Tutorial 6 & 7

Comparison of different controllers:

(1) free PD
(2) free PD+G
(3) Model-based PD-like in joint space
(4) Model-based PID-like in joint space
(5) Model-based PD-like in operational space
   - With & without safety mechanism
(6) Model-based PID-like in operational space
   - With & without safety mechanism
(7) Adaptive PD in joint & operational space and PID in operational space

In the following report, I am going to discuss the above controllers with respect to several measures along a time axis until 30 seconds

- Accumulated error: addition of all three dimensions
- Error for each dimension
- Operational space coordinates X
- Operational space velocities Xp
- Joint space coordinates Q
- Joint space velocities Qp
- Passivity based stability with V and Vp

The errors are controller-dependent in joint or operational space

Applied original gains (same for all controllers):

Kd = [3,15,10]

Kp = [2,15,10]

Ki = [0;0;1]

Code usage:

Define the desired "mode" in the TrajGen.m

Mode	Description
11	Free PD (task 1)
12	Free PD+G (task 1)
21	Break
22	Model-based PD-like (task 1)
23	Model-based PID-like (task 1)
31	Model-based PD-like (task 2)
32	Model-based PID (task 2)
41	Model-based PD operational (task 2)
42	Model-based PID operational (task 2)
51	Model-based PD operational (task 2) -> triggering safety mechanism
52	Model-based PID operational (task 2) -> triggering safety mechanism
8	Adaptive Control (task 2) PD in joint space, PID in joint space, PID in operational space

Task 1: Free PD & PD+G

Description

- First six seconds: Move the robot to desired position
- Next six seconds: Hold the position (Qdp = 0)
- Next 12 seconds: Constantly move along a trajectory with a frequency w1
- Last six seconds: Change to a shorter frequency w2

(1) <u>Free PD controller</u>

The free PD controller is not capable of accomplishing the task with these gains, as the errors are large (see Figure 2, first row)

By increasing the second and third parameter of the Kp gain to [2,50,35], the free PD controller can achieve much better results.

