

Article

Model Predictive Control Technique for Ducted Fan Aerial Vehicles Using Physics-Informed Machine Learning

Tayyab Manzoor ¹, Hailong Pei ^{1,*}, Zhongqi Sun ² and Zihuan Cheng ¹

¹ Key Laboratory of Autonomous Systems and Networked Control, Ministry of Education, Unmanned Aerial Vehicle Systems Engineering Technology Research Center of Guangdong, School of Automation Science and Engineering, South China University of Technology, Guangzhou 510640, China

² School of Automation, Beijing Institute of Technology, Beijing 100081, China

* Correspondence: auhlpei@scut.edu.cn

Abstract: This paper proposes a model predictive control (MPC) approach for ducted fan aerial robots using physics-informed machine learning (ML), where the task is to fully exploit the capabilities of the predictive control design with an accurate dynamic model by means of a hybrid modeling technique. For this purpose, an indigenously developed ducted fan miniature aerial vehicle with adequate flying capabilities is used. The physics-informed dynamical model is derived offline by considering the forces and moments acting on the platform. On the basis of the physics-informed model, a data-driven ML approach called adaptive sparse identification of nonlinear dynamics is utilized for model identification, estimation, and correction online. Thereafter, an MPC-based optimization problem is computed by updating the physics-informed states with the physics-informed ML model at each step, yielding an effective control performance. Closed-loop stability and recursive feasibility are ensured under sufficient conditions. Finally, a simulation study is conducted to concisely corroborate the efficacy of the presented framework.

Keywords: aerial robotics; flight control; machine learning; model predictive control; trajectory tracking; UAV



Citation: Manzoor, T.; Pei, H.; Sun, Z.; Cheng, Z. Model Predictive Control Technique for Ducted Fan Aerial Vehicles Using Physics-Informed Machine Learning. *Drones* **2023**, *7*, 4. <https://doi.org/10.3390/drones7010004>

Academic Editor: Andrey V. Savkin

Received: 28 November 2022

Revised: 18 December 2022

Accepted: 20 December 2022

Published: 21 December 2022



Copyright: © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

1.1. Literature Review

The ducted fan aerial vehicles (DFAVs) are a type of vertical take-off and landing platforms that have the innate capabilities of a helicopter and fixed-wing aircraft [1,2]. These airborne vehicles are used in a wide number of applications in different areas such as transportation, inspection, aerial surveillance, and manipulation [3,4]. Due to their annular fuselage mechanism called duct [5], these aerial robots are known for their safety in congested, cluttered, and hazardous environments [6,7]. Having vector thrust capabilities [8], DFAVs also have several advantages over other configurations, such as fixed-wing platforms, helicopters, open-rotor, and shrouded quadrotors [9]. However, complex flow distribution makes it arduous to model these aircraft accurately [10], consequently making the flight control design more challenging.

Over the years, several flight control systems based on different strategies have been presented to control these vehicles [9]. In this regard, model predictive control (MPC) has proved itself as an advanced control approach that can address many issues associated with aerial applications [11]. One of the simplest techniques is linear MPC, which has been employed for the purpose of attitude control for these aerial platforms [12,13]. An improved unified control framework based on linear MPC for position and attitude control with Kalman filter as a disturbance observer (DOB) has also been studied and validated [14]. Moreover, robust MPC with a DOB and adaptive MPC with two DOBs have been presented

to compensate for the effect of model inaccuracies and disturbances [15]. For achieving similar tasks with some improvements compared to robust MPC [15], composite disturbance rejection approaches based on MPC, referred to as MPC-based compound flight control (MPC-CFC), have been proposed [16,17].

In the aforementioned works, some problems can be solved to improve the control performance of future MPC-driven control systems.

1. The performance of MPC depends on the accurate model of the system dynamics [18,19], which is challenging to obtain. This problem is even more prevalent in flight control, where varying flight conditions, disturbances, faults, and model uncertainties may lead to ineffective control performance.
2. In the disturbance rejection MPC framework based on DOB (e.g., [16,17]), the dynamics of the disturbance profile cannot be faster than the dynamics of DOB, which implies that this type of composite method is not suitable to estimate and compensate for fast time-varying disturbances.

There are two types of dominant techniques for dynamical system modeling [20]: (1) Physics-informed dynamics, where the equations of motion are derived from governing physical rules, and (2) machine learning (ML) based data-driven system modeling [21]. To construct an end-to-end dynamical model, these techniques are employed separately. There is a growing potential for integrating data-driven techniques based on ML in the aerospace industry [22]. One of the commonly used ML techniques is the neural network (NN) [23–26]. Nevertheless, NN-based methods need massive training data, which may be uninterpretable and difficult to include constraints. These reasons limit their utilization for online system identification [27]. One alternative is to use sparse identification of nonlinear dynamics (SINDy) developed in [28], which has shown effective control and predictive capabilities with MPC in the presence of noise and is more suitable for low and medium data than the NN technique. Moreover, it has demonstrated strong parametric robustness, effective training, and execution time [27]. While the SINDy algorithm is an effective strategy for multiple learning, an adaptive SINDy technique is more suitable that can efficiently update and correct the dynamics online in a repetitive fashion [29,30]. However, it is challenging to seamlessly capture the correct model with ML techniques alone to satisfy the conservation laws and constraints. On the other hand, physics-informed dynamical models can capture constraints and conservation laws. However, they are either too simple or ultra-complex that may become too expensive to handle online due to high computational costs, particularly for varying system dynamics for MPC. In this aspect, a physics-informed ML hybrid-modeling technique can be employed to seamlessly incorporate physical models and data in all circumstances involving high-dimensional, physically understood, and uncertain contexts [31–36].

1.2. Contributions

Motivated by the above discussion, this article presents an MPC-based control strategy for DFAVs via a hybrid modeling approach called physics-informed ML. First, the physics-informed model (nominal system) for the aerial robot is constructed by considering the forces and moments offline. Moreover, it is considered that the nominal system is not affected by any disturbances or uncertainties. Thereafter, a data-driven approach called adaptive SINDy is employed for estimating and compensating for any parametric changes online during the flight. The optimization problem is solved online, where in each step, the physics-informed system is updated by the physics-informed ML. This way, the control action applied to the aerial vehicle is optimal with respect to the current state. In Table 1, the advantages and disadvantages of existing techniques over the proposed scheme are provided. The contributions of this manuscript are expressed in the following manner:

- To fully exploit the potential of MPC, an online-based predictive control algorithm using physics-informed ML without the utilization of DOB is presented, unlike [15–17]. By employing the methodology in such a way, the inherent issues encountered in the disturbance rejection MPC framework [16,17] can be solved.

- For efficient response and to avoid any usual computational complexity problem, only the data-driven part in the hybrid modeling is determined online for model correction. To further enhance the computational efficiency of the developed control algorithm, the physics-informed model is also updated by the physics-informed ML model in each step while solving the optimization problem.
- Unlike [12–15], theoretical properties such as recursive feasibility and closed-loop stability with constraints satisfaction under sufficient conditions are derived.
- In contrast to the existing disturbance rejection framework [16], the designed approach demonstrates effective performance. Furthermore, the constructed approach can be implemented in other robotics systems to attain similar goals.

Table 1. A concise comparison between existing MPC-based control methods and the proposed technique for DFAVs (order by year)

References	Features ¹	Advantages	Disadvantages
Linear MPC [12,13]	AMS	Simple design, effective computations.	Lack adequate tracking and robustness.
1. Disturbance rejection RMPC ² 2. Disturbance rejection adaptive MPC [15]	ACDMPS	Considered time-delays, discrete time optimization problem is rather easy to be deployed.	1. RMPC ² : lack effective control performance, 2. Adaptive MPC: design intricacy may not be effective, feasibility analysis is not available for both schemes.
Observer-based MPC [14]	ACDMPS	Effective computations, easy real-time implementation, considered time delays.	Ineffective control performance, recursive feasibility cannot be established for the entire flight.
Compound RMPC ² [16,17]	ACDMPS	-	Inadequate performance, suitable if the DOB's dynamics is faster than disturbance dynamics.

¹ A: attitude tracking, C: composite technique, D: DOB, M: physics-informed modeling, P: position tracking, S: simulation study. ² RMPC: Robust MPC.

1.3. Organization

In Section 2, the problem formulation is constructed in the physics-informed modeling and physics-informed ML model subsections. Furthermore, control objectives are defined, and a few assumptions with preliminaries are established for smooth control operation. The control approach includes MPC development and its feasibility and stability proofs and is provided in Section 3. Section 4 illustrates the numerical implementation that includes comparative analysis with some discussion. Section 5 concludes the article.

1.4. Notation

$\mathbb{N}$ and $\mathbb{R}$ denote all non-negative integers and real space, respectively. $\|(\cdot)\| \triangleq \sqrt{(\cdot)^T(\cdot)}$ is the Euclidean norm. $\text{diag}\{(\cdot)_{11}, (\cdot)_{22}, \dots, (\cdot)_{nn}\}$ denotes the diagonal matrix with entries $(\cdot)_{11}, \dots, (\cdot)_{nn} \in \mathbb{R}$. Q-weighted norm is referred to as $\|(\cdot)\|_Q \triangleq \sqrt{(\cdot)^T Q (\cdot)}$, where Q denotes a positive definite matrix. Consider any given system state/input as $\circ$, $\circ(\tau|t_l)$ means $\circ$ at τ predicted at t_l . Superscript $\circ^*$ is utilized to define state/input after optimization. Accents $\tilde{\circ}$ and $\hat{\circ}$ represent the nominal state/nominal control input and estimated state/input, respectively. In the entire manuscript, physics-informed dynamics and physics-informed ML are referred to as nominal and real systems, and these terms are used interchangeably. A bold font style is employed to represent vectors and matrices.

2. Problem Formulation

The hybrid modeling procedure is divided into two phases, i.e., physics-informed modeling (offline) and data-driven scheme (online), which is based on lots of physics and less data, shown in Figure 1.

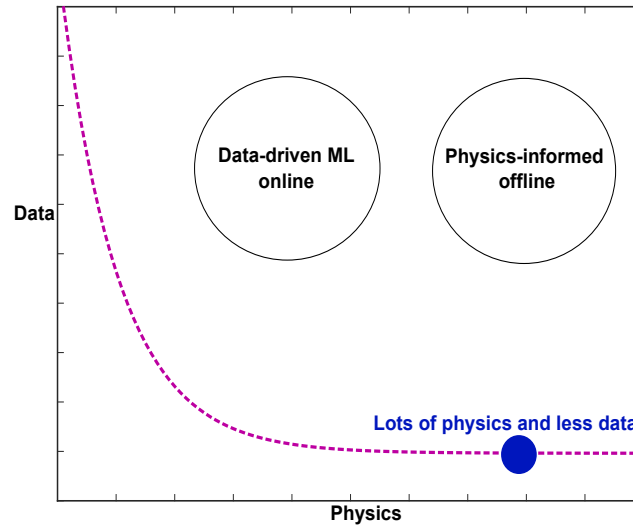


Figure 1. Physics and data scenario in the presented scheme.

2.1. Physics-Informed Modelling

In Figure 2, the ducted fan aerial robot is represented by two coordinate frames, i.e., body-fixed frame $\mathcal{B}_f \triangleq \{\mathcal{B}_o, \mathcal{B}_x, \mathcal{B}_y, \mathcal{B}_z\}$ and inertial frame $\mathcal{I}_f \triangleq \{\mathcal{I}_o, \mathcal{I}_x, \mathcal{I}_y, \mathcal{I}_z\}$. $\xi^{\mathcal{I}} = [\xi_x^{\mathcal{I}}, \xi_y^{\mathcal{I}}, \xi_z^{\mathcal{I}}]^T \in \mathbb{R}^3$, $V_{\mathcal{I}} = [V_x^{\mathcal{I}}, V_y^{\mathcal{I}}, V_z^{\mathcal{I}}]^T \in \mathbb{R}^3$, $V_{\mathcal{B}} = [V_x^{\mathcal{B}}, V_y^{\mathcal{B}}, V_z^{\mathcal{B}}]^T \in \mathbb{R}^3$ and $\omega_{\mathcal{B}} = [p, q, r]^T \in \mathbb{R}^3$ are position, linear velocity in $\mathcal{I}_f$, linear velocity in $\mathcal{B}_f$, and angular velocity, respectively. $\Delta_{wind} = [\Delta_x^{\mathcal{B}}(t), \Delta_y^{\mathcal{B}}(t), \Delta_z^{\mathcal{B}}(t)]^T$ are disturbances acting on the flying robot. $\Theta = [\phi, \theta, \psi]^T \in \mathbb{R}^3$ denotes the attitude of the aircraft, where ϕ , θ and ψ are roll, pitch, and yaw angle, respectively. From these rotation angles, the rotational matrix $T_{\mathcal{B}\mathcal{I}} \in SO(3)$ from $\mathcal{B}_f$ and $\mathcal{I}_f$ can be defined as (see among others, e.g., [9]):

$$T_{\mathcal{B}\mathcal{I}} = \begin{bmatrix} C_\psi C_\theta & C_\psi S_\theta S_\phi - S_\psi C_\phi & C_\psi S_\theta C_\phi + S_\psi S_\phi \\ S_\psi C_\theta & S_\psi S_\theta S_\phi + C_\psi C_\phi & S_\psi S_\theta C_\phi - C_\psi S_\phi \\ -S_\theta & C_\theta S_\phi & C_\theta C_\phi \end{bmatrix}, \quad (1)$$

where $C_{(*)} = \cos(*)$ and $S_{(*)} = \sin(*)$, with $(*)$ denotes any angle. The rotational kinematics of the airborne platform is described in the following form:

$$\dot{\Theta} = \Omega \omega_{\mathcal{B}}, \quad \Omega = \begin{bmatrix} C_\theta & 0 & -S_\theta \\ S_\theta S_\phi / C_\phi & 1 & C_\theta S_\phi / C_\phi \\ S_\theta / C_\phi & 0 & C_\phi C_\phi \end{bmatrix}. \quad (2)$$

