Dynamics and control of quadcopter using linear model predictive control approach

To cite this article: M Islam *et al* 2017 *IOP Conf. Ser.: Mater. Sci. Eng.* **270** 012007

IOP Conf. Series: Materials Science and Engineering **270** (2017) 012007 doi:10.1088/1757-899X/270/1/012007

Dynamics and control of quadcopter using linear model predictive control approach

M Islam, M Okasha* and M M Idres

Department of Mechanical Engineering, International Islamic University Malaysia, Kuala Lumpur, Malaysia

*mokasha@iium.edu.my

Abstract. This paper investigates the dynamics and control of a quadcopter using the Model Predictive Control (MPC) approach. The dynamic model is of high fidelity and nonlinear, with six degrees of freedom that include disturbances and model uncertainties. The control approach is developed based on MPC to track different reference trajectories ranging from simple ones such as circular to complex helical trajectories. In this control technique, a linearized model is derived and the receding horizon method is applied to generate the optimal control sequence. Although MPC is computer expensive, it is highly effective to deal with the different types of nonlinearities and constraints such as actuators' saturation and model uncertainties. The MPC parameters (control and prediction horizons) are selected by trial-and-error approach. Several simulation scenarios are performed to examine and evaluate the performance of the proposed control approach using MATLAB and Simulink environment. Simulation results show that this control approach is highly effective to track a given reference trajectory.

1. Introduction

The interest in quadcopter has gradually increased among researchers due to its structural simplicity and flexibility in flight, in addition to its versatile applications in both civil and military areas [1]. In conjunction to this, it is vital to operate the quadcopter with proper control to gain better performance from it. An effective controller ensures its smooth and collision-free flight in the complex environment considering aerodynamic drag and moments [2]. In literatures, the quadcopter's control problem has been widely investigated using several control approaches including proportional-integral-derivative (PID), linear quadratic regulator (LQR) and H-infinity for linear control system. In the meantime, for nonlinear control system, approaches like backstepping, feedback linearization and model predictive control are applied. The comparison between PID and LQR control techniques on micro quadcopter has been demonstrated in a previous study and the system is shown to be stabilized around the hover position but PID shows poor performance at different operating points [3, 4]. Moreover, another study has applied the backstepping control approach based on Lyapunov theory to stabilize the quadcopter to track a given desired position and attitude. In that work, an under-actuated subsystem is introduced to control the horizontal position through roll and pitch angles while a fully-actuated subsystem is used to control the vertical position through yaw and a propeller subsystem to control propeller forces [5]. Additionally, another study used feedback linearization for trajectory tracking to control rotational and translational dynamics [6].

Model Predictive Control (MPC) becomes one of the widespread controllers nowadays because of its capability in working with constraints and disturbances, predictive behaviour, simplicity in tuning and advanced performance with multi-variable at the same time. It is considered as a nonlinear control system that works on predicting future states and errors [7]. MPC has been used to track the reference trajectory considering disturbances and nonlinear H-infinity to obtain the robustness of the system in quadcopter [8]. In a previous study, MPC is applied to gain robust performance from the system under wind-gust disturbance condition for attitude reference tracking in quadcopter [9]. In that work, MPC has successfully tracked the reference point using a single MPC technique on the quadcopter platform that considers external disturbances in the system and constraints for the actuators saturation at control inputs. The study presented in this paper investigates the quadcopter's dynamics and control using the MPC approach.

2. System Design

Quadcopter basically holds a rigid cross-linked structure that has four independent rotors with fixed pitched propellers. Among the four propellers, two are rotating in clockwise direction while the other two rotate in anti-clockwise direction, as shown in Figure 1. The control of quadcopter is obtained by changing the angular speed of the propellers Ω_i (i = 1, 2, 3, 4). The rotational movement of quadcopter along X, Y and Z axes can be described by roll (ϕ), pitch (θ) and yaw angle (ψ).

