

**федеральное государственное автономное образовательное  
учреждение высшего образования**

# **«Национальный исследовательский университет ИТМО»**

Факультет систем управления и информатики

**Практическая работа 4**

Выполнил:

Нагорный Л.А.

Группа:

R4135c

Преподаватель:

Ракшин Е.А.

Санкт-Петербург

2025

## 1. Задание варианта

Параметры системы:

AMP,Deg	FREQ,Hz	BIAS,Deg
55.34	3.95	0.1035

$$q^{des} = AMP \cdot \sin(FREQ \cdot t) + BIAS$$

## 2. Решение

Изменим конфигурацию механизма, добавив сенсор и актуатор.

XML файл системы имеет вид:

```
<?xml version='1.0' encoding='UTF-8'?>

<mujoco>

    <option timestep="1e-3"/>
    <option gravity="0 0 -9.8"/>

    <asset>
        <texture type="skybox" builtin="gradient" rgb1="1 1 1"
rgb2="0.5 0.5 0.5" width="265" height="256"/>
        <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"/>
        <material name="grid" texture="grid" texrepeat="10 10"
reflectance="0.2"/>
    </asset>

    <worldbody>

        <light pos="0 0 10"/>
        <geom type="plane" size="1.0 1.0 0.1" material="grid"/>

        <!-- Первое звено OAB1 -->
        <body name="OAB1" pos="0 0 1.5">
            <joint name="O" type="hinge" axis="0 -1 0" limited="true"
range="-180 180"/>
            <geom name="point_O" type="sphere" pos="0 0 0" size="0.008"
rgba="0.89 0.14 0.16 1.0"/>
            <geom name="link_OA" type="cylinder" pos="0 0 0.0345"
size="0.005 0.0345" rgba="0.21 0.32 0.82 0.8"/>
        </body>
    </worldbody>
</mujoco>
```

```

<body name="AB1" pos="0 0 0.069">
    <joint name="A" type="hinge" axis="0 -1 0"
limited="true" range="-180 180"/>
        <geom name="point_A" type="sphere" pos="0 0 0"
size="0.008" rgba="0.89 0.14 0.16 1.0"/>
        <geom name="link_AB1" type="cylinder" pos="0 0 0.04485"
size="0.005 0.04485" rgba="0.21 0.32 0.82 0.8"/>
        <site name="end_B1" type="sphere" size="0.01" pos="0 0
0.0897" rgba="1.0 1.0 0.0 1.0"/>
    </body>
</body>

<!-- Второе звено CB2 -->
<body name="CB2" pos="0.2 0 1.5">
    <joint name="C" type="hinge" axis="0 -1 0" limited="true"
range="-180 180"/>
        <geom name="point_C" type="sphere" pos="0 0 0" size="0.008"
rgba="0.14 0.89 0.16 1.0"/>
        <geom name="link_CB2" type="cylinder" pos="0 0 0.05175"
size="0.005 0.05175" rgba="0.14 0.89 0.16 0.8"/>
        <site name="end_B2" type="sphere" size="0.01" pos="0 0
0.1035" rgba="0.0 1.0 1.0 1.0"/>
    </body>

<!-- Третье звено DFB3 -->
<body name="DFB3" pos="-0.2 0 1.5">
    <joint name="D" type="hinge" axis="0 -1 0" limited="true"
range="-180 180"/>
        <geom name="point_D" type="sphere" pos="0 0 0" size="0.008"
rgba="0.89 0.14 0.82 1.0"/>
        <geom name="link_DB3" type="cylinder" pos="0 0 0.0345"
size="0.005 0.0345" rgba="0.89 0.14 0.82 0.8"/>

    <body name="FB3" pos="0 0 0.069">
        <joint name="slider" type="slide" axis="0 0 1"
limited="true" range="-0.3 0.3"/>
            <geom name="point_F" type="sphere" pos="0 0 0"
size="0.008" rgba="0.89 0.14 0.82 1.0"/>
            <geom name="link_FB3" type="cylinder" pos="0 0 0.1725"
size="0.005 0.1725" rgba="0.89 0.14 0.82 0.8"/>
            <site name="end_B3" type="sphere" size="0.01" pos="0 0
0.345" rgba="1.0 0.5 0.0 1.0"/>
    </body>

