Assignment 01: Comparison between different controllers

Alessio Di Casola (mat. 239490), Marco Sterlini (mat. 236865)

1 Answers

1.1 Question 1

The different implementations of the IC controller differ for some characteristics. We imposed stiffness and damping matrices as diagonal with constant stiffness values and damping values dynamically computed using the simulator mass matrix imposing a unitary damping ratio.

Every controller manages to converge to the desired end effector position except for the third one (Exact version). The first one (Simplified version) manifests a continuous movements of the arm even though the end effector remains still, this because we didn't minimize the null-space of the jacobian pseudo-inverse yet.

In the second (Simplified version + postural task) and fourth (Exact version + postural task) controller we don't notice any macroscopic difference judging by the simulations.

When we introduced the friction we noticed that the controller couldn't manage to always make the robot reach the desired e.e. position. This can be physically explained because the control torque is proportional (but not only) to the error position, once it becomes small enough the resulting control torque becomes equal and opposite to the sum of the friction torque and gravity compensation resulting in a torque equilibrium (fig. **??**) without ever reaching the desired position (fig. **??**).

Here are the average tracking errors of the controllers:

Simplified	Simplified + postural task	Exact	Exact + postural task
		Undefined	

1.2 Question 2

In the exact controller we see that the torques diverge and hence we can't finish the simulation, in the friction case scenario instead it manages to converge. This is explained by the behavior of the non controlled joint space: in the first case it diverges with increasing velocities and the $\dot{J}\dot{q}$ term diverges leading to the end of simulation, in the second case the uncontrolled space is mechanically stable due to the addition of friction. Hence we manage to reach an overall converging control law only by controlling the main task (fig. **??**).

The other controls manage to stabilize the desired position because they are imbued with the postural task, the first case instead, even though it doesn't have the postural task, doesn't diverge because we made the assumption that joint velocities won't be high, doing so we don't multiply the term $\dot{J}\dot{q}$ by J^T and we don't encounter the numerical divergence (fig. **??**) Non linear effects are portrayed by h which is simply added to the control law remaining reasonably bounded.

1.3 Question 3

From the observation of the e.e. position plot of OSC and IC controllers in the trajectory tracking task (fig **??**) we concluded that the IC controller performed overall better, both at initial and augmented frequency. As a downside the IC controller showed higher torque peaks respect to the OSC controller at the initial stages of the simulation and this may be an issue in a real implementation (fig. **??**).

In general both controllers worsened their performance with the higher frequency value, in particular we observed a change of general joint configurations with both approaches after around 2 seconds of simulation leading to a higher average tracking error. After this change the OSC controller didn't manage to reach again the desired value in the third coordinate of e.e. position stabilizing itself around an offset value, on the other hand the IC controller stabilized itself with variance lower than the millimeter.

Here are the average tracking errors of the controllers:

IC 1^{st} frequency	IC 2^{nd} frequency	OSC 1^{st} frequency	OSC 2^{nd} frequency

1.4 Question 4

In order to improve the controller performances we thought to tune the feedback parameters for the OSC controller and augment the error multiplied to the stiffness matrix as concerns the IC. Doing so we actually saw some improvements in the tracking error and overall performance. However, while tuning, we noted that exaggerating with these fixes could lead to numerical divergence that is a big problem in a real implementation. For this reason we choose values slightly lower than the empirical upper limit

1.5 Question 5

Following the considerations about the conservative approach we adopted in the tuning process we saw how, enabling the "*randomize_robot_model*", we obtained decent results. The average tracking errors increased as expected but even after several runs of the script we never found a case in which the controllers didn't work thus we concluded the tuning led to two sufficiently robust controllers.

2 Plots

Notes on comparison plot:

On the left we can see the values of h, μ and $\dot{J}\dot{q}$ for some step of the simplified controller, on the right the equivalent for the exact controller moments before divergence. We can see how with increasing velocities the terms tend to diverge in magnitude