Chapter 5

CONTROLLER DESIGNS AND SIMULATION RESULTS

In this chapter, we develop the control system, under a desired performance for balancing and rotating the MIP.
Therefore, in this chapter, we aim to show how the system can be controlled using

- Linear state-feedback controllers.
  In the first part, the LQR and Pole-placement controllers are designed for balancing and rotating the MIP by using the linearized dynamics model developed in the previous chapter.

- LPV controller.
  In the second part, An $\mathcal{H}_\infty$ controller is designed for balancing an MIP by using linear parameter varying control technique with a full block multiplier.

According to the actuator specifications and the MIP parameters in chapter 3, we can setup the design requirements as below:

- Control input is ± 24 V.

- Mass of the pendulum can vary from 6 Kg to 15 Kg.

- Length of the pendulum can change from 0.14 m to 0.16 m.

- Settling time of the pendulum angle should be less than 3 seconds.

- Pendulum angle should be never greater than 0.05 radians from the vertical.

5.1 Linear State-Feedback Controller Design

Equation (4.45) shows that there are two inputs to our system. The voltage V_{aL} and V_{aR} are applied to the left-hand-side and right-hand-side motors can be governed by the control system. In order to impose the desired dynamics on the system, we would like to control the rotation around the z axis independently for the rotation around the y axis. This implies that we should have a controller producing an output signal corresponding to a voltage around the vertical axis and another with an output voltage around the lateral axis. To apply these input voltages on the system, we need a decoupling unit that transforms V_{B} and V_{R} into the motor input voltage V_{aL} and V_{aR}.

Such a decoupling unit is typically be in the form of

$$\begin{pmatrix} V_{\mathrm{aL}} \\ V_{\mathrm{aR}} \end{pmatrix} = \begin{pmatrix} D_{11} & D_{12} \\ D_{21} & D_{22} \end{pmatrix} \begin{pmatrix} V_{\mathrm{B}} \\ V_{\mathrm{R}} \end{pmatrix} \tag{5.1}$$

The resulting equation when substituting equation (5.1) into equation (4.45) needs to be in the following form in order to avoid cross coupling.

$$\begin{pmatrix} \dot{x} \\ \ddot{x} \\ \dot{\phi}_{\mathrm{P}} \\ \ddot{\phi}_{\mathrm{P}} \\ \dot{\delta} \\ \ddot{\delta} \end{pmatrix} = \begin{pmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & \frac{2\eta_{\mathrm{g}}\eta_{\mathrm{m}}K_{\mathrm{m}}K_{\mathrm{g}}^2K_{\mathrm{e}}\left(M_{\mathrm{P}}Lr-J_{\mathrm{P}}-M_{\mathrm{P}}L^2\right)}{R_{\mathrm{a}}r^2\alpha} & \frac{(M_{\mathrm{P}}L)^2g}{\alpha} & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & \frac{2\eta_{\mathrm{g}}\eta_{\mathrm{m}}K_{\mathrm{m}}K_{\mathrm{g}}^2K_{\mathrm{e}}(r\beta-M_{\mathrm{P}}L)}{R_{\mathrm{a}}r^2\alpha} & \frac{M_{\mathrm{P}}gL\beta}{\alpha} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} x \\ \dot{x} \\ \phi_{\mathrm{P}} \\ \dot{\phi}_{\mathrm{P}} \\ \delta \\ \dot{\delta} \end{pmatrix} + \begin{pmatrix} 0 & 0 \\ \frac{\eta_{\mathrm{g}}\eta_{\mathrm{m}}K_{\mathrm{m}}K_{\mathrm{g}}\left(-M_{\mathrm{P}}Lr+J_{\mathrm{P}}+M_{\mathrm{P}}L^2\right)}{R_{\mathrm{a}}r\alpha} & 0 \\ 0 & 0 \\ \frac{\eta_{\mathrm{g}}\eta_{\mathrm{m}}K_{\mathrm{m}}K_{\mathrm{g}}(M_{\mathrm{P}}L-r\beta)}{R_{\mathrm{a}}r\alpha} & 0 \\ 0 & 0 \\ 0 & \frac{\eta_{\mathrm{g}}\eta_{\mathrm{m}}K_{\mathrm{m}}K_{\mathrm{g}}D}{2J_{\mathrm{P}\delta}R_{\mathrm{a}}r^2} \end{pmatrix} \begin{pmatrix} V_{\mathrm{B}} \\ V_{\mathrm{R}} \end{pmatrix} \tag{5.2}$$

We can find cross-coupling by

$$\begin{pmatrix} V_{\mathrm{aL}} \\ V_{\mathrm{aR}} \end{pmatrix} = \begin{pmatrix} 0.5 & 0.5 \\ 0.5 & -0.5 \end{pmatrix} \begin{pmatrix} V_{\mathrm{B}} \\ V_{\mathrm{R}} \end{pmatrix} \tag{5.3}$$

Figure 5.1 shows the decoupling between the two subsystems that will be used for control input of the MIP. The state space equations for the vehicle can now be written as two independent subsystems.

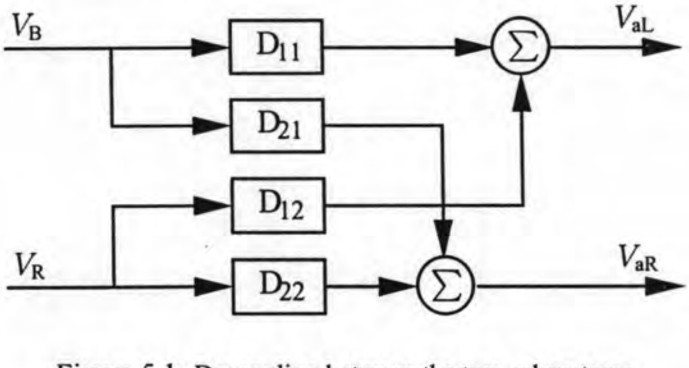

Figure 5.1: Decoupling between the two subsystems.

1. A subsystem for balancing the MIP.

$$\begin{pmatrix} \dot{x} \\ \ddot{x} \\ \dot{\phi}_{\mathrm{P}} \\ \ddot{\phi}_{\mathrm{P}} \end{pmatrix} = \begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & \frac{2\eta_{\mathrm{g}}\eta_{\mathrm{m}}K_{\mathrm{m}}K_{\mathrm{g}}^2K_{\mathrm{e}}\left(M_{\mathrm{P}}Lr-J_{\mathrm{P}}-M_{\mathrm{P}}L^2\right)}{R_{\mathrm{a}}r^2\alpha} & \frac{(M_{\mathrm{P}}L)^2g}{\alpha} & 0 \\ 0 & 0 & 0 & 1 \\ 0 & \frac{2\eta_{\mathrm{g}}\eta_{\mathrm{m}}K_{\mathrm{m}}K_{\mathrm{g}}^2K_{\mathrm{e}}(r\beta-M_{\mathrm{P}}L)}{R_{\mathrm{a}}r^2\alpha} & \frac{M_{\mathrm{P}}gL\beta}{\alpha} & 0 \end{pmatrix} \begin{pmatrix} x \\ \dot{x} \\ \phi_{\mathrm{P}} \\ \dot{\phi}_{\mathrm{P}} \end{pmatrix} + \begin{pmatrix} 0 \\ \frac{\eta_{\mathrm{g}}\eta_{\mathrm{m}}K_{\mathrm{m}}K_{\mathrm{g}}\left(-M_{\mathrm{P}}Lr+J_{\mathrm{P}}+M_{\mathrm{P}}L^2\right)}{R_{\mathrm{a}}r\alpha} \\ 0 \\ \frac{\eta_{\mathrm{g}}\eta_{\mathrm{m}}K_{\mathrm{m}}K_{\mathrm{g}}(M_{\mathrm{P}}L-r\beta)}{R_{\mathrm{a}}r\alpha} \\ 0 \end{pmatrix} V_{\mathrm{B}} \tag{5.4}$$

2. A subsystem for rotating the MIP.

$$\begin{pmatrix} \dot{\delta} \\ \ddot{\delta} \end{pmatrix} = \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix} \begin{pmatrix} \delta \\ \dot{\delta} \end{pmatrix} + \begin{pmatrix} 0 \\ \frac{\eta_{\mathrm{g}}\eta_{\mathrm{m}}K_{\mathrm{m}}K_{\mathrm{g}}D}{2.J_{\mathrm{P}\delta}R_{\mathrm{a}}r^2} \end{pmatrix} V_{\mathrm{R}} \tag{5.5}$$

We are now able to design an independent controller for each of those subsystems with the possibility of assigning different dynamics to each of them.

From equation (5.5), we can conclude that for the rotation of the MIP, we need only the information from $\dot{\delta}$. Figure 5.2 shows the block diagram of state-feedback used for balancing and rotating the MIP, where K_{B} is the feed-back gain for balancing and K_{R} is the feed-back gain for rotating.

5.1.1 Control Problem for Balancing the MIP

The balancing robot system is an excellent test-bed for control theory because it exhibits nonlinear and unstable system dynamics. Control objectives for these systems are always challenging as the full state of the system is not often fully measurable.

Because the system is inherently unstable, an impulse input applied to the open loop system will cause the tilt angle and position of the robot to rise unboundedly. This results in the robot falling over as the tilting range of the robot is limited to 30 degree, that is 0.52 radians on each side. Figure 5.3 shows the simulation when an impulse input is applied to the uncontrolled system. Plotting the Pole-Zero Map (Figure 5.4) of the system verifies that the system is unstable as there is a pole on the right-half plane of the plot. Ideally, all poles should be on the left-half plane of the plot for the system to be stable.

The poles of the system are located at $0, 4.5183, -4.5167, -9.6445$.

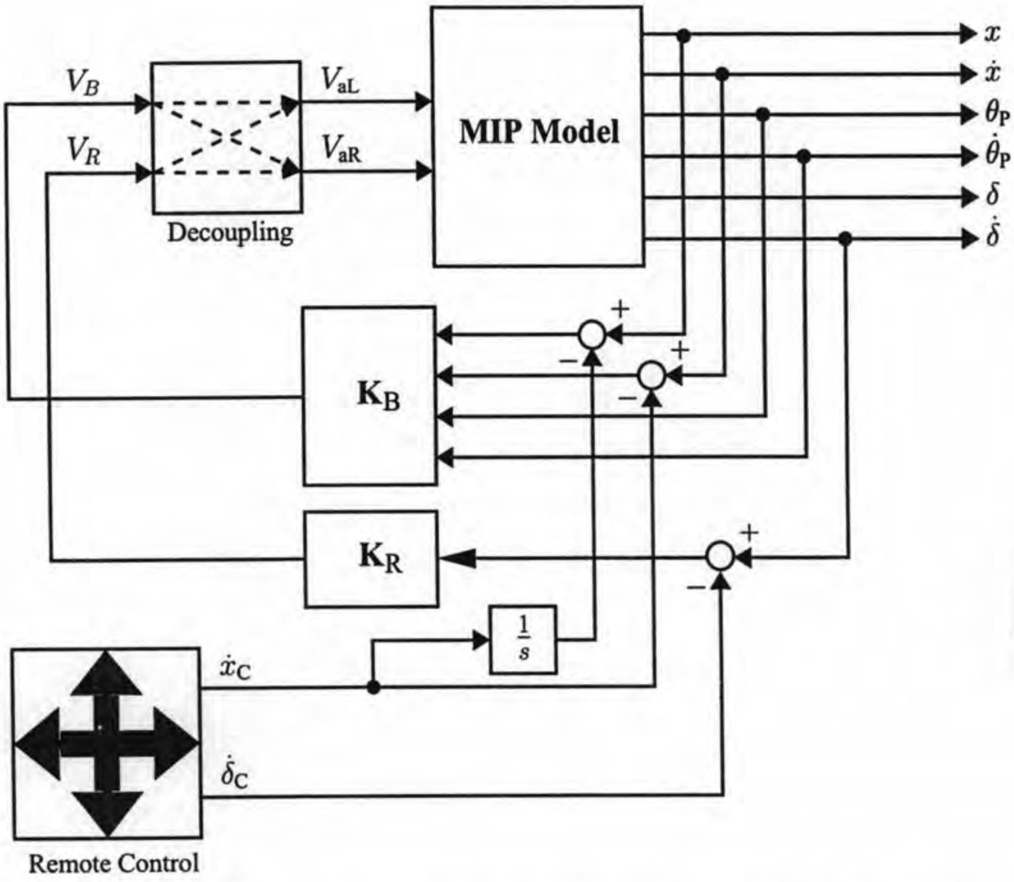

Figure 5.2: Block diagram for balancing and rotating using linear state-feedback controller.

5.1.2 Linear Quadratic Regulator Controller Design for Balancing the MIP

The Q matrix assumes the form of

$$Q = \begin{pmatrix} a & 0 & 0 & 0 \\ 0 & b & 0 & 0 \\ 0 & 0 & c & 0 \\ 0 & 0 & 0 & d \end{pmatrix}$$

where the values of $a = 100$, $b = 0$, $c = 100$, $d = 0$ are the default values of the weighting for the respective states x, $\dot{x}$, ϕ_P, $\dot{\phi}_P$ while the weighting matrix $R = 1$ is a scalar value as there is only one control input to the system. The values in the Q matrix are adjusted according to the required response of the system, a higher value of the weighings indicates the importance of the states compared to others.

