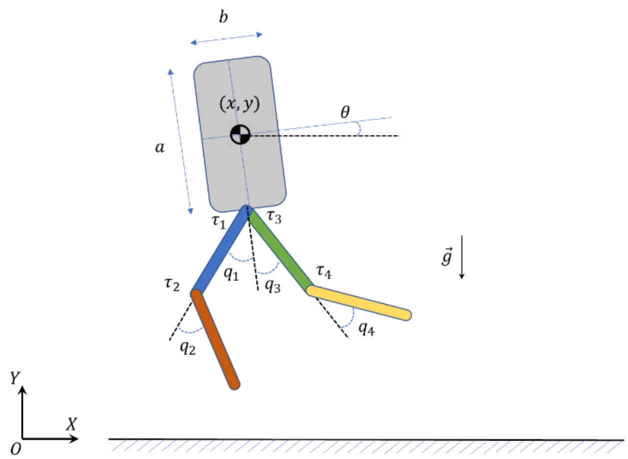
AME 556 – Robot Dynamics and Control – Final Project

Develop controllers for a 2D bipedal robot in MATLAB Simscape for the following tasks. The robot model is similar to HW3.

Instructions for the final report submission and tutorials about simulation settings are included at the end of this document.



Assume that the COM of each link is at the center of the link and the moment of inertia of the link around its COM is $I = \frac{1}{12}ml^2$.

Note that the positive direction of each angle is counterclockwise.

Given:

- Leg link: $m_i = 0.25\ kg, l_i = 0.22\ m$
- Body/trunk: $m_b = 8\ kg, a = 0.25\ m, b = 0.15\ m$
- Simulation settings:
  - $K_p^{ground} = 10^5, K_d^{ground} = 10^3$
  - Coefficient of static friction = 0.7
  - Coefficient of dynamic friction = 0.5

1. Task 1: Simulation and Physical Constraints

Please include the following physical constraints in the simulation. You should set error flags and terminate the simulation if any of the following constraints are violated:

- For hip joints $\{q_1; q_3\}$
  - Joint angle limit: $-120^o \leq q_{hip} \leq 30^o$
  - Joint velocity limit: $|\dot{q}_{hip}| \leq 30\ \left(\frac{rad}{s}\right)$
  - Joint torque limit: $|\tau_{hip}| \leq 30\ (Nm)$

- For knee joints $\{q_2; q_4\}$
  - Joint angle limit: $0^o \leq q_{knee} \leq 160^o$
  - Joint velocity limit: $|\dot{q}_{knee}| \leq 15\ \left(\frac{rad}{s}\right)$
  - Joint torque limit: $|\tau_{knee}| \leq 60\ (Nm)$
- Input saturation: if $|\tau_i| \leq \tau_{max}$, it means that you should apply the following modification before sending torques to the robot model:

$$\begin{cases} \tau_i = \tau_{max} & if \quad \tau_i \geq \tau_{max} \\ \tau_i = -\tau_{max} & if \quad \tau_i \leq -\tau_{max} \end{cases}$$

Please show that the simulation will be terminated if any of these constraints is violated. You may try to start the simulation from different initial conditions and try different joint motions of the robot to illustrate these cases.

You should also display the simulation time in the animation.

You will get $\boldsymbol{score = 20\ (points)}$ for completing this task.

2. Task 2: Standing and Walking

Design a controller that allows the robot to perform the following tasks:

- Standing up and down
  - Start the simulation in an initial condition of your choice.
  - Control the robot the stand at the desired state of $y_d = 0.45\ m, \theta_d = 0\ rad$ for 1 second.
  - Keep the body orientation at $\theta_d = 0\ (rad)$. Move the body up and down with the following trajectory:
    - $y_d = 0.45 \rightarrow 0.55\ (m)$ in 0.5 second
    - then $y_d = 0.55 \rightarrow 0.4\ (m)$ in 1 second
- Walking forward with a speed of $\geq 0.5\ m/s$ in at least 5 seconds.
- Walking backward with a speed of $\geq 0.5\ m/s$ in at least 5 seconds.

You will get a $\boldsymbol{score = 20\ (points)}$ for completing this task.

