FEDERAL STATE AUTONOMOUS EDUCATIONAL INSTITUTION OF HIGHER EDUCATION
NATIONAL RESEARCH UNIVERSITY ITMO
ITMO UNIVERSITY

FACULTY OF CONTROL SYSTEMS AND ROBOTICS

PROGRAM:
ROBOTICS AND ARTIFICIAL INTELLIGENCE

LABORATORY REPORT

SIMULATION OF ROBOTIC SYSTEMS
PRACTICAL WORK №3

STUDENT
Zongo Sabane

PROFESSOR
Rakshin Egor Aleksandrovich

Academic Year: 2025

This report presents the development of a simulation model for a robotic mechanism, implemented in Mujoco using provided geometric parameters.The model represents a three-branch articulated system constrained to meet at a common point.

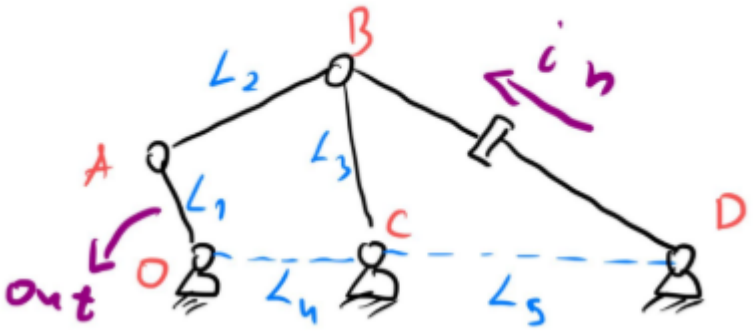
STATEMENT

This task involves developing a Python script that integrates model, data, and viewer based on the parameters provided, and then executing the simulation.

parameters:

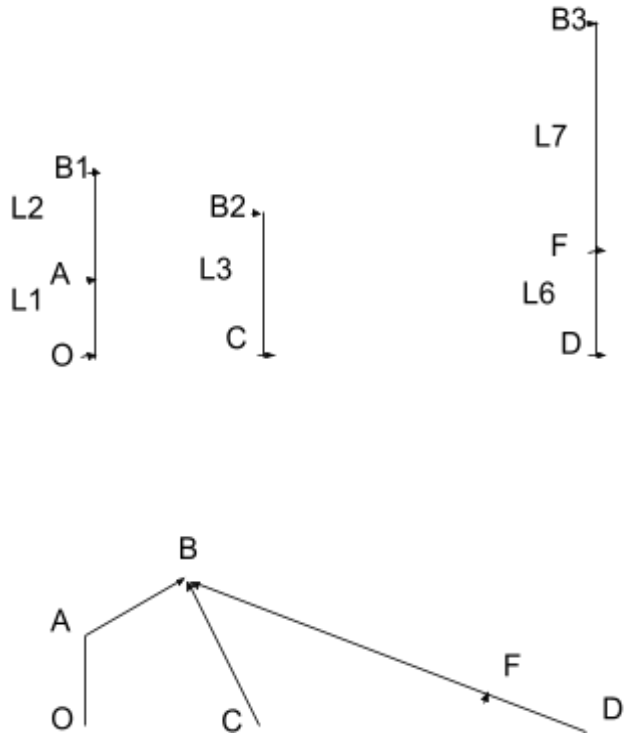
L1, m	L2, m	L3, m	L4, m	L5, m
0.072	0.0936	0.108	0.072	0.36

Variant 2 - Optimus' knee closed-chain mechanism:



SOLVING

The figures below represent the detailed diagram of our mechanism.



Optimus' knee closed chain mechanism

Lengths:

L1, m	L2, m	L3, m	L6, m	L7, m
0.072	0.0936	0.108	0.072	0.36

I. SCRIPT

```
import mujoco
import mujoco.viewer
import time


xml = """

<mujoco>
  <option timestep="0.001"/>
  <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="0.5 0.5 0.1" material="grid"/>


    <!-- Branch 1: L1 + L2 -->
    <body name="OAB1" pos="0 0 0.5">
      <joint name="O" type="hinge" axis="0 -1 0" stiffness="0"
damping="0"/>
      <geom name="point O" type="cylinder" pos="0 0 0" size="0.005
0.005" rgba="0.89 0.14 0.16 0.5" euler="0 0 0" contype="0"/>
      <geom name="link_OA" type="cylinder" pos="0 0 0.036" size="0.005
0.036" rgba="0.2 0.4 0.9 0.7"/>
      <body name="AB1" pos="0 0 0.072">
        <joint name="A" type="hinge" axis="0 -1 0" damping="0.1"/>
        <geom name="point A" type="cylinder" pos="0 0 0" size="0.005
0.005" rgba="0.89 0.14 0.16 0.5" euler="0 0 0" contype="0"/>
        <geom name="link_AB1" type="cylinder" pos="0 0 0.0468"
size="0.005 0.0468" rgba="0.2 0.4 0.9 0.7"/>
```

```
        <site name="sC1" pos="0 0 0.0936" size="0.005"/>
      </body>
    </body>


    <!-- Branch 2: L3 -->
    <body name="CB2" pos="0.072 0 0.5">
      <joint name="C" type="hinge" axis="0 -1 0" damping="0.1"/>
       <geom name="point C" type="cylinder" pos="0 0 0" size="0.005
0.005" rgba="0.89 0.14 0.16 0.5" euler="0 0 0" contype="0"/>
      <geom name="link_CB2" type="cylinder" pos="0 0 0.054" size="0.005
0.054" rgba="0.2 0.4 0.9 0.7"/>
      <site name="sC2" pos="0 0 0.108" size="0.005"/>
    </body>


    <!-- Branch 3: L6 + L7 -->
    <body name="DFB3" pos="0.36 0 0.5">
      <joint name="D" type="hinge" axis="0 -1 0" damping="0"/>
      <geom name="point D" type="cylinder" pos="0 0 0" size="0.005
0.005" rgba="0.89 0.14 0.16 0.5" euler="0 0 0" contype="0"/>
      <geom name="link_DF" type="cylinder" pos="0 0 0.036" size="0.005
0.036" rgba="0.2 0.4 0.9 0.7"/>
      <body name="FB3" pos="0 0 0.072">
        <joint name="slider" type="slide" axis="0 0 1" limited="true"
range="-0.2 0.2" damping="0"/>
        <geom name="point F" type="cylinder" pos="0 0 0" size="0.005
0.005" rgba="0.89 0.14 0.16 0.5" euler="0 0 0" contype="0"/>
        <geom name="link_FB3" type="cylinder" pos="0 0 0.18"
size="0.005 0.18" rgba="0.2 0.4 0.9 0.7"/>
        <site name="sC3" pos="0 0 0.36" size="0.005"/>
      </body>
    </body>
  </worldbody>


  <!-- all branches meet at common point -->
  <equality>
    <connect  site1="sC1" site2="sC2"/>
    <connect  site1="sC1" site2="sC3"/>
  </equality>
</mujoco>
"""
```

```
model = mujoco.MjModel.from_xml_string(xml)
data = mujoco.MjData(model)

with mujoco.viewer.launch_passive(model, data) as viewer:
    start = time.time()
    while viewer.is_running() and time.time() - start < 35:
        mujoco.mj_step(model, data)
        viewer.sync()
        time.sleep(0.11)
```

II. SIMULATION RESULT