
ЛР №2 «Переходные процессы, свободное движение,
устойчивость»

Отчет

Студент
Кирилл Лалаянц
R33352
336700
Вариант - 6

Преподаватель
Пашенко А.В.

Факультет Систем Управления и Робототехники

ИТМО

13.09.2023

Содержание

1	Вводные данные	1
1.1	Цель работы	1
1.2	Инициализация необходимых переменных в Python	1
2	Выполнение работы	2
2.1	Задание 1. Свободное движение.	2
2.1.1	Теория	2
2.1.2	Программная реализация	2
2.1.3	Результаты	3
2.2	Задание 2. Область устойчивости.	6
2.2.1	Теория	6
2.2.2	Результаты	7
2.3	Задание 3. Автономный генератор.	10
2.3.1	Теория	10
2.3.2	Программная реализация	10
2.3.3	Результаты	10
2.4	Задание 4. Одноканальная система в форме вход-состояние-выход.	12
2.4.1	Теория	12
2.4.2	Программная реализация	12
2.4.3	Результаты	12
2.5	Задание 5. Многоканальная система в форме вход-состояние-выход.	13
2.5.1	Теория	13
2.5.2	Программная реализация	13
2.5.3	Результаты	13
3	Заключение	14
3.1	Выводы	14

1 Вводные данные

1.1 Цель работы

В этой работе будет проведено исследование следующих вопросов:

- Характер свободного поведения системы в зависимости от корней ее характеристического уравнения.
- Исследования границ зоны устойчивости конкретной системы.
- Создание автономного генератора для воссоздания поведения системы.
- Изучение фазовых портретов канонической управляемой формы.

1.2 Инициализация необходимых переменных в Python

Импорт необходимых библиотек (для реализации здесь и далее используется [Python Control Systems Library](#)); инициализация массива временных отметок:

```
1 import matplotlib.pyplot as plt
2 import numpy as np
3 import control
4 import sympy
5 import os
6 import math
```

2 Выполнение работы

2.1 Задание 1. Свободное движение.

2.1.1 Теория

Хотим получить ДУ системы вида:

$$\ddot{y} + a_1\dot{y} + a_0y = u$$

Сначала получим характеристическое уравнение для частного решения вида:

$$\ddot{y} + a_1\dot{y} + a_0y = 0$$

Имеем два корня $-\lambda_1$ и λ_2 . Тогда:

$$(p - \lambda_1)(p - \lambda_2) = p^2 - (\lambda_1 + \lambda_2)p + \lambda_1\lambda_2 = 0,$$

умножив которое на y получаем:

$$\ddot{y} + (\lambda_1 + \lambda_2)\dot{y} + \lambda_1\lambda_2y = 0.$$

Для данного ДУ известно частичное решение:

$$y_0(t) = c_1e^{\lambda_1 t} + c_2e^{\lambda_2 t},$$

параметры c_1 и c_2 которого вычисляются из начальных условий.

Также подставим характеристическое уравнение в знаменатель ТФ $W(p)$:

$$W(p) = \frac{1}{p^2 - (\lambda_1 + \lambda_2)p + \lambda_1\lambda_2}$$

Полученная ТФ соответствует системе с желаемыми корнями характеристического уравнения.

2.1.2 Программная реализация

```
1 def task1_output(m1, m2, initial_value, ts, plot_name, save_name):
2     poly = sympy.simplify((p - m1) * (p - m2))
3     coeffs = sympy.Poly(poly, p).all_coeffs()
4
5     ss = control.tf2ss(control.tf(1, np.array(coeffs, dtype=np.float64)))
6     ss_reachable = control.canonical_form(ss, form="reachable")[0]
7
8     tf2_y_0_0 = control.forced_response(ss_reachable, U=0, X0=[0, 0], T=ts)
9     tf2_y_1_1 = control.forced_response(ss_reachable, U=0, X0=initial_value, T=ts)
10    plot_task1(tf2_y_0_0, tf2_y_1_1, initial_value, ts, plot_name, save_name)
```

2.1.3 Результаты

- Устойчивая и неустойчивая апериодические моды

Выбраны $\lambda_1 = -1$ и $\lambda_2 = 1$. Получено уравнение свободного движения вида:

$$y_0(t) = c_1 e^{-t} + c_2 e^t$$

Из аналитического решения видно, что система, при отличных от 0 начальных условиях, со временем стремится к бесконечности, следовательно – неустойчива.

