International Journal of Engineering & Technology, 7 (4.13) (2018) 125-128

International Journal of Engineering & Technology

Website: www.sciencepubco.com/index.php/IJET

Research paper

Trajectory tracking of quaternion based quadrotor using model predictive control

M Islam[1]*, M Okasha[1], M M Idres[1], H Mansor[2]**

[1]*Department of Mechanical Engineering, International Islamic University Malaysia, Kuala Lumpur, Malaysia*
[2]*Department of Electrical and Computer Engineering, International Islamic University Malaysia, Kuala Lumpur, Malaysia*
**Corresponding author E-mail: mislam.dipu@gmail.com,*

Abstract

The aim of this paper is to introduce the trajectory tracking with a quaternion based quadrotor operation using model predictive control (MPC). Since the efficacy of MPC on a system under noise and disturbance has been distinguished, it is a fair and successful attempt to apply MPC on the quaternion based quadrotor, which is a quite well-known system with uncertainties during operation. Quaternion approaches singularity-free orientation that is advantageous to design any trajectory for quadrotor wherein roll or pitch angle reaches at 90^{o}. As a quaternion, with its four-tuple characteristics that incorporate vector elements, is different from Euler-angle orientation, a new cost function has been developed for the respective MPC controller. In order to achieve singularity-free orientations and abate the model infidelity of the system, the quaternion and MPC algorithm have been incorporated for quadrotor flight. Simulation based results elucidate the success of trajectory tracking of quaternion based dynamics of quadrotor using MPC approach.

Keywords: *cost function; MPC; quadrotor; quaternion; tracking.*

1. Introduction

Researchers have a great interest on quadrotor because of its multi-dimensional applications such as mapping, surveillance, package delivering, exploration, aerial filming, etc. Structurally, quadrotor is developed using four electrical motors that are mounted on at the end of the cross-linked frame with two clockwise as well as two anti-clockwise propellers. However, quadrotor encounters instability with model uncertainty like vibration, noise and disturbance and dynamic model complexity because of its characteristics as an under-actuated system. Furthermore, the overlapping of axes during its agile movement creates a singularity problem and hence quaternion can be the only remedy for it.

Model Predictive Control (MPC) becomes one of the widespread controllers nowadays based on its capability in working with constraints and disturbances, predictive behaviour, simplicity in tuning and advance performance with multi-variable at the same time. It is considered as nonlinear control system that works on predicting the future states and error [1]. MPC controller is more effective and accurate than PID controller that is one of the most popular in industrial areas [2]. MPC works on the base of optimization where the cost function is minimized depending on current control inputs and the future time interval by handling the constraints of states and inputs [3]. Raffo *et al.* (2008) proposed a MPC to track the reference trajectory considering disturbances and nonlinear H-infinity to obtain the robustness of the system in quadrotor [4]. In previous study, MPC is applied to attain robust performance from the system under a wind-gust disturbance condition for the attitude reference tracking in a quadrotor. It has successfully tracked the reference point by using a single MPC technique on the quadrotor platform that considers the external disturbances in the system and the constraints for the actuators saturation at the control inputs [5]. Bouffard (2012) has used Learning Based Model Predictive Control (LBMPC) for robustness and it has been demonstrated that the performance can be improved by updating the model online, and it performs better than linear MPC [3].

This work offers a quaternion-based quadrotor model that is operated by the MPC algorithm. As a result, the control algorithm has some notable differences in comparison to usual MPC model that is described in Section 3.

2. Quadrotor modelling

In order to describe a quadrotor platform with six degrees of freedom (DOF), the basic mathematical equation is used for the derivation. As a result, Newton-Euler formulation has been embraced to do so. The quadrotor requires kinematics and dynamics in order to describe the complete mathematical model that has been subsequently represented in this section.

2.1. Kinematics

Kinematic modelling is fully dependent on the coordinate system. In addition, it is required to explain the orientation of a rigid body with the help of a fixed coordinate system. Therefore, E and B have been emerged to describe earth-fixed frame and body frame, respectively, in Figure 1. However, position of the quadrotor can be described by r = $[x, y, z]^T$ and the rotations along the axes (i.e. roll, pitch and yaw) can be symbolized by $\xi = [\phi, \theta, \psi]^T$, respectively.

