Report on Nonlinear Adaptive Control of Permanent Magnet Synchronous Motor

by

Pranav Natu

A report presented in partial fulfillment of the course
MEEN 655- Design of Nonlinear Control Systems

for
Dr. Swaroop Darbha
Department of Mechanical Engineering
Texas A and M University
20 April 2020

Abstract

This report introduces a nonlinear control scheme for controlling the angular position of a permanent magnet synchronous motor (PMSM). The parameters like rotor inertia, friction, motor constant, torque load are not exactly known. The initial estimates of these parameters are known and an adaptive control law is designed such that the estimates of these parameters improves as time progresses and approaches the true value.

Introduction

Permanent magnet synchronous motors (PMSMs) are very popular and used in many applications. PMSM are used for high precision control of exoskeleton of robots and hence, precise control is very important. Conventional PID control which has weak robustness is incapable of dealing with nonlinear control, although it has the advantage of simple structure and easy implementation. PMSM is a multi variable and strongly coupled nonlinear system, thus having many factors in its control system, such as current coupling, external disturbance. Therefore it is desirable to come up with a control strategy to which provides robustness, good precision and easy implementation.
There are many cases in which the exact parameters of the process are not known. For example, the load torque on the motor, although constant, may vary from one process to another. In such a case it is difficult to change the control law every time. Similarly, a parameter like friction force is difficult to estimate beforehand. In such cases, we use adaptive control techniques to design the control law. This serves two purposes- the desired trajectory tracking is achieved and the controller also learns about the parameters by observing the system dynamics and provides a better estimate over time.
In this project, such an adaptive nonlinear controller was implemented on PMSM and the results were presented.

Contents

1 Notations and Known Values

δ : rotor angular position
w : rotor angular velocity
I_a, I_b : currents in the stator windings along d and q axis of the stator
u_a, u_b : control inputs

Known values
p=2 : number of poles
R=1 : winding resistance
L=1 : self inductance

Parameters not known exactly-
K : motor constant
F : friction force
J : rotor inertia
T : load torque

$\delta_{des} = sin(t)$: desired angular position to be tracked

2 Problem Definition

The problem statement for this project is divided into two parts. In the first part, the objective is to control the angular position of the Permanent Magnet Synchronous Motor(PMSM) to a desired trajectory. There are 2 control inputs in the current equations. In the second part of the problem statement, the constant parameters like resistance, inductance etc are not known exactly and only their initial estimates are known. The objective is to devise a control law such that the desired angular position is obtained and the parameters are updated to the true value. Following are the nonlinear equations of the PMSM.

$\dot{\delta} = w$

$\dot{w} = -\frac{K}{J} I_a sin(p\delta) + \frac{K}{J} I_b cos(p\delta) - \frac{F}{J} w - \frac{T}{J}$

$\dot{I}_a = -\frac{R}{L} I_a + \frac{K}{L} w sin(p\delta) + u_a/L$

$\dot{I}_b = -\frac{R}{L} I_b - \frac{K}{L} w cos(p\delta) + u_b/L$

3 All Parameters Known Case

3.1 Linearizing the Equations of PMSM

We have the following nonlinear equations for the permanent magnet synchronous motor

$$
\begin{aligned}
&\dot{\delta} = w \\
&\dot{w} = -\tfrac{K}{J} I_a sin(p\delta) + \tfrac{K}{J} I_b cos(p\delta) - \tfrac{F}{J} w - \tfrac{T}{J} \\
&\dot{I}_a = -\tfrac{R}{L} I_a + \tfrac{K}{L} w sin(p\delta) + u_a/L \\
&\dot{I}_b = -\tfrac{R}{L} I_b - \tfrac{K}{L} w cos(p\delta) + u_b/L
\end{aligned}
$$

Now we linearize the nonlinear equations. We use the following transform-

$$
\begin{bmatrix} I_c \\ I_d \end{bmatrix} = \begin{bmatrix} cos(p\delta) & sin(p\delta) \\ -sin(p\delta) & cos(p\delta) \end{bmatrix} \begin{bmatrix} I_a \\ I_b \end{bmatrix}
$$

$$
\begin{bmatrix} u_c \\ u_d \end{bmatrix} = \begin{bmatrix} cos(p\delta) & sin(p\delta) \\ -sin(p\delta) & cos(p\delta) \end{bmatrix} \begin{bmatrix} u_a \\ u_b \end{bmatrix}
$$

