МИНИСТЕРСТВО ОБРАЗОВАНИЯ РЕСПУБЛИКИ БЕЛАРУСЬ

**УЧРЕЖДЕНИЕ ОБРАЗОВАНИЯ
ГОМЕЛЬСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ
УНИВЕРСИТЕТ ИМЕНИ П. О. СУХОГО**

Заочный факультет
Кафедра «Информатика»

РАСЧЕТНО-ПОЯСНИТЕЛЬНАЯ ЗАПИСКА
к курсовой работе
по дисциплине «Информатика»

на тему: **«КОМПЬЮТЕРНОЕ МОДЕЛИРОВАНИЕ ЭЛЕМЕНТОВ ЭЛЕКТРИЧЕСКИХ ЦЕПЕЙ. РАСЧЕТ ПАРАМЕТРОВ ЕМКОСТИ»**

Вариант 4-2

Исполнитель: студент гр. ________
____
Руководитель: преподаватель
____

Дата проверки: ________________________
Дата допуска к защите: ________________________
Дата защиты: ________________________

Оценка работы: ________________________

Подписи членов комиссии
по защите курсовой работы: ________________________________

Гомель 2022

Содержание

Введение

В теории электрических цепей емкостью называют взаимную емкость между двумя проводниками; параметр емкостного элемента электрической схемы, представленного в виде двухполюсника. Такая емкость определяется как отношение величины электрического заряда к разности потенциалов между этими проводниками. Электрическая емкость – характеристика проводника, мера его способности накапливать электрический заряд.

В данной курсовой работе необходимо с использованием системы *Scilab* исследовать *модель электрических цепей, для расчета параметров емкости*.

В ходе исследования курсовой работы использовались возможности системы компьютерной математики – *Scilab*. Она предназначена для выполнения инженерных и научных вычислений, таких как: решение нелинейных уравнений и систем; решение задач линейной алгебры; решение задач оптимизации; дифференцирование и интегрирование; задачи обработка экспериментальных данных; интерполяция и аппроксимация метод; решение обыкновенных дифференциальных уравнений и систем.

Кроме того предоставляет широкие возможности по созданию и редактированию, *Scilab* различных видов графиков и поверхностей. Пользователь может создать любую новую команду или функцию, а затем использовать ее наравне со встроенными функциями. К тому же система имеет достаточно мощный собственный язык программирования высокого уровня, что говорит о возможности решения новых задач.

Актуальность выбранной темы обусловлена значимостью емкости в электрической цепи, а также значимостью емкости в вопросах работы электрического оборудования, его надежности и предотвращении отказов.

Цель – компьютерное моделирование элементов электрической цепи. В курсовой работе ставятся задачи:

– найти радиус R, соответствующий требуемой емкости, вычисленный при помощи встроенной функции *fsolve* и при помощи численного метода половинного деления;

– исследовать влияние расстояния между дисками на значения радиуса диска;

– подобрать сплайновую интерполирующую зависимость по результатам расчетов. Построить график исходной и интерполирующей функций на одном поле;

– вычислить значения диапазона $L_1 – L_2$, при котором R находится в промежутке $R_1 – R_2$ (значения R_1 и R_2 задать с клавиатуры).

1 Математическое моделирование технических объектов

1.1 Понятие математической модели, их классификация и свойства

Математическая модель – это совокупность математических объектов и отношений между ними, адекватно отображающая физические свойства технического объекта.

На различных этапах и стадиях проектирования сложной технической системы используют различные математические модели. Математические модели могут представлять собой системы дифференциальных уравнений, системы алгебраических уравнений, простые алгебраические выражения, бинарные отношения, матрицы и так далее. Уравнение математической модели связывают физические величины [1, стр. 22].

Используют следующие виды математических моделей: *детерминированные* и *вероятностные, теоретические* и *экспериментальные факторные, линейные* и *не линейные, динамические* и *статистические, непрерывные* и *дискретные, функциональные* и *структурные*.

Деление математических моделей на *функциональные* и *структурные* определяется характером отображаемых свойств технического объекта.

Структурные математические модели отображают только структуру объектов и используются при решении задач структурного синтеза. Параметрами структурных моделей являются признаки функциональных или конструктивных элементов, по которым один вариант структуры объекта отличается от другого.

Функциональные математические модели предназначены для отображения информационных, физических, временных процессов, протекающих в работающем оборудовании, в ходе выполнения технологических процессов и т.д.

По способам получения функциональные математические модели делятся на:

– *теоретические модели* – получают на основе описания физических процессов функционирования объекта;

– *экспериментальные модели* – получают на основе поведения объекта во внешней среде, рассматривая его как кибернетический "черный ящик".

Функциональные математические модели также делятся на:

– *линейные модели*, содержащие только линейные функции фазовых переменных и их производных;

– *нелинейные математические модели*, включающие в себя нелинейные функции фазовых переменных и их производных.

Если при моделировании учитывается инерциальные свойства технического объекта и (или) изменение во времени параметров объекта или внешней среды, то модель называют динамической. В противном случае модель статическая. Динамическая модель может также представлять собой интегральные уравнения, придаточные функции, а в аналитической форме – явные зависимости фазовых координат или выходных параметров технического объекта от времени. Графическая модель представляется в виде графов, эквивалентных схем, динамических моделей, диаграмм и т.п.

При проектировании технических объектов используют множество видов математических моделей, в зависимости от уровня иерархии, степени декомпозиции системы, аспекта, стадии и этапа проектирования.

В общем случае уравнения математической модели связывают физические величины, которые характеризуют состояние объекта и не относятся к перечисленным выше выходным, внутренним и внешним параметрам. Такими величинами являются: скорости и силы - в механических системах; расходы и давления - в гидравлических и пневматических системах; температуры и тепловые потоки - в тепловых системах; токи и напряжения - в электрических системах [1, стр. 26].

Свойства модели:

– модель отображает реальную систему, то есть за моделью всегда должна стоять реальность;

– отображение должно быть упрощенным. Упрощенным в том смысле, что должны отображаться не все свойства (особенности) реальной системы, а лишь те из них, которые в настоящий момент интересуют исследователя, являются важными с точки зрения поставленной задачи.

– с моделью должно быть проще оперировать, чем с реальной системой.

– между реальной системой (оригиналом) и ее моделью должно иметь место определенное соответствие, с помощью которого устанавливается заданная точность отображения моделью свойств реальной системы [2, стр. 41].

1.2 Численные методы решения алгебраических уравнений

Численное решение нелинейных уравнений вида $F(x)=0$ заключается в нахождении значений x, удовлетворяющих (с заданной точностью) данному уравнению и состоит из этапов: отделение корней уравнения и уточнение с помощью некоторого вычислительного алгоритма конкретного выделенного корня с заданной точностью.

Целью первого этапа является нахождение отрезков из области определения функции, внутри которых содержится только один корень решаемого уравнения.

Метод деления отрезка пополам. Интервал изоляции действительно корня всегда можно уменьшить путем деления его, например, пополам, определяя, на границах, какой из частей первоначального интервала функция *f(x)* меняет знак. Затем полученный интервал снова делят на две части и т.д. Такой процесс проводится до тех пор, пока не перестанут изменяться сохраняемые в ответ десятичные знаки [3, стр. 5].

Метод касательных. Пусть действительный корень уравнения $f(x)=0$ изолирован на отрезке $[a,b]$. Возьмем на отрезке $[a,b]$ такое число x_0, при котором $f(x_0)$ имеет тот же знак, что и $f_n(x_0)$, т.е. $f(x_0)\cdot(x_0)>0$. Проведем в точке $M_0[x_0; f(x_0)]$ касательную к кривой $y=f(x)$.За приближенное значение корня примем абсциссу точки пересечения этой касательной с осью Ох. Это приближенное значение корня находится по формуле

$$x_1 = x_0 - f(x_0)/f_1(x_0).$$

Применив этот прием вторично в точке $M_1[x_1; f(x_1)]$, найдем x_2 и т. д. Полученная последовательность $x_0, x_1, x_2 \ldots$ имеет своим пределом искомый корень [3, стр. 9].