Other flight components in Figure 2 are expressed as follows:

$$\begin{aligned} V_{wind} &= \sqrt{(V_x^{\mathcal{B}} - \Delta_x^{\mathcal{B}})^2 + (V_y^{\mathcal{B}} - \Delta_y^{\mathcal{B}})^2 + (V_z^{\mathcal{B}} - \Delta_z^{\mathcal{B}})^2}, \\ \alpha &= -\arccos\left(\frac{V_z^{\mathcal{B}} - \Delta_z^{\mathcal{B}}}{V_{wind}}\right), 0 \leq \alpha \leq \pi, \\ \beta &= \arctan\left(\frac{V_y^{\mathcal{B}} - \Delta_y^{\mathcal{B}}}{V_x^{\mathcal{B}} - \Delta_x^{\mathcal{B}}}\right), -\frac{\pi}{2} \leq \beta \leq \frac{\pi}{2}, \end{aligned} \quad (3)$$

where V_{wind} , α , and β are denoted as airspeed, angle of attack, and side-slip angle, respectively. The translational and rotational dynamics of the VTOL aircraft can be formulated as

$$\begin{cases} \ddot{\xi}_I = \dot{V}_I = T_{BI} F_B / m + G_a, \\ \dot{\omega}_B = I^{-1} (M_B + I \omega_B \times \omega_B), \end{cases} \quad (4)$$

where m , F_B , M_B , and I are the mass, total force, total moments, and diagonal matrix consisting of the moment of inertia, respectively. Moreover, $G_a = [0 \ 0 \ g]^T$, and g is the gravitational acceleration. Velocity of the airflow exhausted from the fan (V_{af}) can be expressed as:

$$V_{af} = -\frac{V_z^B - \Delta_z^B}{2} + \sqrt{\left(\frac{V_z^B - \Delta_z^B}{2}\right)^2 + \frac{T}{2\rho A_{fd}}}, \quad (5)$$

where $T = A_{\omega T} \omega_r^2$ is the thrust, and ω_r and $A_{\omega T}$ denote the rotation speed and thrust coefficient, respectively. Moreover, ρ and A_{fd} are the free air density and fan disk area, respectively. Next, we describe the local airspeed on left-wing (V_{lw}) and right-wing (V_{rw}) with local α on left-wing (α_{lw}) and right-wing (α_{rw}) are given as follows:

$$\begin{aligned} V_{lw} &= \sqrt{(V_x^B + r\ell_w - \Delta_x)^2 + (V_z^B - \Delta_z)^2}, \\ V_{rw} &= \sqrt{(V_x^B - r\ell_w - \Delta_x)^2 + (V_z^B - \Delta_z)^2}, \\ \alpha_{lw} &= C_{\alpha_{lw}} = -(V_z^B - \Delta_z) / V_{lw}, \quad 0 \leq \alpha_{lw} \leq \pi, \\ \alpha_{rw} &= C_{\alpha_{rw}} = -(V_z^B - \Delta_z) / V_{rw}, \quad 0 \leq \alpha_{rw} \leq \pi, \end{aligned} \quad (6)$$

where ℓ_w represents the lever arm from the B_z axis to the aerodynamic center of the left and right/left wing. Generally, an aircraft is affected by four main forces during flight (lift, weight, thrust, and drag), depicted in Figure 2c. In the current scenario, the F_B can be defined as follows:

$$\begin{aligned} F_B &= \underbrace{\begin{bmatrix} 0 \\ 0 \\ -A_{\omega T} \omega_r^2 \end{bmatrix}}_{\text{Thrust force}} + \underbrace{V_{wind}^2 \begin{bmatrix} -(A_{L_d} C_\alpha + A_{D_d} S_\alpha) C_\beta \\ -(A_{L_d} C_\alpha + A_{D_d} S_\alpha) S_\beta \\ -A_{L_d} S_\alpha + A_{D_d} C_\alpha \end{bmatrix}}_{\text{Duct-body force}} \\ &+ \underbrace{\begin{bmatrix} -(A_{L_w} V_{lw}^2 C_{\alpha_{lw}} + A_{L_w} V_{rw}^2 C_{\alpha_{rw}} + A_{D_w} V_{lw}^2 S_{\alpha_{lw}} + A_{D_w} V_{rw}^2 S_{\alpha_{rw}}) \\ 0 \\ -A_{L_w} V_{lw}^2 S_{\alpha_{lw}} - A_{L_w} V_{rw}^2 S_{\alpha_{rw}} + A_{D_w} V_{lw}^2 C_{\alpha_{lw}} + A_{D_w} V_{rw}^2 C_{\alpha_{rw}} \end{bmatrix}}_{\text{Wing force}} \\ &+ \underbrace{\rho A_{fd} V_{af} \zeta_{ratio} \begin{bmatrix} V_x^B - \Delta_x^B \\ V_y^B - \Delta_y^B \\ 0 \end{bmatrix}}_{\text{Momentum drag}}, \end{aligned} \quad (7)$$

where A_{L_d} and A_{D_d} are the lift and drag coefficients, respectively. Moreover, A_{D_w} and A_{L_w} represent the drag and lift coefficients of the half-wing section of the aircraft, respectively. ζ_{ratio} denotes the damping ratio and is defined as

$$\zeta_{ratio} = \frac{V_{sx} - V'_{sx}}{V_{sx}}. \quad (8)$$

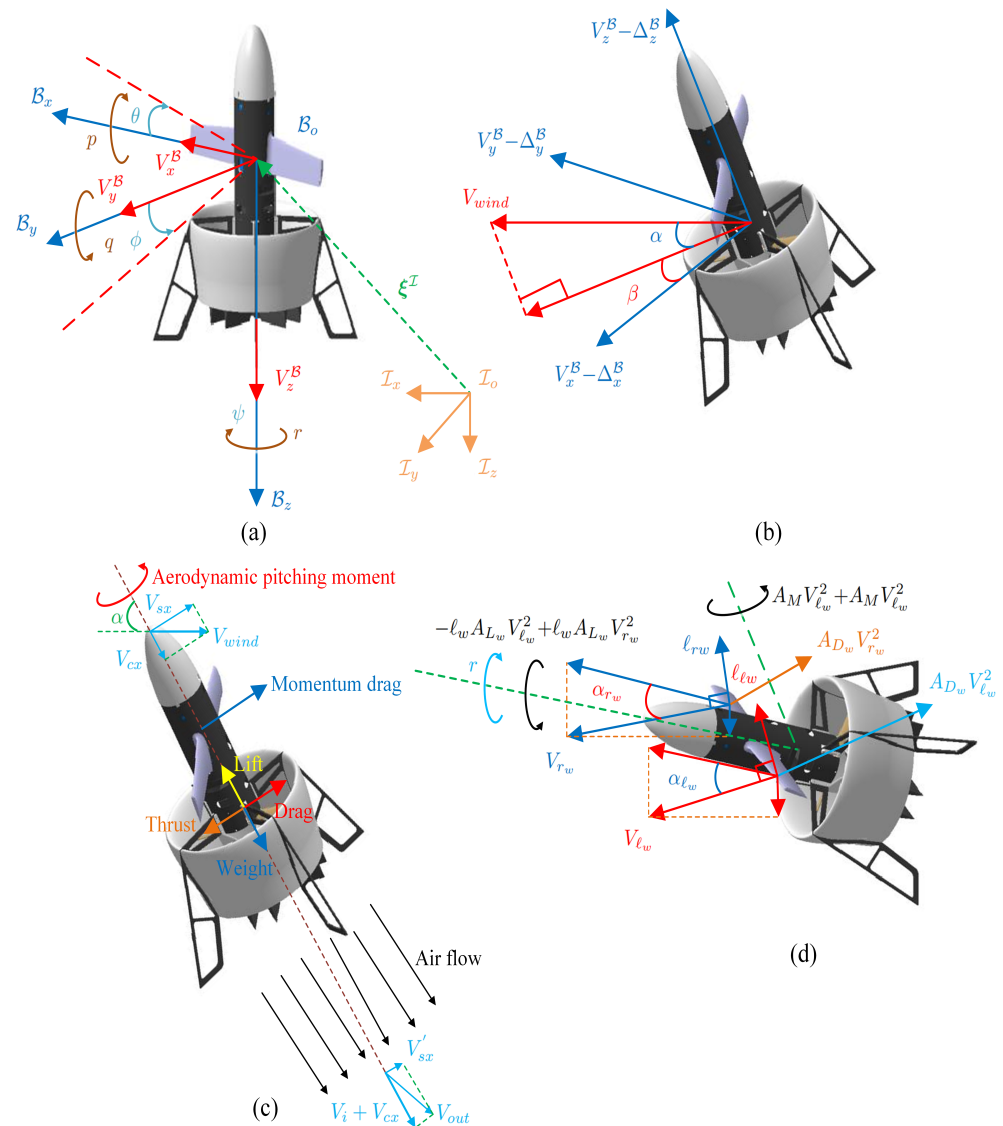


Figure 2. (a) Inertial and body-fixed frame; (b) other components during flight; (c) aerodynamic effects on the duct; (d) aerodynamic effects on the wings.

The total moment is expressed in the following manner:

$$\begin{aligned}
 \mathbf{M}_{\mathcal{B}} = & \underbrace{A_{cs} V_{af}^2 \begin{bmatrix} -l_1 \delta_1 + l_1 \delta_3 \\ -l_1 \delta_2 + l_1 \delta_4 \\ l_2 \delta_1 + l_2 \delta_2 + l_2 \delta_3 + l_2 \delta_4 \end{bmatrix}}_{\text{Control surfaces}} + \underbrace{\begin{bmatrix} 0 \\ 0 \\ k_{\omega s} \omega_r^2 \end{bmatrix}}_{\text{Fan torque}} + \underbrace{I_f \omega_r \begin{bmatrix} -q \\ p \\ 0 \end{bmatrix}}_{\text{Gyroscopic effect}} \\
 & + \underbrace{\epsilon_{drag} \rho A_{fd} V_{af} \zeta_{ratio} \begin{bmatrix} V_x^{\mathcal{B}} - \Delta_x^{\mathcal{B}} \\ V_y^{\mathcal{B}} - \Delta_y^{\mathcal{B}} \\ 0 \end{bmatrix}}_{\text{Aerodynamic pitching moment}} + \underbrace{\epsilon_D V_{wind}^2 \begin{bmatrix} -(A_{L_d} C_{\alpha} + A_{D_d} S_{\alpha}) C_{\beta} \\ -(A_{L_d} C_{\alpha} + A_{D_d} S_{\alpha}) S_{\beta} \\ -A_{L_d} S_{\alpha} + A_{D_d} C_{\alpha} \end{bmatrix}}_{\text{Aerodynamic pitching moment}} \\
 & + \underbrace{\begin{bmatrix} 0 \\ A_M V_{\ell_w}^2 + A_M V_{\ell_w}^2 \\ -\ell_w A_{L_w} V_{\ell_w}^2 + \ell_w A_{L_w} V_{r_w}^2 \end{bmatrix}}_{\text{Wing moment}} + \underbrace{A_{ar} V_{af}^2}_{\text{Anti-rotation torque}}, \quad (9)
 \end{aligned}$$

where $k_{\omega s}$, ϵ_{drag} , ϵ_D , I_f , A_{ar} , l_1 , l_2 , A_{cs} , δ_i , and A_M represent the rotation speed with respect to fan torque coefficient, lever arms of momentum drag, duct body force, fan inertia, constant coefficient, lever arms related to roll/pitch-axis, and yaw-axis, deflection angle to force coefficient, the deflection angle of i -th control surface, and moment coefficient of the half-wing section of the vehicle, respectively. Consider the drone's system defined as follows:

$$\dot{\mathbf{x}}_s = f_s(\mathbf{x}_s(t), \mathbf{u}_s(t); \boldsymbol{\eta}_s), \mathbf{x}_s(0) = \mathbf{x}_0, \quad (10)$$

where $\mathbf{x}_s = [\xi_{\mathcal{I}} \ V_{\mathcal{I}} \ \boldsymbol{\Theta} \ \boldsymbol{\omega}_{\mathcal{B}}]^T \in \mathbb{R}^{12}$ denotes the system states. f_s and $\boldsymbol{\eta}_s$ are the dynamics and the aircraft's parameters. It is considered that the system (10) is directly affected by disturbances and uncertainties. $\mathbf{u}_s(t) = [T \ \mathbf{u}_{s2}^T]^T \in \mathbb{R}^4$ represents control input, where $\mathbf{u}_{s2}$ denotes the regulated moment input and is provided as follows:

$$\mathbf{u}_{s2} = \begin{bmatrix} u_{11} \\ u_{12} \\ u_{13} \end{bmatrix} = \begin{bmatrix} -l_1\delta_1 + l_1\delta_3 \\ -l_1\delta_2 + l_1\delta_4 \\ l_2\delta_1 + l_2\delta_2 + l_2\delta_3 + l_2\delta_4 \end{bmatrix}.$$

Next, by considering only the physics-informed model of the aircraft as a nominal system:

$$\dot{\tilde{\mathbf{x}}}_s(t) = f_s(\tilde{\mathbf{x}}_s(t), \tilde{\mathbf{u}}_s(t)). \quad (11)$$

The discrepancy modeling problem is derived from the difference between the quantity of interest (i.e., forces and moments) from a physics-informed model $v_m(t)$ and estimated value $v_o(t)$. Hence, discrepancy $\Delta v(t)$ is expressed as:

$$\Delta v(t) = v_o(t) - v_m(t). \quad (12)$$

Under ideal conditions, the estimated value and physics-informed model may have the same magnitude in (12), i.e., $\Delta v(t) \approx 0$. Nonetheless, such an outcome in actual flight may not be possible due to several factors.

2.2. Data-Driven ML-Adaptive SINDy

Inspired by the existing work on learning from discrepancy model [20] and integrating it with adaptive SINDy by [30] for compensating for any abrupt changes in the system dynamics during different flight conditions of the DFAV, a hybrid modeling approach based on physics-informed dynamics and adaptive SINDy is constructed, shown in Figure 3. The sudden changes in the adaptive SINDy involve only addition, deletion, and modifications. This is useful as it enables us to effectively determine the new and missing terms with less data than to identify the entire model from scratch. There are three basic kinds of model changes in the adaptive SINDy:

1. Modification: If the whole model is unchanged except for model parameters, then least square regression will be employed on the known model to find new parameters;
2. Deletion: If a few terms are removed, then sparse regression can be utilized on the sparse coefficients to identify which terms have been taken out;
3. Addition: To find the model error, the SINDy regression will identify the sparsest combination of inactive terms.