Therefore the resulting equations are-

$$
\begin{aligned}
&\dot{\delta} = w \\
&\dot{w} = -\tfrac{K}{J} I_d - \tfrac{F}{J} w - \tfrac{T}{J} \\
&\dot{I}_c = -\tfrac{R}{L} I_c + \tfrac{K}{L} w + p I_d + u_c/L \\
&\dot{I}_d = -\tfrac{R}{L} I_d - \tfrac{K}{L} w - p w I_c + u_d/L
\end{aligned}
$$

Now take

$$
\begin{aligned}
&v_c = -\tfrac{R}{L} I_c + \tfrac{K}{L} w + p I_d + u_c/L \\
&v_d = -\tfrac{R}{L} I_d - \tfrac{K}{L} w - p w I_c + u_d/L
\end{aligned}
$$

Therefore-

$$
\begin{aligned}
&\dot{\delta} = w \\
&\dot{w} = -\tfrac{K}{J} I_d - \tfrac{F}{J} w - \tfrac{T}{J} \\
&\dot{I}_d = v_d
\end{aligned}
$$

$\dot{I_c} = v_c$
is the linearized system of equations

It can be clearly seen that the first three equations and the last equation are decoupled. We can set $\dot{I_c}$ to whatever we want.
Thus, we set $\dot{I_c} = v_c = 0$
Thus we can take I_c is constant

3.2 Deriving error dynamics and virtual control

Let
$\delta_{des} = r_1$ and $I_{d,des} = r_2$

$e_1 = \delta - r_1$
$e_2 = w - \dot{r_1}$
$e_3 = I_d - r_2$

$\dot{e_1} = \dot{\delta} - \dot{r_1} = w - \dot{\delta} = e_2$

Now take the paramters as
$\theta_1 = K/J$
$\theta_2 = F/J$
$\theta_3 = T/J$

Therefore, we have the following error equations

$\dot{e_1} = e_2$
$\dot{e_2} = \theta_1 I_d - \theta_2 w - \theta_3$
$\dot{e_3} = v_d - r_2$

We will find $I_{d,des}$ from the Lyapunov function

$V = \frac{1}{2}(e_1^2 + e_2^2)$
$\dot{V} = e_1\dot{e_1} + e_2\dot{e_2}$
$= e_1 e_2 + e_2(\theta_1 I_d - \theta_2 w - \theta_3)$

