ECE6554 Project Report: Planar Bi-Rotor Helicopter

Amey Parundekar
amey@gatech.edu

Chris Slagell
cslagell@gatech.edu

***Abstract*—This report provides a study and comparison of linear, non-linear and adaptive controllers for a planar bi-rotor helicopter. After linearization of the system around $\theta = 0$, we study the linear control of the system using linear quadratic regulator (LQR). Further we look at a linear model with adaptive gains. Noting the gains in the post transient stage, the adaptive systems with these gains are compared to the ones with an optimal gain set by a quadratic solver. Similar treatment is done to a non-linear system by doing a coordinate transform.**

***Index Terms*—systems, control**

I. Introduction

Planar Bi-Rotor Helicopter consists of two counter-torquing (pitch) thrusters. We study this system using the equations of motion for the center of mass, q, of the helicopter and θ, the orientation angle. The organization of this report is as follows. In **Section II** we represent the differential equations of motion in state space. In **Section III** we discuss an LQR implementation of the system and look at various ways to simulate the system. In **Section IV** we discuss an adaptive controller for the same linear model and also compare various trajectories based on pre and post transient gains. In **Section V** we disucss MRAC for a non-linear system alongside comparison on various trajectories from pre and post transient gains.

II. State Space Model

The system is represented in terms of its center of mass q and orientation θ of the helicopter.

$$m\ddot{q} = -d\dot{q} + e_2(\theta)\begin{bmatrix}1 & 1\end{bmatrix}\overrightarrow{f} - m\overrightarrow{g} \tag{1}$$

$$J\ddot{\theta} = r[1-1]\overrightarrow{f} \tag{2}$$

We take $x_2 = \dot{x}$, $x_1 = x$, $x_4 = \dot{y}$, $x_3 = y$, $\dot{\theta} = x_6$, $\theta = x_5$ as the system states for our state space model.

Hence re-writing the equations above with the given substitutions gives

$$\begin{bmatrix}\dot{x_1}\\ \dot{x_2}\\ \dot{x_3}\\ \dot{x_4}\\ \dot{x_5}\\ \dot{x_6}\end{bmatrix} = \begin{bmatrix} x_2 \\ -\frac{d}{m} * x_2 - \frac{sin(x_5)}{m} * (f_1 + f_2) \\ x_4 \\ -\frac{d}{m} * x_2 - \frac{cos(x_5)}{m} * (f_1 + f_2) - g \\ x_6 \\ \frac{r}{J} * (f_1 - f_2) \end{bmatrix} \tag{3}$$

Note that our control input is the fan thrust $\overrightarrow{f}$.

A. Linear Model

The only way to linearize the system is to set $x_5 = \theta = 0$. This removes the sinusoidal terms in the differential equations. Given that we set $\theta = 0$ and the definitions of the state and the control input are clear, we can define the linear state matrices A and B as the Jacobians with respect to the states and the control inputs respectively.

$$\mathbb{A} = \begin{bmatrix} \frac{\partial \mathbf{X}(\mathbf{x})}{\partial x_1} & \cdots & \frac{\partial \mathbf{X}(\mathbf{x})}{\partial x_6} \end{bmatrix}$$

$$= \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & \frac{-d}{m} & 0 & 0 & -\frac{f_1(0)+f_2(0)}{m} & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & \frac{-d}{m} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

and

$$\mathbb{B} = \begin{bmatrix} \frac{\partial \mathbf{X}(\mathbf{f})}{\partial f_1} & \frac{\partial \mathbf{X}(\mathbf{f})}{\partial f_2} \end{bmatrix}$$

$$= \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ \frac{1}{m} & \frac{1}{m} \\ 0 & 0 \\ \frac{r}{J} & -\frac{r}{J} \end{bmatrix}$$

B. Non-linear model

To get a non-linear model, we separate the non-linear terms from equation 3. Thus we get,

$$\begin{bmatrix} \dot{x_1} \\ \dot{x_2} \\ \dot{x_3} \\ \dot{x_4} \\ \dot{x_5} \\ \dot{x_6} \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & \frac{-d}{m} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & \frac{-d}{m} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ x_6 \end{bmatrix} + \begin{bmatrix} 0 \\ -\frac{sin(x_5)}{m} * (f_1 + f_2) \\ 0 \\ -\frac{cos(x_5)}{m} * (f_1 + f_2) - g \\ 0 \\ \frac{r}{J} * (f_1 - f_2) \end{bmatrix}$$