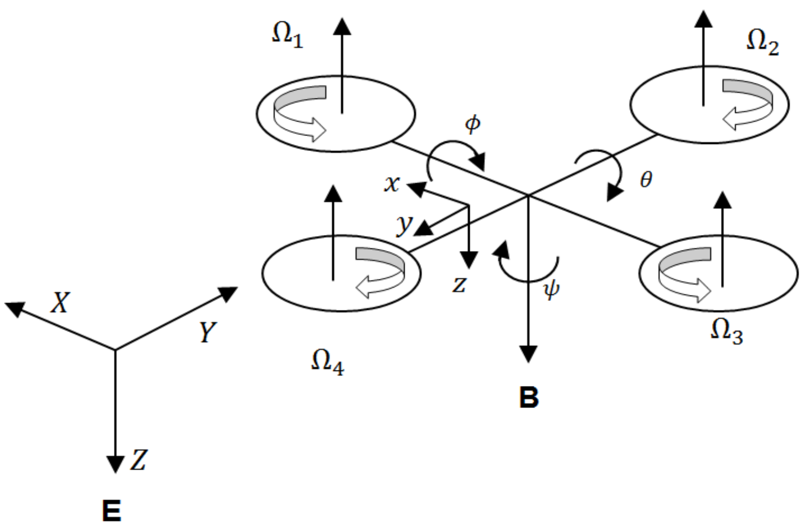


Figure 1: Configuration of quadcopter, where **B** and **E** denote the body fixed frame and the Earth fixed frame, respectively

Every controller input has an effect on a certain movement such as u_2 affects on roll movement, u_3 affects on pitch movement, u_4 affects on yaw movement and u_1 has an effect on upward movement along Z axis. Here, as '+' (plus) configuration is chosen for this quadcopter, the control inputs produce the effects on the system as described in Equation 1 to Equation 4. The initial conditions and nominal parameters for simulation of the quadcopter are shown in Table 1 [10].

$$u_1 = k_f(\Omega_1^2 + \Omega_2^2 + \Omega_3^2 + \Omega_4^2) \quad (1)$$

$$u_2 = k_f(\Omega_4^2 - \Omega_2^2) \quad (2)$$

$$u_3 = k_f(-\Omega_3^2 + \Omega_1^2) \quad (3)$$

$$u_4 = k_M(\Omega_1^2 - \Omega_2^2 + \Omega_3^2 - \Omega_1^2) \quad (4)$$

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 quadcopter	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	$\begin{pmatrix} 0.1 & 0 & 0 \\ 0 & 0.1 & 0 \\ 0 & 0 & 0.1 \end{pmatrix}$	Nm.s

The mathematical model of the quadcopter is given by Equation 5 to Equation 10 [11-13].

$$\ddot{x} = \frac{-1}{m}[k_{t_x}\dot{x} + u_1(sin\phi sin\psi + cos\phi cos\psi sin\theta)] \tag{5}$$

$$\ddot{y} = \frac{-1}{m}[k_{t_y}\dot{y} + u_1(sin\phi cos\psi - cos\phi sin\psi sin\theta)] \tag{6}$$

$$\ddot{z} = \frac{-1}{m}[k_{t_z}\dot{z} - mg + u_1 cos\phi cos\theta] \tag{7}$$

$$\dot{p} = \frac{-1}{I_x}[k_{r_x}p - lu_2 - I_y qr + I_z qr + I_r q\omega_r] \tag{8}$$

$$\dot{q} = \frac{-1}{I_y}[-k_{r_y}q + lu_3 - I_x pr + I_z pr + I_r p\omega_r] \tag{9}$$

$$\dot{r} = \frac{-1}{I_z}[u_4 - k_{r_z}r + I_x pq - I_y pq] \tag{10}$$

Another kinematic relationship is required between Euler rates, $[\dot{\phi}, \dot{\theta}, \dot{\psi}]^T$ on the earth fixed frame and angular velocity, $[p, q, r]^T$ of the quadcopter to describe the whole complete system. This is given by Equation 11 to Equation 13 [14].

