



Министерство образования Республики Беларусь

Учреждение образования
«Гомельский государственный технический
университет имени П. О. Сухого»

Кафедра «Информатика»

РЕШЕНИЕ ИНЖЕНЕРНЫХ ЗАДАЧ В СИСТЕМЕ КОМПЬЮТЕРНОЙ МАТЕМАТИКИ SCILAB

**ПРАКТИКУМ
по курсу «Информатика»
для студентов технических специальностей
дневной и заочной форм обучения**

Гомель 2018

УДК 681.3.06(075.8)
ББК 32.81я73
Т76

*Рекомендовано научно-методическим советом
факультета автоматизированных и информационных систем
ГГТУ им. П. О. Сухого
(протокол № 8 от 23.03.2017 г.)*

Рецензент: доц. каф. «Информационные технологии» ГГТУ им. П. О. Сухого
канд. техн. наук, доц. *В. В. Комраков*

Трохова, Т. А.
Т76 Решение инженерных задач в системе компьютерной математики Scilab : практикум по курсу «Информатика» для студентов техн. специальностей дневн. и заоч. форм обучения / Т. А. Трохова. – Гомель : ГГТУ им. П. О. Сухого, 2018. – 79 с. – Систем. требования: PC не ниже Intel Celeron 300 МГц ; 32 Mb RAM ; свободное место на HDD 16 Mb ; Windows 98 и выше ; Adobe Acrobat Reader. – Режим доступа: <https://elib.gstu.by>. – Загл. с титул. экрана.

Содержит краткие теоретические сведения, методику решения инженерных задач Scilab, примеры алгоритмов и программ и задания для самостоятельной работы.
Для студентов технических специальностей дневной и заочной форм обучения.

УДК 981:3.06 (075.8)
ББК 32.81я73

© Учреждение образования «Гомельский
государственный технический университет
имени П. О. Сухого», 2018

Оглавление

1. Работа с одномерными массивами при решении инженерных задач	4
1.1. Краткие теоретические сведения	4
1.2. Задача о гидравлическом демпфере	6
1.3. Задача о резонансе динамической системы	8
1.4. Вычисление параметров плоских сечений при выполнении расчетов на прочность	9
2. Вычисление параметров технических объектов с использованием численных методов	12
2.1. Краткие теоретические сведения	12
2.2. Задача о кулачковом механизме	15
2.3. Задача о шарнирном четырехзвеннике	21
2.4. Задача о движении точки по траектории	24
3. Вычисление параметров динамических объектов, описанных дифференциальными уравнениями и системами	27
3.1. Краткие теоретические сведения	27
3.2. Задача об электрической схеме	29
3.3. Задача о механической динамической системе	33

1. РАБОТА С ОДНОМЕРНЫМИ МАССИВАМИ ПРИ РЕШЕНИИ ИНЖЕНЕРНЫХ ЗАДАЧ

1.1. Краткие теоретические сведения

В разделе рассматриваются инженерные задачи, при программировании которых используются следующие базовые алгоритмы и приемы обработки информации: обработка одномерных массивов; табулирование функции; поиск экстремума и суммы, построение двумерных графиков. Теоретические сведения по этим направлениям приведены ниже.

В Scilab можно использовать различные типы структурированных данных, т.е. данных, содержащих несколько элементов. К основным структурированным данным относятся массивы чисел (вектора, матрицы, многомерные массивы).

Массив - это последовательность однотипных элементов, снабженных индексами (порядковыми номерами). Вектором принято называть одномерный массив. Матрица - это двумерный массив. Векторы в Scilab делятся на векторы-строки и векторы-столбцы.

Занести числа в вектор можно несколькими способами.

1) Непосредственный ввод

Чтобы задать вектор-строку, значения его элементов следует перечислить в квадратных скобках, разделяя пробелами.

Например,

$V=[2\ 3\ 8\ 0]$

2) Ввод с использованием диапазона

Общий вид диапазона:

$x_n : dx : x_k$

x_n – начальное значение диапазона;

dx – шаг изменения значений диапазона;

x_k – конечное значение диапазона;

Для формирования вектора следует задать:

$X=x_n:dx:x_k$

В результате будет сформирован вектор, первый элемент которого равен x_n , второй – x_n+dx , третий - $x_n+dx+dx$ и т.д. Последний элемент будет не больше x_k для положительного шага dx , и не меньше x_k – для отрицательного. Если величина шага отсутствует, то по умолчанию его значение равно 1 или -1, тогда вид диапазона таков:

$X=x_1: x_k$

Задачу о табулировании функции можно решить с помощью формирования векторов X и Y , например:

$X=0:0.5:6.28$

$Y=\cos(X)$

В этом случае в каждый элемент вектора Y поместится косинус от каждого элемента вектора X .

Над массивами можно выполнять различные операции, заданные системными функциями.

$\max(A)$ - вычисление максимального элемента массива;

$\min(A)$ - вычисление минимального элемента массива;

$\text{sum}(A)$ - вычисление суммы элементов массива;

$\text{prod}(A)$ - вычисление произведения элементов массива;

$\text{mean}(A)$ - вычисление среднего значения элементов массива.

$[\text{Amax}, \text{Nmax}] = \max(A)$ - вычисление максимального элемента в массиве и его номера.

Графические объекты в Scilab строятся в специальном графическом окне (**figure**). Одновременно может быть открыто несколько таких окон, каждому из которых присваивается номер. Для перехода к имеющемуся окну с номером N или открытия нового графического окна необходимо ввести команду **figure(N)**. Кроме того, первое обращение к графической команде автоматически вызывает появление графического окна, которому присваивается первый номер.

Для построения графиков функций одной переменной в декартовой системе координат используются различные формы команды **plot**, которая рисует графики функций по ряду точек, соединяя их отрезками прямых.

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

Команды **xgrid ()** позволяют задавать построение сетки на поле графика.

Заголовок графика и надписи осей графика можно вывести с помощью команды

xtitle(title,xstr,ystr)

где **title** — символьная константа, содержащая название графика;

xstr — символьная константа, содержащая название оси X ;

ystr — символьная константа, содержащая название оси Y .

Идентификацию кривых графика (создание легенды) можно выполнить с использованием команды legend, например,
legend("График Y(X)", "График Y1(X1)").

1.2. Задача о гидравлическом демпфере

Задание

- 1) Вычислить множество значений функции перемещения поршня гидравлического демпфера.
- 2) Найти минимальную высоту подъема поршня и время, при котором высота подъема поршня максимальна.

Постановка задачи

Гидравлический демпфер представляющего собой поршень массой m , движущийся в жидкости, в зависимости от времени по формуле:

$$y(t) = y_0 \cdot e^{-n \cdot t} \cdot \cos(\sqrt{p^2 - n^2} \cdot t) \quad (1)$$

где n - приведенный коэффициент вязкого сопротивления, вычисляется по формуле

$$n = [4\pi\mu H / (mz)] \cdot (D / d)^4$$

Частота собственных колебаний p вычисляется по формуле:

$$p = \sqrt{c / m}$$

Исходные данные

Для исследования движения поршня необходимо знать следующие исходные данные:

$y_0 = 0.05$ – отклонение поршня от положения равновесия в начальный момент времени;

$c = 3000$ Н/м – жесткость пружины;

$D = 0.1$ м – диаметр цилиндра;

$d = 0.01$ м – диаметр отверстия;

$z = 25$ – число отверстий;

$m = 3$ кг – масса поршня;

$H = 0.005$ м – высота поршня;

$\mu = 0.06$ Па*м – динамический коэффициент вязкости жидкости;

$t_k=3$ с – время исследования.

Алгоритм решения задачи

Ниже приведен алгоритм выполнения расчетов в системе Scilab .

1. Задать исходные данные для решения задачи.
2. Вычислить значения n и p .
3. Создать вектор t , изменяющий свои значения от 0 до t_k .
4. Создать вектор y по формуле (1)
5. Построить график функции $y(t)$ по двум векторам t и y .
6. С помощью функции `min` найти значение и номер элемента в векторе y .
7. По номеру элемента найти значение времени, соответствующее минимальной высоте подъема поршня.
8. Нанести полученную точку на график движения поршня, убедиться, что она найдена правильно.

Программа

Программа, реализующая данный алгоритм, приведена ниже.

```
// Исходные данные
m=1.5; H=0.005; c=3e3; D1=0.1; d=0.01; z=25; mu=6e-2; y0=0.05
// Расчет приведенного коэффициента вязкого сопротивления
// и частоты собственных колебаний демпфера
p=sqrt(c/m); n=4*%pi*mu*H/(m*z)*(D1/d)^4;
//Формирование вектора времени
t=0:0.01:3
//Формирование вектора перемещения
y=y0*exp(-n*t).*cos(sqrt(p^2-n^2).*t)
// Поиск координат точки минимума
[ymin,nmin]=min(y)
tmin=t(nmin)
disp(tmin)
disp(ymin)
//Построение графика
plot(t,y,tmin,ymin,'*')
xgrid
```

Результаты

0.07 – значение времени, при котором перемещение минимально.