Let's analyse the non-linear part in the above equation. We can separate the matrix thus so:

$$\begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & sin(x_5) & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & cos(x_5) & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} -g \\ -\frac{f_1 + f_2}{m} \\ 0 \\ 0 \\ 0 \\ \frac{r}{J} * (f_1 - f_2) \end{bmatrix}$$

This can be further split thus so:

$$\begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & sin(x_5) & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & cos(x_5) & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} * \left(\begin{bmatrix} 0 & 0 \\ -\frac{1}{m} & -\frac{1)}{m} \\ 0 & 0 \\ -\frac{1}{m} & -\frac{1}{m} \\ 0 & 0 \\ \frac{r}{J} & -\frac{r}{J} \end{bmatrix} \overrightarrow{f} + \begin{bmatrix} -g \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \right)$$

This equation is of the form $\dot{x} = Ax + B(u + \beta^T \Phi(x))$. In our case Φ contains only x^0 which means we essentially don't have any non-linear disturbances.

Hence in a non-linear system we would have the following state matrices:

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & \frac{-d}{m} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & \frac{-d}{m} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

$$B = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & sin(x_5) & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & cos(x_5) & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

C. *Non-linear system with coordinate transform*

This system and many other engineered or man-made mobile vehicles have the property of differential flatness. There exists a transformation of state that will provide full control of the position variables if control over the orientation is relaxed. The transformation is to consider control of a virtual point somewhere ahead of the robot. The transformation of coordinates is

$$q' = q + \lambda e_2(\theta) \tag{4}$$

Taking a derivative and a double derivative of this we get,

$$\dot{q}' = \dot{q} + \lambda \dot{e_2}(\theta) \tag{5}$$
$$\ddot{q}' = \ddot{q} + \lambda \ddot{e_2}(\theta) \tag{6}$$

Now we substitute equation 1 in 5 to get a differential equation of motion on the chosen virtual point.

$$\ddot{q}' = -\frac{d}{m}\dot{q}' - \lambda R(\theta) \begin{bmatrix} \frac{d}{m} \\ \theta \end{bmatrix} - R(\theta) \begin{bmatrix} \frac{\lambda r}{J} & -\frac{\lambda r}{J} \\ -\frac{1}{m} & -\frac{1}{m} \end{bmatrix} \overrightarrow{f} - \overrightarrow{g} \tag{7}$$

We again take $x_2 = \dot{x}$, $x_1 = x$, $x_4 = \dot{y}$, $x_3 = y$, $\dot{\theta} = x_6$, $\theta = x_5$ as the system states for our state space model. Hence we get a set of differential equations in terms of these state variables. Although a full derivation is not shown here, it is very similar to what is done in the previous subsection.

$$
\begin{bmatrix} \dot{x_1} \\ \dot{x_2} \\ \dot{x_3} \\ \dot{x_4} \\ \dot{x_5} \\ \dot{x_6} \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & \frac{-d}{m} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & \frac{-d}{m} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ x_6 \end{bmatrix} +
$$

$$
\begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & cos(x_5) & 0 & -sin(x_5) & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & sin(x_5) & 0 & cos(x_5) & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} *
$$

$$
\left(\begin{bmatrix} 0 & 0 \\ -\frac{\lambda r}{J} & \frac{\lambda r}{J} \\ 0 & 0 \\ -\frac{1}{m} & -\frac{1)}{m} \\ 0 & 0 \\ \frac{r}{J} & -\frac{r}{J} \end{bmatrix} \vec{f} + \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & \frac{-d\lambda}{m} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -\lambda & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} 1 \\ x_6 \\ 0 \\ x_6^2 \\ 0 \\ 0 \end{bmatrix} \right)
$$

Hence we have some non-linear disturbance terms in our state space equation this time. We can treat these in a way that we have an adaptive cancellation term to compensate for this disturbance.