Therefore,

$I_{d,des} = \frac{1}{\theta_1}(\theta_2 w + \theta_3 - e_1 - \lambda_2 e_2)$

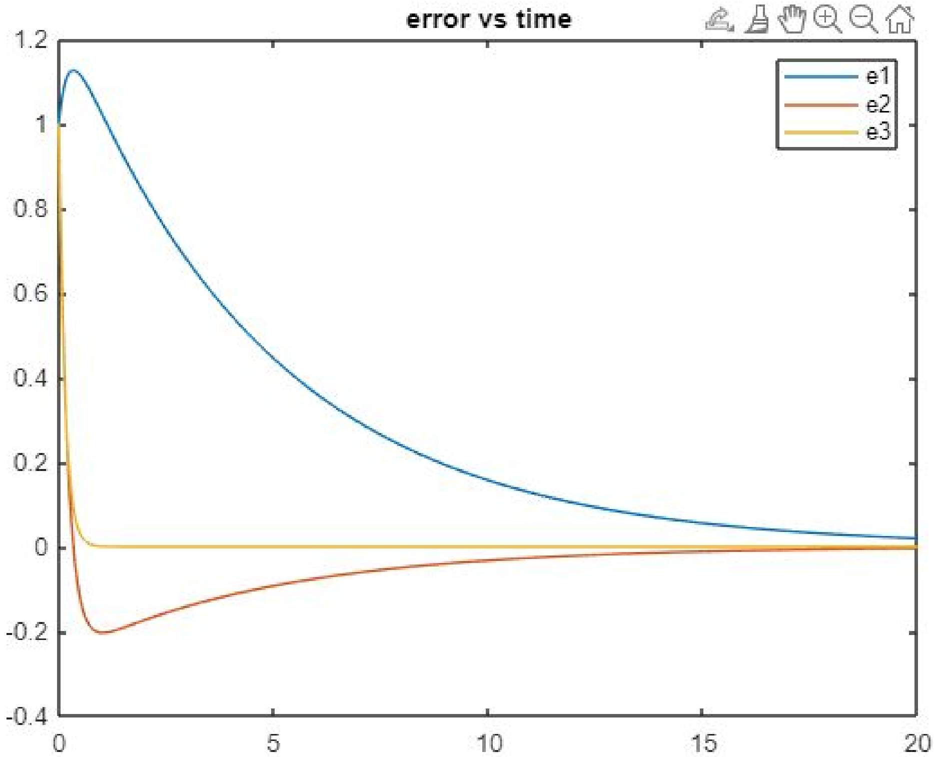
Now

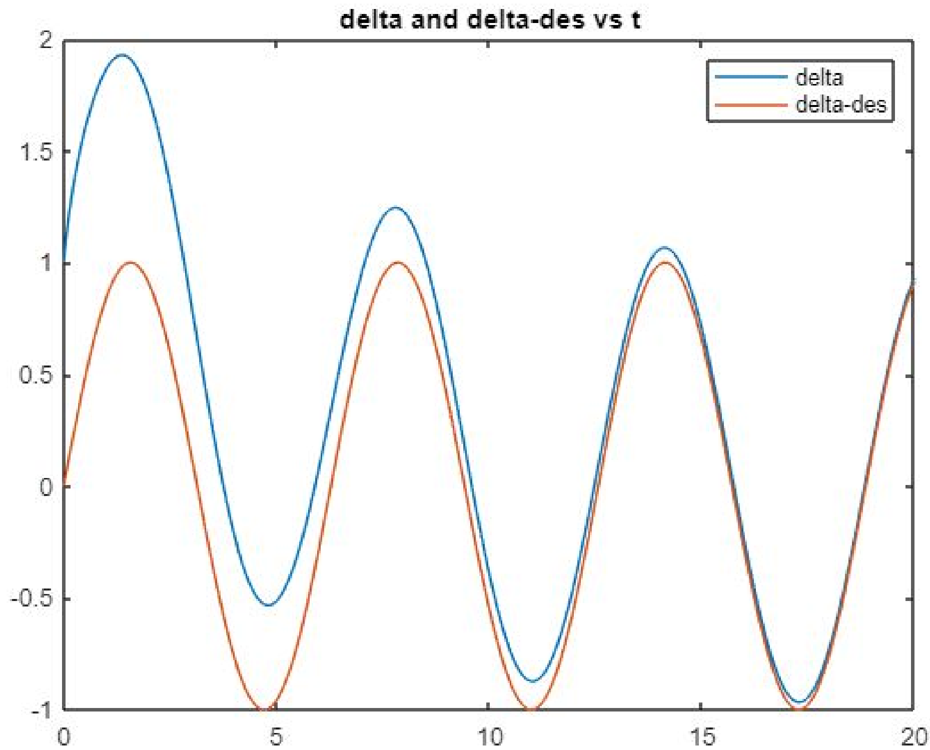
$V_3 = \frac{1}{2}(e_3^2)$
$\dot{V_3} = e_3 \dot{e_3}$
$= e_3(v_d - I_{d,des})$