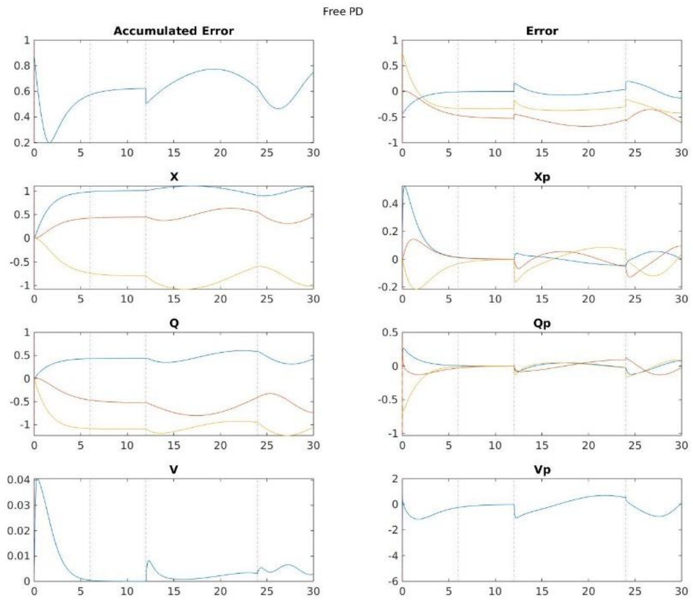

Figure 1: Analysis of free PD controller with original gains

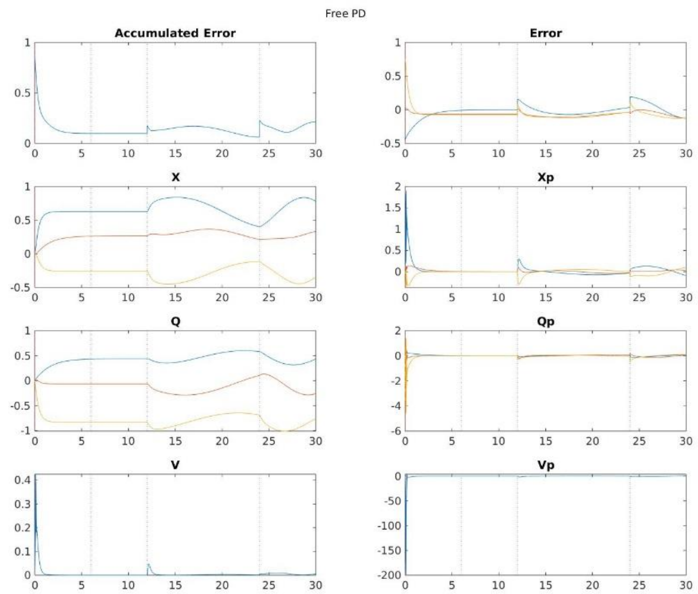

Figure 2: Error-reduction of the free PD controller by increasing gains

(2) <u>Free PD controller with gravity compensation</u>

This controller performs much better in terms that it doesn't need higher gains to accomplish the task.

Both controllers handle abrupt changes well, e.g. keep a decent stability (V jumps only 0.01 after input change).

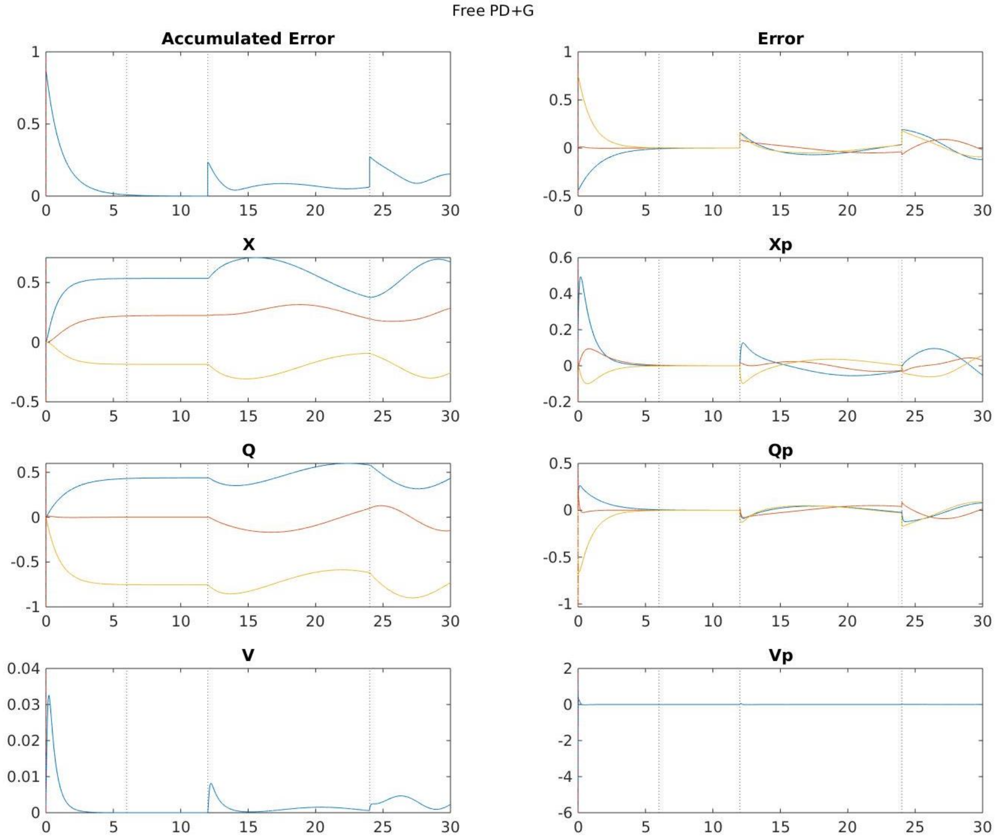

Figure 3 Analysis of the free PD+G controller with original gains

Task 2: Model-based PD & PID, Adaptive Controller

Description

- First six seconds: Move the robot to non-singular position (Q = [0;0;pi/2)]
- Next six seconds: Move the robot to the desired position
- Next three seconds: Hold the position (Qdp = 0)
- Next 12 seconds: Constantly move along a trajectory with a frequency w1
- Last six seconds: Change to a shorter frequency w2

