IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 3 August 2022, accepted 23 August 2022, date of publication 26 August 2022, date of current version 6 September 2022.

Digital Object Identifier 10.1109/ACCESS.2022.3202020

RESEARCH ARTICLE

Design and Implementation of a Constrained Model Predictive Control Approach for Unmanned Aerial Vehicles

MORTEZA ALIYARI[1], WING-KWONG WONG[1], YASSINE BOUTERAA[2,3], (Member, IEEE), SEPIDEH NAJAFINIA[1], AFEF FEKIH[4], (Senior Member, IEEE), AND SALEH MOBAYEN[5], (Senior Member, IEEE)

[1]Department of Electronic Engineering, National Yunlin University of Science and Technology, Douliou, Yunlin 64002, Taiwan
[2]College of Computer Engineering and Sciences, Prince Sattam Bin Abdulaziz University, Al-Kharj 11942, Saudi Arabia
[3]Control and Energy Management Laboratory (CEM Laboratory), Ecole Nationale d'Ingenieurs de Sfax (ENIS) & Institut Superieur de Biotechnologie de Sfax (ISBS), University of Sfax, Sfax 3038, Tunisia
[4]Department of Electrical and Computer Engineering, University of Louisiana at Lafayette, Lafayette, LA 70503, USA
[5]Future Technology Research Center, National Yunlin University of Science and Technology, Douliou, Yunlin 64002, Taiwan

Corresponding authors: Saleh Mobayen (mobayens@yuntech.edu.tw) and Yassine Bouteraa (yassine.bouteraa@isbs.usf.tn)

This work was supported in part by the National Science and Technology Council (NSTC), Taiwan, under Grant NSTC 110-2221-E-224-050-.

ABSTRACT This paper designs and implements a robust and Time-varying Constrained Model Predictive Controller (TCMPC) for the translational and attitude control of a real quadrotor. The proposed approach considers online optimization to find solutions through the hard and soft constraints. All the controller parameters were derived from the experimental test setup and took into consideration the various restrictions and physical constraints associated with the hand-made quadrotor. The proposed controller can possibly linearize and discretize the nonlinear dynamic model of the quadrotor at every sampling time if all constraints and physical restrictions are considered. The performance of the proposed approach was assessed using both a simulation study and a practical implementation. The simulation study considered a quadrotor hovering mode in the presence of wind gusts and encompassed a comparison analysis with a well-tuned Proportional-Integral-Derivative (PID) controller, an Advanced Error model predictive control (AEMPC), and an Efficient MPC (EMPC) approach. For the real-time implementation, an online optimization algorithm was used and tested on the high clock processor ARM A53 on a new attitude test setup. The experimental results, showed that the proposed controller outperformed the unconstrained MPC, the well-tuned PID controller, and EMPC, especially in terms of rejecting the external wind disturbances. The proposed method real-time TCMPC) approach has the advantages of greater robustness and is not heavily dependent upon the accurate dynamics of the model.

INDEX TERMS Quadcopter UAV, constrained MPC, external disturbance, control design, real-time implementation.

I. INTRODUCTION

Over the past two decades, Unmanned Aerial Vehicles (UAVs) have received much attention from both academia and industry. This is mainly due to their broad applications, which include environmental monitoring [1], [2], cultural heritage frames [3], [4], rescuing [5], target tracking [6], pesticide spraying [7] and Value Added Internet of Thing services (VAIOTs) from the sky [8], [9], [10]. Quadrotors, otherwise known as a type of UAVs, have some unique and inherent advantages which make them an important research topic in the field of control engineering. Among their advantages are *a*) simple mechanical structures, *b*) proper maneuverability, *c*) highly nonlinear dynamics of the model, *d*) low-cost structures, and *e*) many civil or defense applications. More importantly, their advantage of vertical take-off and hovering can enable quadrotors to maintain relative flexibility [11], [12], [13], [14], [15].

The associate editor coordinating the review of this manuscript and approving it for publication was Qi Zhou.

When the quadrotors are in their flight missions, they need to precisely follow the set trajectory. Hence, the attitude control of quadrotors plays a crucial role in UAVs when it comes to the accuracy, efficiency, and safety of mission execution. That is, the attitude subsystem of a quadrotor is the vital part of the flight control. There have been many researches into the methods used to control the nonlinear dynamics of the quadrotors. The well-tuned PID, also known as a commercial and popular method on UAVs, Efficient MPC [16], and Advance Error MPC [17], are all used as attitude controllers for quadrotors. However, a major drawback of the above@hyphementioned approaches is the fact that their performance decreases in the presence of disturbances such as wind gusts and parameter uncertainties.

Model Predictive Control (MPC) has proven to be a promising and powerful tool in controlling quadrotors [18], [19]. The remarkable success of MPC hinges on its capability to handle the constraints whilst systematically optimizing the performance in parallel [18], [19]. Moreover, it is capable of tackling problems associated with both the soft and hard constraints on outputs or inputs [17], [20]. MPC has been used to solve many problems pertaining to UAV control such as formation flight control [21], [22], [23], [24], [25], [26], path-planning [5], [27], [28], [29], and trajectory tracking [30], [31]. Nonetheless, standard MPC has high computational complexity [32], [33] since it can predict the future of system behavior [33]. The main advantage of the MPC controller compared to linear controllers is the tuning efficiency. MPC controller has a simpler setup, as object parameters are determined directly. The length of the prediction horizon and the control horizon in this case are determined depending on the time for which the parameter to be monitored must take a given value. While, when setting up the linear controllers, the object parameters are not explicitly taken into account [34].

In [16], the Efficient Model Predictive Control (EMPC) algorithm was used to establish the linear internal model with the dynamics of a quadrotor taken into account. This was to better reduce the prediction points. In the EMPC method, the future output of the quadrotor was calculated with a closed-loop prediction structure based on a linear internal model. Due to this linear structure, the computational time was reduced greatly for the standard MPC. This method was tested on a real quadrotor and demonstrated satisfactory tracking performance while the standard MPC, in contrast, failed to work [16]. Another method is the Advanced Error-based MPC (AEMPC) presented in [17], which adopted an augmented model to remove the tracking error caused by external disturbances. To better reduce the computation time, the control input sequence in MPC was estimated with Laguerre function, which also helped to improve the closed-loop performance. As a result, AEMPC proved to be effective when it comes to disturbance rejection, trajectory fast tracking and quadrotor stability. Reference [30] introduced a high-level lateral-directional trajectory-tracking controller embedded in a nonlinear MPC (NMPC) structure for fixed-wing UAV trajectory tracking in wind and arbitrary path combinations. It was proven that the method could produce better results of trajectory tracking. In addition, [5] used NMPC for navigation and obstacle avoidance of an UAV. This method can also avoid dynamic obstacles. Besides, this method was designed to be compatible with a classification scheme that predicts the positions of possible obstacles. The study into the trajectory tracking of ducted fan aircrafts was conducted by [31]. This article proposed a Compound Flight Control (CFC) technique based on the concept of MPC for attitude control of ducted fan aircrafts. As a result, they achieved flight robustness and adequate tracking without compromising the nominal performance. In [35], the authors combined Gaussian Process (GP) with MPC to develop an accurate real-time feedback controller. The simulation results indicated significant reduction in trajectory tracking errors. A hybrid model was proposed by [36] so as to compensate residual and uncertain dynamics within the system. In comparison with the GP model, this hybrid model showed more accurate predictions of the quadrotor dynamics. The NMPC model in agile quadrotor flight was studied by [37], which proposed a hybrid adaptive NMPC, called L_1-NMPC. This hybrid method can learn model uncertainties online and compensate them immediately, thereby resulting in more flexibility and robustness in comparison with non-adaptive NMPC.

Although these papers demonstrated very good simulation results in this regard, their efficiency in real world experiments are still vague, especially in the presence of external distances. These methods have limitations such as high computational time, the need for an accurate dynamic model of a drone, and the consideration given to the constraints of all motors and sensors. To better reduce these problems, several famous drone companies have managed to use well-tuned PID controllers as commercial flight controllers [38].

This paper presents a practical time varying constrained MPC control method.

The main contributions of this article are as follows:

1. Designing a time-varying model predictive controller with the consideration given to full input/ output constraints of the quadrotor with online optimization.
2. Designing a new attitude control test platform for quadrotors with the equality of rotation and mass center consideration.
3. Extraction of the dynamic model parameters of our handmade quadrotor through experimental and simulated tests.
4. Comparison analysis with EMPC, AEMPC and a well-tuned PID.

The remainder of this paper is organized as follows. First, discussions on the nonlinear dynamics of the quadrotor are provided in Section 2. Section 3 deals primarily with the mathematical formulation of the specific controller, which was designed based on the concept of our proposed MPC. Section 4 includes simulation and experimental studies with multiple test cases involved to validate the effectiveness of the proposed method. Finally, the conclusion is drawn in Section 5.

TABLE 1. **Extracted parameters of the dynamic model of a quadrotor.**

Parameters	Definition	Value	Units
I_{xx}	Moment of inertia of the quadrotor around the B_x axis	2.5×10^{-3}	$kg.m^2$
I_{yy}	Moment of inertia of the quadrotor around the B_y axis	2.5×10^{-3}	$kg.m^2$
I_{zz}	Moment of inertia of the quadrotor around the B_z axis	4.3×10^{-3}	$kg.m^2$
b	Thrust coefficients	1.85×10^{-6}	$N.s^2$
d	Drag coefficients	1.063×10^{-7}	$N.m.s^2$
$m_{quadrotor}$	The whole mass of the quadrotor	6.55×10^{-1}	Kg
m_{rotor}	Mass of rotor	5×10^{-2}	Kg
J_r	Moment of inertia of the rotor about its axis of rotation	2.3×10^{-6}	$kg.m^2$
L	Length of quadrotor arm	1.43×10^{-1}	M

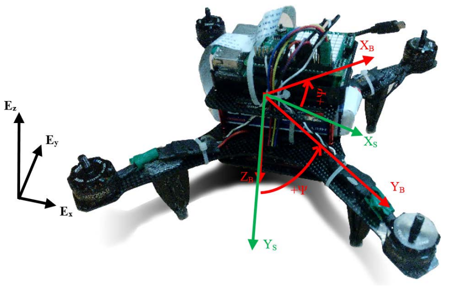


FIGURE 1. **The designed quadrotor configuration frame system.**

II. DYNAMIC MODELING OF THE QUADROTOR

The hand-made quadrotor and its coordinate system are depicted in Fig1. This figure demonstrates the body-fixed frame (BFF) $\{X_B, Y_B, Z_B\}$, quadrotor coordinate system, and Earth-fixed frame (EFF) $\left\{E_x, E_y, E_z\right\}$. As shown in this figure, there is angle differences between the sensor coordinate frame and the body frame $\{X_s, Y_s, Z_s\}$.