Комбинированный метод хорд и касательных. Пусть требуется найти действительный корень уравнения $f(x)=0$, изолированный на отрезке $[a,b]$. Предполагается, что $f(a)$ и $f(b)$ имеют равные знаки, а каждая из производных сохраняет определенный знак на отрезке изоляции. Возьмем на отрезке $[a,b]$ такую точку хо, что $f(x_0)$ и $f''(x_0)$ имеют одинаковые знаки. Воспользуемся формулами методов хорд и касательных:

$$x_{11} = x_0 - f(x_0)/f_1(x_0),\ x_{12} = a-(b-a)f(a)/f(b)-f(a). \tag{1.1}$$

Величины x_{11} и x_{12} принадлежат промежутку изоляции, причем $f(x_{11})$ и $f(x_{12})$ имеют разные знаки. Применив формулу (1.1) к следующим точкам получим ряд последовательностей $x_{11}, x_{21}, x_{31},\ldots x_{n1}, \ldots; x_{12}, x_{22}, x_{32}, \ldots, x_{n2}, \ldots$стремится к искомому корню, причем одна из последовательностей монотонно возрастает, а другая – монотонно убывает.

Численный метод простых итераций. Пусть известно, что нелинейное уравнение $f(x)=0$ имеет на отрезке $[a,b]$ единственный вещественный корень $\xi\in[a,b]$. Требуется найти этот корень с заданной точностью. Приведем уравнение к виду

$$x=\varphi(x). \tag{1.2}$$

Выберем произвольно приближенное значение корня $x_0\in[a,b]$ и вычислим $\varphi(x_0)=x_1$. Найденное значение x_1 подставим в правую часть соотношения (1.2) и вычислим $\varphi(x_1)=x_2$. Продолжая процесс вычислений дальше, получим числовую последовательность.

Если существует предел этой последовательности, то он и является корнем уравнения (1.2). Корень можно вычислить по итерационной формуле

$$x_{n+1} = \varphi(x_n),\ n = 0,1,2,... \qquad (1.3)$$

Достаточное условие, при котором итерационный процесс сходится, определяет следующая теорема: пусть функция $\varphi(x)$ определена и дифференцируема на отрезке $[a,b]$, причем все ее значения $\varphi(x) \in [a,b]$ и выполняется условие

$$|\varphi'(x)| \le q < 1, \forall x \in [a,b].$$

Тогда процесс итераций (1.3) сходится независимо от начального значения, и предельное значение является единственным корнем уравнения (1.2).

Метод простых итераций и почти все другие итерационные методы имеет два достоинства: является универсальным и любая неточность на каком-либо шаге итераций не отразится на конечном результате, а лишь на количестве итераций; позволяет достигнуть любой заданной точности при любом начальном приближении $x_0 \in [a,b]$.

Недостатки метода: трудность приведения уравнения (1.2) к виду (1.3); если начальное приближение x_0 далеко от корня, то число итераций очень много [3, стр. 12].

1.3 Аппроксимация и интерполяция в системе *Scilab*

Интерполяция – способ обработки экспериментальных данных, придуманный еще в 18 веке Ньютоном и Лагранжем. Сущность метода в следующем: имеется таблица значений функции в точках (узлах). Чтобы найти значения в промежуточных точках, построим интерполяционный многочлен степени *n*, значения которого в узлах совпадают с табличными значениями.

Аппроксимация – тоже способ обработки экспериментальных данных, но принцип тут другой. Для приближения искомой функции тоже используется многочлен, но на его коэффициенты накладывается условие: чтобы сумма квадратов отклонений многочлена от экспериментальных данных в узлах была минимальна. Преимущество аппроксимации в том, что она сглаживает случайные ошибки в данных, в то время как интерполяция наоборот, подчеркивает ошибки в отдельных узлах.

Метод наименьших квадратов позволяет по экспериментальным данным подобрать такую функцию, которая проходит настолько близко к экспериментальным точкам, насколько это возможно. Идея метода заключается в том, что функцию необходимо подобрать таким образом, чтобы сумма квадратов отклонений измеренных значений от расчетных была наименьшей.

Для реализации этой задачи в *Scilab* предусмотрена следующая функция

[a,S]=datafit(F,z,c),

где F – функция, параметры которой необходимо подобрать; z – матрица исходных данных; c – вектор начальных приближений; a – вектор коэффициентов; S – сумма квадратов отклонений измеренных значений от расчетных [4, стр. 214].

Показатель, характеризующий тесноту линейной связи между x и y, называемый коэффициентом корреляции, рассчитывается по формуле:

$$r = \frac{\sum_{i=1}^{n}(x_i - M_x)\cdot(y_i - M_y)}{\sqrt{\sum_{i=1}^{n}(x_i - M_x)^2 \cdot \sum_{i=1}^{n}(y_i - M_y)^2}}.$$

Значение коэффициента корреляции удовлетворяет соотношению $-1 \le r \le 1$. Чем меньше отличается абсолютная величина r от единицы, тем ближе к линии регрессии располагаются экспериментальные точки. Если коэффициент корреляции близок к нулю, то это означает, что между x и y отсутствует линейная связь, но может существовать другая, нелинейная, зависимость.

Аналогом коэффициента корреляции r для нелинейных зависимостей является индекс корреляции, рассчитываемый по формуле:

$$R = \sqrt{1 - \frac{\sum_{i=1}^{n}(y_i - Y_i)^2}{\sum_{i=1}^{n}(y_i - M_y)^2}},$$

где y – экспериментальные значения; Y – значения, найденные методом наименьших квадратов; M – среднее значение y.

Индекс корреляции по своему абсолютному значению колеблется в пределах от 0 до 1. Если коэффициент корреляции r является мерой тесноты связи только для линейной формы, то индекс корреляции R является мерой тесноты связи и для линейной, и для криволинейной связи.

Для расчета коэффициентов регрессии в *Scilab* предназначена функция

a=regress(x,y),

где x и y – экспериментальные данные; a – вектор коэффициентов линии регрессии.

Для расчета коэффициента корреляции в *Scilab* также предназначена встроенная функция

a=corr(x,y),

где x и y – экспериментальные данные.

Для решения подобной задачи интерполирования довольно часто используют сплайн-интерполяцию (от английского слова *spline* – рейка, линейка). Один из наиболее распространенных вариантов интерполяции – *интерполяция кубическими сплайнами*. Кроме того, существуют квадратичные и линейные сплайны. В *Scilab* для построения линейной интерполяции служит функция

y=interpln(z,x),

где *z* – матрица исходных данных; *x* – вектор абсцисс; *y* – вектор значений линейного сплайна в точках *x* [4, стр. 219].

Построение кубического сплайна в *Scilab* состоит из двух этапов: вначале вычисляются коэффициенты сплайна с помощью функции *d=splin(x,y)*, а затем рассчитывается значения интерполяционного полинома в точке *y=interp(t,x,y,d)*. Для функции *y=interp(t,x,y,k)* параметры *x, y* и *d* имеют те же значения, параметр *t* – это вектор абсцисс, а *y* – вектор ординат, являющихся значениями кубического сплайна в точках *x*.

1.4 Основные элементы в системе *Scilab*, их применение в моделировании

Язык программирования *Scilab* относится к числу интерпретируемых языков высокого уровня, предоставляя пользователю возможность напрямую манипулировать математическими конструкциями, такими как матрицы или полиномы. Тем самым достигается большая скорость и простота написания программ. Пользователи пакета могут разрабатывать собственные модули расширения для решения конкретных задач.

Функциональные возможности расширились настолько, что охватили большинство разделов научных вычислений, включая: линейную алгебру и разреженные матрицы; полиномы и рациональные функции; интерполяцию и аппроксимацию; линейную, квадратичную и нелинейную оптимизацию; обыкновенные дифференциальные уравнения, алгебраические уравнения.

Имена переменных в *Scilab* могут иметь произвольную длину, однако лишь первые 24 символа имени являются значимыми, поэтому во избежание ошибок следует использовать имена длиной до 24 символов. Допустимыми символами в именах переменных являются латинские буквы, цифры, а также символы *”%”, ”_”, ”#”, ”!”, ”$”, ”?”*. Все элементарные функции векторизованы в том смысле, что могут быть применены к матрицам, при этом в результате также получаем матрицы. Если в инструкции не указано, какой переменной следует присвоить результат вычисления, результат заносится в специальную переменную *ans*.

Функция – это имеющий уникальное имя объект, выполняющий определенные преобразования своих аргументов и при этом возвращающий результаты этих преобразований. Функции в общем случае имеют список аргументов, заключенный в круглые скобки. Имена встроенных функций записываются строчными буквами. Если необходимо вывести данные на экран дисплея в определенной последовательности, отличной от последовательности их вычисления, применяется функция *disp*, которую принято называть оператором вывода.

Графические приложения в пакете *Scilab* выводятся в отдельном окне, называемом графическим. Функции *Scilab* для отображения графиков: *plot* – двухмерный график; *surf* – трехмерный график; *contour* – контурный график; *pie* – круговая диаграмма; *histplot* – гистограмма; *bar* – столбиковая диаграмма; *barh* – горизонтальная столбиковая диаграмма; *hist3d* – трехмерная гистограмма; *polarplot* – график в полярных координатах; *Matplot* – цветной двухмерный график матрицы.