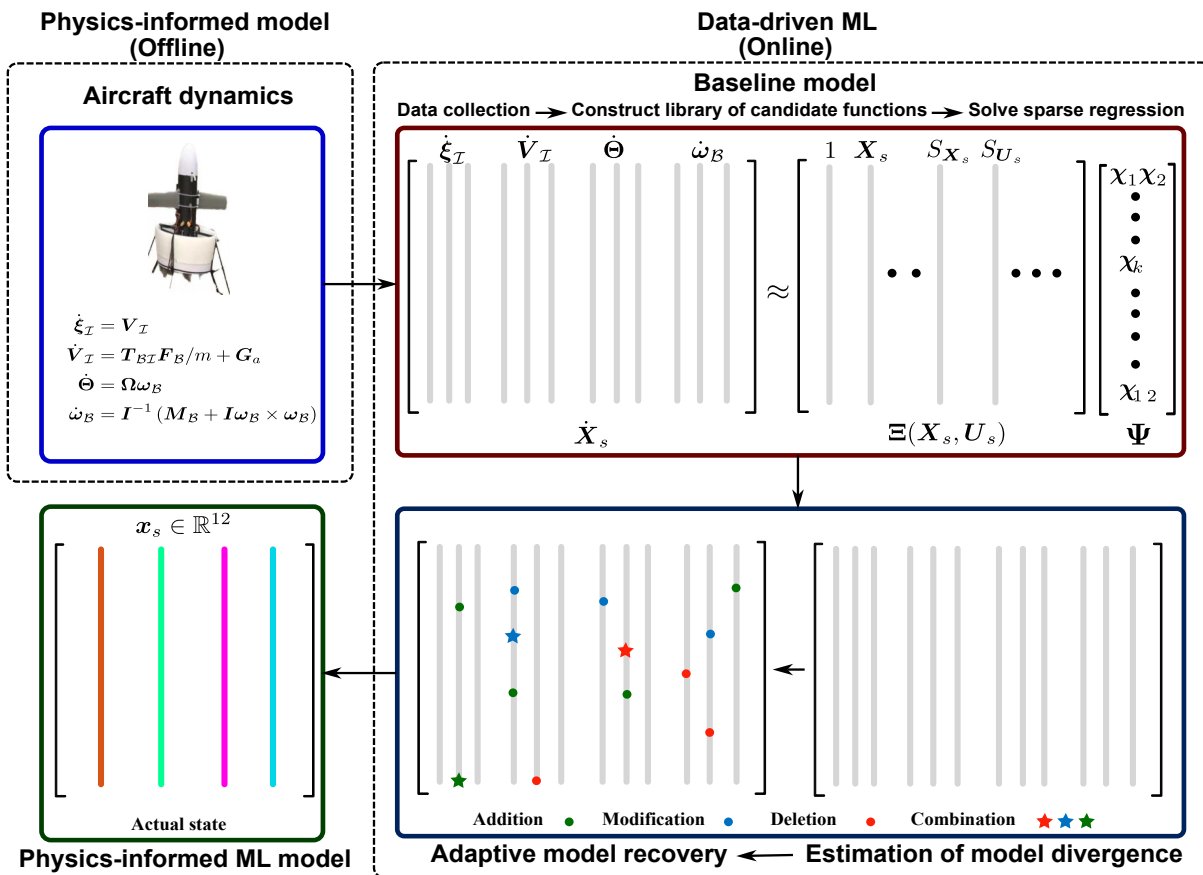


Figure 3. Physics-informed ML model for DFAVs.

2.2.1. Baseline Model

By using the standard procedure of SINDy, we measure the required number of snapshots m_s of x_s and u_s and rearrange them into two data matrices X_s and U_s :

$$\begin{aligned}X_s &= \begin{bmatrix} \zeta_x^I(t_1) & \zeta_y^I(t_1) & \cdots & \psi(t_1) \\ \vdots & \vdots & \ddots & \vdots \\ \zeta_x^I(t_i) & \zeta_y^I(t_i) & \cdots & \psi(t_i) \\ \vdots & \vdots & \ddots & \vdots \\ \zeta_x^I(t_m) & \zeta_y^I(t_m) & \cdots & \psi(t_m) \end{bmatrix}, \\ U_s &= \begin{bmatrix} T(t_1) & u_{11}(t_1) & \cdots & u_{13}(t_1) \\ \vdots & \vdots & \ddots & \vdots \\ T(t_i) & u_{11}(t_i) & \cdots & u_{13}(t_i) \\ \vdots & \vdots & \ddots & \vdots \\ T(t_m) & u_{11}(t_m) & \cdots & u_{13}(t_m) \end{bmatrix},\end{aligned}\quad (13)$$

where t_i is the sampling time. Simplify Equation (13) and transform it into the following form:

$$\begin{aligned}X_s &= [x_{s1}(t_1) \ x_{s2}(t_2) \ x_{s3}(t_3) \ \dots \ x_{ms}(t_m)]^T, \\ U_s &= [u_{s1}(t_1) \ u_{s2}(t_2) \ u_{s3}(t_3) \ \dots \ u_{ms}(t_m)]^T,\end{aligned}\quad (14)$$

where u_{ms} and x_{ms} are control and state vector at the m th sampling time. The value of the aircraft's state is calculated by the numerical differentiation.

$$\begin{aligned}\dot{\mathbf{X}}_s &= [\dot{x}_{s1}(t_1) \ \dot{x}_{s2}(t_2) \ \dot{x}_{s3}(t_3) \ \dots \ \dot{x}_{ms}(t_m)]^T, \\ &= \begin{bmatrix} \ddot{\zeta}_x^I(t_1) & \ddot{\zeta}_y^I(t_1) & \dots & \dot{\psi}(t_1) \\ \vdots & \vdots & \ddots & \vdots \\ \ddot{\zeta}_x^I(t_i) & \ddot{\zeta}_y^I(t_i) & \dots & \dot{\psi}(t_i) \\ \vdots & \vdots & \ddots & \vdots \\ \ddot{\zeta}_x^I(t_m) & \ddot{\zeta}_y^I(t_m) & \dots & \dot{\psi}(t_m) \end{bmatrix}.\end{aligned}\quad (15)$$

The vehicle model depends primarily on the physics-informed modeling method with less concentration on the ML technique as it is mainly useful for any discrepancy and abrupt changes in the parametric values. Therefore, a similar assumption to [29] is required, i.e., either the obtained data are prefiltered through relevant frameworks [37,38] or it contains no noise interference. Next, a candidate library function $\Xi(\mathbf{X}_s, \mathbf{U}_s)$ is developed using the data matrices $\mathbf{X}_s$ and $\mathbf{U}_s$, in which selected function is arbitrary, and is either a trigonometric or polynomial function. In general, polynomial functions are a relatively basic choice. Therefore, the complexity of the library can be increased by including trigonometric functions. Nevertheless, both functions and physics-informed knowledge of the vehicle will be utilized to construct the library function:

$$\Xi(\mathbf{X}_s, \mathbf{U}_s) = [\mathbf{1}^T, \mathbf{X}_s^T, (\mathbf{X}_s \otimes \mathbf{X}_s)^T, (\mathbf{X}_s \otimes \mathbf{U}_s)^T, \dots, S_{\mathbf{X}_s^T}, S_{\mathbf{U}_s^T}, \dots], \quad (16)$$

where $\mathbf{X}_s \otimes \mathbf{X}_s$ and $\mathbf{X}_s \otimes \mathbf{U}_s$ are defined as:

$$\begin{aligned}\mathbf{X}_s \otimes \mathbf{X}_s &= \begin{bmatrix} \ddot{\zeta}_x^I \ddot{\zeta}_x^I(t_1) & \ddot{\zeta}_x^I \ddot{\zeta}_y^I(t_1) & \dots & \ddot{\zeta}_y^I \ddot{\zeta}_y^I(t_1) & \dots & \psi^2(t_1) \\ \ddot{\zeta}_x^I \ddot{\zeta}_x^I(t_2) & \ddot{\zeta}_x^I \ddot{\zeta}_y^I(t_2) & \dots & \ddot{\zeta}_y^I \ddot{\zeta}_y^I(t_2) & \dots & \psi^2(t_2) \\ \vdots & \vdots & \dots & \vdots & \dots & \vdots \\ \ddot{\zeta}_x^I \ddot{\zeta}_x^I(t_m) & \ddot{\zeta}_x^I \ddot{\zeta}_y^I(t_m) & \dots & \ddot{\zeta}_y^I \ddot{\zeta}_y^I(t_m) & \dots & \psi^2(t_m) \end{bmatrix}, \\ \mathbf{X}_s \otimes \mathbf{U}_s &= \begin{bmatrix} \ddot{\zeta}_x^I T(t_1) & \ddot{\zeta}_x^I u_{11}(t_1) & \dots & \ddot{\zeta}_y^I u_{11}(t_1) & \dots & \psi u_{13}(t_1) \\ \ddot{\zeta}_x^I T(t_2) & \ddot{\zeta}_x^I u_{11}(t_2) & \dots & \ddot{\zeta}_y^I u_{11}(t_2) & \dots & \psi u_{13}(t_2) \\ \vdots & \vdots & \dots & \vdots & \dots & \vdots \\ \ddot{\zeta}_x^I T(t_m) & \ddot{\zeta}_x^I u_{11}(t_m) & \dots & \ddot{\zeta}_y^I u_{11}(t_m) & \dots & \psi u_{13}(t_m) \end{bmatrix}.\end{aligned}\quad (17)$$

The dynamics of the vehicle model only depend on the few nonlinear terms in practice. The sparse model of $f_k(\cdot)$ in system (10) can be set as follows:

$$f_k(\mathbf{X}_s, \mathbf{U}_s, \chi_k) = \sum_{j=0}^{p_s} \chi_{kj} \Lambda_j(\mathbf{x}_s, \mathbf{u}_s), \quad k = 1, 2, 3, \dots, 12, \quad (18)$$

where $p_s + 1$ represents the number of candidate functions. χ_{kj} and Λ_j are the candidate functions of the j -th column in $\Xi(\mathbf{X}_s, \mathbf{U}_s)$ and the weighted coefficient of $\Lambda_j(\mathbf{x}_s, \mathbf{u}_s)$ related to the k -th state, i.e., $\chi_k = [\chi_{k0}, \chi_{k1}, \chi_{k2}, \dots, \chi_{kp_s}]^T$, respectively. In the third step, sparse optimization is employed to determine a sparse model:

$$\chi_k = \arg \min_{\chi_k} \left\| \dot{\mathbf{X}}_{ks} - \chi_k \Xi(\mathbf{X}_s, \mathbf{U}_s)^T \right\|_2 + \lambda_k \|\chi_k\|_1, \quad (19)$$

where $\dot{\mathbf{X}}_{ks}$ and λ_k denote the k -th row of $\dot{\mathbf{X}}$ and sparsity-promoting hyper-parameter, respectively. The theoretical results for convergences of SINDy framework have been provided in [39]. The computed value of χ_k is stored in sparse matrix Ψ :

$$\Psi = \begin{bmatrix} \chi_{01} & \chi_{02} & \cdots & \chi_{0k} & \cdots & \chi_{0n} \\ \chi_{11} & \chi_{12} & \cdots & \chi_{1k} & \cdots & \chi_{1n} \\ \vdots & \vdots & \ddots & \vdots & \ddots & \vdots \\ \chi_{p1} & \chi_{p2} & \cdots & \chi_{pk} & \cdots & \chi_{pn} \end{bmatrix} = [\chi_1 \chi_2 \cdots \chi_k \cdots \chi_{12}]. \quad (20)$$

The baseline model of the DFAV's dynamics can be written in the following expression:

$$\mathbf{X}_s \approx \Xi(\mathbf{X}_s, \mathbf{U}_s) \Psi. \quad (21)$$

To summarize the entire process, the baseline model is determined, and a grid search is employed to identify the optimal hyper-parameter selection. In grid-search, all the combinations of hyper-parameters are evaluated, and the best available set is chosen. This process is executed one time to find the best set of hyper-parameters for future updates. The baseline model can be expressed as sparse coefficients in χ_0 .

2.2.2. Estimation of Model Divergence

Due to various factors (e.g., flight condition, nonlinearity, turbulence, actuator faults, etc.), the model parameters may vary at different times, resulting in divergence problems. Hence, a prediction framework that can quickly estimate any variation in the vehicle model is needed. In this work, the classical-predictor-correction approach is utilized to detect the variation caused by the aforementioned factors [40]. The predictor step is executed over a time $\bar{\tau}$ during the interval $t, t + \bar{\tau}$ using a valid model at time t . The divergence of the predictor $\hat{\mathbf{x}}_s$ and measured state $\mathbf{x}_s$ is calculated at $t + \tau$ as $\|\mathbf{x}_{div}\| = \|\hat{\mathbf{x}}_s(t + \tau) - \mathbf{x}_s(t + \tau)\|$. The main theme is to determine when the model and the related state measurements diverge faster as opposed to the predicted ones by the dynamics of the system. Among many techniques, the divergence of the trajectory can be expressed by the Lyapunov exponent:

$$\lambda = \lim_{\tau \rightarrow \infty} \lim_{\mathbf{x}_{div}(t_0) \rightarrow 0} \frac{\left\langle \log \left(\frac{\mathbf{x}_{div}(t_0 + \tau)}{\mathbf{x}_{div}(t_0)} \right) \right\rangle}{\tau}, \quad (22)$$

and its inverse sets of fastest time scale [30]. Since the information about the dynamical system for the prediction step, the Lyapunov exponent is computed via tangent space. The requirement for an effective detection necessitates computing tolerance λ_{div} and determining the divergence time. If the prediction horizon by the local Lyapunov exponent and provided time scale are inconsistent, a divergence issue will arise between the measurements and the model. In the start, when the predicted and measured values deviate, then the divergence horizon is defined as:

$$T_{div}(t) = \arg \min_{t_{div}} \|\hat{\mathbf{x}}_s(t + t_{div}) - \mathbf{x}_s(t + t_{div})\| > \|\lambda_{div}\|. \quad (23)$$

Hence, it is assumed that, at time t , the real system and prediction model diverge if the model value and measured time scale differ:

$$\bar{\lambda}_{div}(t) > \frac{\log(\lambda_{div}) - \log(\|\mathbf{x}_{div}(t)\|)}{T_{div}(t)}, \quad (24)$$

where λ_{div} is the maximum value obtained from (22) during the time scale $[t, t + T_{div}]$, & $\bar{\lambda}_{div} = \langle \lambda_{div}(t') \rangle_{t' \in [t, t + T_{div}]}$ with $\lambda_{div}(t) = \max(\lambda_{div_1}, \lambda_{div_2}, \lambda_{div_3}, \dots, \lambda_{div_{12}})$.