Considering the model dynamics of the quadrotor, some assumptions are made to simplify the dynamic equations: a) The structure of the quadrotor and propellers are rigid and symmetrical b) The quadrotor structure is balanced to match the center of mass with the center of rotation c) Each pair of motors with the same direction of rotation are aligned in one axis extending from the center of rotation. These assumptions lead to a nonlinear six degrees of freedom dynamic equation of the quadrotor model, which is expressed in Equation (1) [39]. In this Equation, m is the mass of the system, g the gravitational acceleration in local position, and U_1 the total thrust that the quadrotor can hover. U_2, U_3, and U_4 are the inputs of the roll, pitch, and yaw channels respectively. Moreover, u_x and u_y are sub-inputs to position channels, which are derived from Equation (2). Moreover, the disturbance vector $\left[0\ \omega_\varphi\ 0\ \omega_\theta 0\ \omega_\psi\ 0\omega_z\ 0\ \omega_x\ 0\omega_y\right]$, is considered and added to the model which could affect the rates of states $\left[\dot\varphi\ \dot\theta\ \dot\psi\ \dot z\ \dot y\ \dot x\right]$ in attitude and position equations [39]. Ω_r is considered as a disturbance caused by the angular velocities of four motors, and other parameters are provided in Table 1.

$$\begin{bmatrix} \dot\varphi \\ \ddot\varphi \\ \dot\theta \\ \ddot\theta \\ \dot\psi \\ \ddot\psi \\ \dot z \\ \ddot z \\ \dot x \\ \ddot x \\ \dot y \\ \ddot y \end{bmatrix} = \begin{bmatrix} \dot\varphi \\ \dot\theta\dot\psi\left(\dfrac{I_{yy}-I_{zz}}{I_{xx}}\right)+\dot\theta\dfrac{J_r}{I_{xx}}\Omega_r+\dfrac{l}{I_{xx}}U_2 \\ \dot\theta \\ \dot\varphi\dot\psi\left(\dfrac{I_{zz}-I_{xx}}{I_{yy}}\right)-\dot\varphi\dfrac{J_r}{I_{yy}}\Omega_r+\dfrac{l}{I_{yy}}U_3 \\ \dot\psi \\ \dot\theta\dot\varphi\left(\dfrac{I_{xx}-I_{yy}}{I_{zz}}\right)+\dfrac{l}{I_{zz}}U_4 \\ \dot z \\ -g+\dfrac{(Cos\varphi Cos\theta)}{m}U_1 \\ \dot x \\ \dfrac{u_x}{m}U_1 \\ \dot y \\ \dfrac{u_y}{m}U_1 \end{bmatrix} + \begin{bmatrix} 0 \\ \omega_\varphi \\ 0 \\ \omega_\theta \\ 0 \\ \omega_\psi \\ 0 \\ \omega_z \\ 0 \\ \omega_x \\ 0 \\ \omega_y \end{bmatrix} \tag{1}$$

$$\begin{aligned} u_x &= (Cos(j)Sin(q)Cos(y)+Sin(f)Sin(y)) \\ u_y &= (Cos(j)Sin(q)Sin(y)-Sin(f)Cos(y)) \end{aligned} \tag{2}$$

$$U = \begin{bmatrix} U_1 \\ U_2 \\ U_3 \\ U_4 \end{bmatrix} = \begin{bmatrix} b & b & b & b \\ 0 & -b & 0 & b \\ b & 0 & -b & 0 \\ -d & d & -d & d \end{bmatrix} \begin{bmatrix} \omega_1^2 \\ \omega_2^2 \\ \omega_3^2 \\ \omega_4^2 \end{bmatrix} \tag{3}$$

Rotor rotation creates vertical forces corresponding to the thrust (T). In addition, each rotor produces a moment perpendicular to the plane of the propeller rotation leading to the horizontal movement (H). Aerodynamic forces like thrust and drag are proportional to the square of the propeller's speed. A full and detailed explanation of these forces can be found in [39].

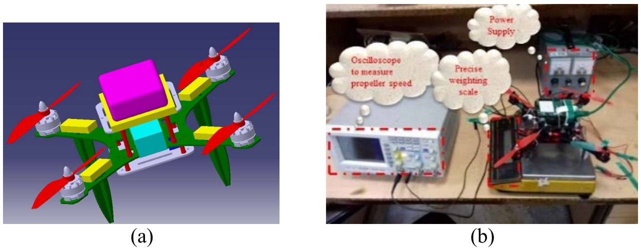


FIGURE 2. (a) Simulated quadrotor with CATIA software, (b) experimental setup to achieve thrust parameter.

To achieve the correct value of these parameters like moments of inertia, experimental tests were implemented and compared with Catia software calculation results. These experimental tests were repeated to show the repeatability of the achieved results.

The results of Catia's software calculation were almost the same as the experimental test results, thus, verifying the accuracy of the experimental tests. The achieved model parameters via experimental tests are illustrated in Table 1.

As it can be seen from Equation (1) [39], the attitude subsystem equations are nonlinear and continuous. To control the attitude sub-system by the MPC method, linearization and discretization are required according to the sampling rate (Ts).

$$\begin{aligned} X_{At}(k+1) &= A_j^* \mathbf{X_{At}}(k) + B_j^* \mathbf{U}(k) + \mathbf{W_{At}} \\ Y_{At}(k) &= C_j^* X_{At}(k \\ & j \in \{0, T_s, 2T_s, \ldots, kT_s\}, \quad k \in Z^+ \end{aligned} \tag{4}$$

$$\begin{aligned} A_j^* &= \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & a_2\Omega_r + a_1 x_6 & 0 & a_1 x_4 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & a_3 x_6 - a_4\Omega_r & 0 & 0 & 0 & a_3 x_2 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & a_4 x_4 & 0 & a_5 x_2 & 0 & 0 \end{bmatrix} \\ B_j^* &= \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & b_1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & b_2 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & b_3 \end{bmatrix} \\ C_j^* &= \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix} \\ W_{At} &= \begin{bmatrix} 0 \\ \omega_\varphi \\ 0 \\ \omega_\theta \\ 0 \\ \omega_\psi \end{bmatrix} \end{aligned} \tag{5}$$

The state-space representation is shown in Equation (4) [39], where the matrices $\boldsymbol{B}_{6\times4}$ and $\boldsymbol{C}_{3\times6}$ are derived from linearization and discretization, U is defined as $U = [\delta u_1 \ \ \delta u_2 \ \ \delta u_3 \ \ \delta u_4]$ and Ω_r is a constant disturbance. The matrix A_j^* is discretized and linearized around the operating points $\left[0\ \dot{\phi}^\circ\ 0\ \dot{\theta}^\circ\ 0\ \dot{\psi}^\circ\right]$ and in every iteration (j), new state feedback updates $\boldsymbol{A}_j^*$ matrix which is a time-varying matrix. This time dependency helps the quadrotor to have high maneuverability and robustness against wind gusts.

III. THE PROPOSED METHOD

The proposed controller uses a time-varying linear model of the handmade drone, and considers the parameters generated in the previous section. The controller updates the nonlinear attitude subsystem at every sampling time around a new equilibrium point. This enables a time-varying model of the MPC to maintain high precision and stability thereby making it suitable for environments with disturbances and system constraints [40]. Moreover, this controller uses Hildreth as the online optimization method. This latter is considered as a powerful way of solving a large system of inequalities with sparse matrices without the involvement of any matrix inversion [41]. The online optimization algorithm is more complex because this algorithm requires a large amount of calculation so as to better solve time-varying systems with different types of constraints. The presented method can possibly augment the state-space matrices if the integrator effect is applied to do so. This enables the system to track the reference point more stably with approximately zero steady-state error.