(3) <u>Model-based PD-like controller in joint space</u>

This controller reacts much faster than the free controllers. However, the error for the position change is quite large with up to 0.2. Instable peaks in the V and Vp function are really short.

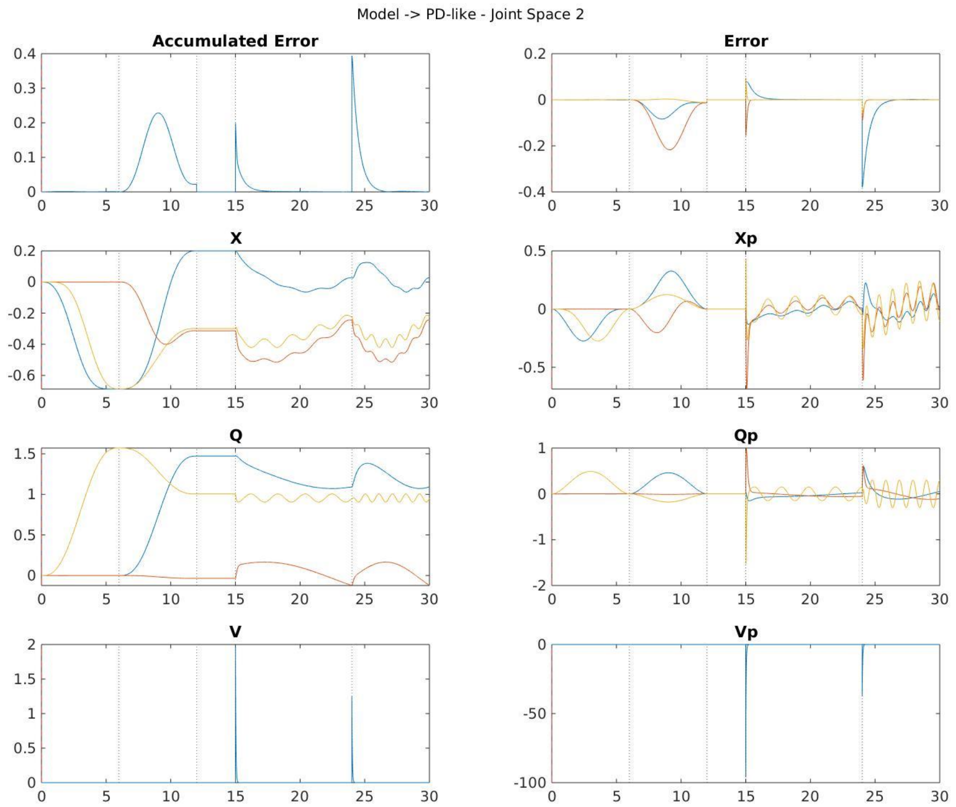

Figure 4 Model-based PD-like controller in joint space

(4) Model-based PID-like controller in joint space

For my test PD-like and PID like controller performed not notably different. Even when changing the Ki gains from [0;0;1] to [10;10;10].

