Constrained Control of Quadrotor Using Laguerre Functions Based Model Predictive Control for Reference Tracking

Prayag Sharma
Dept of Electronics and Communication Engineering,
University Institute of Engineering and Technology,
Panjab University, Chandigarh, India

Utkarsh Bajpai
I-Hub Foundation for Cobotics, Indian Institute of
Technology, Delhi, India

Adnan Jawed
Dept of Mechanical Engineering, Indian Institute of
Technology, Delhi, India

S.K Saha
Dept of Mechanical Engineering, Indian Institute of
Technology, Delhi, India

ABSTRACT

This paper addresses the reference tracking problem of a quadrotor in presence of physical constraints using Laguerre functions-based Model Predictive Control (MPC). MPC necessitates in finding an optimal solution, even under system constraints. The use of Laguerre functions helps in reducing computation time, for the online implementation of the controller. The dynamic equations of the quadrotor are derived using the Newton-Euler method. A quadratic cost function is derived for the Laguerre-based MPC and Hildreth's quadratic programming method is applied for the optimization of the quadratic cost in presence of constraints. The quadrotor model with constraints is simulated in MATLAB and results are presented for circular and helical 3D reference trajectories.

CCS CONCEPTS

• **Computer systems organization**; • **Embedded and cyber-physical systems**; • **Robotics**; • **Robotic control**;

KEYWORDS

UAV, Model Predictive Control, Laguerre Functions, Reference Tracking

ACM Reference Format:
Prayag Sharma, Utkarsh Bajpai, Adnan Jawed, and S.K Saha. 2021. Constrained Control of Quadrotor Using Laguerre Functions Based Model Predictive Control for Reference Tracking. In *Advances in Robotics - 5th International Conference of The Robotics Society (AIR2021), June 30–July 04, 2021, Kanpur, India.* ACM, New York, NY, USA, 7 pages. https://doi.org/10.1145/3478586.3478604

1 INTRODUCTION

Quadrotors are Unmanned Aerial Vehicle (UAV) that use four rotors for generating thrust and maneuvering in 3D space. Quadrotors have gained a lot of popularity among researchers and industries, due to their ease of design and robustness. They are utilized in various applications such as surveying, search and rescue, mapping, inspection, mining, construction, agriculture, public safety, and scientific research. Operation in such demanding applications requires successful tracking of complex trajectories while operating under the physical constraints of the system. Moreover, the quadrotor is an underactuated system making it an important control problem for efficient trajectory tracking under constraints.

Various control designs have been proposed for the efficient design of a quadrotor's control, such as PID and LQ control in [1], [2]. The PID-based control techniques require extensive gain tunning and is inherently a single-input single-output (SISO) control technique, but the quadrotor is a multi-input multi-output (MIMO) system. In [3], backstepping control was presented to stabilize and track reference trajectories for the quadrotor and in [4] Feedback Linearization and Adaptive Sliding Mode Control was analyzed for quadrotor control. However, physical constraints were not considered in either of the controller design. In [5], a MPC and a H∞ controller was deployed for trajectory tracking and to stabilize the attitude respectively. However, as mentioned above, physical constraints were not considered. In [6] a hierarchical control methodology that breaks into multi-level control strategy, and uses MPC for reference tracking, but uses unconstrained inputs and states to have a real-time solution. In [7], the author proposed a hierarchical control scheme in which an integral MPC was used for the translational motions tracking and a MPC based scheme was used for attitude tracking. It takes into consideration the physical constraints of the rotorcraft i.e., constraints on the inputs. However, the control design uses two different controllers, one for translational motions and one for attitude tracking which makes the control design complex. Other control techniques like, [8], [9], [10] deploy adaptive neural network-based control schemes for the control of a quadrotor.

The advantage of MPC over other control techniques is the ability to solve for an optimal solution for reference tracking while taking constraints into the consideration. However, MPC is computationally expensive and requires high processing power for real-time implementation. In [11], wang showed that the number of terms in optimization can be reduced by using Laguerre functions-based Model Predictive Control as compared to the traditional MPC approach, thus reducing the computation time. In our work, a centralized, Linear Model Predictive Control based on Laguerre functions is deployed to control both the position and attitude of the quadrotor. Constraints are considered on both control inputs

AIR2021, June 30–July 04, 2021, Kanpur, India

ACM ISBN 978-1-4503-8971-6/21/06…$15.00
https://doi.org/10.1145/3478586.3478604

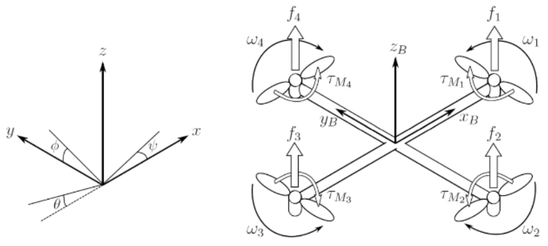

Figure 1: Inertial and body frames of a quadrotor [14]

and rate of control input change simultaneously. Moreover, to solve for the optimal solution in presence of constraints as given in [12] “Hildreth’s Quadratic programming procedure”, [13] was used for the quadratic programming problem. The proposed design was simulated in MATLAB to follow circular and helical reference trajectories. RMSE (Root Mean Square Error) and MAE (Mean Absolute Error). with respect to reference trajectory and actual trajectory were calculated and the results are provided in this paper for each.