The main aim of the control system is to make all the states of the system converge to zero at the shortest time possible. It will be impossible to achieve that goal as an infinite control force is non-existent. Therefore some compromise on system response has to be made.

Figure 5.5, Figure 5.6 and Figure 5.7 show that an increase in weighting for $R = 1000$

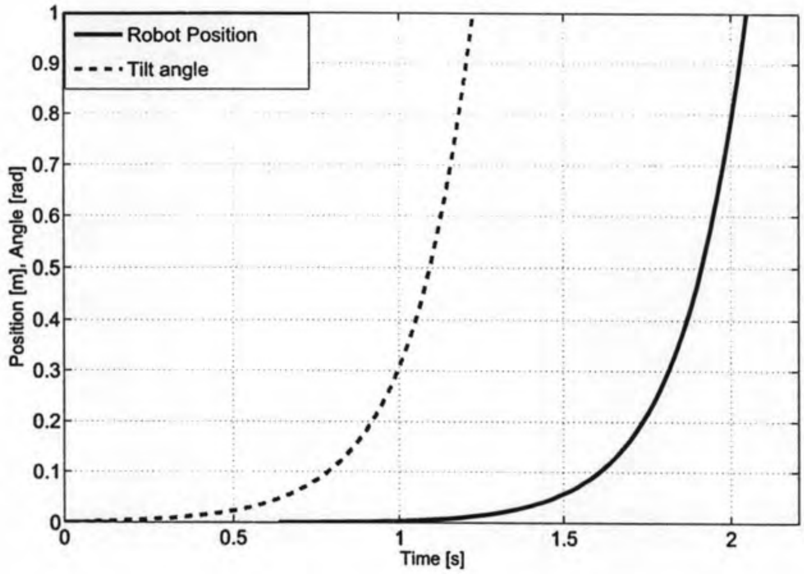

Figure 5.3: Open loop impulse response of the system.

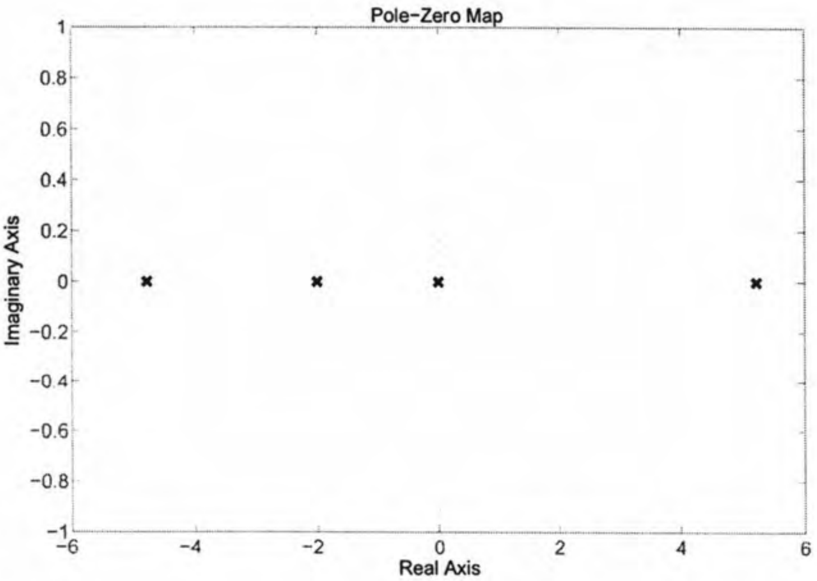

Figure 5.4: Pole-Zero map of the open loop system.

indicates less motor control which is used to balance the system. This results in a low gain value for linear position x, and linear velocity $\dot{x}$ which cause the robot to continue moving in order to balance the system. Figure 5.8, Figure 5.9 and Figure 5.10 show that an increase in weighting $a = 5000000$ for state x results in a high gain which reduces the settling time of

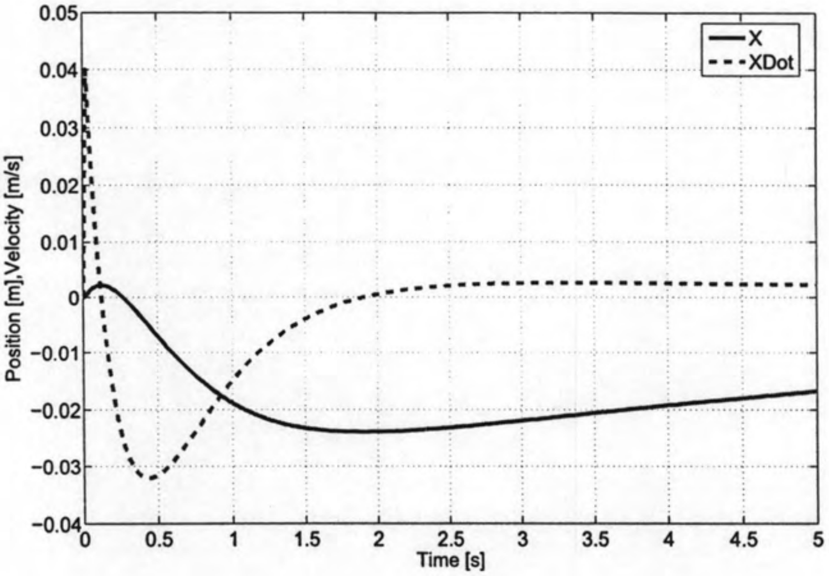

Figure 5.5: Response of position and velocity of the robot with a high R weighting.

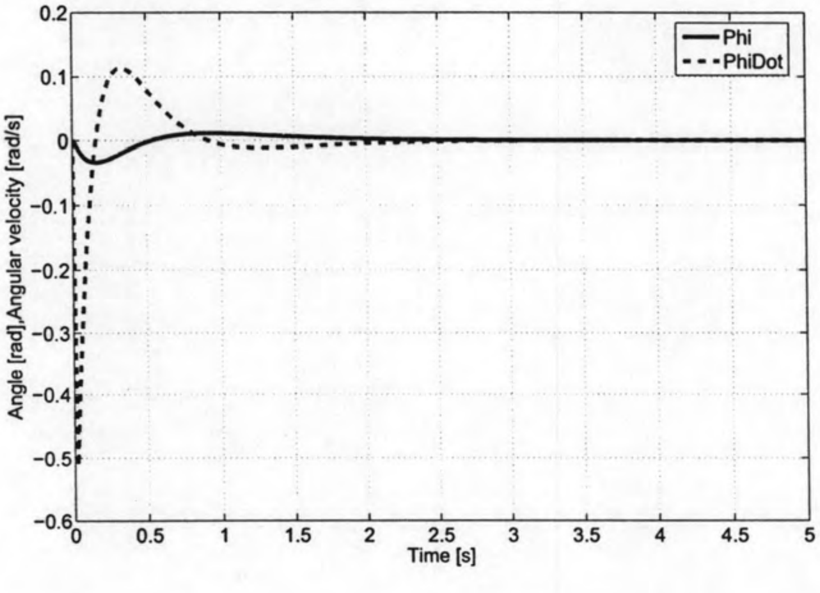

Figure 5.6: Response of angle and angular velocity of the pendulum with a high R weighting.

the robot's position. Unfortunately, the motors will not be able to match the desired response because the required torque exceeds the maximum torque of the motor.

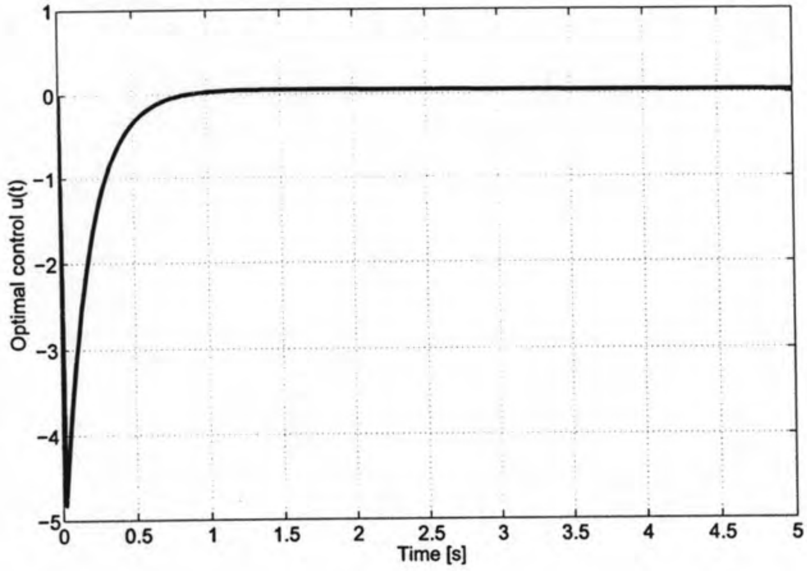

Figure 5.7: Response of optimal control input with a high R weighting.

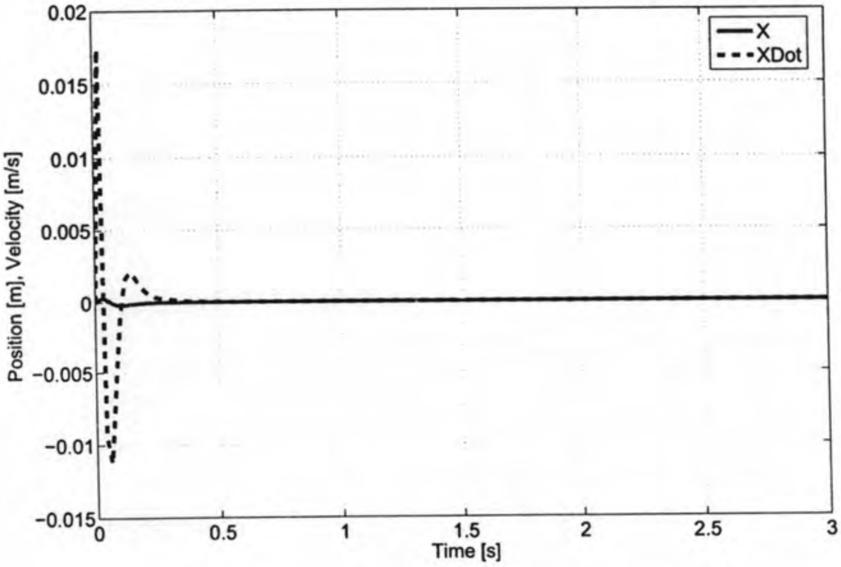

Figure 5.8: Response of position and velocity of the robot with a high position weighting.

Figure 5.11, Figure 5.12 and Figure 5.13 show that an increase in weighting $c = 100000$ for tilt angle results in a high gain for the states. This will compromise the settling time and response of other states.

Since the elements of matrices Q and R are determined arbitrarily, several simulation

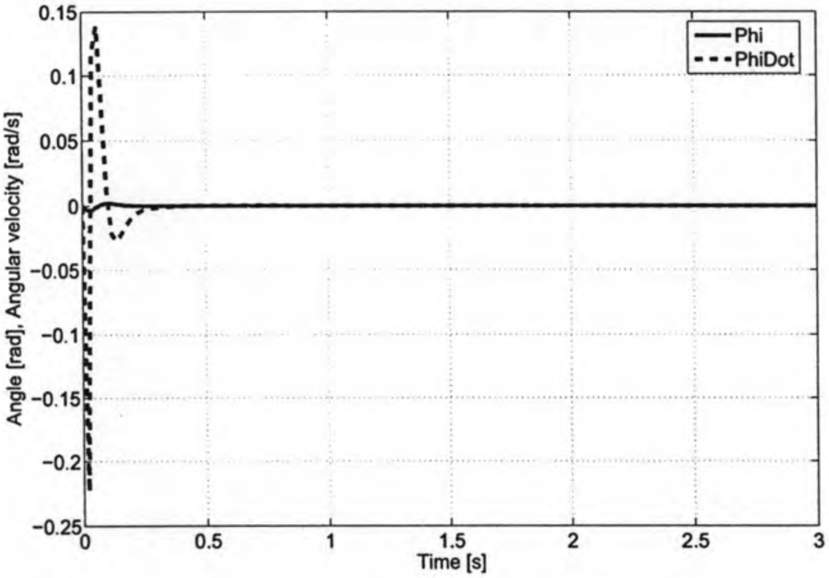

Figure 5.9: Response of angle and angular velocity of the pendulum with a high position weighting.

