THE MINISTRY OF SCIENCE AND HIGHER EDUCATION OF THE RUSSIAN FEDERATION

ITMO University

Report for the task 3 " Optimus knee closed-chain mechanism"

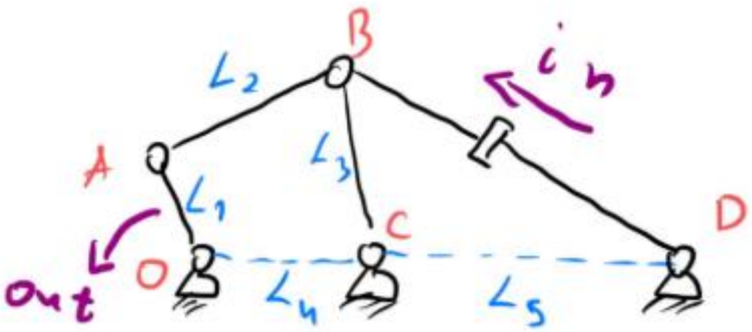
for the subject

SIMULATION OF ROBOTIC SYSYTEMS

Student:

ID_Number 503259 *Afif, Hazem*

Saint Petersburg 14/11/2025



Optimus' knee mechanism overview

A closed-loop linkage system that can convert the linear movement of a slider joint into rotational movement for a hinge joint and vice versa. This system operates on the principle that when the link length follows a specific ratio, half of the slider's movement approximates a straight line. This distinctive feature makes it suitable for use in the knee joints of walking robots.

In this task, we will simulate the passive mechanism **without setting the mechanism in motion**. Thus, we will add a link between the point B and D instead of adding a motor.

XML file

The parameters of the figure are shown in the following table:

Parameter	L_1	L_2	L_3	L_4	L_5
Value (m)	0.044	0.0572	0.066	0.044	0.22

To begin, we wrote an XML file to model the closed-loop mechanism in MuJoCo. This involved creating the links and joints. Implementing a closed-loop in MuJoCo requires the use of the equality connect attribute. To achieve this, we added the necessary sites to apply constraints that connect two bodies at specific points.

```
<mujoco model="Optimus">
 <option gravity="0 -9.81 0" timestep="0.0001" solver="Newton" iterations="200" tolerance="1e-8"/>
 <default>
    <site rgba="0 0 1 1"/>
    <geom rgba="1 0 0 1"/>
 </default>
 <worldbody>
    <light pos="0 0 3"/>
    <geom type="plane" size="5 5 0.1" rgba=".5 .5 .5 .7" />
    <body name="base" pos="0 0 0.5">
        <geom type="sphere" size="0.015" rgba="0 0 1 1"/>
    </body>
    <body name="L1" pos="0 0 0.5">
        <joint name="O" type="hinge" axis="0 1 0" range="-90 90"/>
        <geom type="cylinder" fromto="0 0 0 0 0 0.044" size="0.01"/>
        <body name="L2" pos="0 0 0.044">
            <joint name="A" type="hinge" axis="0 1 0" range="-90 90"/>
            <geom name="A" type="sphere" size="0.015" rgba="1 1 1 1"/>
            <geom name="L2" type="cylinder" fromto="0 0 0 0.044 0 0.022" size="0.01" rgba="1 0 0 1"/>
            <site name="s1" pos="0.044 0 0.022" size="0.005" rgba="0 1 0 1"/>
        </body>
```

```xml
    </body>
    <body name="L3_base" pos="0.044 0 0.5">
        <geom type="sphere" size="0.015" rgba="0 0 1 1"/>
    </body>
    <body name="L3" pos="0.044 0 0.5">
        <joint name="C" type="hinge" axis="0 1 0" range="-90 90"/>
        <geom type="cylinder" fromto="0 0 0 0 0 0.066" size="0.01"/>
        <site name="s2" pos="0 0 0.066" size="0.005" rgba="1 0 1 1"/>
        <geom name="B_to_D" type="cylinder" fromto="0 0 0.066 0.22 0 0" size="0.008" rgba="0 1 0 1"/>
        <body pos="0 0 0.066">
            <geom name="B" type="sphere" size="0.015" rgba="1 1 1 1"/>
        </body>
    </body>
    <body name="L5" pos="0.264 0 0.5">
        <joint name="D" type="hinge" axis="0 1 0" range="-90 180"/>
        <geom type="sphere" size="0.015"/>
        <site name="D_point" pos="0 0 0" size="0.005"/>
    </body>
  </worldbody>
  <equality>
    <connect site1="s1" site2="s2"/>
  </equality>
</mujoco>
```

Python Code

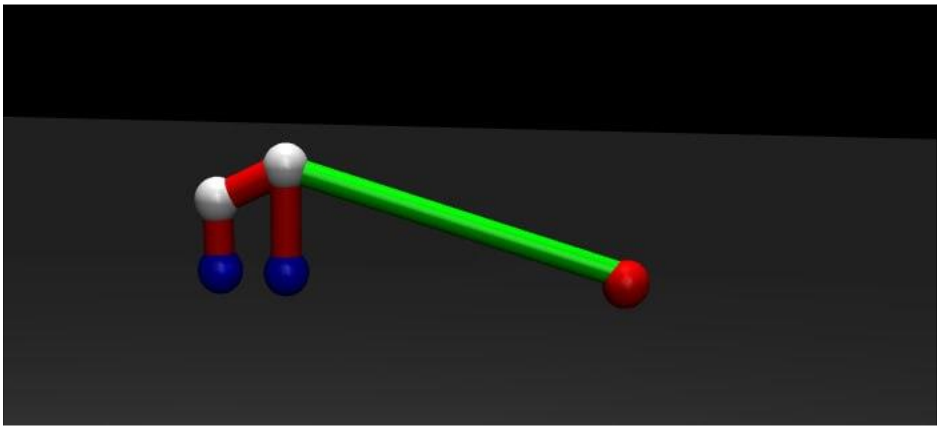
The next step is to create a Python script to load the mechanism model from an XML file that defines all the bodies and joints and then to display the mechanism in the MuJoCo environment. The script keeps the simulation running until we manually stop it by pressing Enter.

```python
import mujoco
from mujoco.viewer import launch_passive

# Load model
model_path = "C:\Users\user\OneDrive\Desktop\MuJoCo\hazem.xml"
model = mujoco.MjModel.from_xml_path(model_path)
data = mujoco.MjData(model)

# Step a few times to initialize simulation
for _ in range(100):
    mujoco.mj_step(model, data)

launch_passive(model, data)

input("Press Enter to exit...")
```

The simulation uses ***mujoco.MjModel*** to load the XML model and ***mujoco.viewer.launch_passive*** to open an interactive viewer.

Results

The model in the MuJoCo environment is shown in the following figure:



In this task, we made an XML file to model the passive Optimus knee closed chain mechanism -without adding an actuator- using MuJoCo environment and we write a Python script using mujoco and mujoco.viewer libraries to run the simulation.