Journal of Robotics and Control (JRC)
Volume 5, Issue 3, 2024
ISSN: 2715-5072, DOI: 10.18196/jrc.v5i3.21725

Fick's Law Algorithm Based-Nonlinear Model Predictive Control of Twin Rotor MIMO System

Omar Y. Ismael [1*], Mohanad N. Noaman [2], Ismael Kh. Abdullah [3]
[1, 2, 3] Department of Systems and Control Engineering, Ninevah University, Mosul, Iraq
Email: [1] omar.ismael@uoninevah.edu.iq, [2] mohanad.noaman@uoninevah.edu.iq, [3] ismael.abdullah@uoninevah.edu.iq
*Corresponding Author

***Abstract*—Nowadays, controlling a Twin Rotor MIMO System (TRMS) is a considerable challenge for engineers due to its high non-linear attributes. The controller's design goals are to achieve the appropriate pitch and yaw angles when there is cross-coupling between its main and tail rotors while minimizing both the angular position error and controller effort. Performance measures can be utilized to evaluate the performance of the controller including integral square error, total variation, and integral absolute control action. In this paper, a Nonlinear Model Predictive Control (NMPC) is proposed to control TRMS rotors, which refer to the vertical and horizontal planes. Fick's Law Algorithm (FLA) has been utilized to offline obtain the best selection for NMPC parameters. That includes best weighting matrices, shorter time steps, and shorter prediction horizons, by minimizing a novel penalty function called robust integral square error. FLA is used due to its flexibility, the ability to avoid suboptimal regions, and simplicity of implementation. The effectiveness of the proposed controller is examined using simulation-based tests conducted with MATLAB, which makes use of the CasADi Toolbox. In comparison to Cross Coupled PID (CC-PID) controller, the simulation results prove that FLA-based-NMPC has better performance and can track trajectories (step, square, and sine) even when there is $\pm 30\%$ in TRMS parameters perturbation. This work has come up with new contributions such as the new tuning strategy, extra state variable consideration, and a new FLA engineering application.**

***Keywords*—TRMS; Nonlinear Model Predictive Control (NMPC); Trajectory Tracking; Parameters Uncertainty; Robust Control; FLA; CasADi Toolbox; Cross Coupled PID.**

I. INTRODUCTION

Control engineers have faced never-before-seen challenges with multivariable systems because of their coupled dynamics, or interactions between output and input variables. It is customary to regulate each output/input pair independently, ignoring the consequences of cross-coupling for simplicity. On the other hand, if the coupling effects are ignored, the system dynamics may be approximated, reducing control performance [1][2]. A Twin Rotor Multi-Input Multi-Output System (TRMS) is a benchmark instance of a multivariable system. Feedback Instruments Ltd. created the TRMS laboratory test rig, which is used for control tests [3]. It is a helicopter prototype that still has cross-coupled modes and system nonlinearities, among other fundamental helicopter characteristics. Similar to helicopters, it is composed of both the main rotor and tail rotor, which are mounted on the beam and supported by a counterbalance. However, in contrast to the majority of flying helicopters, the rotors' angle of attack is fixed, and the motor speeds are used to regulate the aerodynamic forces.

Due to the simultaneous occurrence of highly nonlinear and coupled dynamics, as well as the reality that actuation torques are gained through the aerodynamics of rotor blades instead of straight (as it occurs for example in robot manipulators and mobile robots [4]-[7]), the TRMS control has proven difficult. For these reasons, control techniques are frequently tested on the TRMS control, for example, in [8]–[15]. Mainly because of the intrinsically coupled dynamics of TRMS, controllers designed for SISO systems cannot be easily transferred to them; instead, suitable controllers that consider the coupling effects of TRMS must be developed [16].

Many research efforts have been paid for trajectory tracking of TRMS. In [17]-[21], PID and FOPID controllers were proposed to control vertical and horizontal rotation TRMS. The proposed controllers have enhanced closed-loop performance. However, the key difficulty of using the PID controller for the MIMO system is parameter tuning. In [22][23], LQR has been used. In comparison with the PID controller, the LQR controller showed better performance in both transient and steady-state error with minimum control effort. Though, the LQR controller may not work with constraints. In [24]-[28], a sliding mode controller (SMC) has been employed to control TRMS, which is robust against model disturbances and uncertainties. Nevertheless, the major drawback of the SMC is the chattering effect resulting from discontinuous control. Backstepping control is proposed in [29] for TRMS control. It allows for the incorporation of adaptive control techniques, which can adapt to uncertain system parameters and disturbances, offering robustness against modeling errors and uncertainties. But, every step of the backstepping control process relies on the designer's capacity to supply a legitimate Lyapunov function upon which a control rule may be constructed. Adaptive fuzzy control, Type-2 fuzzy backstepping sliding mode controller, and fuzzy PID control have been applied in [30-33] to control the TRMS. Applied fuzzy logic to these controllers improved the stability analysis of the control system, and minimized chattering problems in SMC. However, the main drawback of fuzzy logic is the difficulty of tuning fuzzy controls' parameters such as the type of the membership functions and the number of rules. The TRMS in its nonlinear structure is stabilized by the development of a multistage feedback linearization-based controller [34][35]. The control rules are derived while maintaining the coupling effects. However,

because the authors use output feedback linearization, any disturbances that influence the TRMS will have a significant impact on how it responds.

To avoid the aforementioned problems, one approach is to utilize Model Predictive Control (MPC) which was developed in the late seventies and has since shown to be a successful control strategy for handling physical system constraints and multivariable interactions inside the optimization framework [36]-[39]. MPC can simultaneously control every output and take into consideration input-output interactions. MPC provides preview features similar to feed-forward control. The fundamental concept behind MPC is to predict the future behavior of the system using an explicit model and then to continuously compute a series of control signals using an optimization technique that minimizes the difference between the desired reference trajectory and the predicted system trajectory within a certain time horizon [40]-[42]. MPC can provide robustness, closed-loop stability, and superior tracking performance. The MPC algorithm's intricacy, which takes longer than that of other controllers, is a disadvantage. But this issue is resolved by the recent invention of microprocessors, which have increased computing capability. In the control community, MPC has grown in popularity due to the aforementioned advantages.

The finest TRMS control outcomes in recent years have been obtained using a variety of MPC adjustments. The authors of the studies in [43]-[48] discuss the use of linearized models in MPC design to provide trajectory tracking or point stabilization for TRMS in coupled form. These techniques reduce the control problem to a set of straightforward, quick, and reliable matrix algebra computations. However, their insufficient accuracy for altered operating conditions is the primary drawback of utilizing such linearized models for highly nonlinear plants close to an operational point. By utilizing nonlinear models in the control application; NMPC; the inadequate accurateness of MPC with linearized models is addressed, but can increase computing cost [49].

The quality of the TRMS model is important while developing NMPC. Due to significant fluctuations in the operating environment, parametric uncertainty affects the system dynamics in the majority of real-time control applications. An optimized-based NMPC can be employed as a control strategy for this kind of system, providing good results even when there are ongoing uncertainties. The appropriate selection of the NMPC parameters is crucial to the design's success, especially when implementing NMPC on quickly changing systems like TRMS, because these parameters significantly impact control effort, reaction time, stability, and computational load. The parameters of NMPC, including the control input weighting factor (R), tracking error weighting factor (Q), time step ΔT, and prediction horizon (N), all have a significant impact on the NMPC performance.

In this work, NMPC is proposed to control a nonlinear TRMS model, which guarantees a specific control performance versus model uncertainties (0, $\pm 30\%$). The tuning law for the selection of NMPC parameters is derived by minimizing a penalty function (i.e. Integral Squared Error (ISE)) penalty function using Fick's Law Algorithm (FLA), which is a physical-inspired population algorithm [50]. FLA has been employed in this work due to several features such as the ability to manage nonlinear optimization problems, flexibility, simplicity of implementation, and the ability to avoid suboptimal regions. For three reference trajectories (step, sine, and square), the controller's performance is compared to that of the Cross-Coupled PID (CC-PID) controller. To assess the effectiveness of the suggested strategy, a simulation is run using MATLAB R2023a and makes use of the CasADi Toolbox [51] with the Interior Point OPTimizer (IPOPT) solver [52]. The following is a list of this work's contributions:

- New tuning strategy for NMPC parameters that includes four parameters which are ΔT, N, Q, and R, with a novel penalty function called Robust Integral Square Error ($RISE$).
- Adding an extra state variable (ξ) that includes the coupling effect, which is advantageous.
- Suggest a new FLA engineering application to optimize the parameters of the NMPC.

The remainder of this paper is organized as follows: The NMPC theory and TRMS nonlinear modeling are presented in Section 2. Section 3 provides a detailed explanation of the control law's tuning strategy. Section 4 presents the simulation results, and Section 5 shows the study conclusions.

