EE6104 CA3 Report: Adaptive Control with All State-Variables Measurable

Hongxuan Wang
A0228929M
hongxuanwang@u.nus.edu

Abstract

An adaptive controller for continuous-time systems with all state-variables measurable is designed. The angular position of a d.c. motor is controlled, and different controlling parameters are tested using simulations on Simulink. The angular velocity control of the d.c. motor is also implemented using a similar controller with a higher dimension. Intermediate data and MATLAB/Simulink files are available at: `https://github.com/hxwangnus/EE6104-Adaptive-Control.git`.

1 Preliminaries

In the CA3 project, it is desired to design an adaptive controller for a d.c. motor apparatus with all state-variables measurable. A model reference adaptive system is built.

The plant model can be expressed by:

$$\dot{x}_p = A_p x_p + gbu, \tag{1}$$

where x_p is the state variables of the motor system:

$$x_p = \begin{pmatrix} \theta \\ \omega \end{pmatrix}, \tag{2}$$

where θ is the angular position and ω is the angular velocity.

The transfer function of the plant is given as:

$$\frac{\dot{\Theta}(s)}{U(s)} = \frac{K}{1+s\tau}, \tag{3}$$

then

$$(1+s\tau)\dot{\Theta}(s) = KU(s), \tag{4}$$

transfer to time domain,

$$\omega + \tau \frac{d}{dt}\omega = Ku, \tag{5}$$

$$\frac{d}{dt}\omega = -\frac{1}{\tau}\omega + \frac{K}{\tau}u, \tag{6}$$

where $\omega = \dot{\theta}$ is the angular velocity.

Combining Eq. (1), (2), and (6),

$$\dot{x}_p = \begin{pmatrix} \dot{\theta} \\ \dot{\omega} \end{pmatrix} = \begin{pmatrix} \omega \\ -\frac{1}{\tau}\omega + \frac{K}{\tau}u \end{pmatrix} = \begin{pmatrix} 0 & 1 \\ 0 & -\frac{1}{\tau} \end{pmatrix} \begin{pmatrix} \theta \\ \omega \end{pmatrix} + \frac{K}{\tau} \begin{pmatrix} 0 \\ 1 \end{pmatrix} u, \tag{7}$$

thus

$$A_p = \begin{pmatrix} 0 & 1 \\ 0 & -\frac{1}{\tau} \end{pmatrix}, \tag{8}$$

$$g = \frac{K}{\tau}, \tag{9}$$

$$b = \begin{pmatrix} 0 \\ 1 \end{pmatrix}. \tag{10}$$

The reference model is built as:

$$\dot{x}_m = A_m x_m + g_m b r, \tag{11}$$

where

$$A_m = \begin{pmatrix} 0 & 1 \\ -\omega_m^2 & -2\xi_m\omega_m \end{pmatrix}, \tag{12}$$

$$g_m = \omega_m^2, \tag{13}$$

which is the standard form for a second-order system.

The rest of the report is arranged as follows. In Section 2, the adaptive algorithm is derived. In Section 3, the calibration of the d.c. motor sensors are performed. In Section 4, the simulations are performed using Simulink. In Section 5, a similar adaptive controller is designed for angular velocity control, and simulations are conducted to explore its effectiveness.

2 Algorithm for the designed adaptive controller

2.1 Non-adaptive control law

To design an adaptive control algorithm, the non-adaptive control law is used as a reference, which is derived in this section.

The non-adaptive control law is designed as:

$$u(t) = \theta_x^T x_p + \theta_r r, \tag{14}$$

and the tracking error is:

$$e_1(t) = x_p - x_m. \tag{15}$$

Take the derivative of Eq. (15), and combining with Eq. (1) and (11),

$$\dot{e_1} = \dot{x}_p - \dot{x}_m = (A_p + gb\theta_x^T)x_p - A_m x_m + (g\theta_r - g_m)br. \tag{16}$$

Let,

$$A_p + gb\theta_x^{*T} = A_m, \tag{17}$$

$$g\theta_r^* = g_m, \tag{18}$$

Eq. (16) becomes:

$$\dot{e_1} = A_m e_1 < 0, \tag{19}$$

for $A_m < 0$ and non-zero e_1.

2.2 Adaptive law with all state-variables measurable

For an adaptive system with A_p and g unknown, the optimal control gain θ_x^* and θ_r^* should be replaced by the time-varying control gain $\theta_x(t)$ and $\theta_r(t)$, where:

$$\dot{\theta}_x(t) = -sgn(g)\Gamma e^T P b x_p, \tag{20}$$

$$\dot{\theta}_r(t) = -sgn(g)\gamma e^T P b r, \tag{21}$$

where Γ is any $n \times n$ positive definite matrix, and γ is any scalar such that $\gamma > 0$. For any stable A_m, there is:

$$A_m^T P + P A_m = -Q, \tag{22}$$

where Q is any symmetric positive definite matrix.

The **proof of convergence** is derived as follows.

The adaptive control law is now:

$$u(t) = \theta_x^T(t) x_p(t) + \theta_r(t) r(t), \tag{23}$$

thus the plant model becomes:

$$\dot{x}_p(t) = A_p x_p(t) + gb(\theta_x^T(t)x_p(t) + \theta_r(t)r(t)). \tag{24}$$

Define two error signals for control gains:

$$\phi_x(t) = \theta_x(t) - \theta_x^*, \tag{25}$$

$$\phi_r(t) = \theta_r(t) - \theta_r^*, \tag{26}$$

Eq. (24) becomes:

$$\dot{x}_p(t) = A_m x_p(t) + g_m b r(t) + gb(\phi_x^T(t)x_p(t) + \phi_r(t)r(t)). \tag{27}$$

According to Eq. (15),

$$\dot{e_1}(t) = \dot{x}_p(t) - \dot{x}_m(t) = A_m e_1(t) + gb(\phi_x^T(t)x_p(t) + \phi_r(t)r(t)). \tag{28}$$

Define a Lyapunov equation as:

$$V(e_1, \phi_x, \phi_r) = \frac{1}{2}e_1^T P e_1 + \frac{1}{2}|g|(\phi_x^T \Gamma^{-1} \phi_x + \frac{1}{\gamma}\phi_r^2), \tag{29}$$

then the derivative of V is:

$$\dot{V} = e_1^T P A_m e_1 + e_1^T P g b(\phi_x^T x_p + \phi_r r) + |g|(\phi_x^T \Gamma^{-1} \dot{\phi}_x + \frac{1}{\gamma}\phi_r \dot{\phi}_r). \tag{30}$$

According to Eq. (25) and (26),

$$\dot{\phi}_x = \dot{\theta}_x, \tag{31}$$

$$\dot{\phi}_r = \dot{\theta}_r, \tag{32}$$

combined with Eq. (20) and (21), then Eq. (30) becomes:

$$\dot{V} = e_1^T P A_m e_1 = -\frac{1}{2}e_1^T Q e_1 \leq 0, \tag{33}$$

which proves the negative semi-definiteness of V, meaning that V is bounded for all t, and e_1, ϕ_x, ϕ_r are bounded.

Given bounded e_1, ϕ_x, ϕ_r, it can be derived that x_p and u are also bounded, thus according to Eq. (28), $\dot{e}_1$ is also bounded.

Moreover, let:

$$\alpha(e_1, \phi_x, \phi_r) = \frac{1}{2}min\{eig(P), eig(|g|\Gamma^{-1}), eig(|g|\frac{1}{\gamma})\}(e_1^T e_1 + \phi_x^T \phi_x + \phi_r^2), \tag{34}$$

then

$$V \geq \alpha \geq 0, \quad \alpha(0) = 0, \tag{35}$$

which proves V is positive-definite.

Similarly, let:

$$\beta(e_1, \phi_x, \phi_r) = \frac{1}{2}max\{eig(P), eig(|g|\Gamma^{-1}), eig(|g|\frac{1}{\gamma})\}(e_1^T e_1 + \phi_x^T \phi_x + \phi_r^2), \tag{36}$$

then:

$$V \leq \beta, \quad \beta(0) = 0, \tag{37}$$

which proves V is decrescent.

Also, as

$$||V|| \to \infty \quad as \quad ||(e_1, \phi_x, \phi_r|| \to \infty, \tag{38}$$

V is radial unbounded, thus the system is uniformly stable in the large.

Lastly, $||e_1||$ and $||\dot{e}_1||$ are bounded, so e_1 is globally asymptotically stable:

$$lim_{t\to\infty} e_1 = 0. \tag{39}$$

3 Calibration of the D.C. motor sensors

In real-time experiments, the angular position and angular velocity would be calculated from the reading of the potentiometer and tachometer, so calibration should be conducted to relate the outputs of the sensors to the actual position and velocity of the motor.

According to [1],

$$X_1 = K_\theta \theta, \tag{40}$$

$$X_2 = K_\omega \omega, \tag{41}$$

where X_1 and X_2 are the outputs of the potentiometer and tachometer, respectively. θ is the actual angular position and ω is the actual angular velocity. K_θ and K_ω are the two coefficients that should be calibrated in this section.

The calibration data are provided in [1], and shown as line $3-4$, and line $8-9$ in the MATLAB codes, which are attached as follows.