$$\dot{\phi} = p + rcos\phi tan\theta + qsin\phi tan\theta \tag{11}$$

$$\dot{\theta} = qcos\phi - rsin\phi \tag{12}$$

$$\dot{\psi} = r\frac{cos\phi}{tan\theta} + q\frac{sin\phi}{cos\theta} \tag{13}$$

Therefore, the complete dynamic model of the quadcopter can be described by four control inputs, $u = [u_1 u_2 u_3 u_4]^T$ and 12 state vectors, $x_s = [x\ y\ z\ \dot{x}\ \dot{y}\ \dot{z}\ \phi\ \theta\ \psi\ p\ q\ r]^T$.

IOP Conf. Series: Materials Science and Engineering **270** (2017) 012007 doi:10.1088/1757-899X/270/1/012007

3. Controller Design

MPC, or known as receding horizon control (RHC), is a control approach that comprises a systematic algorithm where the dynamic model of the system is solved under a finite, moving horizon and closed control problem. It has the ability to use constraints in both control inputs and outputs on the system during design process. It basically predicts a number of outputs of the system such that it can generate an optimized control effort for the system to reach the reference trajectory. The optimization problem is solved for a predefined time interval that is also known as prediction horizon at each sampling time interval. The immediate optimized control signal is applied in the system until the next sampling time interval and this process is repeated for each sampling time interval [15].

3.1. Plant Model and Prediction Horizon

A nonlinear system can be written as in Equation 14, where $x(t)\epsilon R^n$ denotes the states of the system and $u(t)\epsilon R^m$ denotes system inputs.

$$\dot{x} = f((x), u(t)) \tag{14}$$

The quadcopter's dynamic model is linearized at hover condition as in Equation 15 to Equation 18 [15, 17], where the nominal states and control inputs are x_T and u_T, respectively, k is the sample time, $A \epsilon R^{n\times n}$ is the state matrix, $B \epsilon R^{n\times m}$ is input matrix, $y \epsilon R^p$ is system outputs, $C \epsilon R^{p\times n}$ is output matrix and $D \epsilon R^{p\times m}$ is feedforward matrix. For this system, $n = 12$ and $m = 4$ are considered.

$$\Delta x_{k+1} = A\Delta x_k + B\Delta u_k \tag{15}$$

$$\Delta y_k = C\Delta x_k + D\Delta u_k \tag{16}$$

$$\Delta x_k = x_k - x_T \tag{17}$$

$$\Delta u_k = u_k - u_T \tag{18}$$

A prediction horizon has to be determined such that the controller can predict a number of future states to reach the desired states. A state observer with an estimator is also required to be implemented in the controller to predict the future states while the estimator predicts future behaviour of the system. A Linear Quadratic Estimator is applied for the algorithm. By expanding Equation 15 and Equation 16 up to $k + N$, the future states and outputs can be achieved depending on initial states and future inputs as given by Equation 19 and Equation 20, respectively.

$$\Delta x_{k+N} = A^N\Delta x_k + A^{N-1}B\Delta u_k + A^{N-2}B\Delta u_{k+1} + \cdots + AB\Delta u_{k+N-2} + B\Delta u_{k+N-1} \tag{19}$$

$$\Delta y_{k+N} = CA^N\Delta x_k + C(A^{N-1}B\Delta u_k + A^{N-2}B\Delta u_{k+1} + \cdots + AB\Delta u_{k+N-2} + B\Delta u_{k+N-1}) \tag{20}$$

The equations can also be written in matrix form as shown by Equation 21 and Equation 22.