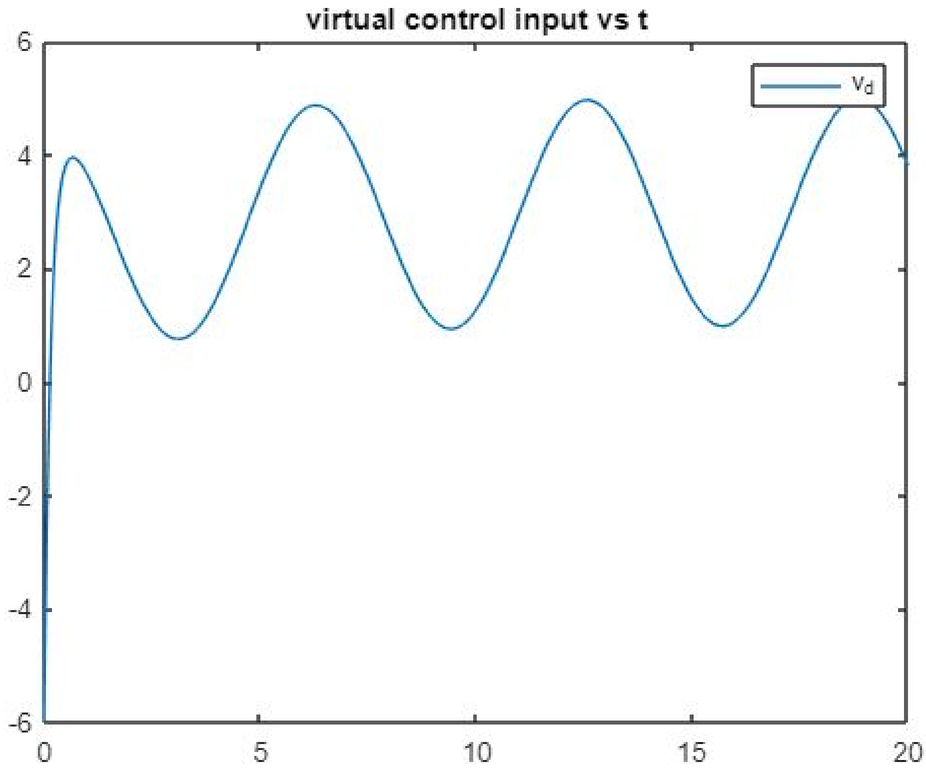
Therefore,
$v_{d,des} = I_{d,des} - \lambda_3 e_3$
This choice of v_d guarantees exponential stability to the system

The following are the results of the simulation in MATLAB with $\delta_{des} = sin(t)$

3.3 Graphs of results







virtual control input vs t
6
4
2
0
-2
-4
-6
0
5
10
15
20
v_d

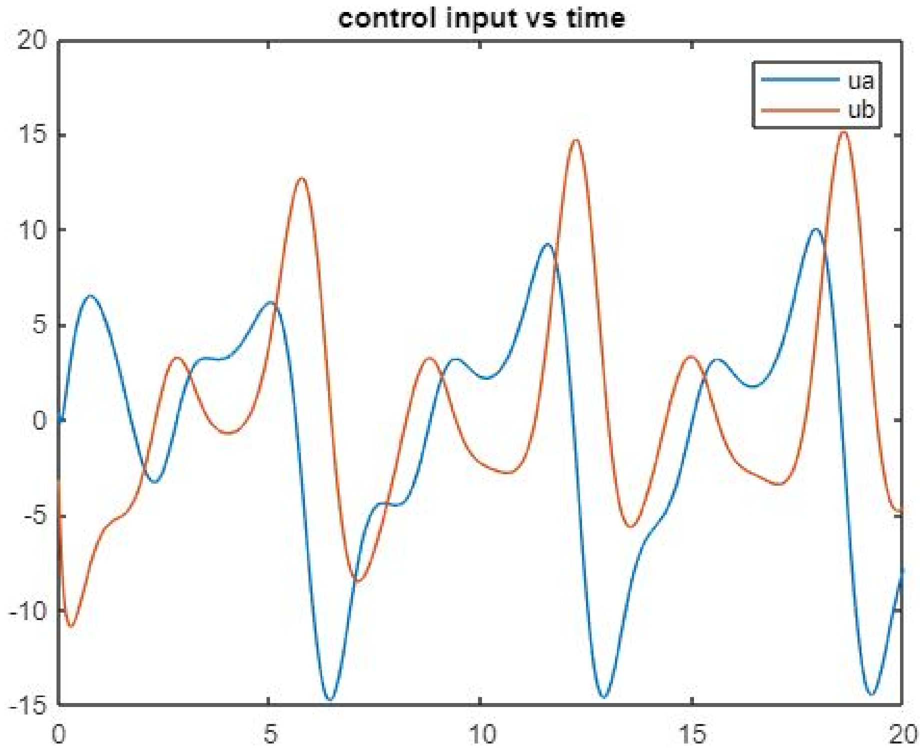
3.4 Deriving required control law

Now to convert the virtual control v_d and v_c to the initial control u_a and u_b

