

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ
УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«РОССИЙСКИЙ ГОСУДАРСТВЕННЫЙ
ПЕДАГОГИЧЕСКИЙ УНИВЕРСИТЕТ им. А. И. ГЕРЦЕНА»
институт информационных технологий и технологического образования
кафедра информационных технологий и электронного обучения

Основная профессиональная образовательная программа
Направление подготовки 09.03.01 Информатика и вычислительная техника
Направленность (профиль) «Технологии разработки программного обеспечения»
форма обучения – очная

Курсовая работа

по дисциплине «Технологии компьютерного моделирования»

Использование системы компьютерной математики Scilab для решения задач
компьютерного моделирования

Обучающегося 2 курса
Воложанина В.О.

Руководитель:
д.п.н, профессор
Власова Е. З.

«_____» _____ 2023 г.

Санкт-Петербург
2023

ОГЛАВЛЕНИЕ

Введение	3
1. Теоретическая часть.....	4
1.1 Введение в Scilab.....	4
1.2 Компьютерное моделирование Основные понятия в Scilab.....	5
1.3 Использование Scilab для статистического анализа данных.....	6
1.4 Компьютерное моделирование физических систем в Scilab.....	7
1.5 Функции Scilab используемые для моделирования.	9
1.6 Функции построение и сохранение графиков.....	12
1.7 Экспорт данных.....	14
2. Практическая часть.....	15
2.1 Оформление графика при помощи функции plot.....	15
2.2 Оформление графика при помощи функции plot2d.....	16
2.3 Оформление графика при помощи функции plot3d.....	16
2.4 Оформление графика при помощи функции scatter3d.....	17
2.5 Модель Лотки – Вольтерры взаимодействия двух видов.....	18
Заключение	21
Литература	22

ВВЕДЕНИЕ

Данная курсовая работа направлена на изучение использования системы компьютерной математики Scilab для решения задач компьютерного моделирования. Цель данной курсовой работы - продемонстрировать, как Scilab возможно эффективно использовать для компьютерного моделирования.

Обоснование актуальности курсовой работы:

В современном мире компьютерное моделирование играет ключевую роль во многих научных, инженерных и прикладных областях. Оно позволяет создавать и анализировать математические модели сложных систем, предсказывать их поведение и принимать обоснованные решения. Таким образом, использование Scilab, мощной системы компьютерной математики, для решения задач компьютерного моделирования является актуальным и важным направлением исследования.

Объектом исследования являются модели. Предмет исследования – система компьютерной алгебры.

Основная часть курсовой работы состоит из двух глав: теоретический – блок теории, в которой дается обзор принципов компьютерного моделирования, методов математического моделирования и введение в Scilab как мощный вычислительный инструмент. Практический – блок посвящен практическому применению Scilab при решении задач компьютерного моделирования.

1. Теоретическая часть

1.1 Введение в Scilab.

Scilab - это система компьютерной математики, предназначенная для выполнения различных вычислений, моделирования и анализа данных. Она обладает широким набором функций и инструментов, которые позволяют решать разнообразные математические задачи.

Основные функции и возможности Scilab включают:

1. Математические операции: Scilab поддерживает обширный набор математических операций, включая арифметические операции (сложение, вычитание, умножение, деление), возведение в степень, извлечение корня, тригонометрические функции, логарифмы и экспоненты, работы с комплексными числами и многое другое.

2. Векторы и матрицы: Scilab предоставляет мощные инструменты для работы с векторами и матрицами. Вы можете создавать, изменять и оперировать векторами и матрицами различных размерностей. Scilab также поддерживает различные операции с матрицами, включая умножение, транспонирование, определение, инверсию и разложение матриц.

3. Графики и визуализация: Scilab обладает возможностями построения разнообразных графиков и визуализации данных. Вы можете создавать двумерные и трехмерные графики, включая линейные графики, графики функций, диаграммы рассеяния, гистограммы и другие типы графиков. Scilab предоставляет множество параметров для настройки внешнего вида графиков.

4. Работа с файлами и данными: Scilab позволяет импортировать и экспортировать данные из различных форматов файлов, включая текстовые файлы, CSV, Excel и другие. Вы можете считывать данные из файлов, обрабатывать их и сохранять результаты в удобном формате. Scilab также поддерживает работу с базами данных и другими источниками данных.

5. Численные методы и решение уравнений: Scilab предоставляет множество численных методов для решения математических задач. Вы можете решать системы линейных и нелинейных уравнений, дифференциальные уравнения, оптимизационные задачи, интегрирование и другие задачи численного анализа. Scilab также поддерживает символьные вычисления для аналитического решения уравнений.