$$\begin{pmatrix} \Delta x_k \\ \Delta x_{k+1} \\ \Delta x_{k+2} \\ \vdots \\ \Delta x_{k+N-1} \end{pmatrix} = \begin{pmatrix} I \\ A \\ A^2 \\ \vdots \\ A^{N-1} \end{pmatrix} \Delta x_k + \begin{pmatrix} 0 & 0 & \cdots & 0 & 0 \\ B & 0 & \cdots & 0 & 0 \\ AB & B & \cdots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ A^{N-2}B & A^{N-3}B & \cdots & B & 0 \end{pmatrix} \begin{pmatrix} \Delta u_k \\ \Delta u_{k+1} \\ \Delta u_{k+2} \\ \vdots \\ \Delta u_{k+N-1} \end{pmatrix} \tag{21}$$

$$\begin{pmatrix} \Delta y_k \\ \Delta y_{k+1} \\ \Delta y_{k+2} \\ \vdots \\ \Delta y_{k+N-1} \end{pmatrix} = \begin{pmatrix} C & 0 & 0 & \cdots & 0 \\ 0 & C & 0 & \cdots & 0 \\ 0 & 0 & C & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & C \end{pmatrix} \begin{pmatrix} \Delta x_k \\ \Delta x_{k+1} \\ \Delta x_{k+2} \\ \vdots \\ \Delta x_{k+N-1} \end{pmatrix} + \begin{pmatrix} D & 0 & 0 & \cdots & 0 \\ 0 & D & 0 & \cdots & 0 \\ 0 & 0 & D & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & D \end{pmatrix} \begin{pmatrix} \Delta u_k \\ \Delta u_{k+1} \\ \Delta u_{k+2} \\ \vdots \\ \Delta u_{k+N-1} \end{pmatrix} \tag{22}$$

As D = 0 in most of the cases, the two equations can be rewritten as in Equation 23 and Equation 24 in a shorter form.

$$\Delta X_k = A_m \Delta x_k + B_m \Delta u_k \tag{23}$$

$$\Delta Y_k = C_m \Delta x_k \tag{24}$$

3.2. Control Design

The MPC technique is helped by cost function in its control algorithm to calculate the optimal solution at every sampling time interval. The cost function is generally designed in a way that the predicted outputs are directed towards the desired states as described in Equation 20 while the control efforts are minimized as well. In this study, the cost function is minimized by the norm of the difference between the current outputs and desired trajectory and the norms of motor inputs as in Equation 25, where $\widehat{W}_u$ and $\widehat{W}_y$ are given by Equation 26 and Equation 27, respectively [16].

$$J(\Delta x, \Delta u) = (\Delta u_k)^T \widehat{W}_u{}^2 (\Delta u_k) + (\Delta Y_k - \Delta Y_k^r)^T \widehat{W}_y{}^2 (\Delta Y_k - \Delta Y_k^r) \tag{25}$$

$$\widehat{W}_u = \begin{bmatrix} W_{u|0,1} & 0 & \cdots & 0 & \cdots & 0 & 0 & \cdots & 0 \\ 0 & W_{u|0,2} & \cdots & 0 & \ddots & 0 & 0 & \ddots & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \vdots & \cdots & \vdots \\ 0 & 0 & \cdots & W_{u|0,m} & \cdots & 0 & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & \cdots & 0 & \cdots & W_{u|N-1,1} & 0 & \cdots & 0 \\ 0 & 0 & \cdots & 0 & \cdots & 0 & W_{u|N-1,2} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 0 & \cdots & 0 & 0 & \cdots & W_{u|N-1,m} \end{bmatrix} \tag{26}$$

$$\widehat{W}_y = \begin{bmatrix} W_{y|0,1} & 0 & \cdots & 0 & \cdots & 0 & 0 & \cdots & 0 \\ 0 & W_{y|0,2} & \cdots & 0 & \ddots & 0 & 0 & \ddots & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \vdots & \cdots & \vdots \\ 0 & 0 & \cdots & W_{y|0,m} & \cdots & 0 & 0 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & \cdots & 0 & \cdots & W_{y|N-1,1} & 0 & \cdots & 0 \\ 0 & 0 & \cdots & 0 & \cdots & 0 & W_{y|N-1,2} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 0 & \cdots & 0 & 0 & \cdots & W_{y|N-1,m} \end{bmatrix} \tag{27}$$

3.3. Quadratic Programming

As the cost function is in quadratic form, a quadratic programming is chosen to solve the optimization problem. The main purpose of the quadratic programming is to reduce the cost function $J(\Delta x, \Delta u)$ by finding out a feasible search direction, Δu.