$u_c = v_c + \frac{R}{L}I_c - pwI_d$
$v_d = \frac{K}{L}w + \frac{R}{L}I_d + pwI_c$

$$\begin{bmatrix} u_a \\ u_b \end{bmatrix} = \begin{bmatrix} cos(p\delta) & -sin(p\delta) \\ sin(p\delta) & cos(p\delta) \end{bmatrix} \begin{bmatrix} u_c \\ u_d \end{bmatrix}$$

Therefore we get the following control law which will give us the desired trajectory



4 Parameter Uncertainty Case

In this section, we will deal with parameter uncertainty. Let us assume that the parameters are not exactly known and we only have an initial estimate of them. For simulation purposes, let us assume the true values of paramters as
$\theta_1 = 1$
$\theta_2 = 2$
$\theta_3 = 3$

and let us take the initial estimate as
$\widehat{\theta_1}(0) = 0.5$
$\widehat{\theta_2}(0) = 1.5$
$\widehat{\theta_3}(0) = 2.5$

Our control law will ensure all estimates approach the true value

4.1 Deriving error dynamics and virtual control

Let us redefine the errors
$e_1 = \delta - \delta_{des}$
$e_2 = w - w_{des}$
$e_3 = I_d - I_{d,des}$

In this part we will use back-stepping to find all desired values and eventually the virtual control.
$\dot{e_1} = \dot{\delta} - \dot{\delta_{des}}$
$\dot{e_1} = w - \dot{\delta_{des}}$

Therefore,
$w_{des} = \dot{\delta_{des}} - \lambda_1 e_1$
$\dot{e1} = -\lambda_1 e_1$

$\dot{e_2} = \dot{w} - \dot{w_{des}}$
$= \theta_1 I_d - \theta_2 w - \theta_3$

Let estimate of θ_1 be $\widehat{\theta_1}$ and error in estimate of θ_1 be $\widetilde{\theta_1}$

Thus,
$\widehat{\theta}_1 I_{d,des} = \widehat{\theta}_2 w + \widehat{\theta}_3 - \lambda_2 e_2$

We have,
$\widetilde{\theta}_1 = \widehat{\theta}_1 - \theta_1$
$\widetilde{\theta}_2 = \widehat{\theta}_2 - \theta_2$
$\widetilde{\theta}_3 = \widehat{\theta}_3 - \theta_3$

Thus,
$\dot{e_2} = \theta_1 I_{d,des} - \widehat{\theta}_1 I_{d,des} + \widehat{\theta}_1 I_{d,des} - \theta_2 w - \theta_3$
$\dot{e_2} = -\widetilde{\theta}_1 I_{d,des} + \widetilde{\theta}_2 w + \widetilde{\theta}_3 - \lambda_2 e_2$

$\dot{e_3} = \dot{I}_d - \dot{I_{d,des}}$
$\dot{e_3} = v_d - \dot{I_{d,des}}$
Thus, $v_d = \dot{I_{d,des}} - \lambda_3 e_3$
$\dot{e_3} = -\lambda_3 e_3$

4.2 Deriving adaptation law

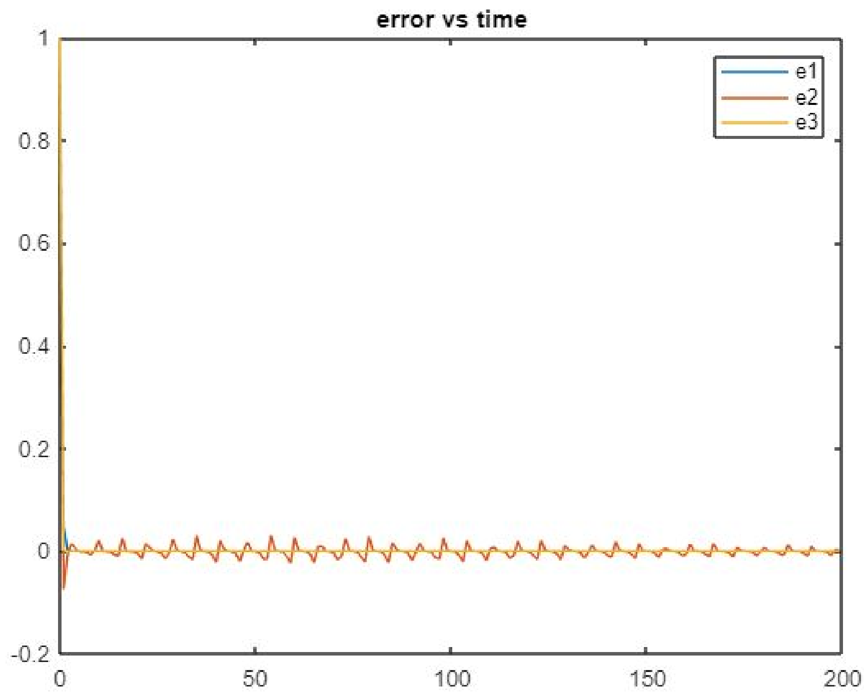
Take the Lyapunov function as
$V = \frac{1}{2}(e_1^2 + e_2^2 + e_3^2) + \frac{1}{2}(\gamma_1 \widetilde{\theta}_1^{\,2} + \gamma_2 \widetilde{\theta}_2^{\,2} + \gamma_3 \widetilde{\theta}_3^{\,2})$
where γ_1, γ_2 and γ_3 are the parameter adaptation gains