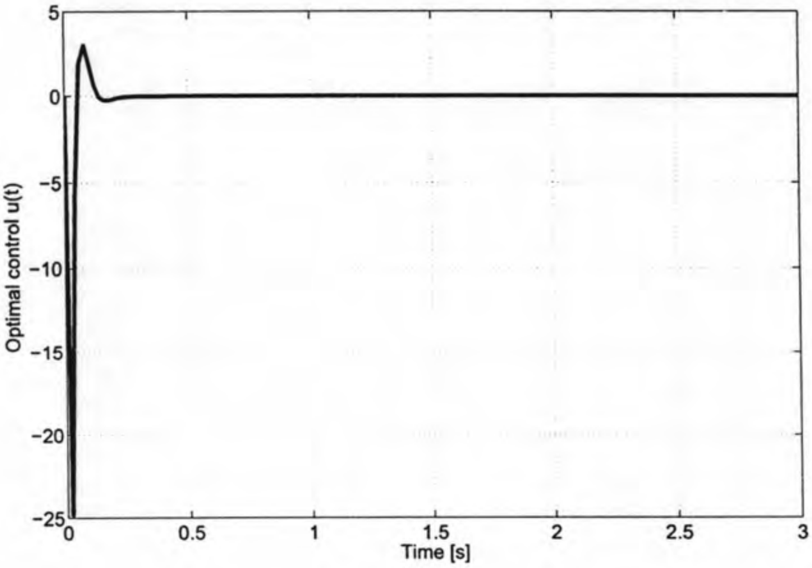

Figure 5.10: Response of optimal control input with a high position weighting.

and implementation trials are conducted to obtain the desired response for the balancing robot system. The most acceptable responses for the system with the weighting $a = 5000$, $b = 1000$ and $R = 1$ are shown in Figure 5.14, Figure 5.15 and Figure 5.16. The weighting of the angular velocity and linear velocity are left unchanged as the response of the states are

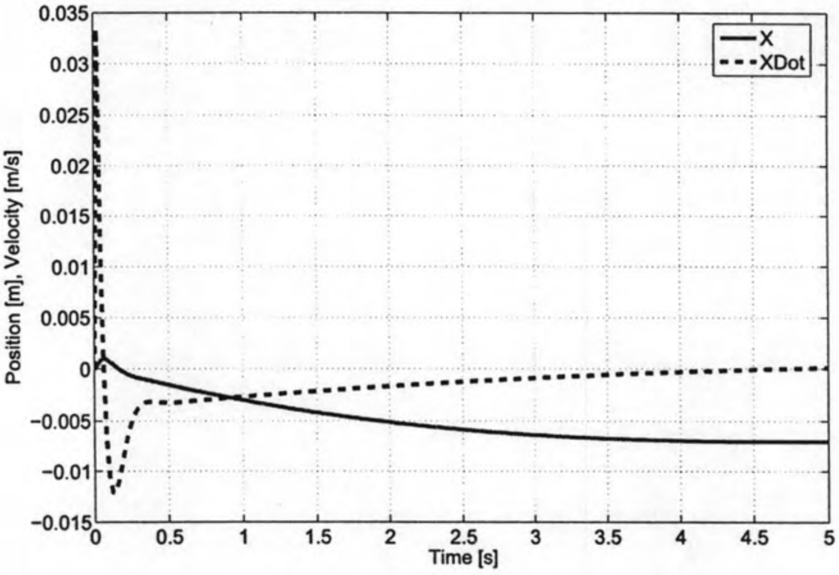

Figure 5.11: Response of position and velocity of the robot with a high tilt angle gain weighting.

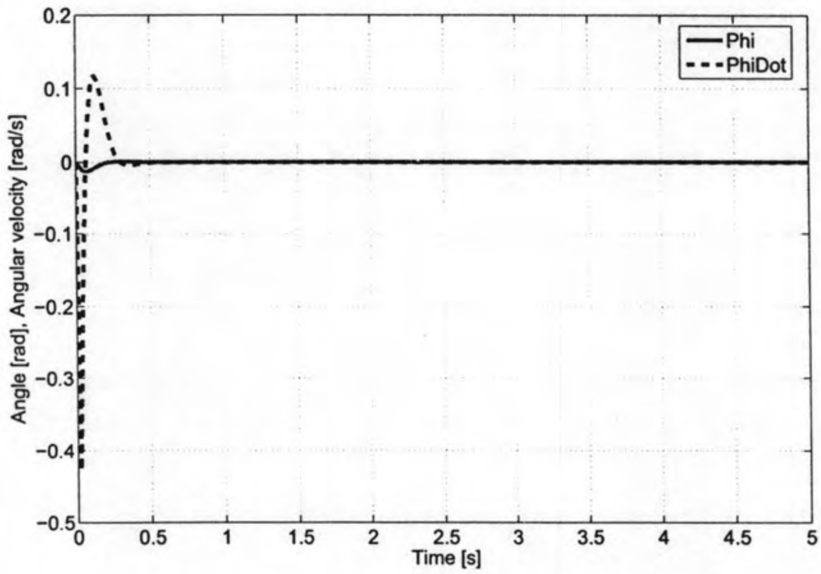

Figure 5.12: Response of angle and angular velocity of the pendulum with a high tilt angle gain weighting.

dependent on the other two states.

Poles of the system are now placed at $-1.7461, \quad -4.5024, \quad -4.5326, \quad -9.4852$.

Figure 5.17, Figure 5.18 and Figure 5.19 show the step responses of the system. From these

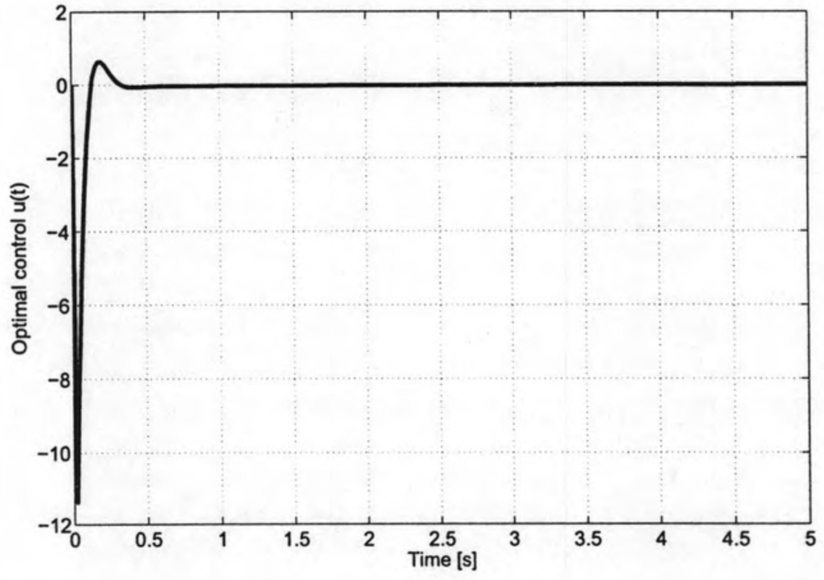

Figure 5.13: Response of optimal control input with a high tilt angle gain weighting.

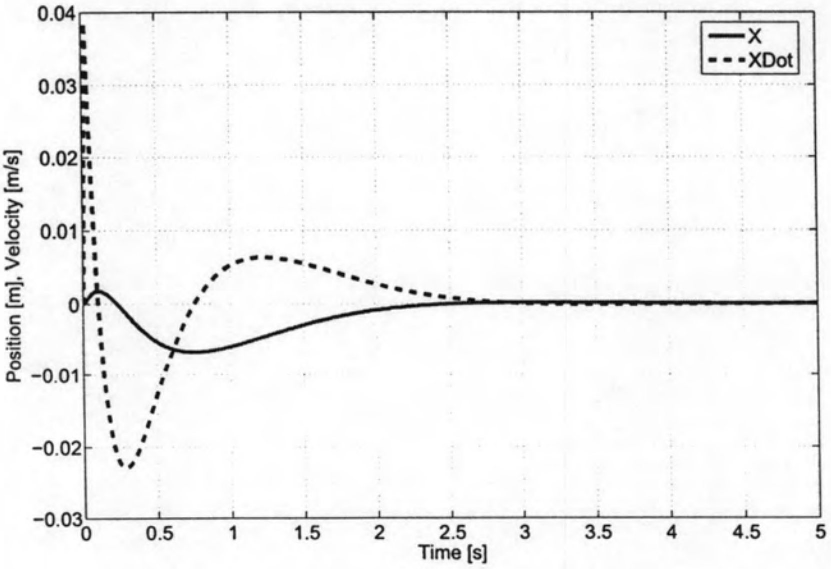

Figure 5.14: Response of position and velocity of the robot with appropriate weighting.

figures we can see that when the robot moves from the origin to the other place, the pendulum is still balanced.

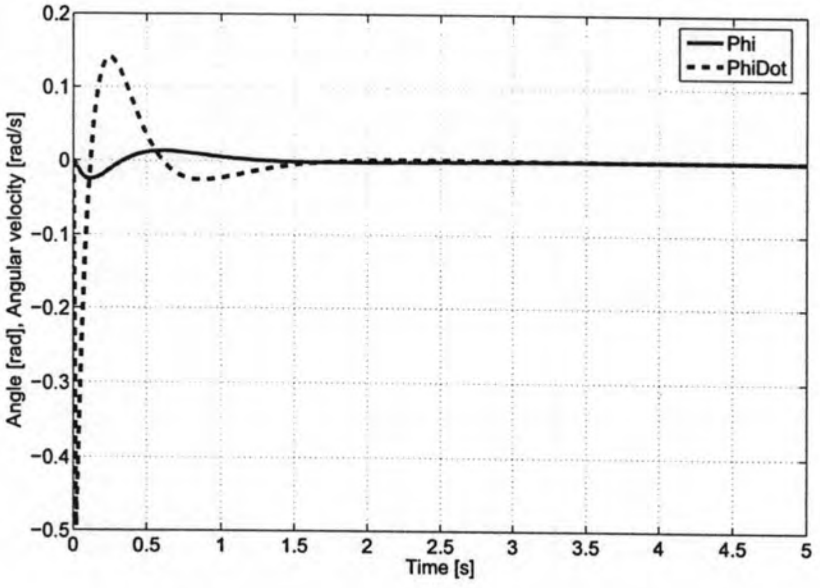

Figure 5.15: Response of angle and angular velocity of the pendulum with appropriate weighting.

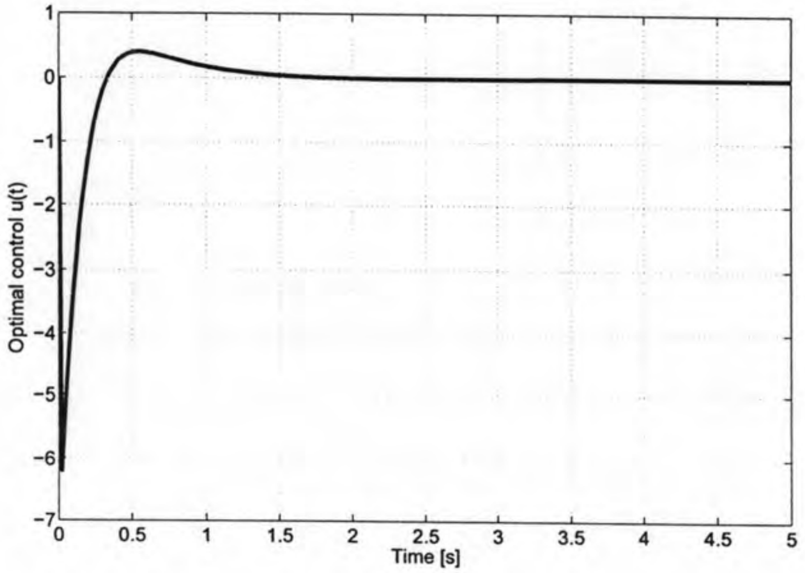

Figure 5.16: Response of optimal control input with appropriate weighting.

5.1.3 Pole Placement Controller Design for Balancing the MIP

First, we try to place poles further away from the imaginary axis and this is illustrated in Figure 5.20, Figure 5.21 and Figure 5.22. The control input is required for this pole configuration peaks to be around 19.5 V.

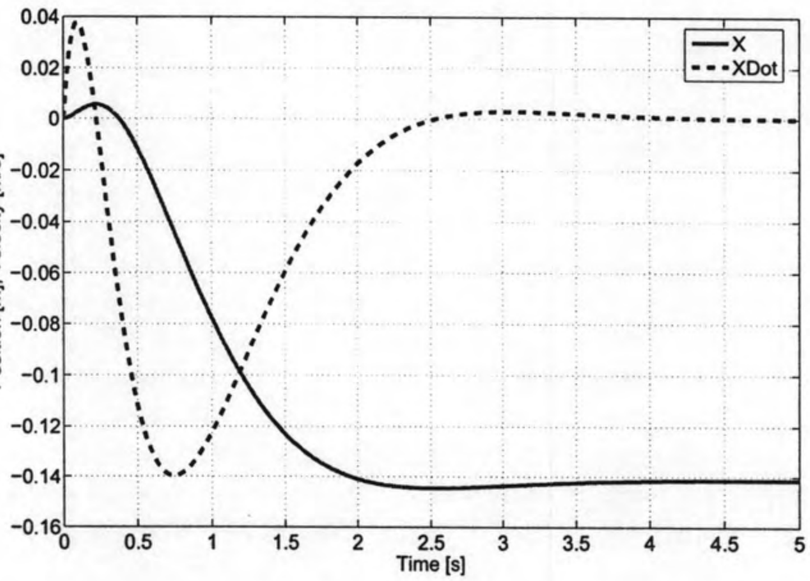

Figure 5.17: Response of position and velocity of the MIP with appropriate weighting.