II. TRMS Modeling and Controller Theory

A. TRMS Kinematic Model

A benchmark instance of a multivariable aero-dynamical system that mimics a typical helicopter setup through flexible maneuvering is the TRMS. Modeling and control design are faced with significant difficulties due to the system characteristics, which include nonlinearities and substantial cross-coupling between the horizontal and vertical axes. Fig. 1 displays a schematic diagram of the TRMS. TRMS is equipped with two DC motor-driven propellers at either end of a beam that is articulated to connect it to the base. The beam's articulation enables rotation so that its ends can travel on spherical surfaces. Whereas the tail rotor permits the beam to move horizontally (yaw angle), the main rotor generates the lifting force that regulates the beam's location in the vertical plane (pitch angle). A stable equilibrium position is achieved by balancing the angular momentum with a counterweight that is anchored to the beam. The pitch angle is only marginally impacted by the tail motor's revolution, whereas yaw and pitch angles are significantly impacted by the main motor's rotation. The cross-coupling effect is explained by this. The TRMS's control inputs are the tail and main motors' DC voltages, (u_t) and (u_m), respectively. A variation in the voltage level causes the propeller's rotation speed to alter, which alters the beam's location accordingly. The TRMS's outputs are the beam locations, or the pitch (θ) and yaw ($\emptyset$) angles. These yaw and pitch angles can be counted by using encoders connected to the TRMS setup. Along the pitch and yaw axis, the corresponding angular velocity and angular acceleration are noted by $\dot{\theta}, \ddot{\theta}, \dot{\emptyset}, \ddot{\emptyset}$ respectively. Table I lists the TRMS's distinctive parameters

[3]. The following equations are utilized to create a mathematical model of the TRMS. The subscripts "m" and "t" in these equations stand for the main and tail rotors, respectively.

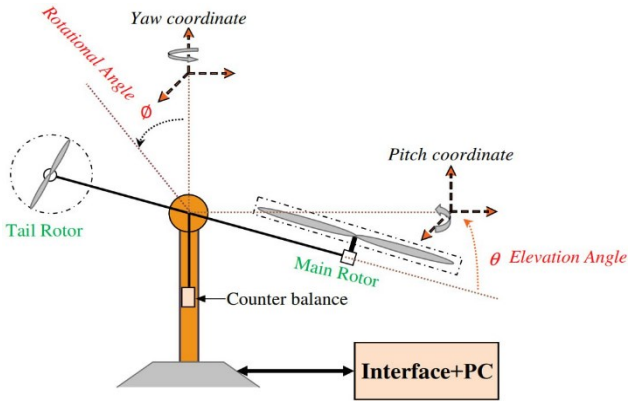

Fig. 1. Twin rotor MIMO system

The momentum equation for the vertical plane motion is specified as:

$$I_1\ddot{\theta} = M_1 - M_G - M_{B\theta} - M_{FG} \tag{1}$$

where, the following roughly represents the nonlinear main propeller thrust (non-linear static characteristics):

$$M_1 = a_1\tau_m^2 + b_1\tau_m \tag{2}$$

where τ_m is the momentum of the main motor. The Gyroscopic momentum is:

$$M_G = k_{gy}M_1\dot{\emptyset}\cos(\theta) \tag{3}$$

friction force momentum:

$$M_{B\theta} = B_{1\theta}\dot{\theta} - B_{2\theta}\sin(2\theta)\dot{\phi}^2 \tag{4}$$

and gravity momentum:

$$M_{FG} = M_g\sin(\emptyset) \tag{5}$$

The Momentum (τ_m) of the main rotor (main DC motor dynamics) has been approximated by the succeeding 1^{st} order transfer function:

$$\tau_m = \frac{k_1}{T_{11}s + T_{10}}u_m \tag{6}$$

The transfer function in (6) can be represented in the time domain as:

$$\dot{\tau}_m = \frac{k_1}{T_{11}}u_m - \frac{T_{10}}{T_{11}}\tau_m \tag{7}$$

Similarly, the momentum equation for the movement of the horizontal plane can be obtained as:

$$I_2\ddot{\emptyset} = M_2 - M_R - M_{B\emptyset} \tag{8}$$

where the following roughly represents the nonlinear tail propeller thrust (non-linear static characteristics):

$$M_2 = a_2\tau_t^2 + b_2\tau_t \tag{9}$$

TABLE I. TRMS PARAMETERS

Symbol	Description	Value
I_1	Moment of inertia of vertical rotor	$0.068\ kg\ m^2$
I_2	Moment of inertia of horizontal rotor	$0.028\ kg\ m^2$
a_1	Parameter of static characteristic	0.0135
b_1	Parameter of static characteristic	0.0924
a_2	Parameter of static characteristic	0.02
b_2	Parameter of static characteristic	0.09
M_g	Gravity momentum	$0.32\ Nm$
$B_{1\theta}$	Function parameter of friction momentum	$0.006\ N\ m\ s/rad$
$B_{2\theta}$	Function parameter of friction momentum	$0.001\ N\ m\ s/rad$
B_ϕ	Function parameter of friction momentum	$0.1\ N\ m\ s/rad$
k_{gy}	Parameter of gyroscopic momentum	$0.05\ s/rad$
k_1	Gain of main Motor	1.1
k_2	Gain of tail Motor	0.8
T_{11}	Denominator parameter of main motor	1.1
T_{10}	Denominator parameter of main motor	1
T_{21}	Denominator parameter of tail motor	1
T_{20}	Denominator parameter of tail motor	1
T_p	Parameter of cross-reaction momentum	2
T_0	Parameter of cross-reaction momentum	3.5
k_c	Gain of cross-reaction momentum	-0.2

where τ_t is the Momentum of the tail motor. The friction forces momentum is:

$$M_{B\emptyset} = B_\emptyset\dot{\emptyset} \tag{10}$$

and approximation of cross-reaction momentum

$$M_R = k_c\frac{(T_o s + 1)}{T_p s + 1}M_1 \tag{11}$$

The cross-feedback term M_R in time-domain can be expressed as:

$$M_R = \frac{1}{T_p}\left(1 - k_c\frac{1 + T_o}{T_p}\right)\xi + k_c\frac{1 + T_o}{T_p}M_1$$
$$\dot{\xi} = -\frac{1}{T_p}\xi + M_1 \tag{12}$$

where ξ is a variable used to represent the cross-reaction momentum in the time domain. Similar to (6), the Momentum (τ_t) of the tail-rotor (tail DC motor dynamics) can be expressed as:

$$\tau_t = \frac{k_2}{T_{21}s + T_{20}}u_t \tag{13}$$

In the time domain, the transfer function in (13) can be written as (14).

$$\dot{\tau}_t = \frac{k_2}{T_{21}} u_t - \frac{T_{20}}{T_{21}} \tau_t \tag{14}$$

The accuracy and realism of the TRMS kinematic model may be impacted by a number of assumptions and simplifications that are made in this section. Furthermore, ignoring some dynamical or nonlinear effects like structural flexibilities or aerodynamic variations could reduce the model's resilience and predictive power in real-world control applications. However, this research investigates the NMPC design for controlling TRMS under consideration of model uncertainty where the TRMS parameters are perturbated from their nominal values, which makes the control system robust.

To design the NMPC controller, it is to represent the mathematical model of the TRMS time domain state space form. The DC voltages of the main and tail motors are the TRMS control inputs and are expressed by:

$$u = [u_m \, u_t]^T \in \mathbb{R}^2 \tag{15}$$

where u is the control input vector. The designated state variables are the TRMS beam vertical position (θ), pitch rate ($\dot{\theta}$), the TRMS beam horizontal position (ϕ), yaw rate ($\dot{\phi}$), main rotor Momentum (τ_m), tail rotor Momentum (τ_t), and (ξ) as an extra state variable to account for the effect of cross-coupling. Consequently, the augmented state vector is written as:

$$x = \left[\theta \; \dot{\theta} \; \phi \; \dot{\phi} \; \tau_m \; \tau_t \; \xi\right]^T \in \mathbb{R}^7 \tag{16}$$

and can be expressed as:

$$\begin{aligned}
\dot{x}_1 &= \dot{\theta} \\
\dot{x}_2 = \ddot{\theta} &= \left(\frac{a_1}{I_1} - k_{gy}\frac{a_1}{I_1}\dot{\phi} cos\theta\right)\tau_m^2 + \frac{b_1}{I_1}\tau_m \\
&\quad -k_{gy}\frac{b_1}{I_1}\dot{\phi} cos\theta \tau_m - \frac{B_{1\theta}}{I_1}\dot{\theta} \\
&\quad + \frac{B_{2\theta}}{I_1}\dot{\phi}^2 sin2\theta - \frac{M_g}{I_1} sin\theta \\
\dot{x}_3 &= \dot{\phi} \\
\dot{x}_4 = \ddot{\phi} &= \frac{a_2}{I_2}\tau_t^2 + \frac{b_2}{I_2}\tau_t - \frac{B_\phi}{I_2}\dot{\phi} \\
&\quad -\frac{1}{T_p}\left(1 - k_c\frac{1+T_o}{T_p}\right)\xi \\
&\quad -k_c\frac{1+T_o}{T_p}(a_1\tau_m^2 + b_1\tau_m) \\
\dot{x}_5 &= \dot{\tau}_m = \frac{k_1}{T_{11}}u_m - \frac{T_{10}}{T_{11}}\tau_m \\
\dot{x}_6 &= \dot{\tau}_t = \frac{k_2}{T_{21}}u_t - \frac{T_{20}}{T_{21}}\tau_t \\
\dot{x}_7 &= \dot{\xi} = -\frac{1}{T_p}\xi + a_1\tau_m^2 + b_1\tau_m
\end{aligned} \tag{17}$$

The output vector of the TRMS is:

$$y = [\theta \; \emptyset] = [x_1 \; x_3] \in \mathbb{R}^2 \tag{18}$$

B. NMPC Theory

NMPC describes a class of control strategies that forecast the future trajectory of the system's states and outputs using an explicit nonlinear system model. This prediction ability enables the online solution of optimum control problems, wherein, potentially subject to constraints on the states, outputs, and inputs, the control input and the error between the reference trajectory and the anticipated output are minimized over a finite horizon. The process of optimization produces an optimum control sequence, whereby the system receives input from just the first element in the sequence. The horizon is shifted and the entire optimization process is carried out again at the following sample interval. This process, known as Receding Horizon Control (RHC), is primarily used because it compensates for disturbances that cannot be measured and modeling mistakes, leading to system outputs that differ from the nonlinear model's predictions. Consider a nonlinear system as follows:

$$\begin{aligned}
x(k+1) &= h\big(x(k), u(k)\big) \\
y(k) &= g\big(x(k)\big) \\
x(0) &= x_0
\end{aligned} \tag{19}$$

where $x(k) \in \mathbb{R}^7$, $u(k) \in \mathbb{R}^2$, and $y(k) \in \mathbb{R}^2$ are the state, inpu, and output vectors, respectively. If $K = \infty$, then for all $k \in \mathbb{N}_0$; if not, then for $k = 0, 1, \ldots, K-1$. We write $x(k, x_0)$ whenever we wish to highlight the reliance on the initial value.