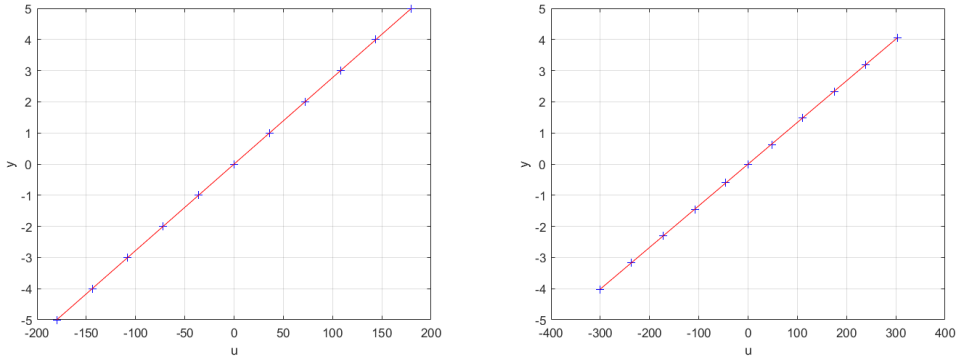
```
1  % Linear Regression Using MATLAB
2  % The linear model is $y=ku$, where k is the parameter to be
       estimated.
3  u1 = [-180,-144,-108,-72,-36,0,36,72,108,144,180]'; % angular position
4  y1 = [-5,-4,-3,-2,-1,0,1,2,3,4,5]';      % potientiometer output
5  mdl1 = fitlm(u1,y1,'intercept',false);
6  disp(['The estimation of k_theta is:
       ',num2str(mdl1.Coefficients{1,1})])
7
8  u2 = [-301,-237,-172,-108,-45,0,48,111,175,239,303]'; % angular
       velocity
9  y2 = [-4.03,-3.17,-2.3,-1.45,-0.6,0,0.62,1.48,2.33,3.2,4.06]';%
       tachometer output
10 mdl2 = fitlm(u2,y2,'intercept',false);
11 disp(['The estimation of k_w is: ',num2str(mdl2.Coefficients{1,1})])
12
13 figure(1)
14 plot(u1,y1,'+b')
15 hold on;
16 plot(u1,mdl1.Fitted,'-r')
17 grid on;
18 xlabel('u'); ylabel('y');
19
20 figure(2)
21 plot(u2,y2,'+b')
22 hold on;
23 plot(u2,mdl2.Fitted,'-r')
24 grid on;
25 xlabel('u'); ylabel('y');
```

According to line $3-4$, $y1$ should be a linear function of $u1$, and the slope value is K_θ. According to line $8-9$, $y2$ should be a linear function of $u2$, and the slope value is K_ω. The MATLAB builtin function "**fitlm**" is used to perform the linear regression, and the results are shown in Figure 1.

The exact values of K_θ and K_ω are shown in Table 1.

Table 1: Calibration results for K_θ and K_ω.

K_θ	K_ω
0.027778	0.013378



(a) Calibration result for angular position.

(b) Calibration result for angular velocity.

Figure 1: Calibration results for angular position and angular velocity.

4 Simulation using Simulink

In this section, a Simulink model is built and simulations are performed accordingly.

The plant is simulated as:

$$\frac{\dot{\Theta}(s)}{U(s)} = \frac{6.2}{1 + 0.25s}, \tag{42}$$

and the initial reference model parameters are set as $\omega_m = 10$, $\xi = 1$.

The initial value for Γ is $[100, 0; 0, 1]$, and for γ is 1000. The initial value for Q is $[100, 0; 0, 100]$. All these parameters will be changed in the following sections, in order to explore the effectiveness of the adaptive controller.

The Simulink models are shown in Figure 2 to Figure 4. Reference signal $r(t)$ is initially chosen as a step signal with final value $= 1$.

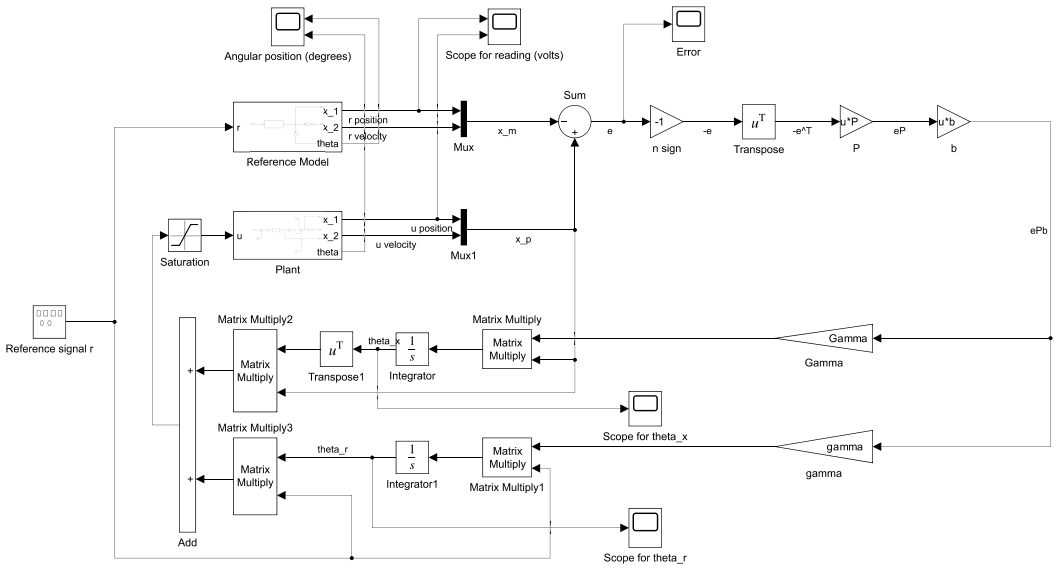


Figure 2: Simulink model of the model reference adaptive control system.

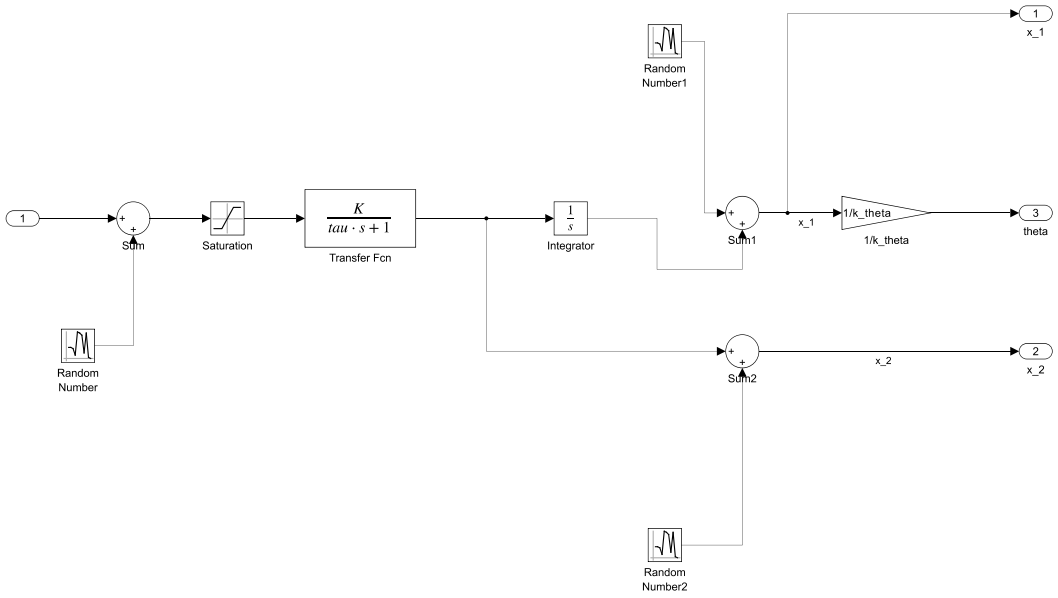


Figure 3: Plant model.

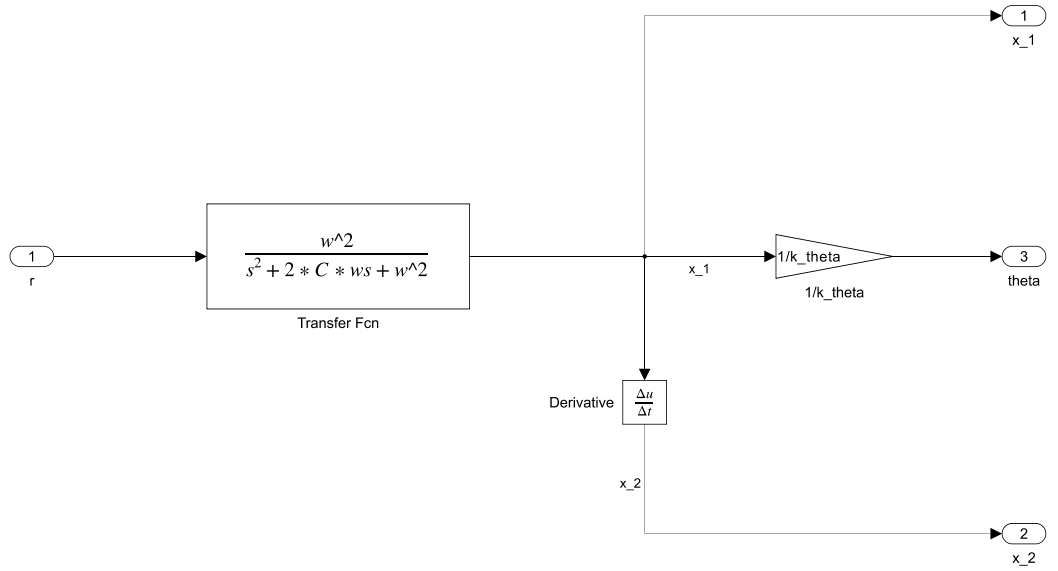


Figure 4: Reference model.