3. Task 2: Running on flat ground

- Control the robot to run on flat ground for **10 meters**.
- The score for this task will be calculated as follows:

$$\boldsymbol{score = \frac{200}{travel\_time\ (s)}}$$

where $\boldsymbol{travel\_time}$ is the time in seconds needed for the robot to complete the 10-m distance run. The distance can be verified by the distance along the x-axis of the robot's COM.

- For running, the requirement is to have a flight phase in the gait schedule.

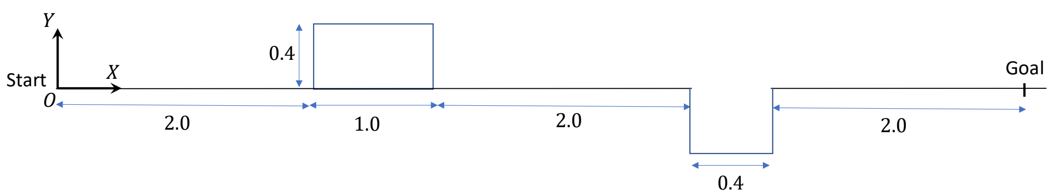
4. **Task 3: Stair Climbing**

- Control the robot to walk/run on 5 consecutive stairs with the stair rise of 10 cm, stair run of 20 cm.
- Please make sure that there is no collision between the robot body, robot legs, and the stair edges.
- The score for this task will be calculated as follows:

$$\boldsymbol{score} = \frac{\mathbf{20}}{\boldsymbol{travel\_time}\ (\boldsymbol{s})}$$

where $\boldsymbol{travel\_time}$ is the time in seconds needed for the robot to complete the task. The time is counted when the robot starts walking when all robot feet are in front of the first stair and ends when all robot feet pass the edge of the last stair.

5. **Task 4: Obstacle course**



- Control the robot to overcome the obstacle course, as shown in the figure above.
- Please make sure that there is no collision between the robot body, robot legs, and the obstacles and terrain.
- The score for this task will be calculated as follows:

$$\boldsymbol{score} = \frac{\mathbf{200}}{\boldsymbol{travel\ time}\ (\boldsymbol{s})}$$

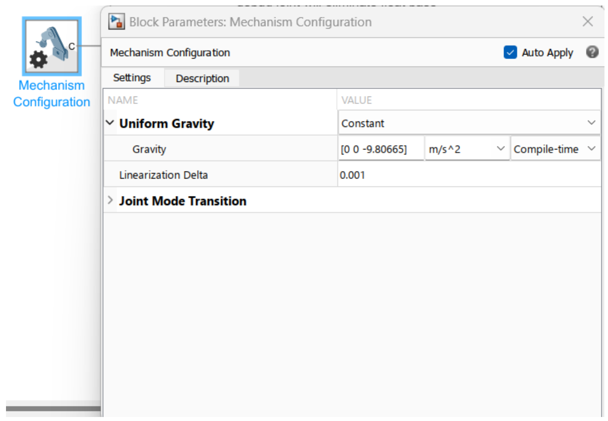
Final Report Format and Submission Guidelines

- For the final project report, please submit a single pdf file and include the following information at the start of the report:
  - group number + team members (with USC email and ID)
  - a link to your simulation video
  - a link to your code
- The final report format will be similar to our HWs. Please include the following contents in the report:
  - Explain briefly your approach and your code structure so that it can help us to grade it.
  - Include necessary plots, links to the videos and code, and details about the calculation of the score for each task.
  - Please verify via plots that your results satisfied all the physical constraints of the robot.
- Format: single column, font size 12pt, page limit 10.
- Submission:
  - Submit 1 single pdf for your team's report.
  - Please remember to include viewable links to simulation videos and codes in the report. Please remember to change the sharable setting so that everybody with the link can view it.
  - Each team will send scores and the final report via email to the TA and cc me in the email.