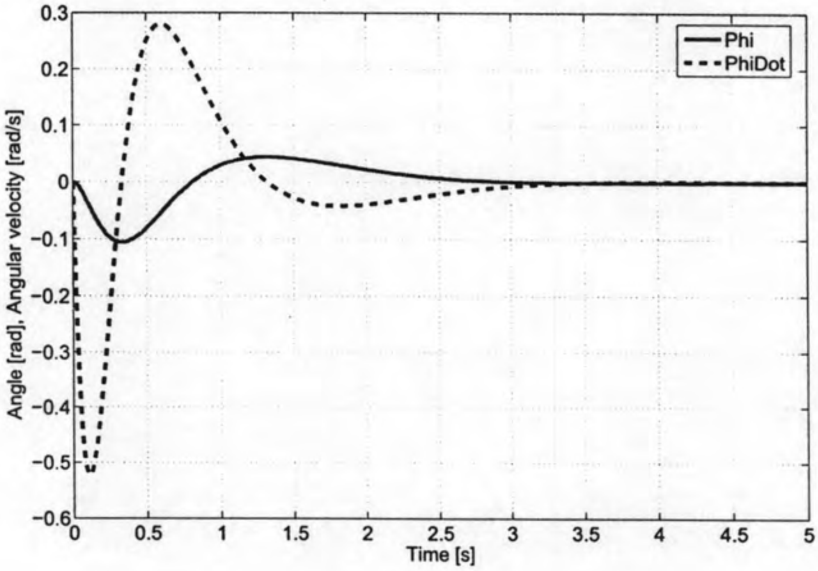

Figure 5.18: Response of angle and angular velocity of the pendulum with appropriate weighting.

The transient response of a system is the response of a system which decays as time increases. Since the purpose of control systems is to provide a desired response, the transient response of control systems must be adjusted until it is satisfied. In Pole-placement controllers, the transient response requirement of the system can be achieved by placing a

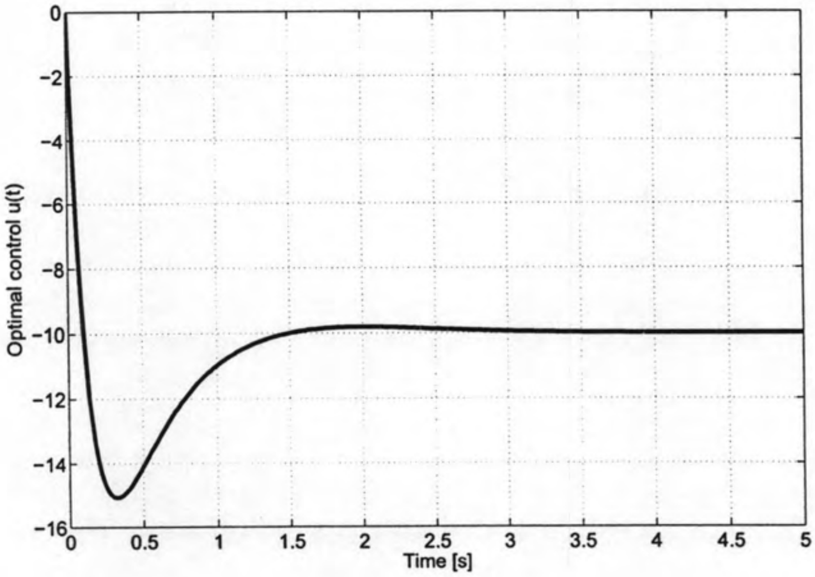

Figure 5.19: Response of optimal control input with appropriate weighting.

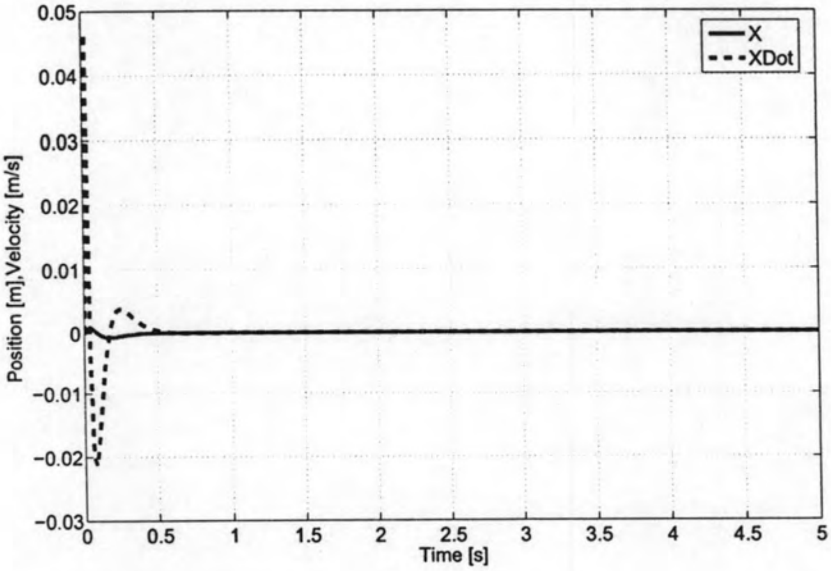

Figure 5.20: Response of position and velocity of the MIP with poles placed too far into the LHP.

pair of complex conjugate poles in the system.

Figure 5.23, Figure 5.24 and Figure 5.25 show the results of the simulation with the poles of the system placed at $-1.2+2j,\ -1.2-2j,\ -6+5j,\ -6-5j$

The combination of complex conjugate poles gives a good transient response to the system

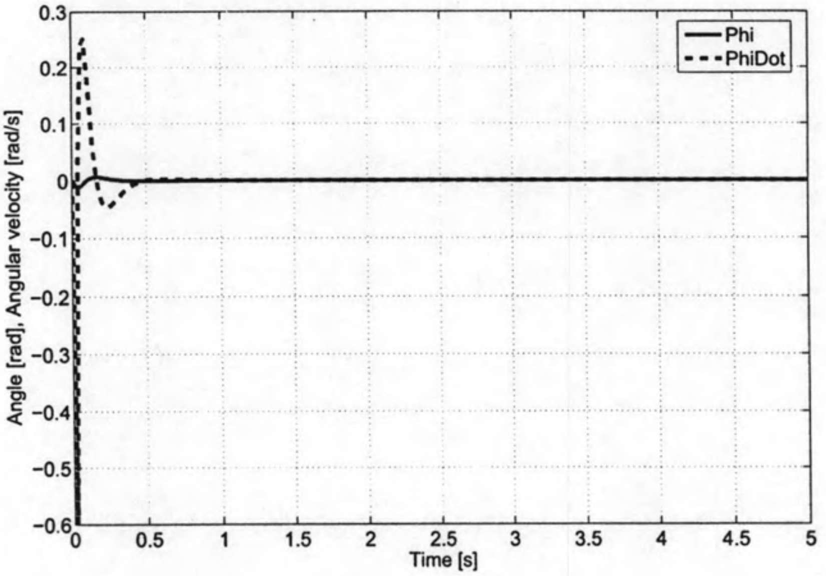

Figure 5.21: Response of angle and angular velocity of the pendulum with poles placed too far into the LHP.

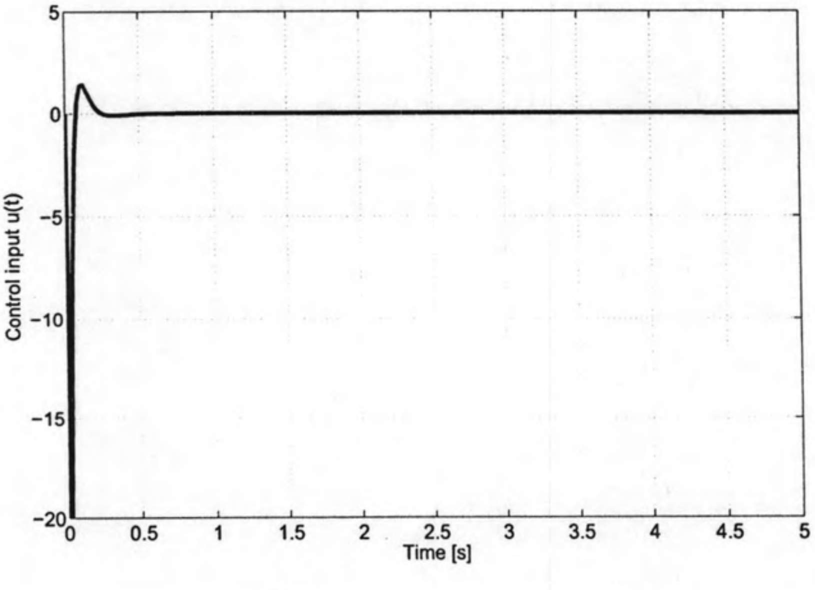

Figure 5.22: Response of control input with poles placed too far into the LHP.

and poles further away from the imaginary axis gives a fast response.

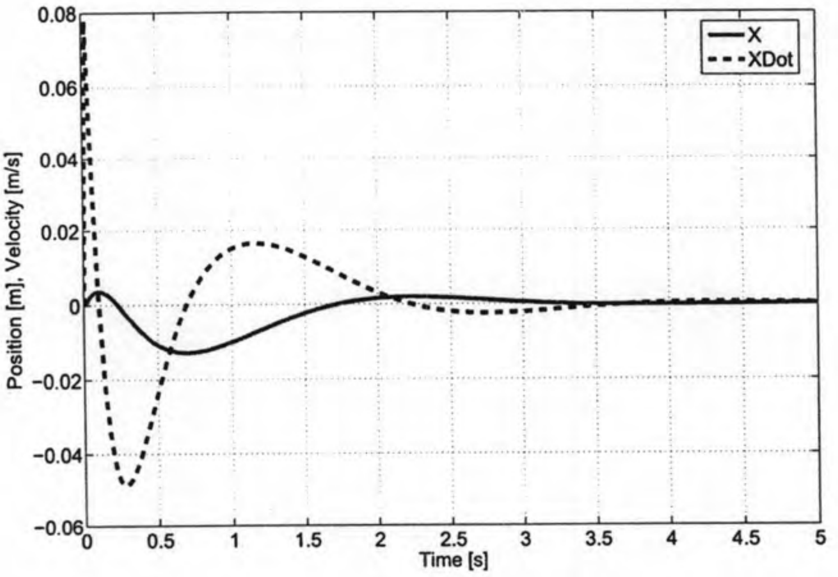

Figure 5.23: Response of position and velocity of the MIP by placing a pair of complex conjugate poles in the system.

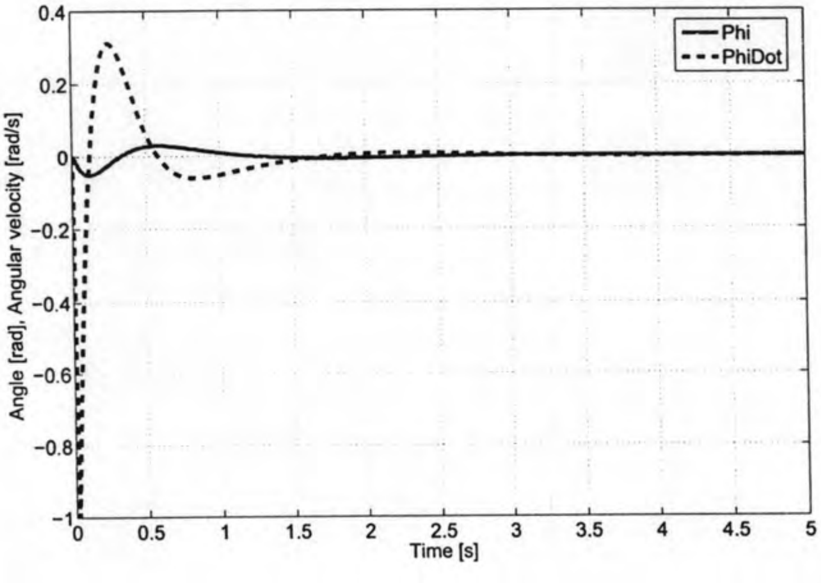

Figure 5.24: Response of angle and angular velocity of the pendulum by placing a pair of complex conjugate poles in the system.

5.1.4 Linear Quadratic Regulator Controller Design for Rotating the MIP

The Q matrix in this case is assumed the form of

$$Q = \begin{pmatrix} m & 0 \\ 0 & n \end{pmatrix}$$

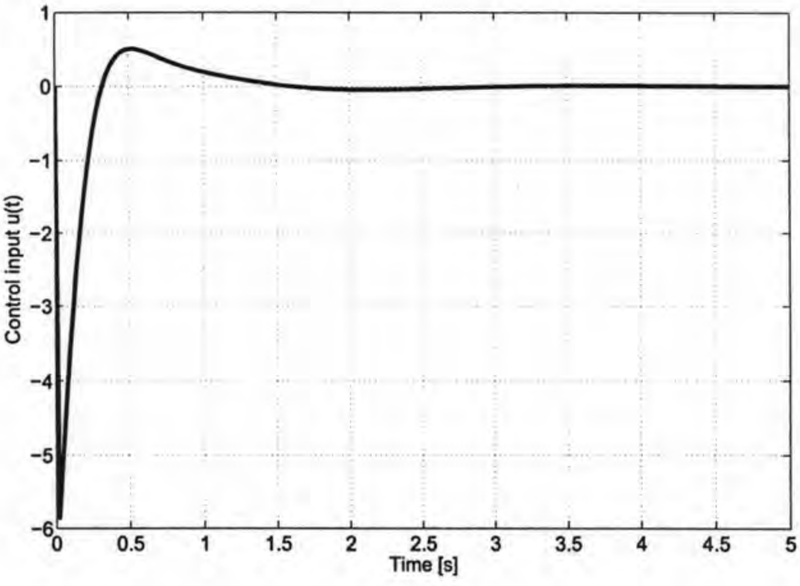

Figure 5.25: Response of input by placing a pair of complex conjugate poles in the system.

where the values of m, n, are the weighting of the respective states δ, $\dot{\delta}$ while the weighting matrix R is a scalar value as there is only one control input to the system. We know from the equation (5.5) that we do not need information about δ, so we can set the weighting m as small as possible. The main aim of the rotation control system is to make the states δ of the system converge to some value at the shortest time possible.