Для создания в графическом окне нескольких графических областей для вывода графиков применяется команда *subplot(m, n, p)*.

Команда *plot(X,Y)* – строит график функции, координаты точек которой берутся из векторов одинаковой размерности *X* и *Y*. Обращение к функции имеет вид:

plot(x,y,[xcap,ycap,caption]) ,

где *x* – массив абсцисс; *y* – массив ординат; *xcap, ycap, caption* – подписи осей *X, Y* и графика соответственно [4, стр. 57].

Следующей функцией, которая может быть использована для построения графиков, является функция *plot2d*. В общем виде обращение к функции имеет вид:

plot2d([loglog],x,y,[key1=value1,key2=value2, ..., keyn=valuen]),

где *logflag* – строка из двух символов, каждый из которых определяет тип осей; *x* – массив абсцисс; *y* – массив ординат; *keyi=valuei* – последовательность значений параметров графиков.

В *Scilab* численное дифференцирование реализовано командой

dy=diff(y [,n]),

где *y* – значения функции *y(x)* в виде вектора вещественных чисел, *n* – порядок дифференцирования.

Более универсальной командой дифференцирования является команда:

g=numdiff(fun,x),

где *fun* – имя функции, задающей выражение для дифференцирования.

Функция должна быть задана в виде *y=fun(x [,p1,p2,..pn])*, где *x* – переменная по которой будет проводиться дифференцирование. Если параметры *p1,p2,..pn* присутствуют в описании функции, то должны быть обязательно определены при вызове, например, так: *g=numdiff(list(fun,p1,p2,..pn),x)*.

Решение алгебраического уравнения в *Scilab* состоит из двух этапов. Необходимо задать полином *P(x)* с помощью функции *poly*, а затем найти его корни, применив функцию *roots*. Численное решение нелинейного уравнения проводят в два этапа. Вначале отделяют корни уравнения, т.е. находят достаточно тесные промежутки, в которых содержится только один корень.

Эти промежутки называют интервалами изоляции корня, определить их можно, изобразив график функции *f(x)* или любым другим методом. На втором этапе проводят уточнение отделенных корней, или, иначе говоря, находят корни с заданной точностью.

Для решения трансцендентных уравнений в *Scilab* применяют функцию

fsolve(x0,f),

где *x0* – начальное приближение; *f* – функция, описывающая уравнение *y(x) = 0*.

Если заданы *m* уравнений с *n* неизвестными и требуется найти последовательность из *n* чисел, которые одновременно удовлетворяют каждому из *m* уравнений, то говорят о системе уравнений. Для решения систем уравнений в *Scilab* также применяют функцию *fsolve(x0,f)* [4, стр. 156].

2 Алгоритмический анализ задачи

2.1 Постановка задачи

1. Найдите радиус R, удовлетворяющий требуемому значению емкости C, при заданных в таблице 2.1 параметрах ε_1 и L. Доказать графически, что значение R найдено верно.

2. Найти значение радиуса R, используя численный метод, указанный в таблице 2.1, при решении уравнения. Выполнить графическую интерпретацию результатов расчетов. Сравнить полученное значение с рассчитанным в пункте 1.

3. Рассчитать значение радиуса R для 6 – 7 значений из диапазона значений варьируемого параметра, указанного в таблице 2.1. Построить сводный график зависимости полученных значений радиуса R от варьируемого параметра.

4. Подобрать сплайновую интерполирующую зависимость по результатам расчетов. Построить график исходной и интерполирующей функций на одном поле.

5. Выполнить расчет по индивидуальному заданию. Дать графическую интерпретацию результатов расчетов. Вычислить значения диапазона $L_1 - L_2$, при котором R находится в промежутке $R_1 - R_2$ (значения R_1 и R_2 задать с клавиатуры).

2.2 Анализ исходных и результирующих данных

Исходными данными для работы являются:

– относительная диэлектрическая проницаемость среды ε_1;

– параметр проницаемости: $\varepsilon_0=0.885$;

– расстояние между дисками L;

– электрическая емкость двух коаксиальных плоских дисков C.

Результирующими данными являются:

– радиус R, соответствующий требуемой емкости, вычисленный при помощи встроенной функции *fsolve*;

– функция поиска корня уравнения численным методом половинного деления;

– радиус R, соответствующий требуемой емкости, вычисленный при помощи численного метода половинного деления;

– вектор варьируемого параметра LL, который содержит значения расстояния между дисками;

– вектор размера *RR* соответствующий требуемой емкости;

– сводный график зависимости между исследуемыми векторами *LL* и *RR*;

– сплайновая интерполяция и ее график;

– диапазон, полученный при выполнении индивидуального задания.

Таблица 2.1 – Таблица исходных данных

ε_1	L, мм	C, пФ	ε_0	Наименование численного метода	варьируемый параметр	Диапазон
2	1	33	0.885	Половинного деления	L	2.0 – 4.0 (0.2)

2.3 Описание математической модели

Электрическая емкость двух коаксиальных плоских дисков (рисунок 2.1) при $L/R<1$ рассчитывается по формуле

$$C = \varepsilon_1 \varepsilon_0 R \left[\frac{\pi R}{L} + \ln\left(\frac{16\pi R}{L} \right) - 1 \right], \tag{2.1}$$

где ε_1 − относительная диэлектрическая проницаемость среды; R − радиус дисков; L − расстояние между дисками; ε_0 = 0.885 пФ/мм.

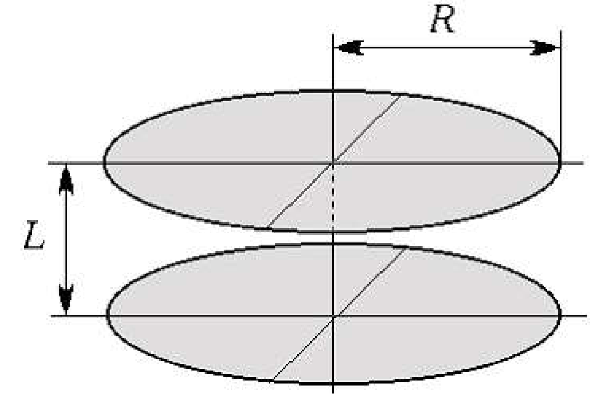


Рисунок 2.1 – Коаксиальные плоские диски

Для нахождения радиуса R при заданном значении емкости используем вспомогательную функцию

$$F(R) = C(R) - C_0, \tag{2.2}$$

где $C(R)$ – функция емкости двух коаксиальных плоских дисков, заданная формулой (2.1); C_0 – заданное значение емкости.

Таким образом, радиус R находим численным решением уравнения

$$F(R)=0. \tag{2.3}$$

Решаем полученное алгебраическое уравнение двумя методами: при помощи встроенной функции *fsolve* и при помощи численного метода половинного деления.