2.2.3. Adaptive Model Recovery

The following process is implemented for the quick recovery of model parameters. In the beginning, new data onto the existing sparse data χ_0 are regressed to determine varying parameters. Thereafter, the deletion of terms is identified by executing the sparse regression on the columns of Ξ that in χ_0 corresponds to the nonzero rows. In the presence

of a residual error, a sparse model in the inactive columns of Ξ that corresponds to zero rows in χ_0 is fit for this error. Through this process, new terms may be introduced. This procedure is only performed when there is a divergence issue present.

2.3. Control Objective

By considering the physics-informed model as a nominal system $\tilde{x}_s$ and real system x_s obtained as a result of the physics-informed ML procedure, suppose the aerial vehicle is flying with speed v , and the plane follows a reference trajectory x_{ref} in the presence of different factors (e.g., disturbances, uncertainties, actuator faults). From a control perspective, the tracking error of both control input $e_u = u_s - u_{\text{ref}}$ and aircraft's state $e_x = x_s - x_{\text{ref}}$ needs to be minimized. Similarly, $\tilde{e}_x$ and $\tilde{e}_u$ represent nominal state and control input tracking error, respectively. Hence, the cost function is designed as follows:

$$\begin{aligned} J(\tilde{e}_x(t), \tilde{e}_u(t)) &= \int_t^{t+t_p} \underbrace{\mathcal{S}_c(\tilde{e}_x(\tau|t), \tilde{e}_u(\tau|t))d\tau}_{\text{Stage cost}} + \underbrace{\mathcal{T}_p(\tilde{e}_x(t+t_p|t))}_{\text{Terminal penalty}}, \\ J(\tilde{e}_x(t), \tilde{e}_u(t)) &= \int_t^{t+t_p} \underbrace{\left(\|\tilde{e}_x(\tau|t)\|_Q^2 + \|\tilde{e}_u(\tau|t)\|_P^2 \right) d\tau}_{\text{Stage cost}} + \underbrace{\|\tilde{e}_x(t+t_p|t)\|_R^2}_{\text{Terminal penalty}}, \end{aligned} \quad (25)$$

where $t_p = n_s t_i$ represents the prediction horizon with n_s denoting the predictive step. Moreover, P , Q , and R are the positive definitive weighting matrices in (25).

For the smooth and effective functioning of controller design, some preliminaries are needed before proceeding to the main control framework. Recall the nominal system (11), where only physics-informed modeling is considered.

Assumption 1. It is assumed that $f_s(\tilde{x}_s(t), \tilde{u}_s(t))$ is Lipschitz continuous with $\tilde{u}_s \in \mathbb{U}_{\text{MPC}}$ is locally Lipschitz with a constant $\mathcal{L}$ in $\tilde{x}_s(t)$. Hence, the following condition $\|f_s(\tilde{x}_{s1}, \tilde{u}_s) - f_s(\tilde{x}_{s2}, \tilde{u}_s)\| \leq \mathcal{L} \|\tilde{x}_{s1} - \tilde{x}_{s2}\|$ holds.

Assumption 2. We assume that nominal system (11) is known and differentiable with all the initial states available. Thus, one can use Jacobian linearization at the origin, which is assumed to be stabilizable. With this assumption, a linear state feedback control law $u_s = K\tilde{e}_x$ can be obtained in such a form that is asymptotically stable if no disturbances are present.

To further elaborate on the nature of the control system, Assumption A2 from [16] with an additional condition can be adopted in the following assumption:

Assumption 3. (i) For the nominal tracking error, the terminal controller and terminal region $\mathbb{T}$ are such that, if $\tilde{e}_x(t_l + t_p|t_l) \in \mathbb{T}$, then by applying $\tilde{u}_s(\tau|t_{l+1}) = K\tilde{e}_x(\tau|t_{l+1})$ to the vehicle over the time interval $\tau \in [t_l + t_p, t_{l+1} + t_p)$, it satisfies the following expressions of the locally stabilizing controller:

$$\tilde{e}_x(\tau|t_l) \in \mathbb{T}, \quad \tilde{u}_s = K\tilde{e}_x \in \mathbb{U}_{\text{MPC}}. \quad (26)$$

(ii) the condition related to stage and terminal cost is provided as follows:

$$\dot{\mathcal{T}}_p(\tilde{e}_x(\tau|t_l)) + \mathcal{S}_c(\tilde{e}_x(\tau|t_l), \tilde{e}_u(\tau|t_l)) \leq 0. \quad (27)$$

Remark 1. Assumption 3 in (26) is a standard assumption employed in the MPC community [41]. However, in the current scenario, its usage is only restricted to an ideal situation, and the linear state feedback control law in Assumption 2 is only available if the tracking error lies in the terminal region, which is not always the case in the control system for an aircraft.

Several techniques to determine the terminal region have been developed over the years (e.g., [42–44]). Inspired by [16], a terminal region is defined as:

$$\mathbb{T} = \{\tilde{e}_x : \tilde{\mathcal{F}}_1|\tilde{e}_\xi| + \tilde{\mathcal{F}}_2|\tilde{e}_V| + \tilde{\mathcal{F}}_3|\tilde{e}_\Theta| + \tilde{\mathcal{F}}_4|\tilde{e}_\omega| < 1 - \varkappa\}, \quad (28)$$

where $\varkappa$ can be appropriately selected based on the control design. $\tilde{\mathcal{F}}$ and $\tilde{e}_{(\bullet)}$ are the feedback gains and state error between reference and actual trajectory with parameters satisfying $p_l q_l \leq 0.25$, $\tilde{\mathcal{F}}_l \in \left(\frac{1 \pm \sqrt{1-4p_l q_l}}{p_l q_l}\right)$, $l \in \{1, 2, 3, \dots, 12\}$.

Furthermore, disturbances are also present in the system. Therefore, an assumption is needed to prove the recursive feasibility and stability in the subsequent section.

Assumption 4. It is assumed that turbulence Δ_{ext} may also be present along with Δ_{wind} . Hence, the dynamics of total disturbances acting on the aircraft are defined as:

$$\begin{aligned} \dot{\Delta}_{\text{total}} &= \Delta_{\text{ext}} + \Delta_{\text{wind}}, \\ &= [\Delta_v \ \Delta_\omega]^T + \Delta_{\text{wind}}, \end{aligned} \quad (29)$$

where Δ_v and Δ_ω denote the turbulence acting on the translational and rotational components of the vehicle. It is assumed that the total disturbances are bounded, i.e., $\|\Delta_{\text{total}}\| \leq \mathfrak{J}$, which can be fast-time varying unlike [16].

Furthermore, crucial definitions related to stability proof are given as follows:

Definition 1. [45] The tracking error system is input-to-state stable (ISS), if there exists a $\mathcal{KL}$ function $\Gamma(\cdot, \cdot) : \mathbb{R}_{\geq 0} \times \mathbb{R}_{\geq 0} \rightarrow \mathbb{R}$ & $\mathcal{K}$ function $\Upsilon(\cdot)$ such that $t \geq 0$, it satisfies that

$$\|e_x(t)\| \leq \Gamma(\|e_x(t_0)\|, t) + \Upsilon(\mathfrak{J}). \quad (30)$$

Definition 2. [45] A function $V(\cdot)$ is referred to as ISS-Lyapunov function for tracking error system, if there exist $\mathcal{K}_\infty$ functions $\mathfrak{J}_i(\cdot)$, and $\mathcal{K}$ function $\mathfrak{T}(\cdot)$ such that $\forall e_x \in \mathbb{R}^2$

$$\mathfrak{J}_1(\|e_x(t_l)\|) \leq V(e_x(t_l)) \leq \mathfrak{J}_2(\|e_x(t_l)\|), \quad (31)$$

$$V(e_x(t_{l+1})) - V(e_x(t_l)) \leq -\mathfrak{J}_3(\|e_x(t_l)\|) + \mathfrak{T}(\mathfrak{J}). \quad (32)$$

If the Definitions 1 and 2 satisfy and no disturbances are acting on the aerial vehicle, then the tracking error also vanishes [46].

3. Control Framework

This section explains the presented control approach, illustrated in Figure 4. Reference trajectory $\mathcal{P}$ is employed to generate a particular flight envelop in different scenarios during the time interval $t \in [0, 10]$, which is expressed as:

$$\mathcal{P}(t) = [\xi_{\text{ref}}, V_{\text{ref}}, \eta_{\text{ref}}, \omega_{\text{ref}}]^T. \quad (33)$$

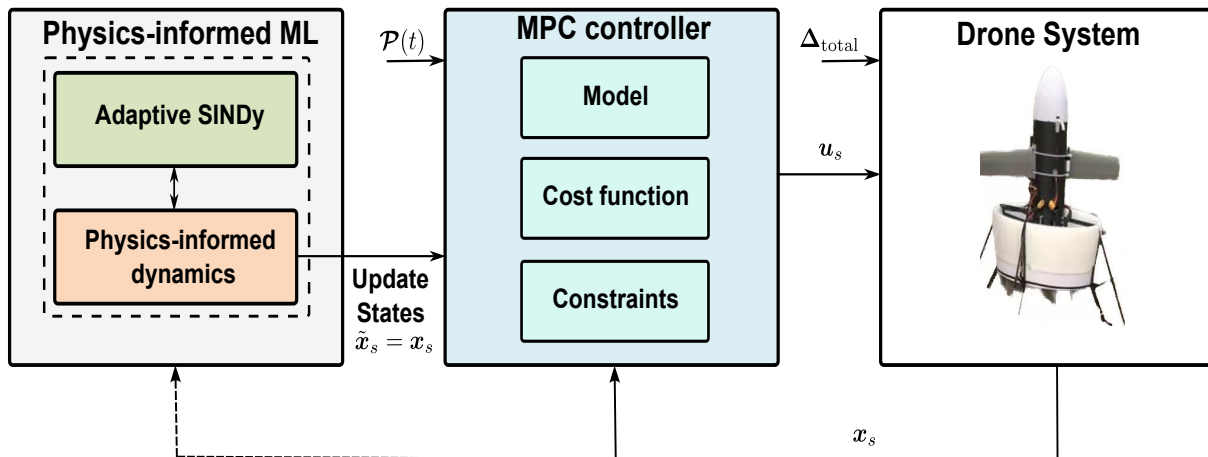


Figure 4. Schematic diagram of the MPC-based control approach using physics-informed ML.

The optimization control problem is solved at a time sequence $\{t_l | l \in \mathbb{N}, t_{l+1} - t_l = t_i\}$:

Problem 1.

$$\min_{\tilde{u}_s(\tau|t_l)} J(\tilde{e}_x(t), \tilde{e}_u(t)), \quad (34a)$$

$$\text{subject to } \tilde{x}_s(t_l|t_l) = x_s(t_l), \quad (34b)$$

$$\dot{\tilde{x}}_s(\tau|t_l) = f_s(\tilde{x}_s(\tau|t_l), \tilde{u}_s(\tau|t_l)), \quad (34c)$$

$$\tilde{u}_s(\tau|t_l) \in \mathbb{U}_{\text{MPC}}, \quad (34d)$$

$$\tilde{x}_s(\tau|t_l) \in \mathbb{X}, \quad (34e)$$

$$\|\tilde{e}_x(\tau|t_l)\| \leq \frac{t_p \times \omega}{\tau - t_l}, \quad (34f)$$

$$\tilde{e}_x(t_l + t_p|t_l) \in \mathbb{T}_\vartheta, \quad (34g)$$

where $\mathbb{T}_\vartheta = \{\tilde{e}_x : \|\tilde{e}_x\| \leq \vartheta\}$ is a robust terminal region, and ω is designed as follows:

$$\omega = \frac{v - \varkappa}{(\mathcal{F}_1^2 + \mathcal{F}_2^2 + \mathcal{F}_3^2 + \mathcal{F}_4^2)^{1/2}}, \quad \vartheta < \omega, \quad (35)$$

Remark 2. For Problem 1, the nominal system (physics-informed dynamics) is updated by the actual state (physics-informed ML model) at each step. Consequently, the provided optimization problem must be solved online. Nevertheless, such an updating framework produces an optimal state with respect to the current one, and since the disturbances are not considered in Problem 1, the designed optimization problem consumes less computational resources.

Problem 1 determines the minimizing sequence over the interval $[t_l, t_l + t_p]$:

$$\tilde{u}_s^*(t_l) = \{\tilde{u}_s^*(t_l|t_l), \tilde{u}_s^*(t_{l+1}|t_l), \dots, \tilde{u}_s^*(t_{l+N}|t_l)\}. \quad (36)$$

Hence, the drone's control signal over interval $[t_l, t_{l+1})$ is defined as:

$$u_s = \tilde{u}_s^*(t_l|t_l), \quad (37)$$

where $\tilde{u}_s^*(t_l|t_l)$ is the initial control action of $\tilde{u}_s^*(t_l)$.