The optimal control signal was achieved by optimizing the desired cost function. The presented controller was based on the MPC principle so required the nonlinear model of the quadrotor with real physical parameters. Nonlinear dynamic equations were converted to nonlinear state-space equations and then discretized. After discretization, the achieved model was linearized around the operating point with respect to the time. The state-space matrices were augmented and used to generate optimal control signals that restricts the quadrotor's motion. The whole process should be repeated in each iteration with respect to each sampling time (T_s). The augmented state-space composed from extracted matrices, A_j^* and B_j^* of the nonlinear model are shown in Equation (6).

$$\begin{aligned} \overbrace{\begin{bmatrix} \Delta\mathbf{x}_\mathrm{m}(k+1) \\ \mathbf{y}(k+1) \end{bmatrix}}^{\mathbf{x}(k+1)} &= \overbrace{\begin{bmatrix} \mathbf{A}_j^* & 0_m^T \\ \mathbf{C}_j^*\mathbf{A}_j^* & I_{q\times q} \end{bmatrix}}^{\mathbf{A}} \overbrace{\begin{bmatrix} \Delta\mathbf{x}_\mathrm{m}(k) \\ \mathbf{Y}(k) \end{bmatrix}}^{\mathbf{x}(k)} \\ &+ \overbrace{\begin{bmatrix} \mathbf{B}_\mathrm{j}^* \\ \mathbf{C}_j^*\mathbf{B}_j^* \end{bmatrix}}^{\mathbf{B}} \Delta\mathbf{u}(k) \end{aligned} \tag{6}$$

$$\mathbf{y}(k) = \overbrace{\begin{bmatrix} 0_m & \mathbf{I}_{q\times q} \end{bmatrix}}^{\mathbf{C}} \begin{bmatrix} \Delta\mathbf{x}_m(k) \\ \mathbf{y}(k) \end{bmatrix} \tag{7}$$

where $x(k)$ is the state vector, $\Delta u(k)$ is control action rate defined as:

$$\begin{cases} \Delta u(k) \in \boldsymbol{U} \subseteq \Re^4 \\ x(k) \in \boldsymbol{X} \subseteq \Re^6 \end{cases} \tag{8}$$

The sets X and U specify possible state and input rate constraints. $\left[A_j^*, B_j^*, C_j^*\right]$ specify the matrices which are linearized around the operating point $\left[0, \dot{\phi}^\circ, 0, \dot{\theta}^\circ, 0, \dot{\psi}^\circ\right]$ at instant kTs. In practice, many control algorithms exhibit decreased performance and even fail when considering quadrotor's mechanical and electrical restrictions. Hence, actuator saturation and restricted mechanical motions should be considered as constraints when deriving an optimal control input. In this pape, constraints on the quadrotor's output states, control effort signals and their variation have been illustrated in Eq. (9). The matrix $C'_{9\times 12}$ consists of outputs that must be constrained. All constraints can be written in a more compact form

$$\begin{bmatrix} \mathbf{H_1} \\ \mathbf{H_2} \\ \vdots \\ \mathbf{H_{15}} \\ \mathbf{H_{16}} \end{bmatrix} \Delta\mathbf{U} \leq \begin{bmatrix} u_1^{\max} \\ \vdots \\ u_4^{\max} \\ \Delta u_1^{\max} \\ \vdots \\ \Delta u_4^{\max} \\ y_{cu} - u\,(k-\mathbf{1}) \\ y_{cd} - u(k-\mathbf{1}) \end{bmatrix} \tag{9}$$

All states and input constraints have been combined by using a set of $\mathbf{H_i}$ zeroed $2 \times (m+n)$ matrices. m is number of state vector of x(k) and n is the number of control actions.

Some experimental tests are carried out to calculate the rotor torque. Then, the ranges of variation are determined for the system states, which are considered asoutputs of the system, and control effort signals as well. The proposed controller tries to minimize the cost function subject to the following constraints:

$$\begin{aligned} &Min\ J\ (\Delta u) = \frac{1}{2}\Delta u^T E \Delta u + \Delta u^T F \\ &Subject\ to:\ H_i \Delta u \leq b \end{aligned} \tag{10}$$

We define the cost function J that reflects the control objective as Eq (10) where the first term reflects the consideration given to the size of ΔU when the objective function J is made to be as small as possible and second term is linked to the objective of minimizing the errors between the predicted output and the set-point signal. E is a diagonal matrix that tuned for the desired close loop performance.

In every iteration, the cost function is minimized and the optimal Δu is calculated. There is a tradeoff between the set point tracking errors and the produced optimal control signals. The block diagram at Fig (3) shows the proposed controller functionality in more clear way.

$$E = \left(H'H + \lambda\right)^{-1} \quad F = -\left(H'W - HfX_f\right) \tag{11}$$

where $\lambda_{(N_p\times n)\times(N_c\times m)}$ is a parameter that can change the mentione trade-off. $H_{(N_p\times n)\times(N_c\times m)}, f_{(N_p\times n)\times(N_c\times m)}$ are matrices including augmented matrices shown in Eq. (6). The number of approximations of the nonlinear model (j) depends on flight time (kT_s). Note that system stability when switching between linear models is not approved, but no stability issues were observed in any experimental tests.

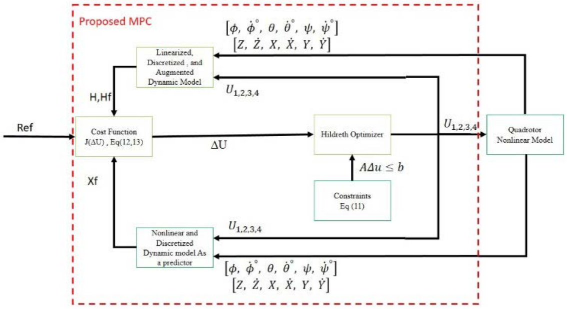

Proposed MPC
$[\phi, \dot{\phi}^\circ, \theta, \dot{\theta}^\circ, \psi, \dot{\psi}^\circ]$
$[Z, \dot{Z}, X, \dot{X}, Y, \dot{Y}]$
Linearized, Discretized , and Augmented Dynamic Model
$U_{1,2,3,4}$
H,Hf
Ref
Cost Function J(ΔU) , Eq(12,13)
ΔU
Hildreth Optimizer
$U_{1,2,3,4}$
Quadrotor Nonlinear Model
$A\Delta u \leq b$
Xf
Constraints Eq (11)
$U_{1,2,3,4}$
Nonlinear and Discretized Dynamic model As a predictor
$[\phi, \dot{\phi}^\circ, \theta, \dot{\theta}^\circ, \psi, \dot{\psi}^\circ]$
$[Z, \dot{Z}, X, \dot{X}, Y, \dot{Y}]$

FIGURE 3. Block diagram of the proposed controller.

IV. SETUP AND RESULTS

In this section, flight experiments and numerical simulations are explained to verify the performance of the TCMPC algorithm in quadrotor attitude and position tracking. The parameters of the hand-made quadrotor are taken from Tables 1 and 2.

A. SIMULATION RESULTS

The attitude, position model, and controllers of the quadrotor are implemented in MATLAB/Simulink. Then, the performance of the TCMPC is compared with AEMPC, EMPC, and PID methods. Since the quadrotor is an under-actuated system, the inner loop controller should be at least four times faster than the outer loop controller if the purpose is to better control the position of the quadrotor by attitude system ($\mathrm{T_{sAttitude}}$ =0.02s, $\mathrm{T_{sPosition}}$ = 0.2s). The constraints on the effort signals rate ($\Delta U_{i=1,2,3,4}$) depend on the time constant of the speed controllers of motor propeller systems and affects the transient performance of the quadrotor. The constraints on the inpu and the output states are computed based on a) The drag and thrust factors, b) the specifications of the IMU sensor used, c) the torque produced by the motors, d) the quadrotor's mass and moments of inertia.

The flight scenario description is provided as follows: At the beginning, th quadrotor goes up from the initial point to the target point, as shown in Fig 4. When the quadrotor reaches to a height of 0.7 meters, the wind as a disturbance $\mathbf{w_{At}}$ starts blowing to the quadrotor. For the better understanding of this to occur, the origin bounding control inputs and its rates are considered as an equilibrium state with u(0)=0. Based on the motors' angular velocities, Ω_k, k = 1, ..., 4 and $\Omega_k \in [0, \Omega_k^{\max}]$, the effort signals bounding set, $U_{k=1,2,3,4}$, are derived (36).

The constraints are define in Equation (13):

$$\begin{cases} 0N \leq U_1 \leq 12N \\ -5N \leq U_2 \leq 5N \\ -5N \leq U_3 \leq 5N \\ -0.5N.m \leq U_4 \leq 0.5N.m \end{cases}$$

$$\begin{cases} -0.5N \leq \Delta U_1 \leq 0.5N \\ -0.5N \leq \Delta U_2 \leq 0.5N \\ -0.5N \leq \Delta U_3 \leq 0.5N \\ -0.1N.m \leq \Delta U_4 \leq 0.1N.m \end{cases} \tag{12}$$

$$\begin{bmatrix} -\frac{\pi}{4} \\ -0.75 \\ -\frac{\pi}{4} \\ -0.75 \\ -\pi \\ -1 \\ -3 \\ -3 \\ -3 \end{bmatrix} \leq y = \begin{bmatrix} \phi(rad) \\ \dot{\phi}\left(\frac{rad}{s}\right) \\ \theta(rad) \\ \dot{\theta}\left(\frac{rad}{s}\right) \\ \psi(rad) \\ \dot{\psi}\left(\frac{rad}{s}\right) \\ \dot{z}\left(\frac{m}{s}\right) \\ \dot{x}\left(\frac{m}{s}\right) \\ \dot{y}\left(\frac{m}{s}\right) \end{bmatrix} \leq \begin{bmatrix} \frac{\pi}{4} \\ 0.75 \\ \frac{\pi}{4} \\ 0.75 \\ \pi \\ 1 \\ 3 \\ 3 \\ 3 \end{bmatrix} \tag{13}$$

Regarding the EMPC method [16], this method uses the simple linear dynamic model of quadrotor, which changes during flight time to generate the prediction points without using an online optimization algorithm. The EMPC needs some prediction points while the TCMPC, in contrast, requires the first prediction point from the augmented model. Hence, the iteration time, otherwise known as the minimum time to generate each control signal, would be different in each method. Also, a comparison between the proposed MPC and EMPC approaches is illustrated in practice in the presence of wind gusts.

The Ziegler-Nichols oscillation method is commonly used for tuning the PID controller parameters of each channel (39). The common PID equation is shown in Equation (14).

$$u(t) = K_c\left(\varepsilon(t) + \frac{1}{\tau_i}\int_0^t \varepsilon(t')\,dt' + \tau_d\frac{d\varepsilon(t)}{dt}\right)$$

$$K_i = K_c\frac{1}{\tau_i}, K_d = K_c\tau_d \tag{14}$$

where u is the control signal, ε is the error, K_c, K_i, K_d are the gains for the PID controller, τ_i is the parameter that scales the integral controller, τ_d is the parameter that scales the derivative controller, t is the time taken for error measurement. In the first step, the gains Ki and K_d are equal to zero. The proportional gain, $\mathrm{K_c}$, is increased until it reaches the ultimate gain. This makes the output of the system oscillate constantly with period of Tu. By using $\mathrm{T_u}$ and $\mathrm{K_c}$, all other parameters listed in Table were obtained. Then, the well-tuned gains are achieved by changing the gains around the obtained values experimentally. Table presents the obtained parameter value that are used as PID gains for each channel in practice.