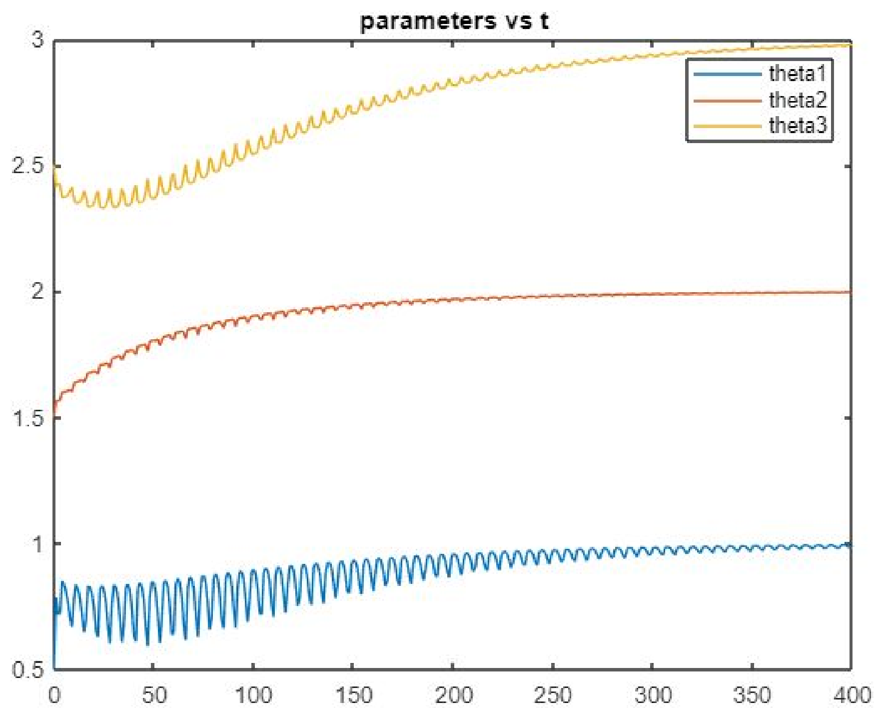
$\dot{V} = e_1 \dot{e_1} + e_2 \dot{e_2} + e_3 \dot{e_3} + \gamma_1 \widetilde{\theta}_1 \dot{\widetilde{\theta}}_1 + \gamma_2 \widetilde{\theta}_2 \dot{\widetilde{\theta}}_2 + \gamma_3 \widetilde{\theta}_3 \dot{\widetilde{\theta}}_3$
$\dot{V} = e_1 \dot{e_1} + e_2 \dot{e_2} + \gamma_1 \widetilde{\theta}_1 \dot{\widehat{\theta}}_1 + \gamma_2 \widetilde{\theta}_2 \dot{\widehat{\theta}}_2 + \gamma_3 \widetilde{\theta}_3 \dot{\widehat{\theta}}_3$
$= -\lambda_1 e_1^2 + e_2(-\widetilde{\theta}_1 I_{d,des} + \widetilde{\theta}_2 w + \widetilde{\theta}_3 - \lambda_2 e_2) + \gamma_1 \widetilde{\theta}_1 \dot{\widehat{\theta}}_1 + \gamma_2 \widetilde{\theta}_2 \dot{\widehat{\theta}}_2 + \gamma_3 \widetilde{\theta}_3 \dot{\widehat{\theta}}_3$