A systematic procedure to apply the control action to the aircraft is provided in Algorithm 1. Nonetheless, if we assume that unknown disturbances are present, a tracking error between the reference and actual trajectory may exist. Furthermore, in order to satisfy

the state constraints in the presence of the disturbances, the error on the upper bound between the actual and predicted nominal state is given as follows:

$$\begin{aligned}
 & \|x_s(t_{l+1}) - \tilde{x}_s^*(t_{l+1}|t_l)\| \\
 &= \|x_s(t_l) + \int_{t_l}^{t_{l+1}} f_s(x_s(\tau), \tilde{u}_s(\tau)) d\tau - \tilde{x}_s^*(t_l|t_l) \\
 &\quad - \int_{t_l}^{t_{l+1}} f_s(\tilde{x}_s^*(\tau|t_l), \tilde{u}_s^*(\tau|t_l)) d\tau\|, \\
 &\leq \mathfrak{J}t_i + v \int_{t_l}^{t_{l+1}} \|x_s(\tau) - \tilde{x}_s^*(\tau|t_l)\| d\tau, \\
 &\leq \mathfrak{J}t_i e^{vt_i}.
 \end{aligned} \tag{38}$$

Algorithm 1: MPC-based control using physics-informed ML

Input : $P, Q, R, \varkappa$, and v

Output: u_s

```

1 while (1) do
2   foreach goal at  $t_{l \geq 0}$  do
3     Determine feedback gains  $\mathcal{F}_i$ .
4     At  $t_l$ , calculate the system tracking error  $e_x(t_l)$ 
5     if  $e_x(t_l) \in \mathbb{T}_\vartheta$  then
6       Apply the control signal  $u_s = K\tilde{e}_x$  (Assumption 2).
7     else
8       At  $t_l$ , initialize the physics-informed state by the physics-informed ML
          model, i.e.,  $\tilde{x}_s(t_l) = x_s(t_l)$  in (34b).
9       Compute Problem 1 to determine  $\tilde{u}_s^*(t_l)$ .
10      Implement the control signal  $u_s(\tau)$  in (37) into the drone over  $[t_l, t_{l+1})$ .
11    end
12  end
13  Update  $t_l \rightarrow t_{l+1}$ 
14 end

```

Theorem 1. Suppose the Assumptions 2 and 3 hold, and the Problem 1 is feasible at initial time t_0 . Then, Problem 1 is feasible for all intervals under Algorithm 1, if the following conditions satisfy:

$$\mathfrak{J} \leq \frac{e^{-vt_p}}{t_i} (\omega - \vartheta), \quad \tilde{\mathcal{F}}t_i \geq \ln(\omega) - \ln(\vartheta), \tag{39}$$

where $\tilde{\mathcal{F}} = \min\{\tilde{\mathcal{F}}_i\}$.

$$\vartheta \geq \frac{\omega(t_p - t_i)}{t_p}. \tag{40}$$

Proof. A feasible control sequence $\tilde{u}_s(\tau|t_{l+1})$ at t_{l+1} is provided as follows:

$$\tilde{u}_s(\tau|t_{l+1}) = \begin{cases} \tilde{u}_s^*(\tau|t_l), & \tau \in [t_{l+1}, t_l + t_p), \\ K\tilde{e}_x(\tau|t_l), & \tau \in [t_l + t_p, t_{l+1} + t_p). \end{cases} \tag{41}$$

As the nominal state is updated by the actual state, i.e., $\tilde{x}_s(t_{l+1}|t_{l+1}) = x_s(t_{l+1})$, and by considering the time interval $\tau \in [t_{l+1}, t_l + t_p)$, we obtain

$$\begin{aligned}
& \|\tilde{\mathbf{x}}_s(\tau|t_{l+1}) - \tilde{\mathbf{x}}_s^*(\tau|t_l)\| \\
&= \|\mathbf{x}_s(t_{l+1}) + \int_{t_{l+1}}^{\tau} f_s(\tilde{\mathbf{x}}_s(s|t_{l+1}), \tilde{\mathbf{u}}_s^*(s|t_l)) ds \\
&\quad - \tilde{\mathbf{x}}_s^*(t_{l+1}|t_l) - \int_{t_{l+1}}^{\tau} f_s(\tilde{\mathbf{x}}_s^*(s|t_l), \tilde{\mathbf{u}}_s^*(s|t_l)) ds\|.
\end{aligned} \tag{42}$$

Utilizing Grönwall–Bellman inequality, we achieve:

$$\|\tilde{\mathbf{x}}_s(\tau|t_{l+1}) - \tilde{\mathbf{x}}_s^*(\tau|t_l)\| \leq \mathfrak{J} t_i e^{v(\tau-t_{l+1}+t_l)}. \tag{43}$$

Employing triangle inequality and substituting $t_l + t_p$ in (43), we have

$$\|\tilde{\mathbf{e}}_x(t_l + t_p|t_{l+1})\| \leq \|\tilde{\mathbf{e}}_x^*(t_l + t_p|t_l)\| + \mathfrak{J} t_i e^{vt_i}. \tag{44}$$

Since the ensuing expressions hold, $\|\tilde{\mathbf{e}}_x^*(t_l + t_p|t_l)\| \leq \vartheta$ and $\mathfrak{J} \leq \frac{e^{-vt_p}}{t_i}(\omega - \vartheta)$. Therefore, one can obtain:

$$\|\tilde{\mathbf{e}}_x(t_l + t_p|t_{l+1})\| \leq \omega, \tag{45}$$

which implies $\tilde{\mathbf{e}}_x(t_l + t_p|t_{l+1}) \in \mathbb{T}$. Next, consider the control input $\mathbf{u}(\tau) = \mathbf{K}\tilde{\mathbf{e}}_x \in \mathbb{U}_{\text{MPC}}$ is directly applied to the drone during $\tau \in [t_l + t_p, t_{l+1} + t_p]$.

$$\begin{aligned}
\frac{d}{d\tau} \|\tilde{\mathbf{e}}_x(\tau|t_{l+1})\|^2 &= -2(\tilde{\mathcal{F}}_1 \tilde{e}_\xi^2 + \tilde{\mathcal{F}}_2 \tilde{e}_V^2 + \tilde{\mathcal{F}}_3 \tilde{e}_\Theta^2 + \tilde{\mathcal{F}}_4 \tilde{e}_\omega^2), \\
&\leq -2\tilde{\mathcal{F}} \|\tilde{\mathbf{e}}_x(\tau|t_{l+1})\|^2.
\end{aligned} \tag{46}$$

By using the comparison principle,

$$\|\tilde{\mathbf{e}}_x(t_{l+1} + t_p|t_{l+1})\| \leq \|\tilde{\mathbf{e}}_x(t_l + t_p|t_{l+1})\| e^{-\tilde{\mathcal{F}} t_i}.$$

Through $\tilde{\mathcal{F}} t_i \geq \ln(\omega) - \ln(\vartheta)$:

$$\|\tilde{\mathbf{e}}_x(t_{l+1} + t_p|t_{l+1})\| \leq \vartheta. \tag{47}$$

This comprehensively ensures that $\tilde{\mathbf{u}}_s(\tau|t_{l+1})$ is able to force the nominal tracking error into the terminal region $\mathbb{T}_\vartheta$, which satisfies the constraint (34g).

To prove the constraint satisfaction of (34f), we initially consider the time interval $\tau \in [t_{l+1}, t + t_p]$. From (43), it follows that:

$$\|\tilde{\mathbf{e}}_x(\tau|t_{l+1})\| \leq \|\tilde{\mathbf{e}}_x^*(\tau|t_l)\| + \mathfrak{J} t_i e^{vt_i}. \tag{48}$$

Because of (39) and $\|\tilde{\mathbf{e}}_x(\tau|t_l)\| \leq \frac{t_p}{\tau - t_l}$, we obtain

$$\|\tilde{\mathbf{e}}_x(\tau|t_{l+1})\| \leq \frac{t_p \times \omega}{\tau - t_l} + (\omega - \vartheta). \tag{49}$$

From (40), we have

$$\omega - \vartheta \leq \frac{t_i \omega}{t_p - t_i} \leq \frac{t_i \omega t_p}{(\tau - t_{l+1})(\tau - t_l)}. \tag{50}$$

Substituting (50) into (49), we achieve

$$\|\tilde{\mathbf{e}}_x(\tau|t_{l+1})\| \leq \frac{t_p \times \omega}{\tau - t_{l+1}}, \tag{51}$$

which comprehensively proves that that state constraint (34f) is satisfied. Next, we consider the time interval $\tau \in [t_k + t_p, t_{l+1} + t_p]$. Through the proof of (45) and (47), and since

$\frac{\omega t_i}{\tau - t_{l+1}} \geq \omega$, $\|\tilde{e}_x(\tau|t_{l+1})\| \leq \frac{t_p \times \omega}{\tau - t_{l+1}}$ is satisfied over $\tau \in [t_k + t_p, t_{l+1} + t_p]$. This completes the proof of constraint satisfaction of (34f). The aforementioned mathematical analysis conclusively proves that the designed candidate solution (41) is a feasible solution. This establishes that Problem 1 is feasible for the entire time and concludes the feasibility proof. $\square$

Theorem 2. *Supposing that all the conditions in Theorem 1 meet, then the aircraft will satisfy constraints in the closed loop control operation.*

Proof. For the entire time interval $t \geq 0$, the constraints satisfied in Theorem 1, then the actual system will ensure constraint satisfaction in the close loop scenario despite the presence of disturbances. $\square$

The following theorem provides the stability analysis of the proposed technique.

Theorem 3. *Suppose the drone is controlled through (37) and by Algorithm 1 with all conditions holding in Theorems 1 and 2, then the tracking error system is ISS if*

$$\underline{q}\vartheta^2 > \frac{1}{2}\mathfrak{I}e^{vt_p}(\omega + \vartheta) + \frac{\bar{q}^2\mathfrak{I}^2t_i}{2v}(e^{2vt_p} - e^{2vt_i}) + \frac{2\bar{q}^2\mathfrak{I}\omega}{\sqrt{2}v}\left(\frac{t_p^2}{t_i} - t_p\right)^{\frac{1}{2}}(e^{2vt_p} - e^{2vt_i})^{\frac{1}{2}}, \quad (52)$$

where $\underline{q} = \min\{q_k\}$ and $\bar{q} = \max\{q_k\}$ with $k = \{1, 2, 3, \dots, 12\}$.

Proof. Choose a Lyapunov function

$$V(e_x(t_l)) = J(\tilde{e}_x^*(t_l), \tilde{e}_u^*(t_l)). \quad (53)$$

Through the Riemann integral principle, a constant r_1 exists such that the subsequent expression satisfies $V(e_x) \geq r_1\mathcal{S}_c(e_x, e_u) \triangleq \mathfrak{I}_1(\|e_x\|)$. From (27), $V(e_x(t_l)) \leq \mathcal{T}_p(e_x(t_l)) + \mathcal{T}_p(e_x(t_l + t_p | t_l))$, $\forall e_x \in \mathbb{T}_\vartheta$. Due to $\mathcal{T}_p(\cdot)$ in the terminal region with respect to time. Hence, $V(e_x(t_l)) \leq 2g(e_x(t_l))$, $\forall e_x \in \mathbb{T}_\vartheta$. As the origin lies in the $\mathbb{T}_\vartheta$, & $2g(e_x(t_l)) \leq \vartheta$ $\forall e_x \in \mathbb{T}_\vartheta$, it satisfies that $2\mathcal{T}_p(e_x(t_l)) \geq \mathbb{T}_\vartheta$. Because of the feasibility of Problem 1, an upper bound $r_2 > \vartheta$ for $V(e_x(t_l))$ exists. Thus, $\mathfrak{I}_2(\|e_x(t_l)\|) = \frac{r_2}{\vartheta}\mathcal{T}_p(e_x(t_l))$ is a $\mathcal{K}_\infty$ function such that $\mathfrak{I}_2(\|e_x(t_l)\|) \geq r_2$, thereby satisfying $\mathfrak{I}_2(\|e_x(t_l)\|) \geq V(e_x(t_l))$. This confirms the existence of $\mathcal{K}_\infty$ functions $\mathfrak{I}_1, \mathfrak{I}_2$ satisfying (31).