Проверим моделированием:

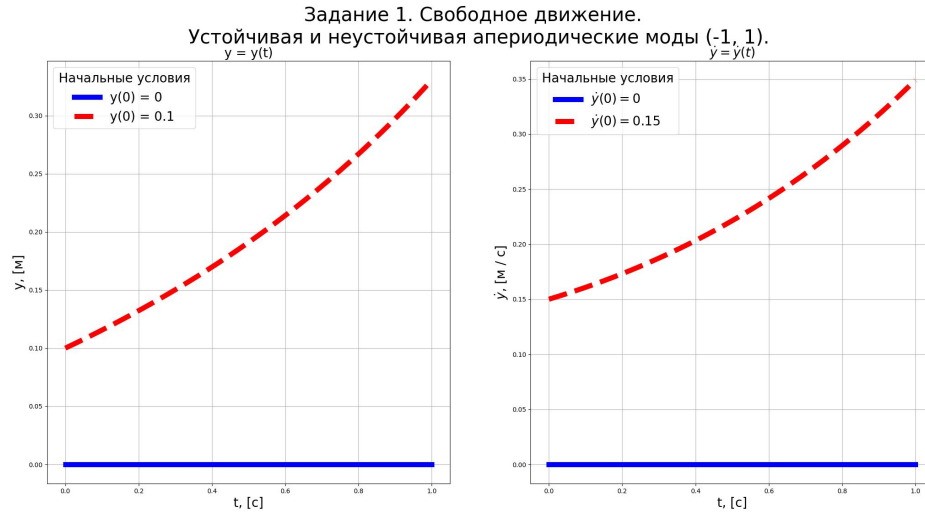


Рис. 1: Результаты устойчивой и неустойчивой апериодических мод.

Действительно, все так.

- Нейтральная и неустойчивая апериодические моды

Выбраны $\lambda_1 = 0$ и $\lambda_2 = 1$. Получено уравнение свободного движения вида:

$$y_0(t) = c_1 e^{-t} + c_2$$

Из аналитического решения видно, что система, при отличных от 0 начальных условиях, со временем стремится к бесконечности, следовательно – неустойчива.

Проверим моделированием:

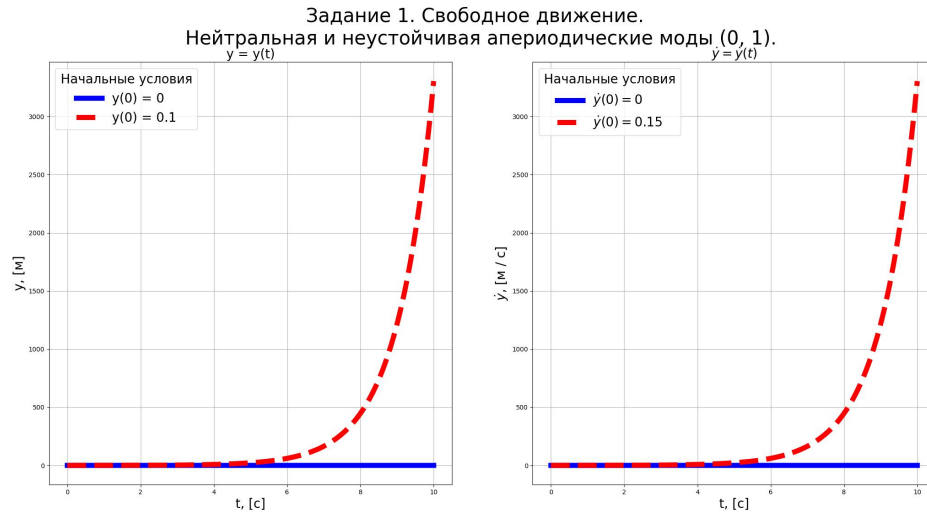


Рис. 2: Результаты нейтральной и неустойчивой апериодических мод.

Действительно, все так.

- Пара консервативных мод

Выбраны $\lambda_1 = i$ и $\lambda_2 = -i$. Получено уравнение свободного движения вида:

$$y_0(t) = c_1 \cos(t) + c_2 \sin(t)$$

Из аналитического решения видно, что система, при отличных от 0 начальных условиях, устойчива по Ляпунову.