III. Linear System

We use a Linear Quadratic Regulator (LQR) to figure out the optimal gain for the controller $u = -Kx$. We have the ability to control how fast the control is applied and the rate of change in states using the weighing matrices Q and R. Q is used to control how fast the states converge to stable values whereas R is used to punish the control effort. The cost function to be optimised is given as follows:

$$
J = x^T(t_1)F(t_1)x(t_1) + \int_{t_0}^{t_1} \left(x^T Q x + u^T R u + 2x^T N u \right) dt \tag{8}
$$

The results of simulation with Q and R as identity matrices is shown in figures 1, 2, 3.

The arrows in figure 3 show the orientation of the body at the given time. Only 10 orientations have been shown.

The state evolution over time shows that they indeed stabilize at 0. The thruster's force values don't go beyond 6 times the gravitational force.

Next let's try to change the Q and R see their effect over the trajectory. For this we take $Q = 100 * I_6$ and $R = I_2$. This will penalise slower state convergence and motivate faster stability. As seen in the figures 4, 5 and 6, the magnitude of forces increased in order to perform a faster state convergence. Similarly we can penalise the force produced by the thrusters and thereby save fuel by using a higher value of R.

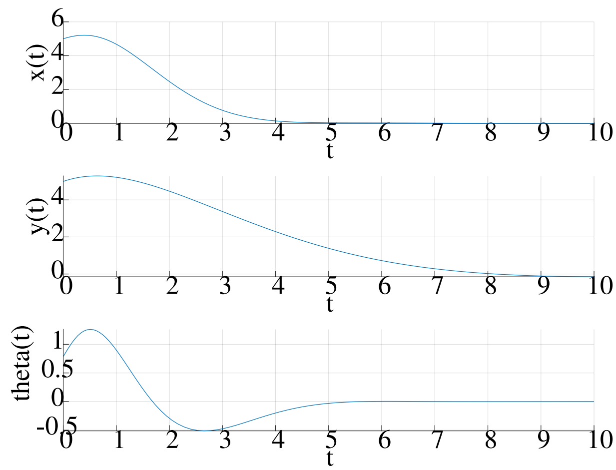

Fig. 1. States: Linear Quadratic Regulator

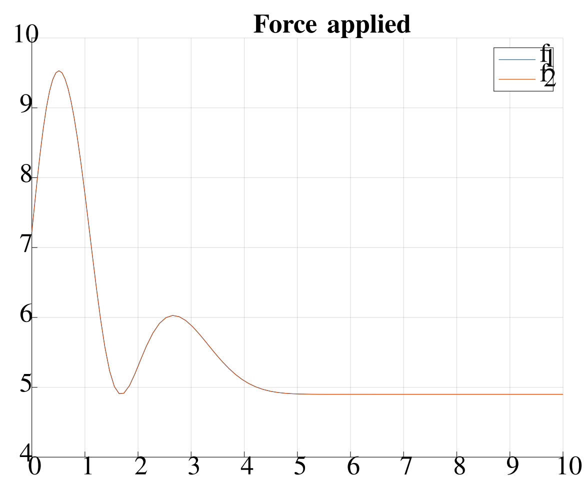

Fig. 2. Control Inputs: Linear Quadratic Regulator

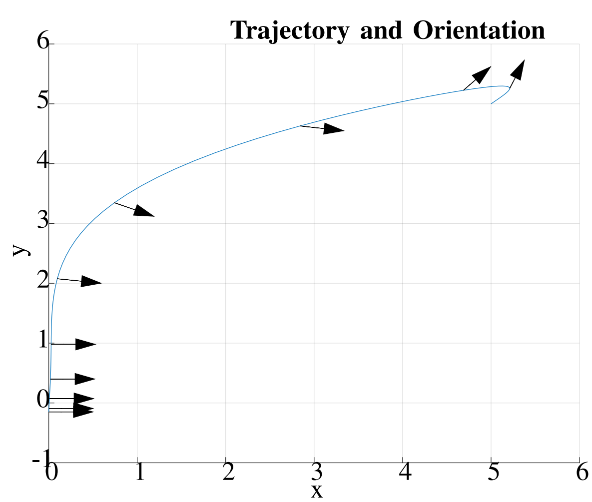

Fig. 3. Trajectory and orientation of the center of mass

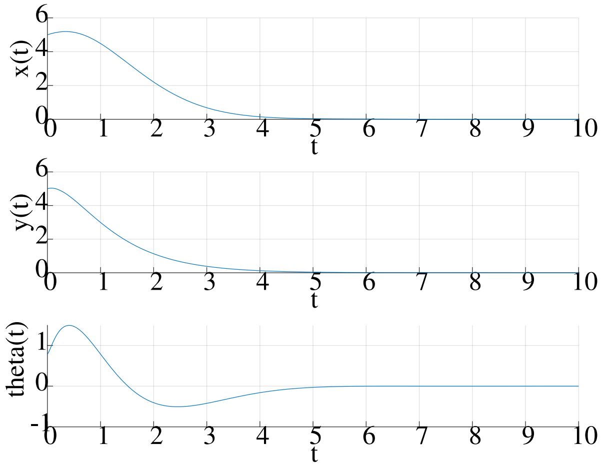

Fig. 4. States: Linear Quadratic Regulator with $Q = 100$,$R = 1$

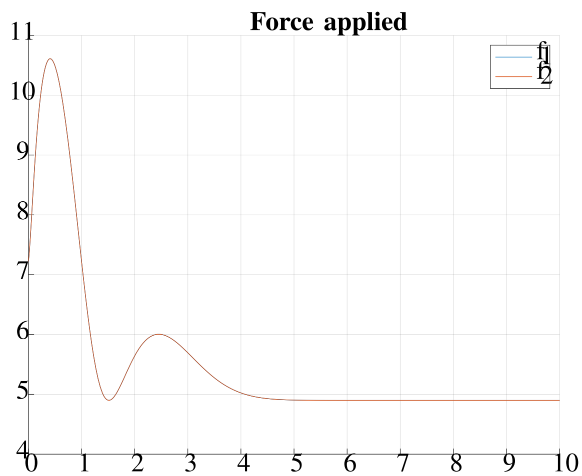

Fig. 5. Control Inputs: Linear Quadratic Regulator with $Q = 100$,$R = 1$

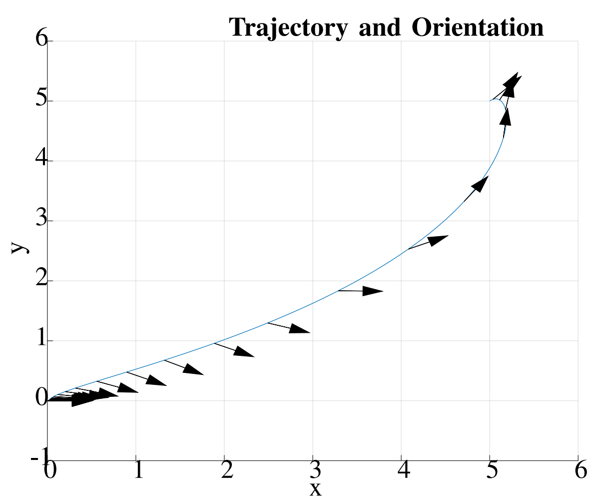

Fig. 6. Trajectory and orientation with $Q = 100$,$R = 1$

IV. Adaptive Control of the Linear Model

Let us now look at adaptive controller for the same system that is able to track a reference controller signal. For this we will use a standard MRAC model [1]. We simulated the system with two different reference signals.

1) $r(t) = [3, 0]$
2) $r(t) = [3, 0]$ if $t < 16$ else $r(t) = [sin(t), cos(t)]]$

Let us look at the resulting system simulation.

