Robot Dynamics & Control

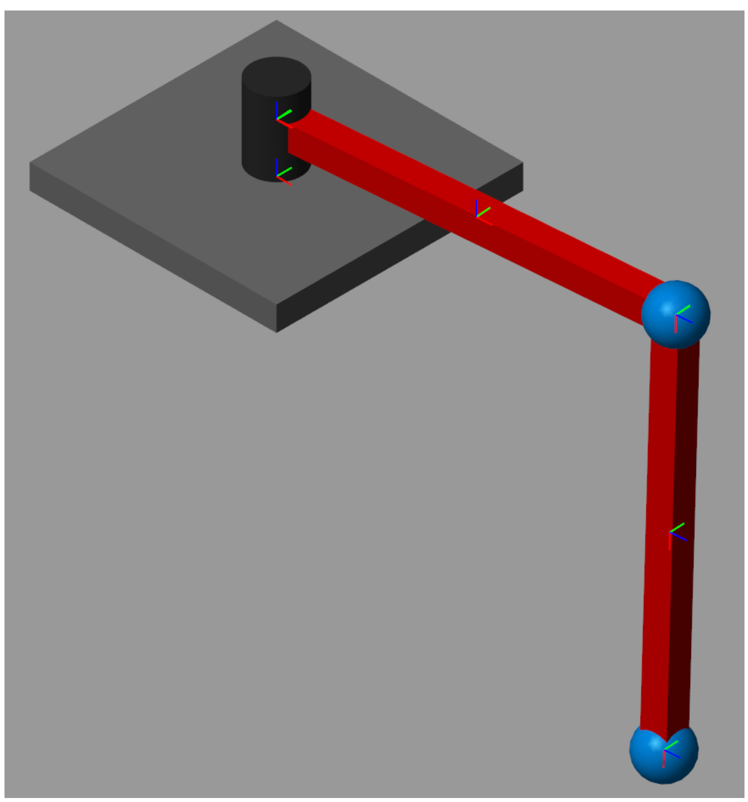
January, 29th, 2024

Duration - 2:30 h

Important notes:

- It is advised to define the link widths and joint sizes reasonably.
- It is strongly advised to define the manipulator frames according to Denavit-Hartenberg convention. Incorrectly defined frames can lead to grade reduction.
- If the model parameters / joint positions / trajectories do not correspond to those defined in the task, the corresponding task will not be graded.
- If the model uses workspace variables, a *.mat* file or an *.m* script defining them must be included, otherwise the corresponding tasks will not be graded.
- In the tasks using *RigidBodyTree*, a *.mat* file or an *.m* script defining it must be included, otherwise the corresponding tasks will not be graded.

2R 3D Robot



- Consider a 3-dimensional 2R robot like shown in figure (**second joint axis should be orthogonal to first joint axis**) with the following set of parameters:
- Mass of link 1: 2 Kg
- Length of link 1: 40 cm
- Mass of link 2: 6.5 Kg

- Length of link 2: 30 cm
- **Gravity** should act on the **z-axis of the world frame**

Task 1: Statics

1) Consider the joint configuration q = [10°, 75°] . Implement gravity compensation using the manipulator statics.
2) Implement gravity compensation in the same configuration using the manipulator statics considering a spherical end-effector of mass Me = 1.6 Kg.
3) Implement gravity compensation in the same configuration using the manipulator statics considering a spherical end-effector of mass Me = 6.8 Kg and an additional mass of 1.2 Kg (modeled as a spherical object) placed on link 2 at 1/3 of its total length.

   Question: How does the gravity torque change if the additional mass described in point 3) is placed on Link 1 instead? Answer on a .txt file and include it to the submission.