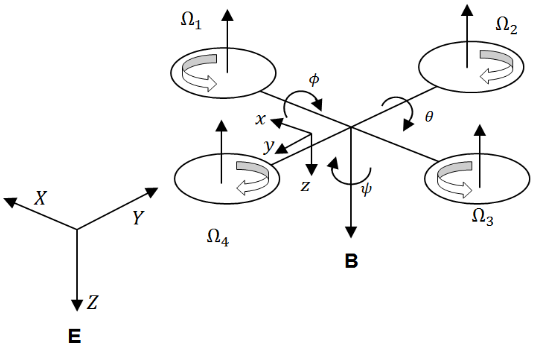

Fig. 1: Configuration of quadrotor where B and E denote the body fixed frame and the Earth fixed frame, respectively

However, the rotational matrix, also known as transformation matrix, Q_{EB} is required to maintain a complementary relation between the coordinate frames whence earth-fixed frame plays a significant role to estimate position along with the body frame and describe thrust and angular velocity of a quadrotor [6]. In addition, the Euler angles, ξ are represented with the quaternion, $q = [q_0, q_1, q_2, q_3]^T$ that has been adopted in the transformation matrix in Eqn. 1.

$$Q_{EB} = \begin{bmatrix} {Q_x}^T \\ {Q_y}^T \\ {Q_z}^T \end{bmatrix} = \begin{bmatrix} 1 - 2({q_2}^2 + {q_3}^2) & 2(q_1q_2 + q_0q_3) & 2(q_1q_3 - q_0q_2) \\ 2(q_1q_2 - q_0q_3) & 1 - 2({q_1}^2 + {q_3}^2) & 2(q_2q_3 + q_0q_1) \\ 2(q_1q_3 + q_0q_2) & 2(q_2q_3 - q_0q_1) & 1 - 2({q_1}^2 + {q_2}^2) \end{bmatrix} \tag{1}$$

Additional kinematics is also required to cooperate quaternion derivative, $\dot{\boldsymbol{q}}$ with angular velocity when angular velocity of quadrotor, $\omega = [p, q, r]^T$ as indicated by Eqn. 2 [7].

$$\dot{\boldsymbol{q}} = \frac{1}{2}\begin{pmatrix} q_0 & -q_1 & -q_2 & -q_3 \\ q_1 & q_0 & -q_3 & q_2 \\ q_2 & q_3 & q_0 & -q_1 \\ q_3 & -q_2 & q_1 & q_0 \end{pmatrix}\begin{pmatrix} 0 \\ p \\ q \\ r \end{pmatrix} \tag{2}$$

2.1. Dynamics

Literatures on the quadrotor highlight three different forces and a moment: u_1 deals with upward and downward forces, u_2 is subjected to rolling, u_3 is liable for pitching and u_4 is a moment that generates yaw movement. The necessary equations that produce the control inputs for the system are as in the following Eqn. 3 to Eqn. 6, where Ω_i is the angular velocity of motor i.

$$u_1 = k_f(\Omega_1^2 + \Omega_2^2 + \Omega_3^2 + \Omega_4^2) \tag{3}$$

$$u_2 = k_f(\Omega_4^2 - \Omega_2^2) \tag{4}$$

$$u_3 = k_f(-\Omega_3^2 + \Omega_1^2) \tag{5}$$

$$u_4 = k_M(\Omega_1^2 - \Omega_2^2 + \Omega_3^2 - \Omega_1^2) \tag{6}$$

Table 1 introduces some of the necessary symbols along with their parameters beforehand and aforementioned as well. Additionally, researchers have also introduced the dynamic model using the Newton's second law and Newton-Euler equation through different literatures as in the following Eqn. 7 and Eqn. 8, where $\dot{\omega}$ is the derivative of angular velocity of quadrotor with respect to time and $\omega_r = -\Omega_1 + \Omega_2 - \Omega_3 + \Omega_4$ [9].