Tutorial on Simulink Simscape Simulation Settings

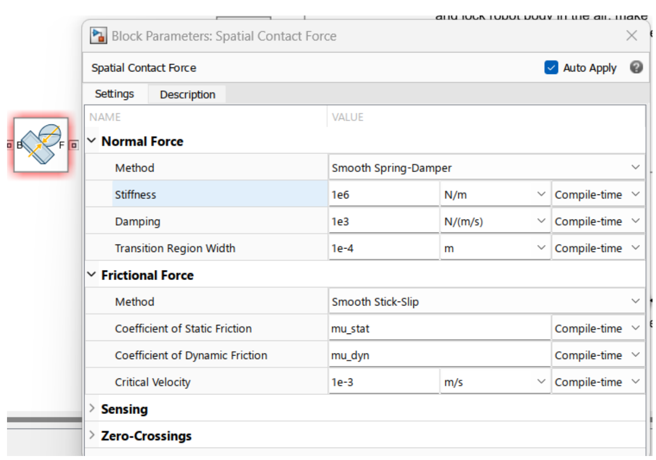
1. Make sure to install all necessary packages! (Simscape and Simscape Multibody are required)

2. 2-D Gravity Configuration
    a. In the mechanism configuration module, please identify the gravity direction (i.e., 2D will be in the y direction)

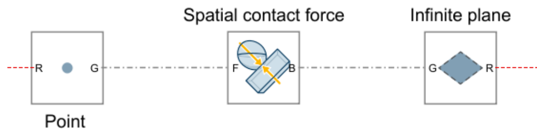


    b. Note that your simulation is still a 3-D environment; you are building a 2-D simulation with only 2-D properties. When filling out properties such as Moment of Inertia, please only fill out the property for the planar rotation axis and leave the rest at zero.

3. Contact Modeling Setting
    a. Use spatial contact force module for contact modeling.
    b. Apply default normal force settings.
    c. Apply dynamic and static friction coefficients according to your final project instructions.

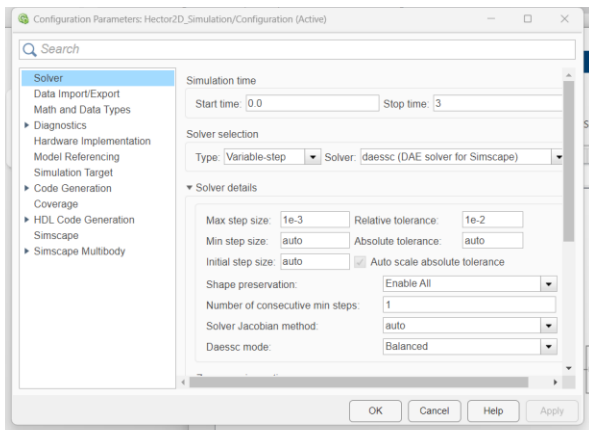


d. It is suggested to use the Point module and the Infinite Plane module to simulate the contact. Any geometry contact will greatly reduce simulation efficiency (e.g. block solid and spherical solid).



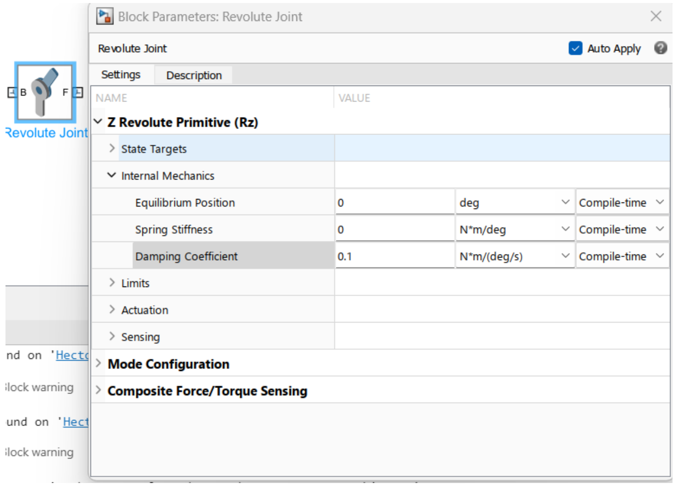
4. Solver setup
    a. Use variable-step size and DAE solver with the maximum stepsize of 1e-3.

    Modeling -> Model Settings -> Solver or Click on the solver at the bottom right corner.



5. Uneven terrain
    a. It is suggested to use a grid surface to simulate uneven terrain, please refer to the documentation https://www.mathworks.com/help/sm/ref/gridsurface.html

6. To better match the robot hardware and avoid simulation instability, you can apply certain joint damping. You can set up the joint damping in planar joint modules. The recommended value is 0.1.



7. Simscape PS signals and Simulink native signals are not compatible, please use PS-simulink/Simulink-PS converter modules to connect signals.