In this case, we set $m = 0.01$, $n = 1000$ and $R = 1$, and find the gain $K_{\text{R}} = [0.01 \quad 31.63]$. The simulation results are shown in Figure 5.26.

We can see that the requirement for the control input is 24V which is the limit voltage for the motor. Figure 5.27 shows the control input voltage requirement for rotating the MIP.

5.1.5 Pole Placement Controller Design for Rotating the MIP

The system for rotation has all the poles at the origin. Thus we try to rearrange the pole for yaw rate to be far away from the origin to get system as fast as possible. In the first case, we set one pole at the origin and the other pole at -1000. The system response is shown in Figure 5.28.

The motor input voltage required to rotate this MIP is 24V. The gain $K_{\text{R}} = [0 \quad 31.62]$ will be implemented into the MIP.

In the second case, we are placed all poles at $P = [0 \quad 725]$ and the simulation of the step response are shown in Figure 5.30 and Figure 5.31.

The results show that yaw rate is 1 rad/s and makes MIP rotate 0.2 rad in 0.2 second.

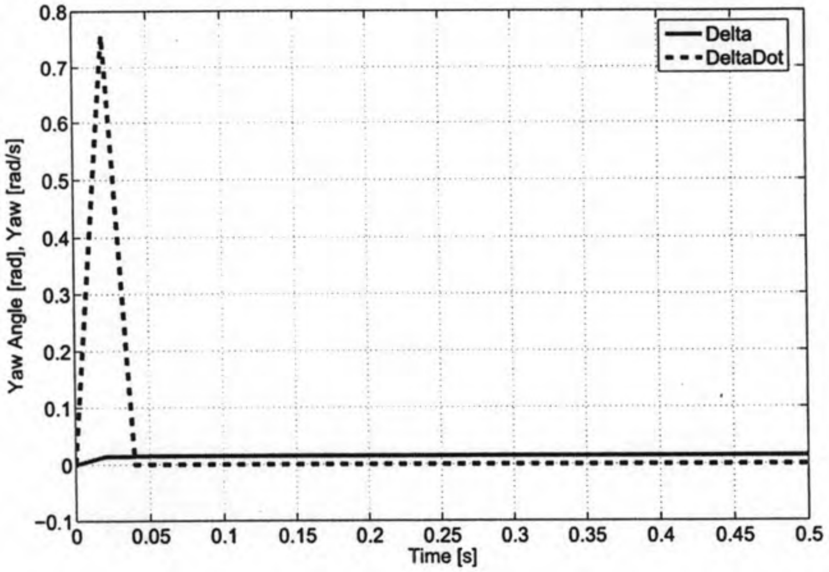

Figure 5.26: Impulse response of the plant with LQR control for rotating MIP.

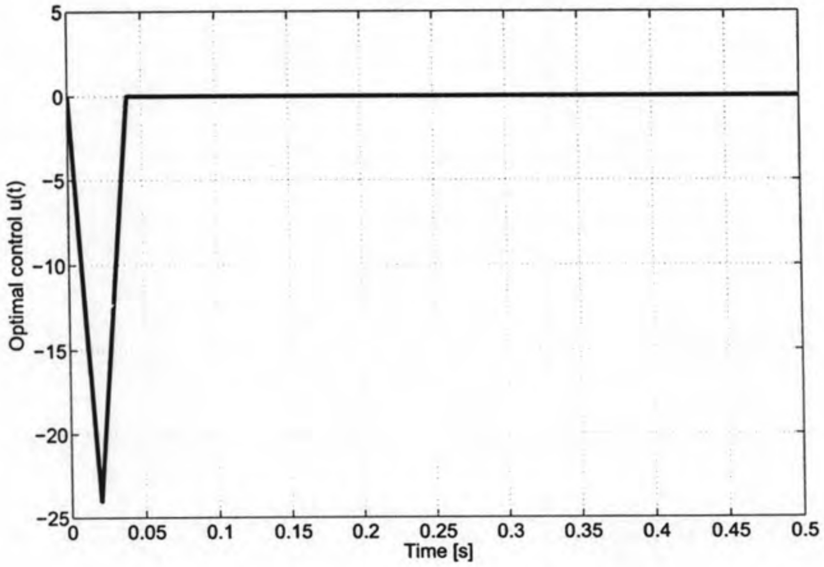

Figure 5.27: Control input of the plant with LQR control for rotating MIP.

5.2 LPV Controller Design for Balancing the MIP

In this section, we describe controller design for balancing by using LPV controller design based on robust $\mathcal{H}_\infty$ controller.

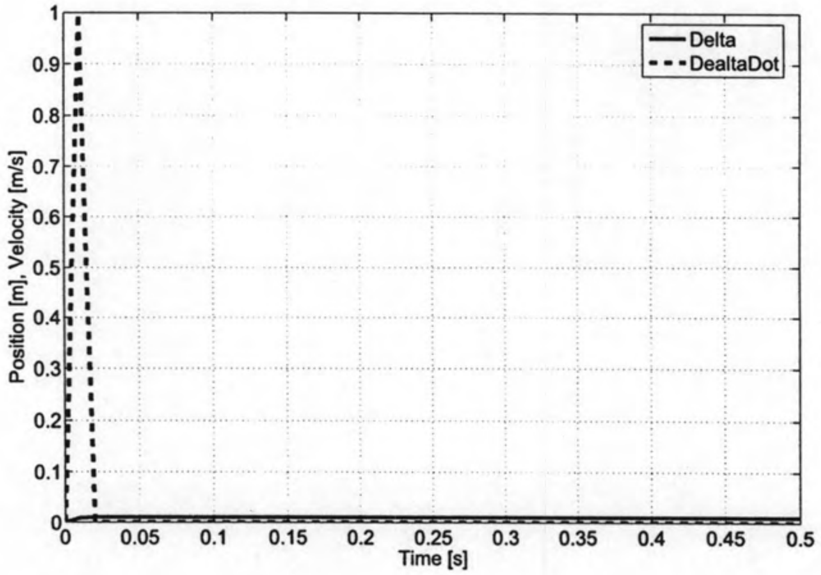

Figure 5.28: Impulse response of the plant with pole placement control for rotating MIP.

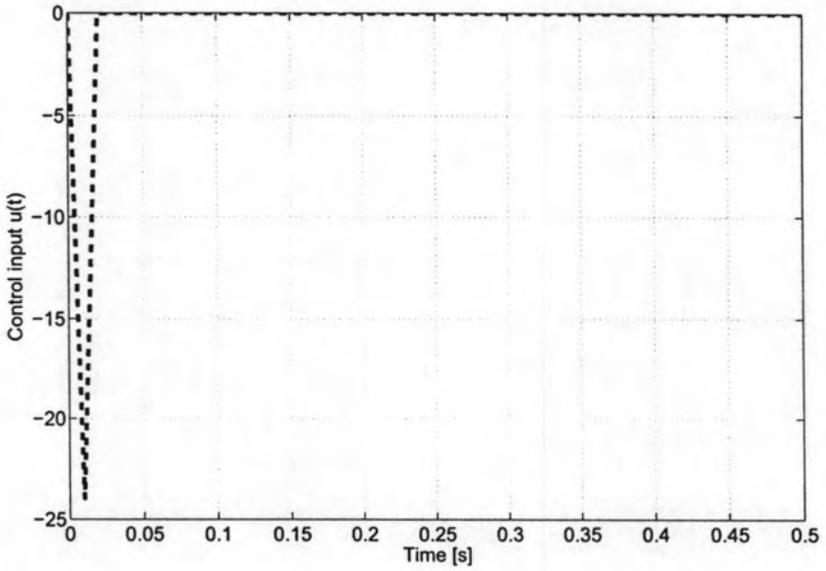

Figure 5.29: Control input of the plant with pole placement control for rotating MIP.

5.2.1 LPV Modeling of MIP

From nonlinear dynamic model of MIP (4.38) and (4.39), we define $\delta = M_\text{P} L \dot{\theta}_\text{P} \sin\theta_\text{P}$ to be the parameter which varies with states. And we approximate $\cos\theta_\text{P} = 1$ and $\sin\theta_\text{P} = \theta_\text{P}$, to

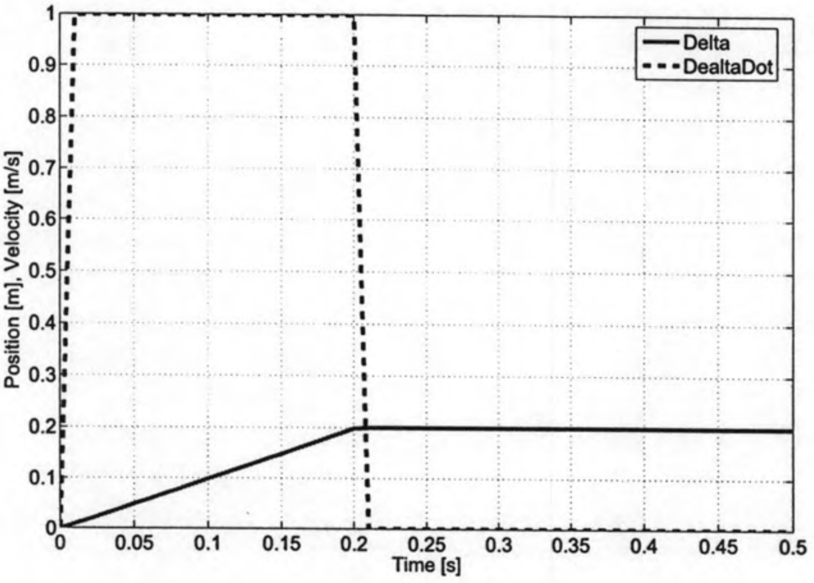

Figure 5.30: Step response of the plant with pole placement control for rotating MIP.

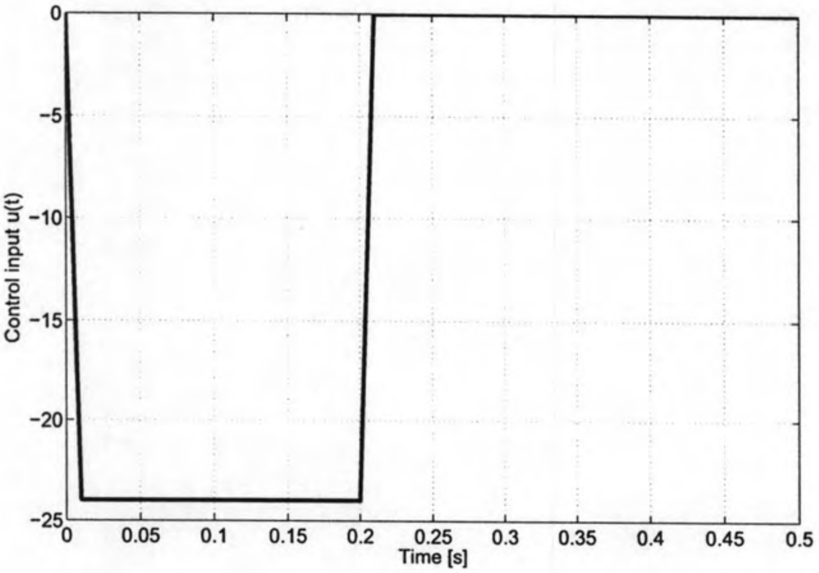

Figure 5.31: Control input of the plant with pole placement control for rotating MIP.