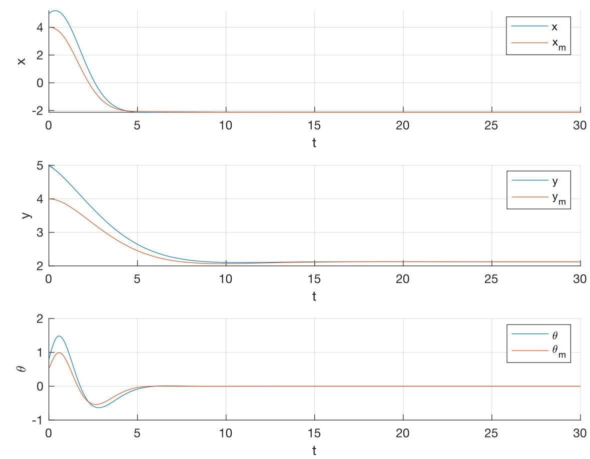

Fig. 7. States: MRAC with $r = [3, 0]$

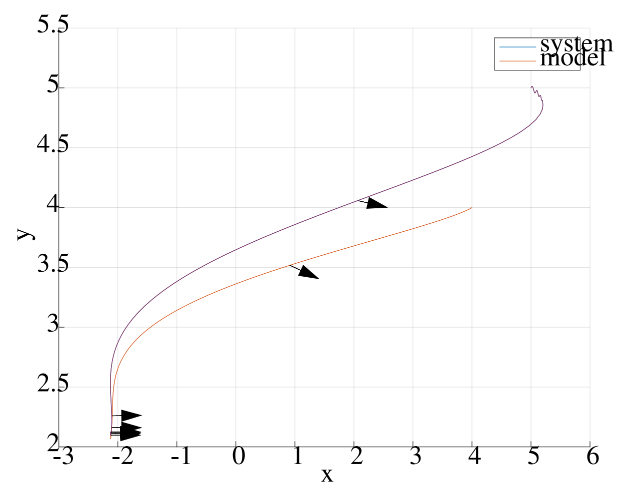

Fig. 8. Trajectory and orientation: MRAC with $r = [3, 0]$

As seen from the simulation figures 7, 11 it is clear that an adaptive implementation works well for the given linear system and we were able to track the chosen reference signals.

A. *Using Post-transient Adaptive Gains*

Let us now see how the system behaves when we use the post-transient gains as initial gains and re-run the system. For a better comparison, we reduced the gamma values on the

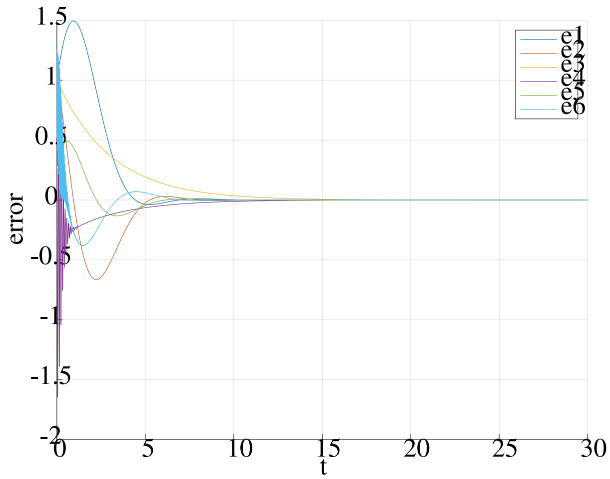

Fig. 9. Errors: MRAC with $r = [3, 0]$

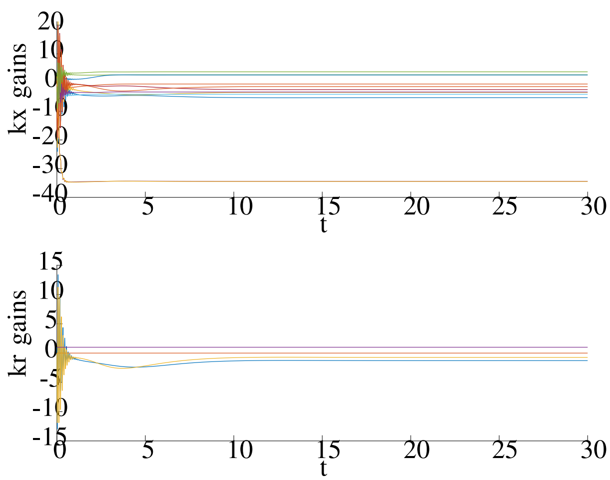

Fig. 10. Gains: MRAC with $r = [3, 0]$

adaptive gains so that the systems behaves poorly. Then we note the post-transient gains and re-run the adaption ODE. This gives us much smoother graphs for the states and the trajectory of the center of mass reflects the same. This can be seen in figures 15 and 16.