A fundamental characteristic of the trajectories is: given a control $u \in U^N$, an initial value $x_0 \in X$, horizon length $N \geq 2$, and $k_1, k_2 \in \{0, \ldots, N-1\}$ with $k_1 \leq k_2$ as time instants the trajectory of the solution meets:

$$x(k_2, x_0) = x_{(\cdot + k_1)}(k_2 - k_1, x(k_1, x_0)) \tag{20}$$

The shifted control sequence in this case, $u(\cdot + k_1) \in U^{N-k_1}$, is provided by:

$$\begin{aligned}
u(\cdot + k_1)(k) &:= u(k + k_1), \\
k &\in \{0, \ldots, N - k_1 - 1\}
\end{aligned} \tag{21}$$

In other words, if the sequence u comprises N elements such as $u(0), u(1), \ldots, u(N-1)$, then the sequence $\tilde{u} = u(\cdot + k_1)$ comprises $N - k_1$ elements such as $\tilde{u}(0) = u(k_1), \tilde{u}(1) = u(k_1 + 1), \ldots, \tilde{u}(N - k_1 - 1) = u(N-1)$. With this definition, it is simple to show the identity (2) via induction with the help of (1) [53].

Within a set of constraints, the control aim is to track reference trajectories with the least amount of error and effort. The formulation of the cost function reflecting the control objective as follows [54][55]:

$$J_{N_i}\left(\tilde{X}_i(k), u_l(k)\right) = \sum_{i=k+1}^{k+N_i} \tilde{X}_i^T Q \tilde{X}_i \tag{23}$$

$$+\sum_{l=k}^{k+N_i-1} u_l^T R u_l$$

where $R \in \mathbb{R}^{2\times2}$ and $Q \in \mathbb{R}^{7\times7}$ are the diagonal positive definite weighting matrices for the state and control input variables, and N_i is the value of the prediction horizon at each ith step. The TRMS control input vector at time l is represented by $u_l u_e(k) = 0 \in \mathbb{R}^2$, and $\tilde{X}_i \in \mathbb{R}^7$, is different between the TRMS variables of predicted state of the system ($\hat{X}_i$) and its reference state (X_i^*) at each instant, as expressed:

$$\tilde{X}_i = \hat{X}_i - X_i^* \tag{24}$$

A realistic system must, in general, operate under some physical limitations that are imposed as output and system state constraints. The constraints for every input and output of a MIMO system are defined separately, as elucidated below: The following constraints apply to each control signal's amplitude in the TRMS:

$$\begin{aligned} u_{v_{min}} \le u_v(k) \le u_{v,max} \\ u_{t_{min}} \le u_t(k) \le u_{t,max} \end{aligned} \tag{25}$$

and the following constraints on the output:

$$\begin{aligned} \theta_{min} \le \theta(k) \le \theta_{max} \\ \emptyset_{min} \le \emptyset(k) \le \emptyset_{max} \end{aligned} \tag{26}$$

The physical constraints imposed by hardware limitations, actuator saturation, or safety considerations can be adequately captured by the given constraints on motor voltages and TRMS angles.

Equation (23) defines the objective function, whereas (25), and (26) provide possible constraints that utilize the quadratic programming algorithm to solve the optimization problem. Reducing the performance metric in (23) results in a series of control actions. The remaining samples in the series are ignored and just the first samples are used to create the incremental optimum control, by the receding horizon concept. A new process outputs measurement takes place at the following sampling moment $(k+1)$, and the entire process is repeated with the prediction horizon being moved ahead by one step and having the same length, N.

When the TRMS is controlled using the standard NMPC mode, the prediction horizon gain value will look like this:

$$N_{i+1} = N_i = N_{i-1}, \qquad i = 1, \dots, N_t \tag{24}$$

where N_t is the total number of time steps in the solution. Whereas, in this work, the prediction horizon gain is offline retrieved separately and independently from the FLA.

C. Stability of the Proposed Controller

To move the system formulation to the equilibrium position indicated by $u_e(k) = 0$, $x_e(k) = 0$, the control's goal is to compute an admissible command input, $u(k)$. According to this description, u_e represents the error between the system's control input and its reference vector, and x_e represents the errors between the system's state vector and its reference vector. It was shown that the usage of terminal equality constraints guaranteed the stability of predictive control [56]. According to [57], the NMPC's stability can be proven by applying the two presumptions that follow:

1) For an appropriate control value $u_r \in U$, there is an equilibrium point represented by a state vector such as, $x_r \in X \in \mathbb{R}^7$. Where X is the state space set for the state vector $\tilde{X}_i(k)$ and x_r is the reference state vector.
2) The function $f(x_r, u_r) = 0$ that $u_r \in U$ is obtained by applying the running cost function as $h: X \times U \to \mathbb{R}_0^+$.

Assumption 1, can be verified using system (17) as a guide. Additionally, the second supposition is met given that the cost function h has the quadratic form shown in (23). Since Algorithm I indicates that the data needed to train the network has been gathered from (25), and the controller is stable, then stability is guaranteed.

III. TUNING THE NMPC PARAMETERS

The selection of NMPC parameters such as N, R, Q, and ΔT significantly influences the response features. It is important to understand how these parameters affect the system response to select an appropriate value for them. A greater horizon is ensured by higher values of N, which improves control by producing more accurate forecasts. Nonetheless, these values must be selected in a way that makes the design workable for actual use. Increased computing load caused by large values of N prevents the control from being updated in system dynamics. The responses get slower as R rises because the controller is tighter. Because the controllers are overacting, the responses exhibit considerable oscillation at lower levels of R. Naturally, greater control effort is required to lower tracking error as Q increases. Therefore, in order to determine appropriate values of the aforementioned NMPC parameters, FLA is utilized to minimize a $RISE$ penalty function.

A. Tuning NMPC Parameters with RISE

To have robust NMPC parameters (ΔT, N, Q, and R) tuning, this work proposed a novel robust penalty function named Robust Integral Square Error ($RISE$) which is calculated from the following equation:

$$RISE = \sum_{i=1}^{c} (ISE_{+30\%} + ISE_{0\%} + ISE_{-30\%}) \tag{25}$$

where $ISE = \sum_{i=1}^{c} ({\theta_{e_i}}^2 + {\phi_{e_i}}^2)$ is the Integral Square Error of the TRMS output (y), $ISE_{+30\%}$ is the ISE of y when there are +30% increase in TRMS parameters in Table I, $ISE_{0\%}$ denotes that there is no change in the TRMS parameters while calculating the ISE of y, and $ISE_{-30\%}$ refers to -30% decrease in parameters of the TRMS. The NMPC parameters are then offline adjusted by using FLA to provide responses that are consistent with the contributions of this study.

B. Implementation of FLA

The inspiration for FLA follows Fick diffusion equations. These equations refer to the mathematical equations that

characterize the tendency of particles, in thermal motion, to move from a higher-density area to a lower-density area. Normal propagation, often known as Fick, is a diffusion that follows the Fick laws. This law stipulates that the flow of diffusion is proportional to the density gradient as follows:

$$F = -D\nabla C \tag{26}$$

where C is the diffusing particles concentration, F is the diffusion flux ($kg/[m^2\ s]$), and D is the diffusion constant (cm^2 per second). Fick's diffusion law reduces to the following for one-dimensional problem:

$$F = -D\frac{dC}{dx} \tag{27}$$

where D = effective diffusivity (m^2 of coal control surface area/[s]) diffusion flux, dC/dx = density gradient ([m^3 of gas]/ [m^3 of coal]/ [m length along gradient]), and C is in [m^3 of gas]/ [m^3 of coal].

The FLA algorithm simulates Fick's law to determine molecules' stable locations. Three phases of operators are provided including: i) Equilibrium Operator (EO), ii) Diffusion Operator (DO), and iii) Steady-State Operator (SSO). In the first phase, it is considered that there is a high difference in concentration between two areas at the beginning of the experiment which makes particles travel across different regions particles move from one region to another depending on concentration. In the EO phase, the concentration of two regions is About the same which makes particles reach equilibrium status. Finally, the particles change their locations in the region due to the best stable locations in the region which refers to the SSO phase. The FLA is mathematically provided as follows:

Step 1: FLA initialization, the process of optimization begins according to the candidate solutions set (X), as shown in equation (28). These solutions are produced at random, with each iteration yielding the best result that is almost optimal.

$$X = \begin{bmatrix} x_{1,1} & \cdots & x_{1,j} & \cdots & x_{1,D-1} & x_{1,D} \\ x_{2,1} & \cdots & x_{1,j} & \cdots & x_{2,D-1} & x_{2,D} \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ x_{N-1,1} & \cdots & x_{N-1,j} & \cdots & x_{N-1,D-1} & x_{N-1,D} \\ x_{N,1} & \cdots & x_{N,j} & \cdots & x_{N,D-1} & x_{N,D} \end{bmatrix} \tag{28}$$

where N is the no. of solutions (population size), D is the no. of variables (dimension of problem), and j refers to the jth decision variable.