3.4. Input and Constraint Handling

During the design of the quadcopter, it is important to apply constraint at the force of each motor. This is to enable the motors to be operated between maximum and minimum rotations per minute (rpm). There is an upper bound, u_{ub} and a lower bound, u_{lb} at the control inputs where $u_{lb} \leq u_{k+i} \leq u_{ub}$ for $i = 0, 1, 2, \ldots, N-1$. As the dynamic model is linearized around a certain operating point, the MPC approach solves the perturbed control inputs for the linearized model. The constraints can be described in matrix form as in Equation 28, where $I_{m\times m}$ is an identity matrix.

$$\begin{bmatrix} \mathrm{I_{m\times m}} \\ -\mathrm{I_{m\times m}} \end{bmatrix} \Delta \mathrm{u_{k+i}} \le \begin{bmatrix} \mathrm{u_{ub}} - \Delta \mathrm{u_T} \\ -(\mathrm{u_{lb}} - \Delta \mathrm{u_T}) \end{bmatrix} \tag{28}$$

After rearranging, Equation 28 can be rewritten as Equation 29, where $\boldsymbol{I_u}$ is given by Equation 30.

$$I_u \Delta u_k \le \Delta u_b \tag{29}$$

$$I_u = \begin{bmatrix} \begin{bmatrix} I_{m\times m} \\ -I_{m\times m} \end{bmatrix} & 0 & \cdots & 0 \\ 0 & \begin{bmatrix} I_{m\times m} \\ -I_{m\times m} \end{bmatrix} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \begin{bmatrix} I_{m\times m} \\ -I_{m\times m} \end{bmatrix} \end{bmatrix}, \Delta u_b = \begin{bmatrix} \begin{bmatrix} u_{ub} - \Delta u_T \\ -(u_{lb} - \Delta u_T) \end{bmatrix} \\ \begin{bmatrix} u_{ub} - \Delta u_T \\ -(u_{lb} - \Delta u_T) \end{bmatrix} \\ \vdots \\ \begin{bmatrix} u_{ub} - \Delta u_T \\ -(u_{lb} - \Delta u_T) \end{bmatrix} \end{bmatrix} \tag{30}$$

It is also necessary to consider a limit on the angles to avoid kinematic singularities because of the limitations of the model. The angles are limited within the bounds and the bounds are given as follows for roll, pitch and yaw, respectively: $-\pi \le \phi \le \pi$, $-\frac{\pi}{2} \le \theta \le \frac{\pi}{2}$ and $-\pi \le \psi \le \pi$.

If a specific output is to be constrained, it can be described as Equation 31 and the constraints can be represented by Equation 32, where z_{lb} and z_{ub} are denoted as lower bound and upper bound for the outputs.

$$\Delta z_{lb} = C_z \Delta x_k \tag{31}$$

$$z_{lb} \le C_z x_{k+i} \le z_{ub};\ i = 0, 1, 2, 3, \dots, N-1 \tag{32}$$

Similarly, it also can be shown in matrix form as in Equation 33.

$$\begin{bmatrix} C_z \\ -C_z \end{bmatrix} \le \begin{bmatrix} z_{ub} - C_z x_T \\ -(z_{lb} - C_z x_T) \end{bmatrix} \tag{33}$$

From Equation 23 and Equation 24, the constraints can be described as in Equation 34 where Δx_k is substituted and Γ_z is given in Equation 35.

