

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ

РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное автономное

образовательное учреждение высшего образования

«Национальный исследовательский университет ИТМО»

(Университет ИТМО)

Факультет Систем управления и робототехники

ЛАБОРАТОРНАЯ РАБОТА № 4

по дисциплине

«Имитационное моделирование робототехнических систем»

Студент:

Группа № R4137c

А. Альмахмуд

Преподаватель:

Ассистент ФСУиР

Е.А. Ракшин

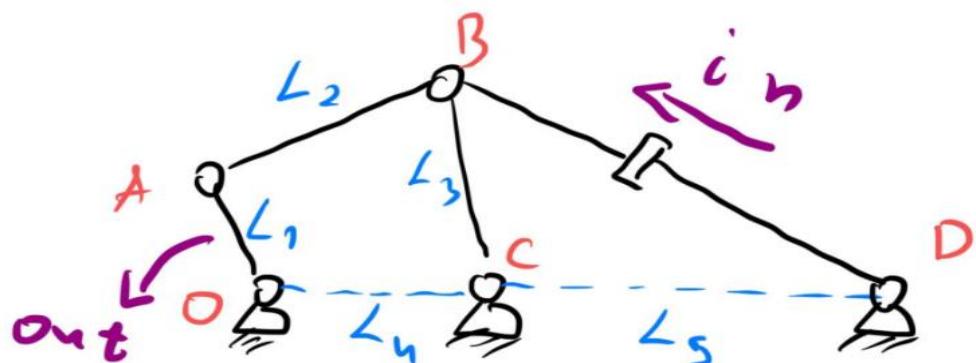
1. Цель

Цель данного проекта — разработка и моделирование механической модели роботизированной руки Optimus с использованием симулятора MuJoCo. Управление движением сустава D осуществляется с помощью PD-регулятора, а движение суставов A, D и C отслеживается с помощью датчиков. Проект направлен на изучение динамики системы, анализ влияния параметров управления на движение и понимание поведения роботизированной руки под воздействием синусоидального управляющего сигнала.

2. Задачи

1. Создать механическую модель роботизированной руки Optimus с использованием симулятора MuJoCo.
2. Добавить исполнительный механизм (Actuator) для управления суставом D.
3. Добавить датчики для отслеживания положения и скорости суставов A, D и C.
4. Применить PD-регулятор для управления движением сустава D с помощью синусоидального управляющего сигнала.
5. Проанализировать влияние параметров управления (K_p , K_d , AMP, BIAS, FREQ) на движение руки.
6. Построить графики результатов и наблюдать отклик системы в реальном времени, смоделировав динамику роботизированной руки.

3. Описание системы



Motors	AMP, deg	FREQ, Hz	BIAS, deg
A	17.49	1.24	-8.7

Mechanism	L1, [m]	L2, [m]	L3, [m]	L4, [m]	L5, [m]
Optimus	0.071	0.0923	0.1065	0.071	0.355

4. Колено робота Optimus

4.1 Построение модели

Таблица 1 – xml код модели

```

1  <mjoco>
2    <option timestep="1e-4"/>
3    <option gravity="0 0 -0.98"/>
4
5    <asset>
6      <texture type="skybox" builtin="gradient" rgb1="1 1 1" rgb2="0.5 0.5 0.5" width="265" height="256"/>
7      <texture name="grid" type="2d" builtin="checker" rgb1="0.1 0.1 0.1" rgb2="0.6 0.6 0.6" width="300" height="300"/>
8      <material name="grid" texture="grid" texrepeat="10 10" reflectance="0.2"/>
9    </asset>
10
11   <worldbody>
12     <light pos="0 0 10"/>
13     <geom type="plane" size="0.5 0.5 1" material="grid"/>
14
15     <camera name="side_view" pos="0.1 -1.5 1.0" euler="0 0 0"/>
16     <camera name="upper_view" pos="0 0 1.5" euler="0 0 0"/>
17
18     <body name="OA" pos="0 0 0" euler="90 0 0">
19       <joint name="O" type="hinge" axis="0 0 1" stiffness="0" springref="0" damping="0"/>
20       <geom name="point O" type="cylinder" pos="0 0 0" size="0.02 0.02" rgba="0.9 0.5 0.1 0.5" euler="0 0 0" contype="0"/>
21       <geom name="link OA" type="cylinder" pos="0 0.071 0" size="0.01 0.071" rgba="0.2 0.8 0.3 0.5" euler="90 0 0" contype="1"/>
22
23     <body name="AB" pos="0 0.092 0" euler="0 0 0">
24       <joint name="A" type="hinge" axis="0 0 1" damping="0" stiffness="0" springref="0"/>
25       <geom name="point B" type="cylinder" pos="0 0 0" size="0.02 0.02" rgba="0.8 0.1 0.7 0.5" euler="0 0 0" contype="0"/>
26       <geom name="link AB" type="cylinder" pos="0 0.0923 0" size="0.01 0.0923" rgba="0.1 0.6 0.9 0.5" euler="90 0 0" contype="1"/>
27       <site name="AB" size="0.015" pos="0 0.1196 0"/>
28     </body>
29
30   <body name="BC" pos="0.057 0 0" euler="0 0 0">
31     <joint name="C" type="hinge" axis="0 0 1" damping="0" stiffness="0" springref="0" stiffness="0"/>
32     <geom name="point C" type="cylinder" pos="0 0 0" size="0.02 0.02" rgba="0.9 0.3 0.4 0.5" euler="0 0 0" contype="0"/>
33     <geom name="link BC" type="cylinder" pos="0 0.1065 0" size="0.01 0.1065" rgba="0.3 0.9 0.2 0.5" euler="90 0 0" contype="1"/>
34     <site name="BC" pos="0 0.138 0" size="0.02"/>
35
36   <body name="BJ" pos="0 0.138 0" euler="0 0 0">
37     <joint name="BJ" type="hinge" axis="0 0 1" damping="0" springref="0" stiffness="0"/>