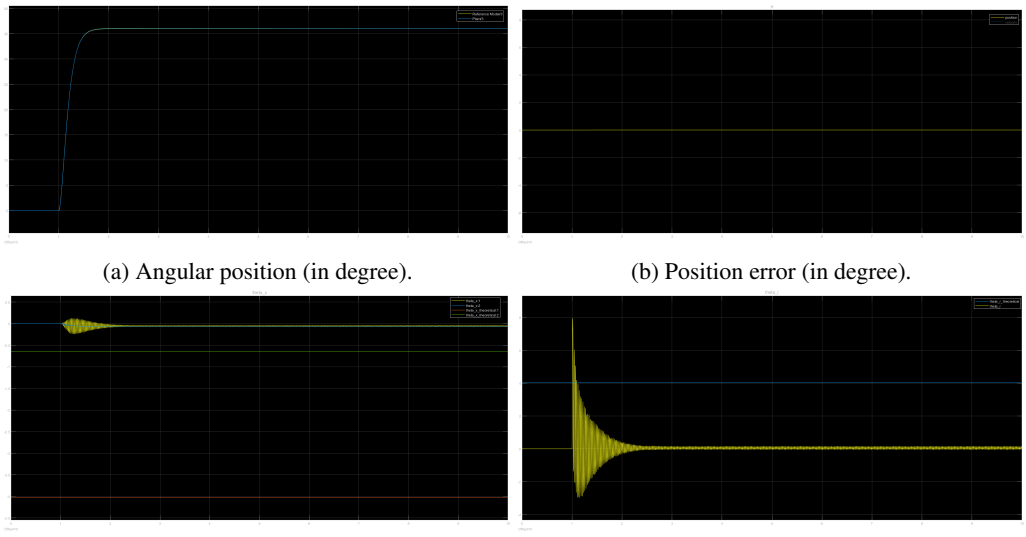
The MATLAB codes are attached as follows, and all initial parameters are used.

```
% parameters from calibration
CA3_calibration;
k_theta=mdl1.Coefficients{1,1};
k_w=mdl2.Coefficients{1,1};

% parameters for the plant
K=6.2;
tau=0.25;
b=[0;1];

```

```
Ap=[0 1;0 -1/tau];
g=K/tau;

% parameters for the reference model
C=1;
w=10;

Am=[0 1;-w^2 -2*C*w];
gm=w^2;

% Gamma & gamma
Gamma=[100 0; 0 1];
gamma=1000;

% Lyapunov
Q=[100 0; 0 100];
P=lyap(Am', Q);

% control gain
theta_x=((b'*(Am-Ap))./g)';
theta_r=(tau*w^2)/K;
disp(['The theoretical control gain theta_x1 is:
    ',num2str(theta_x(1))])
disp(['The theoretical control gain theta_x2 is:
    ',num2str(theta_x(2))])
disp(['The theoretical control gain theta_r is: ',num2str(theta_r)])
```

```
clc
clear all;
CA3_param;

open_system('CA3_model')
evalc('sim(''CA3_model'')');
```

The simulation results are shown in Figure 5.



(a) Angular position (in degree).

(b) Position error (in degree).

(c) θ_x.

(d) θ_r.

Figure 5: Simulation results for the initial set of parameters.

According to Figure 5a and 5b, the plant angular position tracks the reference model angular position well. However, according to Figure 5c and 5d, the adaptive gains are not equal to the theoretical

"optimal" gains calculated using the model parameters. It means that even the real gains are not equal to the theoretical "optimal" gains, the adaptive system is still able to track the reference output well.

In the next sections, the model parameters, control parameters, and reference signal parameters are adjusted, and measurement noise is considered, and plant change is also considered, to explore the effectiveness of the adaptive scheme.

4.1 Effects of A_m and g_m

In this section, the reference model parameters are changed. According to Eq. (12) and (13), the reference model is a standard second-order system, which is defined by its damping ratio ξ_m and natural frequency ω_m. In the codes provided in Section 4, ξ_m is represented by C, and ω_m is represented by w. The initial values are $C = 1$ and $w = 10$.

Four simulations are conducted, where (C, w) pairs are changed with values $(0.1, 10)$, $(10, 10)$, $(1, 1)$ and $(1, 100)$. Simulation results are shown in Appendix, Figure 12 - Figure 15, respectively.

It can be observed that, as C becomes smaller, the imaginary parts of the poles of the reference model are further away from the origin, then the system has an increase in the oscillation frequency. As C becomes larger, the oscillation frequency decreases, and the transient response speed is slower.

As w becomes smaller, the imaginary parts of the poles of the reference system are closer to the origin, so the oscillation frequency decreases. As w becomes larger, the oscillation frequency increases, and the transient response is faster.

In the case that (C, w) equals to $(0.1, 10)$, $(10, 10)$, and $(1, 1)$, the plant positions track the reference model well. However, when w is large, in the case $w = 100$, the plant is not able to track the reference model in such a fast transient response, and significant overshoots are there.

Meanwhile, in all four simulations, the adaptive gains are not equal to the theoretical "optimal" gains, which means that even the real gains are not equal to the theoretical "optimal" gains, the adaptive system is still able to track the reference output.

Comparing the four simulation results, in the following simulations, $C = 1$ and $w = 10$ will be used.

4.2 Effects of Γ and γ

In this section, the positive definite matrix Γ and positive real scalar γ are adjusted. According to Eq. (20) and (21), Γ will affect θ_x and γ will affect θ_r. The initial values are $\Gamma = [100, 0; 0, 1]$ and $\gamma = 1000$.

Four simulations are conducted. The first simulation uses $\Gamma = [1, 0; 0, 1]$ and $\gamma = 1000$; the second simulation uses $\Gamma = [1000, 0; 0, 1000]$ and $\gamma = 1000$; the third uses $\Gamma = [100, 0; 0, 1]$ and $\gamma = 10$; and the fourth uses $\Gamma = [100, 0; 0, 1]$ and $\gamma = 10000$. Simulation results are shown in Appendix, Figure 16 - Figure 19, respectively.

It can be observed that, when the diagonal elements of Γ or the value of γ become larger, the tracking performance becomes better; when the diagonal elements of Γ or the value of γ become smaller, the tracking performance gets worse as some tracking errors are there. This is consistent with the general rule in control, where larger control gains improve the control performance. However, in real applications, improperly large control gains may cause the system unstable.

It can be observed from Figure 16c and Figure 17c, that when the element values of Γ become larger, the change of control gain θ_x becomes sharper. This can be explained by Eq. (20), that when Γ has larger diagonal elements, the derivative of θ_x becomes larger, thus the rate of change of θ_x is larger. This also happens to γ, and the reason is the same as that of Γ.

Comparing the four simulation results, in the following simulations, $\Gamma = [100, 0; 0, 100]$ and $\gamma = 1000$ will be used.

4.3 Effects of positive definite matrix Q

In this section, the positive definite matrix Q is adjusted. Q is used in the Lyapunov function to determine matrix P, which also affects the slope of the control gains. A reasonable guess is that Q acts similarly to Γ and γ.

The initial value of Q is $[100, 0; 0, 100]$. Two simulations are conducted, the first of which uses $Q = [1, 0; 0, 1]$, and the second of which uses $Q = [10000, 0; 0, 10000]$. Simulation results are shown in Appendix, Figure 20 - Figure 21, respectively.

It can be observed that, when the diagonal elements values of Q increase, the tracking error decreases, but the rates of change of the control gains are sharper, which proves our hypothesis correct.

Comparing the results, in the following simulations, $Q = [100, 0; 0, 100]$ will be used.

4.4 Effects of reference signal $r(t)$

In the previous sections, $r(t)$ is chosen to be step signals with steady-state value 1. In this section, several different choices of $r(t)$ are used to conduct simulations.

4.4.1 $r(t)$ is a step signal

In this part, $r(t)$ is still a step signal, but with steady-state values 3. A simulation is conducted, and the simulation results are shown in Appendix, Figure 22.

It can be observed that the tracking performance is a little bit worse than the previous simulations when $r(t)$ has a steady-state value 1. It may be because the control gains are still small in such a condition, or the plant is not "powerful" enough to achieve a fast transient response.

4.4.2 $r(t)$ is a periodic signal

In this part, $r(t)$ is chosen to be periodic signals, e.g., square waves or sinusoidal waves. Two simulations are conducted, one uses square wave and the other uses sine wave, and both with amplitude 1 and frequency $0.05Hz$.

