Assignment 01: TSID for bipedal applications

Elisa Alboni* - elisa.alboni@unitn.it

October 2024

1 Description

This assignment focuses on Task Space Inverse Dynamics control (TSID). The goals of this assignment are:

- understanding the role of gains and weights in TSID;
- using TSID for controlling a walking humanoid robot, analysing the effect of the required tasks.

2 Submission procedure

You are encouraged to work on the assignments in groups of 2 people. **If you have a good reason to work alone, then you can do it, but this has to be previously validated by one of the instructors**. Groups of more than 2 people are not allowed. The mark of each assignment contributes to 10% of you final mark for the class (i.e. 3 points out of 30).

When you are done with the assignment, please submit a single compressed file (e.g., zip). **The file name should contain the ID number of the group members**, and it must contain:

- A pdf file with the answers to the questions, the **names and ID number** of the group members; you are encouraged to include plots and/or numerical values obtained through simulations to support your answers. **This pdf does not need to be long. One or two pages of text should be enough to answer the questions. You can then add other pages for plots and tables.**
- The complete orc folder containing all the python code that you have developed.

If you are working in a group (i.e., 2 people) only one of you has to submit.

Submitting the pdf file without the code is not allowed and would result in zero points. Your code should be consistent with your answers (i.e. it should be possible to produce the results that motivated your answers using the code that you submitted). If your code does not even run, then your mark will be zero, so make sure to submit a correct code.

*Optimization and Learning for Robot Control, Industrial Engineering Department, University of Trento.

3 Tests

In this assignment, TSID, an optimization-based framework for control of mechanical systems, is used for bipedal applications. First, TSID is employed to track a desired center of mass trajectory of the robot, and then it is used to make a biped perform walking motions.

The template code for the assignment is located in our shared folder:

`lab/A1_template`

This folder already contains all the code for solving the assignment.

3.1 Weights and Gains in TSID

In the first part of the assignment, TSID is used to control the center of mass (CoM) of Talos, a biped robot, to track a sinusoidal trajectory along the lateral direction.

Try to answer the following questions:

1. After completing the code in `ex_1_biped.py` so that the CoM follows a sinusoidal trajectory along the lateral direction (setting the following parameters: $amp = [0, 0.05, 0]$, $f = [0, 0.5, 0]$), report and discuss the obtained trajectories.

2. Modify the parameters (weights and gains) in `talos_conf.py` to improve the trajectory tracking. In particular, try setting $w_{CoM} = 10$ (keeping $kp_{CoM} = 10$) and $kp_{CoM} = 100$ (keeping $w_{CoM} = 1$). Report and discuss the results obtained. Which solution would you adopt if using a real robot? In which cases would you use the other solution?

3. Using the parameters found in the previous question (2.), increase the frequency of the sinusoidal trajectory along the longitudinal direction to $f = [0, 1, 0]$. How does the tracking of the desired trajectory change? Why? Compare the results with those from the previous case.

3.2 Importance of tasks for biped walking

In the second part of the assignment, TSID is used to control the biped Romeo to walk, given the reference trajectories of the center of mass, and of the feet of the robot. The trajectories are obtained to make the robot perform 6 steps in the longitudinal direction. In the folder there are two sets of trajectories, `romeo_walking_traj015.npz` and `romeo_walking_traj030.npz`, obtained setting the steps' length to 0.15 m and 0.3 m respectively.

Try to answer the following questions:

1. Try to track the first set of trajectories `romeo_walking_traj015.npz` (in `romeo_conf.py`, set `DATA_FILE_TSID = "romeo_walking_traj015.npz"`) by running `ex_2_biped_walking.py`. Analyse the effect of the different tasks, (i.e., center of mass task, angular momentum task, foot motion task, foot contact task, and joint posture task) by changing their weights and observing the effects on the resulting motion. Tune $w_{posture}$ to make the robot succeed. Report and discuss the results.

2. Try now to track the second set of trajectories `romeo_walking_traj030.npz` (in `romeo_conf.py`, set `DATA_FILE_TSID = "romeo_walking_traj030.npz"`). What weights do you need to change to make the robot succeed? Report and discuss your ideas and the results obtained.