6. Программирование и скриптинг: Scilab предоставляет средства для программирования и скриптинга. Вы можете создавать собственные пользовательские функции, скрипты и программы на языке Scilab. Система имеет понятный синтаксис и поддерживает основные элементы программирования,

такие как условные операторы, циклы, функции, области видимости переменных и другие.

1.2 Компьютерное моделирование. Основные понятия в Scilab

Компьютерное моделирование в Scilab включает в себя несколько базовых концепций, которые являются фундаментальными для процесса представления и моделирования систем реального мира. Вот несколько ключевых концепций компьютерного моделирования с использованием Scilab:

1. Переменные: Переменные представляют величины или свойства, представляющие интерес в модели. Это могут быть числовые значения, массивы или матрицы, которые хранят данные и изменяются со временем в процессе моделирования.

2. Уравнения: Уравнения определяют математические соотношения между переменными в модели. Они выражают, как переменные изменяются и взаимодействуют друг с другом. Уравнения могут включать в себя дифференциальные уравнения, разностные уравнения или другие математические выражения.

3. Функции: Функции в Scilab инкапсулируют наборы операций, которые могут быть повторно использованы во всей модели. Функции позволяют создавать модульный и организованный код, что делает реализацию модели более эффективной и управляемой.

4. Параметры: Параметры - это значения, которые определяют характеристики или свойства моделируемой системы. Обычно это фиксированные значения, которые влияют на поведение модели, но не изменяются во время моделирования. Параметрам могут быть присвоены конкретные значения на основе эмпирических данных или экспертных знаний.

5. Начальные условия: Начальные условия представляют собой начальные значения переменных в начале моделирования. Они определяют начальное состояние системы и необходимы модели для точного прогнозирования последующего поведения системы.

6. Время моделирования: время моделирования относится к промежутку времени, в течение которого моделируется модель. Он определяет продолжительность моделирования и определяет, насколько далеко в будущем модель предсказывает поведение системы.

7. Численные методы: Scilab предоставляет различные численные методы для решения математических уравнений в модели. Эти методы приближают решения дифференциальных уравнений или других математических конструкций и позволяют моделировать динамику системы.

8. Визуализация: Визуализация включает в себя представление результатов моделирования в графическом или числовом форматах. Scilab предоставляет инструменты для визуализации данных, позволяющие разработчикам моделей эффективно анализировать полученные результаты и сообщать о них.

Понимание этих базовых концепций в Scilab помогает эффективно разрабатывать, анализировать и интерпретировать компьютерные модели. Применяя эти концепции, есть возможность представлять и моделировать сложные системы, получать представление об их поведении и принимать обоснованные решения, основанные на предсказаниях модели.

1.3 Использование Scilab для статистического анализа данных.

Использование Scilab для статистического анализа данных предоставляет широкий набор функций и инструментов для обработки, анализа и визуализации данных. Ниже представлены некоторые основные функции и подходы, которые можно использовать при статистическом анализе данных в Scilab:

Загрузка и предварительная обработка данных:

1. Загрузка данных: Scilab предлагает функции для чтения и загрузки данных из различных форматов файлов, таких как текстовые файлы, CSV, Excel и другие. Некоторые из этих функций включают `mgetl`, `csvRead`, `xlsRead`, которые позволяют загружать данные в Scilab.

2. Обработка данных: Scilab предоставляет функции для выполнения различных операций над данными, таких как фильтрация, сортировка, объединение и разделение данных. Некоторые функции для обработки данных включают `filter`, `sort`, `concat`, `submatrix`.

Расчет статистических параметров:

1. Среднее значение и медиана: Scilab предлагает функции для расчета среднего значения и медианы набора данных. Функции `mean`, `median` позволяют вычислять эти статистические параметры.

2. Дисперсия и стандартное отклонение: Scilab также предоставляет функции для расчета дисперсии и стандартного отклонения данных. Функции `variance`, `stdev` позволяют вычислить эти статистические параметры.

3. Квантили: Scilab позволяет рассчитывать квантили набора данных. Функция `quantile` позволяет определить значение, ниже которого попадает указанный процент данных.

Построение диаграмм и визуализация данных:

1. Гистограммы: с помощью функции `histogram` в Scilab можно строить гистограммы, которые позволяют визуально представить распределение данных по различным интервалам.

2. Диаграммы рассеяния: Функция `scatter` в Scilab позволяет строить диаграммы рассеяния для визуализации взаимосвязи между двумя переменными.

Статистический анализ:

1. t-тест: Scilab предлагает функцию `ttest` для выполнения t-теста, который позволяет сравнивать средние значения двух групп данных и определять, являются ли эти различия статистически значимыми.

2. Корреляция: Scilab предоставляет функцию `corrcoef` для расчета коэффициента корреляции Пирсона между двумя переменными, что позволяет определить степень и направление связи между ними.