get

$$\ddot{x} = \frac{-2\eta_{\mathrm{g}}\eta_{\mathrm{m}}K_{\mathrm{m}}K_{\mathrm{g}}^2K_{\mathrm{e}}\left(J_{\mathrm{P}} + M_{\mathrm{P}}L^2 - M_{\mathrm{P}}Lr\right)\dot{x}}{R_{\mathrm{a}}r^2\alpha} - \frac{\left(J_{\mathrm{P}} + M_{\mathrm{P}}L^2\right)\delta\dot{\theta}_{\mathrm{P}} - (M_{\mathrm{P}}L)^2g\theta_{\mathrm{P}}}{\alpha} + \frac{\eta_{\mathrm{g}}\eta_{\mathrm{m}}K_{\mathrm{m}}K_{\mathrm{g}}\left(J_{\mathrm{P}} + M_{\mathrm{P}}L^2 - M_{\mathrm{P}}Lr\right)V_{\mathrm{B}}}{R_{\mathrm{a}}r\alpha} \tag{5.6}$$

$$\ddot{\theta}_{\mathrm{P}} = \frac{2\eta_{\mathrm{g}}\eta_{\mathrm{m}}K_{\mathrm{m}}K_{\mathrm{g}}^{2}K_{\mathrm{e}}\left(-M_{\mathrm{P}}L + r\beta\right)\dot{x}}{R_{\mathrm{a}}r^{2}\alpha} - \frac{M_{\mathrm{P}}L\delta\dot{\theta}_{\mathrm{P}} - M_{\mathrm{P}}gL\beta\theta_{\mathrm{P}}}{\alpha} - \frac{\eta_{\mathrm{g}}\eta_{\mathrm{m}}K_{\mathrm{m}}K_{\mathrm{g}}\left(-M_{\mathrm{P}}L + r\beta\right)V_{\mathrm{B}}}{R_{\mathrm{a}}r\alpha} \tag{5.7}$$

where $\beta = 2M_{\mathrm{W}} + \frac{2J_{\mathrm{eq}}}{r^2} + M_{\mathrm{P}}$ and $\alpha = J_{\mathrm{P}}\beta + 2M_{\mathrm{P}}L^2\left(M_{\mathrm{w}} + \frac{J_{\mathrm{eq}}}{r^2}\right)$

We obtain LPV model in a form of state-space

$$\begin{pmatrix} \dot{x} \\ \ddot{x} \\ \dot{\theta}_{\mathrm{P}} \\ \ddot{\theta}_{\mathrm{P}} \end{pmatrix} = \begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & \frac{-2\eta_{\mathrm{g}}\eta_{\mathrm{m}}K_{\mathrm{m}}K_{\mathrm{g}}^{2}K_{\mathrm{e}}\left(J_{\mathrm{P}}+M_{\mathrm{P}}L^2-M_{\mathrm{P}}Lr\right)}{R_{\mathrm{a}}r^2\alpha} & \frac{(M_{\mathrm{P}}L)^2g}{\alpha} & \frac{-\left(J_{\mathrm{P}}+M_{\mathrm{P}}L^2\right)\delta}{\alpha} \\ 0 & 0 & 0 & 1 \\ 0 & \frac{2\eta_{\mathrm{g}}\eta_{\mathrm{m}}K_{\mathrm{m}}K_{\mathrm{g}}^{2}K_{\mathrm{e}}(-M_{\mathrm{P}}L+r\beta)}{R_{\mathrm{a}}r^2\alpha} & \frac{M_{\mathrm{P}}gL\beta}{\alpha} & \frac{-M_{\mathrm{P}}L\delta}{\alpha} \end{pmatrix} \begin{pmatrix} x \\ \dot{x} \\ \theta_{\mathrm{P}} \\ \dot{\theta}_{\mathrm{P}} \end{pmatrix} + \begin{pmatrix} 0 \\ \frac{\eta_{\mathrm{g}}\eta_{\mathrm{m}}K_{\mathrm{m}}K_{\mathrm{g}}\left(J_{\mathrm{P}}+M_{\mathrm{P}}L^2-M_{\mathrm{P}}Lr\right)}{R_{\mathrm{a}}r\alpha} \\ 0 \\ \frac{-\eta_{\mathrm{g}}\eta_{\mathrm{m}}K_{\mathrm{m}}K_{\mathrm{g}}(-M_{\mathrm{P}}L+r\beta)}{R_{\mathrm{a}}r\alpha} \end{pmatrix} V_{\mathrm{B}} \tag{5.8}$$

We will use LPV model to design controller in next section.

5.2.2 LPV Control of MIP

- **LPV Control with LFT structure**

  Now, we look at a linear time-invariant system in which the uncertainties enter into a linear fractional fashion in order to define the time-varying uncertain system. This is the topic of this investigation.

$$\begin{pmatrix} \dot{x} \\ z_\delta \\ z_p \\ y \end{pmatrix} = \left(\begin{array}{c|cc|c} A & B_\delta & B_p & B_u \\ \hline C_\delta & D_{\delta\delta} & D_{\delta p} & D_{\delta u} \\ C_p & D_{p\delta} & D_{pp} & D_{pu} \\ \hline C_y & D_{y\delta} & D_{yp} & D_{yu} \end{array}\right) \begin{pmatrix} x \\ w_\delta \\ w_p \\ u \end{pmatrix} \tag{5.9}$$

  where $w_\delta = \Delta(\delta)z_\delta = \mathrm{diag}(\delta_1, \cdots, \delta_r)z_\delta$ and $|\delta_i| \leq 1, \forall i$.

  The normalization of the uncertainty is

$$\delta' = \frac{\delta}{\delta_{max}} = \frac{M_{\mathrm{P}}L\dot{\theta}_{\mathrm{P}}\sin\theta_{\mathrm{P}}}{M_{\mathrm{P}}L\dot{\theta}_{\mathrm{Pmax}}\sin\theta_{\mathrm{Pmax}}}$$

  Assume that $\delta_{max} = \frac{\pi}{3}$ rad/sec and $\dot{\delta}_{max} = \pi$ rad/sec.

  We can derive equation (5.8) in LFT form as equation (5.9) where

$$A = \begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & -1.5581 & 0.4904 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 19.8735 & 25.8343 & 0 \end{pmatrix}, \quad B_p = \begin{pmatrix} 0 \\ 0.0946 \\ 0 \\ -1.2066 \end{pmatrix}$$

$$B_\delta = \begin{pmatrix} 0 & 0 \\ -3.4646 & 0 \\ 0 & 0 \\ 0 & -8.6896 \end{pmatrix}, \quad C_\delta = \begin{pmatrix} 0 & 0 & 0 & 0.0235 \\ 0 & 0 & 0 & 0.0235 \end{pmatrix}$$

$$C_p = I_{4\times 4}, \quad D_{\delta\delta} = 0, \quad D_{\delta p} = 0, \quad D_{pp} = 0$$

- **Design Structure LPV Synthesis**

The main objective is to find the LPV controller which makes MIP stable in the presence of large deviation angles. An interconnection diagram for this design example as illustrated in Figure 5.32. From the figure P is the LTI system of equation (5.9), K is the controller from equation (2.37), w_{p1}, w_{p2} are exogenous input, z_{p1}, z_{p2} are exogenous output. Therefore the performance weight w_1 and w_2 are chosen, in the form

$$w(s) = \frac{s/M + \omega_B^*}{s + \omega_B^* A} \tag{5.10}$$

we see that $1/|w(j\omega)|$ is equal to $A \leq 1$ at low frequencies and is equal to $M \geq 1$ at high frequencies. And the asymptote crosses 1 at the frequency ω_B^*, which is approximately the bandwidth requirement. The weight w_u control weighting function which penalizes the size of control input. It is chosen to be a constant.

The LPV design objective is to find the LPV controller K and scheduling function $f_c(\Delta_c(\delta))$ such that

1. the closed-loop system is exponentially stable,
2. the induced L_2 gain from $w_p \rightarrow z_p$ is bounded by γ.

$$\left\| \begin{pmatrix} z_{p1} \\ z_{p2} \\ z_{p3} \end{pmatrix} \right\|_2 \leq \gamma \left\| \begin{pmatrix} w_{p1} \\ w_{p2} \end{pmatrix} \right\|_2 \tag{5.11}$$

for all possible trajectories of $\Delta(\delta)$.

5.2.3 Simulation Results

The LPV and the LQR controller are used with nonlinear MIP model. Because we focus only on the stability of the closed-loop system, the exogenous signals w_{p1} and w_{p2} are set to zero, we therefore select a simple input weight $w_u = 0.1111$ and the performance weight are illustrated in Figure (5.33).

We simulate the controller by changing the initial of the pendulum angle $\theta_P(0) = 0.25$ rad and $\theta_P(0) = 0.5$ rad, the mass of the MIP, the length to the CG and adding some noise to the

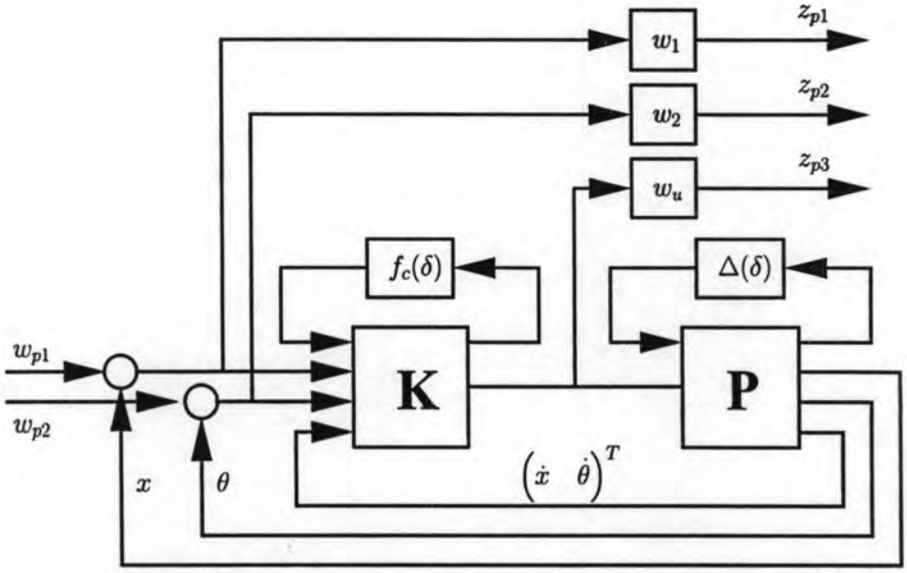

Figure 5.32: Design structure for LPV synthesis.

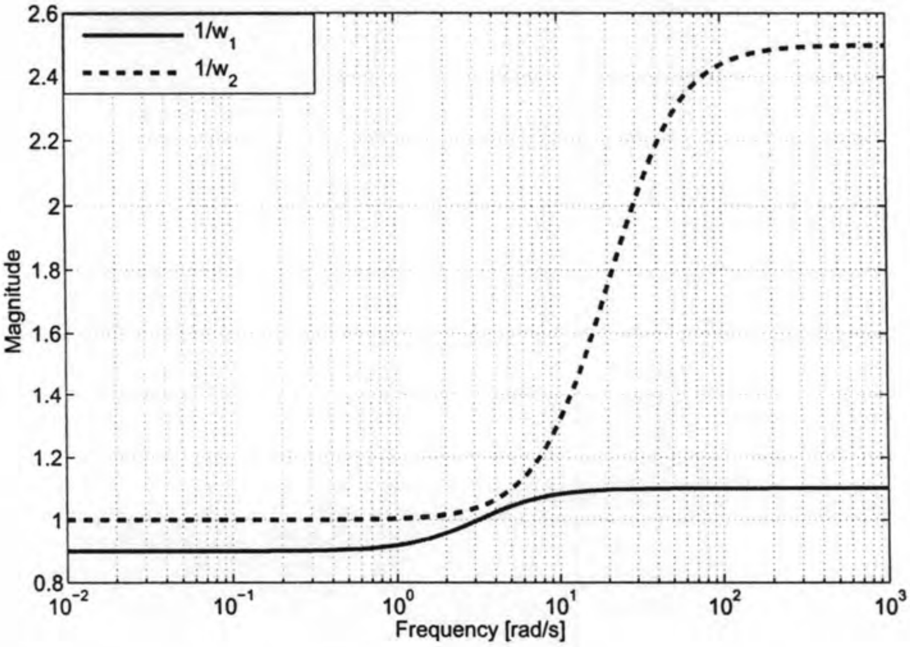

Figure 5.33: Inverse of performance weight w_1 (solid line) and w_2 (dashed line) for $\mathcal{H}_\infty$ design.

angle and the angular velocity of the pendulum.

From SolidWorks we can find the relationship between mass of pendulum and length to the CG as below

- In normal condition the mass of pendulum $M_P = 9.6$ Kg, we got the length to the CG is $L = 0.152.$ m.
- When we increase the mass of pendulum $M_P = 15$ Kg, we got the length to the CG is $L = 0.16$ m.
- When we decrease the mass of pendulum $M_P = 6$ Kg, we got the length to the CG is $L = 0.16$ m.

Now we can separate the simulation in 12 cases and it is shown in Table 5.1.
In all cases, the parameters of all the controllers is found in normal values. When we change the parameters that show in Table 5.1, the parameters of the controllers are not change.

Table 5.1: Number of cases for simulations

Cases	θ_P **(rad)**	M_P **(Kg)**	L **(m)**	**Add Noise from Gyro and Accelero**
1	0.25	9.6	0.152	No
2	0.25	9.6	0.152	Yes
3	0.25	15	0.16	No
4	0.25	15	0.16	Yes
5	0.25	6	0.14	No
6	0.25	6	0.14	Yes
7	0.5	9.6	0.152	No
8	0.5	9.6	0.152	Yes
9	0.5	15	0.16	No
10	0.5	15	0.16	Yes
11	0.5	6	0.14	No
12	0.5	6	0.14	Yes

From Cases 1 to 6, we simulate with the initial angle $\theta_P = 0.25$ rad and from case 7 to 12 , we simulate with the initial angle $\theta_P = 0.5$ rad.