$$\Gamma_z (A_m \Delta x_k + B_m \Delta u_k) \le \Delta z_b \tag{34}$$

$$\Gamma_z = \begin{bmatrix} \begin{bmatrix} C_z \\ -C_z \end{bmatrix} & 0 & \cdots & 0 \\ 0 & \begin{bmatrix} C_z \\ -C_z \end{bmatrix} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \begin{bmatrix} C_z \\ -C_z \end{bmatrix} \end{bmatrix},\ \Delta z_b = \begin{bmatrix} \begin{bmatrix} z_{ub} - C_z x_T \\ -(z_{lb} - C_z x_T) \end{bmatrix} \\ \begin{bmatrix} z_{ub} - C_z x_T \\ -(z_{lb} - C_z x_T) \end{bmatrix} \\ \vdots \\ \begin{bmatrix} z_{ub} - C_z x_T \\ -(z_{lb} - C_z x_T) \end{bmatrix} \end{bmatrix} \tag{35}$$

The control input and output constraints can be described by one single Equation 36, where Π is given in Equation 37 [17].

$$\Pi \Delta u_k \le \Upsilon \tag{36}$$

$$\Pi = \begin{bmatrix} M_u \\ \Gamma_z B_m \end{bmatrix},\ \Upsilon = \begin{bmatrix} \Delta u_b \\ \Delta z_b - \Gamma_z A_m \Delta x_k \end{bmatrix} \tag{37}$$

IOP Conf. Series: Materials Science and Engineering **270** (2017) 012007 doi:10.1088/1757-899X/270/1/012007

4. Simulated Results

As the dynamics of the quadcopter is nonlinear, it has been linearized at a certain point and considered as $[0, 1m, 0, 0, 0, 0, 0, 0, \frac{5\pi}{180}, 0, 0, 0]^T$. For this study, prediction horizon, N = 30 and control horizon, M = 2 are selected after several initial simulations where these two values provided the most efficient performance under the consideration of settling time and overshoot. The effects of different N along x, y and z axes on settling time and overshoot are shown in Figure 2. It can be observed from the figure that the settling time is increasing when N increases while the overshoot is decreasing with increasing N along the x, y and z axes.

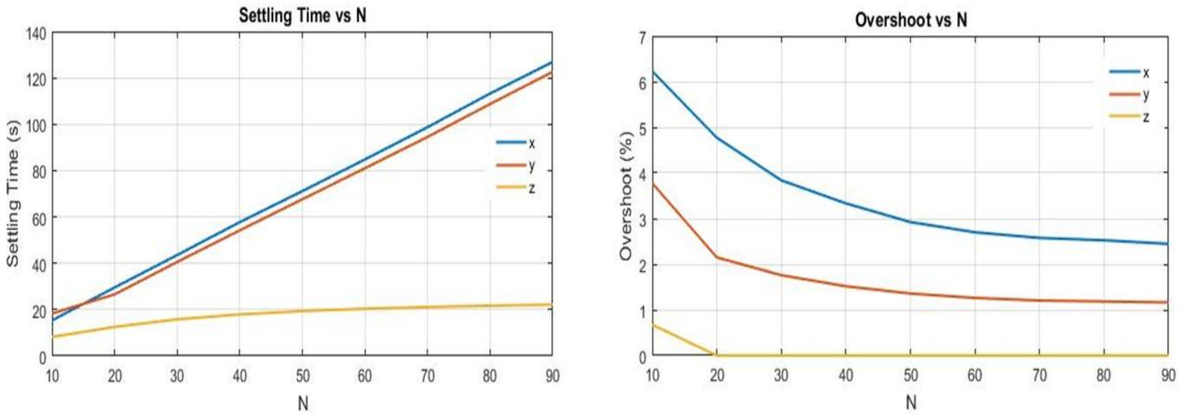


Figure 2: Effects of N on settling time and overshoot

Sample time for the model is chosen as 0.25s after the confirmation of system stability where the pole of the system has been achieved: $[0, 0, 0, -0.1538, -0.1538, -0.1538, 0, 0, 0, 0, 0, 0]^T$. Figure 3 illustrates the efficiency of MPC controller that can reject disturbances without affecting the control outputs. In this scenario, x = 3m, y = 2m and z = -5m is selected as the reference point to show the comparison among the disturbances that range from 0.1 to 1. The RMSE is found to remain exactly the same along x, y and z axes with 5.13%, 5.11% and 3.09%, respectively. This means that the external disturbance has no effect on the system. Moreover, Figure 4 shows the control effort against time for better understanding.