Регрессионный анализ:

1. Линейная регрессия: Scilab предоставляет функции для выполнения линейной регрессии, такие как `polyfit` и `linreg`. Эти функции позволяют аппроксимировать данные линейной моделью и получать коэффициенты регрессии.

2. Нелинейная регрессия: Scilab также предлагает функцию `curvefit`, которая позволяет выполнять нелинейную регрессию для моделирования нелинейных связей между переменными.

1.4 Компьютерное моделирование физических систем в Scilab.

Компьютерное моделирование физических систем в Scilab представляет собой процесс создания математических моделей, описывающих поведение физических систем, и использования численных методов для анализа и предсказания этого поведения. Scilab предоставляет широкий набор функций и инструментов, которые могут быть использованы для моделирования различных физических систем. Вот некоторые основные аспекты и функции, которые можно использовать при компьютерном моделировании физических систем в Scilab:

1. Математическое моделирование: Дифференциальные уравнения: Scilab предоставляет функции для решения дифференциальных уравнений, которые описывают изменение физических величин со временем. Некоторые из этих функций включают `ode`, `ode45`, `ode23`, которые позволяют численно решать дифференциальные уравнения и получать численные решения.

2. Алгебраические уравнения: Scilab также предлагает функции для решения систем алгебраических уравнений, которые могут возникать при моделировании физических систем. Некоторые функции для решения алгебраических уравнений включают `fsolve`, `nsolve`, `roots`.

3. Моделирование механических систем: Кинематика и динамика: С использованием математических функций и операций Scilab, таких как векторные и матричные операции, можно моделировать кинематические и динамические характеристики механических систем. Например, можно использовать функции для расчета скоростей, ускорений, сил и моментов в механических системах.

4. Моделирование электрических систем: Линейные системы: Scilab предлагает функции для моделирования и анализа линейных электрических систем. Некоторые из этих функций включают `ss` (State-Space), `tf` (Transfer Function), `bode`, которые позволяют анализировать и предсказывать поведение линейных систем.

5. Электрические цепи: Scilab также предоставляет функции для моделирования электрических цепей, включая элементы, такие как резисторы, конденсаторы, индуктивности и источники питания. Можно использовать функции для расчета токов, напряжений, импедансов и других параметров электрических цепей.

6. Симуляция и анализ результатов: Численные методы: Scilab предоставляет множество численных методов, которые могут быть применены для анализа результатов моделирования физических систем. Это включает численные методы интегрирования, решения систем уравнений, оптимизации и других.

7. Визуализация: Scilab также предлагает функции для визуализации результатов моделирования. Например, функции `plot`, `plot3d`, `scatter` позволяют строить графики и диаграммы для визуализации данных. Кроме того, можно использовать функцию `animate` для создания анимаций и визуализации динамических процессов.

С помощью этих функций и подходов Scilab позволяет создавать и анализировать модели физических систем.

1.5 Функции Scilab используемые для моделирования.

Решение дифференциальных уравнений: Scilab предоставляет несколько функций для решения обыкновенных дифференциальных уравнений, включая «ode» и «ode45». Эти функции используют численные методы для решения задач с начальными значениями и обеспечивают численные решения дифференциальных уравнений. вызов функции выглядит так: $[t, y] = \text{ode}(\text{'ode_function'}, \text{tspan}, y0, \text{где:}$

- 'ode_function' - это имя пользовательской функции, которая определяет ODE.
- tspan - это вектор, определяющий промежуток времени, в течение которого должно быть выполнено интегрирование.
- y0 - это начальное условие ODE.

Подгонка по кривой: Функция «polyfit» в Scilab выполняет подгонку полиномиальной кривой к заданному набору точек данных. Он находит коэффициенты многочлена, которые наилучшим образом соответствуют данным, позволяя вам создать математическую модель, аппроксимирующую данные. Вызов функции выглядит так: $p = \text{polyfit}(x, y, n)$, где:

- x – вектор x -координат точек данных,
- y – вектор y -координат точек данных,
- n – степень многочлена, подлежащего подгонке.
- polyfit возвращает вектор p , содержащий коэффициенты полиномиальной подгонки.

Интерполяция: Scilab предоставляет такие функции, как «interp» и «spline» для интерполяции. Эти функции позволяют создавать плавные кривые или поверхности на основе набора дискретных точек данных. Интерполяция полезна для создания непрерывных представлений дискретных данных. Вызов функции выглядит так: $y_{\text{int}} = \text{interp}(x, y, x_i)$, где:

- x – вектор x -координат известных точек,
- y – вектор y -координат известных точек, x_i – вектор x -координат, в котором требуется интерполяция.
- "interp" возвращает вектор y , содержащий интерполированные значения при соответствующих значениях x_i .