Проверим моделированием:

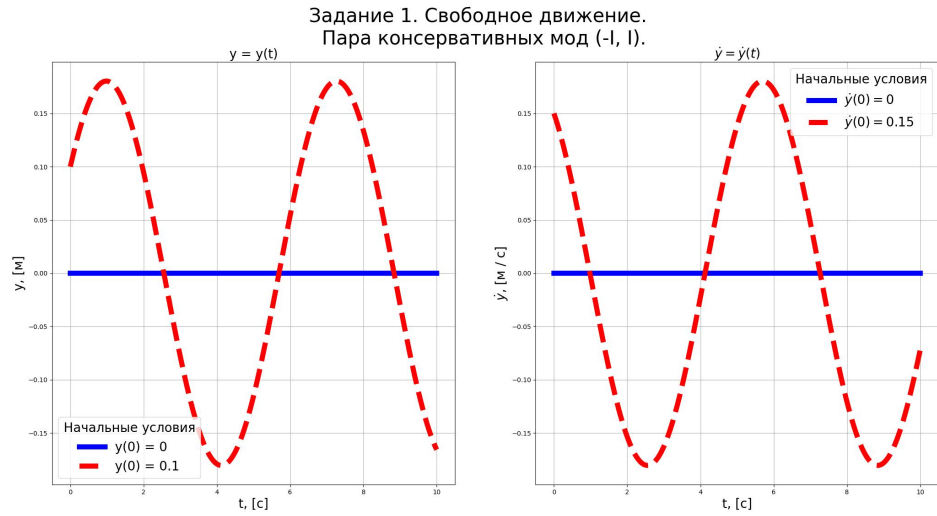


Рис. 3: Результаты пары консервативных мод.

Действительно, все так.

2.2 Задание 2. Область устойчивости.

2.2.1 Теория

Возьмем первый набор корней из первого задания $\lambda_1 = -1$ и $\lambda_2 = 1$. Представим соответствующие им ТФ вида:

$$W(p) = \frac{1}{Tp + 1}$$

Получаем:

$$W_1(p) = \frac{1}{T_1p + 1} = \frac{1}{-1T_1 + 1} \rightarrow T_1 = 1$$

$$W_2(p) = \frac{1}{T_2p + 1} = \frac{1}{1T_2 + 1} \rightarrow T_2 = -1$$

Так же по заданию имеем:

$$W_3(p) = \frac{1}{p}$$

$$W_4(p) = K$$

$$W_{feedback}(p) = -1$$

Тогда получаем общую ТФ системы:

$$W(p) = \frac{W_1(p)W_2(p)W_3(p)W_4(p)}{1 + W_1(p)W_2(p)W_3(p)W_4(p)} = \frac{K}{T_1T_2p^3 + (T_1 + T_2)p^2 + p + K}$$

Для анализа устойчивости воспользуемся следствием из критерия Гурвица для однородного ДУ третьего порядка:

$$\left\{ \begin{array}{l} \frac{K}{T_1T_2} > 0 \\ \frac{T_1+T_2}{T_1T_2} > 0 \\ \frac{K}{T_1T_2} > 0 \\ \frac{T_1+T_2}{(T_1T_2)^2} > \frac{K}{T_1T_2} \end{array} \right.,$$

- Зафиксируем значение $T_2 = -1$

Тогда система принимает вид:

$$\left\{ \begin{array}{l} \frac{K}{T_1(-1)} > 0 \\ \frac{T_1+(-1)}{T_1(-1)} > 0 \\ \frac{K}{T_1(-1)} > 0 \\ \frac{T_1+(-1)}{(T_1(-1))^2} > \frac{K}{T_1(-1)} \end{array} \right.,$$

решений у неё нет. Следовательно, система не может быть устойчивой с таким параметром T_2 .

- Зафиксируем значение $T_1 = 1$

Тогда система принимает вид:

$$\begin{cases} \frac{K}{T_2} > 0 \\ \frac{T_2+1}{T_2} > 0 \\ \frac{K}{T_2} > 0 \\ \frac{T_2+1}{T_2^2} > \frac{K}{T_2} \end{cases},$$

у нее есть решения, представленные на графике ниже.