Step 2: Clustering, two population equal groups are divided into N_1 and N_2.

Step 3: Transfer Function (TF), the exploitation to exploration transition and conversely is the cornerstone of any algorithm's success. a nonlinear TF is proposed for this purpose and defined as equation (29):

$$TF^t = sinh(t/T)^{C_1} \tag{29}$$

where t denotes to the no. of iterations T is the total number of iterations, and C_1 =0.5.

Step 4: Update particle position, in this context, three phases of the aforementioned operators are presented including DO, EO, and SSO. The transition between these three phases follows equation (30):

$$X_i^t = \begin{cases} DO & TF^t < 0.9 \\ EO & TF^t \le 1 \\ SSO & TF^t > 1 \end{cases} \tag{30}$$

Every population-based algorithm seeks to find the optimal answer for a particular optimization problem since obtaining the best answer requires more than one run. Nonetheless, a significant amount of obtaining the global optimal solution for a specific issue is made more likely by random solutions and optimization iterations [58, 59]. Exploration and exploitation are the two stages of the optimization process, nevertheless of the differences in metaheuristic algorithms used in population-based optimization strategies. In the exploitation phase, solution accuracy is improved, whereas, in the exploration phase, search agents of an algorithm cover a significant area of the search space to avoid locally optimal solution. Because it can strike a balance between both stages, FLA is regarded as a strong and effective optimization algorithm that ensures that both exploration and exploitation are carried out fairly. The idea of Fick's laws of diffusion serves as the foundation for this approach. The suggested FLA steps are shown in Algorithm 1 [50]. The range of the five diffusing particles concentration variables are as follows: $C_1 = \in \{0.5, 1\}$, $C_2 = \in \{0.5, 1, 1.5, 2, 2.5\}$, $C_3 = \in \{0.1, 0.2\}$, $C_4 = \in \{0.1, 0.2\}$, $C_5 = \in \{0.5, 1, 2\}$. The diffusion constant $(D) = 0.01$.

Algorithm 1: FLA

Initialization Phase;
Initialize parameters $(D, C1, C2, C3, C4, C5)$;
Initialize population $X_i(i = 1,2, \ldots . N)$ randomly;
Clustering: Divide population into two equal groups N_1, and N_2;
for $s = 1{:}\ 2$ do
Compute the fitness function for each molecule in group N_s;
Find the best molecule in each group and the global optimum;
end for
while $FES \le MAX_{FES}$ do
if $TF < 0.9$ then %Steady state operator (SSO)
for op=1: n_{op} do
Calculate Diffusion rate factor using $DRF_g^t = \exp\left(-\frac{J_{p,ss}^t}{TF^t}\right)$
Calculate Motion step factor using $MS_{p,g}^t = \exp\left(-\frac{FS_{ss}^t}{(FS_{p,g}^t + eps)}\right)$
Update individual position using $X_{p,g}^{t+1} = X_{ss}^t + Q_g^t \times (MS_{p,g}^t \times X_{ss}^t - X_{p,g}^t)$
end for
else if $(TF < rand)$ then %Equilibrium Operator (EO)
for op=1: n_{op} do
Calculate Diffusion rate factor using $DRF_{EO,g}^t = \exp\left(-\frac{J_{p,EO}^t}{TF^t}\right)$
Calculate group relative Quantity using $Q_{EO,g}^t = R_1^t \times DF_g^t \times DRF_{EO,g}^t$
Update individual position using $X_{p,g}^{t+1} = X_{EO,p}^t + Q_{EO,g}^t \times X_{p,g}^t + Q_{EO,g}^t \times (MS_{p,EO}^t \times X_{EO.g}^t - X_{p,g}^t)$
end for
else %Diffusion Operator (EO)
Calculate direction of flow using $DOF = \exp(-C_2(TF^t - r_1)), C_2 = 2$
Determine number of molecules that will travel to region using $NT_{ij} \approx N_i \times r_1 \times (C_4 - C_3) + N_i \times C_3$

25: Update individual position using $X_{p,i}^{t+1} = X_{EO,j}^{t} + DF_{p,i}^{t} \times DOF \times r_2(J_{i,j}^{t} \times X_{EO.j}^{t} - X_{p,i}^{t+1})$
26: Update other molecules in region i using $X_{p,i}^{t+1} = \begin{cases} X_{EO,i}^{t} & rand < .8 \\ X_{EO.i}^{t} + DOF \times (r_3 \times (U - L) + L) & rand < .9 \\ X_{p,i}^{t+1} & otherwise \end{cases}$
27: Update molecules in region j using $X_{p,i}^{t+1} = X_{EO,j}^{t} + DOF \times (r_4 \times (U - L) + L)$
28: Update $FES \leftarrow FES + NP$;
29: end if
30: end while
31: Return best solution;

IV. SIMULATION AND RESULTS

The proposed NMPC scheme of TRMS is validated by simulation using MATLAB R2023b that utilizes CasADi Toolbox with IPOPT solver. The system model (17) and the optimum control problem are specified in the CasADi toolbox, where the NMPC is implemented. The IPOPT solver for nonlinear programming problems and the multiple shooting technique are used to solve the optimal control problem and achieve state integration. The maximum number of iterations is set at 2000, while the convergence criteria of IPOPT is maintained at 10^{-8}. The Runge-Kutta fourth (RK4) order method is used to integrate the states. All simulations are performed on a personal computer with a 2.6 GHz CPU, a core i7-10750H processor, and 16 GB of RAM. The TRMS states initial conditions are taken to be zero in this work.

The proposed NMPC is compared with the CC-PID controller in order to better grasp the benefits of the proposed control scheme. Through simulations, the TRMS's capability to track different reference signals is verified. The selected reference signals are Step, Sinusoidal, and Square. Both controllers' parameters are offline tuned using FLA with $RISE$ Penalty function. The FLA speed at finding an optimal solution and the likelihood of finding the global optimal are influenced by factors like the highest number of iterations and the search agents' number. The choice of these parameters' values depends on the application. Because TRMS is a nonlinear MIMO system with a high likelihood of multi-local optimal, choosing appropriate FLA parameters is more challenging. Given that the suggested work is a new implementation of the newly released FLA, experience and trial and error were used to determine the FLA parameters. As a result, 50, 15, and 7 are selected as the maximum iteration, search agent count, and number of variables (D) respectively, in the FLA parameters. For every controller, the FLA is executed several times for only the square reference signal in order to determine the minimal $RISE$ cost function with the best system performance and meet controller design requirements. Table II displays the NMPC and CC-PID controllers' tuned parameters.

The control design can be significantly impacted by uncertainties in parameters value. To guarantee that the controller can function correctly throughout the whole range of possible parameter values, the design should often take the "worst situation" into account. In this paper, some parameters of the system are varied from their nominal values at the running time in order to examine the impact of the proposed controller's parameter uncertainty. The parameters perturbation is taken as $+30\%$ and -30% from their nominal values. This work considers the tracking performance of the proposed NMPC in the presence and absence parameters perturbation for both subsystems that are horizontal and vertical. Three performance measures are utilized including:

- Integral Square Error (ISE): the square of the error is integrated over time using ISE. Due to the larger square of a significant error, ISE will penalize large errors more than smaller ones. Large errors are often eliminated fast by control systems designed to minimize ISE, while tiny errors may remain for a long time. This frequently results in quick reactions but significant, low-amplitude oscillation.
- Total Variation (TV): TV is used to show how smoothly the pitch and yaw angles are going.
- Integral absolute control action ($\|u\|_2$): the control effort is integrated over time that indicate the controller energy consumption.

TABLE II. CONTROLLER PARAMETERS

Controller	Parameter	Value
NMPC	Q	$\begin{bmatrix} 9e5 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 7e6 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 3.4 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 5e5 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 6e6 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 7.7 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 3e5 \end{bmatrix}$
	R	$\begin{bmatrix} 0.0015 & 0 \\ 0 & 1.5e-4 \end{bmatrix}$
	N	7
	ΔT	0.137
	$u_m(min)$, $u_m(min)$	$-2.5\ v$
	$u_t(max)$, $u_t(max)$	$2.5\ v$
CC-PID	K_{Pi} $(i = 1, \dots 4)$	[94, 98,90,91]
	K_{Ii} $(i = 1, \dots 4)$	[94,88,94,95]
	K_{Di} $(i = 1, \dots 4)$	[97,88,100,97]

A. Step Signal Tracking

The TRMS model's horizontal and vertical subsystems receive two desired step inputs, $r_1(t) = 0.5\ (rad)$, and $r_2(t) = 0.5\ (rad)$, concurrently for pitch and yaw angles tracking for $10\ sec$ period. The initial values of these angles are set to $[0, 0]$.

1) Tracking Performance

The proposed NMPC's tracking performance compared with the CC-PID controller is illustrated in Fig. 2 and Fig. 3 for both subsystems that are horizontal and vertical and with and without TRMS parameters perturbation. It may be observed that while the CC-PID controller is capable of managing parametric uncertainty, it is unable to manage the overshoot for all cases (more than 15%). Also, the settling time for the CC-PID controller is more than twice NMPC. On the other hand, the proposed NMPC can track the reference trajectory with a settling time of less than $3\ sec$ and no

overshoot without experiencing any steady-state error even when the parametric uncertainty is present. To evaluate the proposed NMPC superiority over the CC-PID controller for both pitch and yaw angles, the performance indices in Table III are utilized. As compared to the CC-PID controller, the proposed NMPC offers a lower ISE and TV for pitch and yaw angles.