Following this, the TCMPC method is compared with AEMPC, EMPC and PID controller in simulation. The initial values of states are [0,0,0,0,0,0,0.05,0,1,0,1,0] in the simulation. The value of each parameter of algorithms is taken from Table 4.

TABLE 2. Ziegler-Nichols method.

Control Type	K_c	K_i	K_d
P	$0.5K_u$	-	-
PI	$0.45K_u$	$0.54K_u/T_u$	-
PID	$0.6K_u$	$1.2K_u/T_u$	$3K_u/T_u/40$

TABLE 3. PID gains in practice.

Control Type	K_c	K_i	K_d
PID Roll channel	5.5	2	3
PID Pitch channel	5.5	2	3
PID Yaw channel	3	0.5	2

TABLE 4. Parameters that are used in the simulations and practice.

TCMPC Parameters	Description	Value	AEMPC Parameters	Value	EMPC Parameters	Value
$\boldsymbol{N_p}$	Prediction horizon	15	$\boldsymbol{N_p}$	15	$\boldsymbol{N_p}$	80
$\boldsymbol{N_c}$	Control horizon	3	$\boldsymbol{N_c}$	3	$\boldsymbol{N_c}$	5
$\boldsymbol{T_{s,Attitude}}$	Attitude sampling time	0.02	$\boldsymbol{T_{s,Attitude}}$	0.02	$\boldsymbol{T_{s,Attitude}}$	0.05
$\boldsymbol{T_{s,position}}$	Position sampling time	0.2	$\boldsymbol{T_{s,position}}$	0.2	$\boldsymbol{T_{s,position}}$	0.05
$\boldsymbol{\lambda_p}$	Weight of Pitch channel signal	0.7	$\boldsymbol{\alpha_{s,\varepsilon}}$	0.3		
$\boldsymbol{\lambda_{Roll}}$	Weight of Roll channel signal	0.2	$\boldsymbol{\alpha_{s,n}}$	0.5		
$\boldsymbol{\lambda_\psi}$	Weight of Yaw channel signal	1.5	$\boldsymbol{\lambda_s}$	1.5		
$\boldsymbol{\lambda_x}$	Weight of X channel signal	0.2				
$\boldsymbol{\lambda_y}$	Weight of Y channel signal	0.2				
$\boldsymbol{\lambda_z}$	Weight of Z channel signal	0.1				
$\boldsymbol{n}$	Number of Hildreth algorithm iteration	10				
$\boldsymbol{E}$	Convergence error rate	10^{-o}				
IT	Iteration time	45us	**IT**	88us	IT	149us

Comparison between the effort signals of controllers are presented in Figs 5, 7, and 9. According to these figures, at the beginning, the quadrotor initial position changes instantly by consuming effort signals in a constrained way. Consequently, the TCMPC controller produces optimal effort signals to control the quadrotor while facing wind disturbance. Unconstrained rate of control signals ($\Delta U_{k=1,2,3,4}$) cause the sharp movements and high peaks at $U_{k=1,2,3,4}$ signals when using the AEMPC, EMPC, PID methods as shown in Figs 5, 7, and 9. In addition, the AEMPC, EMPC, and PID have sharper and higher overshoots in presence of wind, whereas the TCMPC shows a less disturbed reaction. Overall, the wind gust has effect on all methods during flight.

The results of the tracking reference signal by AEMPC, EMPC, and the proposed algorithm are depicted in Figs 4 and 6, respectively. The results confirm the ability of the two controllers to track the desired path well. The proposed method, however, shows less fluctuations at takeoff and achieves better performance than AEMPC and EMPC in the presence of disturbances. According to Fig 8, the Roll and Pitch channels change to reach the desired zero value and the Yaw channel tracks the pulse signal. The maximum domain of the pulse is ten degrees and the minimum value is zero.

Both methods start to control the quadrotor from the same initial point and our proposed method shows lower rise time compared to the PID. Fig 9 demonstrates the required energy for stabilizing the quadrotor at roll and pitch channels for both well-tuned PID and TCMPC. Note the large overshoots

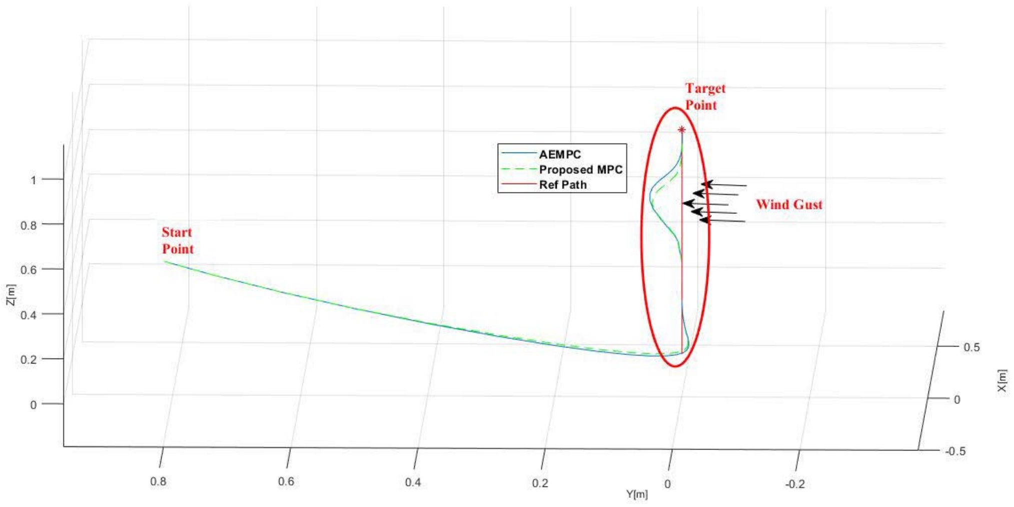


FIGURE 4. The 3D path of the quadrotor in hovering mod.

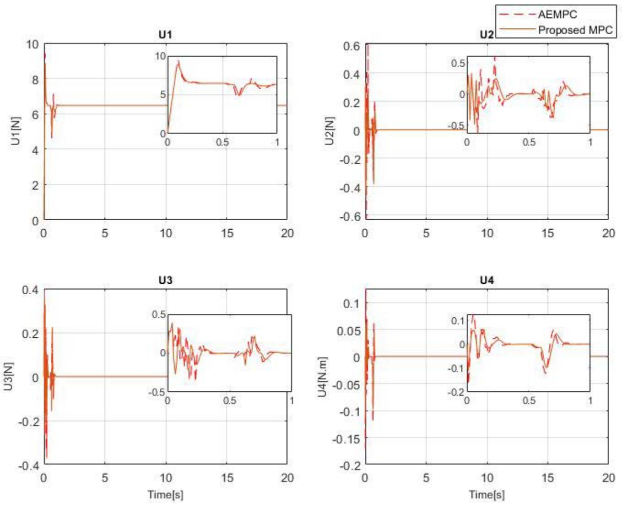


FIGURE 5. The quadroto's effort signals during flight.

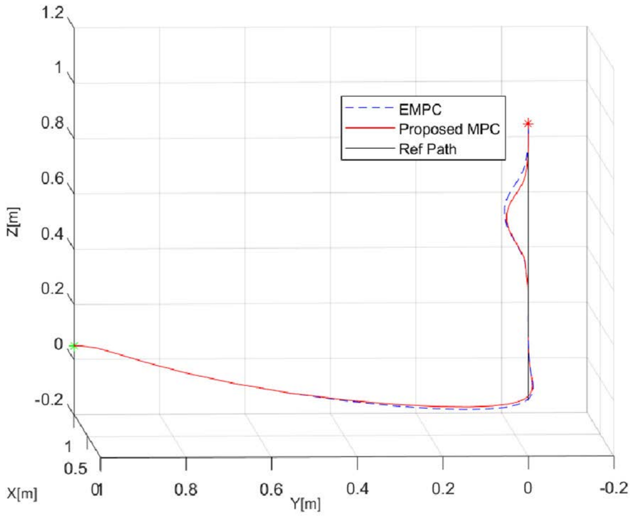


FIGURE 6. Comparison between EMPC and the TCMPC at hovering mode.

and undershoots in the responses of the PID controller when tracking the pulse signal while the TCMPC shows less effort and lower changes in the response. A comparison of the root mean square error (RMS) between TCMPC, AEMPC, EMPC and PID techniques is presented in Fig 22. The RMS is calculated according to the following formula:

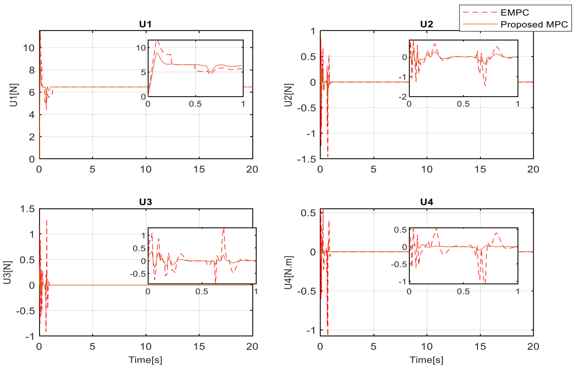


FIGURE 7. Comparison of effort signals between EMPC and the TCMPC algorithm.

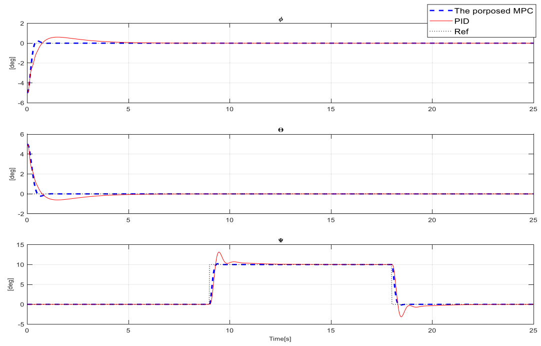


FIGURE 8. Comparison between attitude behavior of the proposed MPC and well-tuned PID simulated in MATLAB.

$$RMS = \sqrt{\sum_{i=1}^{N} \frac{e^2(i)}{N}} \tag{15}$$

In Equation 15, N is defined as the number of samples, $e(i)$ is the position tracking error for Fig 3 and 5. In the other figures, $e(i)$ is considered as the attitude tracking error.