- 0.0465991 – значение минимального перемещения.

График перемещения поршня гидравлического демпфера от времени приведен на рисунке 1.1.

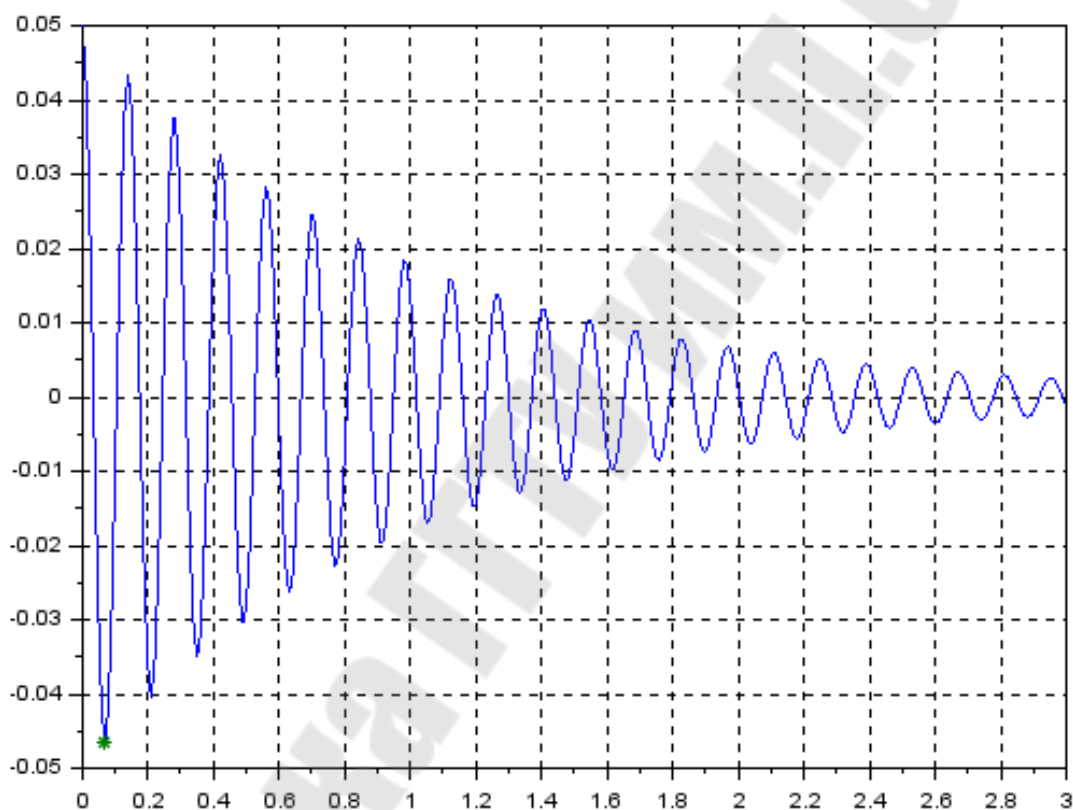


Рис. 1.1. График зависимости перемещения поршня гидравлического демпфера от времени

1.3. Задача о резонансе динамической системы

Задание

1) Вычислить множество значений функции амплитудно-частотной характеристики динамической системы.

2) Построить график зависимости амплитуды колебаний от частоты, нанести координатную сетку, выделить фоновыми линиями точку резонанса.

3) Найти максимальное значение амплитуды при резонансной частоте, нанести точку максимума на график, пометить ее маркером.

Постановка задачи

Для динамической колебательной системы, содержащей массу, пружину и демпфер, вычислить значения амплитуды колебаний H , если значения частоты f изменяются в заданных пределах, по формуле:

$$H = \frac{1/k}{\sqrt{(1 - (f/f_n)^2)^2 + (2sf/f_n)^2}},$$

где коэффициент затухания системы s вычисляется по формуле:

$$s = \frac{c}{2\sqrt{km}}$$

Собственная частота незатухающих колебаний системы рассчитывается по формуле:

$$f_n = \frac{1}{2\pi} \sqrt{\frac{k}{m}}$$

Резонансная частота системы (частота, при которой амплитуда колебаний достигает максимума) вычисляется по формуле:

$$f_r = f_n \sqrt{1 - 2s^2}$$

Исходные данные

В задаче используются следующие исходные данные: m – масса системы; k – коэффициент жесткости пружины; c – коэффициент демпфирования демпфера; f – текущее значение частоты. Числовые значения исходных данных приведены в табл.1.1.

Таблица 1.1

N варианта	m	c	k	Пределы изменения f
1	10	0.6	10	$f_r + 0.05$ $f_r - 0.05$
2	5	0.3	5	$f_r + 0.02$ $f_r - 0.02$
3	20	0.4	7	$f_r + 0.04$ $f_r - 0.04$
4	5	0.1	2	$f_r + 0.05$ $f_r - 0.05$
5	6	0.2	4	$f_r + 0.035$ $f_r - 0.035$

6	8	0.5	9	$fr + 0.055$ $fr - 0.055$
7	12	0.7	4	$fr + 0.025$ $fr - 0.025$

Задание для самостоятельного выполнения

Алгоритм и программу для решения задачи разработайте самостоятельно по аналогии с решением задачи 1.2.

1.4. Вычисление параметров плоских сечений при выполнении расчетов на прочность

Задание

- 1) Вычислить координаты центра тяжести сложной фигуры.
- 2) Вычислить центральные моменты инерции сложной фигуры.

Постановка задачи

Для предложенной задачи составить графическую схему алгоритма и написать программу. Исходные данные для расчетов подобрать самостоятельно.

Дана фигура (рисунок 1.2), которая может быть разбита на n прямоугольников высотой h_i и шириной b_i .

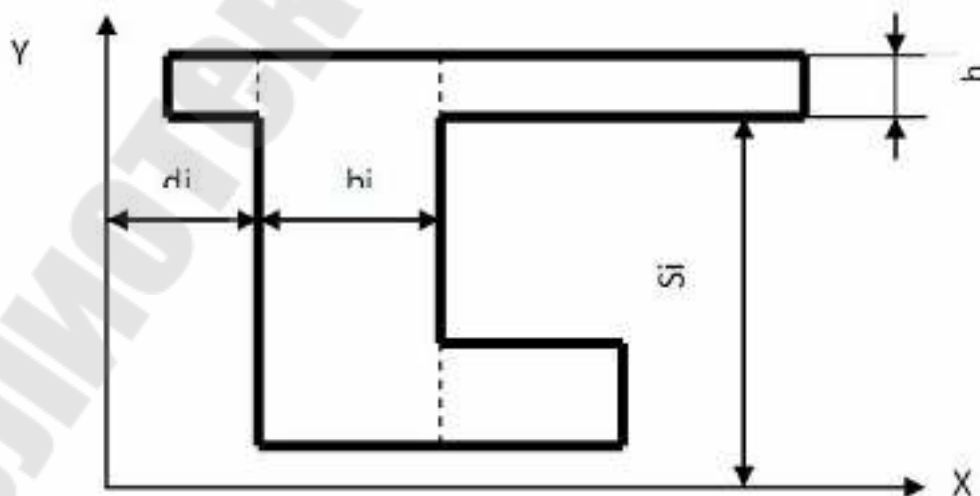


Рис.1.2. Заданное сечение

Оси X и Y проводятся так, чтобы они были параллельны сторонам рассматриваемого сечения. Расстояния от осей X и Y до ближайших сторон прямоугольников обозначаются соответственно S_i и d_i . Координаты центра тяжести сложной фигуры определяются по формулам:

$$X_c = \frac{S_y}{F} = \frac{\sum_{i=1}^n F_i \cdot X_i}{\sum_{i=1}^n F_i}; \quad Y_c = \frac{S_x}{F} = \frac{\sum_{i=1}^n F_i \cdot Y_i}{\sum_{i=1}^n F_i},$$

где S_x и S_y - статические моменты фигуры; F- площадь фигуры; F_i - площади отдельных простых частей; X_i и Y_i - координаты центров тяжести этих частей

$$X_i = d_i + \frac{b_i}{2}; \quad Y_i = S_i + \frac{h_i}{2}$$

Моменты инерции сечения относительно осей X и Y определяются как сумма моментов инерции отдельных прямоугольников. При этом можно воспользоваться формулами преобразования при параллельном переносе осей:

$$I_X = \sum_{i=1}^n \left[\frac{b_i \cdot h_i^3}{12} + b_i \cdot h_i \cdot \left(S_i + \frac{h_i}{2} \right)^2 \right];$$

$$I_Y = \sum_{i=1}^n \left[\frac{h_i \cdot b_i^3}{12} + b_i \cdot h_i \cdot \left(d_i + \frac{b_i}{2} \right)^2 \right];$$

$$I_{XY} = \sum_{i=1}^n h_i \left[\left(S_i + \frac{h_i}{2} \right) \cdot \left(d_i + \frac{b_i}{2} \right) \right] \cdot b_i.$$

Зная значения I_X , I_Y , I_{XY} , можно определить центральные моменты инерции:

$$I_{XC} = I_X - Y_c^2 \cdot F;$$

$$I_{YC} = I_Y - X_c^2 \cdot F;$$

$$I_{X'Y'} = I_{XY} - X_c \cdot Y_c \cdot F.$$

Исходные данные

d – вектор расстояний до оси Y;
b – вектор ширины частей сложной фигуры;
S – вектор расстояний до оси X;
h - вектор длины частей сложной фигуры.