Lyapunov function at t_{l+1} with the actual system

$$V(e_x(t_{l+1})) = J(\tilde{e}_x(t_{l+1}), \tilde{e}_u(t_{l+1})), \quad (54)$$

so

$$\begin{aligned} \Delta V &= V(e_x(t_{l+1})) - V(e_x(t_l)) \\ &\leq J(\tilde{e}_x(t_{l+1}), \tilde{e}_u(t_{l+1})) - J(\tilde{e}_x^*(t_l), \tilde{e}_u^*(t_l)) \\ &\triangleq \Delta V_1 + \Delta V_2 + \Delta V_3, \end{aligned} \quad (55)$$

where

$$\begin{aligned} \Delta V_1 &= \int_{t_{l+1}}^{t_l+t_p} \left(\|\tilde{e}_x(\tau|t_{l+1})\|_Q^2 - \|\tilde{e}_x^*(\tau|t_l)\|_Q^2 \right) d\tau, \\ \Delta V_2 &= \int_{t_l+t_p}^{t_{l+1}+t_p} \left(\|\tilde{e}_x(\tau|t_{l+1})\|_Q^2 + \|\tilde{e}_u(\tau|t_{l+1})\|_P^2 \right) d\tau \\ &\quad + \|\tilde{e}_x(t_{l+1}+t_p|t_{l+1})\|_R^2 - \|\tilde{e}_x^*(t_l+t_p|t_l)\|_R^2, \\ \Delta V_3 &= - \int_{t_l}^{t_{l+1}} \left(\|\tilde{e}_x^*(\tau|t_l)\|_Q^2 + \|\tilde{e}_u^*(\tau|t_l)\|_P^2 \right) d\tau. \end{aligned}$$

Let ΔV_1

$$\begin{aligned}\Delta V_1 &\leq \int_{t_{l+1}}^{t_l+t_p} \left(\|\tilde{e}_x(\tau|t_{l+1}) - \tilde{e}_x^*(\tau|t_l)\|_Q \right) \left(\|\tilde{e}_x(\tau|t_{l+1})\|_Q + \|\tilde{e}_x^*(\tau|t_l)\|_Q \right) d\tau \\ &\leq \int_{t_{l+1}}^{t_l+t_p} \left[\bar{q}^2 \mathfrak{I} t_i e^{v(\tau+t_i-t_{l+1})} \times \left(2\|\tilde{e}_x^*(\tau|t_l)\| + \mathfrak{I} t_i e^{v(\tau+t_i-t_{l+1})} \right) \right] d\tau \\ &= \int_{t_{l+1}}^{t_l+t_p} \left[2\bar{q}^2 e^{v(\tau+t_i-t_{l+1})} \|\tilde{e}_x^*(\tau|t_l)\| + \bar{q}^2 e^{2v(\tau+t_i-t_{l+1})} \right] d\tau \\ &\leq \int_{t_{l+1}}^{t_l+t_p} 2\bar{q} e^{v(\tau+t_i-t_{l+1})} \|\tilde{e}_x^*(\tau|t_l)\| d\tau + \frac{\bar{q}^2}{2v} (e^{2vt_p} - e^{2vt_i}).\end{aligned}\quad (56)$$

Applying Hölder inequality to the first term,

$$\Delta V_1 \leq \left(\int_{t_{l+1}}^{t_l+t_p} \|\tilde{e}_x^*(\tau|t_l)\|^2 d\tau \right)^{\frac{1}{2}} \frac{2\bar{q}^2 \mathfrak{I} t_i}{\sqrt{2v}} (e^{2vt_p} - e^{2vt_i})^{\frac{1}{2}} + \frac{\bar{q}^2 \mathfrak{I}^2 t_i^2}{2v} (e^{2vt_p} - e^{2vt_i}). \quad (57)$$

Suppose ΔV_2

$$\begin{aligned}\Delta V_2 &= \int_{t_l+t_p}^{t_{l+1}+t_p} \|\tilde{e}_x(\tau|t_{l+1})\|_Q^2 + \|\tilde{e}_u(\tau|t_{l+1})\|_P^2 d\tau + \|\tilde{e}_x(t_{l+1}+t_p|t_{l+1})\|_R^2 \\ &\quad - \|\tilde{e}_x^*(t_l+t_p|t_l)\|_R^2 + \|\tilde{e}_x(t_l+t_p|t_{l+1})\|_R^2 - \|\tilde{e}_x(t_l+t_p|t_{l+1})\|_R^2.\end{aligned}\quad (58)$$

Integrating (27) from $t_l + t_p$ into $t_{l+1} + t_p$ and substituting it into (58)

$$\begin{aligned}\Delta V_2 &\leq \|\tilde{e}_x(t_l+t_p|t_{l+1})\|_R^2 - \|\tilde{e}_x^*(t_l+t_p|t_l)\|_R^2 \\ &\leq \left(\frac{1}{2} \|\tilde{e}_x(t_l+t_p|t_{l+1}) - \tilde{e}_x^*(t_l+t_p|t_l)\| \right) \\ &\quad \times (\|\tilde{e}_x(t_l+t_p|t_{l+1})\| + \|\tilde{e}_x^*(t_l+t_p|t_l)\|) \\ &\leq \frac{1}{2} \mathfrak{I} \bar{q} e^{vt_p} (\vartheta + \omega).\end{aligned}\quad (59)$$

Consider ΔV_3

$$\begin{aligned}\Delta V_3 &< - \int_{t_l}^{t_{l+1}} \|\tilde{e}_x^*(\tau|t_l)\|_Q^2 d\tau, \\ &\leq -qt_i \vartheta^2.\end{aligned}\quad (60)$$

By the addition of (57), (59) and (60), $\Delta V \triangleq \Delta V_1 + \Delta V_2 + \Delta V_3$ satisfies

$$\begin{aligned}\Delta V &< -qt_i \vartheta^2 + \frac{1}{2} \mathfrak{I} t_i e^{vt_p} (\vartheta + \omega) + \frac{2\bar{q}^2 \mathfrak{I} t_i}{\sqrt{2v}} (e^{2vt_p} - e^{2vt_i})^{\frac{1}{2}} \\ &\quad + \frac{\bar{q}^2 \mathfrak{I}^2 t_i^2}{2v} (e^{2vt_p} - e^{2vt_i}).\end{aligned}\quad (61)$$

From the condition (52), $\Delta V < 0$ satisfies. When $e_x \in \mathbb{T}_\vartheta$, re-assume ΔV_1 and ΔV_3 :

$$\begin{aligned}\Delta V_1 &\leq \int_{t_{l+1}}^{t_l+t_p} 2\bar{q}^2 \mathfrak{I} t_i \vartheta (e^{v\tau-t_{l+2}} d\tau + \frac{\bar{q}^2 \mathfrak{I}^2 t_i^2}{2v} (e^{2vt_p} - e^{2vt_i})) \\ &= \frac{2\bar{q}^2 \mathfrak{I} t_i \vartheta}{v} (e^{vt_p} - e^{vt_i}) + \frac{\bar{q}^2 \mathfrak{I}^2 t_i^2}{2v} (e^{vt_p} - e^{vt_i}).\end{aligned}\quad (62)$$

Because of the decreasing property of $\|\tilde{e}_x(\tau|t_l)\|_Q^2$ in $\mathbb{T}_\vartheta$, then ΔV_3 follows

$$\Delta V_3 \leq -qt_i \|\tilde{e}_x^*(t_{l+1}|t_l)\|^2. \quad (63)$$

From (48), we obtain $\|e_x(t_{l+1})\|^2 \leq \|e_x^*(t_{l+1}|t_l)\|^2 + \mathfrak{J}^2 t_i^2 e^{2vt_i} + 2\vartheta \mathfrak{J} t_i^2 e^{\vartheta t_i}$. As a result,

$$\Delta V_3 \leq -qt_i \|\tilde{e}_x(t_{l+1})\|^2 q \mathfrak{J}^2 t_i^3 e^{2\vartheta \mathfrak{J}} + 2q\vartheta \mathfrak{J} t_i^2 e^{\vartheta t_i}.$$

Consequently, it satisfies that

$$\Delta V \leq -qt_i \|\tilde{e}_x(t_{l+1})\|^2 + \mathfrak{T}(\mathfrak{J}) \quad (64)$$

where

$$\mathfrak{T}(\mathfrak{J}) = \frac{2\bar{q}^2 \mathfrak{J} t_i \vartheta}{v} (e^{vt_p} - e^{vt_i}) + \frac{\bar{q}^2 \mathfrak{J}^2 t_i^2}{2v} (e^{vt_p} - e^{vt_i}) + \frac{1}{2} \mathfrak{J} \mathfrak{T} e^{vt_p} (\vartheta + \omega) + q \mathfrak{J}^2 t_i^3 e^{2vt_i} + 2q\vartheta \mathfrak{J} t_i^2 e^{vt_i}$$

is a $\mathcal{K}$ function w.r.t. $\mathfrak{J}$. Hence, the stability is comprehensively proved. $\square$

4. Comparative Analysis

4.1. Simulation Studies

The parameters of the aerial robot are provided as follows: $m = 1.6$ kg, $I_{xx} = I_{yy} = 0.025$ kg.m², $I_{zz} = 0.005$ kg.m², $l_1 = 0.17$ m, $l_2 = 0.06$ m, $g = 9.8$ m/s². 30% parametric uncertainty for the entire flight is considered during the simulation study. To validate the reliability of the controller, a high-speed $v = 20$ m/s is selected for the comparative analysis. In the cost function, $R = \text{diag}\{0.1 \ 0.1 \ 0.1 \ 0.1 \ 1 \ 1\}$, and the values of q_i and p_i are 1 and 0.25, respectively. In Problem 1, the numerical values of t_p , t_i , ϑ , $\mathcal{F}_i$, ω and $\varkappa$ are chosen as 10 s, 0.05 s, 0.04, 4, 0.08 and 0.2, respectively. λ is selected as 0.02, 0.03, 0.04, 0.042, 0.05, 0.051, 0.06, 0.07, 0.08, 0.085, 0.09, and 0.095. The constraints are $-1 \leq \xi^T \leq 1$, $-2.5 \leq V_I \leq 2.5$, $0 \leq T \leq 4$, $-\pi/4 \leq \Theta \leq \pi/4$, and $-2 \leq \omega_B \leq 2$.

In order to corroborate the control performance, different trajectories are defined that show that the DFAV has the properties of fixed-wing aircraft and helicopters. Suppose the drone is flying with an angular velocity of 2 rad/s. In the first trajectory (65), the aircraft takes off, and after attaining a certain altitude, it performs a bank-turn motion:

$$\begin{aligned} \xi_x &= \sin(2t), \\ \xi_y &= 5 \cos(2t), \\ \xi_z &= 50 - e^{-3t}. \end{aligned} \quad (65)$$

In the second instance, we validate the control performance with a helical 3D trajectory:

$$\begin{aligned} \xi_x &= \sin(2t), \\ \xi_y &= 5 \cos(2t), \\ \xi_z &= 50t - \cos(-2t). \end{aligned} \quad (66)$$

Moreover, since the aerial robot is flying with high speed v , we validate the performance with a straight and level flight (SLF), a kind of horizontal flight phase of DFAVs [14]. During this flight, the drone must keep a constant altitude (e.g., 40 m) during the entire time, while the aircraft should fly at the desired angles, i.e., $\alpha = 5^\circ$ and $\theta = 5^\circ$. In this case, some initial conditions are required to be considered, e.g., $\xi_x(0) = 0$ m, $\xi_z(0) = 39$ m, $\alpha(0) = 4.8^\circ$, and $\theta(0) = 4.8^\circ$.

Three different cases are considered:

1. Case (1): Performance in the presence of parametric uncertainties;
2. Case (2): Performance in the presence of parametric uncertainties and disturbances;
3. Case (3): Performance in the presence of faults, model uncertainties and disturbances.

Before proceeding to the aforementioned case studies, the correlation between flight performance and endurable wind speed is provided. In the previous work based on this aerial vehicle [47], aircraft was flown at various speeds under different weather conditions. From these real-time experiments, it was found that the maximum endurable wind speed is approximately 8 m/s. Therefore, it can be concluded that effective flight performance can be achieved if the wind speed is within the maximum endurable wind speed limit. However,

the flight performance degrades as the wind speed passes the maximum endurable wind speed threshold. If the aircraft keeps flying over this maximum endurable wind speed, it encounters a structural failure, or a severe fault may occur. Nonetheless, the severity of this structural failure or fault depends on the magnitude and the total duration of the wind gust it encounters. To keep in view the contributions and main objective of this article, we deal with the control performance only under three case studies, under calm weather (Case 1), and wind gust is present (Case 2). However, it is assumed that it may not cause any structural faults because the magnitude of the wind gust is mostly under 8 m/s during the entire flight. Lastly, it is considered in the third case study that the partial fault exists associated with the unavailability of the dynamical information of the wings whose probability of occurrence is less than 3%. This occurrence level is uncommon, but to precisely verify the efficacy of the control approach, the worst-case situation (less than 3%) shall be analyzed in the third case study.

4.1.1. Case(1): Trajectory Tracking Response under Model Uncertainties

A comparative analysis of both strategies under model uncertainties is shown in Figure 5. In Figure 5a, the simulation results based on 3D trajectory (65) and (66) are illustrated. Furthermore, the tracking response under model inaccuracies derived from SLF flight is depicted in Figure 5b. Both controllers are primarily constructed on the principle of nominal MPC with similar attributes, so they exhibit adequate tracking performance. Nevertheless, the proposed technique demonstrates effective control performance with fast response and rapid convergence time.

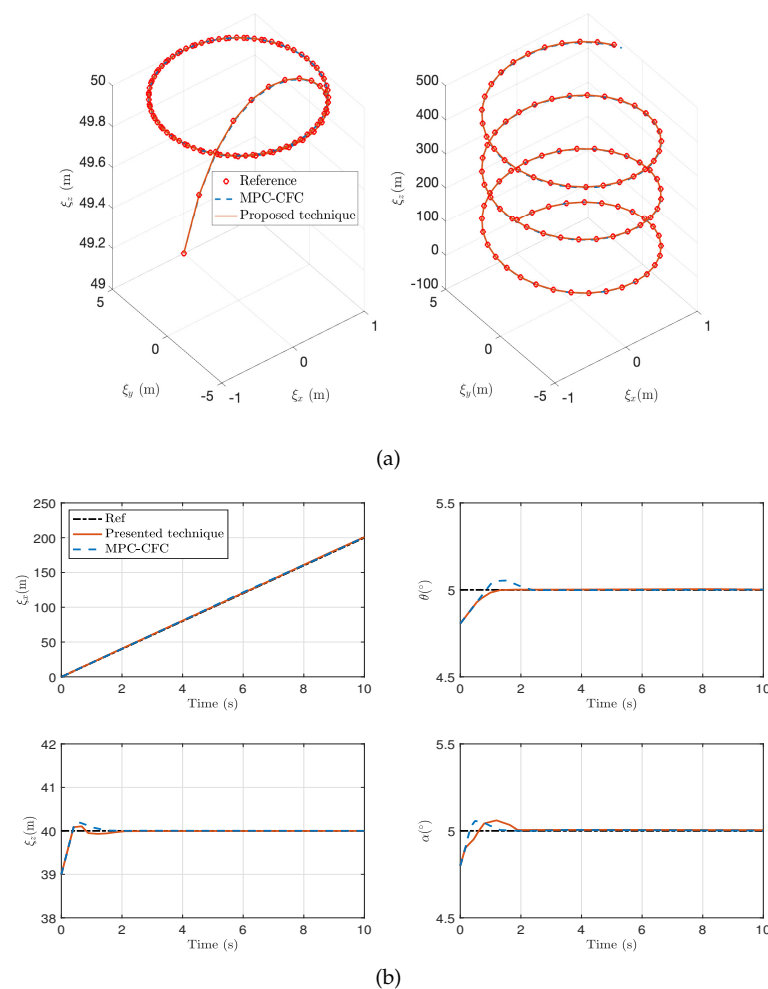


Figure 5. Tracking response under model uncertainties (a) 3D trajectory tracking (b) SLF flight.