```

```
38 | | | | |     <geom name="point BJ" type="cylinder" pos="0 0 0" size="0.02 0.02" rgba="0.7 0.2 0.8 0.5" euler="0 0 0" contype="1"/>
39 | | | | |   </body>
40 | | | | | </body>
41 | | | | | </body>
42 |
43 | | | <body name="BD" pos="0.287 0 0" euler="90 0 0">
44 | | | | <joint name="D" type="slide" axis="1 0 0"/>
45 | | | | <geom name="point D" type="cylinder" pos="0 0 0" size="0.02 0.02" rgba="0.2 0.7 0.9 0.8" euler="0 0 0" contype="0"/>
46 | | | | <geom name="link DB" type="capsule" fromto="0 0 0 -0.23 0.138 0" size="0.01" rgba="0.9 0.6 0.1 0.5" euler="90 0 0" contype="1"/>
47 | | | | <site name="DB" size="0.015" pos="-0.23 0.138 0"/>
48 | | | </body>
49 | | </worldbody>
50 |
51 <equality>
52 | <connect site1="DB" site2="BC"/>
53 | <connect site1="BC" site2="AB"/>
54 </equality>
55 |
56 <actuator>
57 | <motor name="m1" joint="D" gear="1"/>
58 </actuator>
59 |
60 |
61 <sensor>
62 | <jointpos joint="A" name="A_pos"/>
63 | <jointvel joint="A" name="A_vel"/>
64 | <jointpos joint="C" name="C_pos"/>
65 | <jointvel joint="C" name="C_vel"/>
66 | <jointpos joint="D" name="D_pos"/>
67 | <jointvel joint="D" name="D_vel"/>
68 | </sensor>
69 </mjoco>
```