```

```

</body>

</worldbody>

<!-- ===== Закрытие контура (равенство B1, B2, B3) ===== -->
<equality>
    <connect site1="end_B1" site2="end_B2"/>
    <connect site1="end_B1" site2="end_B3"/>
</equality>

<!-- ===== SENSORS + ACTUATOR ===== -->

<sensor>
    <jointpos joint="D" name="sens_pos_D"/>
    <jointvel joint="D" name="sens_vel_D"/>
</sensor>

<actuator>
    <motor joint="D" name="act_D" ctrlrange="-200 200"/>
</actuator>

</mujoco>

```

## Получим следующий механизм

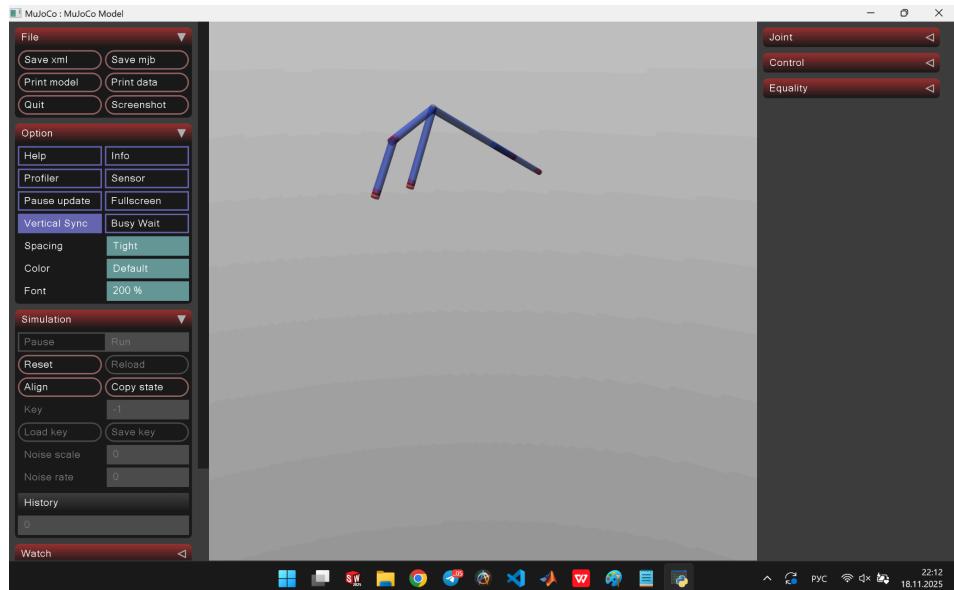
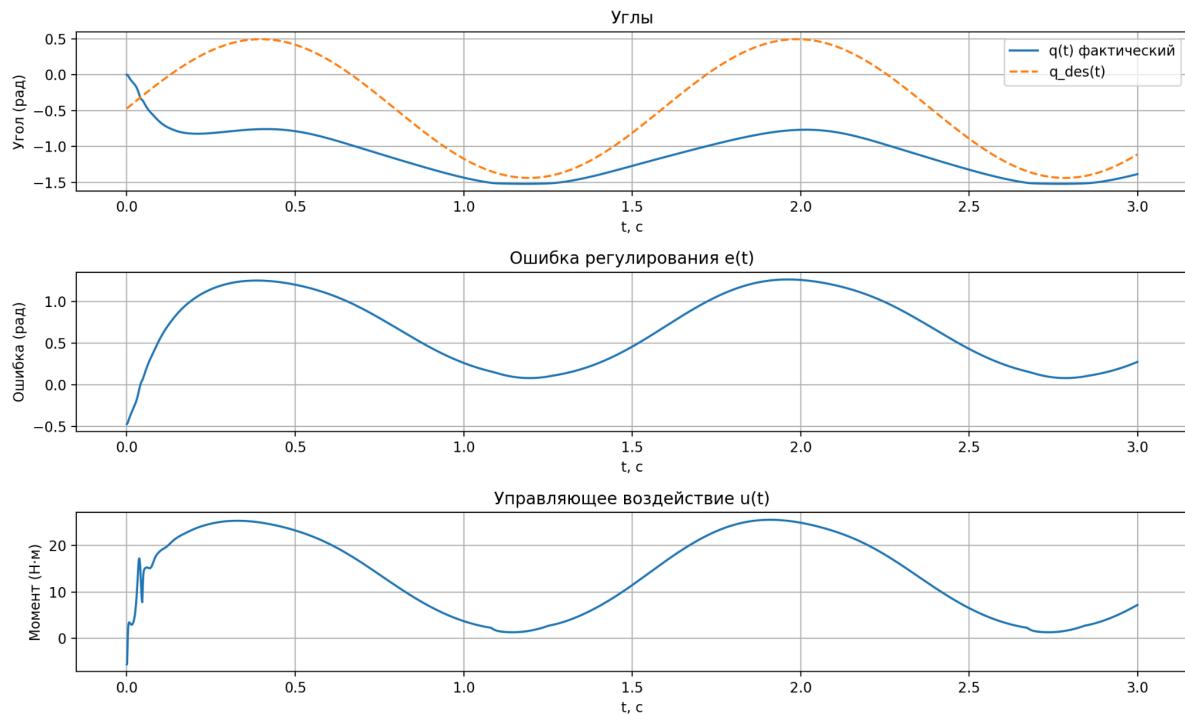