2 DYNAMICAL MODEL OF A QUADROTOR

A body frame B and an inertial frame I are introduced to define the attitude. Let the position and attitude vector of the quadrotor with respect to the inertial frame(I) of the quadrotor be ξ and ζ, respectively, where ξ and ζ are defined as:

$\xi \equiv [\ x \quad y \quad z\ ]^T$ and $\zeta \equiv [\ \phi \quad \theta \quad \psi\ ]^T$,

and ϕ, θ, ψ denote the roll, pitch, and yaw angles respectively. The body frame’s origin lies at the center of mass of the quadrotor. Defining linear and angular velocities with respect to body frame (B) of the quadrotor as

$V_B \equiv [\ v_x \quad v_y \quad v_z\ ]^T$ and $\Omega \equiv [\ p \quad q \quad r\ ]^T$.

the following equations describe the relationship between ξ, ζ, V_B and Ω ,

$$\dot{\xi} = \mathrm{RV_B}, \dot{\zeta} = \mathrm{J}\Omega \tag{1}$$

where

$$\mathrm{R} = \begin{bmatrix} C\theta C\psi & C\psi S\theta S\phi - S\psi C\phi & C\psi S\theta C\phi + S\psi S\phi \\ C\theta S\psi & S\psi S\theta S\phi + C\psi C\phi & S\psi S\theta C\phi - C\psi S\phi \\ -S\theta & C\theta S\psi & C\theta C\phi \end{bmatrix} \tag{2}$$

and

$$\mathrm{J} = \begin{bmatrix} 1 & 0 & -S\theta \\ 0 & C\phi & S\phi C\theta \\ 0 & -S\phi & C\phi C\theta \end{bmatrix} \tag{3}$$

The rotational matrix from the body frame of reference to the inertial reference frame is given by **R**. Similarly, **J** is the angular velocity transformation matrix from the body frame to the inertial frame. In (2) and (3), $Sx \equiv Sin(x)$, $Cx \equiv Cos(x)$. From small angle assumption around the hovering point, $Cos(x) \approx 1$ and $Sin(x) \approx 0$. Thus, $\dot{\phi} \approx p$, $\dot{\theta} \approx q$, $\dot{\psi} \approx r$.

From aerodynamics [15], the lift force and moment are proportional to the square of the rotation speed of the propeller. They are represented below:

$$F_i = k_f \omega_i^2 \text{and } \tau_i = k_m \omega_i^2,\ i = 1, 2, 3, 4. \tag{4}$$

where, k_f and k_m are the aerodynamical force and moment constants and ω_i is the rotational speed of the propeller. Thus, the total torque and thrust generated by the quadrotor are given by the following equations:

$$\begin{cases} T = k_f \left[\omega_1^2 + \omega_2^2 + \omega_3^2 + \omega_4^2\right] \\ \tau_\phi = lk_f \left[(\omega_2^2 + \omega_3^2\right) - (\omega_1^2 + \omega_4^2)] \\ \tau_\theta = lk_f \left[(\omega_1^2 + \omega_2^2\right) - (\omega_3^2 + \omega_4^2)] \\ \tau_\psi = k_m \left[(\omega_1^2 + \omega_3^2\right) - (\omega_2^2 + \omega_4^2)] \end{cases} \tag{5}$$

In (5), l denotes the length of the quadrotor arm. From the Newton-Euler equations of motion and considering small angle approximation, the rotational motion can be described using the following equations given in (6)-(9),

$$I\ddot{\zeta} + \dot{\zeta} \times I\dot{\zeta} = \begin{bmatrix} \tau_\phi \\ \tau_\theta \\ \tau_\psi \end{bmatrix} \tag{6}$$

$$I = \begin{bmatrix} I_{xx} & 0 & 0 \\ 0 & I_{yy} & 0 \\ 0 & 0 & I_{zz} \end{bmatrix} \tag{7}$$