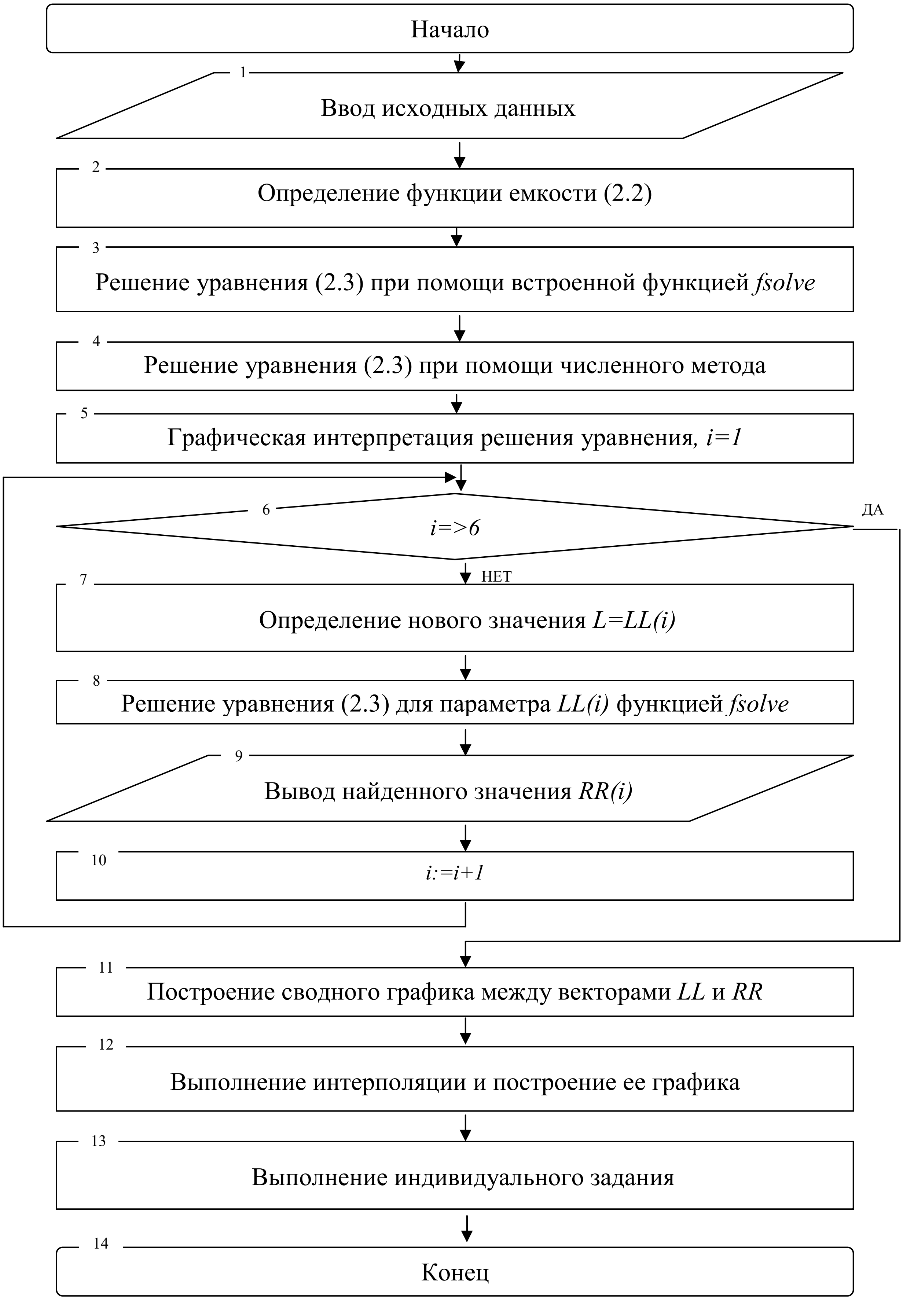
По результатам решения уравнений выполняем графическую интерпретацию решения: строим график функций *F(R)* и отмечаем на нем найденное значение.

Далее выполняем исследование математической модели. Для этого вводим значения варьируемого параметра *L* из заданного диапазона (таблица 2.1) в виде вектора *LL*. Для каждого из значений варьируемого параметра *L*: LL_1, LL_2, … , LL_6 переопределяем функцию *F(R)* и решаем уравнение при помощи функции *fsolve*. Полученные результаты расчетов представим в виде векторов *LL*, содержащем значения расстояний между дисками, и *RR*, содержащем значения найденного размера.

Выполняем сплайн интерполяцию между вектором варьируемого параметра и вектором размера, при котором емкость равна заданному значению. Для этого используем встроенную в *Scilab* функцию *splin*.

Выполняем расчет по индивидуальному заданию. Вычисляем значение диапазона $L_1 – L_2$, при котором *R* находится в промежутке $R_1 – R_2$, а также выполняем графическую интерпретацию результата.

2.4 Графическая схема алгоритма решения задачи и ее описание



Рассмотрим подробнее представленную схему (рисунок 2.2):

– вводим начальные значения для расчета: относительная диэлектрическая проницаемость среды; параметр проницаемости: *ε0=0.885*; емкость; расстояние между дисками (номер символа 1);

– определяем функцию емкости вида (2.2) при помощи встроенной функции *deff,* а также определяем уравнение (2.3), при решении которого определим размер *R*, при котором значение емкости равно заданному значению (номер символа 2);

– решаем полученное уравнение вида (2.3) при помощи встроенной функции *fsolve* (номер символа 3);

– решаем полученное уравнение вида (2.3) при помощи численного метода половинного деления (номер символа 4);

– выполняем графическую интерпретацию: строим график функции вида (2.2) и отмечаем на нем найденные значения размера (номер символа 5);

– осуществляется изменение варьируемого параметра. Для каждого значения варьируемого параметра *LL* (расстояние между дисками) решаем уравнение вида (2.3) при помощи встроенной функции *fsolve* (номер символа 6 – 10);

– строим сводный график между вектором варьируемого параметра *LL* и вектором значений размера *RR*, при котором значение емкости равно заданному значению (номер символа 11);

– выполняем сплайн интерполяцию между векторами *LL* и *RR* при помощи встроенной функции *splin*, а также строим график полученной аппроксимирующей функции (номер символа 12);

– выполняем индивидуальное задание. Вычисляем значение диапазона L_1 – L_2, при котором *R* находится в промежутке R_1 – R_2, а также выполняем графическую интерпретацию результата (номер символа 13).

3 Описание задания в пакете *Scilab*

3.1 Описание модели в пакете *Scilab*

Для реализации задачи необходимо ввести исходные данные (рисунок 3.1).

```
//Исходные данные
eps0=0.885; // пФ/мм
eps1=2;     // Относительная диэлектрическая проницаемость
L=1;        // Расстояние между дисками, мм
c=33;       // Электрическая емкость, пФ
t=1:0.01:3; // Время исследования
```

Рисунок 3.1 – Исходные данные к проекту

При помощи встроенной функции определяем уравнение *C(R)=0*, где *C(R)* определяется формулой (2.2) (рисунок 3.2).

```
// Определим исходную функцию
function z=C(R)
 z=eps0*eps1*R*((%pi*R)/L+log((16*%pi*R)/L)-1)-c;
endfunction
```

Рисунок 3.2 – Определение функции *C*(*R*)

Решаем полученное уравнение при помощи встроенной функции *fsolve*. В результате получили, что при значении радиуса *R=1.932544* мм емкость равна заданному значению, равному 33 пФ. Графическая интерпретация представлена на рисунке 3.3.

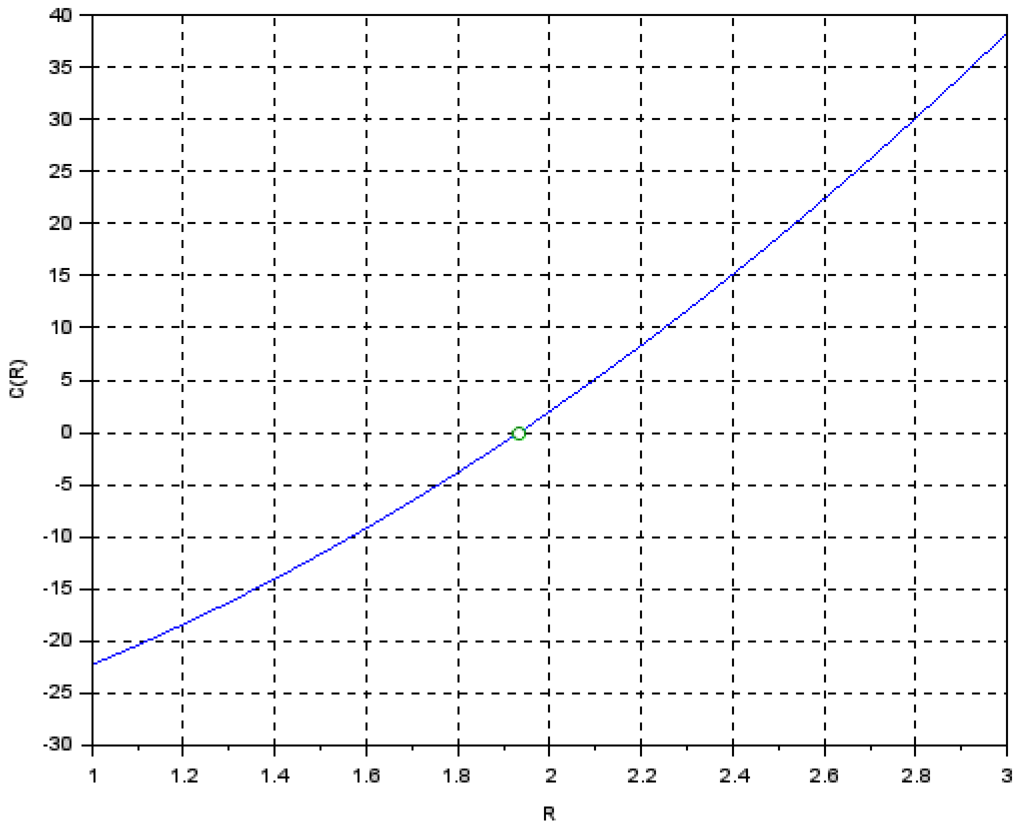


Рисунок 3.3 – Графическая интерпретация

Затем определяем функцию, в которой реализуется численный метод половинного деления (рисунок 3.4) и решаем уравнение вида (2.3) при помощи данного метода.

```
//Реализация численного метода половинного деления
Xn=1; Xk=3;eps=0.001;
while (abs(Xk-Xn)>eps)
  x=(Xn+Xk)/2;
  if(C(Xk)*C(x)<0) Xn = x; else Xk = x; end
  Rr=(Xk+Xn)/2;
end
```

Рисунок 3.4 – Реализация численного метода в системе *Scilab*

В результате получили, что при значении радиуса *R=1.9321289* мм емкость равна заданному значению, равному 33 пФ.