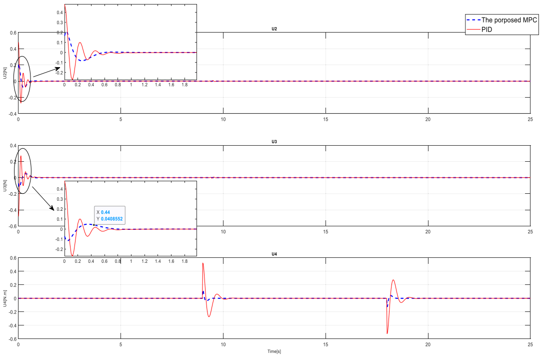


FIGURE 9. Comparison between effort signal of both controllers in simulation environment.

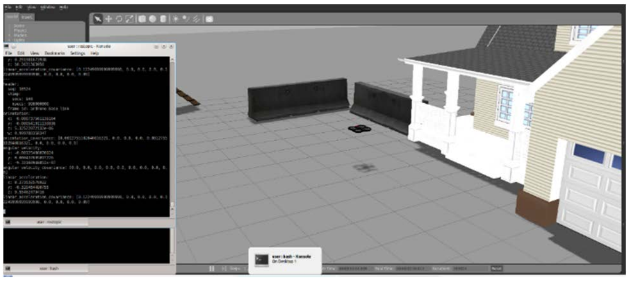


FIGURE 10. Gazebo simulator.

Fig 22 shows the comparisons of tracking and control performances. It is clear that whether there is wind disturbance or not, the TCMPC algorithm gets the smallest RMS value among the four controllers in simulation.

B. SETUP AND EXPERIMENTAL RESULTS

In order to test whether the proposed controller can be implemented feasibly on the real quadrotor, the Gazebo simulator was used in the first stage (Fig 10). By using the AR.Drone2.0 on Gazebo7 simulator and writing the specific C++ node and topics, the proposed controller shows its ability to control the drone during hovering motion in this simulator. The tum-simulator package was used to communicate with the drone. Meanwhile, all physical parameters such as the momentum of inertia, mass, gravity, and the noise on the output of the IMU sensor were considered as well. ROS Indigo version was used, and a new launch file run in a virtual machine.

In the Gazebo simulator, the effort signals are published to the quadrotor, and feedback signals of the quadrotor are logged during hovering. Based on the flight scenario described in section 4.1, first, the quadrotor takes off to reach desire point (0, 0, 1) at space with respect to EEF coordinate. Fig 10 is related to the effort signals of TCMPC. Moreover, as can be seen from Fig 11, the position errors around three axes converge to zero.

The TCMPC was applied to the real quadrotor and its controlling performance was compared to a well-tuned PID controller and EMPC method. This controller is a commercial method, which can be employed to demonstrate how the states and control signals affect performance. All the components and modules involved in the design of this quadrotor are illustrated in Fig 13. LSM9DS1 was used as an Inertia Measurement Unit (IMU) sensor to produce angular rates and accelerations in three axes.

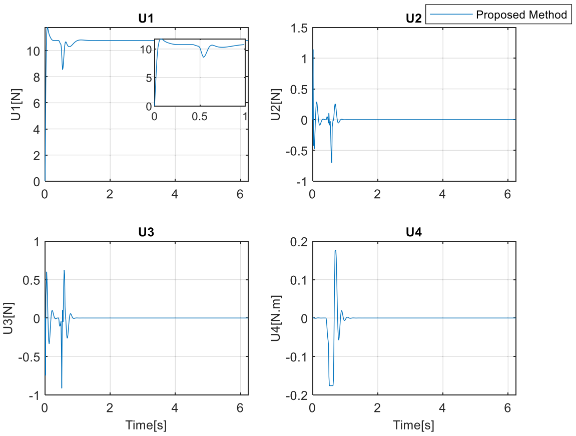


FIGURE 11. The effort signals logged and plotted in the MATLAB software.

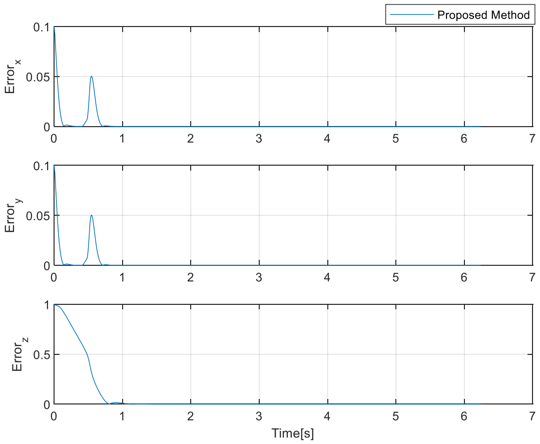


FIGURE 12. The position error logged and plotted in the MATLAB software.

A Kalman filter was utilized to estimate the target variables such as roll, pitch, and yaw. In addition, the IMU sensor was calibrated, and the bias and scale factor were obtained by the least square algorithm. The accuracy of the obtained Euler angles was 0.1 degrees and Kalman filter initial covariance value was $Q_\circ = [0.0010; 00.0003]\,\frac{rad}{s}$. As the motors receive commands from an electronic speed controller (ESC), an interface board such as Arduino DEU is needed. This board produces pulse width modulation (PWM) to the ESC driver as controller command. The sensor output frequency was 100 Hz. The bandwidth was set to 100 Hz, which is two times more than the quadrotor dynamic range. The ESC driver rotor receives PWM signals with frequency of 250 Hz, and so both controllers were designed to operate at a frequency of 50 Hz. The other parameters of this controller are listed in Table 4. According to this table, the obtained state-space matrices are presented in Equation (16)

Fig 14 demonstrates the new attitude test setup. This helps limit quadrotor motion to three degrees of freedom and only its attitude can be changed. Since the center of mass should be on the rotation center, four packs with the same weights

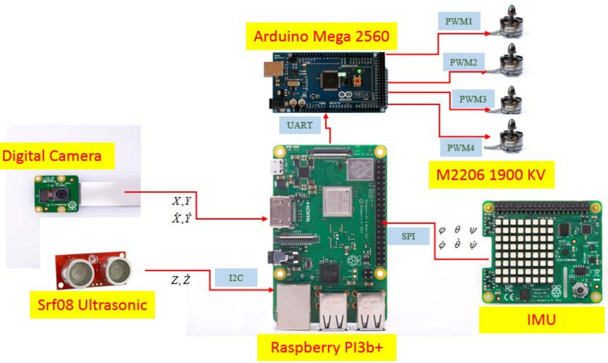


FIGURE 13. Quadrotor hardware components.

should be added to the quadrotor setup. These packs can change the center of mass to model the dynamic equations accurately.

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -0.0258\dot{\psi} & 0 & -0.0258\dot{\theta} \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0.0251\dot{\psi} & 0 & 0 & 0 & 0.0251\dot{\phi} \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0.00065087\dot{\theta} & 0 & 0.0006508\dot{\phi} & 0 & 0 \end{bmatrix}$$

$$B = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 18.7657 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 18.7782 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 130.1744 \end{bmatrix}$$

$$C = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix} \tag{16}$$

$$\mathrm{W_{At}} = \begin{bmatrix} 0 \\ \omega_\varphi \\ 0 \\ \omega_\theta \\ 0 \\ \omega_\psi \\ 0 \\ \omega_z \\ 0 \\ \omega_x \\ 0 \\ \omega_y \end{bmatrix} = \begin{bmatrix} 0 \\ 3.5 \\ 0 \\ 3.5 \\ 0 \\ 2 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \tag{17}$$

In order to exhibit how effectively constrains can be added to the controller, the performance of the quadrotor was compared with the normal MPC without constraints, well-tuned PID, and EMPC in a practical environment. Two different scenarios are considered in this regard: the first one is the same as flight scenario described at Section 4.1; the second one is sinusoidal tracking and disturbance rejection.

Regarding the first scenario, the specific command was sent to change the yaw channel of the quadrotor, as shown in Figs 14, 17 and 19. This command in the Yaw channel had a 15-degree amplitude positive pulse with a period of 7 seconds. The frequency of the control loop and control parameters were found to be the same in all methods. Constraints were chosen according to Equations (18) and (19).