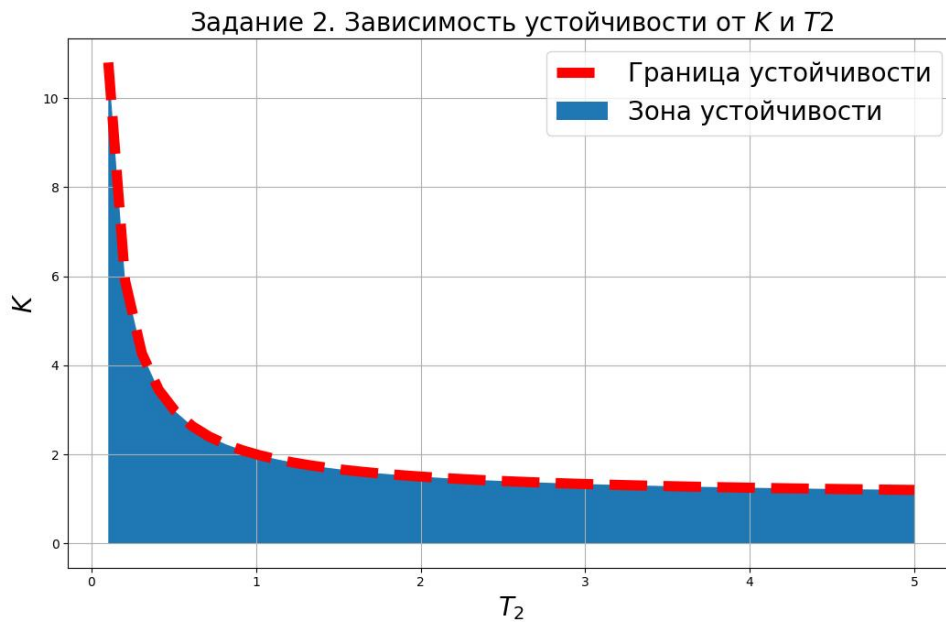


Рис. 4: Зависимость устойчивости от K и T_2 .

2.2.2 Результаты

По приведенным ниже графикам видно, что полученная выше зона устойчивости соответствует действительности.

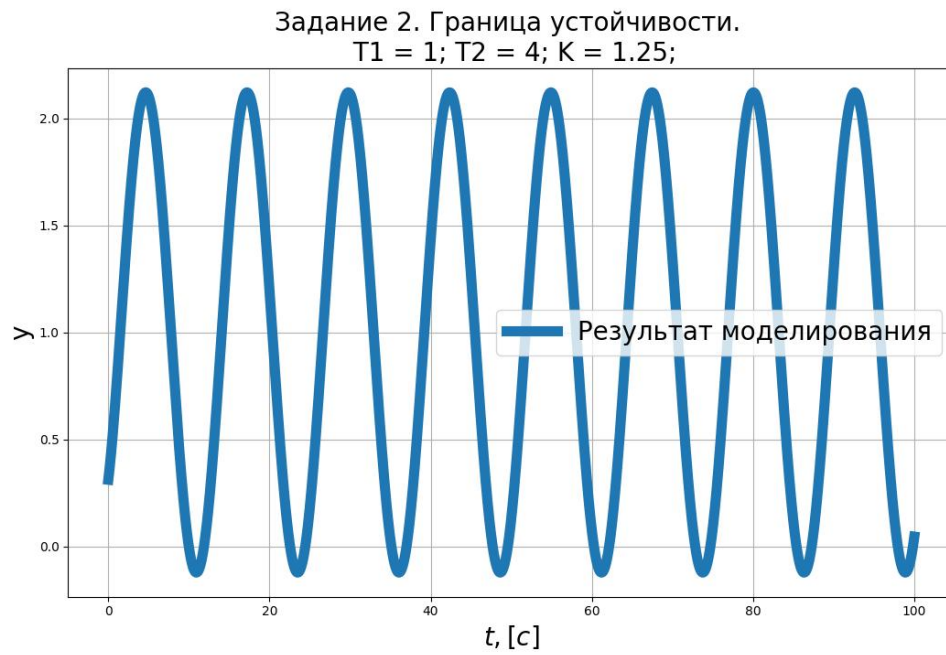


Рис. 5: Устойчивые по Ляпунову параметры.