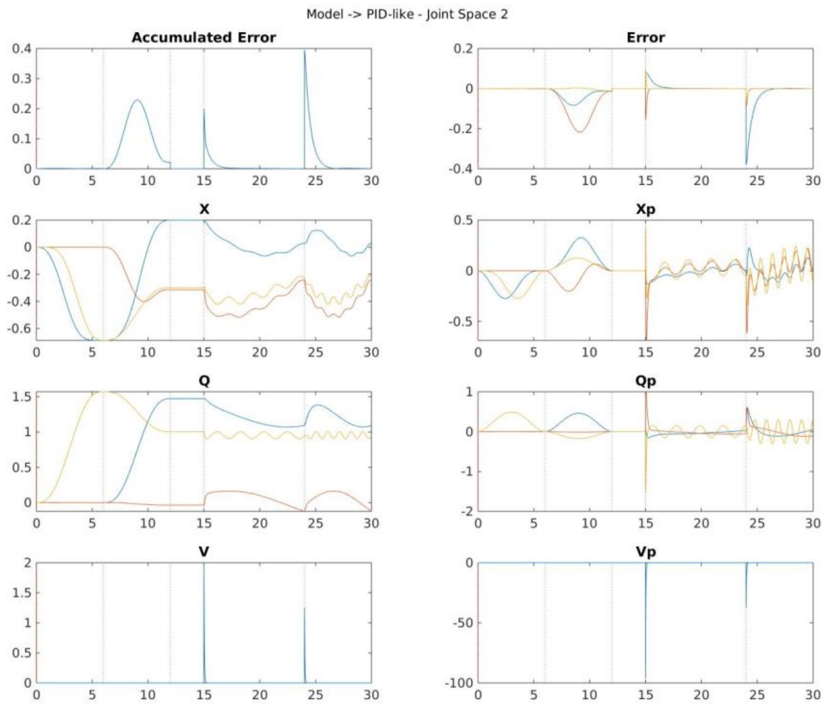

Figure 5 Model-based PID controller in joint space

(5) Model-based PD-like controller in operational space

The best performance achieves the PD & PID controllers in operational space with an accumulated error below 0.1 (Figure 6,Figure 8). Especially the movement to a desired position returns a lower error than those controllers in joint space.

Here, also the manipulability is shown. When the threshold of 0.05 is reached the safety-mechanism is triggered and the robot returns to the home position. (see red-dashed line in Figure 7,Figure 8)