The simulation results are shown in Appendix, Figure 23 - Figure 24. It can be observed that in both cases, the plant outputs track the reference positions well, although the adaptive gains are not equal to the theoretical "optimal" gains.

4.4.3 $r(t)$ is a random signal

In this part, random $r(t)$ is used, and the simulation results are shown in Appendix, Figure 25. Although the tracking error is larger than those of the previous simulations, the plant output has a significant trend in tracking the reference angular position, which shows the adaptive performance in some aspects.

4.5 Effects of realistic noise in measurements

In real-time experiments, measurement noise should be considered during the design of controllers. In this section, three random noise signals with zero mean and small variances are introduced to the plant model, as shown in Figure 3. The reference signal is still chosen as a step signal with a steady-state value 1. The simulation results are shown in Figure 26.

4.6 Adaptive performance verification through plant change

In this section, the plant is initially:

$$H(s) = \frac{6.2}{1 + 0.25s}. \tag{43}$$

After $80s$, it is changed to:

$$H(s) = \frac{4.5}{1 + 0.45s}, \tag{44}$$

in order to verify the effectiveness of the adaptive controller.

The re-designed Simulink model is shown in Figure 6.

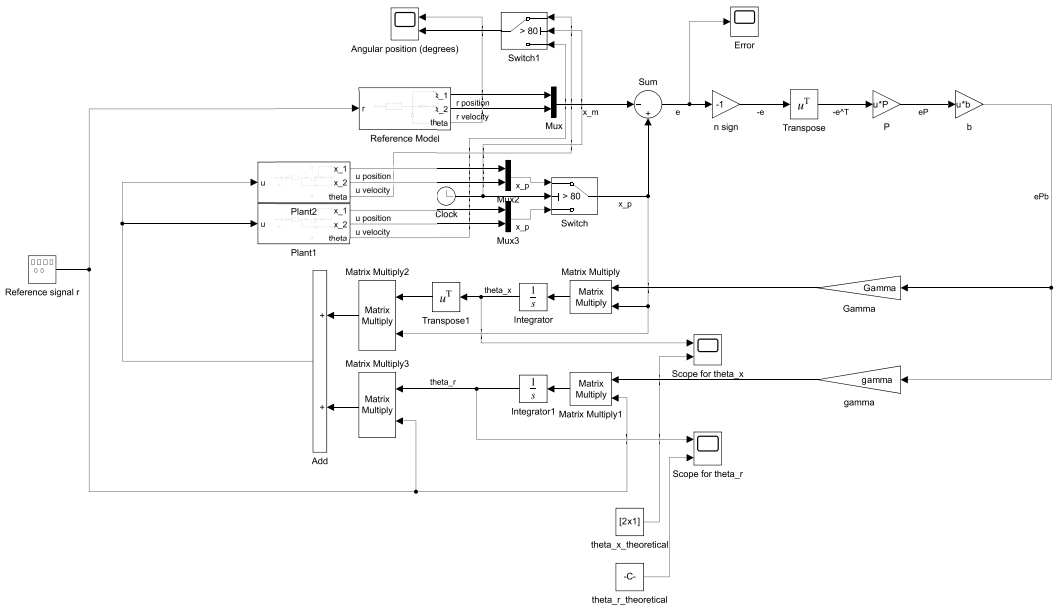
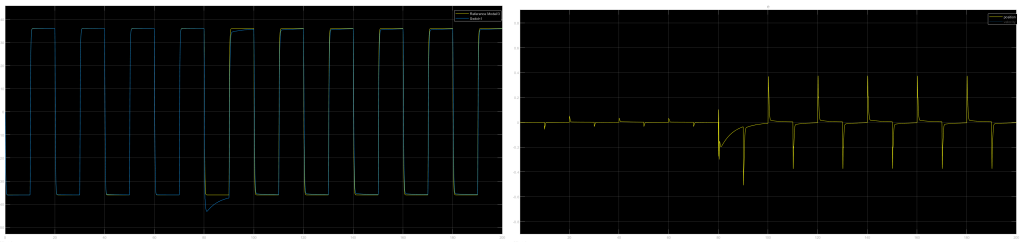


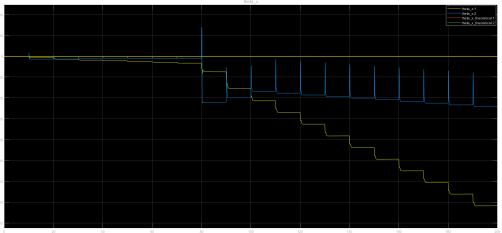
Figure 6: Simulink model of the change plant simulation.

The simulation results are shown in Figure 7. It can be observed that during the initial $80s$, the plant output tracks the reference signal well. At $80s$, the plant parameters change suddenly, so the tracking performance becomes worse for a short period (within $20s$). Starting from $100s$, although the tracking error is larger than the initial condition, the plant output still can track the reference signal. According to Figure 7c and 7d, the adaptive gains automatically change their value due to the plant change, which also proves the effectiveness of the adaptive controller.

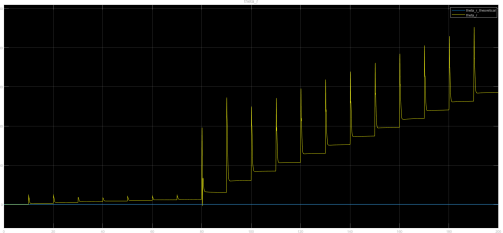


(a) Angular position (in degree).

(b) Position error (in degree).



(c) θ_x.



(d) θ_r.

Figure 7: Simulation results for plant-change experiment.

5 Angular velocity control

In this section, we modify the previous adaptive control method to control the angular velocity of the d.c. motor.

5.1 Design of control law and adaptive law

The plant model should be modified to:

$$\frac{\ddot{\Theta}(s)}{U(s)} = \frac{K}{1+s\tau}, \tag{45}$$

so in the time domain, it can be transformed as:

$$\frac{d}{dt}\ddot{\theta} = -\frac{1}{\tau}\ddot{\theta} + \frac{K}{\tau}u. \tag{46}$$

For the plant model equation:

$$\dot{x}_p = A_p x_p + gbu, \tag{47}$$

x_p is the state variables of the new system:

$$x_p = \begin{pmatrix} \theta \\ \omega \\ \dot{\omega} \end{pmatrix}, \tag{48}$$

thus

$$A_p = \begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & -\frac{1}{\tau} \end{pmatrix}, \tag{49}$$

$$g = \frac{K}{\tau}, \tag{50}$$

$$b = \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix}. \tag{51}$$

The reference model's output changes from angular position θ to angular velocity ω:

$$\frac{\dot{\Theta}(s)}{U(s)} = \frac{\omega_m^2}{s^2 + 2C_m\omega_m s + \omega_m^2}. \tag{52}$$

However, an observer for the angular position can be built using integration:

$$\dot{\Theta}(s) = s\Theta(s), \tag{53}$$

thus we have:

$$\frac{\Theta(s)}{U(s)} = \frac{\Theta(s)}{\dot{\Theta}(s)}\frac{\dot{\Theta}(s)}{U(s)} = \frac{1}{s}\frac{\omega_m^2}{s^2 + 2C_m\omega_m s + \omega_m^2}. \tag{54}$$

In time domain representation,

$$\dot{x}_m = A_m x_m + g_m br, \tag{55}$$

we have:

$$x_m = \begin{pmatrix} \theta \\ \omega \\ \dot{\omega} \end{pmatrix}, \tag{56}$$

$$A_m = \begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & -\omega_m^2 & -2C_m\omega_m \end{pmatrix}, \tag{57}$$

$$g_m = \omega_m^2. \tag{58}$$

After completing the definition of the plant model and reference model, the adaptive law design and convergence proof in Section 2 are still fully established, then the design and proof steps are exactly the same.

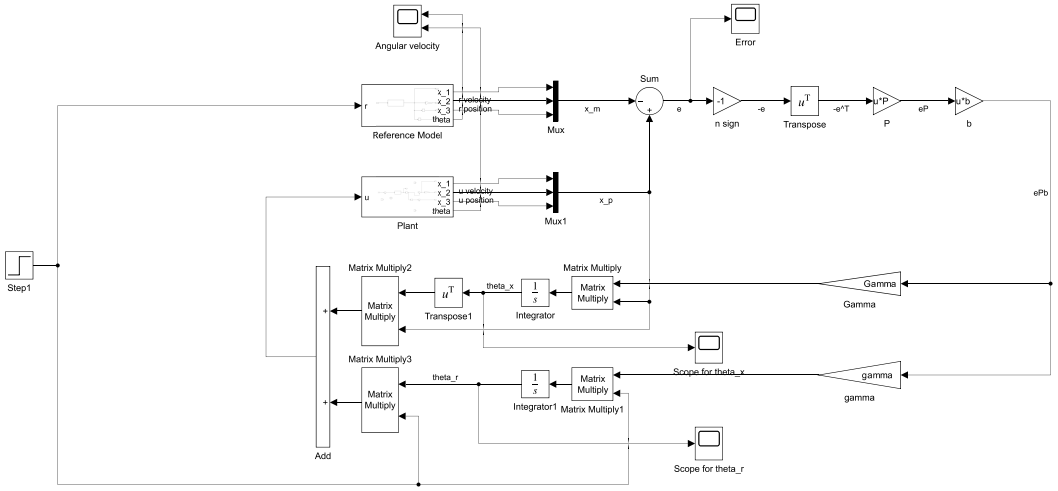


Figure 8: Simulink model for angular velocity control.