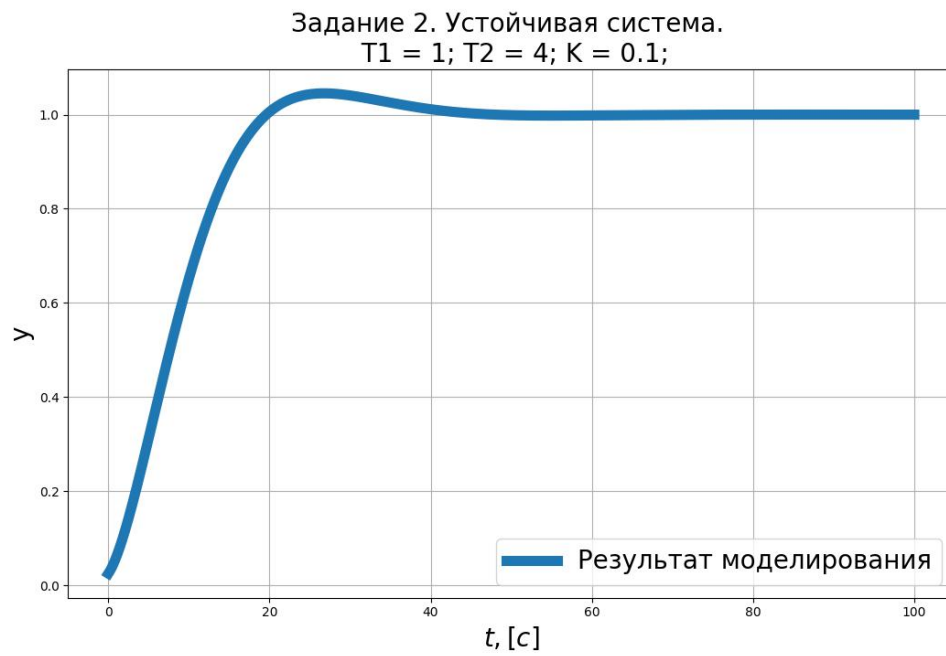


Рис. 6: Устойчивые асимптотически параметры.

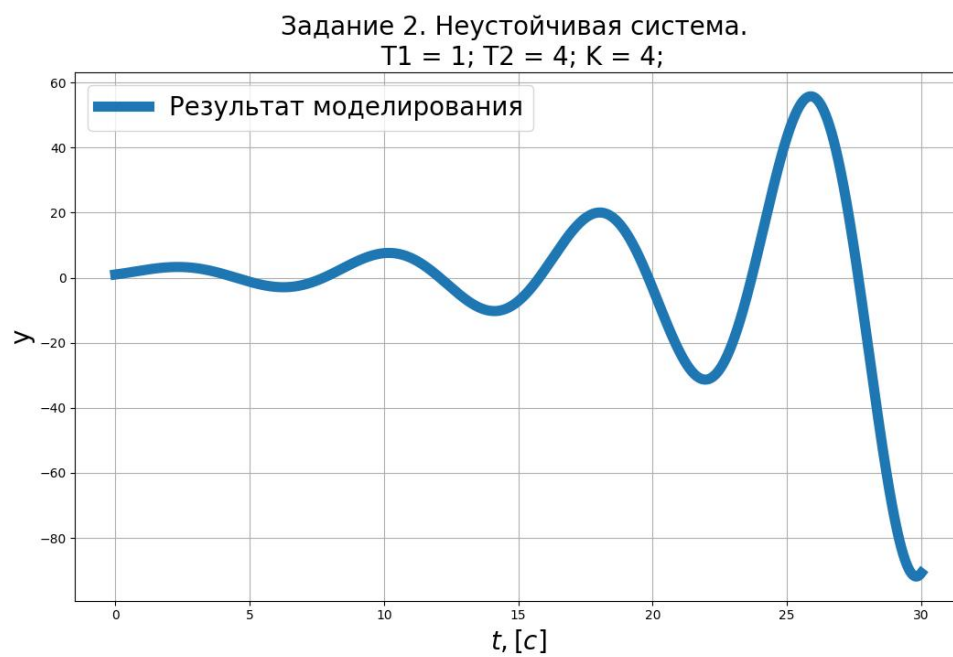


Рис. 7: Неустойчивые параметры.

2.3 Задание 3. Автономный генератор.

2.3.1 Теория

Необходимо найти матрицы A и C такие, чтобы выход системы совпадал с функцией:

$$y(t) = \cos(5t) + e^t + e^{-5t}$$

Из функции ясно видно, что модами будут: $\lambda_1 = 1$; $\lambda_2 = -5$; $\lambda_3 = 5i$; $\lambda_4 = -5i$.

Напишем Жорданову матрицу:

$$A = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & -5 & 0 & 0 \\ 0 & 0 & 0 & 5 \\ 0 & 0 & -5 & 0 \end{bmatrix},$$