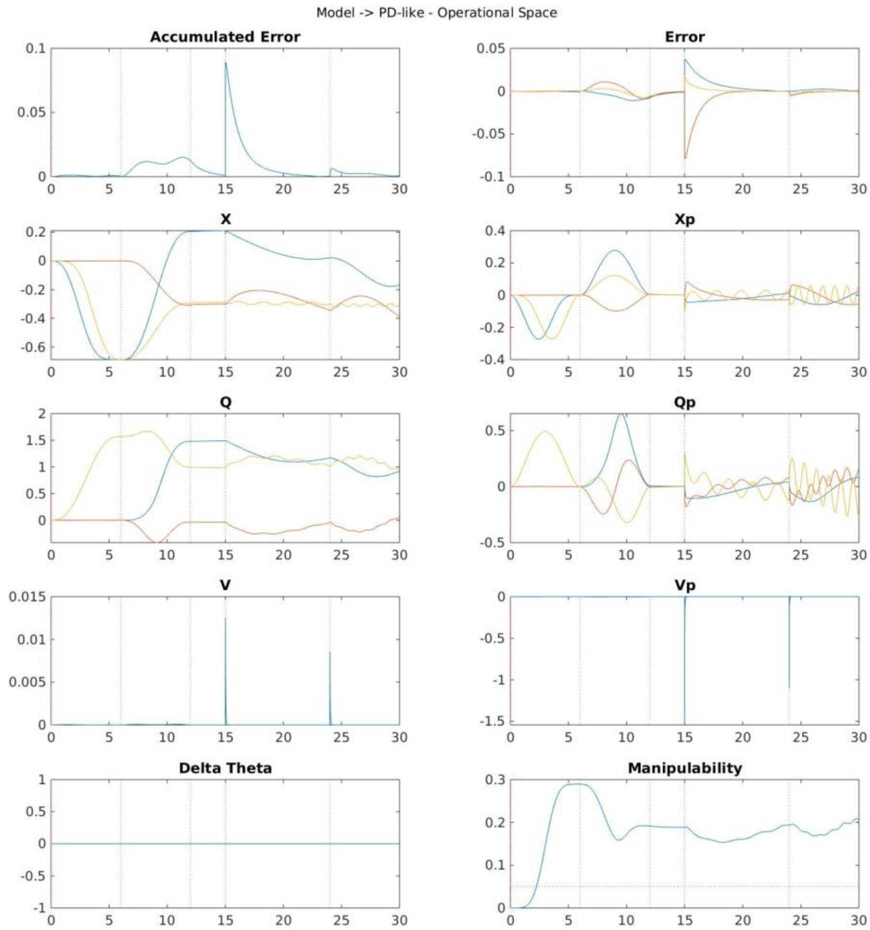

Figure 6 Model-based PD-like controller in operational space

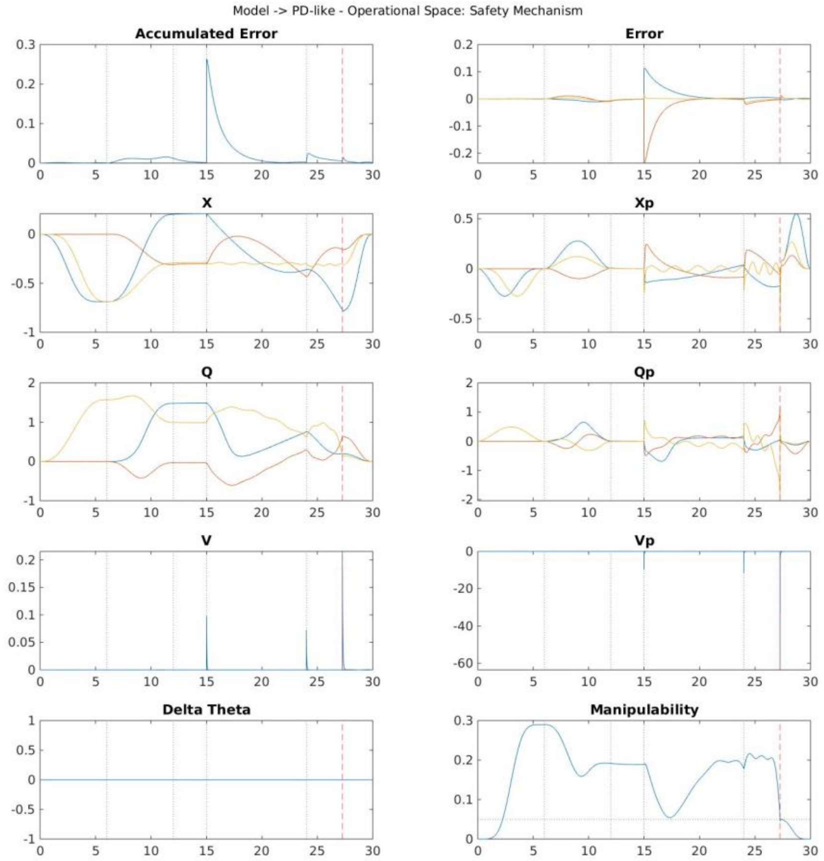

Figure 7 Model-based PD-controller in operational space: safety mechanism triggered