$y_p = \text{spline}(x, y, x_p)$, где:

- x – вектор x -координат известных точек,
- y – вектор y -координат известных точек,
- x_p – вектор x -координат, в котором требуется интерполяция.
- spline – возвращает вектор y_p , содержащий интерполированные значения при соответствующих значениях x_p .

Оптимизация: Scilab предлагает функции оптимизации, такие как «optim» и «fminsearch», которые можно использовать для поиска оптимальных значений параметров в модели. Эти функции позволяют вам определить целевую функцию и выполнить поиск значений параметров, которые минимизируют или максимизируют целевую величину. Вызов функции выглядит так: [optimum, fval, exitflag, output] = optim(function, x0), где:

- function – целевая функция, которая должна быть сведена к минимуму или максимизирована,
- x0 – первоначальное предположение или отправная точка для оптимизации.
- optim возвращает следующие выходные данные: optimum – оптимальное решение,
- fval – значение целевой функции в оптимальном состоянии,
- exitflag – флаг условия выхода (0 для успешной конвергенции).
выходные данные: Дополнительная информация и статистические данные о процессе оптимизации.

[optimum, fval, exitflag] = fminsearch(function, x0), где:

- function – целевая функция, которая должна быть сведена к минимуму,
- x0 – первоначальное предположение или отправная точка для оптимизации,
- fminsearch возвращает следующие выходные данные, optimum – оптимальное решение (свести к минимуму),
- fval – значение целевой функции в оптимальном состоянии,
- exitflag – флаг условия выхода (0 для успешной конвергенции).

Проектирование системы управления: Scilab включает в себя функции для проектирования и анализа систем управления. Функция «syslin» позволяет создавать линейные системы, не зависящие от времени, и вы можете использовать такие функции, как «tf2ss» и «ss2tf», для преобразования между передаточной функцией и представлениями пространства состояний. Scilab также предоставляет функции для размещения полюсов, проектирования контроллера и анализа отклика системы. Вызов функции выглядит так: sys = syslin(type, num, den), где:

- type – тип системы,
- num – коэффициенты многочлена числителя передаточной функции,
- den – коэффициенты многочлена знаменателя передаточной функции.
Функция "
- syslin" возвращает системный объект sys, представляющий систему.

[A, B, C, D] = tf2ss(num, den), где:

- num – коэффициенты многочлена числителя передаточной функции,
- den – коэффициенты многочлена знаменателя передаточной функции.
- tf2ss возвращает представление пространства состояний системы в виде матриц A, B, C и D.

[num, den] = ss2tf(A, B, C, D), где:

A, B, C, D: Матрицы пространства состояний системы.

ss2tf возвращает представление передаточной функции системы в виде полиномов числителя и знаменателя num и den.

Статистическое моделирование: Scilab имеет комплексный статистический модуль, который предоставляет функции для статистического моделирования и анализа. Такие функции, как «regress» и «anova», позволяют выполнять линейную регрессию и дисперсионный анализ соответственно. Также можно использовать функции для распределения вероятностей, проверки гипотез и визуализации данных. Вызов функции выглядит так: [b, stats] = regress (y, X), где:

- y – вектор значений переменной отклика,
- X – матрица значений предикторных переменных.
- regress возвращает следующие выходные данные:
- b – оценки коэффициентов модели линейной регрессии,
- stats –дополнительная статистика и информация о регрессионном анализе.

result = anova(y, factors), где:

- y – вектор значений переменной отклика,
- factors – матрица или вектор уровней факторов или предикторных переменных.
- anova возвращает таблицу, суммирующую результаты анализа.

Символьное вычисление: Scilab интегрируется с символьным модулем Scilab/Xcos, который предоставляет возможности символьных вычислений. Вы можете выполнять символьные вычисления, решать символьные уравнения и манипулировать символьными выражениями для получения математических моделей.

1.6 Функции построение и сохранение графиков

Мощные инструменты для работы с графиками и визуализации результатов моделирования. Вот некоторые основные возможности и функции, которые можно использовать для создания графиков и визуализации в Scilab:

Функция plot: Функция plot используется для построения двухмерных графиков. Возможно передать массивы данных для осей x и y и настроить внешний вид графика, такой как цвет линий, типы линий, маркеры точек. Описание функции plot:

Синтаксис: plot(x, y), где:

- x - массив значений по оси X. Может быть вектором или матрицей.
- y - массив значений по оси Y. Может быть вектором или матрицей.

Параметры функции: 'r' - задает цвет линии (например, 'r' для красного цвета), 'linewidth' - задает толщину линии, 'marker' - задает символы для отображения точек (например, 'o' для круглых точек), 'markersize' - задает размер символов точек, 'linestyle' - задает стиль линии (например, '--' для пунктирной линии).