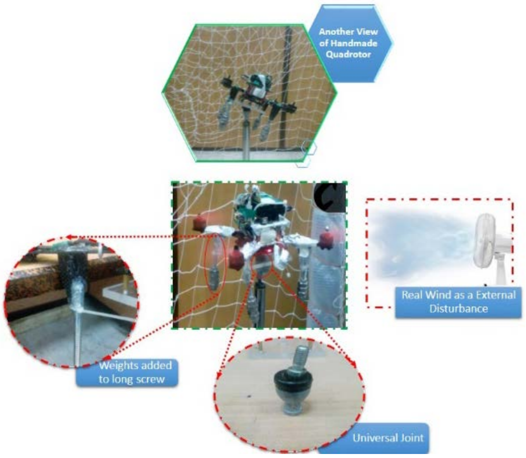

Another View of Handmade Quadrotor
Real Wind as a External Disturbance
Weights added to long screw
Universal Joint

FIGURE 14. Quadrotor attitude control setup.

$$\begin{cases} -0.5N \leq \Delta U_1 \leq 0.5N \\ -0.5N \leq \Delta U_2 \leq 0.5N \\ -0.5N \leq \Delta U3 \leq 0.5N \\ -0.1N.m \leq \Delta U_4 \leq 0.1N.m \end{cases}$$
$$\begin{cases} 0N \leq U_1 \leq 12N \\ -5N \leq U_2 \leq 5N \\ -5N \leq U_3 \leq 5N \\ -0.5N.m \leq U_4 \leq 0.5N.m \end{cases} \tag{18}$$

$$\begin{bmatrix} -\dfrac{\pi}{4} \\ -0.75 \\ -\dfrac{\pi}{4} \\ -0.75 \\ -\pi \\ -1 \\ -3 \\ -3 \\ -3 \end{bmatrix} \leq y = \begin{bmatrix} \phi(rad) \\ \dot{\phi}\left(\dfrac{rad}{s}\right) \\ \theta(rad) \\ \dot{\theta}\left(\dfrac{rad}{s}\right) \\ \psi(rad) \\ \dot{\psi}\left(\dfrac{rad}{s}\right) \\ \dot{z}\left(\dfrac{m}{s}\right) \\ \dot{x}\left(\dfrac{m}{s}\right) \\ \dot{y}\left(\dfrac{m}{s}\right) \end{bmatrix} \leq \begin{bmatrix} \dfrac{\pi}{4} \\ 0.75 \\ \dfrac{\pi}{4} \\ 0.75 \\ \pi \\ 1 \\ 3 \\ 3 \\ 3 \end{bmatrix} \tag{19}$$

The experimental results are presented in Figs 15, 17. According to these figures, three controllers could control the quadrotor from same initial position to the desired set point. The quadrotor's yaw channel tracks the input pulse command while the roll and pitch channels, in contrast, track zero input commands. The comparison indicates that the TCMPC method presents less overshoots and undershoots than the unconstrained MPC and well-tuned PID approaches.

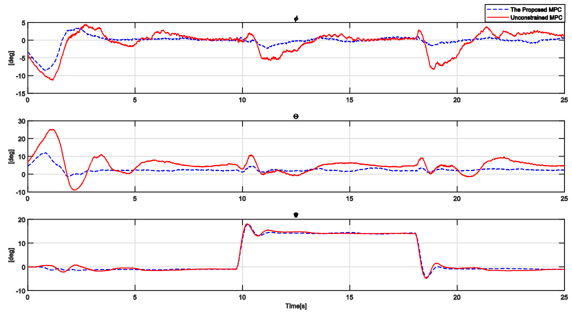


FIGURE 15. Comparison of the proposed and unconstrained MPC applied to quadrotor.

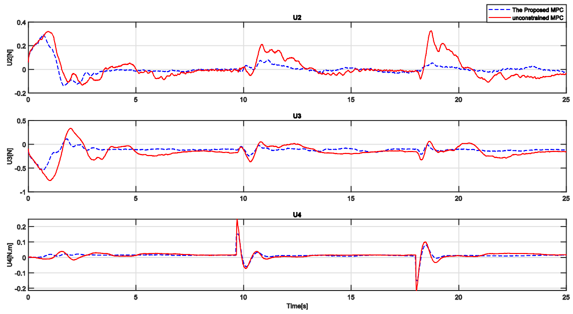


FIGURE 16. Control effort signals of the proposed controller and unconstrained MPC to track.

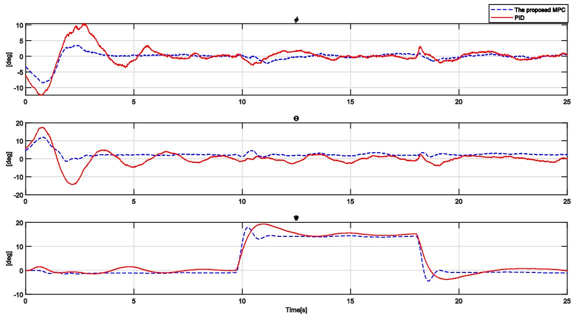


FIGURE 17. Attitude sub-system comparison between the proposed MPC and PID controllers.

Figs 16, 18 present the effort control signals of the three controllers. It is obvious that the proposed TCMPC method can produce less control effort in almost all the time of command tracking in all the three channels. The rates of change on control effort signals are limited for the proposed method, and so control signals are smoother, when compared to unconstrained MPC and PID.

To assess the performance of the TCMPC and the well-tuned PID, the first scenario was applied in the presence of wind gusts. At the specific time, when the yaw command was received by the controller, a fan was placed at a distance of 0.8 m from the test platform to generate a wind gust. The wind blew in three directions to the quadrotor, BFF coordinate, and its speed was $V_B = [1.3,\ 2.8,\ 0.2]\ m/s$. Fig 20 shows the performance of the controllers in the presence of wind. The initial values of both controllers were approximately the same and the wind blew from time = 9.5 sec to time = 18 sec, as illustrated in Fig 20. According to Fig 21, the wind gusts produced greater disturbance against the EMPC method controller than the constrained MPC. Hence, the EMPC controller signal efforts have more fluctuations than the TCMPC signal efforts during wind gusts.

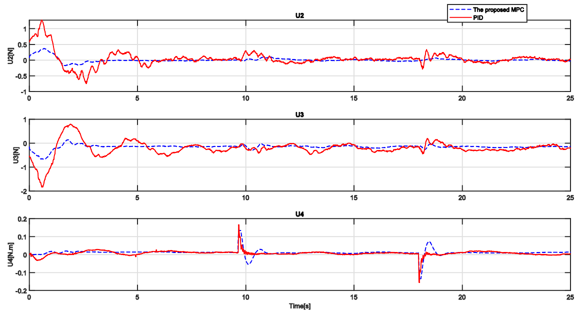


FIGURE 18. Control effort signals of the proposed MPC and PID to track set points.

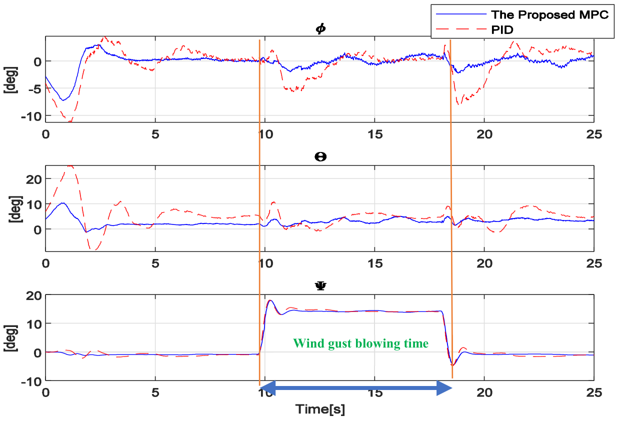


FIGURE 19. Attitude subsystem controlled by the TCMPC and PID in the presence of wind.

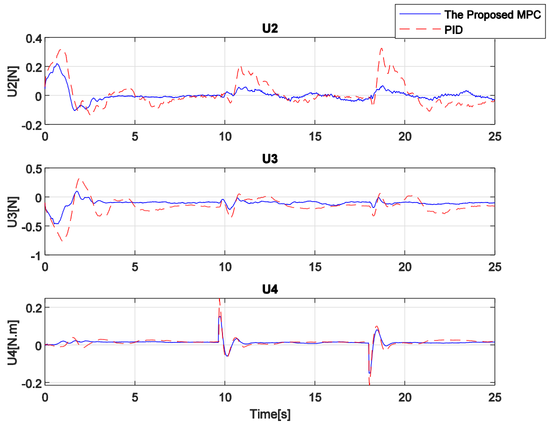


FIGURE 20. Control effort of the proposed MPC and PID in the presence of wind.

As far as the second scenario is concerned, the attitude tracking results under the two algorithms are provided in Figs. 21 and 22. Noticeably, under the influence of sensor measurement noises, unmodeled uncertainties of the system, and wind gust as external disturbance, the two controllers seem to have different degrees of tracking errors. As can be seen from Fig 22, the fluctuation of control signals tend to increase after adding the wind to system. Hence, the tracking performance of TCMPC can possibly outperform the EMPC. The value of each parameter of algorithms is taken from Table 4.

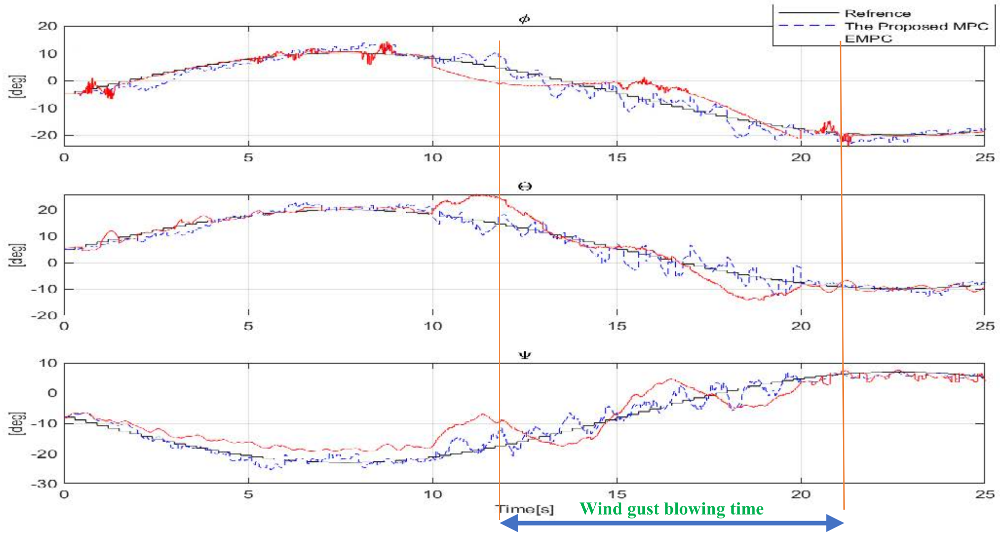


FIGURE 21. Attitude subsystem controlled by the proposed MPC and EMPC in the presence of wind gust.

