Министерство образования Российской Федерации

МОСКОВСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ им. Н.Э. БАУМАНА

Факультет: Информатика и системы управления
Кафедра: Информационная безопасность (ИУ8)

ОСНОВЫ ТЕОРИИ УПРАВЛЕНИЯ

Лабораторная работа №1 на тему: «Исследование динамики линейных систем, описываемых обыкновенными дифференциальными уравнениями»

Вариант 4

Преподаватель:
Чернега Е.В.

Студент:
Девяткин Е.Д.

Группа:
ИУ8-44

Репозиторий работы: https://github.com/ledibonibell/Module04-BMT

Москва 2024

Цель работы

Ознакомиться с пакетом моделирования MatLab. Освоить основные приемы моделирования САУ, описываемых при помощи обыкновенных дифференциальных уравнений.

Порядок выполнения работы

1. Ознакомиться с пакетом прикладных программ MatLab.

2. Записать дифференциальное уравнение по исходным данным.

Порядок модели n	a_0	a_1	a_2	a_3	b_0
3	5	4	2	1	5

Порядок модели n	$x(0)$	$\dot{x}(0)$	$\ddot{x}(0)$
3	1	0.5	-1

3. Привести исходную систему к нормальной форме Коши.

4. Написать скрипт и осуществить моделирование системы дифференциальных уравнений в математическом пакете MatLab , используя численный метод интегрирования Рунге-Кутта и задавая н.у. векторами.

5. Проанализировать системы при двух видах входных воздействий:

$$y_1 = 1(t); y_2 = 2 \cdot \sin(t)$$

6. На каждое из воздействий найти решение для двух различных начальных условий:

   6.1. нулевые н.у.

   6.2. $\ddot{x}(0) =- 1, \dot{x}(0) = 0.5, x(0) = 1$

На дисплей выводить графики сигналов $x_1(t)$ (синий цвет, сплошная линия) и $x_2(t)$ (зеленый цвет, пунктирная линия). Продолжительность интервалов наблюдения выбрать равной $T = 25$ с.

Ход работы

Исходная система:

$$x^{(3)} + 2x^{(2)} + 4x^{(1)} + 5x = 5y$$

А также приведем ее к нормальной форме Коши:

$$\dot{x}_3 = \ 5y - 2x_3 - 4x_2 - 5x_1$$

$$\begin{cases} x = x_1 \\ \dot{x}_1 = x_2 \\ \dot{x}_2 = x_3 \end{cases}$$

Теперь рассмотрим различные входные воздействия на систему (листинги кодов в конце отчета):

1. Входное воздействие - $y_1 = 1(t)$

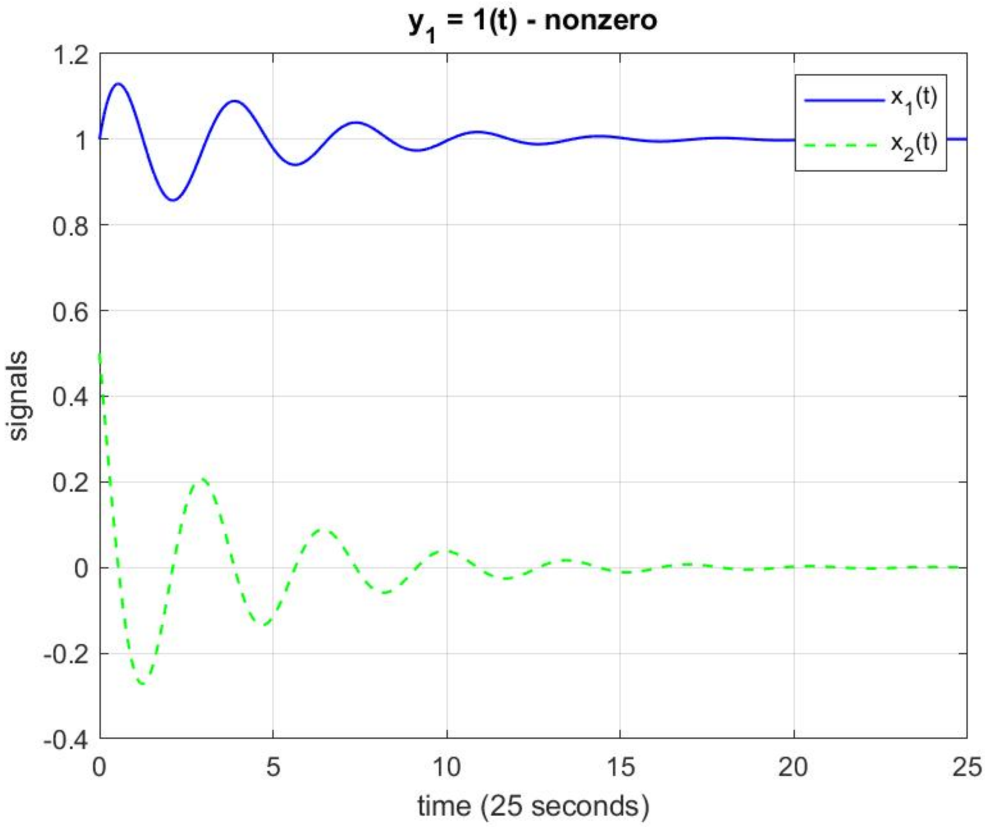

Рис. 1 – Реакция системы на единичное воздействие при ненулевых начальных условиях

