Trajectory tracking in quadrotor platform by using PD controller and LQR control approach

Maidul Islam¹, Mohamed Okasha² and Moumen Mohammad Idres³

¹, ², ³ Dept. of Mechanical Engineering, International Islamic University Malaysia (IIUM), Kuala Lumpur 50725, Malaysia

Email: ¹mislam.dipu@gmail.com, ²mokasha@ium.edu.my and ³moumen@iium.edu.my

Abstract. The purpose of the paper is to discuss a comparative evaluation of performance of two different controllers i.e. Proportional-Derivative Controller (PD) and Linear Quadratic Regulation (LQR) in Quadrotor dynamic system that is under-actuated with high nonlinearity. As only four states can be controlled at the same time in the Quadrotor, the trajectories are designed on the basis of the four states whereas three dimensional position and rotation along an axis, known as yaw movement are considered. In this work, both the PD controller and LQR control approach are used for Quadrotor nonlinear model to track the trajectories. LQR control approach for nonlinear model is designed on the basis of a linear model of the Quadrotor because the performance of linear model and nonlinear model around certain nominal point is almost similar. Simulink and MATLAB software is used to design the controllers and to evaluate the performance of both the controllers.

1. Introduction

The popularity of quadrotor is increasing because of its versatile applications. It is popular because of not having any runway rather direct vertical take-off and landing capabilities for flight, hovering on a position constantly, agility, easy manoeuvrability and being small in size gives an additional advantage of monitoring an area or a building [1, 2].

Generally controlling a quadrotor is highly challenging because it has to maintain six degrees of freedom by using four motors [3]. In both classical control theory and modern control theory, a good number of controllers have been developed to deal with multiple inputs and multiple outputs (MIMO) system. Several sorts of control techniques are being designed to improve the performance of a quadrotor while these vary from linear control technique to nonlinear control techniques. A linear model for a quadrotor is achieved by the procedure of linearization of the non-linear model around a nominal point while any hover position is chosen in most of the time. Then all the chosen nominal are summed up to generate the trajectory for the quadrotor.

Linear models are mostly chosen because it is simple to design and the implementation of the model is easy on any real platform. But the main drawback of the linear model is using the linearized model when the controller is being designed otherwise the controller will perform extremely bad [4]. PD controller, LQR controller and H∞ controller are three different well-known linear control systems that are frequently used for quadrotor. Emil Fresk and George Nikolakopoulos [5] presented a solution of attitude control problem of a quadrotor by using PID control. M. Alhaddad et al [6] designed a
mathematical model and then by applying LQR control technique, they achieved proper stability in attitude control.

Some popular nonlinear control techniques for quadrotors are feedback linearization, model predictive control (MPC), Backstepping controller and Sliding mode controller. R. Bonna and J. F. Camino [7] applied feedback linearization to achieve proper trajectory tracking while small angle approximation was avoided when the controller was designed for quadrotor platform. Bouabdullah and Siegwart [8] applied a modified Backstepping control technique, named as Integral Backstepping control method to achieve a flexible control of the quadrotor.

In this work, a comparative discussion of control techniques with performance indices are presented for both LQR control approach and PD control approach on a quadrotor platform. LQR is an optimal control technique of tuning gains for linear systems by minimizing a cost function in an infinite control horizon. Besides, in PD control technique, two different control parameters are used to minimize the error between desired set points and measured process variables. So, the performance for both the controllers can be improved by proper tuning.

2. Dynamic Model
The quadrotor basically holds a rigid cross-linked structure that has four independent rotors with fixed pitched propellers. Three basic movements are necessary to describe all the movements of a quadrotor. The rotational movement around X axis is described as Roll movement ($\phi$) and Y axis is denoted as Pitch movement ($\theta$). Roll movement and pitch movement can be achieved by balancing the speed of motor 2 & 4 and motor 1 & 3 respectively. Lateral and longitudinal acceleration can be achieved respectively by changing $\phi$ and $\theta$ angle.

But another rotational movement Yaw ($\psi$), around Z axis can be achieved by balancing the speed of the motor pair (1, 3) and (2, 4) at the same time.