Алгоритм решения задачи

Шаг 1.

Зададим вектора исходных данных, однозначно определяющих простые части сложной фигуры.

Шаг 2.

Вычисляем координаты центров тяжести частей сложной фигуры.

Шаг 3.

Вычисляем площади частей сложной фигуры.

Шаг 4.

Вычисляем координаты центра тяжести сложной фигуры. Выведем результаты расчетов в командную область.

Программа

```
//Ввод исходных векторов
d=[1 2 4.5 4.5]
b=[1 2.5 4.5 2.5]
S=[5 0.5 5 0.5]
h=[0.5 5 0.5 1.3]
//Координаты центров тяжести частей сложной фигуры
X=d+b/2
Y=S+h/2
//Площади частей сложной фигуры
F=b.*h
//Координаты центра тяжести сложной фигуры
Xc=sum(F.*X)/sum(F)
Yc=sum(F.*Y)/sum(F)
//Вывод координат центра тяжести сложной фигуры
disp(Xc)
disp(Yc)
```

Результаты

4.0675676 - координата X_c центра тяжести сложной фигуры

3.0094595 - координата Y_c центра тяжести сложной фигуры

Задание для самостоятельного выполнения

Для вычисления моментов инерции сечения относительно осей X и Y и центральных моментов инерции сложной фигуры доработайте алгоритм и программу самостоятельно.

2. ВЫЧИСЛЕНИЕ ПАРАМЕТРОВ ТЕХНИЧЕСКИХ ОБЪЕКТОВ С ИСПОЛЬЗОВАНИЕМ ЧИСЛЕННЫХ МЕТОДОВ

2.1. Краткие теоретические сведения

В разделе рассматриваются инженерные задачи, при программировании которых используются следующие базовые алгоритмы и приемы обработки информации: вычисление определенного интеграла методом трапеций; программирование операторов условия и цикла; организация операторов-функций и подпрограмм-функций.

В Scilab вычисление определенного интеграла методом трапеций реализовано функцией

inttrap(x,y),

где x – вектор значений аргумента подынтегральной функции на отрезке интегрирования, y – вектор значений, полученных при вычислении подынтегральной функции для элементов вектора x .

Например, для вычисления $\int_2^{5.3} \frac{2x dx}{\sin x + 1.5}$ нужно выполнить следующий набор команд:

Фрагмент программы	Результат
<pre>x =2:0.01:5.3 y =2*x./(sin(x)+1.5) int1= inttrap(x,y) disp(integral)</pre>	30.436962

Для программирования разветвляющихся алгоритмов в Scilab существуют оператор условия и оператор выбора.

Условный оператор представлен в нескольких формах и имеет следующий общий вид.

Полная форма 1

```
if ЛВ then  
    ОПЕРАТОР1  
    else ОПЕРАТОР2,  
end
```

Полная форма 2

```
if ЛВ  
    then ОПЕРАТОР1,  
elseif ЛВ2 then ОПЕРАТОР2,  
else ОПЕРАТОР,  
end
```

Краткая форма

```
if ЛВ then ОПЕРАТОР1  
end
```

Здесь

ЛВ, ЛВ1, ЛВ2, ЛВ3 – логические выражения;

ОПЕРАТОР1, ОПЕРАТОР2, ОПЕРАТОР – любые операторы или группы операторов языка Scilab.

Разделителями в операторах могут быть запятая или точка с запятой. Если ОПЕРАТОР расположен в следующей строке, то разделитель ставить не обязательно.

Порядок выполнения оператора условия следующий.

Полная форма 1

- Логическое выражение вычисляется до константы (0 или 1).
- Если логическое выражение истинно, то выполняется оператор, стоящий после слова **then**, а затем следующий за оператором **if** оператор.
- Если логическое выражение ложно, то выполняется оператор, стоящий после слова **else**, а затем следующий за оператором **if** оператор.

Краткая форма

- Логическое выражение вычисляется до константы (0 или 1).
- Если логическое выражение истинно, то выполняется оператор, стоящий после слова **then**, а затем следующий за оператором **if** оператор.
- Если логическое выражение ложно, то выполняется следующий за оператором **if** оператор.

Оператор цикла **for** предназначен для программирования циклических алгоритмов, когда переменная цикла явно выражена и изменяется от начального значения до конечного значения с постоянным шагом.

Общий вид оператора цикла **for**

```
for x= xn : dx : xk
      ОПЕРАТОР,
end
```

Здесь :

x – переменная цикла,

xn – начальное значение переменной цикла,

dx – шаг изменения переменной цикла,

xk – конечное значение переменной цикла,

ОПЕРАТОР – любой оператор или группа операторов рабочей части цикла.

Порядок выполнения оператора цикла:

- проверяется условие $xn \leq xk$;
- если условие не выполняется, то оператор цикла прекращает свою работу, и выполняется следующий за ним оператор программы;
- если условие выполняется, то переменной цикла присваивается ее начальное значение $x=xn$;
- выполняется оператор рабочей части цикла;
- до тех пор, пока $x \leq xk$ переменная цикла увеличивается на шаг dx и выполняется рабочая часть цикла.
- Для решения нелинейных уравнений в Scilab используется функция
- **fsolve(x0,f)**,
- где **x0** – начальное приближение корня, **f** – функция, описывающая левую часть уравнения $f(x)=0$.
- Например, для решения уравнения $\sin(2x) = \cos(3x^2) + \sin(3x)$ для начального приближения $x \approx 7$ нужно выполнить следующие команды:

Фрагмент программы	Результат
deff('y=F(x)', 'y=sin(2*x)-cos(3*x.^2)-sin(3*x)') root=fsolve(7,F)	6.9755674

disp(root)

Для решения систем нелинейных уравнений можно использовать функцию

fsolve(x0,f),

где **x0** – вектор начальных приближений для неизвестных, **f** – функция, определяющая систему

Например, решение системы $\begin{cases} 2 \cdot x + y = 6 \\ x^2 + y^2 = 14 \end{cases}$ можно выполнить

следующим образом:

Фрагмент программы	Результат
function [y]=fun(x)	
y(1)=2*x(1)+x(2)-6	1.2338096
y(2)=x(1)^2+x(2)^2-14	3.5323808
endfunction	
X0=[1;1]	
R=fsolve(X0,fun)	
disp(R)	

2.2. Задача о кулачковом механизме

Задание

1) Вычислить зависимости аналога ускорения роликового толкателя от времени. Построить график этой функции.

2) Вычислить функции скорости и перемещения роликового толкателя от времени. Построить графики этих функций.

Постановка задачи

Дан кулачковый механизм (рис. 2.1), график зависимости аналога ускорения от времени механизма приведен на рис.2.2.

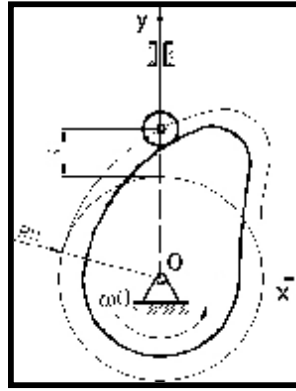


Рис. 2.1. Кулачковый механизм

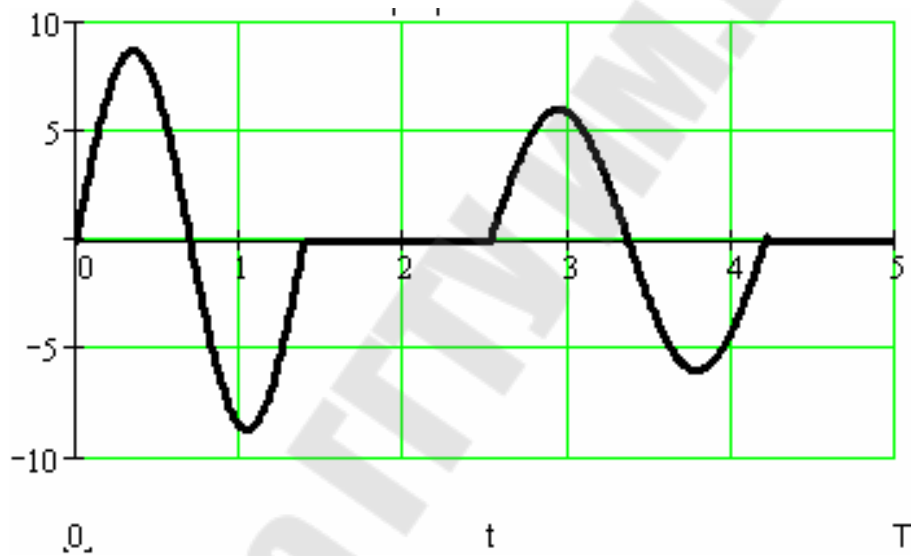


Рис. 2.2. Функция аналога ускорения роликового толкателя

Функция аналога ускорения роликового толкателя имеет вид:

$$S_{11}(t) = \begin{cases} a_1 \sin(b_1 \omega_0 t) & \text{при } 0 \leq t < t_1 \\ 0 & \text{при } t_1 \leq t < t_2 \\ a_2 \sin(b_2 \omega_0 t) & \text{при } t_2 \leq t < t_3 \\ 0 & \text{при } t_3 \leq t < \frac{2\pi}{\omega_0} \end{cases}$$

$$t_1 = \frac{2\pi}{b_1\omega_0} \quad t_2 = \frac{\varphi_2}{\omega_0} \quad t_3 = t_2 + \frac{2\pi}{b_2\omega_0}$$

$$b_1 = \frac{2\pi}{\varphi_1} \quad b_2 = \frac{2\pi}{\varphi_3 - \varphi_2}$$

Функция скорости роликового толкателя имеет вид:

$$S1(t) = \int_0^t \omega_0^2 S11(t) dt$$

Функция перемещения роликового толкателя имеет вид:

$$S(t) = \int_0^t S1(t) dt$$

Исходные данные

Исходными данными для решения задачи являются:

$\varphi_1, \varphi_2, \varphi_3, a_1, a_2$ – параметры закона изменения аналога ускорения толкателя кулачкового механизма;

ω_0 – угловая частота вращения кулачка;

$\nu=0.675$ – угол давления в кулачковом механизме.

Таблица 2.1. Таблица исходных данных

№ варианта	Параметры закона S					
	φ_1	φ_2	φ_3	a_1	a_2	ω_0
1	1,75	3,18	5,22	8,2	6,0	1,256
2	1,7	3,2	5,3	8,6	5,8	1,256
3	1,77	3,1	5,2	8,25	6,1	1,256
4	1,85	3,14	5,22	8,0	6,3	1,256
5	1,87	3,05	5,12	8,5	7,0	1,256
6	1,8	3,1	5,3	8,1	6,2	1,256

Алгоритм решения задачи

Шаг 1.

Присвоить имена исходным данным и поместить в них указанные числовые значения, например, φ_1 назвать $f1$, φ_2 – $f2$ и т.д.

Шаг 2.

Вычислить значения переменных, необходимых для разработки функции аналога ускорения роликового толкателя:

-b1,b2

-t1, t2, t3

Убедиться, что переменные приняли конкретные числовые значения.

Шаг 3.

Начинаем программировать вычисление функции аналога ускорения роликового толкателя, для этого создадим вектор времени, при котором рассчитывается эта функция. Он должен содержать набор чисел от 0 до tk с

шагом 0.01, $t_k = \frac{2\pi}{\omega_0}$, например:

T=0:0.01:tk

Шаг 4.

Теперь нужно каждому значению времени в векторе T поставить в соответствие значение функции аналога в зависимости от того, на каком временном промежутке оно вычисляется.

Для этого нужно организовать цикл, в котором переменная цикла будет принимать новые значения времени при каждом новом прохождении цикла.

Например, заголовок цикла можно представить так:

for t=0:0.01:2*%pi/w0

Шаг 5.

В рабочей части цикла нужно использовать оператор if для того, чтобы проверить, попадает ли текущее значение времени в заданный интервал, например,

if (t>=0)&(t<=t1),S11(i)=a1*sin(b1*w0*t); end;

Здесь знак & означает операцию «И».

Переменная i предназначена для того, чтобы управлять номером элемента в векторе S11 аналога ускорения, поэтому при каждом выполнении рабочей части цикла ее значение должно увеличиваться на 1, например, i=i+1.

Шаг 6.

После выполнения цикла в памяти должны находиться два вектора – вектор времени и вектор аналога ускорения, по которым строим график функции с помощью оператора plot.

Шаг 7.

Организуем цикл для вычисления определенного интеграла с помощью функции `inttrap`. Получим вектор интеграла `S1` и построим график функции аналога скорости в зависимости от времени.

Программа

```
clear
fi1=1.745; fi2=3.18; fi3=5.29;
a1=8.75; a2=6; w0=1.256;v=0.675
b1=2*%pi/fi1; b2=2*%pi/(fi3-fi2);
t1=2*%pi/(b1*w0);
t2=fi2/w0;
t3=t2+2*%pi/(b2*w0); i=0
for t=0:0.01:2*%pi/w0,
    i=i+1;
    if (t>=0)&(t<=t1),S11(i)=a1*sin(b1*w0*t); end;
    if (t>t1)&(t<=t2),S11(i)=0;end;
    if (t>t2)&(t<=t3),S11(i)=a2*sin(b2*w0*t); end;
    if (t>t3)&(t<2*%pi/w0),S11(i)=0; end;
end;
figure(1);
T=0:0.01:2*%pi/w0;
figure(1)
plot(T,S11)
xgrid
i=0
for t=0:0.01:2*%pi/w0,
    i=i+1;
S1(i)=inttrap(T(1:i),S11(1:i))*w0^2
end
S1=S1'
figure(2);
plot(T,S1)
xgrid
```

Результаты

На рис. 2.3 приведен график зависимости аналога ускорения роликового толкателя кулачкового механизма в зависимости от времени.

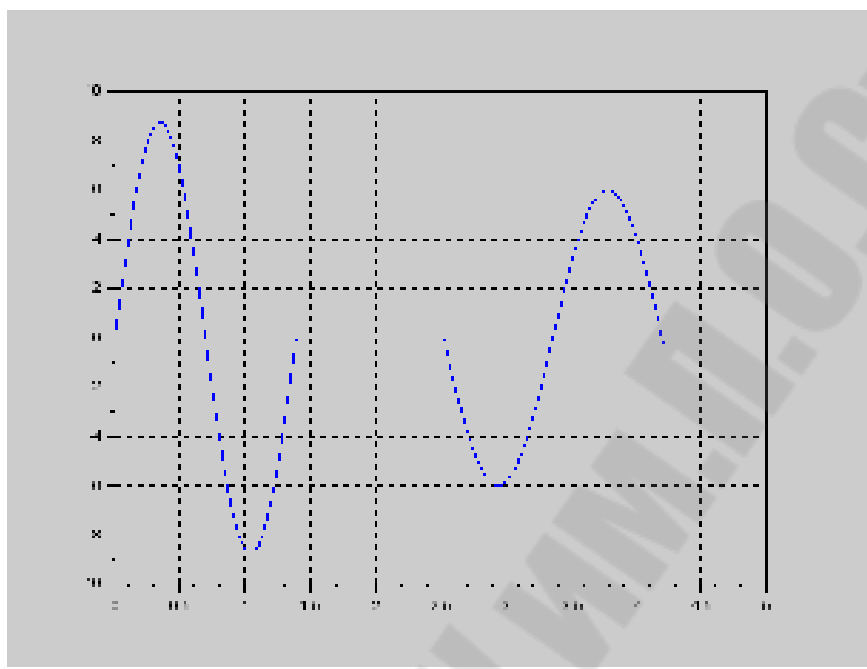


Рис. 2.3. График зависимости аналога ускорения роликового толкателя кулачкового механизма от времени

На рис. 2.4 приведен график зависимости аналога скорости роликового толкателя кулачкового механизма в зависимости от времени.

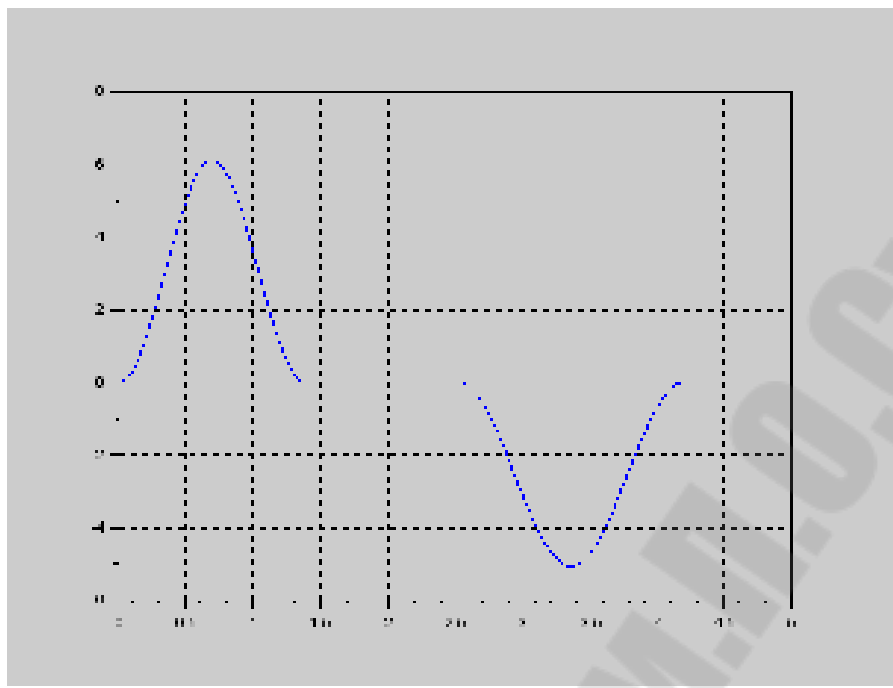


Рис. 2.4. График зависимости аналога скорости роликового толкателя кулачкового механизма от времени

Задание для самостоятельного выполнения

Алгоритм и программу для вычисления функции скорости и перемещения роликового толкателя от времени разработать самостоятельно. Построить графики этих функций

2.3. Задача о шарнирном четырехзвеннике

Задание

Рассчитать длины звеньев шарнирного кривошипно-коромыслового четырехзвенника.