Функция plot3d: Функция plot3d в Scilab предназначена для построения трехмерных графиков. Она позволяет визуализировать зависимости и распределения данных в трехмерном пространстве. Функция plot3d имеет множество параметров, которые позволяют настраивать внешний вид трехмерного графика. Вот подробное описание функции plot3d:

Синтаксис: plot3d(x, y, z), где:

- x - массив значений по оси X. Может быть вектором или матрицей,
- y - массив значений по оси Y. Может быть вектором или матрицей,
- z (обязательный) - массив значений по оси Z. Может быть вектором или матрицей.

Параметры функции: 'r' - задает цвет линии (например, 'r' для красного цвета), 'linewidth' - задает толщину линии, 'marker' - задает символы для отображения точек (например, 'o' для круглых точек), 'markersize' - задает размер символов точек, 'linestyle' - задает стиль линии (например, '--' для пунктирной линии).

Функция scatter: Функция scatter в Scilab предназначена для построения точечных диаграмм. Она позволяет визуализировать наборы данных, отображая отдельные точки на плоскости. Функция scatter имеет множество параметров, которые позволяют настраивать внешний вид точек и их свойства. Вот подробное описание функции scatter:

Синтаксис: scatter(x, y), где:

- x - массив значений по оси X . Может быть вектором или матрицей,
- y - массив значений по оси Y . Может быть вектором или матрицей.

Параметры функции: 'r' - задает цвет точек (например, 'r' для красного цвета), 'markersize' - задает размер точек, 'marker' - задает символы для отображения точек (например, 'o' для круглых точек), 'filled' - указывает, будут ли точки заполнены цветом.

В Scilab доступно несколько функций для настройки внешнего вида графиков и добавления различных элементов, таких как подписи осей, заголовков, сетка, легенда и другие. Вот подробное описание этих функций:

`xlabel` - функция `xlabel` используется для добавления подписи оси X к графику. Она принимает строковый аргумент, который указывает текст подписи. Пример: `xlabel('Время (сек)')`

`ylabel` - функция `ylabel` используется для добавления подписи оси Y к графику. Она также принимает строковый аргумент, указывающий текст подписи. Пример использования: `ylabel('Амплитуда')`

`zlabel` - если вы работаете с трехмерным графиком, функция `zlabel` позволяет добавить подпись оси Z . Она также принимает строковый аргумент для указания текста подписи. Пример: `zlabel('Глубина')`

`title` - функция `title` используется для добавления заголовка к графику. Она принимает строковый аргумент, указывающий текст заголовка. Пример: `title('График зависимости')`

`grid` - функция `grid` позволяет включить или выключить отображение сетки на графике. Она принимает один булевый аргумент (Т или F), где Т включает сетку, а F выключает. Пример использования: `grid(T)`

`legend` - функция `legend` используется для добавления легенды к графику, чтобы идентифицировать различные линии или точки на графике. Она принимает строковые аргументы, представляющие текстовые метки для каждого элемента. Пример: `legend('Линия 1', 'Линия 2')`.

1.7 Экспорт данных

Сохранение графиков: Использование функции `xs2pdf` в Scilab позволяет сохранять построенные графики в файлы различных форматов, включая PNG, JPEG, SVG и другие. Вот подробное описание этой функции:

Синтаксис: `xs2pdf(handle, filename, format)`, где:

- `filename` (обязательный) - строка, указывающая имя файла, в котором будет сохранен график.
- `format` (обязательный) - строка, определяющая формат файла. Допустимые значения включают 'png', 'jpg', 'svg' и другие, в зависимости от поддерживаемых форматов в вашей установке Scilab.

Параметры: `figure_handle` (обязательный) - указатель на графический объект или номер окна графика, который требуется сохранить. Вы можете использовать команду `gcf()` для получения текущего графического объекта.

2. Практическая часть

2.1 Оформление графика при помощи функции plot

В качестве примера построим графики функций $y = \sin(x/2)$, $z = \cos(x)$ и $v = e(\cos(x))$ в одних координатных осях и независимо определим внешний вид каждого графика. Пусть график функции $y = \sin(x/2)$ будет штриховым, $z = \cos(x)$ — черного цвета, с маркером в виде звездочки, $v = \exp(\cos(x))$ — штриховым, красного цвета, с маркером в виде точки (см. рисунок 1, рисунок 2).

```
x = -6.28:0.2:6.28;
y = sin(x/2);
z = cos(x);
v = exp(cos(x));
figure;
subplot(3, 1, 1);
plot(x, y, '--');
title('График sin(x/2)');
xlabel('x');
ylabel('y');
subplot(3, 1, 2);
plot(x, z, 'k*');
title('График cos(x)');
xlabel('x');
ylabel('z');
subplot(3, 1, 3);
plot(x, v, 'r.--');
title('График exp(cos(x))');
xlabel('x');
ylabel('v');
```

