	$m = 3$ $\sigma = 2$	$m = 4$ $\sigma = 2$

Вычислите выборочные оценки среднего значения и среднеквадратического отклонения каждого массива.

Вычислите коэффициент корреляции между вашими двумя массивами.

Вычислите выборочные статистические параметры двух первых признаков ирисов «Setosa» и коэффициент корреляции между ними.

Содержание отчета:

- название, цель и задачи работы;
- тексты программ, которые требовалось сохранять в ходе выполнения работы;
- объяснение полученных результатов и выводы.

3.4 Лабораторная работа № 4 Автоматизация выполнения расчетов при обработке результатов испытаний

Цель работы: закрепить навыки использования MATLAB при проведении метрологической обработки результатов измерений.

Задание:

- написать программу на языке MATLAB, предназначенную для обработки результатов испытаний;
- провести метрологическую обработку экспериментальных данных.

Методические указания.

При проведении серии опытов, результатами которых является набор результатов измерений, актуальной задачей является оценка погрешности измерений. Для расчётов погрешностей существуют строгие алгоритмы обработки результатов испытаний, которые можно запрограммировать на любом из языков программирования. Это позволит не только ускорить обработку результатов испытаний, но и снизить вероятность возникновения ошибок, неизбежных при обработке большого объёма экспериментальных данных, уменьшить трудозатраты на расчёты и, как следствие, повысить производительность труда.

Сформулируем задачу и рассмотрим алгоритм её решения.

Пусть в результате серии из n экспериментов по измерению физической величины y был получен набор её возможных значений y_i , $i = \overline{1, n}$. Требуется определить доверительный интервал, в котором с заданной вероятностью лежит измеренная величина y , а также относительную погрешность её измерения.

Алгоритм решения поставленной задачи заключается в следующем.

Вычисляем среднее арифметическое результатов измерений по формуле

$$\bar{y} = \sum_{i=1}^n \frac{y_i}{n}$$

Вычисляем среднее квадратичное отклонение результатов измерений по формуле

$$S_y = \sqrt{\sum_{i=1}^n \frac{(y_i - \bar{y})^2}{n-1}}$$

Вычисляем среднее квадратичное отклонение среднего арифметического результатов измерений по формуле $S_{\bar{y}} = S_y / \sqrt{n}$.

Определяем границы доверительного интервала среднего арифметического результатов измерений по формуле $t_p \cdot S_{\bar{y}}$, где t_p – коэффициент Стьюдента, определяемый по таблицам при заданной доверительной вероятности.

Результаты измерений представляем в виде $y = \bar{y} \pm t_p \cdot S_{\bar{y}}$.

Программа, предназначенная для метрологической обработки результатов испытаний, должна отвечать следующим требованиям.

Исходные данные (экспериментальные данные) должны загружаться из текстового файла.

Вначале программы с клавиатуры, в зависимости от числа экспериментальных данных, вводится значение коэффициента Стьюдента.

Для ввода с клавиатуры значения переменной t_p в MATLAB можно использовать выражение $t_p = \text{input}(\text{'Введите } t_p \text{'})$.

Варианты заданий для самостоятельной работы

Сгенерируйте исходные данные, воспользовавшись следующим выражением MATLAB

$$y = a + \text{sqrt}(b) \cdot \text{randn}(c, 1).$$

2.1 Значения параметров a, b, c по вариантам

Вариант	Значения параметров			Вариант	Значения параметров		
	a	b	c		a	b	c
1	10	1	20	9	15	3	25
2	5	0,3	30	10	65	8	35

3	100	5	50	11	4	0,5	45
4	200	10	60	12	90	10	55
5	0,9	0,03	70	13	35	7	65
6	1,5	0,4	80	14	0,7	0,1	75
7	50	2	90	15	0,3	0,05	85
8	1000	45	100	16	76	8	95

Результаты выполнения лабораторной работы

$\bar{y}$	S_y	$S_{\bar{y}}$	t_p	$t_p \cdot S_{\bar{y}}$

Контрольные вопросы.

1. Привести функцию MATLAB для вычисления среднего арифметического значения результатов измерений.
2. Для чего применяется функция randn() в MATLAB.
3. Привести функцию MATLAB для вычисления среднего квадратичного отклонения результатов измерений.
4. Построить график рас
пределения случайной величины y в соответствии с вариантом.
5. Как проверить гипотезу о нормальности распределения результатов измерений?

