LAB 5: Floating base robot: quasi-static control of stability

Notes:

- You can check the frequency of the controller inspecting the topic **/hyq/command** with **rostopic hz /hyq/command**.
- It is possible to set the verbosity of the output, by setting the parameter *verbose* in ex_5_conf.py.

1) *Sinusoidal reference:* Set a sinusoidal reference for Z direction (f_z = 0.5 Hz, A_z = 0.05 rad) together with another for the pith direction (f_θ = 1 Hz, A_θ = 0.1 rad)

2) *Projection-based controller:* Design a controller for base frame position and orientation with a quasi-static approach.

2.1)Virtual Impedance: Design a virtual impedance to track references for the base frame position and orientation. Compute the desired wrench $W^d = W_{fbk} \in \mathbb{R}^6$ to realize the virtual impedance:

$$W^d_{lin} = K_{lin}(x^d_b - x_b) + D_{lin}(\dot{x_b}^d - \dot{x}) \tag{1}$$

$$W^d_{ang} = -K_{ang}e_o + D_{ang}(\dot{\omega_b}^d - \omega) \tag{2}$$

where $e_o \in \mathbb{R}^3$ is the orientation error.

2.2) Mapping to torques: Map the desired wrench into torques with a projection-based approach:

$$\tau^d = J^T_{cj}(J^T_b)^\dagger W^d \tag{3}$$

where:

$$J^T_b = \begin{bmatrix} I_{3\times3} & \dots & I_{3\times3} \\ [x_{f_1} - x_b]_\times & \dots & [x_{f_c} - x_b]_\times \end{bmatrix} \tag{4}$$

Note that $W = J^T_b f$ represents the contact wrench that is given as input to the Newton-Euler equations, but we are doing an approximation here: we are controlling the orientation of the base link rather than an equivalent rigid body whose angular velocity $\bar{\omega}$ is the *average* angular velocity of all the links of the robot (as Newton-Euler equations state). In essence we are approximating $\bar{\omega}$ with ω_b. Check the robot is tracking the sinusoidal reference trajectory of point 1). How can this be improved? Play with the frequency of the controller (e.g. check parameter dt) and see how the system can tolerate (without being unstable) higher damping values for lower frequencies.

3) *Control of CoM:* Modify the controller to control the position of the CoM in place of the base frame. Which changes are necessary?

$$J^T_{com} = \begin{bmatrix} I_{3\times3} & \dots & I_{3\times3} \\ [x_{f_1} - x_{com}]_\times & \dots & [x_{f_c} - x_{com}]_\times \end{bmatrix} \tag{5}$$

4) *Gravity compensation:* Using the controller for the CoM position, add a gravity term W_g to the desired wrench.

$$W^d = W_{fbk} + W_g \tag{6}$$

How does W_g look like? (hint: remember that gravity is applied to CoM) How is the tracking error in steady state? and in motion?

5) *Feed-forward term:* Add a feed-forward term the desired wrench.

$$W^d = W_{fbk} + W_g + W_{ffwd} \tag{7}$$

How would you implement that at the impedance level? (hint: exploit the mass matrix and the tensor of inertia). How is the tracking of the CoM and of the trunk orientation improved during motion? Compare also the tracking of ground reaction forces with and without feed-forward term.

6) *Check static stability:*

Design an horizontal trajectory for the CoM (e.g. moving along Y direction) starting from the default configuration q_0. What happens when the CoM goes out of the polygon? Plot the ground reaction forces and check they are going to zero in the RF and RH leg and the whole weight is only on two legs.

8) *quasi-static QP controller:* Re-implement the previous mapping from wrench to torques by casting as an optimization problem (QP) that optimizes for ground reaction forces $f^d \in \mathbb{R}^k$. Enforce unilateral constraints for the leg that are in stance $f_{z,min} = 0$.