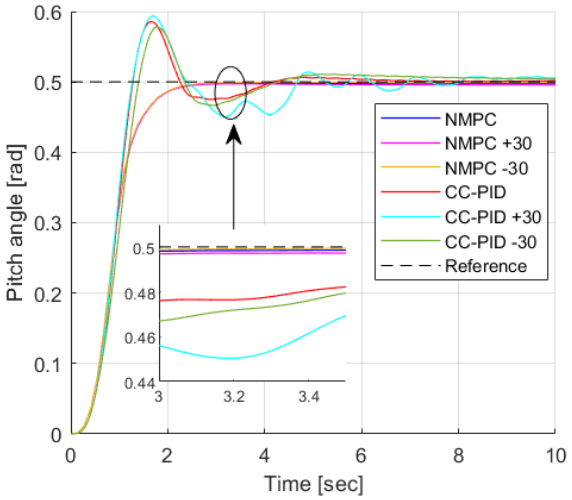

Fig. 2. Step response for pitch angle

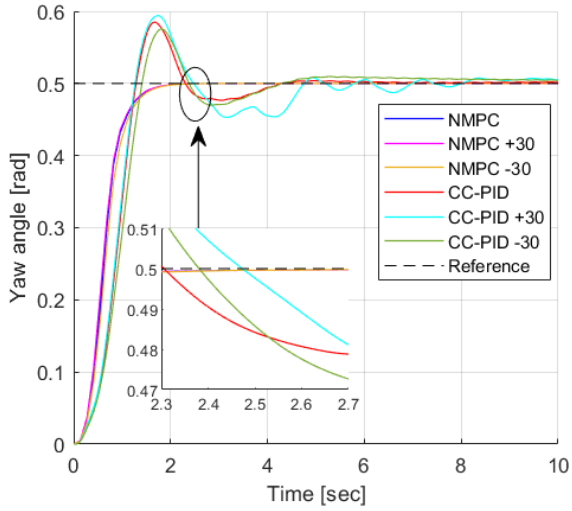

Fig. 3. Step response for yaw angle

TABLE III. PERFORMANCE ANALYSIS FOR STEP INPUT SIGNAL

Controller		*ISE*	*TV*	$\|\|u\|\|^2$
Pitch angle	NMPC	1.39	0.49	104.03
	NMPC (+30)	1.39	0.5	103.80
	NMPC (-30)	1.38	0.49	104.135
	CC-PID	18.19	0.76	3970
	CC-PID (+30)	18.34	1.07	4140
	CC-PID (-30)	19.00	0.77	3910
Yaw angle	NMPC	1.03	0.49	111.15
	NMPC (+30)	1.00	0.49	103.08
	NMPC (-30)	1.07	0.49	115.23
	CC-PID	18.19	0.75	4910
	CC-PID (+30)	17.32	0.99	4960
	CC-PID (-30)	18.22	0.77	4950

2) *Control Signal Analysis*

The control signal analysis in Fig. 4 and Fig. 5, and Table II also shows that, in comparison to the CC-PID controller, the proposed NMPC offers a lower $\|u\|_2$ value for main and tail control inputs. Therefore, under parametric uncertainty, the suggested NMPC is more energy-efficient than the CC-PID controller.

3) *NMPC Computational Efficiency*

The computed average computing time for the suggested NMPC is $0.017\ sec$.

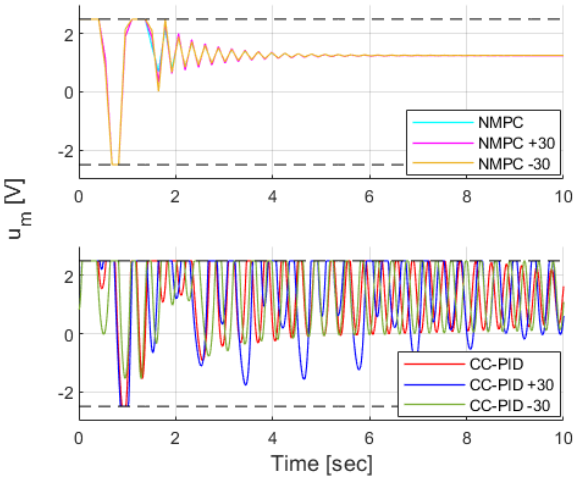

Fig. 4. Step response main rotor control signal and limits

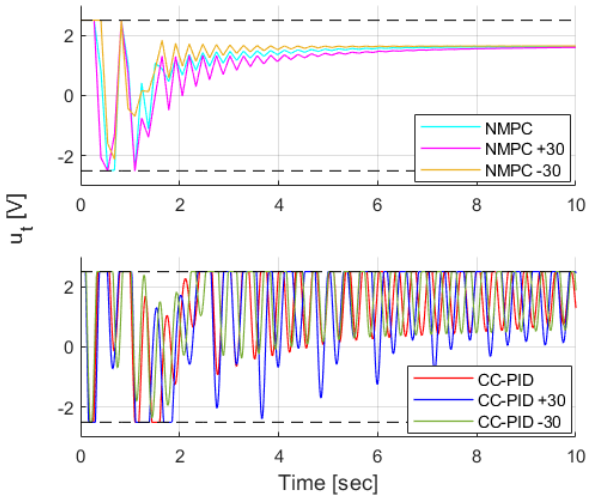

Fig. 5. Step response tail rotor control signal and limits

B. *Sine Signal Tracking*

A sine wave with amplitude of $0.2\ rad$ and frequency of $1/30\ Hz$ is used as the reference for the pitch angle, while a sine wave with amplitude of $0.3\ rad$ and frequency of $1/30\ Hz$ is used for the yaw angle, each of period of $45\ sec$.

1) *Tracking Performance*

Fig. 6 and Fig. 7, respectively, illustrate the comparative tracking performance of the NMPC controller versus the CCPID controller for the TRMS system with and without parameters perturbation for the vertical and horizontal subsystems. It is noted that the $+30\%$ parameters perturbation has a considerable impact on the sinusoidal response for pitch and yaw angles for the CC-PID controller with unacceptable large overshoot, steady state error, and settling time while the NMPC successfully handles this perturbation. For the zero and -30% parameters

perturbation, NMPC maintains control while offering resilience against parametric uncertainty while the CC-PID consists of continuous deviation (steady sate error) from the reference signal. Additionally, a comparison analysis is conducted using a few performance metrics for NMPC and the CC-PID controller for pitch and yaw angles. The results are shown in Table IV. Comparing the proposed control approach to the CC-PID controller, it is noted that the former yields lower ISE, and TV for pitch and yaw angles.

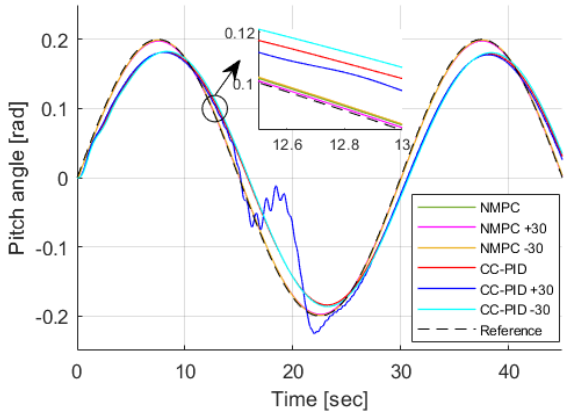

Fig. 6. Sine response for pitch angle

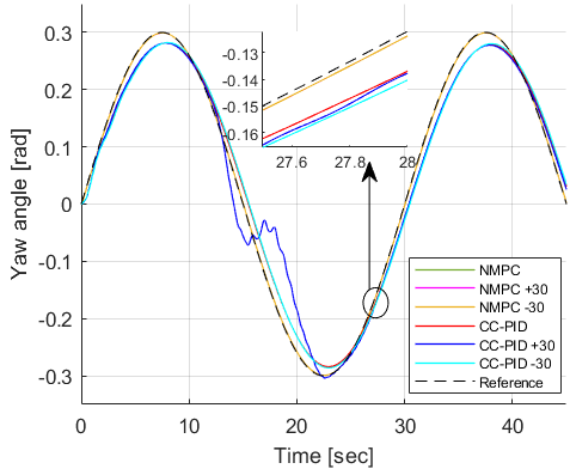

Fig. 7. Sine response for yaw angle

TABLE IV. PERFORMANCE ANALYSIS FOR SINE INPUT SIGNAL

Controller		*ISE*	*TV*	$\|u\|^2$
Pitch angle	NMPC	0.22	1.19	137.65
	NMPC (+30)	0.22	1.18	136.97
	NMPC (-30)	0.23	1.18	138.13
	CC-PID	349.95	1.25	2210
	CC-PID (+30)	336.86	1.68	5840
	CC-PID (-30)	347.57	1.24	2290
Yaw angle	NMPC	0.47	1.18	168.57
	NMPC (+30)	0.47	1.18	159.31
	NMPC (-30)	0.47	1.19	177.29
	CC-PID	810.7	1.92	2770
	CC-PID (+30)	789.15	2.06	5910
	CC-PID (-30)	807.23	1.91	2720

2) *Control Signal Analysis*

Table IV, and Fig. 8 and Fig. 9 show that, in comparison to the CC-PID controller, the proposed NMPC offers a lower $\|u\|_2$ value for main and tail control inputs. Therefore, in the presence and absence of parameters perturbation, the proposed NMPC has greater energy efficiency than the CC-PID algorithm.