Due to the symmetric structure of a quadrotor, the Inertia matrix I is diagonal. From the substitution of (7) in (6), we get,

$$\begin{bmatrix} \dot{p} \\ \dot{q} \\ \dot{r} \end{bmatrix} = \begin{bmatrix} (I_{yy} - I_{zz})qr/I_{xx} \\ (I_{zz} - I_{xx})pr/I_{yy} \\ (I_{xx} - I_{yy})pq/I_{zz} \end{bmatrix} + \begin{bmatrix} \frac{\tau_\phi}{I_{xx}} \\ \frac{\tau_\theta}{I_{yy}} \\ \frac{\tau_\psi}{I_{zz}} \end{bmatrix} \tag{8}$$

8) can be linearized around the hovering point i.e. $\dot{\phi} \approx 0$, $\dot{\theta} \approx 0$, $\dot{\psi} \approx 0$ to yield,

$$\begin{bmatrix} \dot{p} \\ \dot{q} \\ \dot{r} \end{bmatrix} = \begin{bmatrix} \frac{\tau_\phi}{I_{xx}} \\ \frac{\tau_\theta}{I_{yy}} \\ \frac{\tau_\psi}{I_{zz}} \end{bmatrix} \tag{9}$$

The translational equations of motion can be derived by applying Newton’s Second law as:

$$m\ddot{\xi} = \begin{bmatrix} 0 \\ 0 \\ mg \end{bmatrix} + \boldsymbol{R} \begin{bmatrix} 0 \\ 0 \\ -T \end{bmatrix} \tag{10}$$

where **R** is the rotation matrix given in (2). Similarly, (10) can be linearized around hovering point to get,

$$\begin{bmatrix} \ddot{x} \\ \ddot{y} \\ \ddot{z} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ g \end{bmatrix} + \begin{bmatrix} -g\theta \\ g\phi \\ -\frac{T}{m} \end{bmatrix} \tag{11}$$

For the linearized state-space model, the internal state vector, control vector, and output vector are given by,

$$\boldsymbol{X^T} \equiv \begin{bmatrix} x & y & z & \dot{x} & \dot{y} & \dot{z} & \phi & \theta & \psi & \dot{\phi} & \dot{\theta} & \dot{\psi} \end{bmatrix}^{\boldsymbol{T}} \tag{12}$$

$$\boldsymbol{U^T} \equiv [\ u_1 \quad u_2 \quad u_3 \quad u_4\ ]^T \tag{13}$$

In (13), $[u_1]$ = [T-mg] and $[\ u_2 \quad u_3 \quad u_4\ ]^T$ = $[\ \tau_\phi \quad \tau_\theta \quad \tau_\psi\ ]^T$, and the output vector is as follows,

$$\boldsymbol{Y^T} \equiv [\ x \quad y \quad z \quad \psi\ ]^T \tag{14}$$

Linearized State-space form is then expressed as

$$\dot{\boldsymbol{X}} = \boldsymbol{AX} + \boldsymbol{BU} \tag{15}$$

$$\boldsymbol{Y} = \boldsymbol{CX} \tag{16}$$

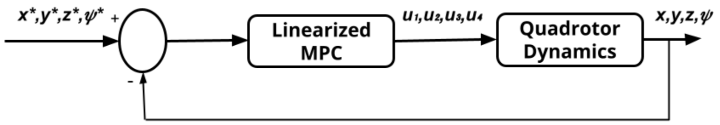

Figure 2: Control diagram for Linearized MPC (x^*, y^*, z^*, ψ^*: reference signals)

Using 1), (2), (9), (11) to (16), the values of ***A, B & C*** can be evaluated.

3 MODEL PREDICTIVE CONTROL USING LAGUERRE FUNCTIONS

An MPC can handle hard constraints in control design. An optimal solution is calculated at each sampling instant in the presence of constraints. Using Laguerre functions to parameterize control signal [12], one can provide two explicit tuning parameters, namely, Laguerre pole (a) and no. of terms (N), for achieving the desired closed loop performance and providing the flexibility to choose future constraints locations. Moreover, the number of terms for optimization are reduced. Furthermore, the smoothness of Laguerre polynomials avoids steep changes in the control signal.

3.1 Derivation of Laguerre based MPC

Assuming the plant has m inputs, q outputs, and n states, the discretized state-space form of the plant to be controlled can be represented as

$$x_m(k+1) = A_m x_m(k) + B_m u(k)$$

$$y(k) = C_m x_m(k) \tag{17}$$

Where $x_m(k)$ denotes the state vector, $u(k)$ denotes the control input vector, and $y(k)$ is the output vector. Defining $\Delta x_m(k+1) = x_m(k+1) - x_m(k)$ and $\Delta u(k) = u(k) - u(k-1)$, the augmented state-space form of the plant (17) can be given as

$$\begin{bmatrix} \Delta x_m(k+1) \\ y(k+1) \end{bmatrix} = \begin{bmatrix} A_m & \mathrm{O}_m^T \\ C_m A_m & I_{q\times q} \end{bmatrix} \begin{bmatrix} \Delta x_m(k) \\ y(k) \end{bmatrix} + \begin{bmatrix} B_m \\ C_m B_m \end{bmatrix} \Delta u(k)$$

$$y(k) = \begin{bmatrix} \mathrm{O}_m & I_{q\times q} \end{bmatrix} \begin{bmatrix} \Delta x_m(k) \\ y(k) \end{bmatrix} \tag{18}$$