Three different trajectories are chosen for tracking under certain constraints at control inputs and addition of disturbances in the system where the angular velocity of each motor is taken as 848 rad/s. This helps to find the constraints as: $0 < u_1 < 90$, $-22.52 < u_2 < 22.52$, $-22.52 < u_3 < 22.52$ and $-1.08 < u_4 < 1.08$, and the disturbances considered for the four control inputs are [0.1, 0.1, 0.1, 0.1] as in Figure 5.

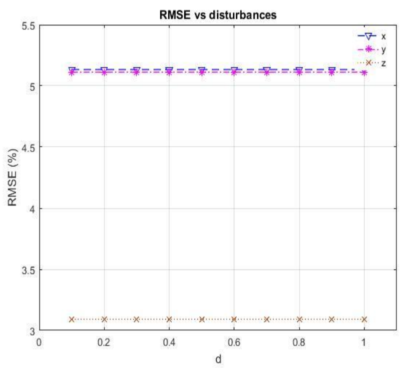


Figure 3: Effects of different disturbances on RMSE

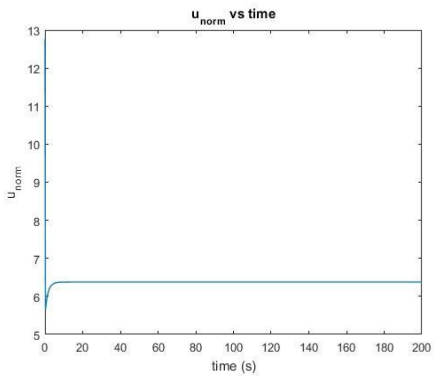
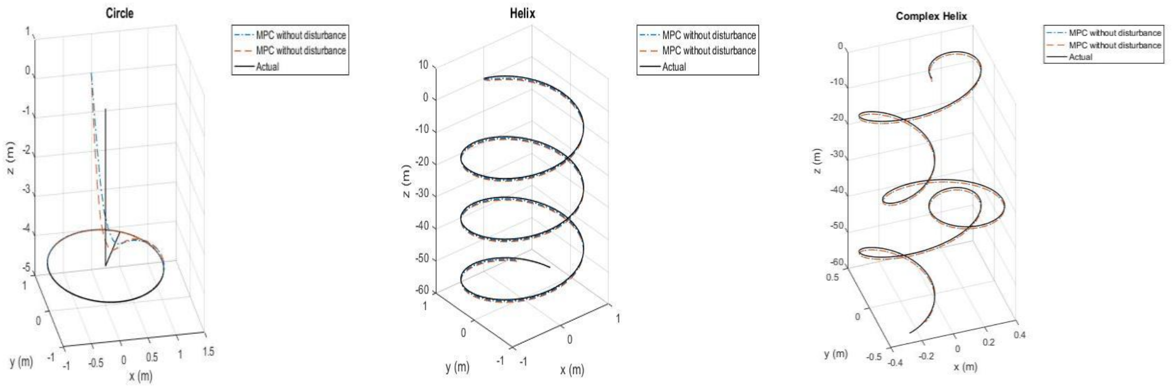


Figure 4: Control efforts (u_{norm}) against time

IOP Conf. Series: Materials Science and Engineering **270** (2017) 012007 doi:10.1088/1757-899X/270/1/012007



(a) (b) (c)

Figure 5: (a) Circular trajectory, (b) helix trajectory and (c) complex helix trajectory

For circular trajectory, $x = sin(0.1t), y = cos(0.1t), z = -4m$

For helix trajectory, $x = sin(0.1t), y = cos(0.1t), z = -0.3t$

For complex helix, $x = cos(0.05t) - cos^3(0.05t),\ y = sin(0.05t) - sin^3(0.05t), z = -0.3t$

Root-Mean-Square (RMS) is an approach to evaluate the accuracy of the data by comparison. The performance of the controller is evaluated using RMS error (RMSE). Table 2 shows the comparison between reference trajectory and achieved trajectory values with respect to time for circular, helix and complex helix trajectory under disturbance and without disturbance. From Table 2, it is found that the RMSE is less than 5% for the three trajectories, which is considered tolerable.