V. Non-linear adaptive control

In this section we revisit our non-linear system derivations from Section II and implement an MRAC based system for the same. There after we also try to compare the MRAC system with an MRAC system given post-transient gains. Here we will only consider the coordinate transformed non-linear system.

From figures 17 and 18 it is clear that the post-transient-gains-based system observes less oscillations and is much better than the system with arbitrary/LQR based gains for the initial values.

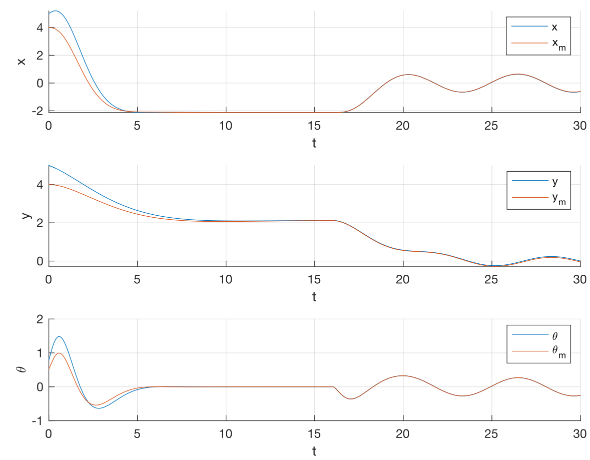

Fig. 11. States: MRAC with $r(t) = [3, 0]$ if $t < 16$ else $r(t) = [sin(t), cos(t)]]$

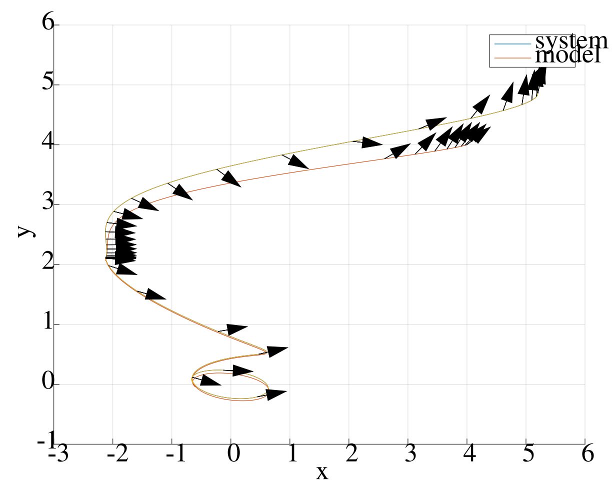

Fig. 12. Trajectory and orientation: MRAC with $r(t) = [3, 0]$ if $t < 16$ else $r(t) = [sin(t), cos(t)]]$

It was observed that the non-linear system is unstable with lower gamma values for the adaptive gains.

VI. Conclusion

Taking the differential equations for the motion of a bi-rotor planar helicopter, we made a state space model of the system. Using this model, we first simulated a linear model using LQR and showed that a stable system was produced. We tried various means to change the way the states would converge to stability. After that we analysed an adaptive system using MRAC for linear model and observed that a stable system was produced using variable gain adaptation law. We also used the post-transient gains from a previous simulation for the initial gain values and saw that the system behaved much better and had less oscillations even with low gamma values for the gains. Furthermore, we also tried to do the exact same for a non-

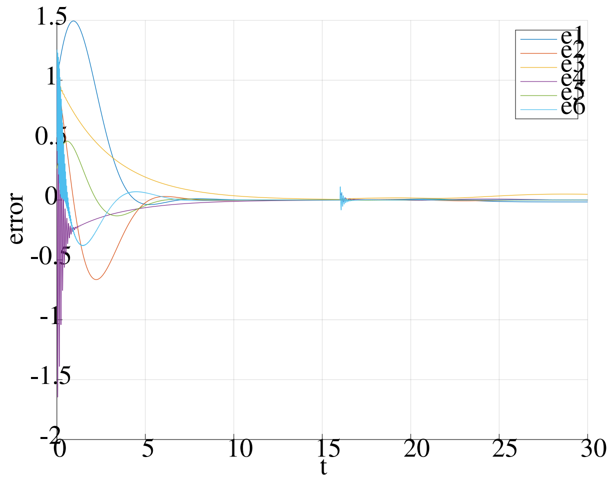

Fig. 13. Errors: MRAC with $r(t) = [3, 0]$ if $t < 16$ else $r(t) = [sin(t), cos(t)]]$