Every controller input has an effect on a certain movement like $U_2$ effects on roll movement while $U_3$ effects on pitch movement and $U_4$ affects on yaw movement. Besides, $U_1$ has an effect on upward movement along Z axis. So, the control inputs are as followed.

$$U_1 = k_f(\Omega_1^2 + \Omega_2^2 + \Omega_3^2 + \Omega_4^2)$$
$$U_2 = k_f(\Omega_4^2 - \Omega_2^2)$$
$$U_3 = k_f(-\Omega_3^2 + \Omega_1^2)$$
$$U_4 = k_M(\Omega_1^2 - \Omega_2^2 + \Omega_3^2 - \Omega_4^2)$$
Where, 
\( \Omega_i \) represents the \( i^{th} \) motor speed

The initial conditions and nominal parameters for simulation of the quadrotor are shown in table 1 [9].

**Table 1. Parameters and initial conditions for simulation.**

| Symbol | Description                              | Value                  | Unit       |
|--------|------------------------------------------|------------------------|------------|
| \( I \) | Moment of inertia                        | \( \begin{pmatrix} 7.5e-3 & 0 & 0 \\ 0 & 7.5e-3 & 0 \\ 0 & 0 & 1.3e-2 \end{pmatrix} \) | kg.m\(^2\) |
| \( l \) | Arm length                               | 0.23                   | M          |
| \( I_r \) | Inertia of motor                         | 6e-5                   | kg.m\(^2\) |
| \( k_f \) | Thrust coefficient                       | 3.13e-5                | Ns\(^2\)   |
| \( k_M \) | Moment coefficient                       | 7.5e-7                 | Nms\(^2\)  |
| \( m \) | Mass of quadrotor                        | 0.65                   | Kg         |
| \( g \) | Gravity                                  | 9.81                   | ms\(^2\)   |
| \( k_t \) | Aerodynamic thrust drag coefficient       | \( \begin{pmatrix} 0.1 & 0 & 0 \\ 0 & 0.1 & 0 \\ 0 & 0 & 0.1 \end{pmatrix} \) | Ns/m       |
| \( k_r \) | Aerodynamic moment drag coefficient       | \( \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} \) | Nm.s       |

2.1. Transformation Matrix

Since it is required to deal with two different coordinate systems i.e. Earth fixed frame and Body fixed frame simultaneously in order to explain the position and movement of a quadrotor, a transformation matrix is to be derived. Here, \( R \) is the required transformation matrix that can describe the position and movement from Earth-inertial frame to Body fixed frame [10].

\[
R = \begin{pmatrix}
c\theta c\phi & c\phi s\theta & c\phi s\psi + s\theta c\psi \\
c\theta s\phi & s\phi s\theta + c\phi c\psi & c\phi s\theta s\psi - s\theta c\psi \\
-s\theta & c\phi c\theta & s\phi c\theta
\end{pmatrix}
\]

where, \( c \) denotes \( \cos \) and \( s \) denotes \( \sin \).

Generally on-board Inertial Measurement Unit (IMU) is used to measure the angular velocity of a quadrotor and it is measured on Body fixed frame. To generate another relationship between Euler rates, \( [\phi, \dot{\theta}, \dot{\psi}]^T \) that describes the angular rates with respect to time on Earth fixed frame and angular velocity, \( [p, q, r]^T \) of quadrotor, a new rotational matrix \( R_r \) is to be derived again, as follows [11]:

\[
\begin{pmatrix}
p \\ q \\ r
\end{pmatrix} =
\begin{pmatrix}
1 & s\phi t\theta & c\phi t\theta \\
0 & c\phi & -s\phi \\
0 & s\theta & c\theta
\end{pmatrix}
\begin{pmatrix}
\phi \\ \theta \\ \psi
\end{pmatrix}
\]

(1)
2.2. Translational Motion
Gravitational force \((F_g)\) and aerodynamic drag force \((F_a)\) are to be introduced in order to describe the translational movement of a quadrotor though these forces must be overcome by the motors’ thrust \((F_t)\) to achieve any horizontal movement or yaw movement. Here, the translational motion of the quadrotor is described by Newton’s second law as follows.