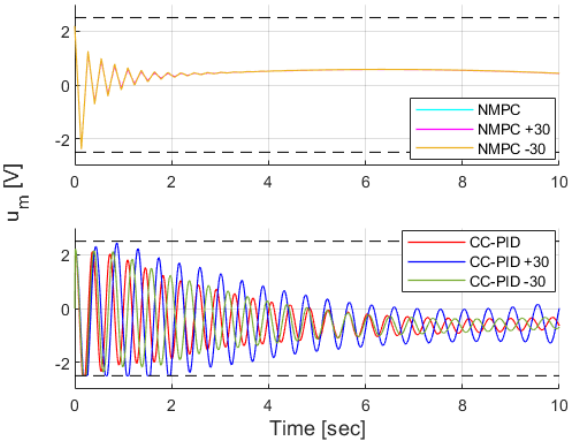

Fig. 8. Sine response main rotor control signal and limits

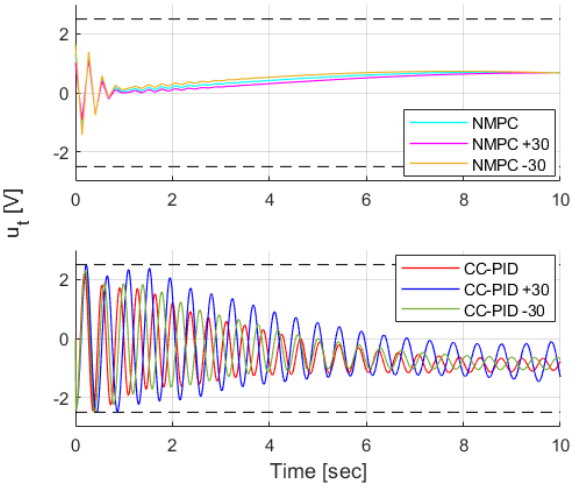

Fig. 9. Sine response tail rotor control signal and limits

3) *NMPC Computational Efficiency*

For the proposed NMPC, the computed average computation time is $0.014\ sec$.

C. *Square Signal Tracking*

A Square wave with amplitudes of $0.2\ rad$ and frequencies of $1/30\ Hz$ are used to reference pitch angles, and square waves with amplitudes of $0.3\ rad$ and frequencies of $1/30$ Hz are used to reference yaw angles, each of period $45\ sec$. One may observe that low frequencies are used in the selection of the excitation signals (square and sine). This makes sense given the system's dynamics [60].

1) *Tracking Performance*

The tracking performance of the NMPC controller and the CCPID controller for the TRMS system in the presence and absence of parameter perturbation for both subsystems that are horizontal and vertical are illustrated in Fig. 10 and Fig. 11, respectively. The selection of the parameters of MPC allows the controller to track the reference trajectory with a settling time of less than $3\ sec$ and no overshoot without encountering any steady-state error. Conversely, though, it can be seen that the CC-PID controller cannot handle parametric uncertainty and cannot control overshoot (greater

than 25%). The CC-PID controller's settling time is much longer than that of NMPC.

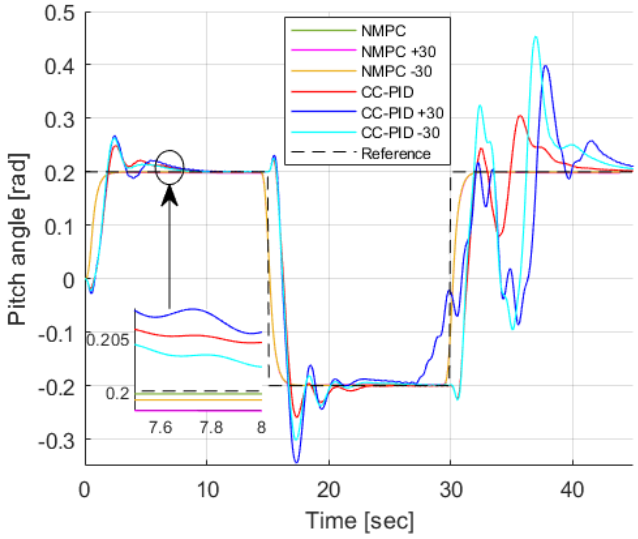

Fig. 10. Square response for pitch angle

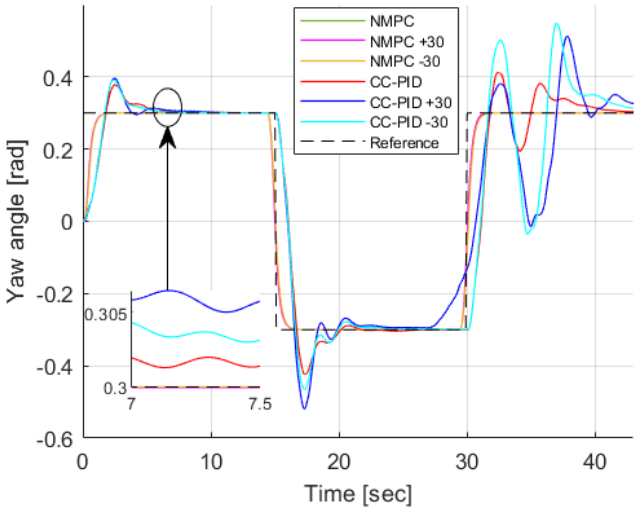

Fig. 11. Square response for yaw angle

2) *Control Signal Analysis*

Table V, Fig. 12 and Fig. 13 demonstrate that the proposed NMPC delivers a lower $\|u\|_2$ value for the main and tail control inputs when compared to the CC-PID controller. Consequently, the suggested NMPC control algorithm uses less energy than the CC-PID method, both with and without a parametric uncertainty.

TABLE V. PERFORMANCE ANALYSIS FOR SQUARE INPUT SIGNAL

Controller		*ISE*	*TV*	$\|u\|^2$
Pitch angle	NMPC	3.28	1.07	332.31
	NMPC (+30)	3.26	1.07	331.50
	NMPC (-30)	3.29	1.07	333.38
	CC-PID	65.02	2.63	5680
	CC-PID (+30)	73.92	4.29	7670
	CC-PID (-30)	82.06	3.82	6300
Yaw angle	NMPC	6.21	1.07	389.12
	NMPC (+30)	6.05	1.07	372.64
	NMPC (-30)	6.44	1.07	414.80
	CC-PID	98.03	3.43	6480
	CC-PID (+30)	107.88	4.85	7870
	CC-PID (-30)	121.05	4.70	6800

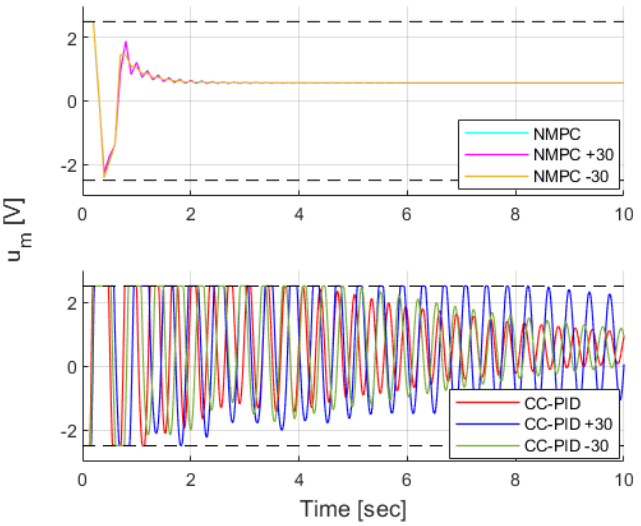

Fig. 12. Square response main rotor control signal and limits

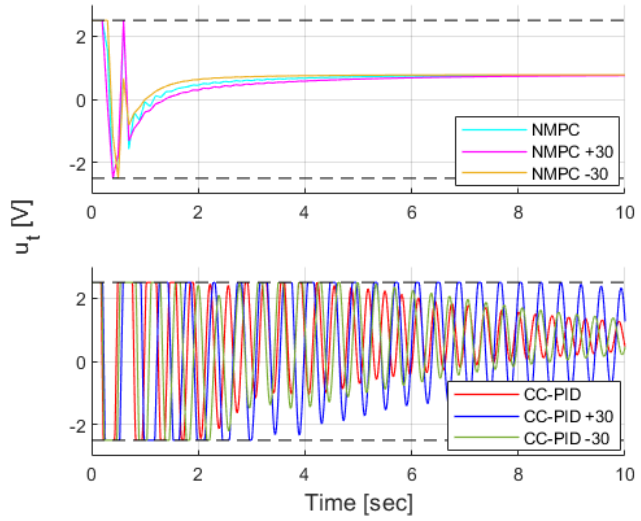

Fig. 13. Square response tail rotor control signal and limits

3) *NMPC Computational Efficiency*

A $0.013\ sec$ is the computed average computation time for the proposed NMPC.

V. CONCLUSION

In this paper, a nonlinear model predictive controller (NMPC) has been proposed to control the horizontal and vertical plane of the highly nonlinear TRMS system. A comparative study with the CC-PID controller was performed to show the superiority of the proposed controller. Both controllers' parameters were tuned by utilizing a Fick's Law of diffusion inspired algorithm. Using the CasADi Toolbox with IPOPT solver, a simulation was run in MATLAB. Three reference trajectories (step, sine, and square) were followed by both controllers. Both with and without TRMS parameters perturbation. Three performance criteria, including ISE, TV and $\|u\|_2$, were used to assess the effectiveness of the NMPC. The NMPC has shown respectable real-time performance based on the previously described metrics during numerical simulations of TRMS setpoint tracking compared to the CC-PID controller. The proposed NMPC offered a lower ISE, TV, and $\|u\|_2$ for pitch and yaw angles for all the three reference signals in the presence and absence of TRMS

parameters perturbation. Therefore, under TRMS parametric uncertainty, the suggested NMPC was more energy-efficient than the CC-PID controller. As part of this research's future work, the proposed NMPC can be practically implemented on an actual TRMS test rig. The following suggestions may be considered for the practical implementation in the future:

- To reduce the hardware's time delay, a U2D2 interface can be used. In this sense, the TRMS receives controlling information straight from MATLAB; the laptop and this interface connect via the USB port.
- The TRMS may need to run in practical slower than the simulation in order to reduce the dynamic impacts that the kinematic model does not consider. To quantify model disturbances, the proposed controller may be combined with observer-based controllers, as outlined in references [38], by estimating certain features of the environment.