- Cases 1 and 2 are simulated with the normal parameters of the pendulum. In case 2 we added some noises to the angle and the angular velocity to simulate the noise from accelerometter and gyroscope respectively. We can see that the simulation results are guarantee stability with reject the noise from the system. With LQR controller the performance of the system is better than LPV and pole-placement controllers. LQR control input is required hight input voltage than the other.
- Cases 3 and 4, we add more mass to the body of pendulum, so the length to the CG increase. This even to simulate like the human stand on the chassis of the MIP. After

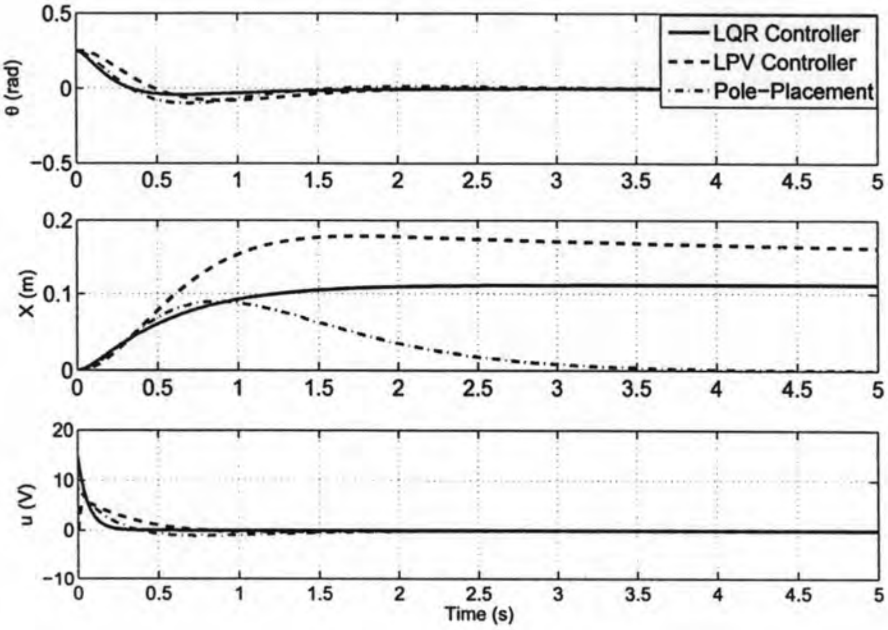

Figure 5.34: Case 1: simulation with initial angle $\theta_P = 0.25$ rad, $M_P = 9.6$ Kg and $L = 0.152$ m.

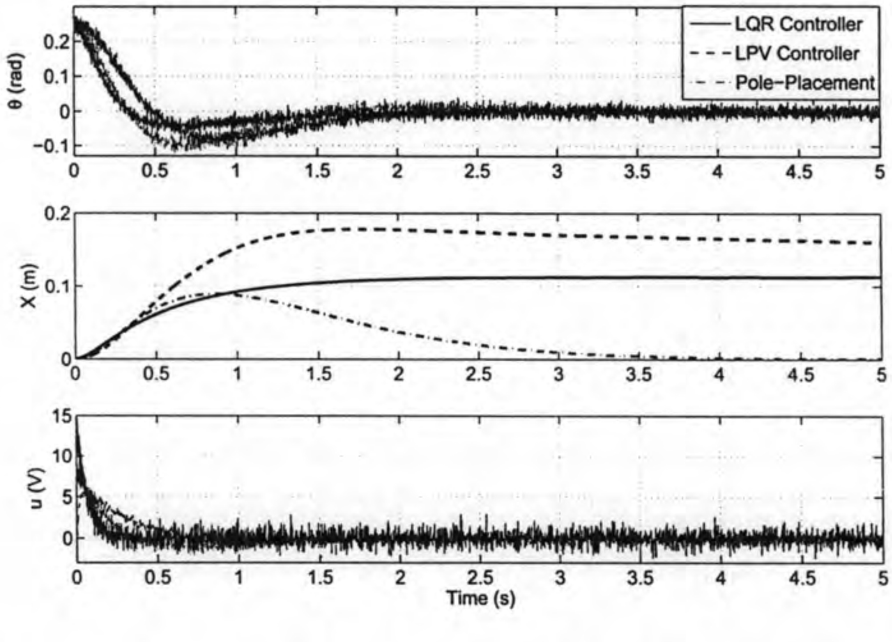

Figure 5.35: Case 2: simulation with initial angle $\theta_P = 0.25$ rad, $M_P = 9.6$ Kg and $L = 0.152$ m and add some noises to the angle and the angular velocity.

simulation, we can see that LPV and LQR controllers can make the MIP stable. The responses of pole-placement controller are oscillation, this mean that if we add more mass the pole placement controller cannot control the MIP. With LPV controller the performance of the system is better than LQR and pole-placement controllers.

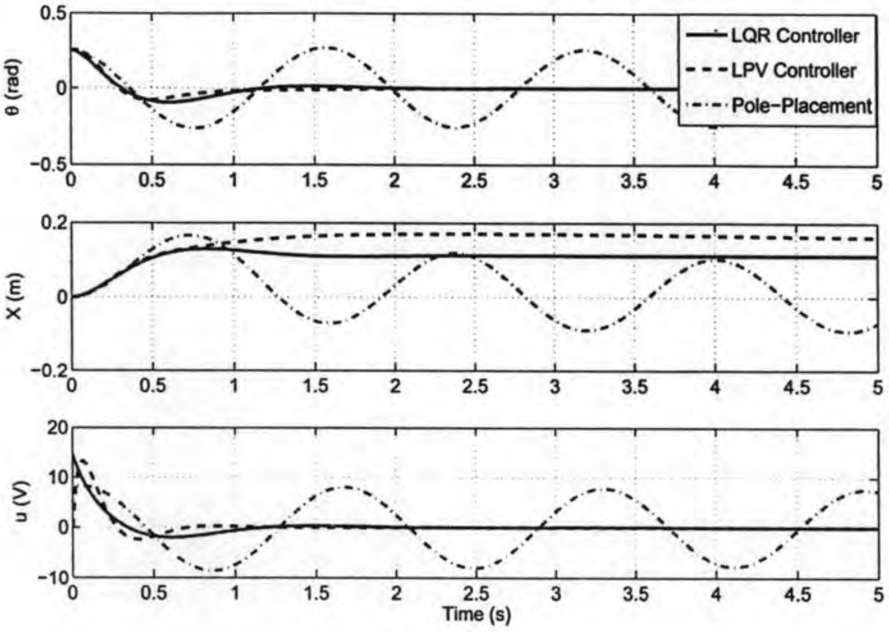

Figure 5.36: Case 3: simulation with initial angle $\theta_P = 0.25$ rad, $M_P = 15$ Kg, $L = 0.16$ m.

- Cases 5 and 6 we decrease the mass of the MIP to 6 Kg and also the length of the CG decrease, we can see that the responses of the states are slows. In these cases all the controllers have good performance and require small input voltage to apply to the motor.

- Cases 7 and 8 are similar to the cases 1 and 2, when we set the initial angle to $\theta_P = 0.5$ rad, these controllers can swing up the pendulum and balance the MIP within 2.5 second. With LQR controller the performance of the system is still better than LPV and pole-placement controllers. At the start up LQR required 24V control input voltage to perform the balancing MIP. This voltage value is the limit of the nominal voltage of the motor.

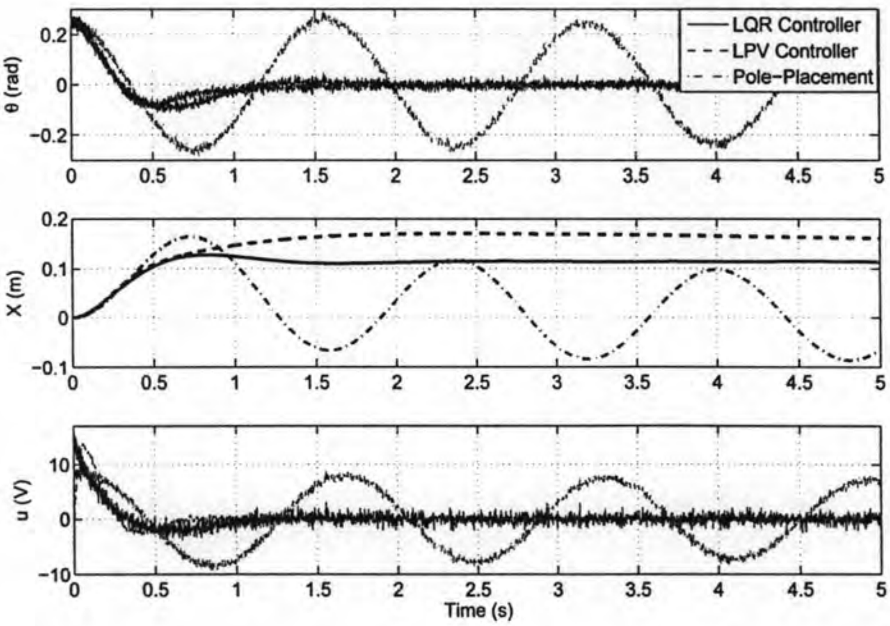

Figure 5.37: Case 4: simulation with initial angle $\theta_P = 0.25$ rad, $M_P = 15$ Kg, $L = 0.16$ m and add some noises to the angle and the angular velocity.

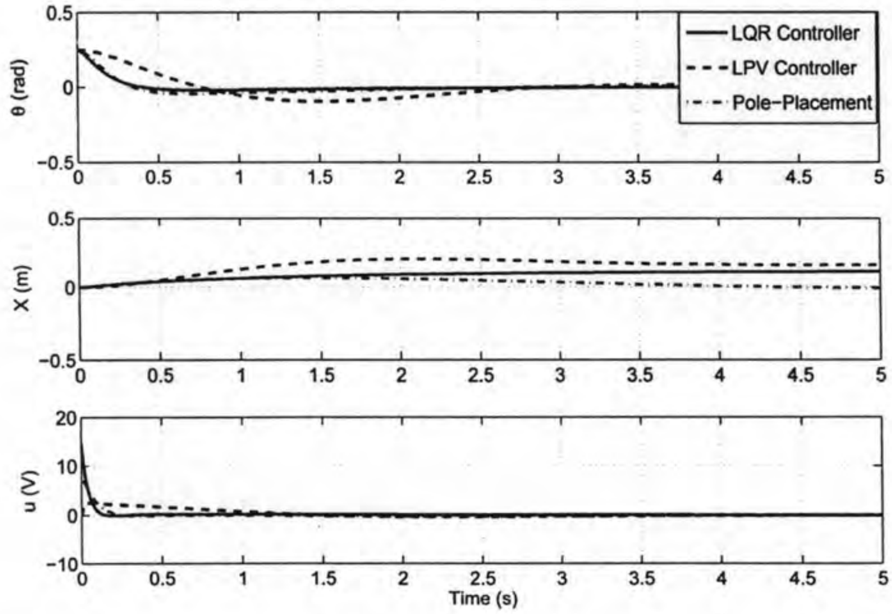

Figure 5.38: Case 5: simulation with initial angle $\theta_P = 0.25$ rad, $M_P = 6$ Kg, $L = 0.14$ m.

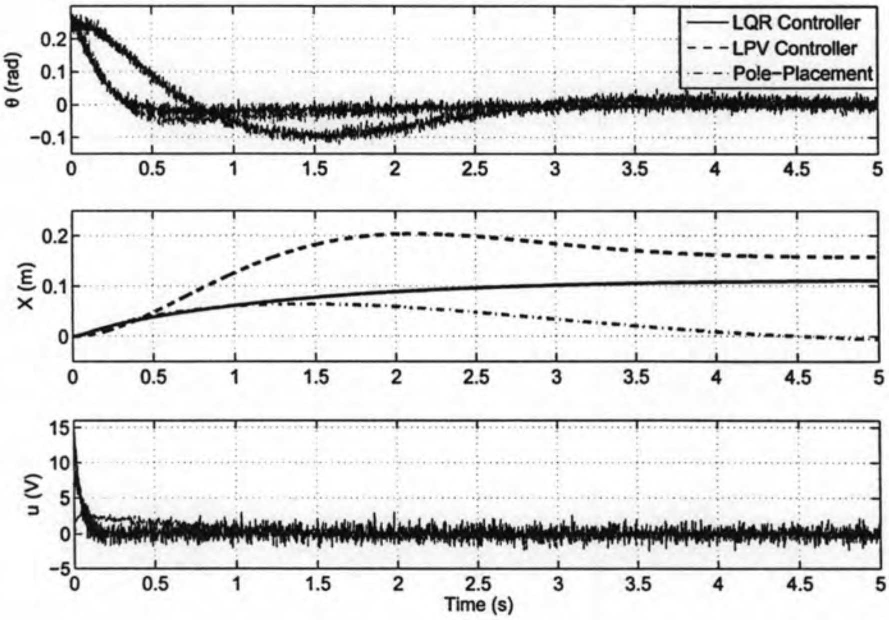

Figure 5.39: Case 6: simulation with initial angle $\theta_P = 0.25$ rad, $M_P = 6$ Kg, $L = 0.14$ m and add some noises to the angle and the angular velocity.