$$m\begin{pmatrix} \ddot{x} \\ \ddot{y} \\ \ddot{z} \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ mg \end{pmatrix} + Q_{EB}\begin{pmatrix} 0 \\ 0 \\ -u_1 \end{pmatrix} - k_t\begin{pmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \end{pmatrix} \tag{7}$$

$$I\dot{\omega} = -\omega \times I\omega - \omega \times \begin{pmatrix} 0 \\ 0 \\ I_r\omega_r \end{pmatrix} + \begin{pmatrix} lu_2 \\ lu_3 \\ u_4 \end{pmatrix} - k_r\begin{pmatrix} p \\ q \\ r \end{pmatrix} \tag{8}$$

Table 1. Parameters and initial conditions for simulation [8]

Symbol	Value	Symbol	Value	Symbol	Value
J_x	$7.5e-3$ kg.m^2	k_{t_x}	0.1 Ns/m	k_{r_x}	0.1 Nm.s
J_y	$7.5e-3$ kg.m^2	k_{t_y}	0.1 Ns/m	k_{r_y}	0.1 Nm.s
J_z	$1.3e-2$ kg.m^2	k_{t_z}	0.15 Ns/m	k_{r_z}	0.15 Nm.s
J_r	6e-5 kg.m^2	k_f	3.13e-5 Ns^2	g	9.81 m/s^2
l	0.23 m	k_M	7.5e-7 Nms^2	m	0.65 kg

Finally, dynamics and kinematics formulation for the complete model of the system are represented by the following Eqn. 9 to Eqn. 18:

$$\ddot{x} = \frac{-1}{m}[k_{t_x}\dot{x} + u_1(2q_0q_2 + 2q_1q_3)] \tag{9}$$

$$\ddot{y} = \frac{-1}{m}[k_{t_y}\dot{y} - u_1(2q_0q_2 - 2q_1q_3)] \tag{10}$$

$$\ddot{z} = \frac{-1}{m}[k_{t_z}\dot{z} - mg + u_1(2{q_0}^2 + 2{q_3}^2 - 1)] \tag{11}$$

$$\dot{q_0} = \frac{1}{2}[-pq_1 - qq_2 - rq_3] \tag{12}$$

$$\dot{q_1} = \frac{1}{2}[pq_0 - qq_3 + rq_2] \tag{13}$$

$$\dot{q_2} = \frac{1}{2}[pq_3 + qq_0 - rq_0] \tag{14}$$

$$\dot{q_3} = \frac{1}{2}[-pq_2 + qq_1 + rq_0] \tag{15}$$

$$\dot{p} = \frac{-1}{I_x}[k_{r_x}p - lu_2 - I_yqr + I_zqr + I_rq\omega_r] \tag{16}$$

$$\dot{q} = \frac{-1}{I_y}[-k_{r_y}q + lu_3 - I_xpr + I_zpr + I_rp\omega_r] \tag{17}$$

$$\dot{r} = \frac{-1}{I_z}[u_4 - k_{r_z}r + I_xpq - I_ypq] \tag{18}$$

3. Model predictive control

As the control system of a quadrotor is multiple input and multiple output (MIMO) and disturbances may be added with the dynamic system, the quadrotor system needs a controller that can handle both conditions. Moreover, it is quite impossible to employ constraints directly on the actual control signals to get optimized solutions because of coupling. So, MPC technique can be an option to overcome all of these problems. MPC, or also 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 a closed control problem. It has the ability to use constraints in both control inputs and outputs on the system during the design process. It basically predicts the number of outputs of the system so 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. The process is repeated for each sampling time interval. In this section, linear MPC control algorithm is described briefly.