Графическая интерпретация расчета представлена на рисунке 3.5.

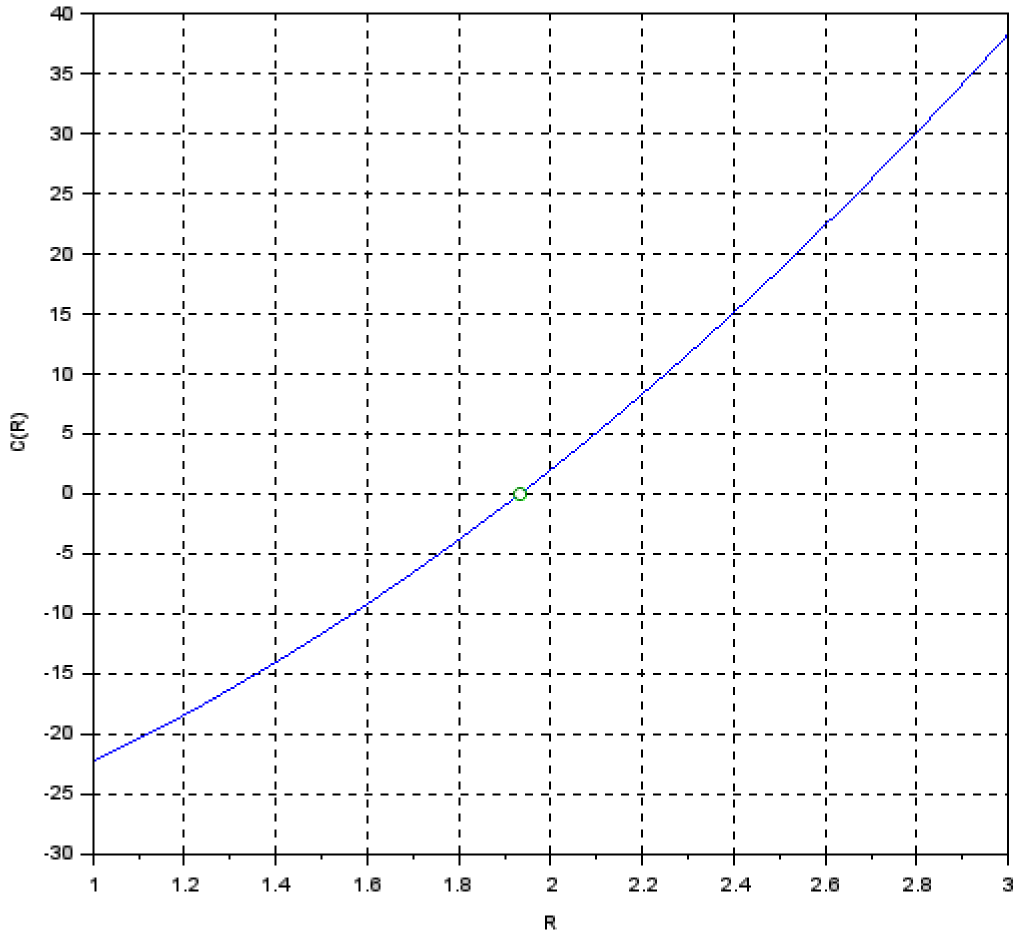


Рисунок 3.5 – Графическая интерпретация

Таким образом, различными способами (численным методом половинного деления и при помощи встроенной функции *fsolve*, причем значения совпадают с заданной точностью) найдено значение радиуса *R=1.932* мм при котором емкость равна заданному значению, равному 33 пФ. Выполнена графическая интерпретация результата расчета.

3.2 Описание исследований

Для исследования влияния значений расстояния между дисками на вид функции емкости, необходимо решить уравнение *C(R)=0*, где *C(R)* определяется формулой (2.2) для различных значений варьируемого параметра в диапазоне его изменения.

Задаем начальное и конечное значения расстояния между дисками, а также шаг изменения варьируемого параметра (рисунок 3.6).

```
LL=2.0:0.4:4.0;
disp('Вектор варьируемого параметра: '); disp(LL);
```

Рисунок 3.6 – Определение значений варьируемого параметра *LL*

Затем после определения вектора варьируемого параметра (*LL*) переопределяем уравнение вида (2.3) и решаем при помощи встроенной функции (приложение А).

В результате получили два вектора: вектор варьируемого параметра (*LL*) и вектор радиусов, при которых емкость равна заданному значению 33 пФ (рисунок 3.7).

```
-->LL'          -->RR
 ans  =          RR  =

  2.             2.5803714
  2.4            2.780781
  2.8            2.9613023
  3.2            3.1264137
  3.6            3.2791751
  4.             3.4217804
```

Рисунок 3.7 – Исследуемые вектора *LL* и *RR*

После этого строим график зависимости исследуемых векторов (рисунок 3.8).

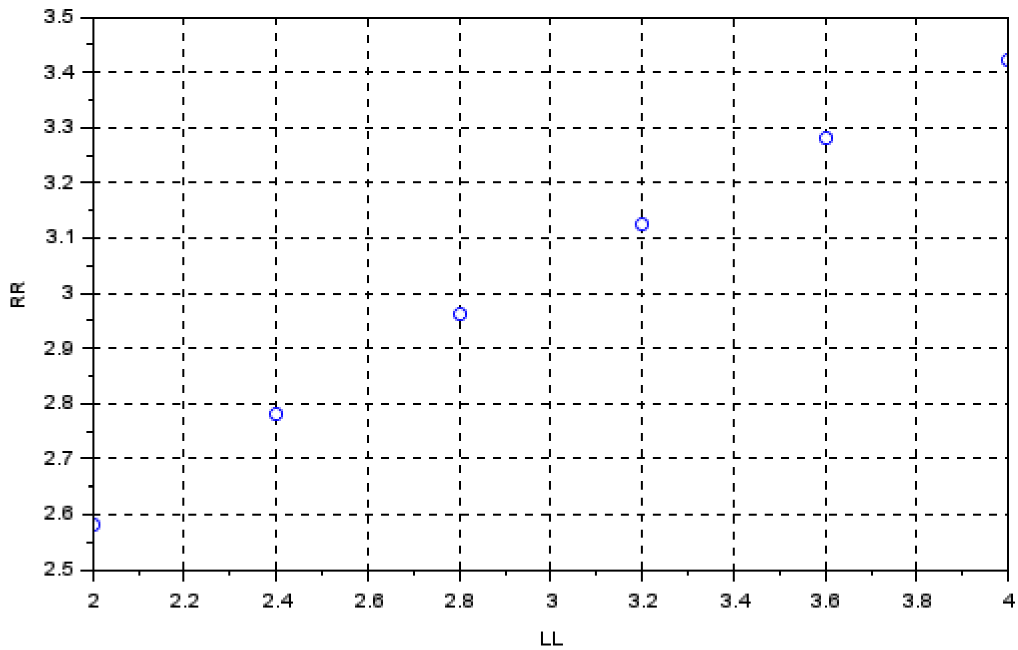


Рисунок 3.8 – График зависимости между векторами *LL* и *RR*

Чтобы установить функциональную зависимость влияния варьируемого параметра, необходимо провести интерполяцию значений при помощи функции *splin*.

График зависимости между векторами *LL* и *RR* и интерполирующая эту зависимость функция, представлены на рисунке 3.9.