Рисунок 1: код для реализации графика с помощью plot

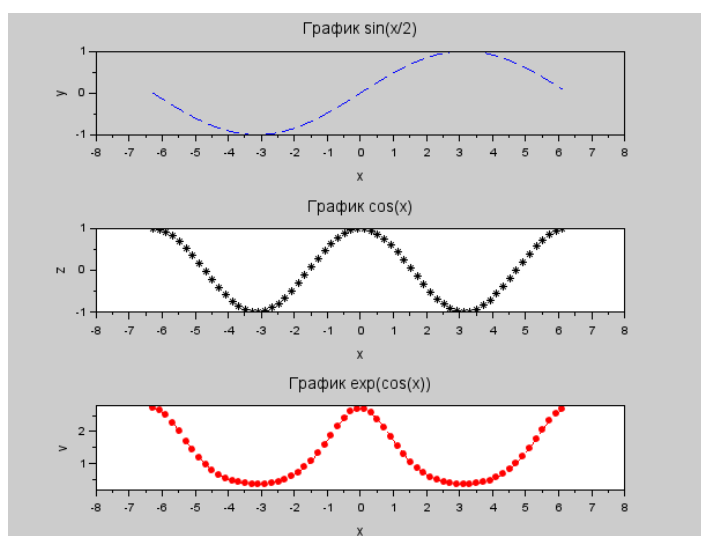


Рисунок 2: двухмерные графики

2.2 Оформление графика при помощи функции plot2d

В качестве примера построим в одних координатных осях графики функций $y = \sin(x)$ и $y = \cos(x)$, для синусоиды с помощью параметра style определим имя цвета — красный («red»), а для косинусоиды — id зеленого цвета (см. рисунок 3, рисунок 4).

```
x = [-2 * %pi : 0.1 : 2 * %pi];
y = [sin(x); cos(x)];
plot2d(x, y', style = [color("red"), color(0, 176, 0)], rect = [-8, -2, 8, 2]);
```

Рисунок 3: код для реализации графика с помощью plot2d

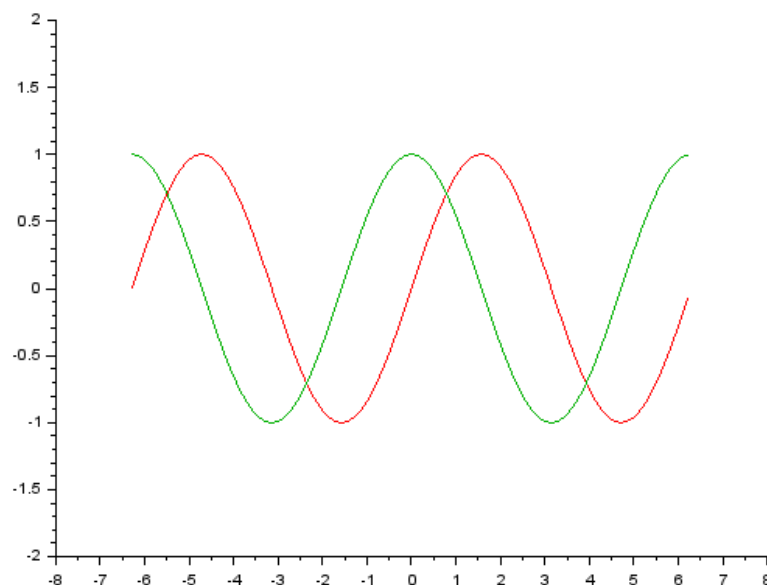


Рисунок 4: двухмерный график

2.3 Оформление графика при помощи функции plot3d

Построение трехмерного графика рассмотрим на примере функции $z(x, y) = 5y^2 - x^2$ в области $x \in [-2; 2]$, $y \in [-3; 3]$ (см. рисунок 5, рисунок 6).


```

x=[-2:0.1:2];
y=[-3:0.1:3];
for i=1:length(x)
for j=1:length(y)
z(i,j)=5*y(j)^2-x(i)^2;
end
end
plot3d(x',y',z,35,45);