In which $I_{q\times q}$ is the identity matrix having a dimension of $q \times q$, O_m is a $q \times n$ zero matrix. By selecting a new state vector $x(k) = [\ \Delta x_m(k)^T \quad y(k)^T\ ]^T$ and defining $A = [\begin{smallmatrix} A_m & O_m^T \\ C_m A_m & I_{q\times q} \end{smallmatrix}]$, $B = [\begin{smallmatrix} B_m \\ C_m B_m \end{smallmatrix}]$ and $C = [\ O_m \quad I_{q\times q}\ ]$, (18) can be written as

$$x(k+1) = Ax(k) + B\Delta u(k)$$

$$y(k) = Cx(k) \tag{19}$$

Let at sampling instant k, $\Delta U \equiv [\Delta u(k), \Delta u(k+1), \ldots, \Delta u(k+N_c-1)]^T$ denote the control trajectory. Laguerre functions are a set of orthonormal functions and they can be used to approximate the incremental terms in ΔU. The z-transform of the discrete-time Laguerre functions is then given by

$$\Gamma_k(z) = \Gamma_{k-1}(z) \left(\frac{z^{-1} - a}{1 - az^{-1}} \right) \tag{20}$$

$$\Gamma_k(z) = \frac{\sqrt{1-a^2}}{1-az^{-1}} \tag{21}$$

Let $l_i(k)$ denote the inverse z-transfrom of $\Gamma_i(z, a)$, then the vector form for such discrete-time Laguerre functions is given by

$$L(k) = [l_1(k) \ \ l_2(k) \ \ l_3(k) \ldots l_N(k)]^T \tag{22}$$

Using (20), a difference equation can be derived that is satisfied by Laguerre functions given by,

$$L(k+1) = A_l L(k) \tag{23}$$

where A_l is $(N \times N)$ and is defined using parameters a and $\beta = (1 - a^2)$, where a denotes the Laguerre pole and $0 \leq a < 1$ for the Laguerre network to be stable. The initial condition is given by

$$L(0)^T = \sqrt{\beta} \begin{bmatrix} 1 & -a & a^2 & -a^3 & \ldots & (-1)^{N-1}(a)^{N-1} \end{bmatrix}^T \tag{24}$$

For $N = 5$,

$$A_l = \begin{bmatrix} a & 0 & 0 & 0 & 0 \\ \beta & a & 0 & 0 & 0 \\ -a\beta & \beta & a & 0 & 0 \\ a^2\beta & -a\beta & \beta & a & 0 \\ -a^3\beta & a^2\beta & -a\beta & \beta & a \end{bmatrix} ; L(0) = \sqrt{\beta} \begin{bmatrix} 1 \\ -a \\ a^2 \\ -a^3 \\ a^4 \end{bmatrix}$$

At instant k, the control trajectory, i.e., $\Delta u(k), \Delta u(k+1), \ldots, \Delta u(k+N_c-1)$ is referred as a stable dynamic system's impulse response. The set of Laguerre functions $l_i(k)$ are used to define the control trajectory as (at instant k_i):

$$\Delta u(k_i + k) = L(k)^T \eta \tag{25}$$

In (25), the parameter $\eta = [c_1 \ c_2 \ \ldots \ c_N]$ and c_j, $j = 1, 2, \ldots, N$ are the coefficients and are functions of the initial time of k_i, whereas, N denotes the no. of expansion terms. For a Multi-Input, Multi-Output (MIMO) system,

$$\Delta u(k_i + m) = \begin{bmatrix} L_1(m)^T & 0_2^T & \ldots & 0_m^T \\ 0_1^T & L_2(m)^T & \ldots & 0_m^T \\ \vdots & \vdots & \ddots & \vdots \\ 0_1^T & 0_2^T & \ldots & L_m(m)^T \end{bmatrix} \eta, \tag{26}$$

where $m = 0, 1, 2, 3\ldots$, and the input matrix is denoted by $B = [\ B_1 \quad B_2 \quad \cdots \quad B_m\ ]$. The term $\Delta u(k_i + m)$ denotes the change of control at sampling instant m, and $L_i(m)^T$ denotes the discrete-time Laguerre function vector for the ith control, and 0_m is a zero vector. The objective is to find the vector η to minimize the cost function:

$$J = \sum_{m=1}^{N_p} x(k_i + m \mid k_i)^T Q x(k_i + m \mid k_i) + \eta^T R_L \eta \tag{27}$$

Note, the weighting matrices are $Q \geq 0$ and $R_L > 0$, where R_L has the same dimension as η, Q has dimension equal to state variables.