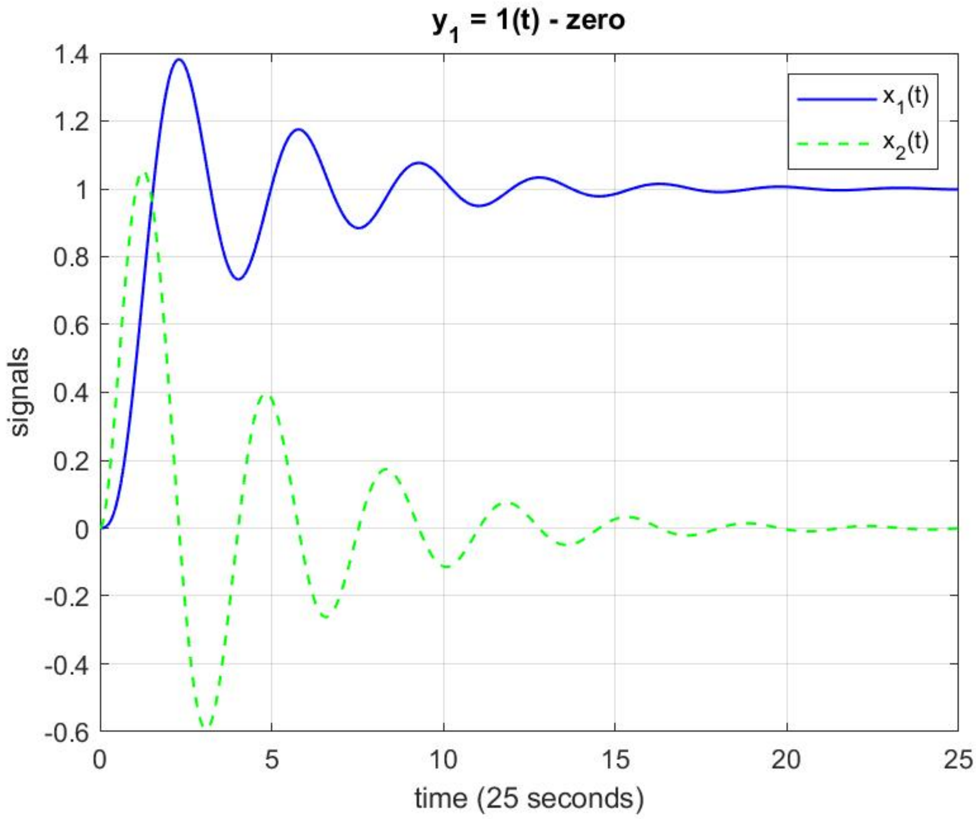

Рис. 2 – Реакция системы на единичное воздействие при нулевых начальных условиях

2. Входное воздействие - $y_2 = 2 \cdot \sin(t)$

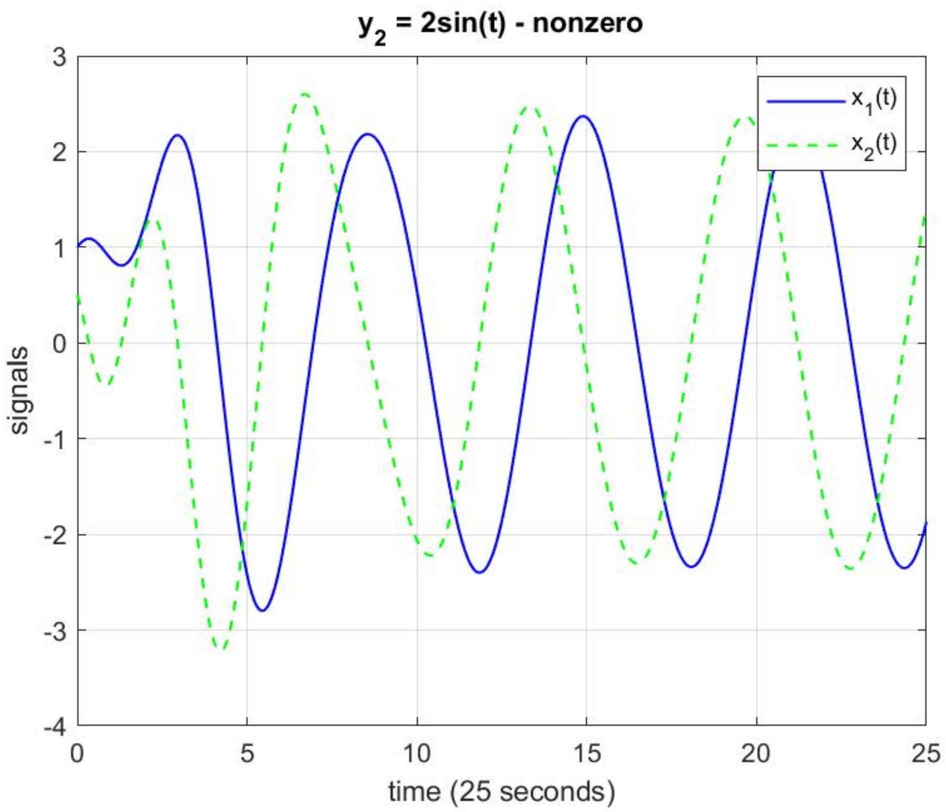

Рис. 3 – Реакция системы на синусоидальное воздействие при ненулевых начальных условиях

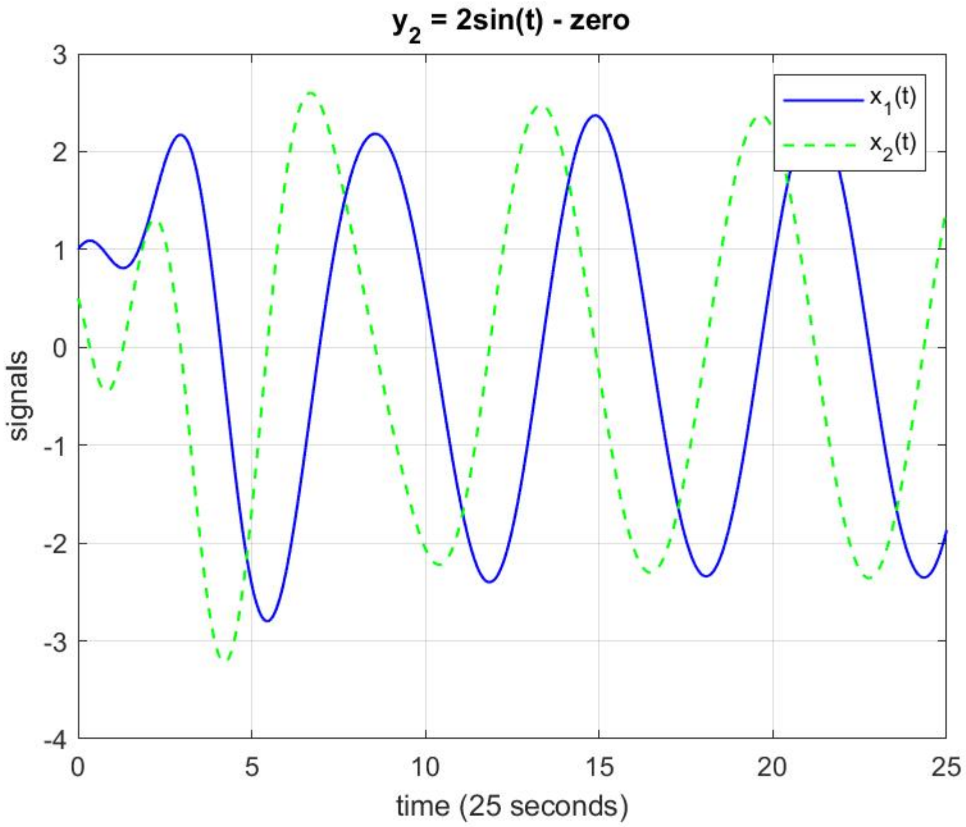

Рис. 4 – Реакция системы на синусоидальное воздействие при нулевых начальных условиях

Вывод

В ходе лабораторной работы были получены начальные (базовые) умения работы со средой программирования MatLab и моделирования САУ.

Были решены данные в условии уравнения с помощью метода понижения порядкам дифференциального уравнения, а также смоделирована система д.у. с применением численного метода Рунге-Кутта.

Также был произведен анализ входных воздействий на различные уравнения, с различными начальными условиями, в ходе которого был сделан вывод, о его компенсации функцией и ее последующем возвращении к исходному состоянию.