```

Рисунок 5: код для реализации графика с помощью plot3d

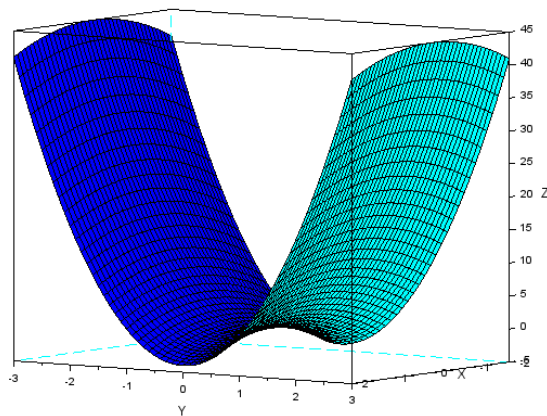


Рисунок 6: трехмерный график

2.4 Оформление графика при помощи функции scatter3d

Построение трехмерного графика с помощью функции scatter3d рассмотрим на примере $z = e(-x^2 - y^2)$ (см. рисунок 7, рисунок 8).

```

n=20;
[x,y]=meshgrid(linspace(-2,2,n));
z=-exp(-x.^2-y.^2);
subplot(2,1,1);
scatter3d(gca(),x(:),y(:),z(:));
gca().rotation_angles=[60,45];
subplot(2,1,2);
scatter3d(gca(),x(:),y(:),z(:),'filled','markerFaceColor',[0.8,0.8]);
gca().rotation_angles=[60,45];

```

Рисунок 7: код для реализации графика с помощью scatter3d

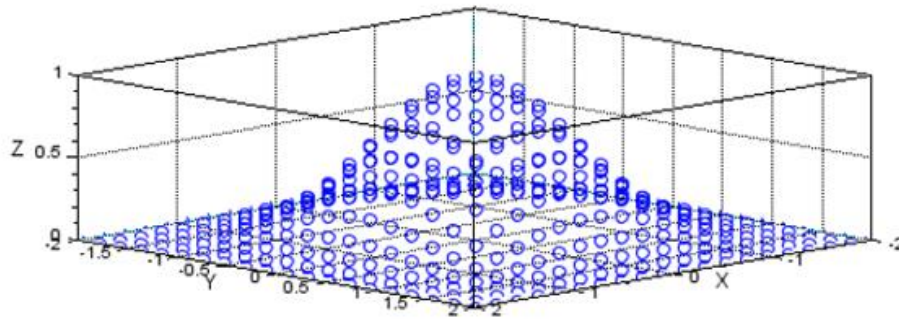


Рисунок 8: точечная диаграмма

2.5 Модель Лотки – Вольтерры взаимодействия двух видов «хищники – жертва»

Скорость роста численности жертв уменьшается пропорционально численности хищников

$$\frac{dx}{dt} = -ax(t) + bx(t)y(t)$$

$$\frac{dy}{dt} = cy(t) - dx(t)y(t)$$

Где,

a, d – коэффициенты смертности

b, c – коэффициенты прироста популяции

x_0 – начальные значения (популяции хищников и жертв)

```

a = 0.2;
b = 0.5;
c = 0.05;
d = 0.02;
function dx=syst2(t, x)
...dx(1) = -a * x(1) + c * x(1) * x(2);
...dx(2) = b * x(2) - d * x(1) * x(2);
endfunction
t0 = 0;
x0 = [5;10];
t = [0:0.1:400];
y = ode(x0, t0, t, syst2);
n = size(y, 'c');
for i = 1:n
...y2(i) = y(2, i);
...y1(i) = y(1, i);
end
plot(t, y1); // построение графика колебаний изменения числа популяции хищников
plot(t, y2); // построение графика колебаний изменения числа популяции жертв
plot(y1, y2); // построение графика зависимости изменения численности хищников от изменения численности жертв

```

Рисунок 9. Код для реализации Модели Лотки

Первый график показывает колебания изменения численности популяции хищников в зависимости от времени. Ось x представляет собой время (t), а ось y отражает численность популяции хищников (y_1).

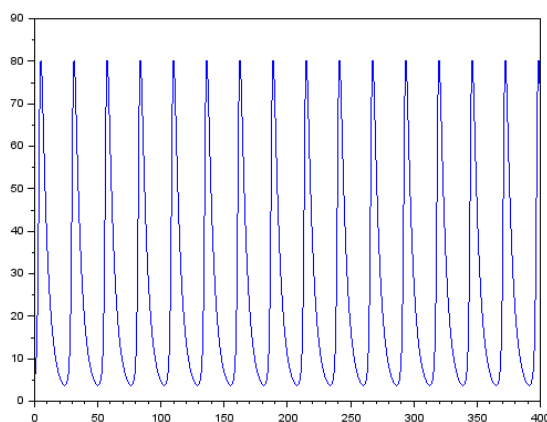


Рисунок 10. колебания изменения численности популяции хищников

Второй график отображает колебания изменения численности популяции жертв в зависимости от времени. Ось x представляет собой время (t), а ось y отражает численность популяции жертв (y_2).