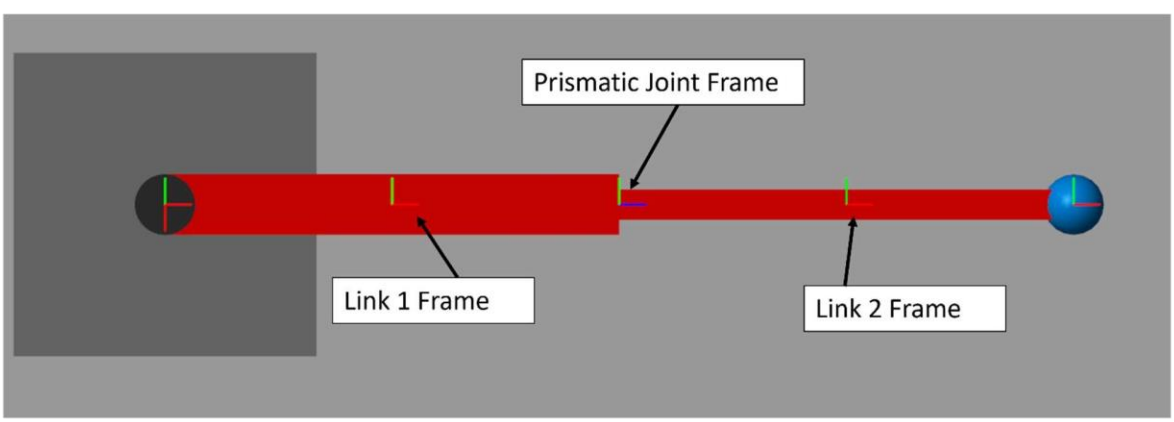
Task 2: Newton-Euler Algorithm

1) Implement the Newton-Euler algorithm for recursive inverse dynamics. Generate a cubic polynomial trajectory with the following initial and final configurations: $q_i = [-25°, 45°]$, $q_f = [-40°; 90°]$. Consider gravity effects. Do not consider damping in the joints.

Task 3: Control

1) Implement the PD controller with gravity compensation term. Given the following initial joint configuration $q_i = [-75°, 55°]$ control the robot to reach the final configuration $q_f = [100°; 80°]$.
2) Implement the Computed Torque control law using the Robotics System Toolbox blocks. Generate a quintic polynomial trajectory with the following initial and final configurations: $q_i = [20°, 65°]$, $q_f = [-15°; 100°]$. Also, the trajectory should pass by an intermediate joint configuration $q = [0°, 40°]$. Also consider gravity action.

RP Robot



Consider a planar RP robot like the one shown in figure (**second joint is a prismatic joint with motion axis along the x-axis of the link**) with the following set of parameters:

Mass of link 1: 2 Kg

Length of link 1: 50 cm

Mass of link 2: 6 Kg

Length of link 2: 25 cm

Task 1: Statics

1) Joint position q = [40°, -15 cm] . Implement gravity compensation using the manipulatorstatics.
2) Implement gravity compensation in the same configuration using the manipulator statics considering a spherical end-effector of mass Me = 5.2 Kg.
3) Implement gravity compensation in the same configuration using the manipulator statics considering a spherical end-effector of mass Me = 2 Kg and an additional mass of 4 Kg (modeled as a spherical object) placed on link 1 at 2/3 of its total length.

Task 2: Newton-Euler Algorithm

1) Implement the Newton-Euler algorithm for recursive inverse dynamics. Generate a cubic polynomial trajectory with the following initial and final configurations: $q_{\mathrm{i}} = [45°, 0\ cm]$, $q_{\mathrm{f}} = [25°; \text{-}15\ cm]$. Consider gravity effects. Do not consider damping in the joints.

Task 3: Control

1) Implement the PD controller with gravity compensation term. Given the following initial joint configuration $q_{\mathrm{i}} = [30°,\ 0\ cm]$, control the robot to reach the final configuration $q_{\mathrm{f}} = [0°; \text{-}20cm]$.
2) Implement the Computed Torque control law using the Robotics System Toolbox blocks. Generate a quintic polynomial trajectory with the following initial and final configurations: $q_{\mathrm{i}} = [30°, 0\ cm]$, $q_{\mathrm{f}} = [-40°; \text{-}25cm]$. Also, the trajectory should pass by an intermediate joint configuration $q\ = [0°, \text{-}10cm]$. Also consider gravity action.