2.3.2 Программная реализация

1

2.3.3 Результаты

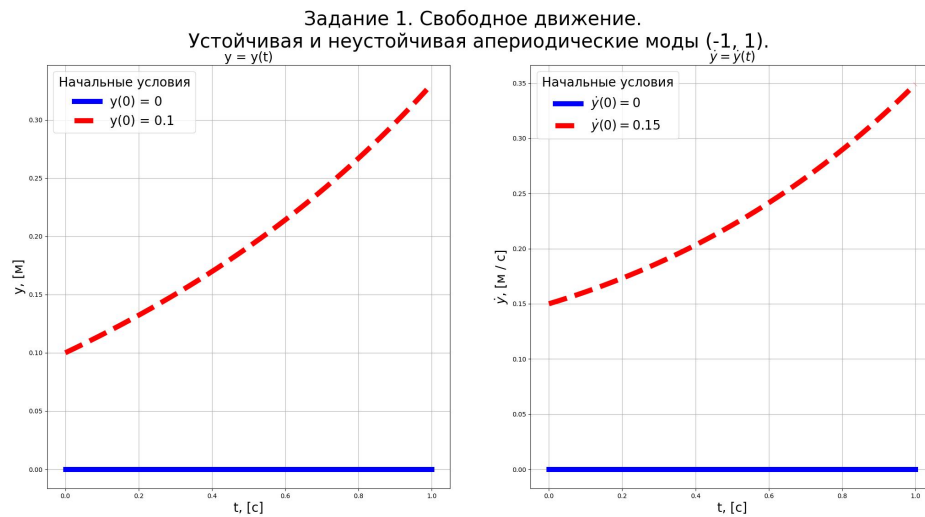


Рис. 8: Результаты третьего задания.

Видно, что графики выхода начинаются не в 0. Это связано с тем, что система не strictly-proper (то есть степени полиномов числителя и знаменателя равны).

Следствие этого заметно при переводе системы в формат SS:

$$A = [-1.67], B = [-0.607 \quad -0.794], C = \begin{bmatrix} 0.594 \\ -7.73 \end{bmatrix}, D = \begin{bmatrix} 0.167 & 0.0833 \\ -0.167 & -0.0833 \end{bmatrix};$$

Матрица связи D получается не нулевой, в связи с чем выход начинает зависеть не только от внутреннего состояния системы через матрицу наблюдения C, но и от входного взаимодействия через матрицу связи D.

2.4 Задание 4. Одноканальная система в форме вход-состояние-выход.

2.4.1 Теория

Необходимо промоделировать SISO систему в формате SS:

$$\begin{cases} \dot{x} = Ax + Bu \\ y = Cx + Du \end{cases},$$

где:

$$A = \begin{bmatrix} 0 & -9 \\ 1 & -6 \end{bmatrix}, B = \begin{bmatrix} 1 \\ 5 \end{bmatrix}, C = \begin{bmatrix} 2 & 5 \end{bmatrix}, D = \begin{bmatrix} 0 \end{bmatrix};$$

2.4.2 Программная реализация

```
1 # Создание системы из матриц B-C-D
2 siso_SS = control.ss(task4_A, task4_B, task4_C, [0])
3
4 # Симуляция
5 small_ts = ts[:5000]
6 siso_SS_y = control.forced_response(siso_SS, U=1, X0=[0, 0], T=small_ts).outputs
```

2.4.3 Результаты

2.5 Задание 5. Многоканальная система в форме вход-состояние-выход.

2.5.1 Теория

Необходимо промоделировать MIMO систему в формате SS:

$$\begin{cases} \dot{x} = Ax + Bu \\ y = Cx + Du \end{cases},$$

где:

$$A = \begin{bmatrix} 0 & -9 \\ 1 & -6 \end{bmatrix}; B = \begin{bmatrix} 1 & 4 \\ 3 & 5 \end{bmatrix}; C = \begin{bmatrix} 2 & 7 \\ 4 & 6 \end{bmatrix}; D = \begin{bmatrix} 0 \end{bmatrix};$$

2.5.2 Программная реализация

```

1 # Создание системы из матриц B-C-B
2 mimo_SS = control.ss(task5_A, task5_B, task5_C, 0)
3
4 # Симуляция
5 mimo_SS_y = control.forced_response(mimo_SS, U=mimo_Us, X0=[0, 0], T=ts).outputs

```

2.5.3 Результаты

3 Заключение

В процессе работы была проведена симуляция в среде Python при помощи Python Control Systems Library для SISO и MIMO систем, данных в двух основных представлениях: TF и SS.

3.1 Выводы

1. Проведено моделирование SISO и MIMO Transfer Function систем с различным ненулевым входным воздействием.
2. Проведено моделирование SISO и MIMO State Space систем с различным ненулевым входным воздействием.
3. Выход системы не зависит от формы представления.
4. Не strictly proper системы могут давать выход отличный от 0 в момент времени $t = 0$ при 0 начальных условиях из-за ненулевой матрицы связи D .