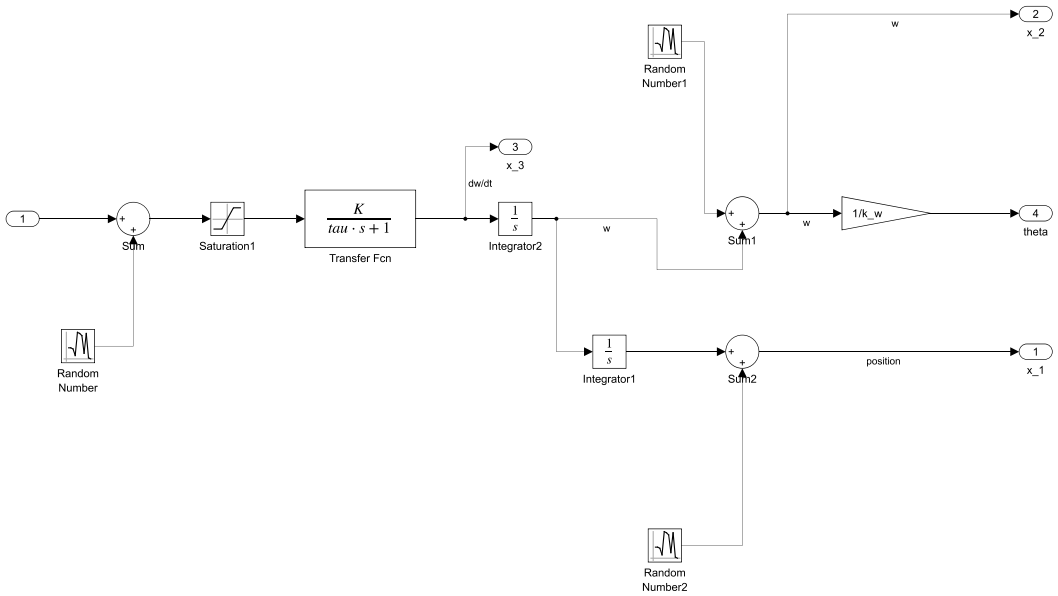


Figure 9: Plant model for velocity control.

5.2 Simulation

The Simulink models are shown in Figure 8 - Figure 10.

The MATLAB codes are modified according to the new dimensions of the parameters, which are attached as follows.

```
% parameters from calibration
CA3_calibration;
k_theta=mdl1.Coefficients{1,1};
k_w=mdl2.Coefficients{1,1};

% parameters for the plant
K=6.2;
tau=0.25;
```

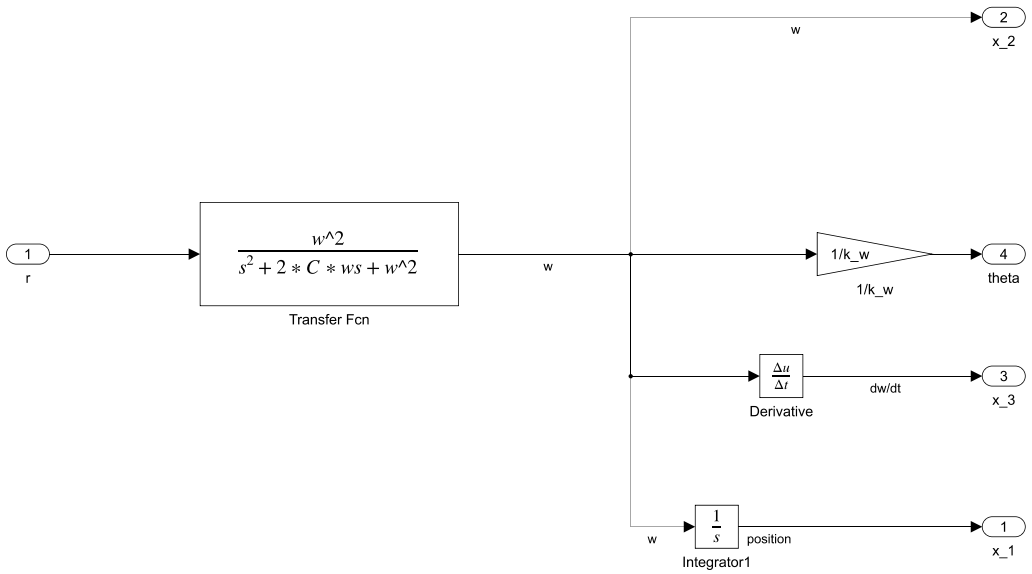
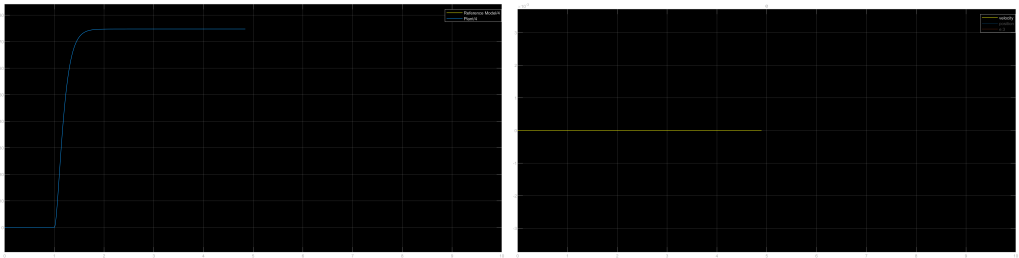


Figure 10: Reference model for velocity control.

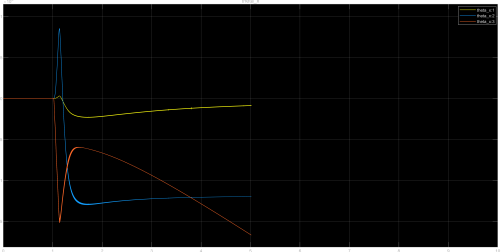
```
b=[0;0;1];

Ap=[0 1 0;0 0 1;0 0 -1/tau];
g=K/tau;

% parameters for the reference model, value are changed to explore
C=1;
w=10;

Am=[0 1 0;0 0 1;0 -w^2 -2*C*w];
gm=w^2;

% Gamma & gamma, value are changed to explore effect
Gamma=[1 0 0; 0 1 0; 0 0 1];
gamma=1;

% Lyapunov
Q=[1 0 0; 0 1 0; 0 0 1]; % value are changed to explore
P=lyap(Am', Q);

% control gain
theta_x=((b'*(Am-Ap))./g)';
theta_r=(tau*w^2)/K;
disp(['The theoretical control gain theta_x1 is:
    ',num2str(theta_x(1))])
disp(['The theoretical control gain theta_x2 is:
    ',num2str(theta_x(2))])
disp(['The theoretical control gain theta_r is: ',num2str(theta_r)])
```

The simulation results are shown in Figure 11, where a step signal is used as the reference signal $r(t)$. Due to the limitation of computational resources, the simulation only executes for about $5s$. However, it can be observed that the plant angular velocity tracks the reference signal well.

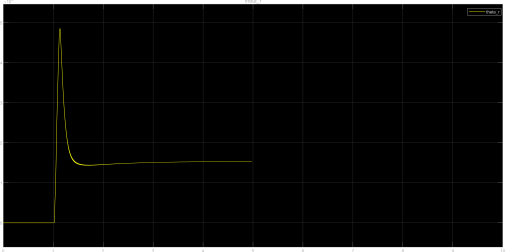


(a) Angular velocity (rad/s).

(b) Velocity error.



(c) θ_x.



(d) θ_r.

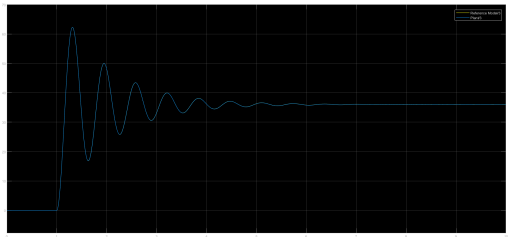
Figure 11: Simulation results for angular velocity control.

References

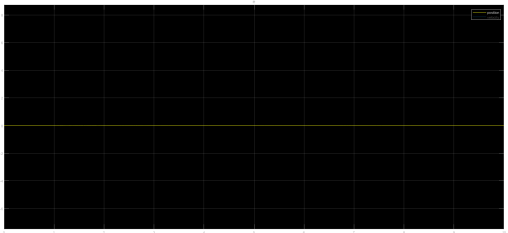
[1] Briefing Notes for CA3 Mini Project / Sep 2018: Adaptive Control of Angular Position with Full State Measurable (EE6104 & EE5104) and Adaptive Control of Angular Velocity (EE6104 only) being explored on the D.C. Motor.

[2] Lecture Notes of EE5104/6104 Advanced/Adaptive Control Systems.