REFERENCES

[1] R. Raghavan and S. Thomas, "Practically Implementable Model Predictive Controller for a Twin Rotor Multi-Input Multi-Output System," *Journal of Control, Automation and Electrical Systems*, vol. 28, no. 3, pp. 358–370, Mar. 2017, doi: 10.1007/s40313-017-0311-5.

[2] T. Roy and R. Kumar Barai, "LFT modelling and H∞ control of a non-linear MIMO system with cross coupled dynamics," *International Journal of Automation and Control*, vol. 7, no. 1-2, pp. 105-133, Jul. 2014, doi: 10.1504/IJAAC.2013.055099.

[3] M. Twin Roto. System Control Experiments 33-949S Feedback Instruments Ltd. *East susex, UK*, 2006.

[4] M. Qasim and O. Y. Ismael, "Shared Control of a Robot Arm Using BCI and Computer Vision," *Journal Européen des Systèmes Automatisés*, vol. 55, no. 1, pp. 139–146, Feb. 2022, doi: 10.18280/jesa.550115.

[5] M. N. Alghanim, M. Qasim, K. P. Valavanis, M. J. Rutherford, and M. Stefanovic, "Comparison of Controller Performance for UGV-Landing Platform Self-Leveling," *2020 28th Mediterranean Conference on Control and Automation (MED)*, pp. 471-478, 2020, doi: 10.1109/med48518.2020.9182837.

[6] M. N. Noaman, M. Qasim, and O. Y. Ismael, "Landmarks exploration algorithm for mobile robot indoor localization using VISION sensor," *Journal of Engineering Science & Technology*, vol. 16, no. 4, pp. 3165-3184, 2021.

[7] M. N. Alghanim, M. Qasim, K. P. Valavanis, M. J. Rutherford, and M. Stefanovic, "Passivity-Based Adaptive Controller for Dynamic Self-Leveling of a Custom-Built Landing Platform on Top of a UGV," *2020 28th Mediterranean Conference on Control and Automation (MED)*, 2020, doi: 10.1109/med48518.2020.9182807.

[8] S. Zeghlache, L. Benyettou, A. Djerioui, and M. Z. Ghellab, "Twin Rotor MIMO System Experimental Validation of Robust Adaptive Fuzzy Control Against Wind Effects," *IEEE Systems Journal*, vol. 16, no. 1, pp. 409–419, Mar. 2022, doi: 10.1109/jsyst.2020.3034993.

[9] M. Z. Ghellab, S. Zeghlache, A. Djerioui, and L. Benyettou, "Experimental validation of adaptive RBFNN global fast dynamic terminal sliding mode control for twin rotor MIMO system against wind effects," *Measurement*, vol. 168, p. 108472, Jan. 2021, doi: 10.1016/j.measurement.2020.108472.

[10] L. Dutta and D. K. Das, "A New Adaptive Explicit Nonlinear Model Predictive Control Design for a Nonlinear MIMO System: An Application to Twin Rotor MIMO System," *International Journal of Control, Automation and Systems*, vol. 19, no. 7, pp. 2406–2419, Mar. 2021, doi: 10.1007/s12555-020-0272-5.

[11] D. M. Ezekiel, R. Samikannu, and M. Oduetse, "Modelling of the Twin Rotor MIMO System (TRMS) Using the First Principles Approach," *2020 International Conference on Computer Communication and Informatics (ICCCI)*, pp. 1-7, 2020, doi: 10.1109/iccci48352.2020.9104156.

[12] R. Roman, M. Radac, and R. Precup, "Multi-input–multi-output system experimental validation of model-free control and virtual reference feedback tuning techniques," *IET Control Theory & Applications*, vol. 10, no. 12, pp. 1395–1403, Aug. 2016, doi: 10.1049/iet-cta.2016.0028.

[13] L. Belmonte, R. Morales, A. Fernández-Caballero, and J. Somolinos, "Robust Decentralized Nonlinear Control for a Twin Rotor MIMO System," *Sensors*, vol. 16, no. 8, p. 1160, Jul. 2016, doi: 10.3390/s16081160.

[14] D. E. Konovalov, S. A. Vrazhevsky, I. B. Furtat, and A. S. Kremlev, "Modified Backstepping Algorithm with Disturbances Compensation for Nonlinear MIMO Systems," *IFAC-PapersOnLine*, vol. 53, no. 2, pp. 6012–6018, 2020, doi: 10.1016/j.ifacol.2020.12.1665.

[15] V. K. Singh, S. Kamal, and S. Ghosh, "Prescribed-time constrained feedback control for an uncertain twin rotor helicopter," *Aerospace Science and Technology*, vol. 140, p. 108483, Sep. 2023, doi: 10.1016/j.ast.2023.108483.

[16] S. Zacher, "Multivariable Control and Management," *Closed Loop Control and Management*, pp. 317–353, 2022, doi: 10.1007/978-3-031-13483-8_10.

[17] S. K. Pandey, J. Dey, and S. Banerjee, "Design of Optimal PID Controller for Control of Twin Rotor MIMO System (TRMS)," *Advances in Power and Control Engineering*, pp. 93–106, Nov. 2019, doi: 10.1007/978-981-15-0313-9_7.

[18] F. Gopmandal and A. Ghosh, "LQR-based MIMO PID control of a 2-DOF helicopter system with uncertain cross-coupled gain," *IFAC-PapersOnLine*, vol. 55, no. 22, pp. 183–188, 2022, doi: 10.1016/j.ifacol.2023.03.031.

[19] A. T. Azar, A. S. Sayed, A. S. Shahin, H. A. Elkholy, and H. H. Ammar, "PID Controller for 2-DOFs Twin Rotor MIMO System Tuned with Particle Swarm Optimization," *Proceedings of the International Conference on Advanced Intelligent Systems and Informatics 2019*, pp. 229–242, Oct. 2019, doi: 10.1007/978-3-030-31129-2_22.

[20] S. Sengupta and C. Dey, "Optimal Auto-Tuned PID Controller for Twin Rotor MIMO System," *Lecture Notes in Mechanical Engineering*, pp. 591–601, 2023, doi: 10.1007/978-981-19-9285-8_56.

[21] V. Mihaly, M. Şuşcă, and E. H. Dulf, "μ-Synthesis FO-PID for Twin Rotor Aerodynamic System," *Mathematics*, vol. 9, no. 19, p. 2504, Oct. 2021, doi: 10.3390/math9192504.

[22] S. K. Pandey and V. Laxmi, "Optimal Control of Twin Rotor MIMO System Using LQR Technique," *Smart Innovation, Systems and Technologies*, pp. 11–21, Dec. 2014, doi: 10.1007/978-81-322-2205-7_2.

[23] S. K. Valluru, M. Singh, Ayush, and A. Dharavath, "Design and Experimental Implementation of Multi-loop LQR, PID, and LQG Controllers for the Trajectory Tracking Control of Twin Rotor MIMO System," *Intelligent Communication, Control and Devices*, pp. 599–608, Aug. 2019, doi: 10.1007/978-981-13-8618-3_62.

[24] M. Z. Ghellab, S. Zeghlache, A. Djerioui, and L. Benyettou, "Experimental validation of adaptive RBFNN global fast dynamic terminal sliding mode control for twin rotor MIMO system against wind effects," *Measurement*, vol. 168, p. 108472, Jan. 2021, doi: 10.1016/j.measurement.2020.108472.

[25] K. R. Palepogu and S. Mahapatra, "Design of sliding mode control with state varying gains for a Benchmark Twin Rotor MIMO System in Horizontal Motion," *European Journal of Control*, p. 100909, Oct. 2023, doi: 10.1016/j.ejcon.2023.100909.

[26] K. Raj, S. K. Choudhary, and V. Muthukumar, "Robust finite-time sliding mode control of twin rotor MIMO system," *International Journal of Modelling, Identification and Control*, vol. 35, no. 1, p. 1, 2020, doi: 10.1504/ijmic.2020.113292.

[27] O. Y. Ismael, M. Qasim, and M. N. Noaman, "Equilibrium Optimizer-Based Robust Sliding Mode Control of Magnetic Levitation System," *Journal Européen des Systèmes Automatisés*, vol. 54, no. 1, pp. 131–138, Feb. 2021, doi: 10.18280/jesa.540115.

[28] K. R. Palepogu and S. Mahapatra, "Pitch orientation control of twin-rotor MIMO system using sliding mode controller with state varying gains," *Journal of Control and Decision*, pp. 1–11, Jan. 2023, doi: 10.1080/23307706.2023.2165977.

[29] M. Panda, N. K. Peyada, and A. Ghosh, "Saturated adaptive backstepping control of uncertain nonlinear systems with validation using twin rotor system," *Journal of the Franklin Institute*, vol. 357, no. 18, pp. 13477–13510, Dec. 2020, doi: 10.1016/j.jfranklin.2020.10.003.