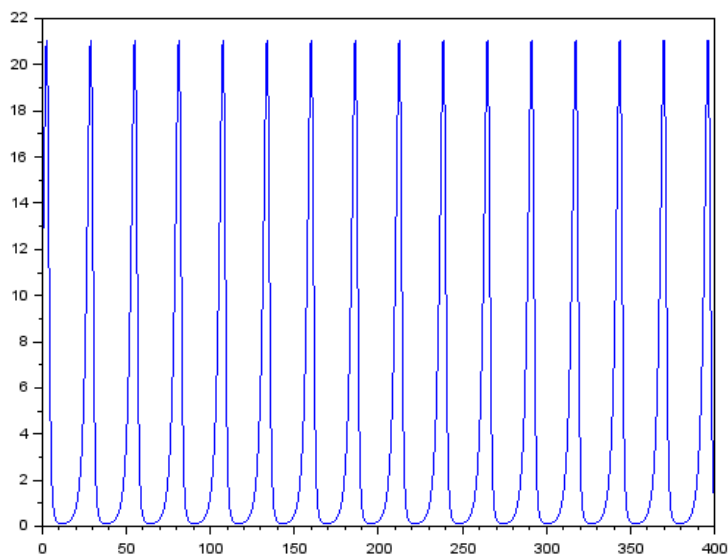


Рисунок 11: колебания изменения численности популяции жертв в зависимости от времени

Третий график представляет собой зависимость изменения численности хищников от изменения численности жертв. Ось x представляет собой численность жертв (y_1), а ось y отражает численность хищников (y_2).

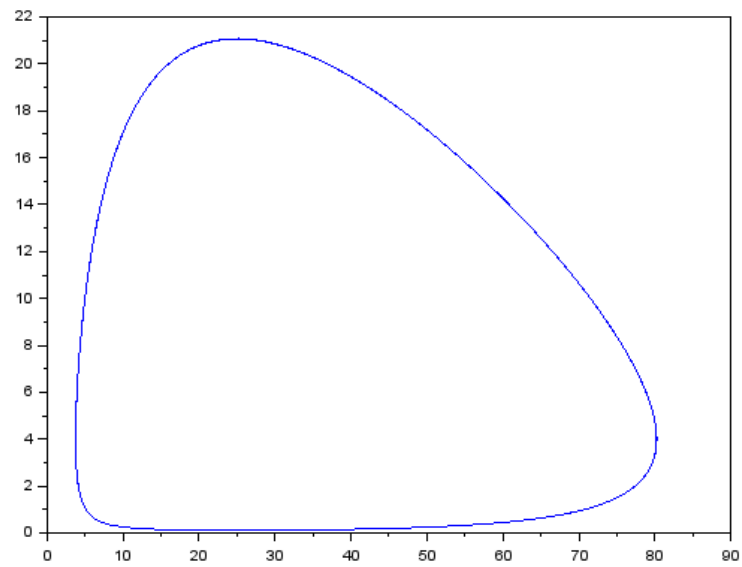


Рисунок 12: зависимость изменения численности хищников от изменения численности жертв

ЗАКЛЮЧЕНИЕ

Для написания курсовой работы была изучена специальная литература – учебные пособия и статьи.

Цель работы состояла в демонстрации эффективного использования Scilab для компьютерного моделирования. В ходе работы были рассмотрены различные аспекты использования Scilab и его функций для моделирования.

Практическая часть работы включает конкретные примеры использования функций Scilab для построения и оформления графиков. Были рассмотрены функции `plot`, `plot2d`, `plot3d` и `scatter3d`, которые позволяют создавать разнообразные графические представления данных и моделей. В конце работы приведена модель Лотки-Вольтерры взаимодействия двух видов. Это классическая модель, используемая для описания взаимодействия популяций в экологии. Ее пример позволяет проиллюстрировать применение Scilab для моделирования конкретных систем и анализа их поведения.

Работа продемонстрировала, что Scilab представляет собой мощный инструмент для компьютерного моделирования. Таким образом цель была достигнута.

ЛИТЕРАТУРА

1. Григорьев, С. В. Руководство по Scilab: для начинающих. Москва: Бином, (2015).
2. Григорьев, С. В. Компьютерное моделирование в Scilab: практикум. Москва: Бином, (2018).
3. Демидов, Л. А. Математическое моделирование и компьютерное моделирование. Москва: Издательство "Лань", (2014).
4. Журавлев, Ю. И., & Ткачук, В. И. Моделирование в математическом пакете Scilab. СПб: Питер, (2017).
5. Наймарк, В. Е., & Волынский, А. Л. Практическое моделирование в приложениях Scilab/Matlab. Москва: ФИЗМАТЛИТ, (2015).
6. Фомин, С. А., Красноперов, А. Н., & Яковлев, А. Н. Моделирование и синтез систем автоматического управления в Scilab: учебное пособие. СПб: БХВ-Петербург, (2016).