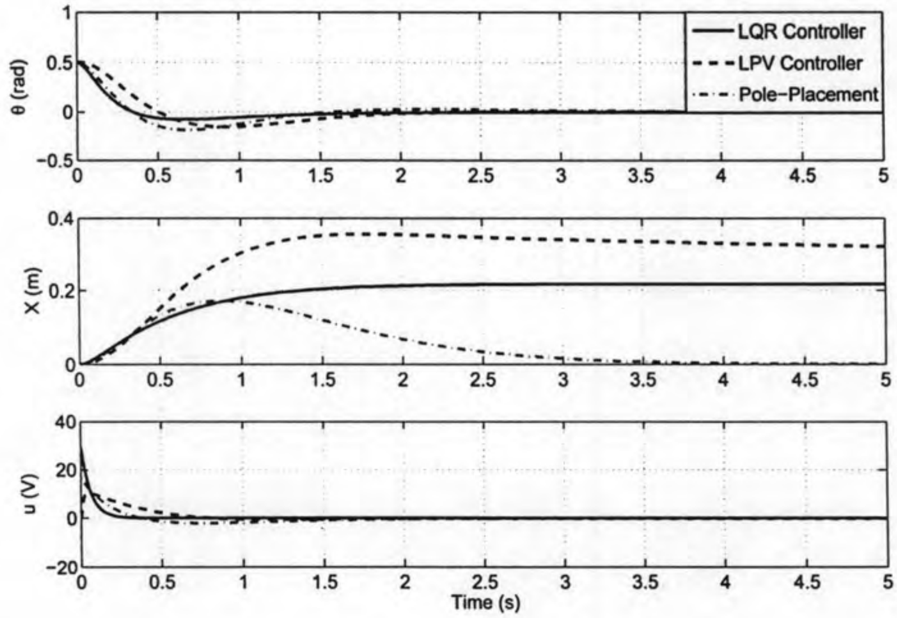

Figure 5.40: Case 7: simulation with initial angle $\theta_P = 0.5$ rad, $M_P = 9.6$ Kg, $L = 0.152$ m.

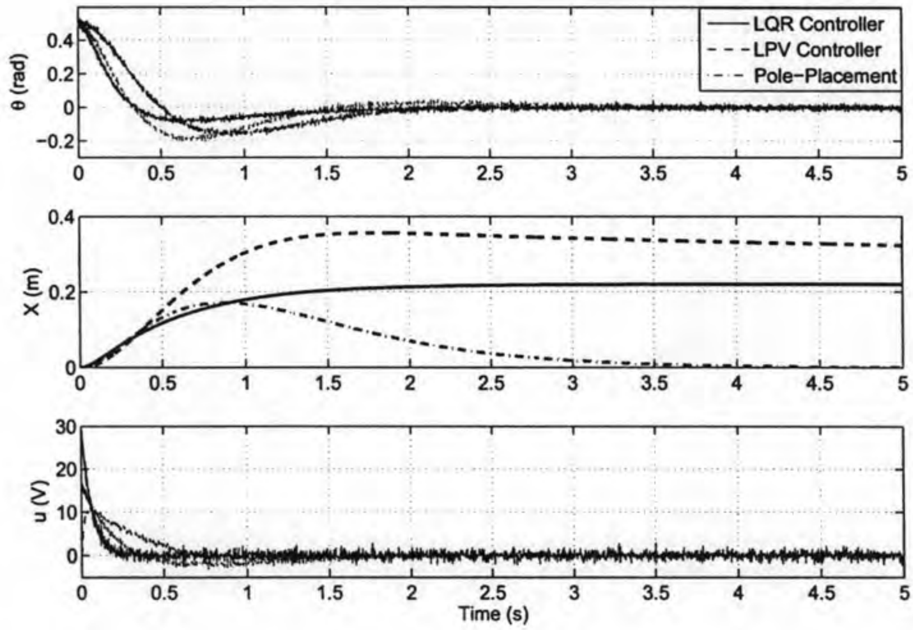

Figure 5.41: Case 8: simulation with initial angle $\theta_P = 0.5$ rad, $M_P = 9.6$ Kg, $L = 0.152$ m and add some noises to the angle and the angular velocity.

- Cases 9 and 10 are similar to the cases 3 and 4. In these cases, With LPV controller the performance of the system is better than LQR and pole-placement controllers. The responses of pole-placement controller are oscillation.

- Cases 11 and 12 are similar to the cases 5 and 6. The responses for LQR controller are better than the other controller with the requirement input voltage are limited to the nominal voltage of the motor.

5.3 Conclusions

We have designed three controllers for balancing the MIP. They are LQR controller, pole placement controller and LPV controller. From the simulation results, it is shown that the LPV controller has the performance and robustness are better than the other controller for balancing MIP. We can see from the simulation results that those controllers could balance the MIP even the mass and length of the MIP are changing. In this chapter we also design the controller for rotating the MIP with LQR controller and pole placement. The controllers

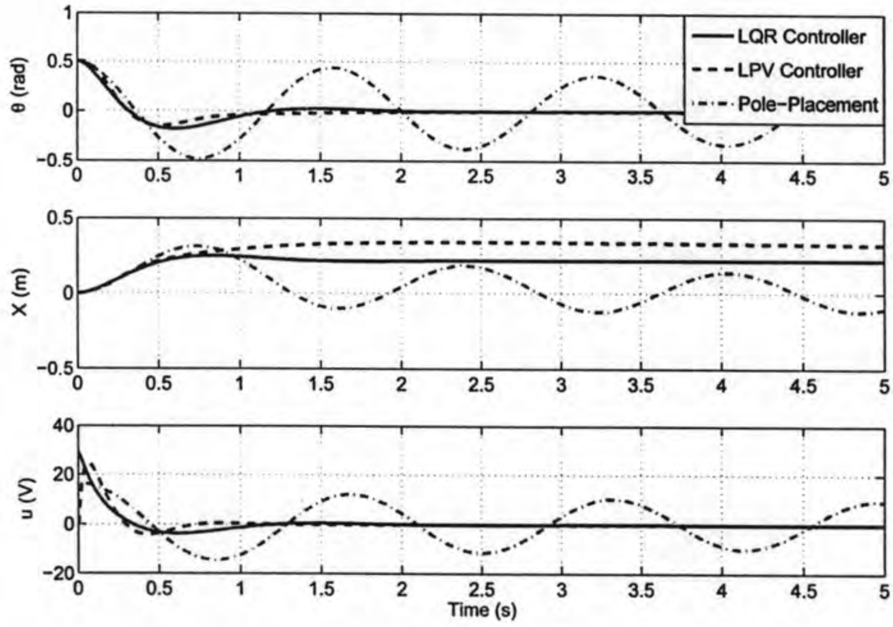

Figure 5.42: Case 9: simulation with initial angle $\theta_P = 0.5$ rad, $M_P = 15$ Kg, $L = 0.16$ m.

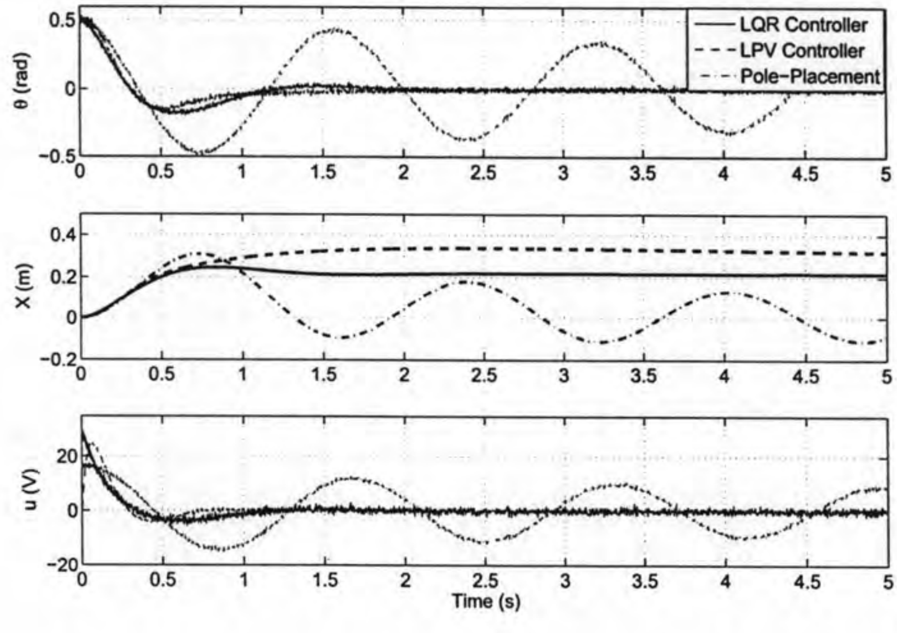

Figure 5.43: Case 10: simulation with initial angle $\theta_P = 0.5$ rad, $M_P = 15$ Kg, $L = 0.16$ m and add some noises to the angle and the angular velocity.

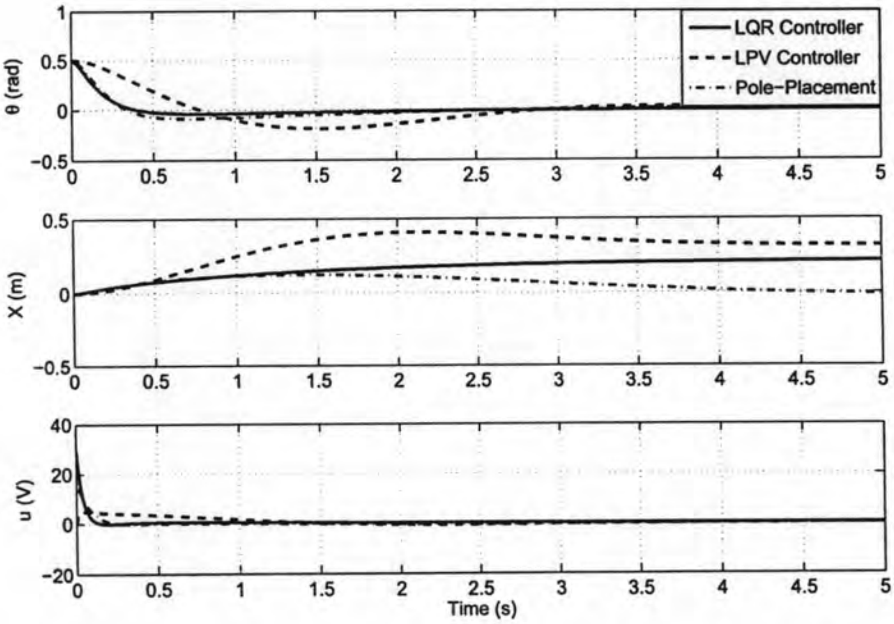

Figure 5.44: Case 11: simulation with initial angle $\theta_P = 0.5$ rad, $M_P = 6$ Kg, $L = 0.14$ m.

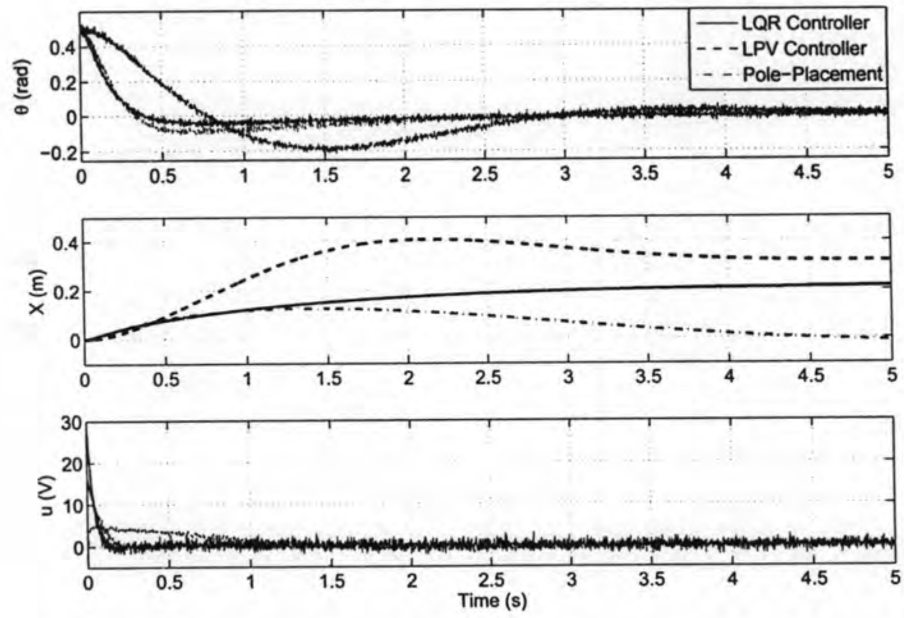

Figure 5.45: Case 12: simulation with initial angle $\theta_P = 0.5$ rad, $M_P = 6$ Kg, $L = 0.14$ m and add some noises to the angle and the angular velocity.

K_B, K_R for balancing and rotating are across the decoupling and are applied to the input voltage of the left and right motor.