4.1.2. Case(2): Trajectory Tracking Response under Uncertainties and Disturbances

Recall disturbance model in (29): the continuous Dryden wind turbulence model MIL-HDBK-1797B is employed using Simulink with default parameters chosen. Due to some minor limitations in this turbulence model, wind components are added, depicted in Figure 6. These wind components are composed of sinusoidal waveform $\Delta_{\text{wind}} = [\sin(8t) \sin(6.01t) \sin(4.02t)]^T$. The trajectory tracking performance under model uncertainties and disturbances are shown in Figure 7. Note that the MPC-CFC performance, as opposed to the developed approach, deteriorates despite having a dedicated DOB mechanism. This conclusively indicates that DOB in MPC-CFC [16,17] is only suitable for slow time-varying disturbances.

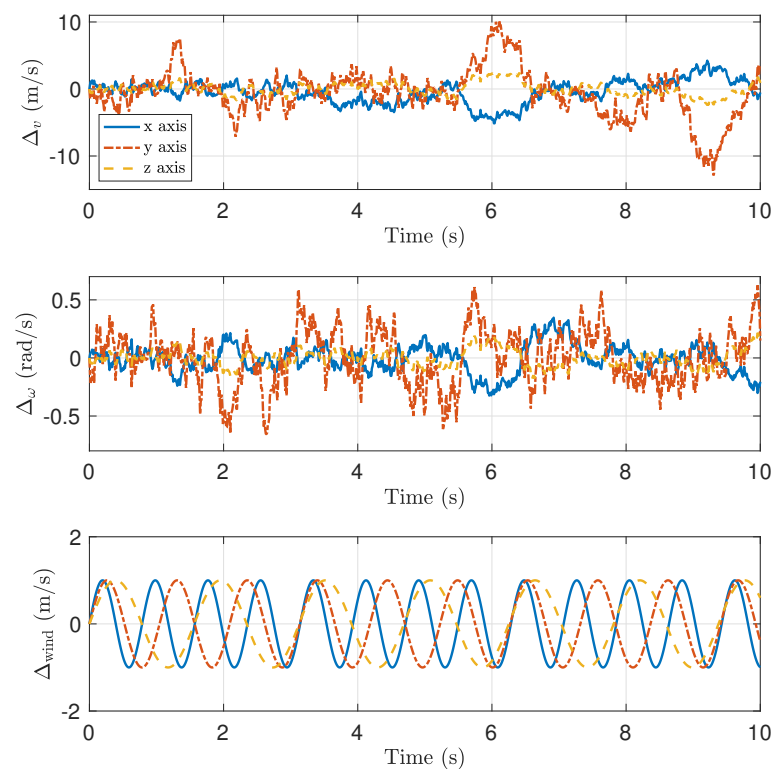


Figure 6. Disturbance profile involving Dryden wind turbulence (Δ_w, Δ_v) and other time-varying wind disturbances Δ_{wind} .

4.1.3. Case(3): Tracking Response in the Presence of Model Uncertainties, Disturbances, and Fault

In the third case study, suppose a fault occurs during the initial few seconds of the simulations due to the presence of a strong wind gust, considered in (29), and this fault is considered as a partial malfunction in one of the wings of the aircraft, whose role is depicted in Figure 2, is now either unavailable or unknown in this case. The tracking performance subject to disturbances, uncertainties, and fault is shown in Figure 8. More specifically, this fault has an adverse effect while controlling the attitude of the aerial platform illustrated in Figure 8b. In this aspect, as adaptive SINDY is equipped to handle and compensate for model divergence and correction, unlike DOB developed offline in [16,17], the presented approach by exploiting the full potential of MPC reveals effective control performance compared to the MPC-CFC technique [16,17].

4.1.4. Performance Analysis Based on Tracking Error and Computations

To analyze the effectiveness of both techniques, mean absolute error (MAE) is utilized and is defined as follows:

$$\text{MAE} = \sum_{j=1}^{n_{\text{obs}}} |e(t_j)| / n_{\text{obs}}, \quad (67)$$

where $e(t_j)$ and n_{obs} are the tracking error in each time step and the sum of a number of observations, respectively. In Table 2, a comparative analysis is illustrated. The trajectory error suggests that the designed approach shows effective tracking performance.

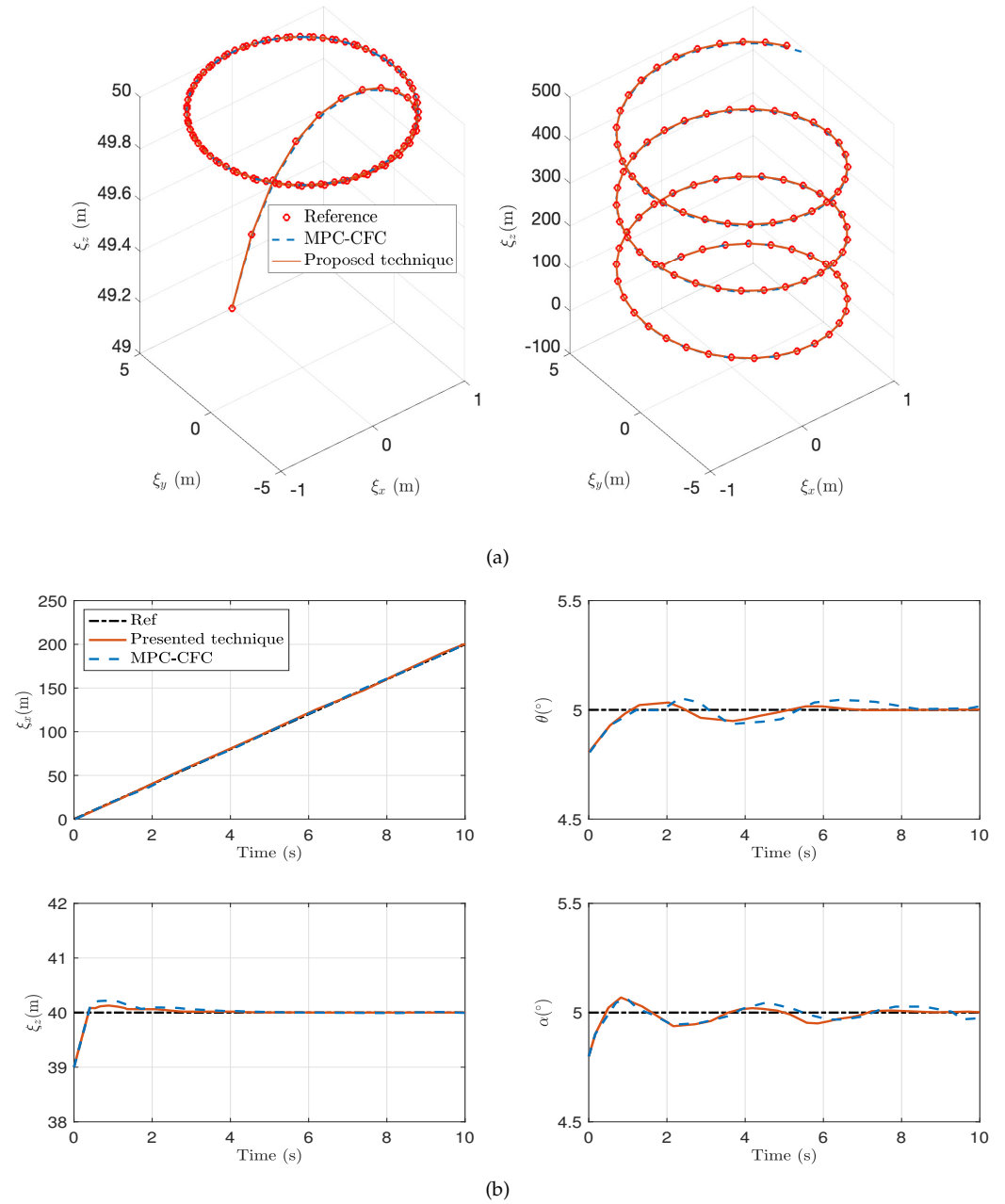


Figure 7. Tracking response under uncertainties and disturbances (a) 3D trajectory tracking; (b) SLF flight.

The numerical simulations are executed on Windows-driven Intel-based dual-core processor (2.2 GHz) with 8 GB of RAM. The optimization Problem 1 is written in MATLAB 2021a via CasADi [48], 3.5.5 version, and computed by NLP solver IPOPT [49], 3.12.3 version. The computational cost of these schemes is determined by the time required to compute the control action during the trajectory tracking response, depicted in Figure 9. Based on these findings, the average normalized computational time for the entire simulation study for the developed algorithm and MPC-CFC strategy is 18.62 ms and 19.51 ms, respectively.

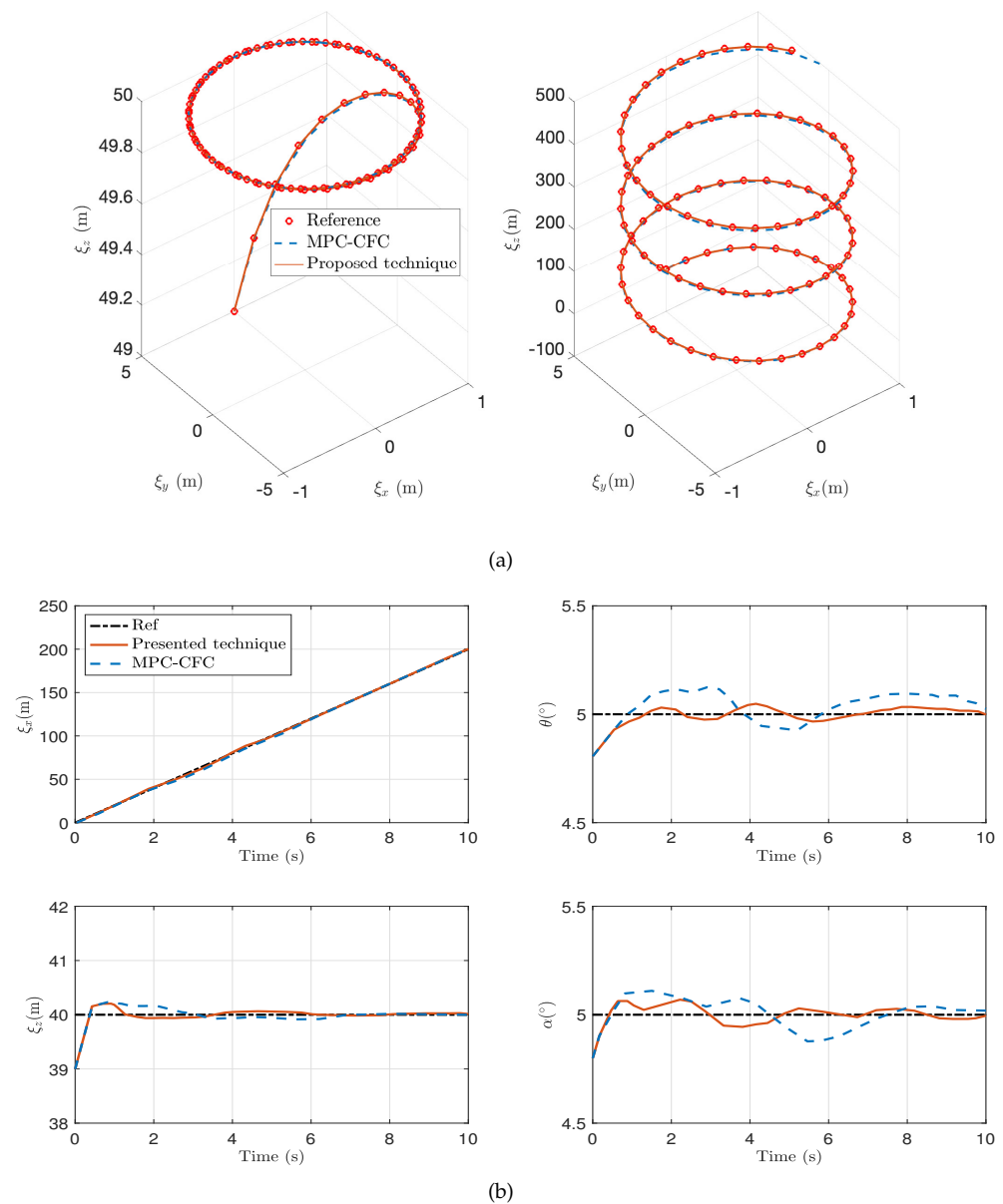


Figure 8. Tracking response under model uncertainties, disturbances, and fault (a) 3D trajectory tracking; (b) SLF flight.

Table 2. Performance analysis based on tracking error (MAE).