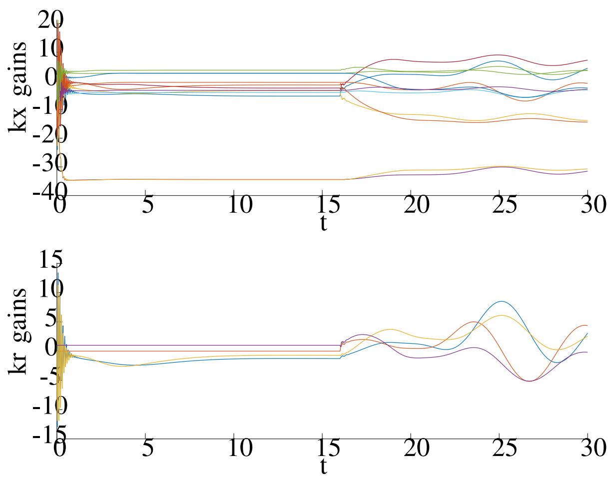

Fig. 14. Gains: MRAC with $r(t) = [3, 0]$ if $t < 16$ else $r(t) = [sin(t), cos(t)]]$

linear MRAC. The system was observed to be stable under certain conditions when the gamma values for the gains were high.

References

[1] MRAC model and equations for the adaptive control: https://in.mathworks.com/help/slcontrol/ug/model-reference-adaptive-control.html?s_eid=PSM_15028

[2] GitHub repository for the code toward this project: https://github.com/the-lost-explorer/bi-roto-heli-mrac