(6) Model-based PID-like controller in operational space

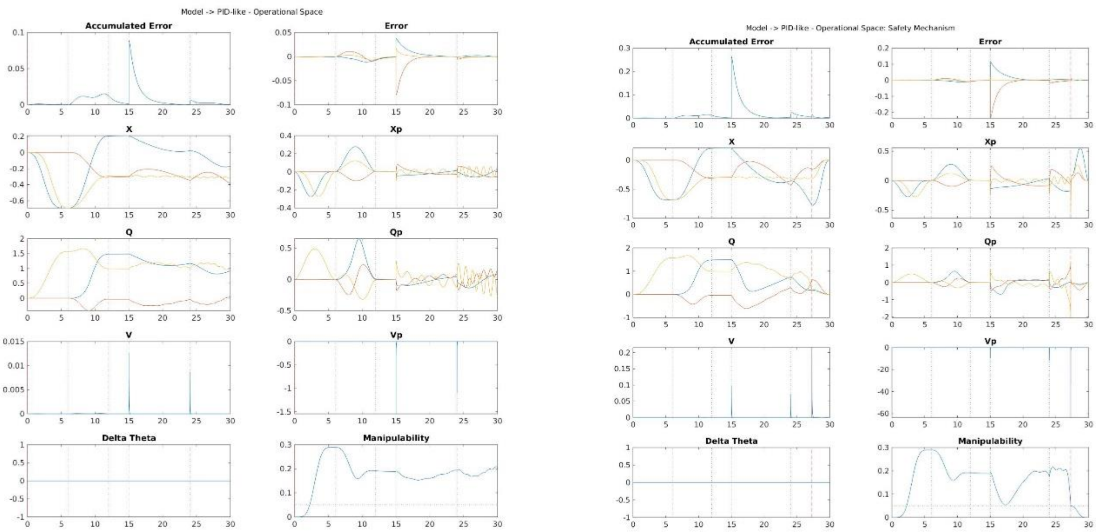

Figure 8 Model-based PID-like controller with and without triggered safety mechanism

(7) Adaptive Controller

The performance is similar to the PD-&PID-controllers. Even though, delta theta is still very large after 30 seconds (Figure 9). Most interestingly, using an identity matrix for the gamma results in a lower delta gamma, but higher error scores and much more instable (Figure 10).

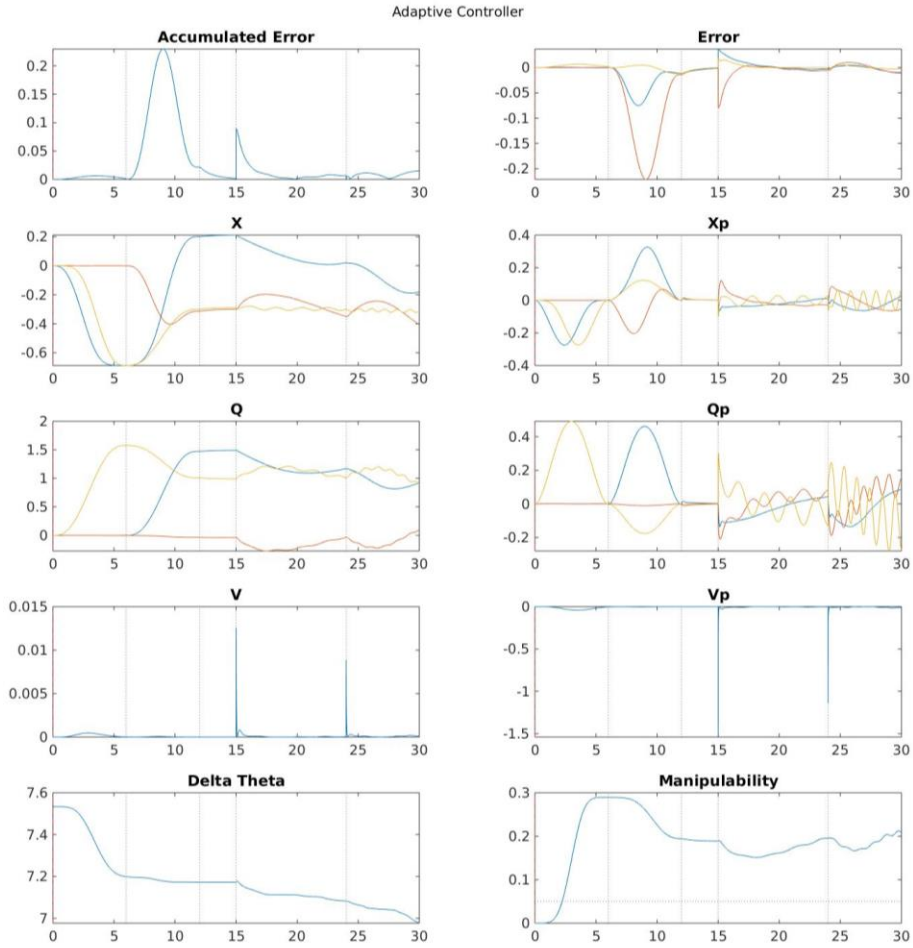

Figure 9 Adaptive controller, gamma is chosen to relate to the magnitudes of the original theta vector