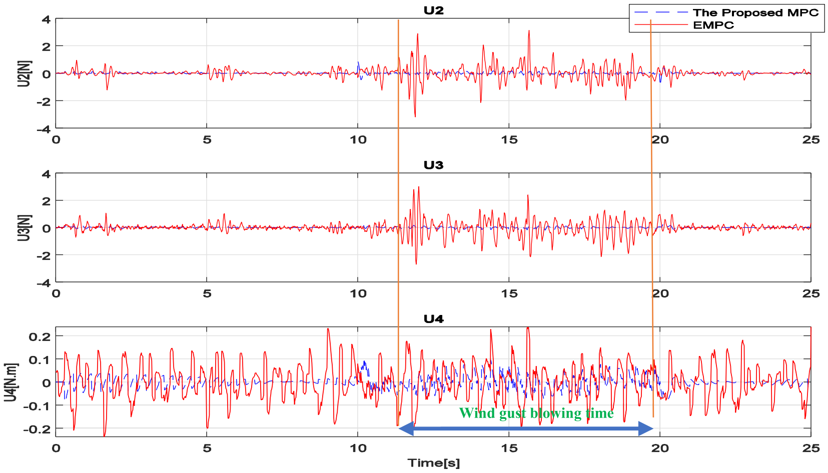


FIGURE 22. Control effort of the proposed MPC and EMPC in the presence of wind gusts.

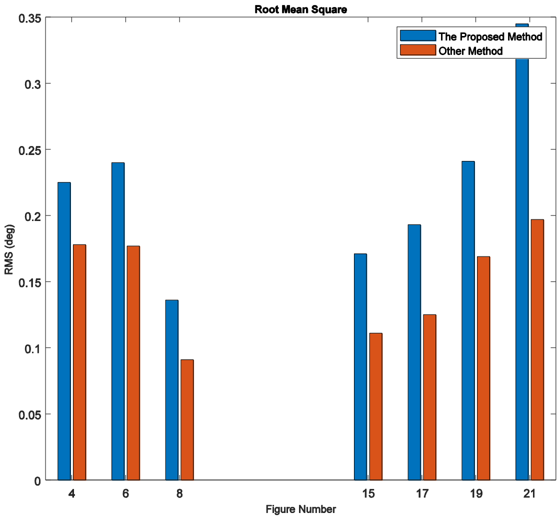


FIGURE 23. Comparisons of the tracking control performance.

According to Fig. 21, despite the wind disturbance, both methods converge to sinusoidal commands. To verify the tracking performance of the proposed attitude controller, we compared the RMS value of each method. Fig 23 exhibits that TCMPC has a smaller RMS value and achieves better performance in the second scenario. Based on the result of numerical simulation and practical tests, it can be concluded that the proposed controller improves the attitude tracking performance with/without presence of wind gust considered as external disturbance in comparison with AEMPC, EMPC, and well-tuned PID.

V. CONCLUSION

This paper proposed a Time-varying Constrained MPC (TCMPC) approach for the translation and attitude control of a real quadrotor in the presence of external disturbances. The proposed method can better update the nonlinear attitude subsystem at every sampling time around a new equilibrium point of the system for the provision of a time-varying model of the MPC designed with high precision and stability to occur. To better achieve the high tracking performance and robustness in the attitude control, the dynamic model parameters were extracted, and the equality of mass center with rotation center was considered as well. The proposed controller was tested and used to control a hand-made quadrotor with all restrictions and physical constraints of the real quadrotor taken into account. The comparison analysis and experimental results indicate that the proposed TCMPC outperforms the AEMPC, EMPC, and PID methods. The most important limitation of this method is that the parameters are derived from the experimental setting; hence, making this task is difficult and time-consuming for highly-complex multi-variable systems. Our future work will be a maneuver control of a practical quadrotor UAV by using the proposed method.

DATA AVAILABILITY STATEMENT

The data that support the findings of this study are available within the article.

CONFLICTS OF INTEREST

The authors declare that they have no conflict of interest.

REFERENCES

[1] C.-W. Chen, H.-A. Hung, P.-H. Yang, and T.-H. Cheng, "Visual servoing of a moving target by an unmanned aerial vehicle," *Sensors*, vol. 21, no. 17, p. 5708, Aug. 2021, doi: 10.3390/S21175708.

[2] A. Abdelmaboud, "The Internet of Drones: Requirements, taxonomy, recent advances, and challenges of research trends," *Sensors*, vol. 21, no. 17, p. 5718, Aug. 2021, doi: 10.3390/S21175718.

[3] D. Cafolla, M. Russo, and M. Ceccarelli, "Experimental validation of HeritageBot III, a robotic platform for cultural heritage," *J. Intell. Robot. Syst. Theory Appl.*, vol. 100, no. 1, pp. 223–237, Oct. 2020, doi: 10.1007/s10846-020-01180-6.

[4] M. Ceccarelli, D. Cafolla, M. Russo, and G. Carbone, "HeritageBot platform for service in Cultural Heritage frames," *Int. J. Adv. Robot. Syst.*, vol. 15, no. 4, Aug. 2018, Art. no. 1729881418790692, doi: 10.1177/1729881418790692.

[5] B. Lindqvist, S. S. Mansouri, A. A. Agha-Mohammadi, and G. Nikolakopoulos, "Nonlinear MPC for collision avoidance and control of UAVs with dynamic obstacles," *IEEE Robot. Autom. Lett.*, vol. 5, no. 4, pp. 6001–6008, Oct. 2020, doi: 10.1109/LRA.2020.3010730.

[6] T. Tomic, K. Schmid, P. Lutz, A. Domel, M. Kassecker, E. Mair, I. L. Grixa, F. Ruess, M. Suppa, and D. Burschka, "Toward a fully autonomous UAV: Research platform for indoor and outdoor urban search and rescue," *IEEE Robot. Autom. Mag.*, vol. 19, no. 3, pp. 46–56, Sep. 2012, doi: 10.1109/MRA.2012.2206473.

[7] B. H. Wang, D. B. Wang, Z. A. Ali, B. T. Ting, and H. Wang, "An overview of various kinds of wind effects on unmanned aerial vehicle," *Meas. Control*, vol. 52, nos. 7–8, pp. 731–739, May 2019, doi: 10.1177/0020294019847688.

[8] N. H. Motlagh, M. Bagaa, and T. Taleb, "Energy and delay aware task assignment mechanism for UAV-based IoT platform," *IEEE Internet Things J.*, vol. 6, no. 4, pp. 6523–6536, Aug. 2019, doi: 10.1109/JIOT.2019.2907873.

[9] N. H. Motlagh, T. Taleb, and O. Arouk, "Low-altitude unmanned aerial vehicles-based Internet of Things services: Comprehensive survey and future perspectives," *IEEE Internet Things J.*, vol. 3, no. 6, pp. 899–922, Dec. 2016, doi: 10.1109/JIOT.2016.2612119.

[10] N. H. Motlagh, M. Bagaa, and T. Taleb, "UAV selection for a UAV-based integrative IoT platform," in *Proc. IEEE Glob. Commun. Conf. GLOBECOM*, Dec. 2016, pp. 1–6, doi: 10.1109/GLOCOM.2016.7842359.

[11] B. B. Kocer, T. Tjahjowidodo, and G. G. L. Seet, "Model predictive UAV-tool interaction control enhanced by external forces," *Mechatronics*, vol. 58, pp. 47–57, Apr. 2019. Accessed: May 21, 2019. [Online]. Available: https://www.sciencedirect.com/science/article/abs/pii/S0957415819300042

[12] S. N. Ghazbi, Y. Aghli, M. Alimohammadi, and A. A. Akbari, "Quadrotors unmanned aerial vehicles: A review," *Int. J. Smart Sens. Intell. Syst.*, vol. 9, no. 1, pp. 309–333, Mar. 2016, doi: 10.21307/IJSSIS-2017-872.

[13] F. F. M. El-Sousy, K. A. Alattas, O. Mofid, S. Mobayen, J. H. Asad, P. Skruch, and W. Assawinchaichote, "Non-singular finite time tracking control approach based on disturbance observers for perturbed quadrotor unmanned aerial vehicles," *Sensors*, vol. 22, no. 7, p. 2785, 2022.

[14] M. Pouzesh and S. Mobayen, "Event-triggered fractional-order sliding mode control technique for stabilization of disturbed quadrotor unmanned aerial vehicles," *Aerosp. Sci. Technol.*, vol. 121, 2022, Art. no. 107337.

[15] A. Najafi, S. Mobayen, J. H. Asad, and A. Fekih, "Adaptive barrier fast terminal sliding mode actuator fault tolerant control approach for quadrotor UAVs," *Mathematics*, vol. 10, no. 16, p. 3009, 2022.

[16] M. Abdolhosseini, Y. M. Zhang, and C. A. Rabbath, "An efficient model predictive control scheme for an unmanned quadrotor helicopter," *J. Intell. Robot. Syst.*, vol. 70, nos. 1–4, pp. 27–38, Apr. 2013, doi: 10.1007/s10846-012-9724-3.

[17] A. Eskandarpour and I. Sharf, "A constrained error-based MPC for path following of quadrotor with stability analysis," *Nonlinear Dyn.*, vol. 99, no. 2, pp. 899–918, Apr. 2019, doi: 10.1007/S11071-019-04859-0.

[18] U. Eren, A. Prach, B. B. Koçer, S. V. Rakovic, E. Kayacan, and B. Açikmese, "Model predictive control in aerospace systems: Current state and opportunities," *J. Guid., Control, Dyn.*, vol. 40, no. 7, pp. 1541–1566, Apr. 2017, doi: 10.2514/1.G002507.

[19] I. K. Erunsal, A. Martinoli, and R. Ventura, "Decentralized nonlinear model predictive control for 3D formation of multirotor micro aerial vehicles with relative sensing and estimation," in *Proc. Int. Symp. Multi-Robot Multi-Agent Syst. (MRS)*, Aug. 2019, pp. 176–178, doi: 10.1109/MRS.2019.8901098.

[20] K. Alexis, G. Nikolakopoulos, and A. Tzes, "Switching model predictive attitude control for a quadrotor helicopter subject to atmospheric disturbances," *Control Eng. Pract.*, vol. 19, no. 10, pp. 1195–1207, Oct. 2011, doi: 10.1016/j.conengprac.2011.06.010.