Постановка задачи

На рис. 2.5 приведена схема шарнирного четырехзвенника.

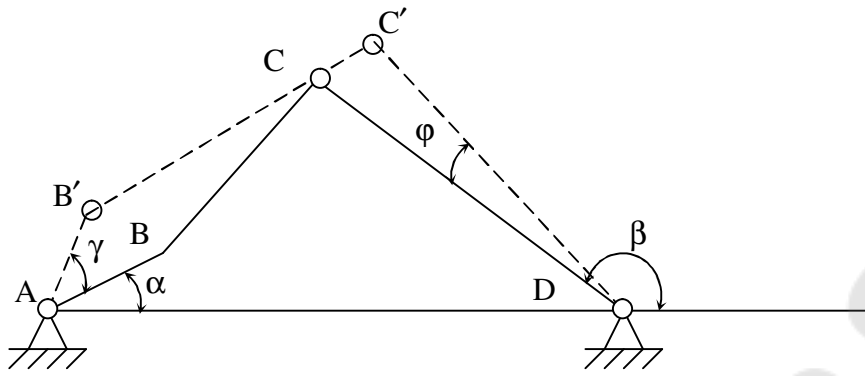


Рис. 2.5. Схема шарнирного четырехзвенника

Закон движения шарнирного четырехзвенника описывается функцией вида:

$$\psi(\phi) = k_1 \cdot \sin(\phi) + k_2$$

Параметры механизма вычисляются по формулам:

$$a = 1 / p_1 \quad c = -1 / p_0$$

$$b = \sqrt{a^2 + c^2 + 1 - 2cap_2} \quad (1)$$

где параметры p_i ($i=0,1,2$) - неизвестные в системе линейных уравнений вида:

$$p_0 \cdot \cos(\phi_1 + \alpha) + p_1 \cdot \cos(\psi(\phi_1) + \beta) + p_2 = \cos(\phi_1 + \alpha - \psi(\phi_1) - \beta)$$

$$p_0 \cdot \cos(\phi_2 + \alpha) + p_1 \cdot \cos(\psi(\phi_2) + \beta) + p_2 = \cos(\phi_2 + \alpha - \psi(\phi_2) - \beta)$$

$$p_0 \cdot \cos(\phi_3 + \alpha) + p_1 \cdot \cos(\psi(\phi_3) + \beta) + p_2 = \cos(\phi_3 + \alpha - \psi(\phi_3) - \beta) \quad (2)$$

$\psi(\phi)$ - исходная функция закона движения шарнирного четырехзвенника, вычисляется по формуле:

$$\psi(\phi) = k_1 \cdot \sin(\phi) + k_2$$

Углы ϕ_1, ϕ_2, ϕ_3 вычисляются по формулам:

$$\phi_1 = 0.25 \cdot \phi_m$$

$$\phi_2 = 0.75 \cdot \phi_m$$

$$\phi_3 = \phi_m$$

$$\phi_m = 2 \cdot \pi$$

Исходные данные

Исходными данными для решения задачи являются:

- углы α и β , определяющие взаимное расположение звеньев АВ и CD относительно стойки, заданы в таблице 2.2;
- k_1, k_2 – параметры закона движения механизма;
- пределы изменения угла ϕ ($0 \leq \phi \leq 2\pi$).

Таблица 2.2

Номер варианта	α , рад	β	k_1, k_2
1	3	140°	0.313 0.13
2	3,1	100°	0.3 0.15
3	3,02	115°	0.317 0.14
4	2,87	125°	0.31 0.15
5	2,9	135°	0.295 0.135

Алгоритм решения задачи

Шаг 1.

Присвоить имена исходным данным и поместить в них указанные числовые значения (перевести градусы в радианы).

Шаг 2.

Вычислить значения $f_{im}, f_{i1}, f_{i2}, f_{i3}$.

Шаг 3.

Организовать функцию с именем f_{un} , вычисляющую значение угла ψ по формуле $\psi(\phi) = k_1 \cdot \sin(\phi) + k_2$.

Шаг 4.

Рассчитать с использованием функции f_{un} значения углов $\psi_{i1}, \psi_{i2}, \psi_{i3}$, которые затем будут участвовать в решении системы уравнений.

Шаг 5.

Для того, чтобы найти длины звеньев механизма, нужно вычислить значения величин p_0 , p_1 , p_2 . Их можно узнать, решив систему линейных уравнений матричным методом. Для этого заполним матрицу A коэффициентов перед неизвестными и вектор свободных членов B .

Шаг 6.

Получаем вектор неизвестных P , используя стандартную функцию `inv`. Для удобства дальнейших вычислений будем считать, что p_0 это $P(1)$, p_1 это $P(2)$, p_2 это $P(3)$.

Шаг 7.

Вычисляем по заданным формулам и выводим значения длин звеньев.

Программа

```
clear
alf=3
bet=125*%pi/180
fim=2*%pi
fi1=0.25*fim
fi2=0.75*fim
fi3=fim
function y=fun(fi)
    k1=0.3
    k2=0.6
    y=k1*sin(fi)+k2
endfunction
psi1=fun(fi1)
psi2=fun(fi2)
psi3=fun(fi3)
A=[cos(fi1+alf) cos(psi1+bet) 1;cos(fi2+alf) cos(psi2+bet) 1;...
cos(fi3+alf) cos(psi3+bet) 1]
B=[cos(fi1+alf-psi1-bet); cos(fi2+alf-psi2-bet); cos(fi3+alf-psi3-bet)]
P=inv(A)*B
a=(1/P(2))
disp(a)
c=(-1/P(1))
disp(c)
b=sqrt(a^2+c^2+1-(2*c*a*P(3)))
disp(b)
```

Результаты

0.32 - значение длины звена a

1.21 - значение длины звена c

0.42 - значение длины звена b

Задание для самостоятельного выполнения

Самостоятельно проведите расчет по программе для любого варианта исходных данных из табл.2.2.

2.4. Задача о движении точки по траектории

Задание

1. Рассчитать значения проекций скоростей материальной точки на оси X и Y при заданном изменении x.

Постановка задачи

Точка движется по плоской кривой $Y=Y(X)$ с постоянной скоростью v (рис.2.6).

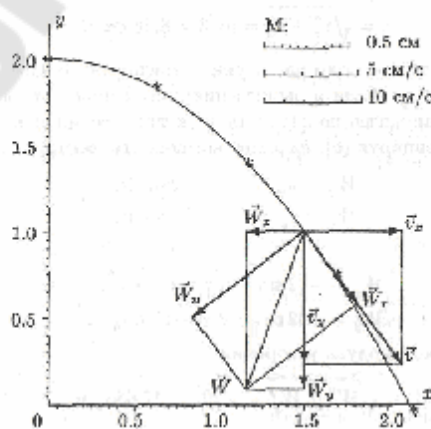


Рис. 2.6. График движения материальной точки

Для нахождения проекций скорости движения точки на оси X и Y (v_x, v_y) нужно при заданном значении x решить систему уравнений вида:

$$v_y = y_1(x) \cdot v_x$$

$$v_x^2 + v_y^2 = v^2, \text{ где}$$

$$y_1(x) = \frac{d}{dx} y(x)$$

- первая производная по координате x , $y(x)$ – заданный закон движения точки.

Исходные данные

- v – постоянная скорость движения точки;
- вид функций закона движения точки;
- x – значение координаты x , при котором нужно выполнить расчет (выбрать любое значение из заданного диапазона в табл.2.3).

Таблица 2.3

№	V (м/с)	Диапазон значений X	Функция закона движения
1	4	0.1 – 3.1	$Y(x)=2x-4x^2/9$
2	5.1	0.4 – 2.5	$Y(x)=2x^3-3x^2$
3	5	0.1 – 1.7	$Y(x)=3x^2-2\sin x$
4	4.8	0.3 – 3.9	$Y(x)=0.5x^2+2x\sin x$
5	5.5	0.2 – 1.4	$Y(x)=2x^3/4+3\cos^2 x$
6	4.9	0.2 – 2.6	$Y(x)=2x^2+5\cos x$

Алгоритм решения задачи

Шаг 1.

Присвоим числовые значения исходным данным. Например, $v=5$ для скорости, $x=0.2$ для координаты x .