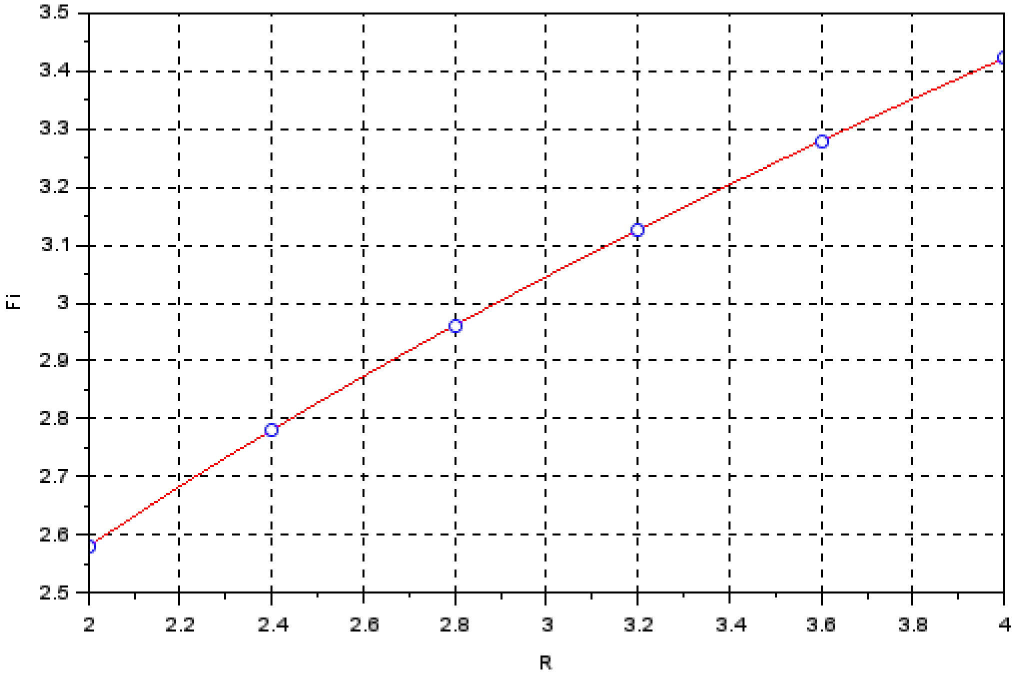


Рисунок 3.9 – График интерполирующей функции

Вычислим значение диапазона $L_1 - L_2$, при котором R находится в промежутке $R_1 - R_2$ (значения $R_1=2.8$ и $R_2=3.3$ задали с клавиатуры). В результате получили, что при $R_1=2.8$ и $R_2=3.3$ значение L находится в промежутке [2.44;3.65].

Графическая интерпретация результата решения представлена на рисунке 3.10.

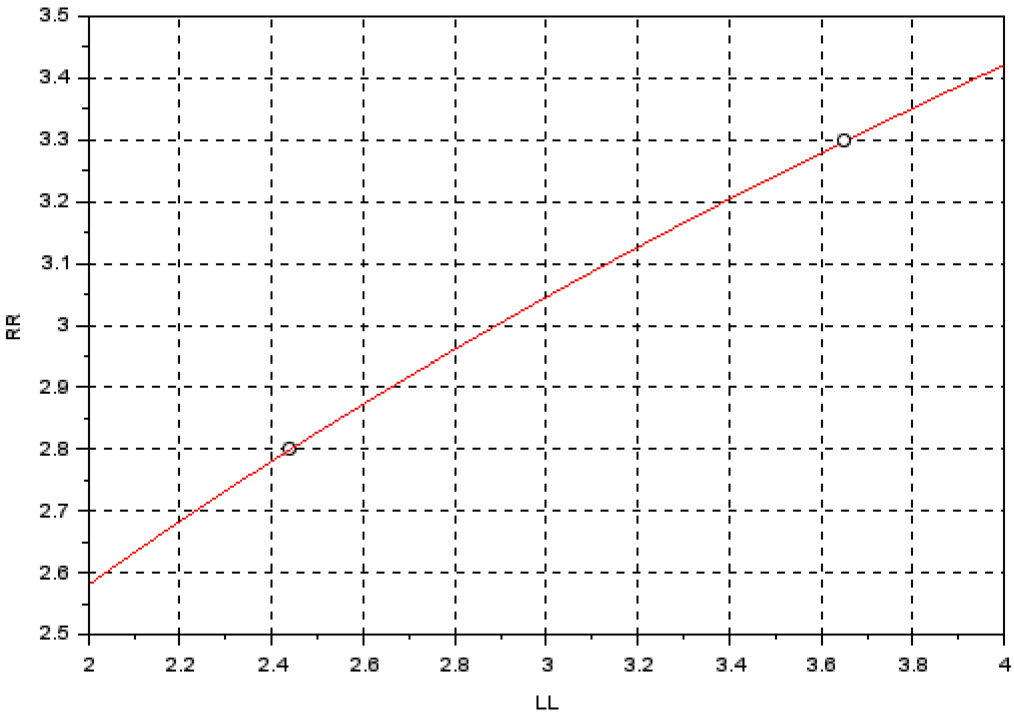


Рисунок 3.10 – Графическая интерпретация

3.3 Выводы по полученным результатам

Таким образом, различными способами (численным методом половинного деления и при помощи встроенной функции *fsolve*, причем значения совпадают с заданной точностью) найдено значение радиуса $R=1.932$ мм при котором емкость равна заданному значению 33 пФ. Выполнена графическая интерпретация результата расчета.

Выполнены исследования зависимости емкости от варьированного параметра L. Для заданного диапазона изменения параметра рассчитаны соответствующие величины. Получили, что при увеличении варьируемого параметра значения радиуса увеличивается.

Полученный результат представлен в численном и графическом видах. Для полученных экспериментальных значений выполнена сплайновая интерполяция с помощью функции *splin*, а также построен график интерполирующей функции (приложение Б).

Выполнено индивидуальное задание. В результате его выполнения получили, что при $R_1=2.8$ и $R_2=3.3$ значение L находится в промежутке [2.44;3.65].

Заключение

В курсовой работе изучены и приведены теоретические вопросы, связанные с математическим моделированием, численным решением алгебраических уравнений и интерполяцией экспериментальных данных. Описаны средства пакета символьных вычислений *Scilab*, предоставляемые для реализации математических моделей.

Во второй главе приведена постановка задачи и построена математическая модель, описывающая параметры емкости. На основании математической модели разработан алгоритм ее реализации в пакете *Scilab* и алгоритм исследования модели при различных значениях варьируемого параметра.

В третьей главе описана реализация модели и ее исследования в системе *Scilab*. Приведены полученные численные результаты и графические зависимости. На основании проведенных исследований сделаны общие выводы и приведены полученные численные значения. Получили, что при увеличении варьируемого параметра значения радиуса увеличивается.

В данной курсовой работе были выполнены следующие задачи:

– найден радиус *R*, соответствующий требуемой емкости, вычисленный при помощи встроенной функции *fsolve* и при помощи численного метода;

– проведено исследование влияние варьируемого параметра (расстояния между дисками) на значения радиуса диска;

– выполнена сплайн-интерполяция. Построен график исходной и интерполирующей функций на одном поле;

– выполнено индивидуальное задание.

На сегодняшний день такое сочетание вычислительных технологий и теоретических навыков студентов является основополагающим курсом для всех электротехнических, энергетических, электронных и многих других специальностей ВУЗов, которые в будущем столкнутся с ещё более совершенными информационными системами.

В процессе выполнения и оформления работы были использованы такие пакеты как *Scilab*, *Microsoft Word*.

Поставленные задачи в курсовой работе решены в полном объеме.

Список используемых источников

1 Звонарев, С. В. Основы математического моделирования: учебное пособие / С.В. Звонарев. – Екатеринбург: Изд-во Урал. ун-та, 2019г. – 112с.

2 Аюпов, В.В. Математическое моделирование технических систем: учебное пособие / В.В. Аюпов; М-во с.-х. РФ, федеральное гос. бюджетное образов. учреждение высшего образования «Пермская гос. с.-х. акад. им. акад. Д.Н. Прянишникова». – Пермь: ИПЦ «Прокростъ», 2017г. – 242с.

3 Зинина, А.И. Численные методы линейной и нелинейной алгебры: Учебное пособие / А.И. Зинина, В.И. Копнина. – Саратов, 2016г. – 48с.

4 Алексеев, Е. Р. Чеснокова О. В., Рудченко Е. А. Scilab: Решение инженерных и математических задач – М.: БИНОМ, 2008г. – 269с.