[21] Z. Weihua and T. H. Go, "Robust decentralized formation flight control," *Int. J. Aerosp. Eng.*, vol. 2011, Sep. 2011, Art. no. 157590, doi: 10.1155/2011/157590.

[22] W. Zhao, H. G. Tiauw, and L. Eicher, "Formation flight control using model predictive approach," in *Proc. 47th AIAA Aerosp. Sci. Meeting Including New Horizons Forum Aerosp. Expo.*, 2009, p. 59, doi: 10.2514/6.2009-59.

[23] J. Shin and H. J. Kim, "Nonlinear model predictive formation flight," *IEEE Trans. Syst., Man, Cybern. A, Syst., Humans*, vol. 39, no. 5, pp. 1116–1125, Sep. 2009, doi: 10.1109/TSMCA.2009.2021935.

[24] X. Zhang, H. Duan, and Y. Yu, "Receding horizon control for multi-UAVs close formation control based on differential evolution," *Sci. China Inf. Sci.*, vol. 53, no. 2, pp. 223–235, Feb. 2010, doi: 10.1007/S11432-010-0036-6.

[25] Z. Cai, L. Wang, J. Zhao, K. Wu, and Y. Wang, "Virtual target guidance-based distributed model predictive control for formation control of multiple UAVs," *Chin. J. Aeronaut.*, vol. 33, no. 3, pp. 1037–1056, Mar. 2020, doi: 10.1016/J.CJA.2019.07.016.

[26] K. Lin, Y. Li, J. Sun, D. Zhou, and Q. Zhang, "Multi-sensor fusion for body sensor network in medical human–robot interaction scenario," *Inf. Fusion*, vol. 57, pp. 15–26, May 2020, doi: 10.1016/j.inffus.2019.11.001.

[27] V. K. Sivakumar and P. B. Sujit, "MPC-based multi-UAV path planning for convoy protection in 3D," in *Proc. IEEE Int. Conf. Autom. Sci. Eng.*, Aug. 2021, pp. 1554–1559, doi: 10.1109/CASE49439.2021.9551557.

[28] M. Su, H. Liu, J. Hu, C. Zhao, X. Hou, Q. Pan, and C. Jia, "Path planning based on improved MPC for fixed wing UAVs with collision avoidance," in *Advances in Guidance, Navigation and Control* (Lecture Notes in Electrical Engineering), vol. 644. 2022, pp. 2287–2296, doi: 10.1007/978-981-15-8155-7_192.

[29] H. Chen and X. Zhang, "Path planning for intelligent vehicle collision avoidance of dynamic pedestrian using Att-LSTM, MSFM, and MPC at unsignalized crosswalk," *IEEE Trans. Ind. Electron.*, vol. 69, no. 4, pp. 4285–4295, Apr. 2022, doi: 10.1109/TIE.2021.3073301.

[30] T. Stastny, A. Dash, and R. Siegwart, "Nonlinear MPC for fixed-wing UAV trajectory tracking: Implementation and flight experiments," *AIAA Guid. Navig. Control Conf.*, 2017, p. 1512, doi: 10.2514/6.2017-1512.

[31] T. Manzoor, Z. Sun, Y. Xia, and D. Ma, "MPC based compound flight control strategy for a ducted fan aircraft," *Aerosp. Sci. Technol.*, vol. 107, Dec. 2020, Art. no. 106264, doi: 10.1016/J.AST.2020.106264.

[32] K. M. Abughalieh and S. G. Alawneh, "A survey of parallel implementations for model predictive control," *IEEE Access*, vol. 7, pp. 34348–34360, 2019, doi: 10.1109/ACCESS.2019.2904240.

[33] M. Schwenzer, M. Ay, T. Bergs, and D. Abel, "Review on model predictive control: An engineering perspective," *Int. J. Adv. Manuf. Technol.*, vol. 117, no. 5, pp. 1327–1349, Aug. 2021, doi: 10.1007/S00170-021-07682-3.

[34] L. Jiayu, X. Zongchou, Y. Jiahao, and E. Gartlib, "Analysis of an automatic control system based on linear controllers and MPC controller," in *Proc. Int. Conf. Ind. Eng. Appl. Manuf. (ICIEAM)*, May 2020, pp. 1–5, doi: 10.1109/ICIEAM48468.2020.9111874.

[35] G. Torrente, E. Kaufmann, P. Fohn, and D. Scaramuzza, "Data-driven MPC for quadrotors," *IEEE Robot. Autom. Lett.*, vol. 6, no. 2, pp. 3769–3776, Apr. 2021, doi: 10.1109/LRA.2021.3061307.

[36] K. Y. Chee, T. Z. Jiahao, and M. A. Hsieh, "KNODE-MPC: A knowledge-based data-driven predictive control framework for aerial robots," *IEEE Robot. Autom. Lett.*, vol. 7, no. 2, pp. 2819–2826, Apr. 2022, doi: 10.1109/LRA.2022.3144787.

[37] D. Hanover, P. Foehn, S. Sun, E. Kaufmann, and D. Scaramuzza, "Performance, precision, and payloads: Adaptive nonlinear MPC for quadrotors," *IEEE Robot. Autom. Lett.*, vol. 7, no. 2, pp. 690–697, Apr. 2022, doi: 10.1109/LRA.2021.3131690.

[38] *Naza-M V2 Flight Control System—DJI*. Accessed: Mar. 25, 2022. [Online]. Available: https://www.dji.com/tw/naza-m-v2

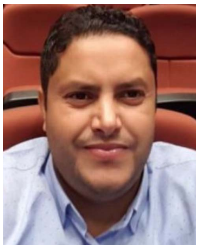
[39] K. Alexis, G. Nikolakopoulos, and A. Tzes, "Model predictive quadrotor control: Attitude, altitude and position experimental studies," *IET, Control Theory Appl.*, vol. 6, no. 12, pp. 1812–1827, Aug. 2012, doi: 10.1049/iet-cta.2011.0348.

[40] H. Pang, N. Liu, C. Hu, and Z. Xu, "A practical trajectory tracking control of autonomous vehicles using linear time-varying MPC method," *Proc. Inst. Mech. Eng. D, J. Automobile Eng.*, vol. 236, no. 4, pp. 709–723, Jun. 2021, doi: 10.1177/09544070211022904.

[41] T. Konyalioglu, S. Alnipak, and E. Altug, "Model predictive control of a hybrid VA V for parcel delivery applications," in *Proc. Aerial Robot. Syst. Phys. Interacting Environ.*, 2021, pp. 1–8, doi: 10.1109/AIRPHARO52252.2021.9571035.

MORTEZA ALIYARI received the B.Sc. degree in control engineering from the University of Zanjan, in 2015. He is currently pursuing the Ph.D. degree with the National Yunlin University of Science and Technology. He is also a Robotic Engineer and an Embedded System Developer with many hands-on experiences. He has done different projects in his interesting field, which is robotic, such as AGVs and UAVs, with different groups. His research interests include control theory, nonlinear systems, tracking of multi-agent, and teleoperation systems.

WING-KWONG WONG received the B.Sc. degree in computer science from the University of Toronto, Canada, in 1983, and the Ph.D. degree in computer science from The University of Texas at Austin, USA, in 1992. Since August 1993, he has been an Associate Professor with the Department of Electronic Engineering, National Yunlin University of Science and Technology (YunTech), Taiwan, where he is currently the Director of the Future Technology Research Center. His research interests include robotics, machine learning, the IoT, and E-learning.



YASSINE BOUTERAA (Member, IEEE) received the National Engineering degree in electrical engineering from the National Engineering School of Sfax, in June 2006, the Master of Science degree in control and computer science, in 2007, the Ph.D. degree in electrical and computer engineering from both the University of Orleans, France and the University of Sfax, in 2012, and the HDR (accreditation to supervise research) in electrical and computer engineering, from the University of Sfax in 2017. His research interests include robotics, embedded systems, and real time implementation. He is the author/coauthor of more than 50 scientific papers. He is a reviewer in some indexed journals and a TPC member of some international conferences.

SEPIDEH NAJAFINIA received the master's degree in digital electronic major from Shahid Beheshti University. She is currently pursuing the Ph.D. degree with the National Yunlin University of Science and Technology. She gained many skills in VLSI and FPGA design, during her master's degree. At the same time, she worked as an Embedded System Developer and she did different projects, such as designing a flight control circuit for a quad-copter, implementing a logger circuit, and implementing compression and tension load cell amplifier circuit.

AFEF FEKIH (Senior Member, IEEE) received the B.Sc., M.Sc., and Ph.D. degrees in electrical engineering from the National Engineering School of Tunis, Tunisia, in 1995, 1998, and 2002, respectively. She is currently a Full Professor with the Department of Electrical and Computer Engineering and the Chevron Professor/a BORSF Professor in engineering with the University of Louisiana at Lafayette. Her research interests include control theory and applications, including nonlinear and robust control, optimal control, fault tolerant control with applications to power systems, wind turbines, unmanned vehicles, and automotive engines. She has authored or coauthored more than 200 publications in international journals, chapters, and conference proceedings. She is a member of the Editorial Board of the IEEE Conference on Control Technology and Applications, the IEEE TC on Education, and IFAC TC on Power and Energy Systems.

SALEH MOBAYEN (Senior Member, IEEE) is currently an Associate Professor with the National Yunlin University of Science and Technology (YunTech), Taiwan, and collaborated with the Future Technology Research Center (FTRC). He has published several articles in the national and international journals. He is an associate editor of several international scientific journals and has acted as the symposium co-chair/the track co-chair in numerous IEEE flagship conferences. He has been a world's top 2% scientist with Stanford University, since 2019, and has been ranked among 1% top scientists in the world in the broad field of electronics and electrical engineering. He is also recognized in the list of Top Electronics and Electrical Engineering Scientists in Iran. His research interests include control theory, sliding mode control, robust tracking, nonholonomic robots, and chaotic systems.