Шаг 2.

Зададим закон движения точки в виде функции
`function y=fun(x),y=2*x^2+5*cos(x), endfunction`

Шаг 3.

Найдем значение функции в точке x . Найдем производную функции в точке x с помощью стандартной функции `numdiff(fun,x)`.

Шаг 4.

Опишем функцию с именем sur, содержащую вид системы нелинейных уравнений.

Шаг 5.

Решим систему нелинейных уравнений с помощью функции fsolve. Получим вектор из двух элементов, первый элемент – значение vx, второй – значение vy. Выведем эти результаты в командную область.

Программа

```
clear
v=5
x=0.2
function y=fun(x),y=2*x^2+5*cos(x), endfunction
y=fun(x)
y1=numdiff(fun,x)
function [V]=sur(vxy)
    V(1)=y1*vxy(1)-vxy(2)
    V(2)=vxy(1)^2+vxy(2)^2-v^2
endfunction
vxy=fsolve([0.5,0.5],sur)
disp(vxy(1))
disp(vxy(2))
```

Результаты

4.9090839 - значение vx

- 0.9491549 – значение vy

Задание для самостоятельного выполнения

Самостоятельно проведите расчет по программе для любого варианта исходных данных из табл.2.3.

3. ВЫЧИСЛЕНИЕ ПАРАМЕТРОВ ДИНАМИЧЕСКИХ ОБЪЕКТОВ, ОПИСАННЫХ ДИФФЕРЕНЦИАЛЬНЫМИ УРАВНЕНИЯМИ И СИСТЕМАМИ

3.1. Краткие теоретические сведения

Стандартные функции решения ОДУ в Scilab содержат ряд параметров, общий вид вызова функции следующий:

ode (y0,t0,t,f)

Здесь:

y0 - начальные условия,

t0 – начальное значение диапазона, на котором ищется решение ОДУ,

t – вектор точек диапазона, в которых ищется решение,

f - имя функции, описывающей одно дифференциальное уравнение или система дифференциальных уравнений.

Пример 1.

Решить ОДУ вида:

$$y' = -y + 2x \quad y(0) = 1$$

```
clear
function ur=vid(x, y);
ur = -y(1)+2*x;
endfunction
y0=1
x=0:0.01:1
// Решение дифференциального уравнения
Y=ode(y0,0,x,vid);
plot(x,Y)
```

На рис. 3.1 приведен графический вид функции решения ОДУ первого порядка.

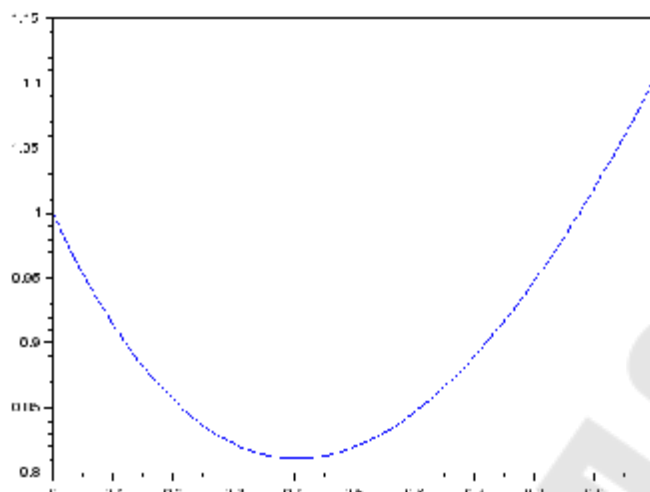


Рис. 3.1. Графический вид функции решения ОДУ первого порядка

Пример 2.

Решить систему дифференциальных уравнений вида:

$$\begin{cases} \frac{dx}{dt} = y - x^2 - x & x(0) = 0 \\ \frac{dy}{dt} = 3x - x^2 - y & y(0) = 1 \end{cases}$$

Выполнена замена переменных:

$$v(1) - x, v(2) - y.$$

Для решения системы ОДУ создается функция, описывающая вид системы, которая затем используется в стандартной функции решения ОДУ – ode.

```
clear
function ur2=vid2(t, v);
ur2=zeros(2,1);
ur2(1)=v(2)-v(1)^2-v(1);
ur2(2)=3*v(1)-v(1)^2-v(2);
endfunction
y0=[0;1];
t=0:0.01:1
// Решение дифференциального уравнения
V=ode(y0,0,t,vid2);
plot(t,V)
xgrid
```

На рис. 3.2. приведен графический вид двух функций, полученных в результате решения ОДУ.

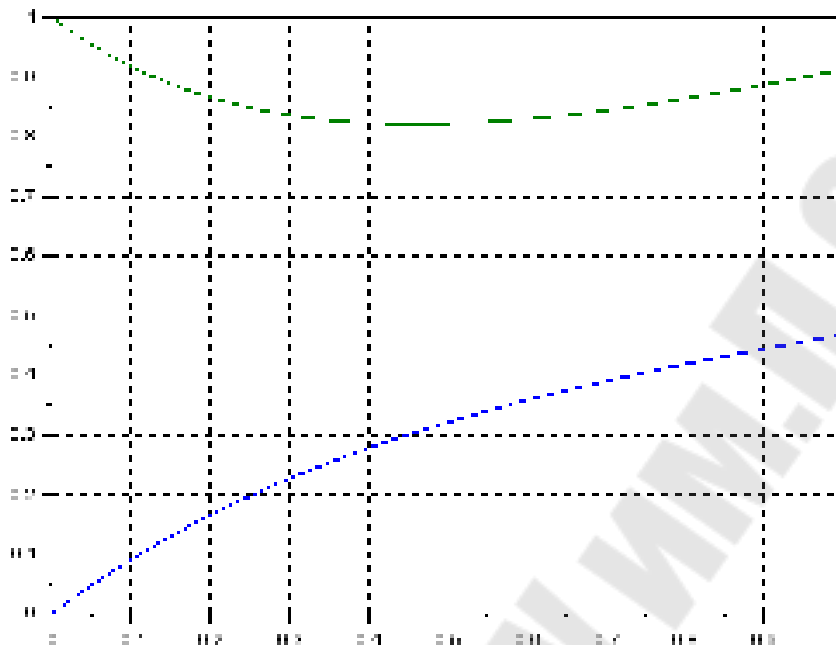


Рис. 3.2. Графики двух функций, полученных в результате решения ОДУ

3.2. Задача об электрической схеме

Задание

1. Рассчитать значения функций тока и напряжения в схеме с туннельным диодом.
2. Построить графики полученных функций.

Постановка задачи

Электрическая цепь, приведенная на рис. 3.3, описывается системой дифференциальных уравнений вида:

$$\frac{di}{dt} = \frac{E - i \cdot R - u}{L}$$

$$\frac{du}{dt} = \frac{i - I(u)}{C}$$

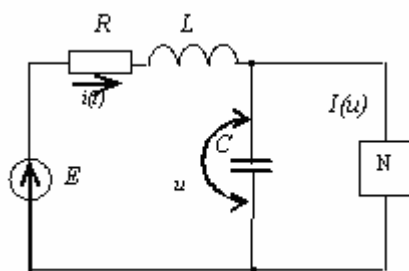


Рис. 3.3 Вид электрической цепи с туннельным диодом

Вольт-амперная характеристика туннельного диода имеет вид:

$$I(u) = A \cdot u \cdot e^{-\alpha \cdot u} + D \cdot (e^{\beta \cdot u} - 1)$$

Параметры характеристики вычисляются по формулам:

$$A = e \cdot \frac{I_p}{U_1} \quad \alpha = \frac{1}{U_1}$$

Исходные данные

Исходными данными для решения задачи являются следующие:

I_p, U_1, D, β – параметры вольт-амперной характеристики диода;

$E = 0.3$ – ЕДС;

R – исходное сопротивление;

C – исходная емкость;

L – исходная индуктивность;

u_0 – начальное значение напряжения;

i_0 – начальное значение тока;

T – время исследования.

При решении системы дифференциальных уравнений принять i_0 и u_0 (начальные значения тока и напряжения) равными 0.

Таблица 3.1- Таблица исходных данных

N варианта	C Ф	R Ом	L Гн	I_p (А)	D (А)	β (1/В)	U1 (В)	T с
1	10^{-11}	10	10^{-7}	$1,05 \cdot 10^{-2}$	$1,15 \cdot 10^{-8}$	20	0.1	$1.4 \cdot 10^{-8}$
2	$0.8 \cdot 10^{-11}$	8	$90 \cdot 10^{-9}$	$1,15 \cdot 10^{-2}$	$1,25 \cdot 10^{-8}$	18,5	0,12	$1.6 \cdot 10^{-8}$
3	$0.88 \cdot 10^{-11}$	7	$95 \cdot 10^{-9}$	$1,07 \cdot 10^{-2}$	$1,55 \cdot 10^{-8}$	19,2	0,15	$1.5 \cdot 10^{-8}$

4	$0.82 \cdot 10^{-11}$	6	$85 \cdot 10^{-9}$	$1,1 \cdot 10^{-2}$	$1,75 \cdot 10^{-8}$	18,7	0,18	$1.5 \cdot 10^{-8}$
5	$0.81 \cdot 10^{-11}$	9	$0.95 \cdot 10^{-7}$	$1,12 \cdot 10^{-2}$	$1,35 \cdot 10^{-8}$	18,2	0,17	$1.55 \cdot 10^{-8}$

Алгоритм решения задачи

Шаг 1.