4.2 Симуляция

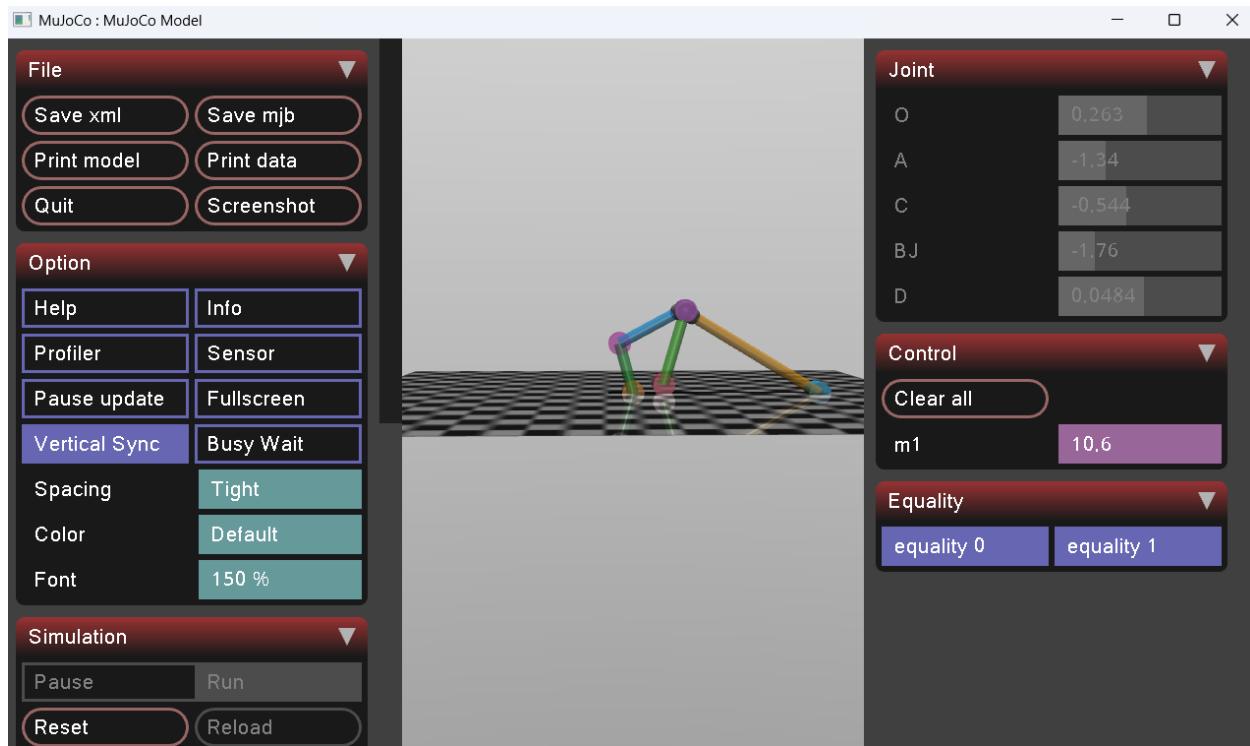
Таблица 2 - Код для симуляции системы

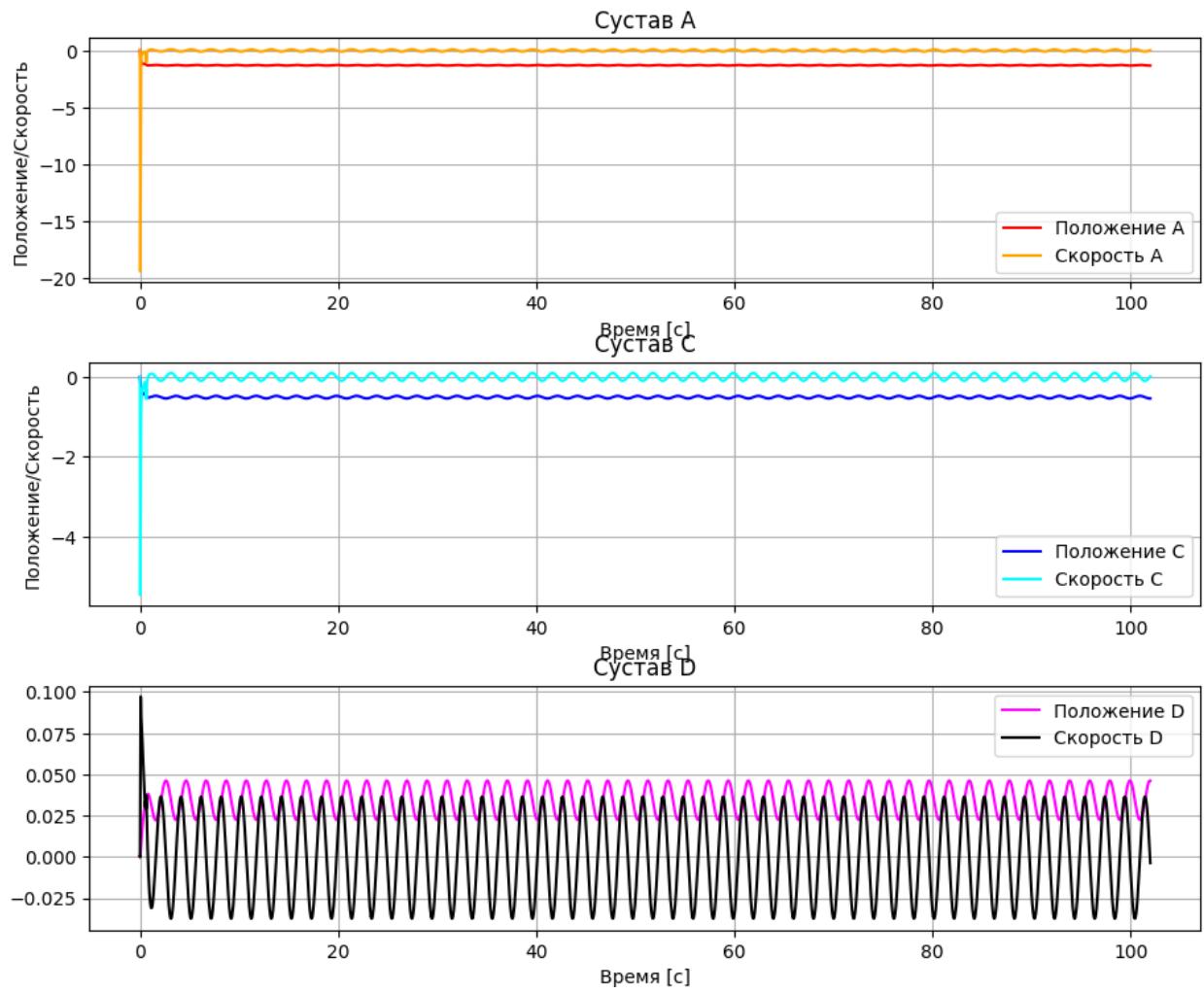
```
1 import mujoco
2 import mujoco.viewer
3 import numpy as np
4 import matplotlib.pyplot as plt
5
6 # Загрузка модели
7 model = mujoco.MjModel.from_xml_path("optimus_knee_myprojectttt.xml")
8 data = mujoco.MjData(model)
9
10 # ---- Параметры синусоидального сигнала ----
11 AMP = np.deg2rad(17.49)
12 FREQ = 1.24
13 BIAS = np.deg2rad(-8.7)
14
15 # ---- Параметры PD-регулятора ----
16 KP = 50
17 KD = 20
18
19 # ---- Индекс сустава "D" для управления ----
20 idx_D = model.joint("D").id
21
22 # --- Списки для хранения данных ---
23 t = 0
24 dt = model.opt.timestep
25
26 time_list = []
27 pos_list_A = []
28 vel_list_A = []
29 pos_list_C = []
30
31 vel_list_C = []
32 pos_list_D = []
33 vel_list_D = []
34 force_list = []
35
36 with mujoco.viewer.launch_passive(model, data) as viewer:
37     while viewer.is_running():
38         # ---- Желаемые значения для управления ----
39         q_des = AMP * np.sin(FREQ * t) + BIAS
40         qd_des = AMP * FREQ * np.cos(FREQ * t)
41
42         # ---- Текущее состояние сустава D ----
43         q = data.qpos[idx_D]
44         qdot = data.qvel[idx_D]
45
46         # ---- PD-регулятор ----
47         force = KP*(q_des - q) + KD*(qd_des - qdot)
48         data.ctrl[0] = force
49
50         # ---- Шаг симуляции ----
51         mujoco.mj_step(model, data)
52
53         # ---- Сохранение данных ----
54         time_list.append(t)
55         force_list.append(force)
56         pos_list_A.append(data.sensordata[0])
57         vel_list_A.append(data.sensordata[1])
58         pos_list_C.append(data.sensordata[2])
59         vel_list_C.append(data.sensordata[3])
```

```
59     pos_list_D.append(data.sensordata[4])