\[
m \begin{pmatrix} \ddot{x} \\ \ddot{y} \\ \ddot{z} \end{pmatrix} = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & -U_1 \\ 0 & 0 & 0 \end{pmatrix} - k_t \begin{pmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \end{pmatrix}
\]

(2)

2.3. Rotational Motion
Newton-Euler method is used to get rotational equations of motion for the quadrotor [12].

\[
I \dot{\omega} = - \begin{pmatrix} p \\ q \\ r \end{pmatrix} \times I \begin{pmatrix} p \\ q \\ r \end{pmatrix} - \begin{pmatrix} p \\ q \\ r \end{pmatrix} \times \begin{pmatrix} 0 \\ 0 \\ I_r \omega_r \end{pmatrix} + \begin{pmatrix} lU_2 \\ lU_3 \\ U_4 \end{pmatrix} - k_r \begin{pmatrix} p \\ q \\ r \end{pmatrix}
\]

(3)

The dynamic model of a quadrotor can be described by 12 states.

\[
X_s = [x \ y \ z \ \dot{x} \ \dot{y} \ \dot{z} \ \phi \ \theta \ \psi \ p \ q \ r]^T
\]

(4)

A state-space representation can be achieved as follows by considering from equation (2) to (3) for dynamic model.

3. Control Techniques

3.1. LQR (Linear Quadratic Regulator)
LQR is one of the most popular optimal control techniques where a cost function is used that is dependent on the states of any system and control inputs. As LQR needs a linearized model, the dynamic model of the quadrotor is linearized by Jacobian method as follows:

\[
\delta \dot{X}_s = A \delta X_s + B \delta U
\]

If the model is linearized at an nominal point \((X_{ss}, U_{ss})\) where \(X_{ss} = [x_{ss}, y_{ss}, z_{ss}, \psi_{ss}]^T\) and others states are considered as zero with \(U_{ss} = [mg \ 0 \ 0 \ 0]^T\)

Where,

\[
\delta X = X_d - X_a
\]

\[
U = U_{ss} + \delta U
\]

\[
X = X_{ss} + \delta X
\]

(5)

(6)

Now according to the control approach of LQR, a feedback control is to be designed by following equation.

\[
U = -K \delta X_s + U_{ss}
\]

(7)

Where, \(K\) is the feedback gain matrix. It has been calculated by minimizing the cost function

\[
J = \int_{0}^{\infty} (\delta X_s Q \delta X_s + \delta U R \delta U) dt
\]

(8)
Where, $Q$ is considered as a $12 \times 12$ semi-positive definite matrix and $R$ is a $12 \times 4$ positive definite matrix for this quadrotor.

So, the closed loop control system appears as

$$\delta \dot{X}_s = (A - BK)\delta X_s$$

where, $K$ has been calculated by using $Q$ and $R$ matrices.

Then the linear model is replaced by nonlinear model to have proper control of quadrotor with the same gain $K$.

### 3.2. PD Controller

PID control approach is one of the most popular control approaches because it is very simple to design. For this model, integrator part of PID controller is avoided because the model contains integrator twice. Therefore, the general model for a PD controller will be as follows:

$$e(t) = X_d - X_a$$

$$u(t) = K_p e(t) + K_d \frac{d}{dt} e(t)$$

Where $e(t)$ symbolizes the error between desired states ($X_d$) and actual states ($X_a$), $u(t)$ represents the control output $K_p$ denotes proportional gain and $K_d$ is derivative gain.

In this system, four different control inputs $U$ are achieved by PD controller while $U_2$, $U_3$ and $U_4$ is directly solved by PD control technique and $U_1$ is derived from the equation of $\ddot{z}_d$ that is also solved by PD control technique as follows.

$$U_1 = m \left[ g - \frac{(\ddot{x}_d + \frac{k_{tz}}{m} \dot{z}_d)}{c\phi c\theta} \right]$$

$$U_2 = K_p (\phi_d - \phi_a) + K_a (\dot{\phi}_d - \dot{\phi}_a)$$

$$U_3 = K_p (\theta_d - \theta_a) + K_a (\dot{\theta}_d - \dot{\theta}_a)$$

$$U_4 = K_p (\psi_d - \psi_a) + K_a (\dot{\psi}_d - \dot{\psi}_a)$$

Where, desired roll $\theta_d$ and desired pitch $\phi_d$ are derived from the equation (2) as follows.