Описать функцию, в которой запрограммировать выполнение следующих действий: ввод исходные данные для решения задачи; вычисление значений A и α ; описание правых частей системы ОДУ.

Шаг 2.

Задать вектор начальных условий, задать вектор времени (независимой переменной для решения ОДУ)

Шаг 3.

Задать оператор, вызывающий стандартную функцию ode с необходимыми параметрами и получить матрицу значений результирующей функции.

Шаг 4.

Построить графики полученных функций тока и напряжения с использованием оператора plot.

Программа

```

clc
clear
function ur2=u2(t, y);
ur2=zeros(2,1);
// Исходные данные
L=1e-7; R=10; C=1e-11; D=1.15e-8; Ip=1.05e-2; bet=20; U1=0.1; E=0.3
// Расчет параметров вольт-амперной характеристики
A=exp(1)*Ip/U1; alfa=1/U1;
I=A*y(2) *exp(-alfa*y(2))+D*(exp(bet*y(1))-1)
// Описание системы дифференциальных уравнений
ur2(1)=(E-y(1)*R-y(2))/L;
ur2(2)=(y(1)-I)/C;
endfunction

```

```

//Задание начальных условий
y0=[0 ;0];
t=0:1e-10:1.5e-8
// Решение дифференциального уравнения
Y=ode(y0,0,t,u2);
//Графики функций тока и напряжения
figure(1)
plot(t,Y(1,:))
xgrid
figure (2)
plot(t,Y(2,:))
xgrid

```

Результаты

На рис. 3.4 приведен результат решения ОДУ – график функции тока, а на рис. 3.5 приведен график функции напряжения в электрической схеме.

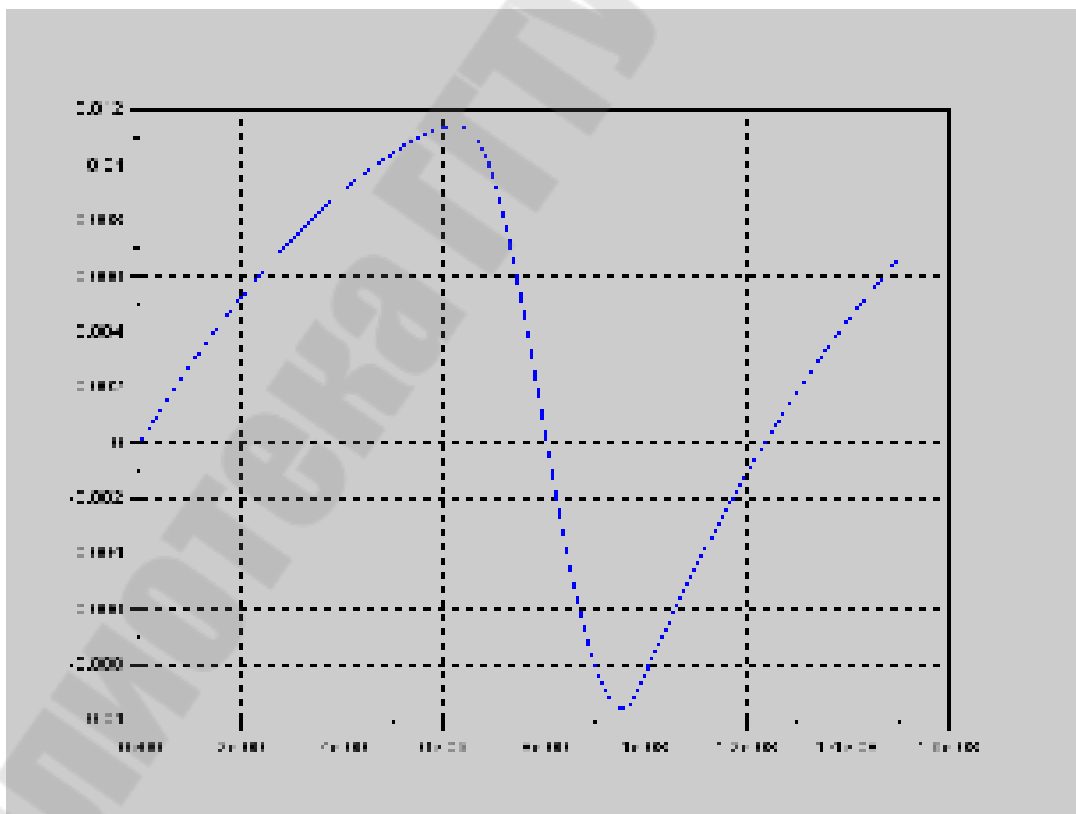


Рис. 3.4. Результат решения ОДУ – график функции тока

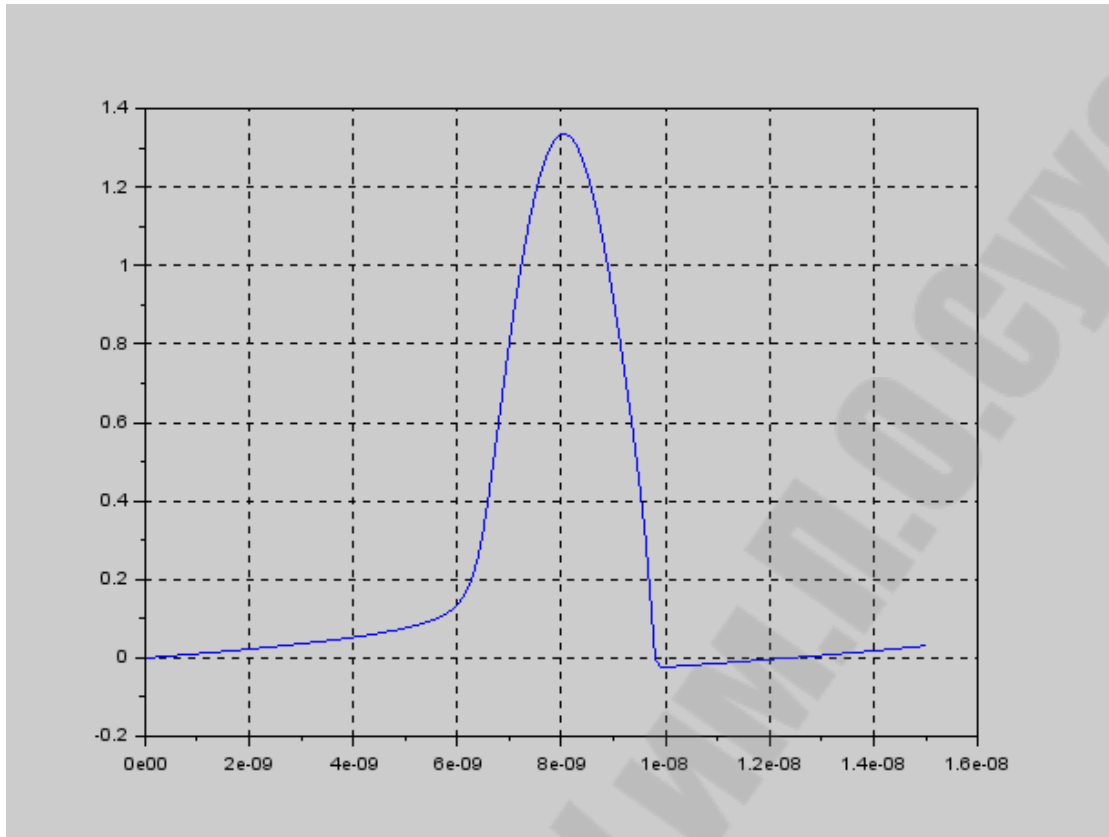


Рис. 3.5. Результат решения ОДУ – график функции напряжения

Задание для самостоятельного выполнения

Самостоятельно проведите расчет по программе для любого варианта исходных данных из табл.3.1.

3.3. Задача о механической динамической системе

Задание

1. Рассчитать значения функций перемещения и скорости движения поршня гидравлического демпфера под действием начального значения перемещения.
2. Построить графики полученных функций.

Постановка задачи

Дан гидравлический демпфер, представляющий собой поршень массой m , движущийся в жидкости (рис. 3.6).

Дифференциальное уравнение движения поршня имеет вид:

$$\ddot{y} + 2n\dot{y} + py = 0,$$

где n - приведенный коэффициент вязкого сопротивления вычисляется по формуле:

$$n = [4\pi\mu H / (mZ)] \cdot (D/d)^4,$$

p - частота собственных колебаний системы вычисляется по формуле:

$$p = c / m.$$

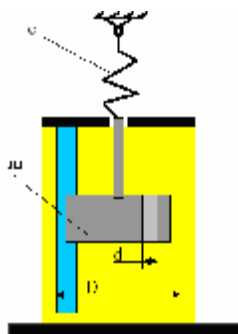


Рис. 3.6. Гидравлический демпфер

Необходимо найти функции перемещения и скорости демпфера в зависимости от времени.