[30] S. Zeghlache, L. Benyettou, A. Djerioui, and M. Z. Ghellab, "Twin Rotor MIMO System Experimental Validation of Robust Adaptive Fuzzy Control Against Wind Effects," *IEEE Systems Journal*, vol. 16, no. 1, pp. 409–419, Mar. 2022, doi: 10.1109/jsyst.2020.3034993.

[31] Y. Li, H. Zhang, Z. Wang, C. Huang, and H. Yan, "Hybrid adaptive bionic fuzzy control method for MIMO systems with dead-zone input," *Journal of the Franklin Institute*, vol. 360, no. 10, pp. 6804–6826, Jul. 2023, doi: 10.1016/j.jfranklin.2023.04.029.

[32] B. Loutfi, Z. Samir, D. Ali, and G. M. Zinelaabidine, "Real Time Implementation of Type-2 Fuzzy Backstepping Sliding Mode Controller for Twin Rotor MIMO System (TRMS)," *Traitement du Signal*, vol. 36, no. 1, pp. 1–11, Apr. 2019, doi: 10.18280/ts.360101.

[33] J.-G. Juang, W.-K. Liu, and R.-W. Lin, "A hybrid intelligent controller for a twin rotor MIMO system and its hardware implementation*," ISA Transactions*, vol. 50, no. 4, pp. 609–619, Oct. 2011, doi: 10.1016/j.isatra.2011.06.006.

[34] H. R. Karimi and M. R. J. Motlagh, "Robust Feedback Linearization Control for a non Linearizable MIMO Nonlinear System in the Presence of Model Uncertainties," *2006 IEEE International Conference on Service Operations and Logistics, and Informatics*, pp. 965-970, 2006, doi: 10.1109/soli.2006.328881.

[35] O. Y. Ismael, M. Qasim, M. N. Noaman, and A. Kurniawan, "Salp Swarm Algorithm-Based Nonlinear Robust Control of Magnetic Levitation System Using Feedback Linearization Approach," *Proceedings of the 3rd International Conference on Electronics, Communications and Control Engineering*, pp. 58-64, 2020, doi: 10.1145/3396730.3396734.

[36] K. U. Ebirim, A. Lecchini-Visintini, M. Rubagotti, and E. Prempain, "Constrained Model Predictive Control With Integral Action for Twin Rotor MIMO Systems," *Journal of Dynamic Systems, Measurement, and Control*, vol. 145, no. 8, Jul. 2023, doi: 10.1115/1.4062735.

[37] R. Raghavan and S. Thomas, "Practically Implementable Model Predictive Controller for a Twin Rotor Multi-Input Multi-Output System," *Journal of Control, Automation and Electrical Systems*, vol. 28, no. 3, pp. 358–370, Mar. 2017, doi: 10.1007/s40313-017-0311-5.

[38] L. Dutta and D. K. Das, "Nonlinear Disturbance Observer Based Adaptive Explicit Nonlinear Model Predictive Control Design for a Class of Nonlinear MIMO System*," IEEE Transactions on Aerospace and Electronic Systems*, pp. 1–12, 2022, doi: 10.1109/taes.2022.3211252.

[39] L. Dutta and D. K. Das, "A New Adaptive Explicit Nonlinear Model Predictive Control Design for a Nonlinear MIMO System: An Application to Twin Rotor MIMO System," *International Journal of Control, Automation and Systems*, vol. 19, no. 7, pp. 2406–2419, Mar. 2021, doi: 10.1007/s12555-020-0272-5.

[40] S. V. Raković and W. S. Levine, *Handbook of Model Predictive Control*. Springer International Publishing, 2019. doi: 10.1007/978-3-319-77489-3.

[41] N. Wu, D. Li, Y. Xi, and B. de Schutter, "Distributed Event-Triggered Model Predictive Control for Urban Traffic Lights," in *IEEE Transactions on Intelligent Transportation Systems*, vol. 22, no. 8, pp. 4975-4985, Aug. 2021, doi: 10.1109/TITS.2020.2981381.

[42] M. H. Moradi, "Predictive control with constraints, J.M. Maciejowski; Pearson Education Limited, Prentice Hall, London, 2002, pp. IX+331, price £35.99," *International Journal of Adaptive Control and Signal Processing*, vol. 17, no. 3, pp. 261–262, Mar. 2003, doi: 10.1002/acs.736.

[43] R. G. Tiwalkar, S. S. Vanamane, S. S. Karvekar, and S. B. Velhal, "Model predictive controller for position control of twin rotor MIMO system," *2017 IEEE International Conference on Power, Control, Signals and Instrumentation Engineering (ICPCSI)*, pp. 952-957, 2017, doi: 10.1109/icpcsi.2017.8391852.

[44] U. M. Nath, C. Dey, and R. K. Mudi, "Controlling of Twin Rotor MIMO System (TRMS) based on Multivariable Model Predictive Control," *Lecture Notes in Electrical Engineering*, pp. 493–499, Nov. 2020, doi: 10.1007/978-981-15-7486-3_44.

[45] R. Raghavan and S. Thomas, "Practically Implementable Model Predictive Controller for a Twin Rotor Multi-Input Multi-Output System," *Journal of Control, Automation and Electrical Systems*, vol. 28, no. 3, pp. 358–370, Mar. 2017, doi: 10.1007/s40313-017-0311-5.

[46] D.-A. Dutescu, M.-B. Radac, and R.-E. Precup, "Model predictive control of a nonlinear laboratory twin rotor aero-dynamical system," *2017 IEEE 15th International Symposium on Applied Machine Intelligence and Informatics (SAMI)*, pp. 000037-000042, 2017, doi: 10.1109/sami.2017.7880339.

[47] A. Rahideh and M. H. Shaheed, "Constrained output feedback model predictive control for nonlinear systems," *Control Engineering Practice*, vol. 20, no. 4, pp. 431–443, Apr. 2012, doi: 10.1016/j.conengprac.2011.12.003.

[48] A. Rahideh and M. Hasan Shaheed, "Stable model predictive control for a nonlinear system," *Journal of the Franklin Institute*, vol. 348, no. 8, pp. 1983–2004, Oct. 2011, doi: 10.1016/j.jfranklin.2011.05.015.

[49] M. T. Milojkovic, A. D. Djordjevic, S. Lj. Peric, M. B. Milovanovic, Z. H. Peric, and N. B. Dankovic, "Model Predictive Control of Nonlinear MIMO Systems Based on Adaptive Orthogonal Polynomial Networks," *Elektronika ir Elektrotechnika*, vol. 27, no. 2, pp. 4–10, Apr. 2021, doi: 10.5755/j02.eie.28780.

[50] F. A. Hashim, R. R. Mostafa, A. G. Hussien, S. Mirjalili, and K. M. Sallam, "Fick's Law Algorithm: A physical law-based algorithm for numerical optimization," *Knowledge-Based Systems*, vol. 260, p. 110146, Jan. 2023, doi: 10.1016/j.knosys.2022.110146.

[51] J. A. E. Andersson, J. Gillis, G. Horn, J. B. Rawlings, and M. Diehl, "CasADi: a software framework for nonlinear optimization and optimal control," *Mathematical Programming Computation*, vol. 11, no. 1, pp. 1–36, Jul. 2018, doi: 10.1007/s12532-018-0139-4.

[52] A. Wächter and L. T. Biegler, "Line Search Filter Methods for Nonlinear Programming: Motivation and Global Convergence," *SIAM Journal on Optimization*, vol. 16, no. 1, pp. 1–31, Jan. 2005, doi: 10.1137/s1052623403426556.

[53] L. Grüne and J. Pannek. *Nonlinear Model Predictive Control*. Springer International Publishing, 2017, doi: 10.1007/978-3-319-46024-6.

[54] E. F. Camacho and C. Bordons. *Model Predictive control*. Springer London, 2007, doi: 10.1007/978-0-85729-398-5.

[55] O. Y. Ismael, M. Almaged, and A. I. Abdulla, "Nonlinear Model Predictive Control-based Collision Avoidance for Mobile Robot," *Journal of Robotics and Control (JRC)*, vol. 5, no. 1, pp. 142-151, doi: 10.18196/jrc.v5i1.20615.

[56] J. B. Rawlings and K. R. Muske, "The stability of constrained receding horizon control," *IEEE Transactions on Automatic Control*, vol. 38, no. 10, pp. 1512–1516, 1993, doi: 10.1109/9.241565.

[57] L. Grüne and J. Pannek. *Nonlinear Model Predictive Control*. Springer London, 2011, doi: 10.1007/978-0-85729-501-9.

[58] S. Li, H. Chen, M. Wang, A. A. Heidari, and S. Mirjalili, "Slime mould algorithm: A new method for stochastic optimization," *Future Generation Computer Systems*, vol. 111, pp. 300–323, Oct. 2020, doi: 10.1016/j.future.2020.03.055.

[59] A. G. Hussien, A. E. Hassanien, E. H. Houssein, M. Amin, and A. T. Azar, "New binary whale optimization algorithm for discrete optimization problems," *Engineering Optimization*, vol. 52, no. 6, pp. 945–959, Jun. 2019, doi: 10.1080/0305215x.2019.1624740.

[60] S. M. Ahmad, A. J. Chipperfield, and M. O. Tokhi, "Dynamic modelling and linear quadratic Gaussian control of a twin-rotor multi-input multi-output system," *Proceedings of the Institution of Mechanical Engineers, Part I: Journal of Systems and Control Engineering*, vol. 217, no. 3, pp. 203–227, May 2003, doi: 10.1177/095965180321700304.