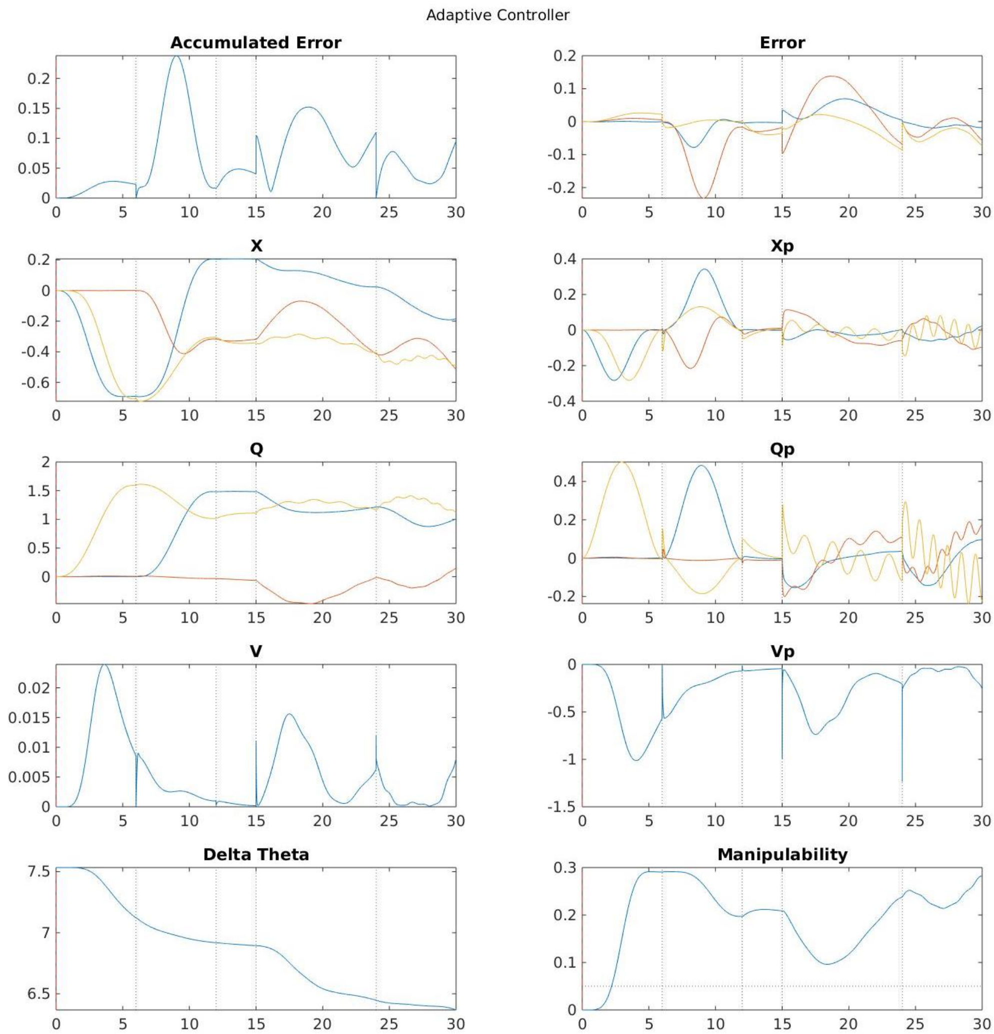

Figure 10 Adaptive controller with gamma as an identity matrix