$$\begin{aligned} f^d = &\underset{f \in \mathbb{R}^k}{\operatorname{argmin}} \|W - W^d\| \\ &s.t. \quad \underline{d} < Cf < \bar{d}, \end{aligned} \tag{8}$$

where in the cost term we are trying to track the desired wrench W^d. This can be expanded into:

$$\begin{aligned} f^d = &\underset{f \in \mathbb{R}^k}{\operatorname{argmin}} (Af - b)^T S (Af - b) \\ &s.t. \quad \underline{d} < Cf < \bar{d}, \end{aligned} \tag{9}$$

where:

$$C = \begin{bmatrix} C_0 & \dots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \dots & C_c \end{bmatrix} \quad \underline{d} = \begin{bmatrix} \underline{d}_0 \\ \vdots \\ \underline{d}_c \end{bmatrix} \quad \bar{d} = \begin{bmatrix} \bar{d}_0 \\ \vdots \\ \bar{d}_c \end{bmatrix}, \tag{10}$$

with: $A = J_b^T$ and $b = W^d$, $C_i = n_i^T$, $\underline{d}_i = f_{min_i}$, $\bar{d}_i = f_{max_i}$. The $S \in \mathbb{R}^6$ is a selection matrix that allows you to promote the tracking in certain directions rather that others when the constraints are hit.

Is it something missing in this optimization? There is not unique solution and this is due to the fact it exists a null-space for A, that makes the cost positive semi-definite. What is its dimension? Try to add a regularization term with adjustable gain α to make the Hessian of the cost positive definite:

$$\begin{aligned} f^d = & \underset{f \in \mathbb{R}^k}{\text{argmin}} (Af - b)^T S (Af - b) + \alpha f^T W f \\ & s.t. \quad \underline{d} < Cf < \bar{d}, \end{aligned} \tag{11}$$

a) Compare with the previous projection-based controller of point 6). b) Does it tolerates higher frequencies on the reference trajectories? c) Check the Z component of the ground reaction forces is always positive. Does it makes sense to set f_{min} exactly at zero? or it is safer to allow a bit of positive margin? why? Try to play with the f_{min} increasing it for just one leg and verity that the constraint on the Z component is not violated for that leg.

9) *Some experiences with the quasi-static QP controller:*

a) Try to set a static target for the com inside the triangle formed by the LF, RF and LH leg. b) After 1.5 s remove the RH leg from the set of stance legs. How the load is redistributed on the other three legs? Check the contact force goes to zero for that leg.

10) *Friction cones:* Did you notice some slippage at the feet while following the sinusoidal reference? Modify the C_i matrix to add the friction cone constraints for each stance leg, with friction coefficient μ_i:

$$C_i = \begin{bmatrix} (t_{x_i} - \mu_i n_i)^T \\ (-t_{x_i} - \mu_i n_i)^T \\ (t_{y_i} - \mu_i n_i)^T \\ (-t_{y_i} - \mu_i n_i)^T \end{bmatrix} \quad \bar{d}_i = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}, \tag{12}$$

Note that the unilaterality constraints are naturally incorporated in the friction cones and are no longer needed. Run again a simulation setting a conservative friction coefficient of 0.7 (the Gazebo simulator use 0.8 by default). a) Is the slippage still happening? has the tracking error increased? b)Plot the constraint violations to check is there is any. Try to reduce the friction coefficient and observe the number constraint violations increases. c) Try to tilt the cones setting the normals at the feet pointing 45 degs inward. See the robustness becomes low because the contact forces are lying on the boundary of the cone.

11) *Friction cones and robustness:* Modify the regularization term. Exploit the force redundancy to regularize the contact forces to stay in the middle of the cone. Set the regularization matrix to:

$$W = \begin{bmatrix} {}_wR_1 W_{n_0} {}_wR_1^T & \dots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \dots & {}_wR_c W_{n_c} {}_wR_c^T \end{bmatrix} \tag{13}$$

where ${}_wR_i = \begin{bmatrix} t_{x_i} & t_{y_i} & n_i \end{bmatrix}$ is the rotation matrix that maps vector from the contact frame of leg i to the world frame. Set $W_{n,xy}$ higher to penalize tangential forces in the contact frame. What is the size of the internal force redundancy with 4 legs on the ground? Now modify the regularization term to minimize joint torques (e.g. $\tau^T\tau$):

$$W = J_c W_\tau J_c^T \tag{14}$$

with $W_\tau \in \mathbb{R}^n$ is the weighting matrix for the torque vector.