Also $Q = C^T C$ for this formulation and $R_L = r_u I_{N\times N}$, whereas r_u is a scaler and $I_{N\times N}$ is the identity matrix of dimension N, and N_p is the prediction horizon. The term $x(k_i + m \mid k_i)$ is given next as

$$\begin{aligned} x(k_i + m \mid k_i) &= A^m x(k_i) + \sum_{j=0}^{m-1} A^{m-j-1} B \Delta u(k_i + j) \\ &= A^m x(k_i) + \sum_{j=0}^{m-1} A^{m-j-1} B L(j)^T \eta \\ = A^m x(k_i) + &\sum_{j=0}^{m-1} A^{m-j-1} \begin{bmatrix} B_1 L_1(j)^T & B_2 L_2(j)^T & \cdots & B_m \;\; L_m(j)^T \end{bmatrix} \eta \\ &= A^m x(k_i) + \phi(m)^T \eta \end{aligned} \tag{28}$$

where $\eta^T = [\; \eta_1^T \quad \eta_2^T \quad \dots \quad \eta_m^T \;]$ and $\phi(m)^T$ is given by

$$\phi(m)^T = \sum_{j=0}^{m-1} A^{m-j-1} \begin{bmatrix} B_1 L_1(j)^T & B_2 L_2(j)^T & \cdots & B_m \;\; L_m(j)^T \end{bmatrix} \tag{29}$$

From (27) and (28), the cost function becomes:

$$J = \eta^T \Omega \eta + 2\eta^T \Psi x(k_i) + \sum_{m=1}^{N_p} x(k_i)^T \left(A^T\right)^m Q A^m x(k_i) \tag{30}$$

In which Ω and Ψ are:

$$\Omega = \sum_{m=1}^{N_p} \phi(m) Q \phi(m)^T + R_L \tag{31}$$

$$\psi = \sum_{m=1}^{N_p} \phi(m) Q A^m \tag{32}$$

3.2 Constrained Control using Laguerre based MPC

The optimization of Laguerre-based MPC subject to constraints is a quadratic programming problem. Although MPC can handle constraints of many kinds, constraints on control input and rate of change in control input are considered in this formulation. For a standard Quadratic programming problem, if x is the decision variable, then J is given by,

$$J = \frac{1}{2} x^T E x + x^T F \tag{33}$$

$$Mx \leq \gamma \tag{34}$$

In which E, F, M and γ are vectors and matrices in quadratic programming. The objective function (33) needs to be optimized subject to the constraints in (34). E is assumed to be a symmetric and positive definite matrix. A simple algorithm was introduced, which is called Hildreth's quadratic programming procedure [13], this does not involve any matrix inversion as it is an element-by-element search technique. Hildreth's quadratic programming method is as follows:

$$\lambda_i^{m+1} = \max\left(0, w_i^{m+1}\right), \tag{35}$$

where

$$w_i^{m+1} = -\frac{1}{h_{ii}} \left[k_i + \sum_{j=1}^{i-1} h_{ij} \lambda_j^{m+1} + \sum_{j=i+1}^{n} h_{ij} \lambda_j^m \right], \tag{36}$$

In (36), the scaler h_{ij} is the $ijth$ matrix element of $H = ME^{-1}M^T$, k_i is the ith element in the vector $K = \gamma + ME^{-1}F$, m is the iteration number and λ is the vector of Lagrange multipliers. The converged solution of Lagrange multipliers after a complete iteration is given by λ^*. It is a vector that only has values which are zero or positive. The optimized solution of x in the presence of constraints is given by

$$x = -E^{-1}\left(F + M^T \lambda^*\right) \tag{37}$$

For this formulation, the quadratic cost function to be minimized becomes,

$$J = \eta^T \Omega \eta + 2\eta^T \Psi x(k_i), \tag{38}$$

Note in (38), η is the decision variable, and Ω and Ψ are given by (31) and (32) respectively. The following inequality ensure the constraints on the rate of change of control input:

$$\Delta u^{min} \leq \Delta u(k_i + m) \leq \Delta u^{max}$$

where Δu^{min} and Δu^{max} are the lower and upper limits of the rate of change in control input. From (26), one can re-write (38) as (39), while constraints on control input are given by (40):