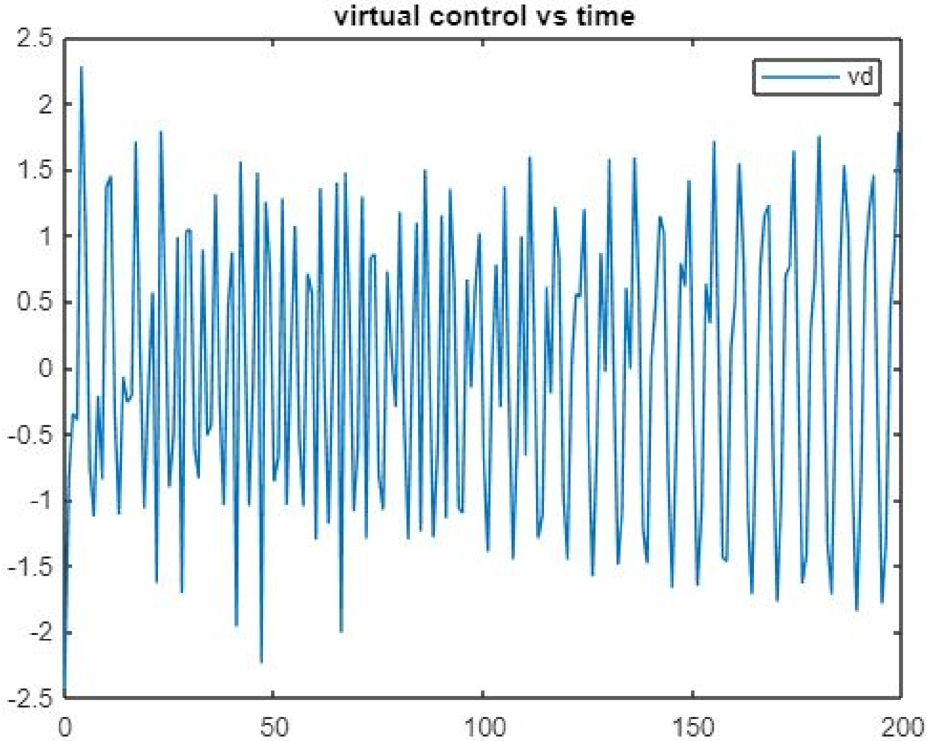
Therefore, we have the following parameter adaptation law-
$\dot{\widehat{\theta}}_1 = \frac{1}{\gamma_1} e_2 I_{d,des}$
$\dot{\widehat{\theta}}_2 = -\frac{1}{\gamma_2} e_2 w$
$\dot{\widehat{\theta}}_3 = -\frac{1}{\gamma_3} e_2$

4.3 Graphs of results

The following are the results of the simulation in MATLAB with $\delta_{des} = sin(t)$