Приложение А

Программный режим

```
funcprot(0); clear;
//Исходные данные
eps0=0.885; // пФ/мм
eps1=2;    // Относительная диэлектрическая проницаемость среды
L=1;       // Расстояние между дисками, мм
c=33;      // Электрическая емкость, пФ
t=1:0.01:3; // Время исследования
disp('Исходные значения:');
disp('EPS0 = '+string(eps0));
disp('Относительная диэлектрическая проницаемость среды: EPS = '+string(eps1));
disp('Расстояние между дисками: L = '+string(L));
disp('Электрическая емкость: C = '+string(c));
// Определим исходную функцию
function z=C(R)
 z=eps0*eps1*R*((%pi*R)/L+log((16*%pi*R)/L)-1)-c;
endfunction
// Вычислим значение радиуса
r=fsolve(2,C);
disp('Задание 1. Решение уравнения встроенной функцией = '); disp(r);
// Докажем графически, что корень найден верно
figure(1); plot(t,C,r,C(r),'o');
xtitle('График функции C(R)', 'R', 'C(R)'); xgrid;
//Реализация численного метода половинного деления
Xn=1; Xk=3;eps=0.001;
while (abs(Xk-Xn)>eps)
  x=(Xn+Xk)/2;
  if(C(Xk)*C(x)<0) Xn = x; else Xk = x; end
  Rr=(Xk+Xn)/2;
end
//Вывод результатов
disp('Задание 2. Решение уравнения методом половинного деления = ');disp(Rr);
// Докажем графически, что корень найден верно
figure(2); plot(t,C,Rr,C(Rr),'o');
xtitle('График функции C(R)', 'R', 'C(R)'); xgrid;
disp('Задание 3');
LL=2.0:0.4:4.0;
disp('Вектор варьируемого параметра: '); disp(LL);
// Реализация опытов для варьируемого параметра
for i=1:length(LL)
 deff('[z]=f(R)','z=eps0*eps1*R*((%pi*R)/LL(i)+log((16*%pi*R)/LL(i))-1)-c');
 RR(i)=fsolve(0.3,f);
 disp('Опыт ' + string(i)+': для параметра LL('+string(i)+') = '+string(LL(i)));
 disp('Корень = ' + string(RR(i)));
end
// Построим сводный график
figure(3); plot(LL,RR','o'); xtitle('График исследуемых векторов', 'LL', 'RR'); xgrid;
```

```
// Сплайновая интерполяция
disp('Задание 4');
tt=2.0:0.01:4.0;
d=splin(LL,RR');
disp('Значения сплайна: ');disp(d);
Fi=interp(tt,LL,RR',d);
// Строим график полученной функции
figure(4); plot(tt,Fi,'r',LL,RR','o');
xtitle('Сплайновая интерполяция', 'R', 'Fi'); xgrid;
// Индивидуальное задание
R_1=input(' Введите начальное значение = ');;
R_2=input(' Введите конечное значение = ');;
disp('Промежуток для R: ['+string(R_1)+';'++string(R_2)+']');
i=1;p1=0;p2=0;
while i<=length(Fi)
 if Fi(i)<=R_1  p1=p1+1; end
 if Fi(i)<=R_2  p2=p2+1; end
i=i+1;
end
disp('Получили L: ['+string(tt(p1))+';'++string(tt(p2))+']');
//Графическая интерпретация
figure(5);
plot(tt,Fi,'r',tt(p1),R_1,'rok');plot(tt,Fi,'r',tt(p2),R_2,'rok');
xtitle('Графическая интерпретация', 'LL', 'RR'); xgrid;
```

Приложение Б

Командный и графический режимы

Исходные значения:
EPS0 = 0.885
Относительная диэлектрическая проницаемость среды: EPS = 2
Расстояние между дисками: L = 1
Электрическая емкость: C = 33
Задание 1. Решение уравнения встроенной функцией = 1.932544

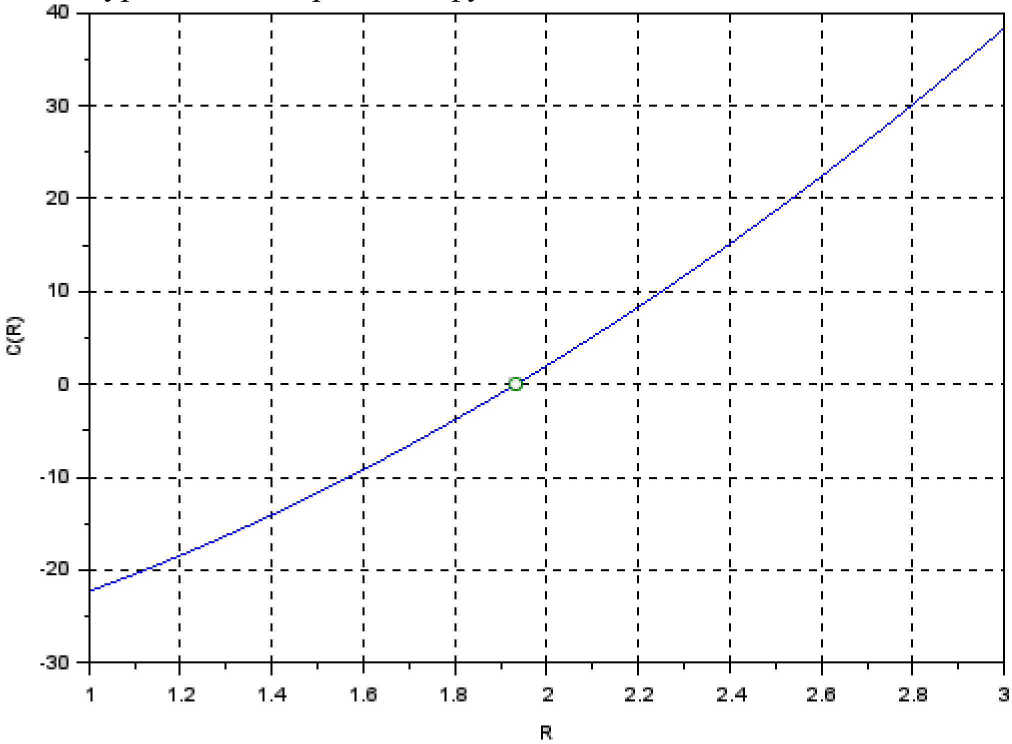


Рисунок Б1 – Графическая интерпретация

Задание 2. Решение уравнения методом половинного деления = 1.9321289

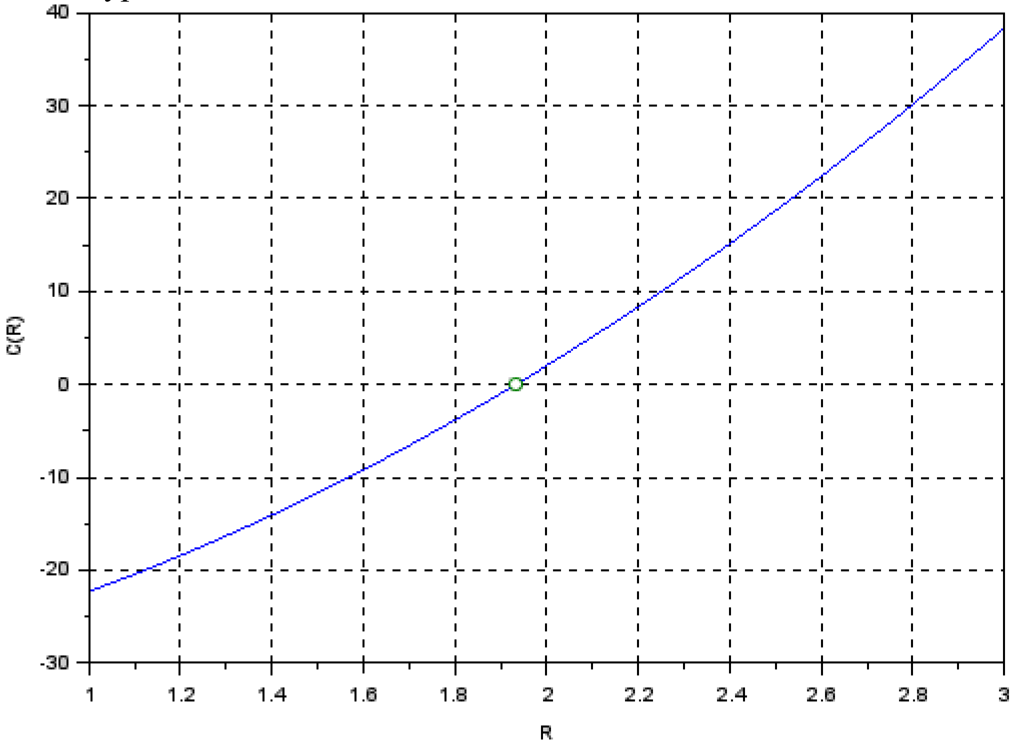


Рисунок Б2 – Графическая интерпретация

Задание 3. Вектор варьируемого параметра: 2. 2.4 2.8 3.2 3.6 4.
Опыт 1: для параметра LL(1) = 2 Корень = 2.5803714
Опыт 2: для параметра LL(2) = 2.4 Корень = 2.780781
Опыт 3: для параметра LL(3) = 2.8 Корень = 2.9613023
Опыт 4: для параметра LL(4) = 3.2 Корень = 3.1264137
Опыт 5: для параметра LL(5) = 3.6 Корень = 3.2791751
Опыт 6: для параметра LL(6) = 4 Корень = 3.4217804