Рисунок 1 - Механизм варианта

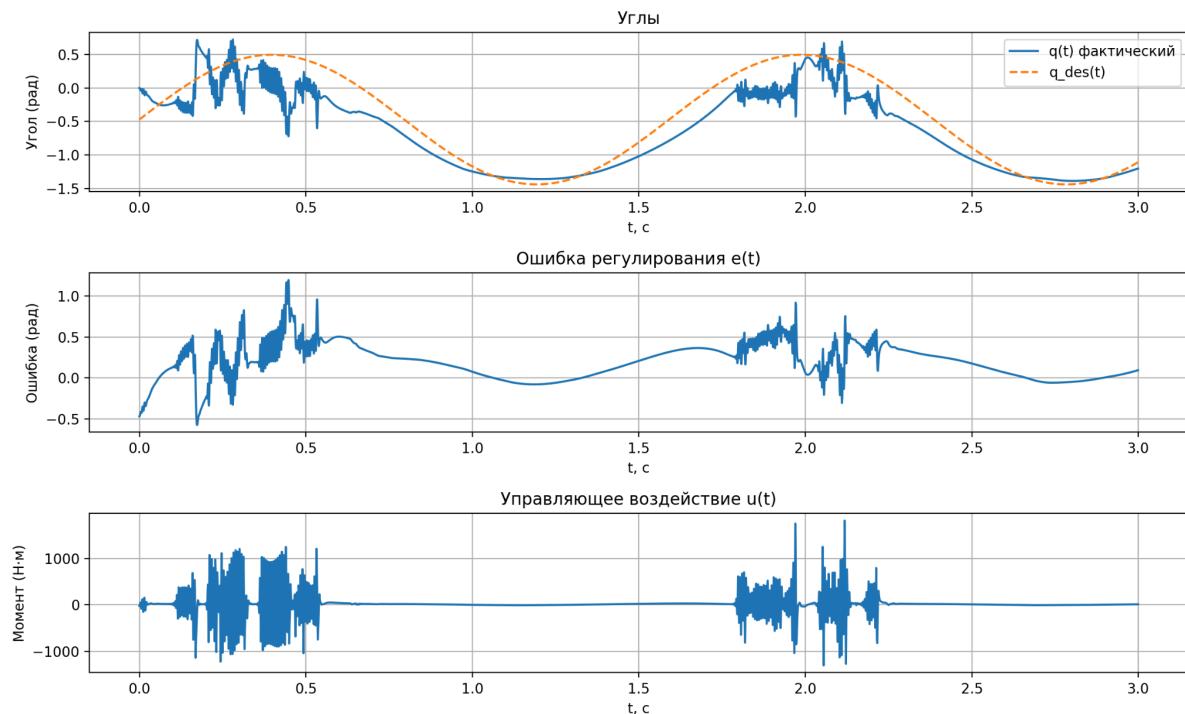
### 3. Результаты

При разных значениях линейной и дифференциальной составляющих получим следующее:

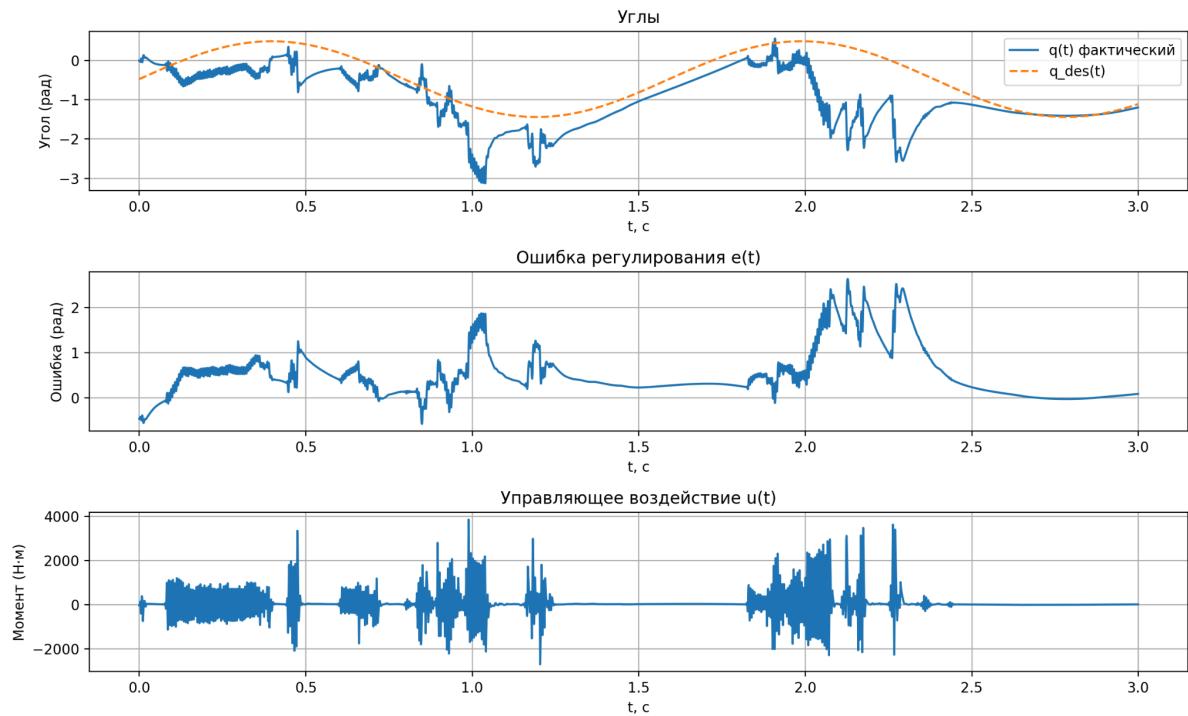
Нестабильный режим | KP=20, KD=1



Промежуточный режим | KP=80, KD=4



Устойчивый режим | KP=120, KD=8



Получившийся код:

```
import time
import numpy as np
import mujoco
import matplotlib.pyplot as plt

XML_PATH = "optimus.xml"

# ===== Управляющая траектория Optimus =====
AMP = np.deg2rad(55.34)
FREQ = 3.95
BIAS = np.deg2rad(-27)

def q_desired(t):
    return AMP * np.sin(FREQ * t) + BIAS

def dq_desired(t):
    return AMP * FREQ * np.cos(FREQ * t)

# ===== Испытательные наборы параметров =====
```

```

TEST_CASES = [
    ("Нестабильный режим", 20, 1),
    ("Промежуточный режим", 80, 4),
    ("Устойчивый режим", 120, 8),
]

def run_test(model, case_name, KP, KD, sim_time=3.0, dt=0.001):

    steps = int(sim_time / dt)

    # Буферы для переходных процессов
    t_arr = np.zeros(steps)
    q_arr = np.zeros(steps)
    u_arr = np.zeros(steps)
    qd_arr = np.zeros(steps)

    data = mujoco.MjData(model)

    motor_id = mujoco.mj_name2id(model, mujoco.mjtObj.mjOBJ_ACTUATOR,
"act_D")

    for i in range(steps):
        t = i * dt

        qd = q_desired(t)
        dq = dq_desired(t)

        q = data.sensordata[0]
        dq_real = data.sensordata[1]

        u = KP * (qd - q) + KD * (dq - dq_real)
        data.ctrl[motor_id] = u

        mujoco.mj_step(model, data)

        # Записываем графики
        t_arr[i] = t
        q_arr[i] = q
        qd_arr[i] = qd
        u_arr[i] = u

# ===== РИСОВАНИЕ ГРАФИКОВ =====

```

```

plt.figure(figsize=(12, 8))

# >>> Добавлен заголовок с коэффициентами <<<
plt.suptitle(f"case_name | KP={KP}, KD={KD}", fontsize=14)

plt.subplot(3, 1, 1)
plt.plot(t_arr, q_arr, label="q(t) фактический")
plt.plot(t_arr, qd_arr, label="q_des(t)", linestyle="--")
plt.xlabel("t, c")
plt.ylabel("Угол (рад)")
plt.title("Углы")
plt.grid()