$$\theta_d = \tan^{-1} \left( \frac{-c\psi \left( \ddot{x}_d + \frac{k_{tx}}{m} \dot{x}_d \right) - s\psi \left( \ddot{y}_d + \frac{k_{ty}}{m} \dot{y}_d \right)}{g - \left( \ddot{z}_d + \frac{k_{tz}}{m} \dot{z}_d \right)} \right)$$

$$\phi_d = \tan^{-1} \left( \frac{-s\psi \left( \ddot{x}_d + \frac{k_{tx}}{m} \dot{x}_d \right) + c\phi \left( \ddot{y}_d + \frac{k_{ty}}{m} \dot{y}_d \right) c\theta_d}{g - \left( \ddot{z}_d + \frac{k_{tz}}{m} \dot{z}_d \right)} \right)$$

### 4. Simulation Results

The two control technique i.e. PD controller and LQR have been implemented on nonlinear model of the quadcopter respectively. The performances for both the controllers are shown in 3D plot for circular and helix trajectory in figure 4 and 5 respectively.
Figure 2. Circular Trajectory.  

Figure 3. Helix Trajectory.  

The suitable gains which are tuned upon several iterations for PD controller are mention in table 2.

Table 2. PD controller gains.

| States | $K_p$ | $K_d$ |
|--------|-------|-------|
| x      | 2.2   | 2.8   |
| y      | 2.2   | 2.8   |
| z      | 7.5   | 5.0   |
| $\phi$ | 141.0 | 20.0  |
| $\theta$ | 5.5   | 1.5   |
| $\psi$ | 141.0 | 20.0  |

And the tuned Q and R matrices are given as follows for LQR control techniques in table 3.

Table 3. Q and R matrices for LQR controller

| Q | R |
|---|---|
| 21 0 0 0 0 0 0 0 0 0 | 0 0 0 0 |
| 0 14 0 0 0 0 0 0 0 0 | 0 0 0 0 |
| 0 0 81 0 0 0 0 0 0 0 | 0 0 0 0 |
| 0 0 0 6 0 0 0 0 0 0 | 0 0 0 0 |
| 0 0 0 0 6 0 0 0 0 0 | 0 0 0 0 |
| 0 0 0 0 0 6 0 0 0 0 | 0 0 0 0 |
| 0 0 0 0 0 0 11 0 0 0 | 0 0 0 0 |
| 0 0 0 0 0 0 16 0 0 0 | 0 0 0 0 |
| 0 0 0 0 0 0 0 56000 0 0 | 0 0 0 1 |
| 0 0 0 0 0 0 0 0 9 0 | 0 0 0 6 |
| 0 0 0 0 0 0 0 0 0 6 | 0 0 0 6 |

Figure 6 and 7, depict the performance of the dynamic model for both the controllers through position tracking ($x$, $y$, $z$) where figure 6 and 7 describe circular trajectory and helix trajectory respectively.

Root-Mean-Square (RMS) error helps to know about the accuracy of actual data with respect to desired data. So, the performance of both the controller can be evaluated by using RMS error while table 2 shows comparison for circular trajectory and table 3 shows for helix trajectory.
Figure 4. $x$ vs $t$, $y$ vs $t$ and $z$ vs $t$ in circular trajectory

Figure 5. $x$ vs $t$, $y$ vs $t$ and $z$ vs $t$ in helix trajectory

From table 4, it is found that LQR performs better comparatively PD controller along x and z axes while PD controller shows better performance along y axis than LQR.

Table 4. RMS Error of the controllers for circular trajectory

|       | x (%) | y (%) | z (%) |
|-------|-------|-------|-------|
| RMSE for PD | 6.28  | 3.05  | 0.21  |
| RMSE for LQR | 3.38  | 3.11  | 0.21  |

Table 5. RMS Error of the controllers for helix trajectory

|       | x (%) | y (%) | z (%) |
|-------|-------|-------|-------|
| RMSE for PD | 5.15  | 3.69  | 2.40  |
| RMSE for LQR | 2.40  | 3.26  | 2.03  |

From the result of table 5, between these two controllers along x, y and z axes tracking in helix trajectory, LQR shows better performance comparing to PD control technique.