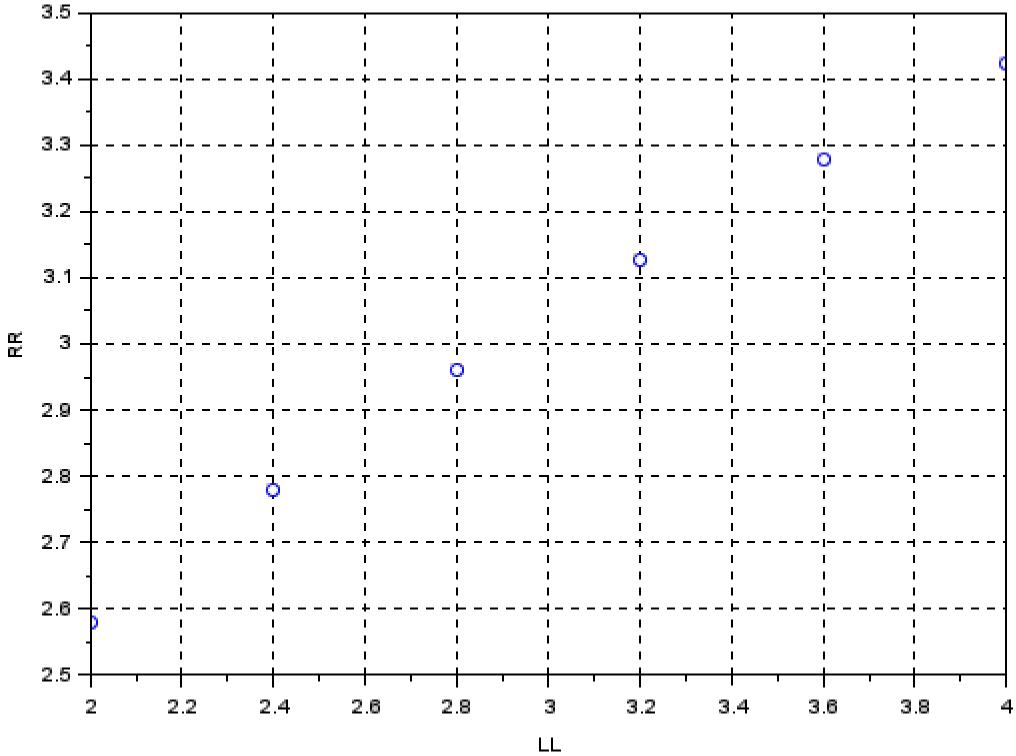


Рисунок Б3 – График зависимости между векторами

Задание 4. Значения сплайна:
d = 0.529884 0.4741638 0.430442 0.3963134 0.3683510 0.3455327

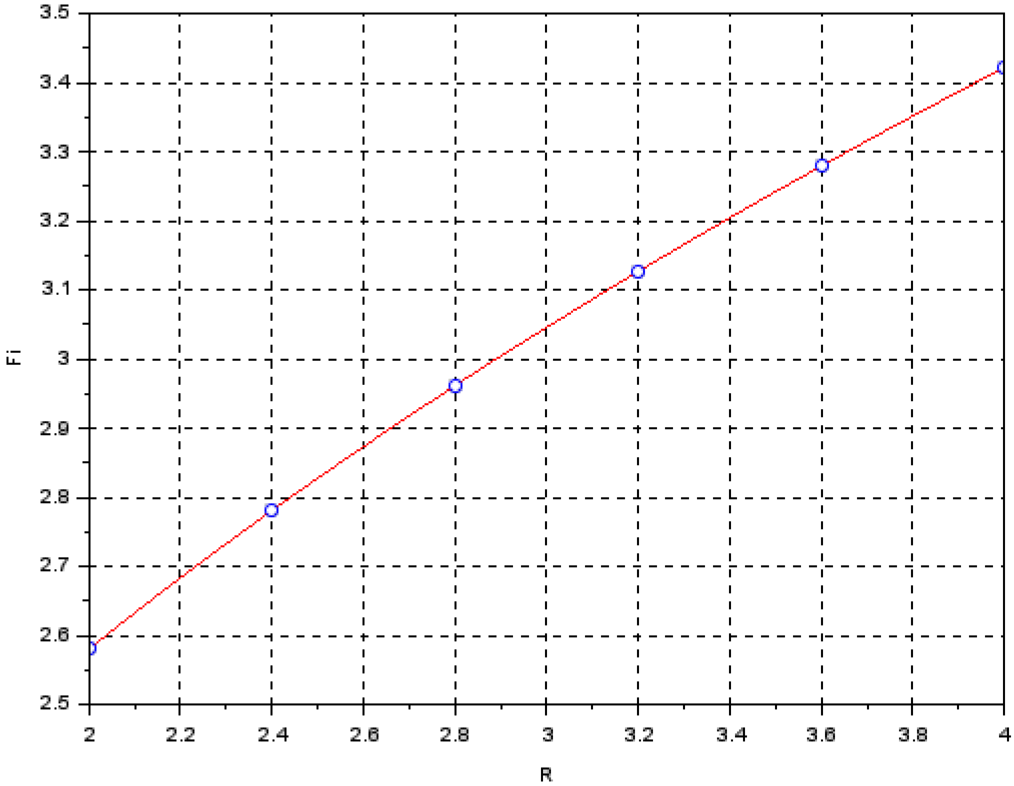


Рисунок Б4 – Сводный график зависимости и интерполирующая функция

Введите начальное значение = 2.8
Введите конечное значение = 3.3
Промежуток для R: [2.8;3.3]
Получили L: [2.44;3.65]

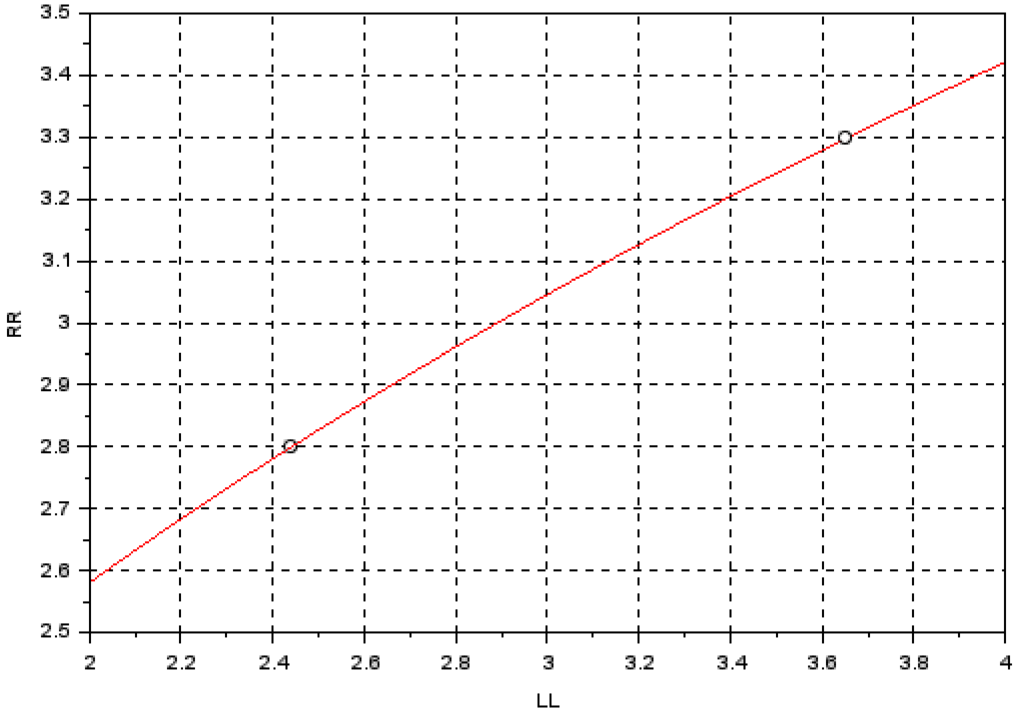


Рисунок Б5 – Графическая интерпретация