plt.legend()

plt.subplot(3, 1, 2)
plt.plot(t_arr, qd_arr - q_arr)
plt.xlabel("t, c")
plt.ylabel("Ошибка (рад)")
plt.title("Ошибка регулирования e(t)")
plt.grid()

plt.subplot(3, 1, 3)
plt.plot(t_arr, u_arr)
plt.xlabel("t, c")
plt.ylabel("Момент (Н·м)")
plt.title("Управляющее воздействие u(t)")
plt.grid()

# Чтобы suptitle не перекрывал графики
plt.tight_layout(rect=[0, 0, 1, 0.96])

plt.savefig(f"transition_{case_name}.png", dpi=200)
plt.close()

print(f"[OK] Сохранён рисунок: transition_{case_name}.png")

# ===== Главный вызов =====
def main():
    print("Загрузка модели...")
    model = mujoco.MjModel.from_xml_path(XML_PATH)

```

```
for case_name, KP, KD in TEST_CASES:
    print(f"\n--- Тест: {case_name} ---")
    print(f"KP={KP}, KD={KD}")
    run_test(model, case_name, KP, KD)

print("\nГотово. Все графики сохранены.")

if __name__ == "__main__":
    main()
```

#### 4. Вывод

В ходе работы была выполнена численная симуляция динамики механизма Optimus в MuJoCo и анализ переходного процесса при синусоидальном законе изменения желаемого угла. В XML-модель добавлены секции `<actuator>` и `<sensor>`, что позволило использовать мотор в шарнире 0 и датчик его положения. В Python-скрипте реализован пропорционально-дифференциальный закон управления.

Проведены испытания трёх наборов коэффициентов регулятора (KP, KD), соответствующих неустойчивому, промежуточному и устойчивому режимам. Построенные графики угла, ошибки и управляющего момента показали, что увеличение коэффициентов улучшает слежение за траекторией и уменьшает колебательность системы. Наилучший переходный процесс достигался при максимальных параметрах (KP = 120, KD = 8).

Таким образом, симуляция позволила оценить влияние параметров П-Д регулятора и подтвердить эффективность увеличения коэффициентов для повышения устойчивости и точности управления.