5. Conclusion
The work introduces with the two different control approaches with their singular characteristics that where these control approaches are used because of wide popularity and design simplicity. In general,
LQR control approach shows robustness and generates very low steady-state error but keeping update twelve states at the same time may create a transitional delay whereas the fast response is important during flight and the states have no direct access to the plant. However, PD controller can give faster response and overcome steady-state error by adding an additional integral part in controller but it is failed to give robust performance like LQR control approach.

Looking back at the responses, the controllers could not give exact performances as these are supposed to follow the trajectories because of not tuning proper gains while it is the toughest part of designing controllers perfectly.

The gains for PD controllers are tuned by genetic algorithm. As it is difficult to handle ten gains for GA optimization simultaneously, the optimizer tuned the best possible gains as it could. However, if the optimization can be done iteratively, it is possible to tune the perfect gains. On the other hand, Q and R matrices are chosen arbitrarily to calculate gains for LQR control approach and it also can be improved by several processes like Bryson’s rule or optimization etc. If the proper gains can be tuned properly, the settling time and rise time can be decreased more.

In the future work, the designed controllers will be implemented on real quadrotor platform considering constraints at control inputs that can help to comprehend and evaluate the most suitable controllers for the quadrotor practically.

6. References
[1] J. Li and Y. Li, "Dynamic analysis and PID control for a quadrotor," in Mechatronics and Automation (ICMA), 2011 International Conference on, 2011, pp. 573-578.
[2] S. Bouabdallah and R. Siegwart, "Backstepping and sliding-mode techniques applied to an indoor micro quadrotor," in Robotics and Automation, 2005. ICRA 2005. Proceedings of the 2005 IEEE International Conference on, 2005, pp. 2247-2252.
[3] M. Chipofya, D. J. Lee, and K. T. Chong, "Trajectory tracking and stabilization of a quadrotor using model predictive control of Laguerre functions," in Abstract and Applied Analysis, 2015.
[4] A. Chovancová, T. Fico, P. Hubinský, and F. Duchoň, "Comparison of various quaternion-based control methods applied to quadrotor with disturbance observer and position estimator," Robotics and Autonomous Systems, vol. 79, pp. 87-98, 2016.
[5] E. Fresk and G. Nikolakopoulos, "Full quaternion based attitude control for a quadrotor," in Control Conference (ECC), 2013 European, 2013, pp. 3864-3869.
[6] M. Alhaddad, M. Tyupkin, and L. Aeshina, "MODELING AND ATTITUDE CONTROL OF A QUADROPTER USING LINEAR QUADRATIC REGULATOR," Молодежь. Общество. Современная наука, техника и инновации, pp. 173-176, 2016.
[7] R. Bonna and J. Camino, "Trajectory Tracking Control of a Quadrotor Using Feedback Linearization," in Proc. of the XVII International Symposium on Dynamic Problems of Mechanics DINAME-2015.–2015.
[8] S. Bouabdallah and R. Siegwart, "Full control of a quadrotor," in Intelligent robots and systems, 2007. IROS 2007. IEEE/RSJ international conference on, 2007, pp. 153-158.
[9] S. Bouabdallah, "Design and control of quadrotors with application to autonomous flying," 2007.
[10] A. L. Salih, M. Moghavvemi, H. A. Mohamed, and K. S. Gaeid, "Modelling and PID controller design for a quadrotor unmanned air vehicle," in Automation Quality and Testing Robotics (AQTR), 2010 IEEE International Conference on, 2010, pp. 1-5.
[11] H. Bolandi, M. Rezaei, R. Mohsenipour, H. Nemati, and S. M. Smailzadeh, "Attitude control of a quadrotor with optimized PID controller," Intelligent Control and Automation, vol. 4, p. 335, 2013.
[12] L. E. Romero, D. F. Pozo, and J. A. Rosales, "Quadcopter stabilization by using PID controllers," Maskana, pp. 175-186, 2016.