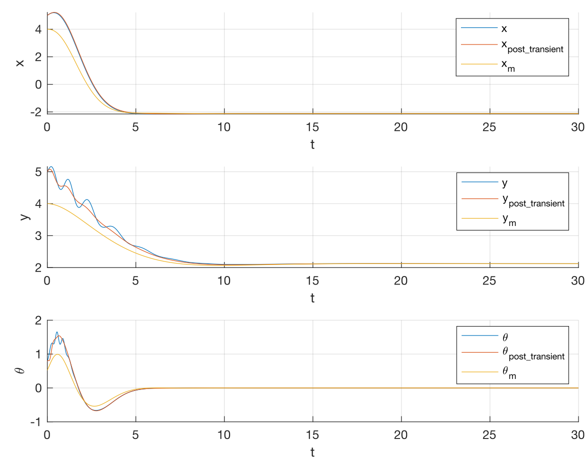

Fig. 15. States comparison of MRAC with post-transient gains

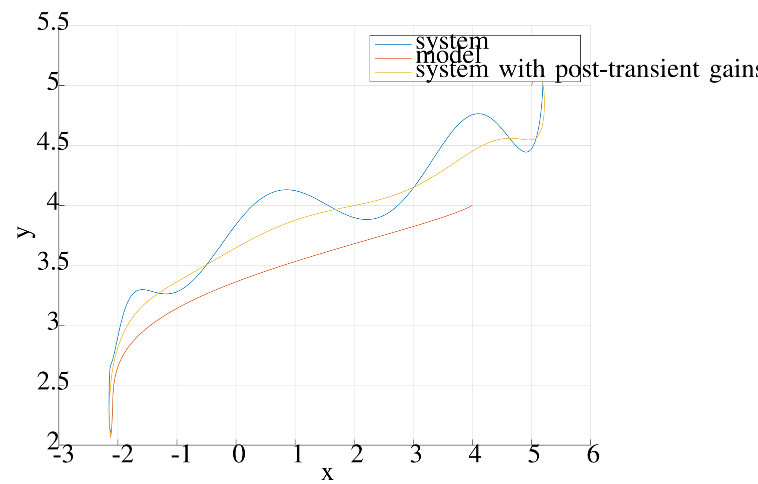

Fig. 16. Trajectory comparison of MRAC with post-transient gains

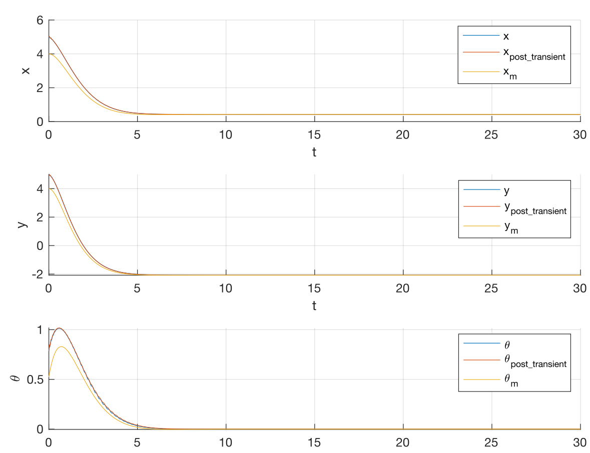

Fig. 17. States: Non-linear MRAC

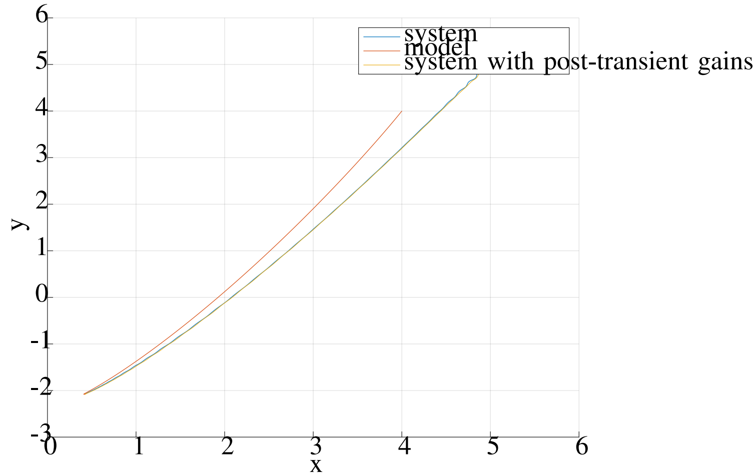

Fig. 18. Trajectory for non-linear MRAC

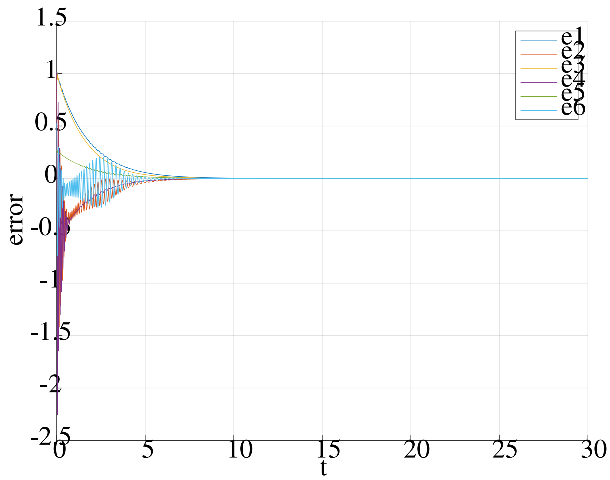

Fig. 19. Errors for non-linear MRAC

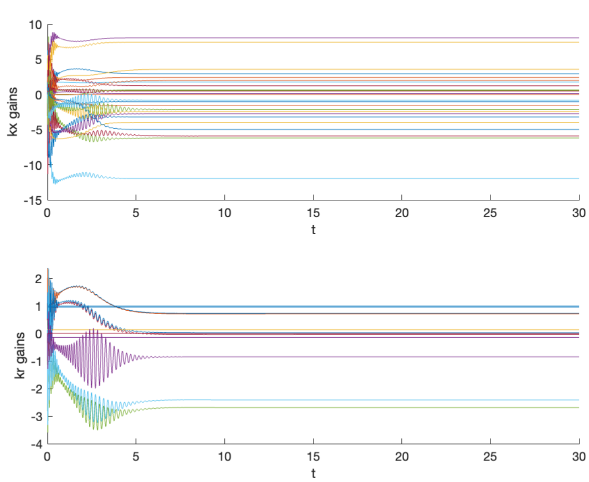

Fig. 20. Gains for non-linear MRAC