3.1. Plant

A discretized linear model of a nonlinear model can be represented by the following Eqn. 19 and Eqn. 20, where x_k and u_k are states and inputs to the system at a sampling time, k.

$$x_{k+1} = Ax_k + Bu_k \tag{19}$$

$$y_k = C_m x_k + D_m u_k \tag{20}$$

$$\begin{pmatrix} x_k \\ x_{k+1} \\ x_{k+2} \\ \vdots \\ x_{k+N-1} \end{pmatrix} = X_k, \begin{pmatrix} I \\ A \\ A^2 \\ \vdots \\ A^{N-1} \end{pmatrix} = A_m, \begin{pmatrix} 0 & 0 & \cdots & 0 & 0 \\ B & 0 & \cdots & 0 & 0 \\ AB & B & \cdots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ A^{N-2}B & A^{N-3}B & \cdots & B & 0 \end{pmatrix} = B_m,$$

$$\begin{pmatrix} u_k \\ u_{k+1} \\ u_{k+2} \\ \vdots \\ u_{k+N-1} \end{pmatrix} = u_k, \begin{pmatrix} y_k \\ y_{k+1} \\ y_{k+2} \\ \vdots \\ y_{k+N-1} \end{pmatrix} = Y_k, \begin{pmatrix} C & 0 & 0 & \cdots & 0 \\ 0 & C & 0 & \cdots & 0 \\ 0 & 0 & C & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & C \end{pmatrix} = C_m,$$

$$\begin{pmatrix} D & 0 & 0 & \cdots & 0 \\ 0 & D & 0 & \cdots & 0 \\ 0 & 0 & D & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & D \end{pmatrix} = D_m$$

3.2. Controller algorithm

The MPC technique must have a cost function in its control algorithm to calculate the optimal solution at every sampling time interval. In this study, the cost function is being minimized by the norm of the difference between the current outputs and the desired trajectory and the norms of motor inputs [10]. However, the quaternion error affects the cost function as indicated in Eqn. 21, where q_{ref} is the reference quaternion, $\bar{q}_{act}$ is the conjugate quaternion of actual value and $\widehat{W}_y$, $\widehat{W}_u$ are the weight matrices [11].

$$J(x,u) = \left[(\Delta Y_k)\widehat{W}_y\right]^2 + \left[(\Delta u_k \widehat{W}_{\Delta u})\right]^2 \tag{21}$$

The quaternion errors are computed using $q_{error} = q_{ref} \otimes \bar{q}_{act}$ with the help of quaternion algebra at every time interval instead of conventional subtraction method [7].

3.3. Quadratic programming

As cost function in this study is in quadratic form, quadratic programming can be applied to solve the optimization problem. The main purpose of quadratic programming is to reduce the cost function $J(x,u)$ by finding out a feasible search direction u [12].

3.4. Input constraints

During the design of the quadrotor, it is important to apply constraint at the force of each motor so that it will behave like a realistic model. So, there will be obviously an upper and lower bounds, u_{ub} and u_{lb} , respectively, at control inputs where $u_{lb} \le u_{k+i} \le u_{ub}$ for $i = 0, 1, 2, \ldots, N-1$. Hence, the linearized model with bounded control inputs can be represented as $u_{lb} - u_T \le u_{k+i} \le u_{ub} - u_T$. In the meantime, $u_{k+i} = u_T + u_{k+i}$ and the matrix form is shown as in Eqn. 22 where $I_{m\times m}$ is an identity matrix.

$$\begin{bmatrix} \mathrm{I}_{m\times m} \\ -\mathrm{I}_{m\times m} \end{bmatrix} \mathrm{u}_{k+i} \le \begin{bmatrix} \mathrm{u}_{ub} - \mathrm{u}_T \\ -(\mathrm{u}_{lb} - \mathrm{u}_T) \end{bmatrix} \tag{22}$$

Eqn. 22 can also be represented in the form of $I_u u_k \le u_b$ where I_u and u_b are given by Eqn. 23 and Eqn. 24, respectively.

$$I_u = \begin{bmatrix} \begin{bmatrix} I_{m\times m} \\ -I_{m\times m} \end{bmatrix} & 0 & \cdots & 0 \\ 0 & \begin{bmatrix} I_{m\times m} \\ -I_{m\times m} \end{bmatrix} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \begin{bmatrix} I_{m\times m} \\ -I_{m\times m} \end{bmatrix} \end{bmatrix} \tag{23}$$

$$u_b = \begin{bmatrix} \begin{bmatrix} u_{ub} - u_T \\ -(u_{lb} - u_T) \end{bmatrix} \\ \begin{bmatrix} u_{ub} - u_T \\ -(u_{lb} - u_T) \end{bmatrix} \\ \vdots \\ \begin{bmatrix} u_{ub} - u_T \\ -(u_{lb} - u_T) \end{bmatrix} \end{bmatrix} \tag{24}$$