$$\Delta u^{min} \leq \begin{bmatrix} L_1(m)^T & 0_2^T & \dots & 0_m^T \\ 0_1^T & L_2(m)^T & \dots & 0_m^T \\ \vdots & \vdots & \ddots & \vdots \\ 0_1^T & 0_2^T & \dots & L_m(m)^T \end{bmatrix} \eta \leq \Delta u^{max} \tag{39}$$

$$u^{min} \leq \begin{bmatrix} \sum\limits_{i=0}^{k-1} L_1(i)^T & 0_2^T & \dots & 0_m^T \\ 0_1^T & \sum\limits_{i=0}^{k-1} L_2(i)^T & \dots & 0_m^T \\ \vdots & \vdots & \ddots & \vdots \\ 0_1^T & 0_2^T & \dots & \sum\limits_{i=0}^{k-1} L_m(i)^T \end{bmatrix} \eta + u(k_i - 1) \leq u^{max}, \tag{40}$$

In (40), $u(k_i - 1)$ is the previous control input and u^{max} and u^{min} are the upper and lower limits of control input respectively. From comparing (33), (34) to (38), (39), and (40), E, F, M, and γ can be evaluated.

4 SIMULATION AND RESULTS

To verify the ability and effectiveness of the proposed controller, simulation of the quadcopter was performed in which it was made to follow two different 3D reference trajectories in the presence of constraints on control input and the change in rate of control input. For both the trajectories, the performance metrics chosen to establish the accuracy of the controller were RMSE and MAE. Parameters of the quadrotor are taken as in [15], and are presented in Table 1

4.1 Constraints Evaluation and Controller Parameters

For the given quadrotor model, the maximum angular speed of the motor ω_{max} was taken as 4720 rpm [15]. For constraints to be evaluated the rpm of the motors at the hovering point needs to be

Table 1: Parameters for simulation [15]

Parameter	Value
Mass, m	$1.587kg$
Gravity, g	$9.81\frac{m}{s^2}$
Length of arm,l	$0.243m$
Thrust constant, k_f	4.0687×10^{-7}N/rpm^2
Drag constant, k_m	8.4367×10^{-9}Nm/rpm^2
MOI about x-axis, I_{xx}	$0.0213kgm^2$
MOI about y-axis, I_{yy}	$0.02217kgm^2$
MOI about z-axis, I_{zz}	$0.0282kgm^2$

determined first along with the maximum thrust produced by the quadrotor. At hovering position,

$$weight = Total\ thrust \tag{41}$$

$$weight = mg = 15.568N \tag{42}$$

$$Total\ thrust = k_f \sum_{i=1}^{4} \omega_i^2 \tag{43}$$

From (41), (42), and (43) speed of each rotor at the hovering point comes out to be approximately 3093 rpm. The maximum thrust produced by the quadrotor T_{max} is given by,

$$T_{max} = 4k_f\omega_{max}^2 \tag{44}$$

The value of T_{max} was calculated as $36.257N$. From values of angular speed mentioned above, parameters of Table 1 and the equations given in (5) the maximum and minimum values of control input torques are obtained as:

$$\begin{cases} -1.257 \le \tau_\phi \le 1.257 \\ -1.257 \le \tau_\theta \le 1.257 \\ -0.2145 \le \tau_\psi \le 0.2145 \end{cases} \tag{45}$$

As $[\ u_2 \quad u_3 \quad u_4\ ]^T = [\ \tau_\phi \quad \tau_\theta \quad \tau_\psi\ ]^T$ and $[u_1] = [T\text{-}mg]$, thus constraints on u_1 are given by,

$$-mg \le u_1 \le T_{max} - mg$$

$$-15.568 \le u_1 \le 20.689 \tag{46}$$

For the rate of change in control input the value is set to 60% of the input value, and They are given as:

$$\begin{cases} -9.3408 \le \Delta u_1 \le 12.413 \\ -0.7542 \le \Delta u_2 \le 0.7542 \\ -0.7542 \le \Delta u_3 \le 0.7542 \\ -0.7542 \le \Delta u_4 \le 0.7542 \end{cases} \tag{47}$$

For the control design $Q = C^TC$ and $R_L = r_uI_{N\times N}$, $r_u = 0.1$. The other parameters of the controller were tuned via trial and error performed over various simulations. They are given in Table 2

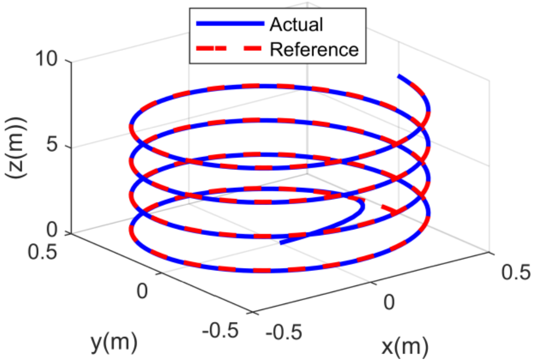

Figure 3: Tracking of the reference signal (red and dashed) by quadrotor (solid blue) in 3D space.

