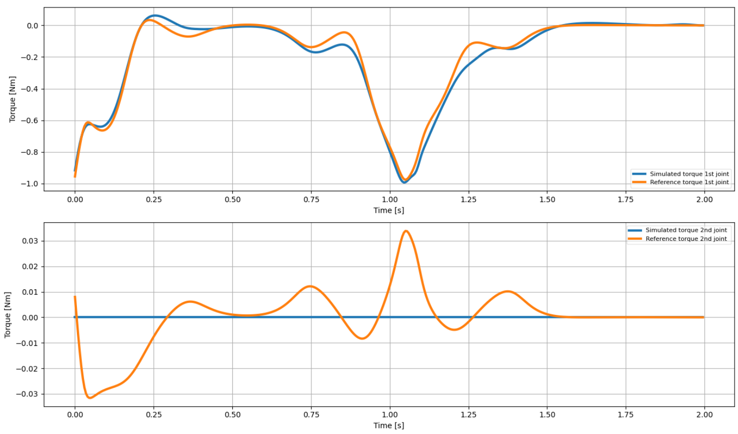
Assignment 02: Comparison between different methods for taking account of underactuation in DDP

Jianming Han [230401], Basem Shaker [230637]

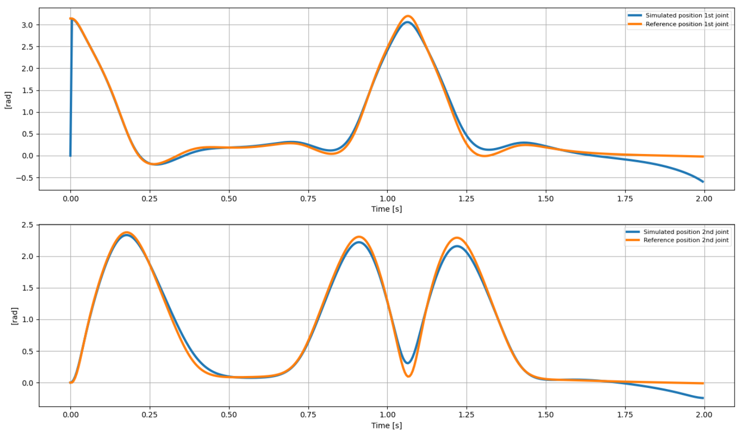
November 29, 2021

Question 1

The reference and simulated torque and angle for the additional penalty method are shown in Figure 1. The cost was 741.68 and cost sim is 737.53. It took 23 iteration for the state reference trajectory to reach the final desired state successfully. However, the system was not able to simulate the Pendubot successfully performing the swing-up maneuver.



(a) Torque at the joints

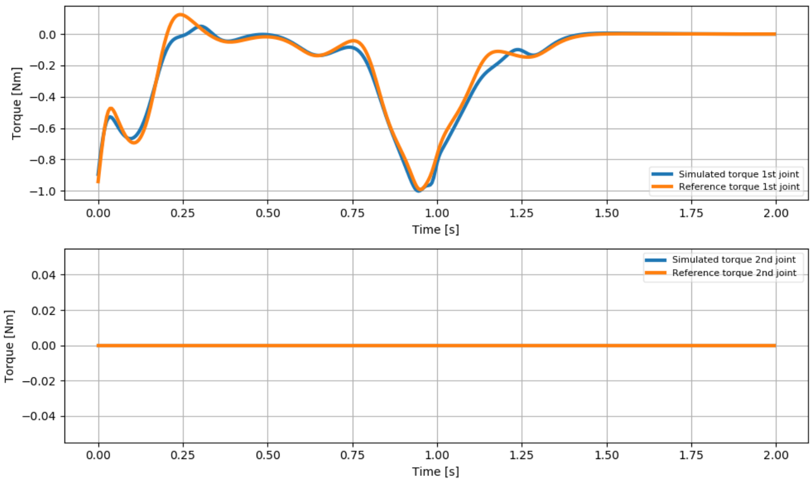


(b) Position of the joints

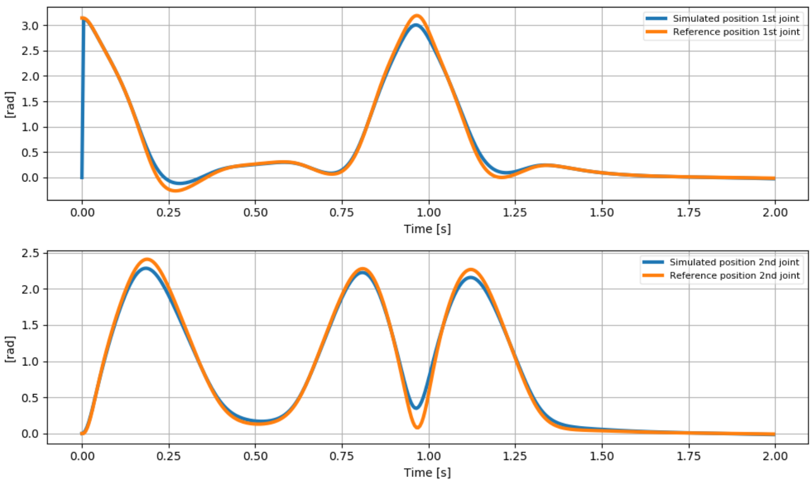
Figure 1: Results of additional penalty method for under-actuation

Question 2

The reference and simulated torque and angle for the selection matrix method are shown in Figure 2. The cost was 749.32 and cost sim was 736.52. It took 32 iterations for the state reference trajectory to reach the final desired state successfully. In simulation, the controller was able to make the Pendubot successfully perform the swing-up maneuver.