Table 2: RMSE for circle, helix and complex helix trajectories with and without disturbances

	RMSE for with disturbance			**RMSE for without disturbance**		
	x (%)	**y (%)**	**z (%)**	**x (%)**	**y (%)**	**z (%)**
Circle	2.3320	2.8844	0.8866	2.3321	2.8878	1.2029
Helix	2.2526	1.2278	2.1067	2.2524	1.2281	2.9558
Complex Helix	1.0628	2.5308	1.3508	1.0628	2.5308	1.3508

5. Conclusion

The presented work shows the use of Linear Model Predictive Control (LMPC) approach for different trajectories (i.e. circle, helix and complex helix trajectory) under disturbances. It is designed with the help of MPC toolbox in Simulink. The main advantage of MPC controller that make it different from other controllers such as PID, LQR, H-infinity or feedback linearization is the optimization of control inputs and outputs under the consideration of disturbances, noise and constraints. This has been shown to help achieve proper inputs and outputs under certain requirements of the system although the noise factor is not considered in this work. The most crucial issues to design a MPC model include choosing proper prediction horizon, control horizon and sample time because they all affect the system stability. After the confirmation of system stability, the system tracking can be improved by tuning to the proper gains. This study has successfully demonstrated a proper tracking with minimal RMSE under different disturbances that have been shown to be of negligible effect to the system outputs. In future, nonlinear Model Predictive Control (NMPC) approach will be designed that is expected to be more suitable for nonlinear quadcopter model.

References

[1] Zhao W and Go T H 2014 *Journal of the Franklin Institute* **351** 1335-55
[2] Tanveer M H, Hazry D, Ahmed S F, Joyo M K, Warsi F A, Razlan Z M, Wan K and Hussain A T 2014 *Australian Journal of Basic and Applied Sciences* **8** 428-31
[3] Bouabdallah S, Noth A and Siegwart R 2004 *IEEE/RSJ International Conference on Intelligent Robots and Systems*
[4] Pounds P E I 2007 *Design, Construction and Control of A Large Quadrotor Micro Air Vehicle* Australian National University
[5] Madani T and Benallegue A 2006 *IEEE/RSJ International Conference on Intelligent Robots and Systems*
[6] Bonna R and Camino J 2015 *International Symposium on Dynamic Problems of Mechanics*
[7] Kendoul F 2012 *Journal of Field Robotics* **29** 315-78
[8] Raffo G V, Ortega M G and Rubio F R 2008 *International Federation of Automatic Control World Congress*
[9] Alexis K, Nikolakopoulos G and Tzes A 2010 *18th Mediterranean Conference on Control and Automation*
[10] Bouabdallah S 2007 *Design and Control of Quadrotors with Application to Autonomous Flying* Thesis, Ecole Polytechnique Federale de Lausanne
[11] Efe M Ö 2011 *IEEE Transactions on Industrial Informatics* **7** 354-61
[12] Khan H S and Kadri M B 2015 *International Conference on Information and Communication Technologies*
[13] Elkholy H M 2014 *Dynamic Modeling and Control of a Quadrotor using Linear and Nonlinear Approaches* Thesis, American University in Cairo
[14] Bolandi H, Rezaei M, Mohsenipour R, Nemati H and Smailzadeh S M 2013 *Intelligent Control and Automation* **4** 335
[15] Tule C 2014 *Trajectory Generation and Constrained Control of Quadrotors* Thesis, University of Texas Arlington
[16] Mathworks *https://www.mathworks.com/help/mpc/ug/optimization-problem.html*
[17] Ru P and Subbarao K 2017 *Aerospace* **4** 31