4.2 Trajectory Tracking and Results

Two trajectories are considered for testing, helical and circular. Following are the reference signals generated for each of the trajectories:

$$\begin{aligned} &\text{Circular} : x^* = 0.5\cos(0.0630t), \\ &y^* = 0.5\sin(0.0630t),\ z^* = 1,\ \psi^* = 0 \end{aligned} \tag{48}$$

$$\begin{aligned} &\text{Helical} :\ x^* = 0.5\cos(0.0630t), \\ &y^* = 0.5\sin(0.0630t),\ z^* = 0.0201t,\ \psi^* = 0 \end{aligned} \tag{49}$$

The starting position of the quadrotor in (x, y, z) is set to be $(0, 0, 0)$ for both the trajectories. The helical trajectory starts from (x^*, y^*, z^*)=$(0.5, 0, 0)$. For the helical trajectory, figures (3) to (5) show the simulation of reference tracking for the quadrotor in presence of constraints. It can be seen from Figures (4) and (5) that the control signals and their rate of change do not violate any constraints and Figure (3) shows reference tracking for helical trajectory in 3D space. As it can be seen in Table 3 the maximum MAE received is in the z-axis equal to 1.159×10^{-1} m in case of helical trajectory. For the entire trajectory, the percentage of error comes out to be only 1.4% in the z-axis. It was observed that the attitude values of the UAV adhered to the small angle approximation considered during the control design. Thus, the controller performs reference tracking with a maximum error of less than 1.5% in each axis, which is satisfactory and tolerable.

5 CONCLUSION

A centralized MPC based on Laguerre functions was presented in this paper. The use of Laguerre functions reduces constraints in the prediction horizon hence, making it suitable for online implementation. Laguerre-based MPC can handle systems where rapid

Table 2: Controller parameters for Laguerre based MPC

Parameter	Sampling time (Δt)	Laguerre pole(a)	No. of terms (N)	Prediction Horizon (N_p)	Control Horizon (N_c)
Value	$0.1s$	0.5	10	100	10

Table 3: MAE and RMSE for circular and helical trajectories.

Performance Metric	Trajectory		Mean axis	
		Error in $x(m)$	Error in $y(m)$	Error in $z(m)$
MAE	Circle	3.4×10^{-3}	6.6×10^{-3}	5.9×10^{-4}
	Helix	3.4×10^{-3}	6.6×10^{-3}	1.159×10^{-1}
RMSE	Circle	1.456×10^{-1}	1.418×10^{-1}	1×10^{-2}
	Helix	1.456×10^{-1}	1.418×10^{-1}	1.6×10^{-1}

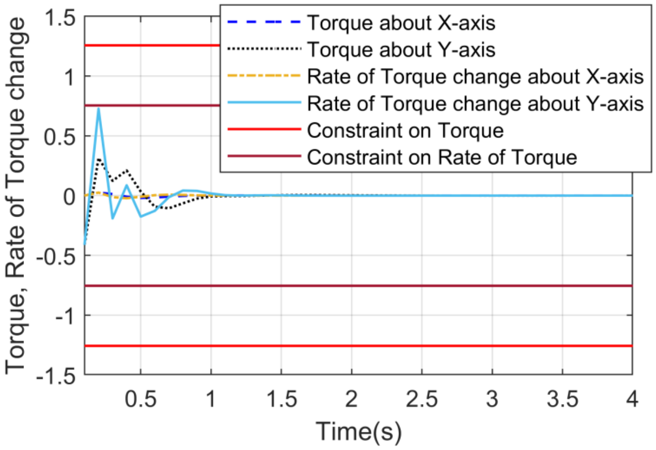

Figure 4: Torque and their respective rate of change for Helical reference trajectory.

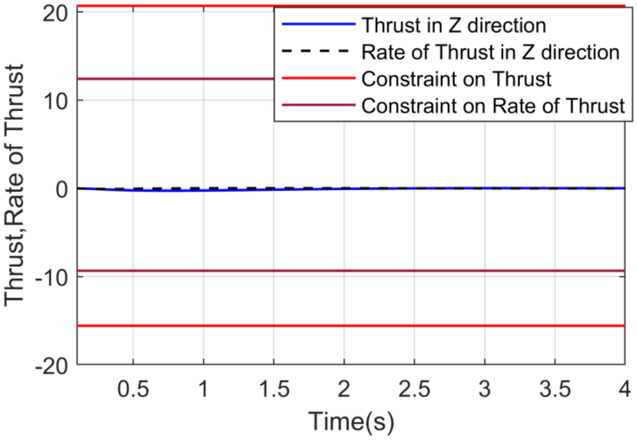

Figure 5: Thrust and rate of change of thrust for Helical reference trajectory.

sampling is involved and more complicated process dynamics are required [16]. A quadratic cost function was derived for the Laguerre-based MPC, and Hildreth's quadratic programming technique was applied for the optimization of the quadratic cost in presence of constraints. Constraints were calculated and the quadrotor model was simulated for tracking circular and 3D helical reference trajectories. The controller was able to follow the reference trajectories with a maximum error of less than 1.5% in each axis, which shows satisfactory performance while operating under the physical constraints of the system.