Листинг 1

Приведение дифференциального уравнения к нормальной форме Коши при единичном воздействии:

```
options:
n = 3;
a0 = 5;
a1 = 4;
a2 = 2;
a3 = 1;
b0 = 5;

model:
function xp = model(t, x, y)

% x - вектор длины 3
% y - некоторая функция
% t - время

xp = zeros(3, 1);

in = y(t);
xp(1) = x(2);
xp(2) = x(3);
xp(3) = b0 * in - a2 * x(3) - a1 * x(2) - a0 * x(1);
```

Листинг 2.1

Решение дифференциального уравнения методом Рунге-Кутта при ненулевых начальных условиях и единичном воздействии:

```
main:
ode_opts = odeset('AbsTol',[1e-5,1e-5,1e-5],'RelTol',1e-5);

y = @(t) 1;
model_1 = @(t, x) model(t, x, y);

x_nonzero = [ 1, 0.5, -1];
T = 25;

[time, x] = ode45(model_1, [0, T], x_nonzero, ode_opts);
```

Листинг 2.2

Решение дифференциального уравнения методом Рунге-Кутта при нулевых начальных условиях и единичном воздействии:

main:

```
ode_opts = odeset('AbsTol',[1e-5,1e-5,1e-5],'RelTol',1e-5);

y = @(t) 1;
model_1 = @(t, x) model(t, x, y);

x_zero = [0, 0, 0];
T = 25;

[time, x] = ode45(model_1, [0, T], x_zero, ode_opts);
```

Листинг 3

Приведение дифференциального уравнения к нормальной форме Коши при синусоидальном воздействии:

options:

```
n = 3;
a0 = 5;
a1 = 4;
a2 = 2;
a3 = 1;
b0 = 5;
```

model:

```
function xp = model(t, x, y)

% x - вектор длины 3
% y - некоторая функция
% t - время

xp = zeros(3, 1);

in = y(t);
xp(1) = x(2);
xp(2) = x(3);
xp(3) = b0 * in - a2 * x(3) - a1 * x(2) - a0 * x(1);
```

Листинг 4.1

Решение дифференциального уравнения методом Рунге-Кутта при ненулевых начальных условиях и синусоидальном воздействии:

main:
```
ode_opts = odeset('AbsTol',[1e-5,1e-5,1e-5],'RelTol',1e-5);

y_sin = @(t) 2 * sin(t);
model_sin = @(t, x) model(t, x, y_sin);

x_nonzero = [ 1, 0.5, -1];
T = 25;

[time, x] = ode45(model_sin, [0, T], x_nonzero, ode_opts);
```

Листинг 4.2

Решение дифференциального уравнения методом Рунге-Кутта при нулевых начальных условиях и синусоидальном воздействии:

main:
```
ode_opts = odeset('AbsTol',[1e-5,1e-5,1e-5],'RelTol',1e-5);

y_sin = @(t) 2 * sin(t);
model_sin = @(t, x) model(t, x, y_sin);

x_zero = [0, 0, 0];
T = 25;

[time, x] = ode45(model_sin, [0, T], x_zero, ode_opts);
```

Листинг программы

main.m:

```
options;
ode_opts = odeset('AbsTol',[1e-5,1e-5,1e-5],'RelTol',1e-5);

y = @(t) 1;
y_sin = @(t) 2 * sin(t);

model_1 = @(t, x) model(t, x, y);
model_sin = @(t, x) model(t, x, y_sin);

graphic(model_1, x_zero, 'y_1 = 1(t) - zero', 'graphics/signal - 1 - zero.png');
graphic(model_1, x_nonzero, 'y_1 = 1(t) - nonzero', 'graphics/signal - 1 -
nonzero.png');

graphic(model_sin, x_nonzero, 'y_2 = 2sin(t) - zero', 'graphics/signal - sin - zero.png')
graphic(model_sin, x_nonzero, 'y_2 = 2sin(t) - nonzero', 'graphics/signal - sin -
nonzero.png')

pause
```

model.m:

```
function xp = model(t, x, y)

% x - vector of length 3
% y - some function
% t - time

options;
disp(t);
xp = zeros(3, 1);

in = y(t);
xp(1) = x(2);
xp(2) = x(3);
xp(3) = b0 * in - a2 * x(3) - a1 * x(2) - a0 * x(1);
```

options.m:

```
n = 3;
a0 = 5;
a1 = 4;
a2 = 2;
a3 = 1;
b0 = 5;

x_zero = [0, 0, 0];
x_nonzero = [ 1, 0.5, -1];

T = 25; %time of plot's interval
```

graphic.m:

```
function graphic(model, zero, uptitle, name)

% model - equation with 1(t) or 2 * sin(t)
% zero - initional conditions
% uptitle - titles
% name - name of plot in directory

options;
ode_opts = odeset('AbsTol',[1e-5,1e-5,1e-5],'RelTol',1e-5);


[time, x] = ode45(model, [0, T], zero, ode_opts);
figure;
plot(time, x(:,1), 'b-', time, x(:,2), 'g--', 'LineWidth', 1);

legend('x_1(t)', 'x_2(t)');

title(uptitle);
xlabel('time (25 seconds)');
ylabel('signals');

grid on;
saveas(gcf, name);
```