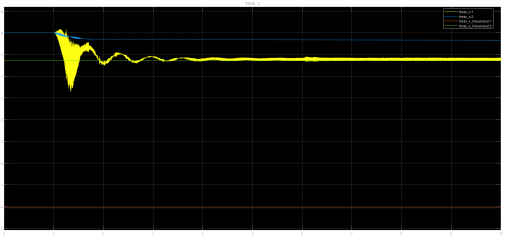
Appendix



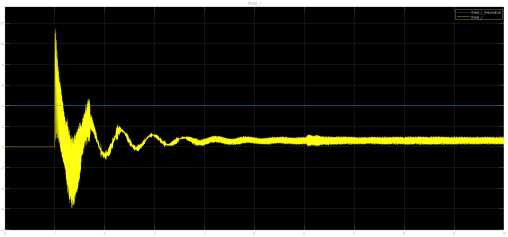
(a) Angular position (in degree).



(b) Position error (in degree).

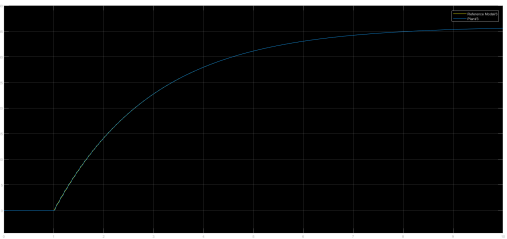


(c) θ_x.

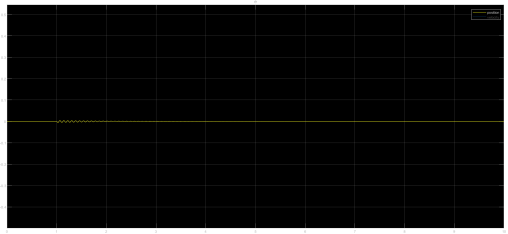


(d) θ_r.

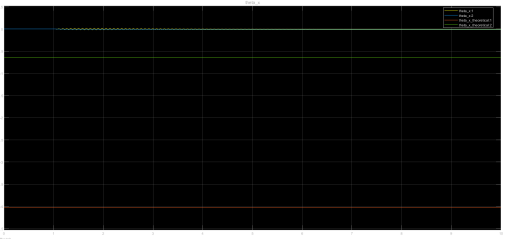
Figure 12: Simulation results for $C = 0.1$ and $w = 10$.



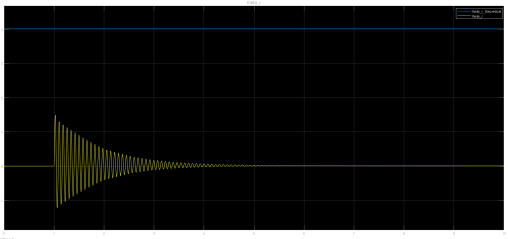
(a) Angular position (in degree).



(b) Position error (in degree).

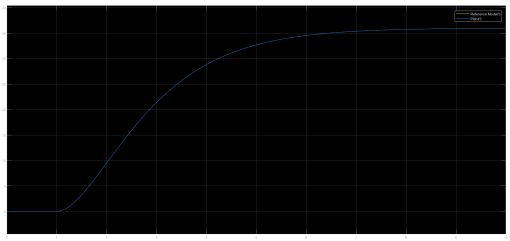


(c) θ_x.

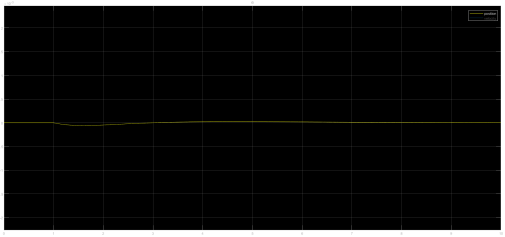


(d) θ_r.

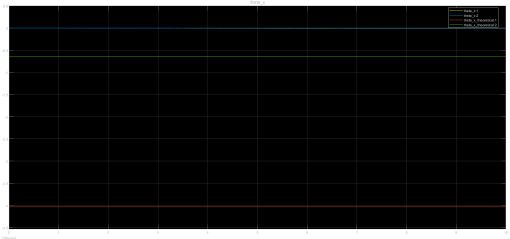
Figure 13: Simulation results for $C = 10$ and $w = 10$.



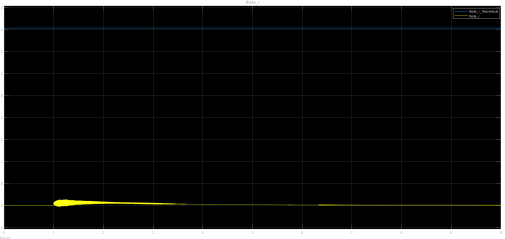
(a) Angular position (in degree).



(b) Position error (in degree).

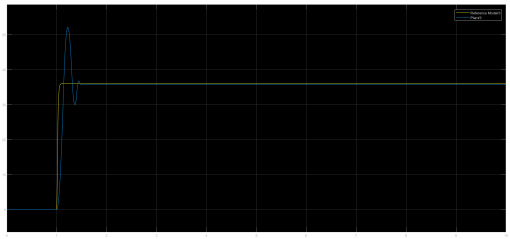


(c) θ_x.

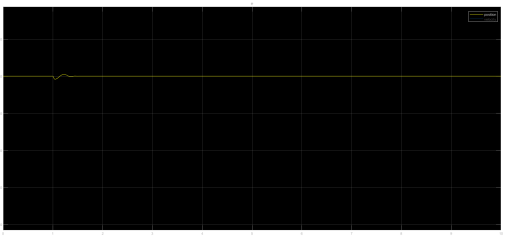


(d) θ_r.

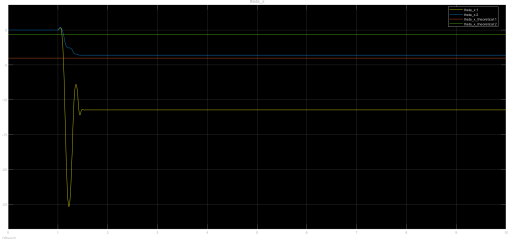
Figure 14: Simulation results for $C = 1$ and $w = 1$.



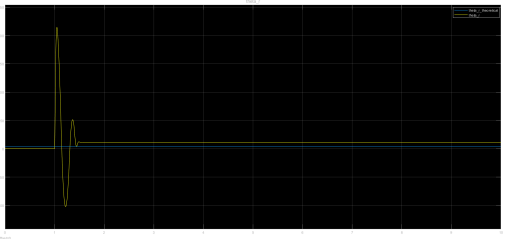
(a) Angular position (in degree).



(b) Position error (in degree).

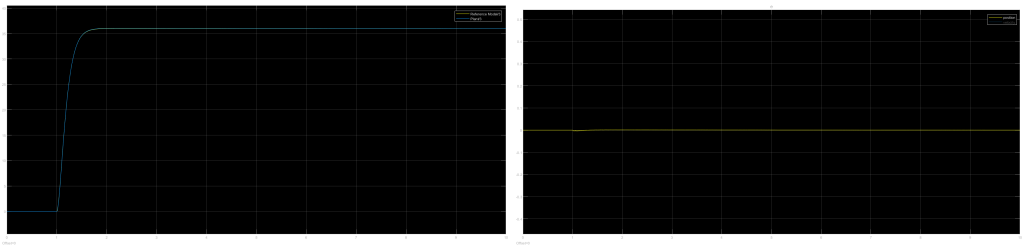


(c) θ_x.

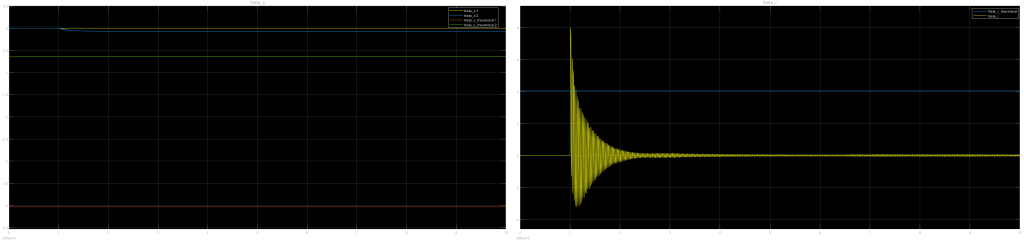


(d) θ_r.

Figure 15: Simulation results for $C = 1$ and $w = 100$.

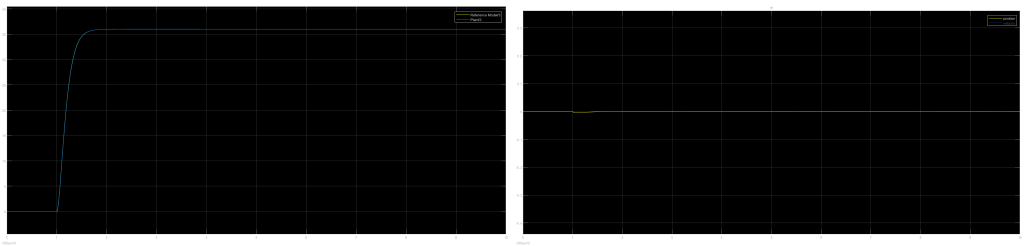


(a) Angular position (in degree). (b) Position error (in degree).