4. Simulated results

The algorithm shows that a linear discrete model design is necessarily deemed as the first step toward MPC control algorithm design. Here, the linearized model has been designed at a hovering point, z = 1 m. MATLAB and Simulink provide the platform to develop the model and the control algorithm. However, prediction horizon and control horizon affect the performance (e.g. overshoot, settling time) of the system [13]. The prediction horizon has been confirmed as 20 while the adopted control horizon is 2 in this work. The resultant overshoot and settling time along the axes as tabulated in Table 2.

Table 2. Performance of the controller

Axes	Settling time	Overshoot (%)
x	4.4486	1.6522
y	8.1195	3.9700
z	4.9083	2.2303

Additionally, sampling time plays a significant role in maintaining the stability of the projected model. The sampling time is ensured at 0.25s after the confirmation of the poles as [0, 0, 0, -0.1538, -0.1538, -0.2308, 0, 0, 0, -13.3333, -13.3333, -11.5385]. The rpm of motors are bounded by certain limits and the angular velocity of each motor is considered as 600 rad/s, and this engenders the control effort as $0 < u_1 < 45.0720$, $-11.2680 < u_2 < 11.2680$, $-11.2680 < u_3 < 11.2680$, $-0.54 < u_4 < 0.54$.

Two different trajectories, i.e. circular and helix, have been chosen to check the performance of the designed controller. It is noted that the performance under disturbance and without disturbance are very similar to each other, as evident from Figure 2 and Figure 3. In addition, Table 3 shows the comparison through Root Mean Square Error (RMSE) formulation wherein performance along the helix trajectory is not expected, especially along the y-axis, although the overall performance is optimal.

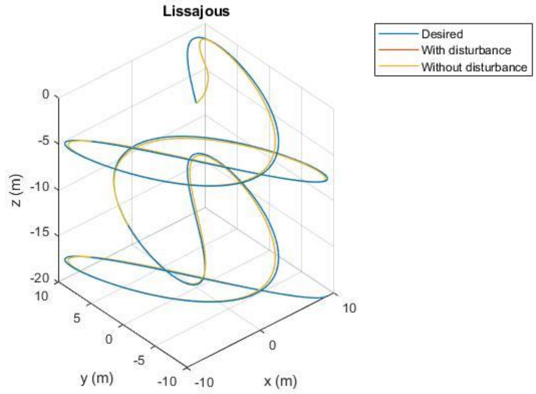

Fig. 2: Lissajous trajectory including disturbance

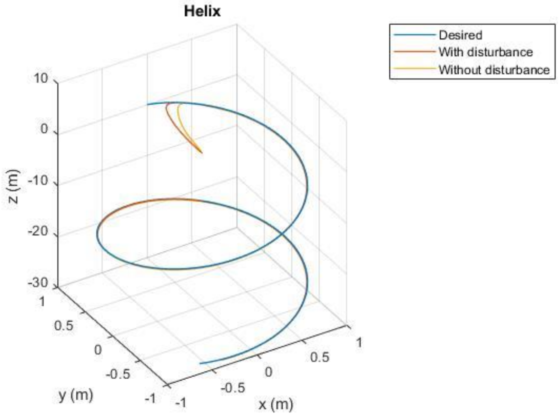

Fig. 3: Helix trajectory including disturbance

Table 3: RMSE for lissajous and helix trajectories with and without disturbances

RMSE for with disturbance			
	x (%)	**y (%)**	**z (%)**
Lissajous	3.9491	6.7345	3.2222
Helix	0.8524	3.9068	2.9437
RMSE for without disturbance			
	x (%)	**y (%)**	**z (%)**
Lissajous	6.4351	3.7419	3.0089
Helix	2.3864	6.5058	2.5975

Figure 4 depicts the control effort while Figure 5 illustrates angular velocity for the quadrotor that is substantially maintained between the upper and lower bounds.