60     vel_list_D.append(data.sensordata[5])
61
62     viewer.sync()
63     t += dt
64
65     # ---- Построение графиков для каждого сустава ----
66     plt.figure(figsize=(12,10))
67
68     # Сустав А
69     plt.subplot(*args: 3,1,1)
70     plt.plot(*args: time_list, pos_list_A, label="Положение А", color="red")
71     plt.plot(*args: time_list, vel_list_A, label="Скорость А", color="orange")
72     plt.title("Сустав А")
73     plt.xlabel("Время [с]")
74     plt.ylabel("Положение/Скорость")
75     plt.grid(True)
76     plt.legend()
77
78     # Сустав С
79     plt.subplot(*args: 3,1,2)
80     plt.plot(*args: time_list, pos_list_C, label="Положение С", color="blue")
81     plt.plot(*args: time_list, vel_list_C, label="Скорость С", color="cyan")
82     plt.title("Сустав С")
83     plt.xlabel("Время [с]")
84     plt.ylabel("Положение/Скорость")
85     plt.grid(True)
```

```
85     plt.grid(True)
86     plt.legend()
87
88     # Сустав D
89     plt.subplot(*args: 3,1,3)
90     plt.plot(*args: time_list, pos_list_D, label="Положение D", color="magenta")
91     plt.plot(*args: time_list, vel_list_D, label="Скорость D", color="black")
92     plt.title("Сустав D")
93     plt.xlabel("Время [с]")
94     plt.ylabel("Положение/Скорость")
95     plt.grid(True)
96     plt.legend()
97
98     plt.tight_layout()
99     plt.show()
100
```

4.3 Графики





Вывод:

Использование трех датчиков для суставов A, C и D позволило эффективно отслеживать движение ключевых точек роботизированной руки Optimus. Применение PD-регулятора обеспечило ожидаемые синусоидальные колебания сустава D, а графики положения и скорости показали корректное и предсказуемое поведение всей системы. Проект подтвердил правильность настройки параметров управления и продемонстрировал влияние амплитуды, частоты и коэффициентов PD на динамику движения.