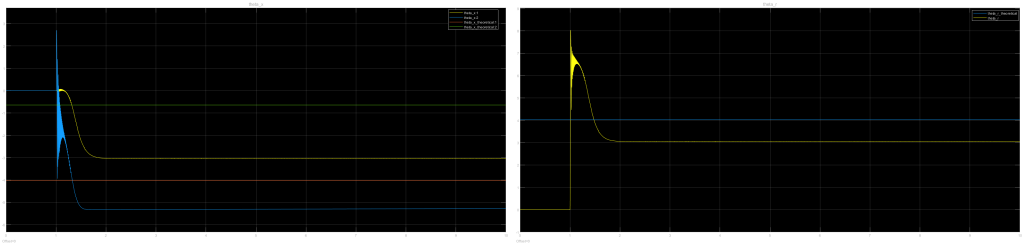


(c) θ_x. (d) θ_r.

Figure 16: Simulation results for $\Gamma = [1, 0; 0, 1]$ and $\gamma = 1000$.

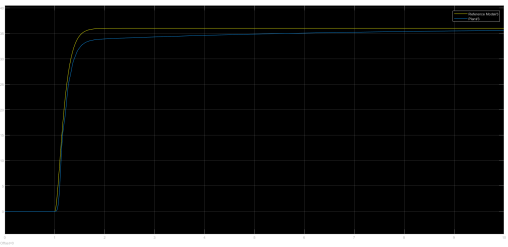


(a) Angular position (in degree). (b) Position error (in degree).

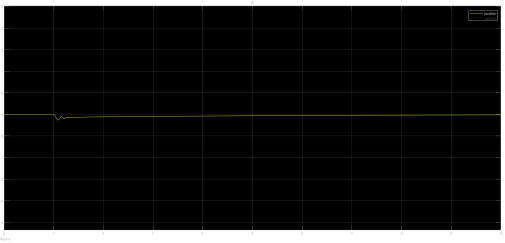


(c) θ_x. (d) θ_r.

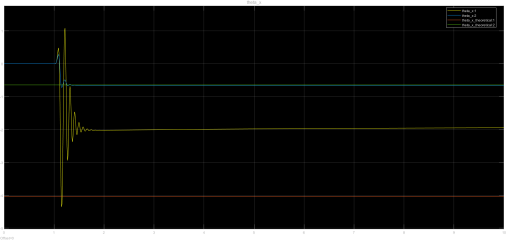
Figure 17: Simulation results for $\Gamma = [1000, 0; 0, 1000]$ and $\gamma = 1000$.



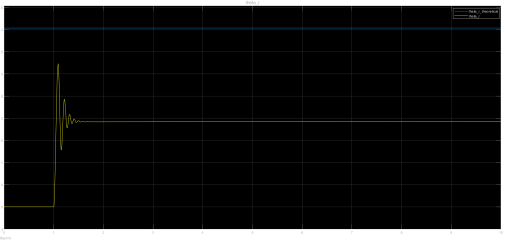
(a) Angular position (in degree).



(b) Position error (in degree).

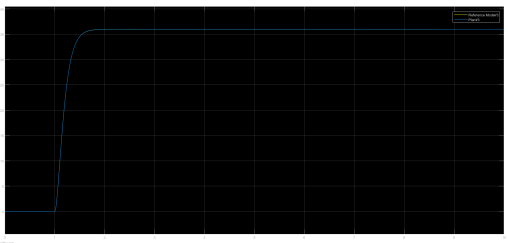


(c) θ_x.

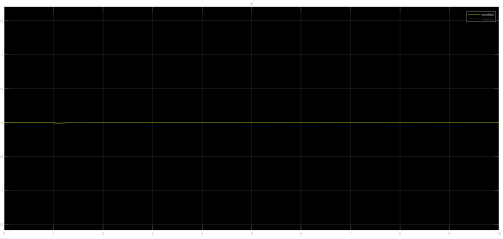


(d) θ_r.

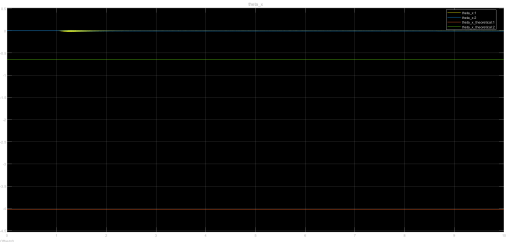
Figure 18: Simulation results for $\Gamma = [100, 0; 0, 1]$ and $\gamma = 10$.



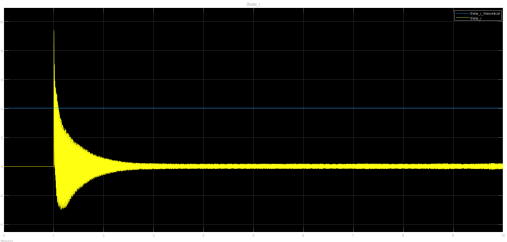
(a) Angular position (in degree).



(b) Position error (in degree).

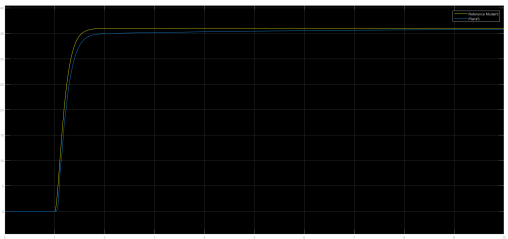


(c) θ_x.

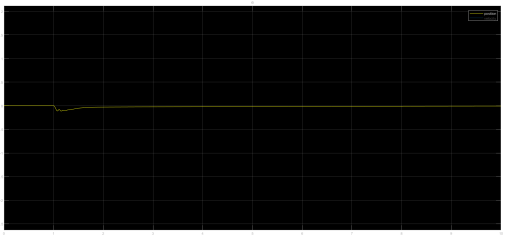


(d) θ_r.

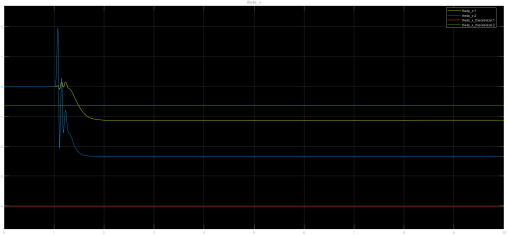
Figure 19: Simulation results for $\Gamma = [100, 0; 0, 1]$ and $\gamma = 10000$.



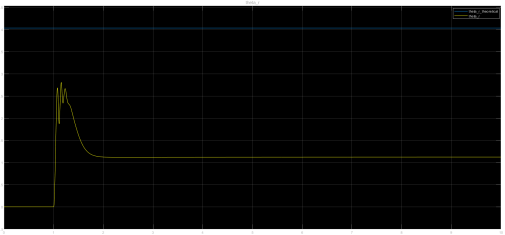
(a) Angular position (in degree).



(b) Position error (in degree).

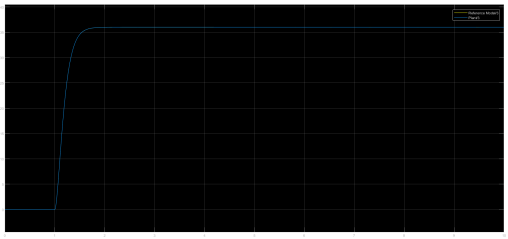


(c) θ_x.

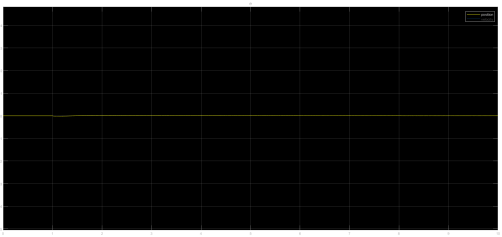


(d) θ_r.

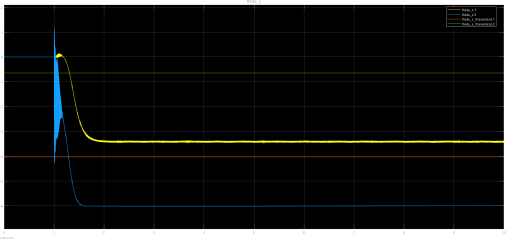
Figure 20: Simulation results for $Q = [1, 0; 0, 1]$.



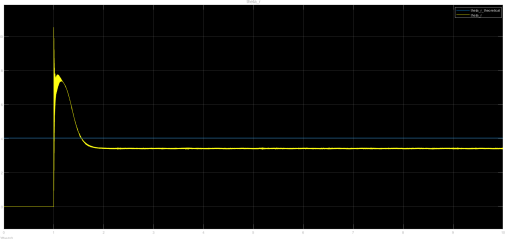
(a) Angular position (in degree).



(b) Position error (in degree).

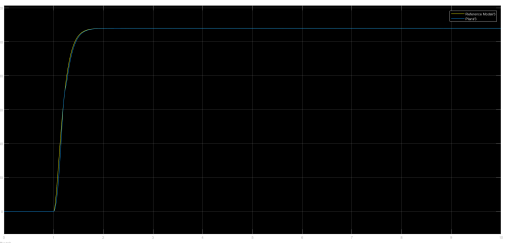


(c) θ_x.