ACKNOWLEDGMENTS

This research was partially supported by the Indo-Korean Joint Network Center on Robotics under the reference No. INT/Korea/Robotics dated 23-03-18

REFERENCES

[1] Bouabdallah, Samir & Noth, A. & Siegwart, Roland. (2004). PID vs LQ Control Techniques Applied to an Indoor Micro Quadrotor. Proceedings of 2004 IEEE/RSJ International Conference on Intelligent Robots and Systems. 3. 2451 - 2456 vol.3. 10.1109/IROS.2004.1389776.

[2] K. Alexhs, G. Nikolakopoulos, and A. Tzes, “Autonomous quadrotor position and attitude PID/PIDD control in GPS-denied environments,” International Review of Automatic Control, vol. 4, no. 3, 2011.

[3] Madani, Tarek & Benallegue, Abdelaziz. (2006). Backstepping Control for a Quadrotor Helicopter. IEEE International Conference on Intelligent Robots and Systems. 3255 - 3260. 10.1109/IROS.2006.282433.

[4] Lee, Daewon & Sastry, Shankar. (2009). Feedback linearization vs. adaptive sliding mode control for a quadrotor helicopter. International Journal of Control, Automation and Systems, 7(3), 419-428. International Journal of Control, Automation and Systems. 7. 419-428. 10.1007/s12555-009-0311-8.

[5] Raffo, G.V. & Ortega, Manuel & Rubio, Francisco. (2014). MPC with Nonlinear $H\infty$ Control for Path Tracking of a Quad-Rotor Helicopter. Asian Journal of Control. 17. 10.1002/asjc.823.

[6] Bangura, M. and Mahony, R., 2014. Real-time model predictive control for quadrotors. IFAC Proceedings Volumes, 47(3), pp.11773-11780.

[7] K. Alexis, G. Nikolakopoulos and A. Tzes, "Model predictive control scheme for the autonomous flight of an unmanned quadrotor," 2011 IEEE International Symposium on Industrial Electronics, Gdansk, Poland, 2011, pp. 2243-2248, doi: 10.1109/ISIE.2011.5984510.

[8] A. Mishra and S. Ghosh, "An Intelligent Control of Quadcopter for Efficient Path Following," 2019 Sixth Indian Control Conference (ICC), 2019, pp. 134-139, doi: 10.1109/ICC47138.2019.9123162.

[9] Y. Teng, B. Hu, Z. Liu, J. Huang and Z. Guan, "Adaptive neural network control for quadrotor unmanned aerial vehicles," 2017 11th Asian Control Conference (ASCC), 2017, pp. 988-992, doi: 10.1109/ASCC.2017.8287305.

[10] A. K. Shastry, M. T. Bhargavapuri, M. Kothari and S. R. Sahoo, "Quaternion based adaptive control for package delivery using variable-pitch quadrotors," 2018 Indian Control Conference (ICC), 2018, pp. 340-345, doi: 10.1109/INDIANCC.2018.8308002.

[11] Liuping Wang, "Discrete time model predictive control design using Laguerre functions," Proceedings of the 2001 American Control Conference. (Cat. No.01CH37148), Arlington, VA, USA, 2001, pp. 2430-2435 vol.3, doi: 10.1109/ACC.2001.946117.

[12] Wang, Liuping. (2009). Model Predictive Control System Design and Implementation using MATLAB.

[13] Hildreth, C. (1957), A quadratic programming procedure. Naval Research Logistics, 4: 79-85.

[14] Luukkonen, T., 2011. Modelling and control of quadcopter. Independent research project in applied mathematics, Espoo, 22, p.22.

[15] Amadi, Chinedu & Smit, Willie. (2018). Design and Implementation of Model Predictive Control on Pixhawk Flight Controller.

[16] J. A. Rossiter and L. Wang, "Exploiting Laguerre functions to improve the feasibility/performance compromise in MPC," 2008 47th IEEE Conference on Decision and Control, Cancun, Mexico, 2008, pp. 4737-4742, doi: 10.1109/CDC.2008.4738969.