4 Вопросы к зачету по курсу «Компьютерные технологии»

1. Общие сведения о MATLAB'e.
2. Ввод чисел.
3. Ввод векторов.
4. Ввод матриц.
5. Некоторые специальные матрицы.
6. Некоторые простые команды.
7. Выдачи на экран. Команда `format` с различными опциями.
8. Текстовые переменные.
9. Контроль за переменными.
10. Другие типы переменных - ячейки и структуры.
11. Система `help`.
12. Элементы XY-графики.
13. Как открывать графическое окно?
14. Автоматическое чередование цветов.
15. Графики в полярных координатах.
16. Управление осями.
17. Простые примеры, иллюстрирующие эффективность MATLAB'a.
18. Суммирование.
19. Произведения.
20. Логические задачи.
21. Графический способ решения уравнений.
22. Полиномы.
23. Системы линейных алгебраических уравнений.
24. Состав библиотеки «SimPowerSystems».
25. Использование среды SIMUTINK для моделирования.
26. Создание модели среде SIMUTINK.
27. Построение модели электрической цепи в пакете Simulink.
28. Функции `save` и `load`.
29. Функции `fwrite` и `fread`.

30. Функции fscanf и fprintf:
31. Функции imread и imwrite.
32. Что означает в MatLab режим прямых вычислений?
33. Какие переменные относятся к системным переменным MatLab?
34. Как обозначается в MatLab символьная константа?
35. Сколько символов в MatLab может содержать имя переменной? Какие символы могут содержаться в имени переменной MatLab?
36. Какая команда MatLab сохраняет рабочую среду? Какая команда в MatLab сохраняет результаты работы в журнале сессии?
37. Чем могут отделяться в MatLab элементы одной строки матрицы при ее вводе?
Чем могут отделяться в MatLab строки матрицы при ее вводе?
38. Что будет создано при вводе в MatLab оператора $x=[1 : 0.2 : 2]$?
39. Что будет создано при вводе в MatLab оператора $W1=W(3 : 5)$?
40. Что будет создано при вводе в MatLab оператора $P1=P(2, :)$?
41. Какой оператор относится к почленному умножению матриц в MatLab?
42. Какие существуют типы М-файлов в MatLab?
43. Какая функция MatLab предусмотрена для построения двухмерного графика одной функции?
44. Какая функция MatLab предусмотрена для построения двухмерного графика двух функций?
45. Какая функция MatLab предусмотрена для построения трехмерного графика?
46. Какая функция MatLab предусмотрена для построения анимированного графика?
47. Какая функция MatLab позволяет разбить графическое окно на несколько подграфиков?
48. Какие операторы MatLab используются для построения цикла с использованием счетчика циклов?
49. Какая библиотека Simulink содержит источники сигналов?
50. Какая библиотека Simulink содержит регистрирующие виртуальные приборы?

- 51.Какая библиотека Simulink содержит блоки дифференцирования и интегрирования?
- 52.Какая библиотека Simulink содержит блоки вычисления элементарных функций?
- 53.Сколько значений параметров шага устанавливается в Simulink при переменном шаге моделирования?
- 54.Сколько значений параметров шага устанавливается в Simulink при постоянном шаге моделирования?
- 55.Как в MatLab и Simulink устанавливается машинная бесконечность?
- 56.Как изменить на осциллограмме в Simulink максимальное и минимальное значения по оси Y?
- 57.Для чего служит в Simulink специализированное дополнение Power System Blockset?
- 58.Сколько типов источников электроэнергии содержит библиотека «Electrical Sources» дополнения Power System Blockset пакета Simulink?
- 59.Что содержит библиотека «Connectors» дополнения Power System Blockset пакета Simulink?
- 60.Какие компоненты относятся к библиотеке «Elements» дополнения Power System Blockset пакета Simulink?
- 61.В режиме прямых вычислений в Matlab выполнить следующие действия над матрицей: транспонировать, инвертировать, найти определитель.
- 62.В режиме прямых вычислений в Matlab выполнить следующие действия над матрицей: транспонировать, возвести в квадрат, выполнить поэлементное возведение в квадрат.
- 63.Создать и запустить в MatLab на выполнение М-файл программы построения анимированного графика синусоидальной функции.
- 64.Создать и запустить в MatLab на выполнение М-файл программы построения анимированного графика синусоидальной функции.
- 65.Составить М-файл программы построения трехмерного графика функции.
- 66.Составить в MatLab файл-функцию синусоидальной функции в М-файле.

67. Описать на примере как осуществляется связь MatLab с текстовым редактором Word.
68. Описать на примере как осуществляется связь MatLab с электронными таблицами Excel.
69. С помощью пакета Simulink создать модель, содержащую виртуальный осциллограф (Scope) тремя входами и три подключенных к нему блоков: Pulse Generator, Ramp, RepeatingSequence.
70. С помощью пакета Simulink создать модель, содержащую виртуальный осциллограф (Scope) тремя входами и три подключенных к нему блоков: RepeatingSequence, Sine Wave, Step.
71. Создать модель, состоящую из трех блоков: блок синусоидального воздействия → дифференцирующий блок → блок ограничения. К выходу каждого блока подключить осциллограф.