(a) Torque of joints



(b) Position of joints

Figure 2: Results of Selection Matrix method for under-actuation.

Question 3

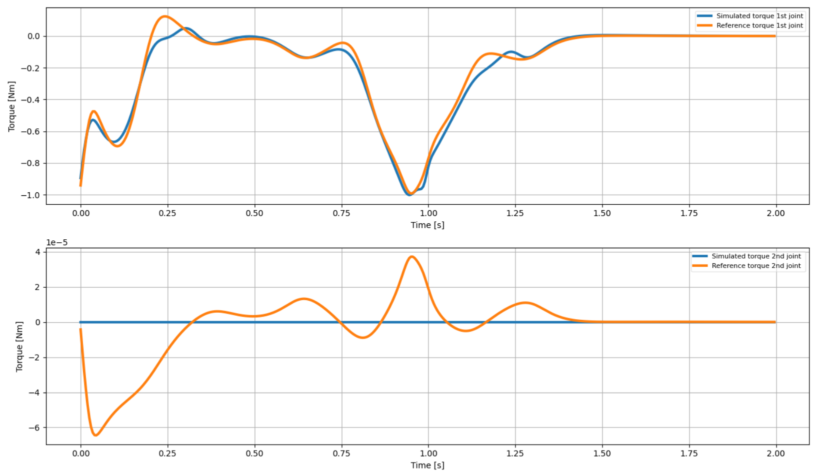
As we can see from Figure 1b and 2b, the reference motion trajectories of the two methods both reach the target successfully. The overall features of the motion between the two methods are quite similar. However, it differs in the time when these features occur. The features appear to be delayed in the under-actuation penalty method. In terms of the simulated motion, the same can be observed in terms of features and their occurrence time. The simulated trajectory was able to follow the reference trajectory [except when there was a sudden change in angle]. The selection matrix method succeeded to follow the reference trajectory at the end, while the additional penalty method failed to follow the reference trajectory at the end.

If we look at the torques applied to the 2nd joint in Figure 1a, we can see that the selection matrix method doesn't request any torque to be applied there, which matches what the simulation does as well. For the additional penalty method, it tries to apply a non-negligible torque to the 2nd joint for its reference as shown in Figure 2a, even though it is not possible to apply torque there. This in return makes the system unstable, because the simulation doesn't apply any torque in the 2nd joint at all.

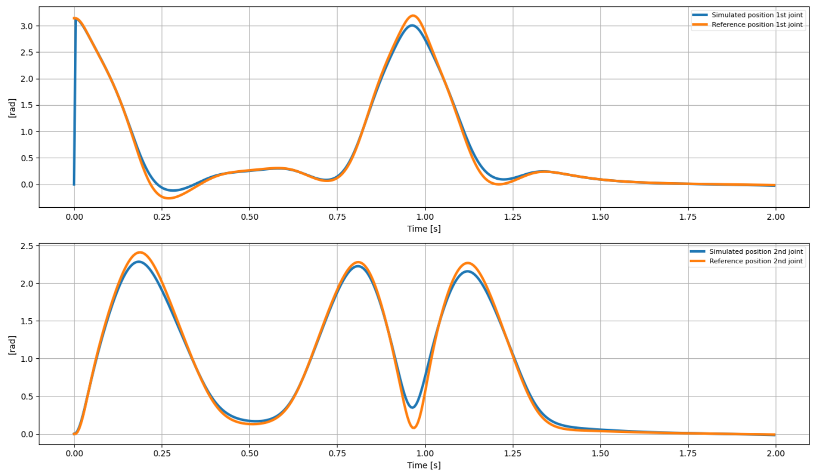
In terms of reference motion, the difference in cost between the two methods is only 7.64, with the selection matrix method being higher. In terms of simulated motion cost, the difference is 1.01, with the additional penalty method being higher.

Question 4

The results of increasing the under-actuation penalty from 1e2 to 1e5 are shown in Figure 3 below. It took 32 iterations instead of 23 for the state reference trajectory to reach the final desired state. This however made the simulation able to successfully perform the swing-up maneuver.



(a) Torque at the joints



(b) Position of the joints

Figure 3: Results of increasing the under-actuation penalty to 1e5 in additional penalty method

When the under-actuation penalty increased, it decreased the reference torque applied to the 2nd joint to very small negligible values. As discussed in Question 3, the reference torque and simulated torque are now quite much similar [almost zero] making it possible for the simulation to succeed.

In terms of reference trajectory, the under-actuation penalty of 1e2 was able to successfully calculate it in less iterations compared to the 1e5. It seems that having the capability of applying torque, albeit small, to a second joint made it easier to reach target and stabilize the system compared to only applying torque to one joint.

Question 5

If we had to use such controllers in a real robot, we would prefer the additional penalty method. DDP assumes a deterministic system, while a real robot is a stochastic system because there are a lot more variables that are not as controlled as a simulated system. This means the torque on the second joint couldn't be perfectly zero. For the selection matrix method, it requires the torque of joints that are not actuated to be exact zero, which is not possible in a real robot. For the additional penalty method, it can produce more adaptive torque [which is small but not exact zero] to the joints that are not actuated by applying penalty to them. We can fine-tune the value of penalty to make the robot work better. Comparing it to the selection matrix, which is not as adaptive as the additional penalty method and won't be as easy to tune to make it work properly.