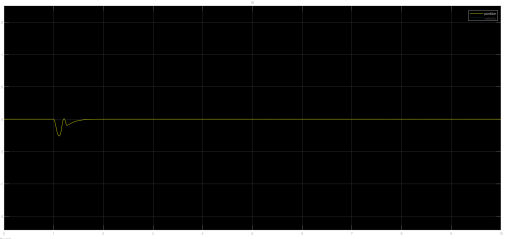


(d) θ_r.

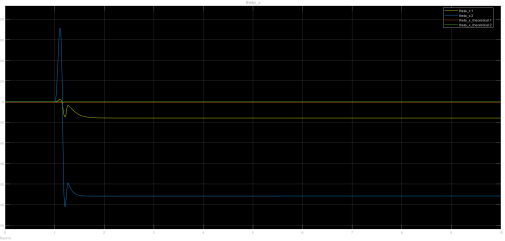
Figure 21: Simulation results for $Q = [10000, 0; 0, 10000]$.



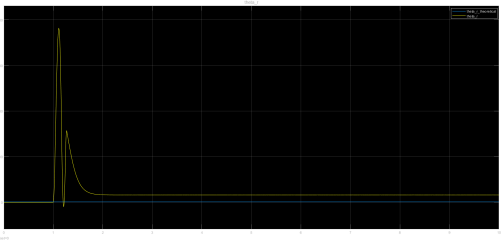
(a) Angular position (in degree).



(b) Position error (in degree).

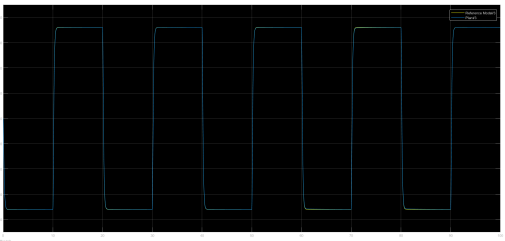


(c) θ_x.

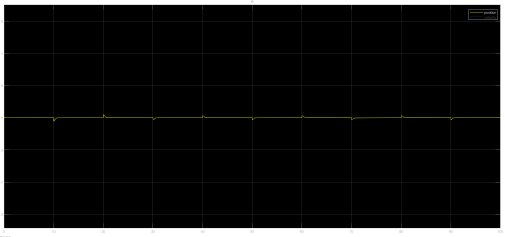


(d) θ_r.

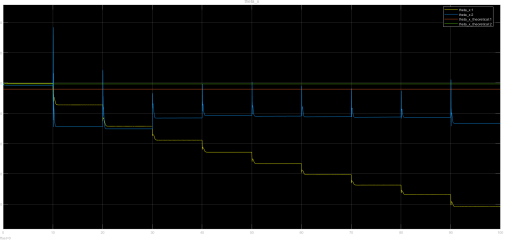
Figure 22: Simulation results for step signal $r(t)$ with steady-state value 3.



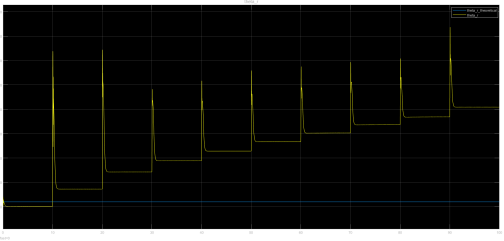
(a) Angular position (in degree).



(b) Position error (in degree).

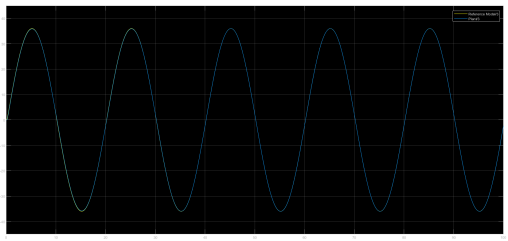


(c) θ_x.

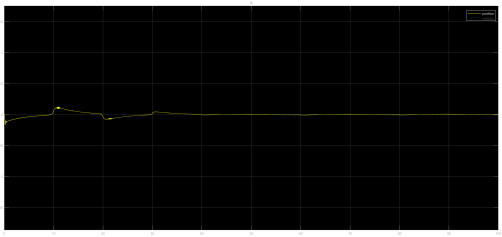


(d) θ_r.

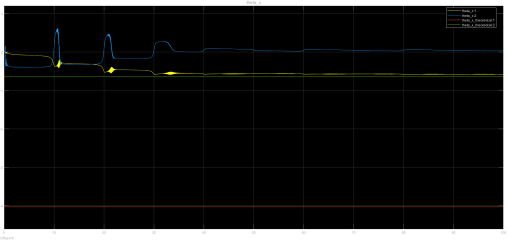
Figure 23: Simulation results for square signal $r(t)$.



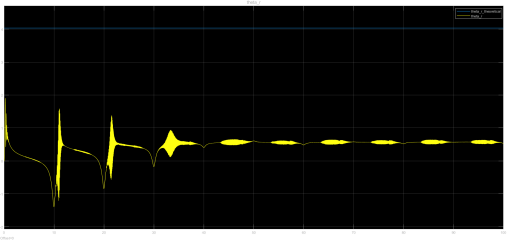
(a) Angular position (in degree).



(b) Position error (in degree).

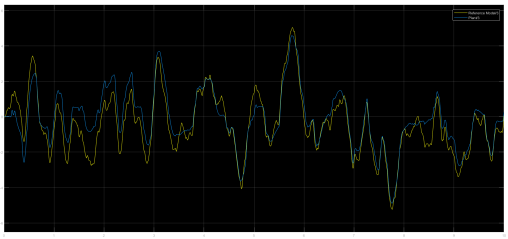


(c) θ_x.

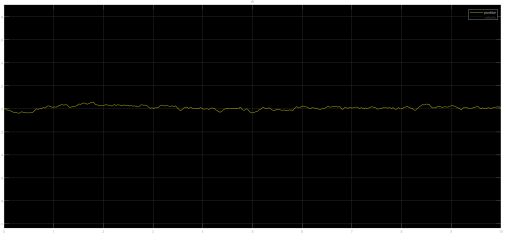


(d) θ_r.

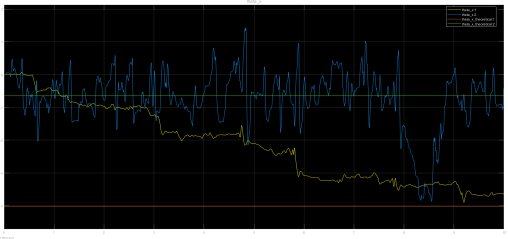
Figure 24: Simulation results for sine signal $r(t)$.



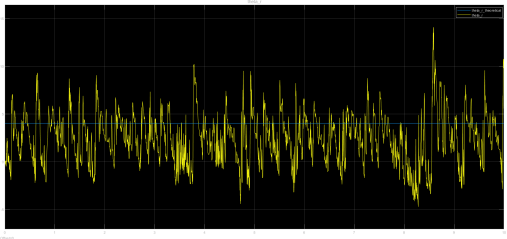
(a) Angular position (in degree).



(b) Position error (in degree).

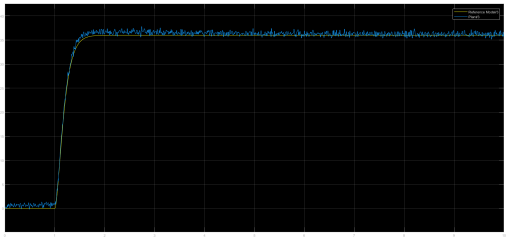


(c) θ_x.

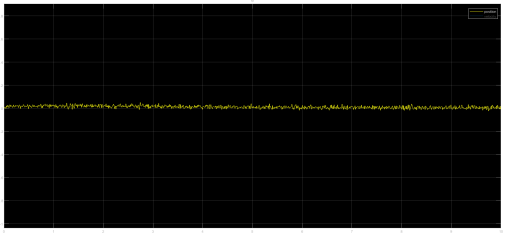


(d) θ_r.

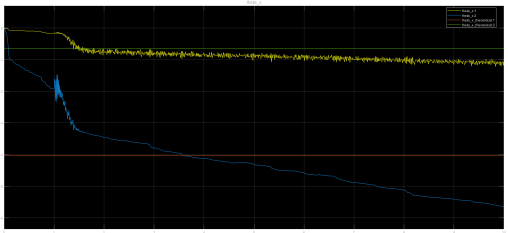
Figure 25: Simulation results for random signal $r(t)$.



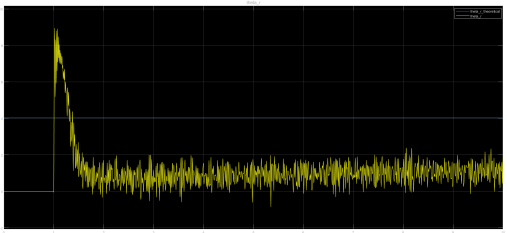
(a) Angular position (in degree).



(b) Position error (in degree).



(c) θ_x.



(d) θ_r.

Figure 26: Simulation results for noised measurements.