Для решения дифференциального уравнения в Scilab с помощью функции `ode`, его нужно привести к системе из двух дифференциальных уравнений вида:

$$\begin{aligned} \dot{y}_1 &= y_2 \\ \dot{y}_2 &= -2n y_2 - p y_1 \end{aligned}$$

где y_1 - перемещение демпфера; y_2 - скорость демпфера.

Исходные данные

Исходными данными для решения задачи являются следующие:
 y_0 - отклонение поршня демпфера от положения равновесия;
 m - масса поршня;
 c - жесткость пружин;
 μ - динамический коэффициент вязкости жидкости;
 D - диаметр цилиндра;
 d - диаметр отверстия;
 H - высота поршня;
 Z - число отверстий.

Таблица 3.2 – Исходные данные для задачи 3.3

N варианта	H (мм)	c (кН/м)	D (м)	d (мм)	z	m (кг)	μ (Па*с)	y ₀ (мм)	t _к (с)
1	50	3	0,1	10	25	2,73	0,06	5	0,9
2	45	4,5	0,11	11	27	5,23	0,065	5	1
3	40	3	0,08	9	30	3,5	0,02	5	1,5
4	50	1,5	0,1	8	26	2,5	0,058	5,5	0,9
5	45	1,6	0,12	10	10	2,7	0,059	4,8	1

Алгоритм решения задачи

Ниже приведен алгоритм выполнения расчетов в системе Scilab .

Шаг 1.

Описать функцию, в которой запрограммировать выполнение следующих действий: ввод исходные данные для решения задачи; вычисление значений n и p ; описание правых частей системы ОДУ.

Шаг 2.

Задать вектор начальных условий, задать вектор времени (независимой переменной для решения ОДУ)

Шаг 3.

Задать оператор, вызывающий стандартную функцию `ode` с необходимыми параметрами и получить матрицу значений результирующей функции.

Шаг 4.

Построить графики полученных функций перемещения и скорости от времени с использованием оператора `plot`.

Программа

```
clc
clear
function ur2=vid2(t, y);
ur2=zeros(2,1);
// Исходные данные
```

```

m=2.73; H=0.05; c=3e3; D1=0.1; d=0.01; z=25; mu=6e-2;
// Расчет приведенного коэффициента вязкого сопротивления
// и частоты собственных колебаний демпфера
p=sqrt(c/m); n=4*%pi*mu*H/(m*z)*(D1/d)^4;
// Описание системы дифференциальных уравнений
ur2(1)=y(2);
ur2(2)=-2*n*y(2)-p^2*y(1);
endfunction
//Задание начальных условий
y0=[0.05 ;0];
t=0:0.01:1
// Решение дифференциального уравнения
Y=ode(y0,0,t,vid2);
//Графики функций перемещения и скорости демпфера
figure(1)
plot(t,Y(1,:))
figure (2)
plot(t,Y(2,:))

```

Результаты

На рис. 3.7., 3.8 приведен результат решения ОДУ – график функции перемещения поршня демпфера.

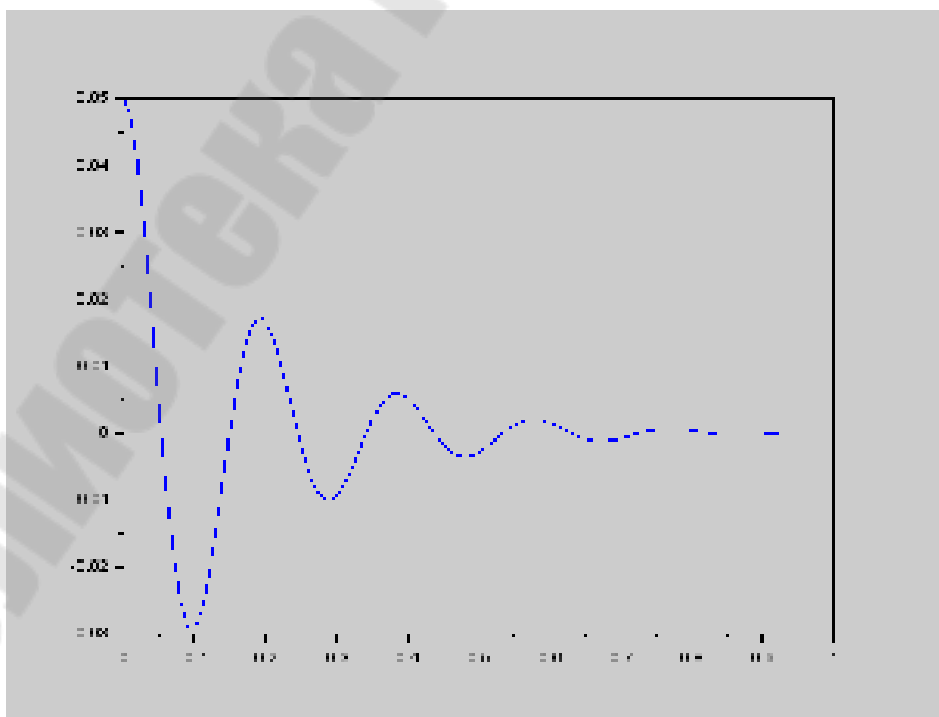


Рис. 3.7. График функции перемещения поршня демпфера.

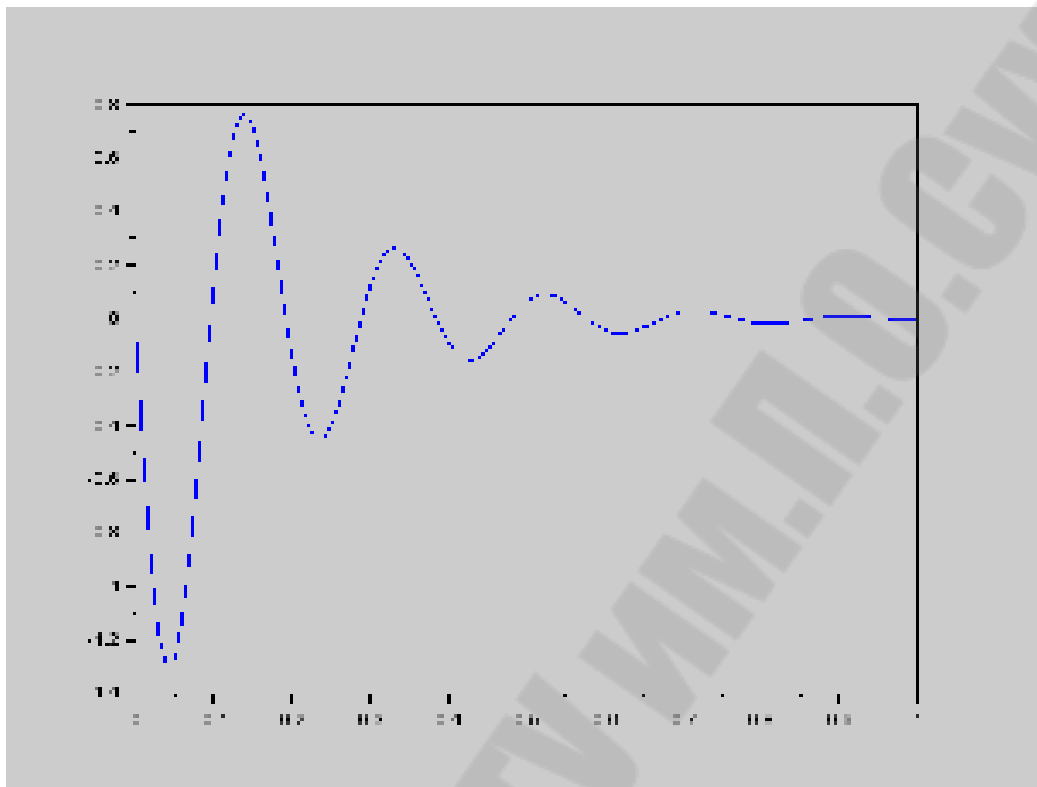


Рис. 3.8. График функции перемещения поршня демпфера.

Задание для самостоятельного выполнения

Самостоятельно проведите расчет по программе для любого варианта исходных данных из табл.3.2.

Трохова Татьяна Анатольевна

**РЕШЕНИЕ ИНЖЕНЕРНЫХ ЗАДАЧ
В СИСТЕМЕ КОМПЬЮТЕРНОЙ
МАТЕМАТИКИ SCILAB**

**Практикум
по курсу «Информатика»
для студентов технических специальностей
дневной и заочной форм обучения**

Подписано к размещению в электронную библиотеку
ГГТУ им. П. О. Сухого в качестве электронного
учебно-методического документа 06.03.18.

Рег. № 39Е.
<http://www.gstu.by>