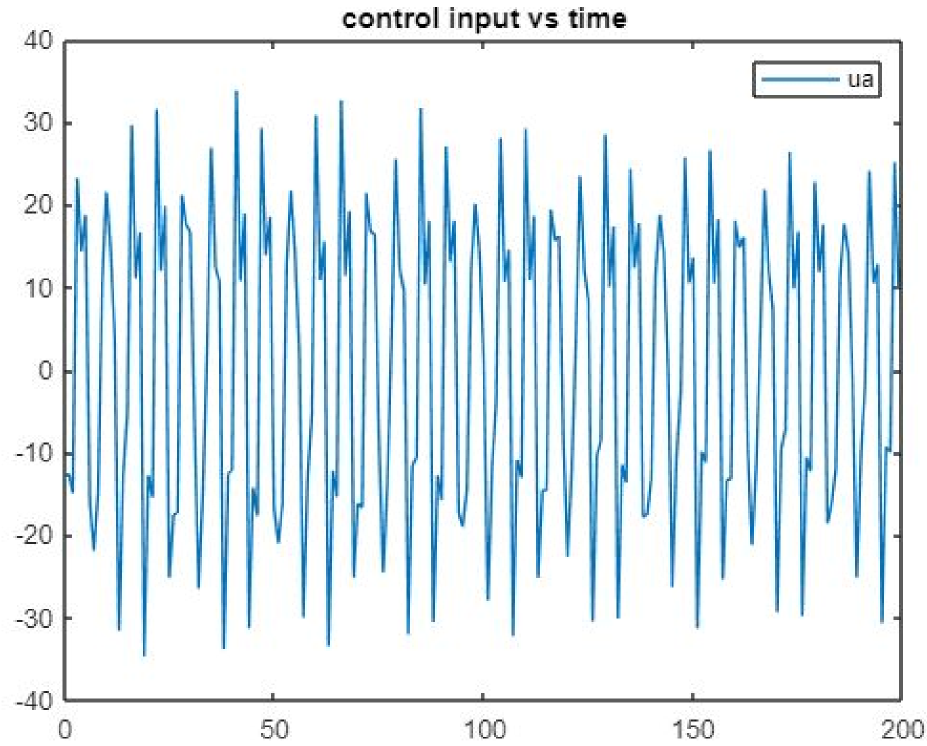


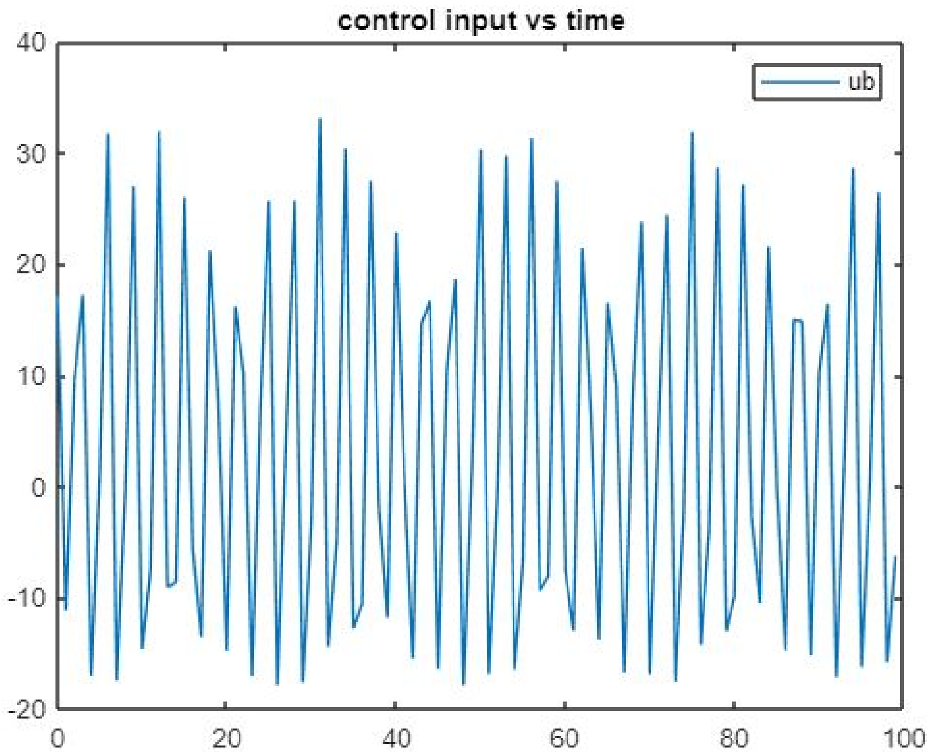



virtual control vs time
vd
2.5
2
1.5
1
0.5
0
-0.5
-1
-1.5
-2
-2.5
0
50
100
150
200

4.4 Deriving required control law

Following the same procedure as in previous section we can derive u_a and u_b which is the control input.





5 Conclusion

The control law to track the desired trajectory was derived for the case when all parameters and known are for the case when parameters are unknown.
It was found from the plots that the parameter adaptation law was such that parameters converged to the actual value. However, large number of time steps were required for convergence. The rate of convergence could be increased by adjusting the parameter adaptation gains to a certain value. But after the gains were increased until a certain value, the simulation took a long time to execute.
Thus, the control law is effective in a range of parameter adaptation gains and a long time is required for parameter convergence.

6 References

[1]J. Solsona, M.I. Valla, c. Muravchik "Nonlinear Control of a Permanent Magnet Synchronous Motor with Disturbance Torque Estimation"
[2] "Nonlinear Control Design: Geometric, Adaptive and Robust", R.Marino and P. Tomei.
[3]Texas A and M University MEEN 655 Class Notes and Video Lectures