5. Conclusion

This work presents the successful application of MPC in a quaternion-based quadrotor through the two different trajectories tracking. The simulation tests were accomplished using MATLAB and Simulink environment wherein MPC is developed using MPC toolbox in Simulink. In addition, it is successful in the sense that disturbance cannot make much impact on the model and thus the RMSE is appreciable. Furthermore, proposed future work will be gain-scheduling MPC with quaternion that may open a new dimension for nonlinear dynamic model.

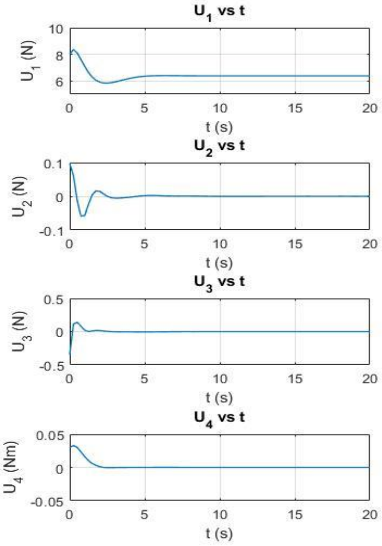

Fig. 4: Control efforts against time

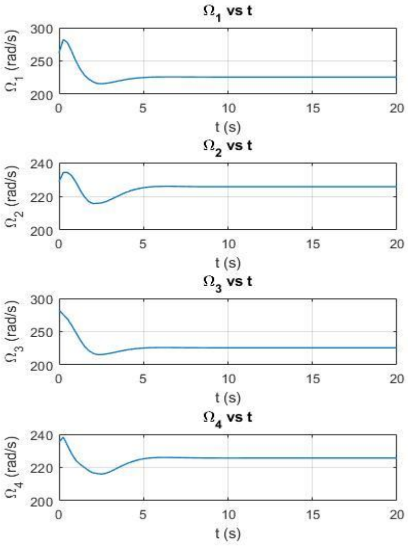

Fig. 5: Angular velocity against time

References

[1] Kendoul F (2012), Survey of advances in guidance, navigation, and control of unmanned rotorcraft systems. *Journal of Field Robotics* 29(2), 315-378

[2] Kozák Š (2012), Advanced control engineering methods in modern technological applications. Carpathian Control Conference (ICCC), 13th International IEEE.

[3] Bouffard P (2012), *On-board Model Predictive Control of a Quadrotor Helicopter: Design, Implementation and Experiments*, University of California: Berkeley.

[4] Raffo GV, Ortega MG, & Rubio FR (2008), MPC with nonlinear $\mathcal{H}\infty$ control for path tracking of a quadrotor helicopter. IFAC Proceedings 41(2), 8564-8569

[5] Alexis K, Nikolakopoulos G & Tzes A (2010), Experimental model predictive attitude tracking control of a quadrotor helicopter subject to wind-gusts. 18th IEEE Mediterranean Conference on Control & Automation

[6] Fresk E & Nikolakopoulos G (2013), Full quaternion based attitude control for a quadrotor. IEEE European Control Conference (ECC)

[7] Reyes-Valeria E, Enriquez-Caldera R, Camacho-Lara S & Guichard J (2013), LQR control for a quadrotor using unit quaternions: Modeling and simulation. *IEEE International Conference on Electronics, Communications and Computing (CONIELECOMP)*

[8] Bouabdallah S (2007), *Design and control of quadrotors with application to autonomous flying*. PhD Thesis, Ecole Polytechnique.

[9] Lindblom S & Lundmark A (2015), *Modelling and control of a hexarotor UAV*. Thesis, Linkoping University

[10] Bemporad A, Morari M & Lawrence RN (2017), *Model Predictive Control Toolbox™ User's Guide: Optimization Problem*. The MathWorks, Inc.

[11] Mathworks (2016), Optimization Problem

[12] Bemporad A, Morari M & Lawrence RN (2017), *Model Predictive Control Toolbox™ User's Guide in QP Solver*. The MathWorks, Inc.

[13] Islam M, Okasha M & Idres M (2017), Dynamics and control of quadcopter using linear model predictive control approach. *IOP Conference Series: Materials Science and Engineering* 270, 012007