СПИСОК РЕКОМЕНДУЕМОЙ ДЛЯ ИЗУЧЕНИЯ ЛИТЕРАТУРЫ

Основная литература:

- 1 Цифровая обработка сигналов и MATLAB: учеб. пособие / А. И.Солонина, Д. М. Клионский, Т. В. Меркучева, С. Н. Перов. – СПб.: БХВ-Петербург, 2013. – 512 с.
- 2 Глибин, Е.С. Моделирование источников питания электротехно- логических установок в Scilab Xcos : лабораторный практикум / Е.С. Глибин. – Тольятти : Изд-во ТГУ, 2016.
- 3 Системы автоматизированного расчёта в управлении качеством и при защите информации : лабораторные работы / сост. :П.В. Балабанов, С.В. Пономарёв. – Тамбов : Изд-во Тамб.гос. техн. ун-та, 2009. – 32 с.
- 4 Обработка данных в среде MATLAB: Методические указания к практическим и лабораторным занятиям / Сост.: А. П. Немирко, Л. А. Манило. СПб.: Изд-воСПбГЭТУ «ЛЭТИ», 2013. 35 с.
- 5 Шутов, Е.А. Компьютерные технологии решения задач электроснабжения: учебное пособие / Е.А. Шутов, Д.Е. Бабинович. – Электрон. дан. – Томск : ТПУ, 2013. – 104 с.
- 6 Гринев, А. Ю. Основы электродинамики с Matlab: учеб. пособие / А. Ю. Гринев, Е. В. Ильин. – М.: Логос, 2012. – 176 с. – ISBN 978-5-98704-700-2.
- 7 Борисевич, А. В. Теория автоматического управления: элементарное введение с применением MATLAB / А. В. Борисевич. – М.: Инфра-М, 2014. – 200 с. – ISBN 978-5-16-101828-6.
- 8 Моделирование систем управления с применением Matlab: Учебное пособие / Тимохин А.Н., Румянцев Ю.Д; Под ред. А.Н.Тимохина – М.: НИЦ ИНФРА-М, 2016. – 256 с.: 60х90 1/16. – (Высшее образование: Бакалавриат) (Переплёт) ISBN 978-5-16-010185-9 .

Дополнительная литература:

1 MATLAB, Simulink и SimPowerSystems в электроэнергетике: учебное пособие для студентов, обучающихся по направлению подготовки 140400.62 "Электроэнергетика и электротехника", профиль "Электроснабжение" / Джендубаев А. -З. Р., Алиев И. И. – Черкесск: БИЦ СевКавГГТА, 2014. – 136 с.

2 Плохотников, К.Э. Методы разработки математических моделей и вычислительный эксперимент на базе пакета Matlab: курс лекций / К.Э. Плохотников. – М. : СОЛОН-Пр., 2017. – 628 с. – (Библиотека студента). - ISBN 978-5-91359-211-8

3 Теория электрических цепей: учебное пособие / В.И. Никулин. – М.: ИЦ РИОР: НИЦ Инфра–М, 2013. – 240 с.: 60х90 1/16. – (Высшее образование: Бакалавриат).

РЕКОМЕНДУЕМЫЕ ИНФОРМАЦИОННЫЕ СПРАВОЧНЫЕ СИСТЕМЫ

1. Электронно-библиотечная система «Издательства «Лань»: [Электронный ресурс]. – URL : <https://e.lanbook.com/> (Дата обращения 13.04.2018).
2. Электронно-библиотечная система znanium.com. ООО «Научно-издательский центр Инфра-М»: [Электронный ресурс]. – URL: <https://znanium.com/>. (Дата обращения 13.04.2018).
3. Электронно-библиотечная система образовательных и просветительских изданий: [Электронный ресурс]. – URL: <https://www.iqlib.ru>. (Дата обращения 13.04.2018).
4. Российское образование. Федеральный портал: [Электронный ресурс]. – URL: <https://www.edu.ru/modules>. (Дата обращения 13.04.2018).
5. Единое окно доступа к образовательным ресурсам: Информационная система: [Электронный ресурс]. – URL: <https://window.edu.ru>. (Дата обращения 13.04.2018).
6. Scilab [Электронный ресурс]. — URL: <https://www.scilab.org>(Дата обращения 13.04.2018).
7. Преподавание, наука и жизнь. [Электронный ресурс]. — URL: <https://kpolyakov.spb.ru/>(Дата обращения 13.04.2018).