Scenario		MPC-CFC	Proposed	Scenario		MPC-CFC	Proposed
ξ_y (m)	Case(1)-Figure 5a	1.5752	0.0016	ξ_z (m)	Case(1)-Figure 5b	0.1711	0.1424
	Case(1)-Figure 5a	0.1569	0.0157		Case(1)-Figure 5a	0.0011	1.09×10^{-6}
	Case(2)-Figure 7a	2.3279	0.0141		Case(2)-Figure 7b	0.0999	0.0987
	Case(2)-Figure 7a	0.2357	0.0235		Case(2)-Figure 7a	0.0016	9.82×10^{-6}
	Case(3)-Figure 8a	2.8687	0.0157		Case(3)-Figure 8b	0.122	0.0782
	Case(3)-Figure 8a	0.4729	0.0314		Case(3)-Figure 8a	0.002	1.091×10^{-5}
ξ_x (m)	Case(1)-Figure 5a	0.03066	0.0003	θ (°)	Case(1)-Figure 5b	0.0482	0.0378
	Case(1)-Figure 5a	0.0325	0.0033		Case(2)-Figure 7b	0.0376	0.0349
	Case(2)-Figure 7a	0.4429	0.0029		Case(3)-Figure 8b	0.0748	0.0284
	Case(2)-Figure 7a	0.0486	0.049	α (°)	Case(1)-Figure 5b	0.0466	0.0460
	Case(3)-Figure 8a	0.5446	0.0033		Case(2)-Figure 7b	0.0371	0.0344
	Case(3)-Figure 8a	0.0966	0.0065		Case(3)-Figure 8b	0.069	0.0389

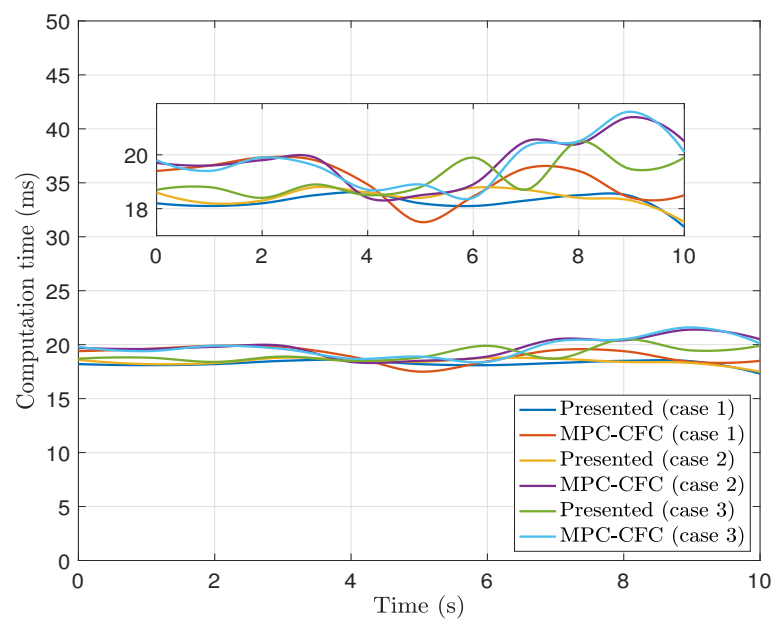


Figure 9. Normalized computational time of both frameworks.

4.2. Discussion

Different case studies have shown that the developed algorithm demonstrates an efficient tracking performance compared to the MPC-CFC technique. It is primarily due to the hybrid modeling approach integrated with the control mechanism that enables the MPC-based controller to perform effectively despite uncertainties, disturbances, and faults. This also confirms our earlier assertion that physics-informed dynamics and DOB in the disturbance rejection MPC approaches (e.g., [16,17]) limited the capability of MPC due to inadequate modeling and ineffective estimation. Nevertheless, the computational complexity may be affected if a large amount of data are utilized online rather than with less data. With regard to the ML technique related to adaptive SINDy, this work has not addressed any shortcomings concerning this method (e.g., noise sensitivity and long-term memory), as they are beyond the scope of this article. Regarding flight robustness, the proposed framework is designed based on a more suitable selection of the terminal region, which yields adequate stability margins as opposed to [16,17]. In spite of all the aforementioned benefits, the presented MPC-based technique using a physics-informed ML model still ensures sub-optimal performance.

5. Conclusions and Future Directions

This paper has presented an MPC-based control design for the ducted fan aircraft utilizing a hybrid modeling scheme known as physics-informed ML. In the beginning, the physics-informed model was derived offline from the drone with sufficient capabilities. Thereafter, an online-based data-driven modeling technique is integrated with physics-informed dynamics to determine the actual model. Afterwards, an MPC-based control algorithm was developed by updating the physics-informal dynamics (nominal system) with real states. ISS stability and recursive feasibility were proven under adequate conditions. Finally, simulations were conducted under three different scenarios starting from challenging (case 1), worse (case 2), and worst-case situation (case 3), which revealed that the developed framework exhibits effective control performance.

For future studies, the worst-case scenario can be utilized as a baseline to effectively construct a more refined fault-tolerant controller. Nevertheless, it is noteworthy that the developed framework is not designed as a fault-tolerant controller. Moreover, the worst-case scenario is only verified under a partial fault occurrence. Furthermore, the designed strategy also ensures sub-optimal performance like [16,17] for aerial applications.

In the end, several future directions can be pursued to improve the efficacy of the proposed strategy:

- Physics-informed modeling: the designed model is basically derived from the principle of the Newton–Euler method. Other mathematical models, such as the Lagrange–Euler approach, can be employed;
- Data-driven ML: the developed ML scheme was inspired by adaptive SINDy [30], where different ways of improving the ML model’s capability can be explored. Moreover, further investigation is required to find an effective technique to enhance computational efficiency with more data and less physics;
- Real-time implementation: future work will involve real-time testing of the presented framework. For this purpose, a more suitable solver can be used for code generation especially developed for real-time embedded optimization.

Author Contributions: Conceptualization, T.M.; methodology, T.M., Z.S. and Z.C.; software, T.M.; validation, T.M.; formal analysis, T.M., Z.S. and H.P.; investigation, T.M.; resources, H.P.; data curation, T.M. and Z.C.; writing—original draft preparation, T.M.; writing—review and editing, T.M., Z.S., Z.C. and H.P.; visualization, T.M., Z.S., Z.C. and H.P.; supervision, H.P.; project administration, H.P.; funding acquisition, T.M. and H.P. All authors have read and agreed to this version of the manuscript.

Funding: This work was supported in part by the Scientific Instruments Development Program of NSFC of China: 615278010, in part by Fundamental Research Funds for the Central Universities, in part by Science and Technology Planning Project of Guangdong, China: 2017B010116005, and in part by 2022 Foreign Expert Program (Foreign Youth Talent Program) of Ministry of Science and Technology of China: QN2022163002.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data presented in this article are available on reasonable request from the authors.

Acknowledgments: The authors would like to thank all the funders who supported this work.

Conflicts of Interest: The authors declare no conflict of interest.

Abbreviations

The following abbreviations are used in this manuscript:

DFAV	Ducted Fan Aerial Vehicle
DOB	Disturbance Observer
MPC	Model Predictive Control
MPC-CFC	MPC-based Compound Flight Control
MAE	Mean Absolute Error
ML	Machine Learning
NN	Neural Network
SINDy	Sparse Identification of Nonlinear Dynamics
SLF	Straight and Level Flight

References

1. Cheng, Z.; Pei, H. Control Effectiveness Enhancement for the Hovering/Cruising Transition Control of a Ducted Fan UAV. *J. Intell. Robot. Syst.* **2022**, *105*. [[CrossRef](#)]
2. Cheng, Z.; Pei, H. Transition Analysis and Practical Flight Control for Ducted Fan Fixed-Wing Aerial Robot: Level Path Flight Mode Transition. *IEEE Robot. Autom. Lett.* **2022**, *7*, 3106–3113. [[CrossRef](#)]
3. Cheng, Z.; Pei, H. Flight Transition Control for Ducted Fan UAV with Saturation on Control Surfaces. In Proceedings of the 2021 International Conference on Unmanned Aircraft Systems (ICUAS), Athens, Greece, 5–18 June 2021; pp. 439–446. [[CrossRef](#)]
4. Marconi, L.; Naldi, R.; Gentili, L. Modelling and control of a flying robot interacting with the environment. *Automatica* **2011**, *47*, 2571–2583. [[CrossRef](#)]
5. Naldi, R.; Macchelli, A.; Mimmo, N.; Marconi, L. Robust Control of an Aerial Manipulator Interacting with the Environment. *IFAC-PapersOnLine* **2018**, *51*, 537–542. [[CrossRef](#)]

6. Marconi, L.; Naldi, R. Control of Aerial Robots: Hybrid Force and Position Feedback for a Ducted Fan. *IEEE Control Syst. Mag.* **2012**, *32*, 43–65. [\[CrossRef\]](#)
7. Naldi, R.; Torre, A.; Marconi, L. Robust Control of a Miniature Ducted-Fan Aerial Robot for Blind Navigation in Unknown Populated Environments. *IEEE Trans. Control Syst. Technol.* **2015**, *23*, 64–79. [\[CrossRef\]](#)
8. Roberts, A.; Tayebi, A. Adaptive position tracking of VTOL UAV. *IEEE Trans. Robot.* **2011**, *27*, 129–142. [\[CrossRef\]](#)
9. Manzoor, T.; Xia, Y.; Ali, Y.; Hussain, K. Flight control techniques and classification of ducted fan aerial vehicles. *Kongzhi Lilun Yu Yingyong/Control Theory Appl.* **2022**, *39*, 201–221. [\[CrossRef\]](#)
10. Hua, M.; Hamel, T.; Morin, P.; Samson, C. Introduction to feedback control of underactuated VTOL vehicles: A review of basic control design ideas and principles. *IEEE Control Syst. Mag.* **2013**, *33*, 61–75. [\[CrossRef\]](#)
11. Eren, U.; Prach, A.; Kocer, B.; Rakovic, S.V.; Kayacan, E.; Acikmese, B. Model Predictive Control in Aerospace Systems: Current State and Opportunities. *J. Guid. Control. Dyn.* **2017**, *40*, 1541–1566. [\[CrossRef\]](#)
12. Banazadeh, A.; Emami, S.A. Control effectiveness investigation of a ducted-fan aerial vehicle using model predictive controller. In Proceedings of the 2014 International Conference on Advanced Mechatronic Systems, Kumamoto, Japan, 10–12 August 2014; pp. 532–537. [\[CrossRef\]](#)
13. Emami, A.; Banazadeh, A. Robustness investigation of a ducted-fan aerial vehicle control, using linear, adaptive, and model predictive controllers. *Int. J. Adv. Mechatron. Syst.* **2015**, *6*, 108–117. [\[CrossRef\]](#)
14. Manzoor, T.; Xia, Y.; Zhai, D.H.; Ma, D. Trajectory tracking control of a VTOL unmanned aerial vehicle using offset-free tracking MPC. *Chin. J. Aeronaut.* **2020**, *33*, 2024–2042. [\[CrossRef\]](#)
15. Emami, A.; Rezaeizadeh, A. Adaptive model predictive control-based Attitude and Trajectory Tracking of a VTOL Aircraft. *IET Control Theory Appl.* **2018**, *12*, 2031–2042. [\[CrossRef\]](#)
16. Manzoor, T.; Sun, Z.; Xia, Y.; Ma, D. MPC based compound flight control strategy for a ducted fan aircraft. *Aerosp. Sci. Technol.* **2020**, *107*, 106264. [\[CrossRef\]](#)
17. Manzoor, T.; Pei, H.; Cheng, Z. Composite observer-based robust model predictive control technique for ducted fan aerial vehicles. *Nonlinear Dyn.* **2022**. [\[CrossRef\]](#)
18. Hewing, L.; Wabersich, K.P.; Menner, M.; Zeilinger, M.N. Learning-Based Model Predictive Control: Toward Safe Learning in Control. *Annu. Rev. Control. Robot. Auton. Syst.* **2020**, *3*, 269–296. [\[CrossRef\]](#)
19. Brunke, L.; Greeff, M.; Hall, A.W.; Yuan, Z.; Zhou, S.; Panerati, J.; Schoellig, A.P. Safe Learning in Robotics: From Learning-Based Control to Safe Reinforcement Learning. *Annu. Rev. Control. Robot. Auton. Syst.* **2022**, *5*, 411–444. [\[CrossRef\]](#)
20. Kaheman, K.; Kaiser, E.; Strom, B.; Kutz, J.N.; Brunton, S.L. Learning Discrepancy Models From Experimental Data. *arXiv* **2019**, arXiv:1909.08574. [\[CrossRef\]](#)
21. Brunton, S.L.; Kutz, J.N. *Data-Driven Science and Engineering: Machine Learning, Dynamical Systems, and Control*; Cambridge University Press: Cambridge, UK, 2019. [\[CrossRef\]](#)
22. Brunton, S.L.; Nathan Kutz, J.; Manohar, K.; Aravkin, A.Y.; Morgansen, K.; Klemisch, J.; Goebel, N.; Buttrick, J.; Poskin, J.; Blom-Schieber, A.W.; et al. Data-Driven Aerospace Engineering: Reframing the Industry with Machine Learning. *AIAA J.* **2021**, *59*, 2820–2847. [\[CrossRef\]](#)
23. Zhang, W.; Shen, J.; Ye, X.; Zhou, S. Error model-oriented vibration suppression control of free-floating space robot with flexible joints based on adaptive neural network. *Eng. Appl. Artif. Intell.* **2022**, *114*, 105028. [\[CrossRef\]](#)
24. Hosseini, S.; Poormirzaee, R.; Hajihassani, M. Application of reliability-based back-propagation causality-weighted neural networks to estimate air-overpressure due to mine blasting. *Eng. Appl. Artif. Intell.* **2022**, *115*, 105281. [\[CrossRef\]](#)
25. Floriano, B.R.; Vargas, A.N.; Ishihara, J.Y.; Ferreira, H.C. Neural-network-based model predictive control for consensus of nonlinear systems. *Eng. Appl. Artif. Intell.* **2022**, *116*, 105327. [\[CrossRef\]](#)
26. Park, B.S.; Yoo, S.J. Quantized-communication-based neural network control for formation tracking of networked multiple unmanned surface vehicles without velocity information. *Eng. Appl. Artif. Intell.* **2022**, *114*, 105160. [\[CrossRef\]](#)
27. Kaiser, E.; Kutz, J.N.; Brunton, S.L. Sparse identification of nonlinear dynamics for model predictive control in the low-data limit. *Proc. R. Soc. A: Math. Phys. Eng. Sci.* **2018**, *474*, 20180335. [\[CrossRef\]](#) [\[PubMed\]](#)
28. Brunton, S.L.; Proctor, J.L.; Kutz, J.N. Discovering governing equations from data by sparse identification of nonlinear dynamical systems. *Proc. Natl. Acad. Sci. USA* **2016**, *113*, 3932–3937. [\[CrossRef\]](#) [\[PubMed\]](#)
29. Cao, R.; Lu, Y.; He, Z. System identification method based on interpretable machine learning for unknown aircraft dynamics. *Aerosp. Sci. Technol.* **2022**, *126*, 107593. [\[CrossRef\]](#)
30. Quade, M.; Abel, M.; Nathan Kutz, J.; Brunton, S.L. Sparse identification of nonlinear dynamics for rapid model recovery. *Chaos: Interdiscip. J. Nonlinear Sci.* **2018**, *28*, 063116. [\[CrossRef\]](#)
31. Karniadakis, G.E.; Kevrekidis, I.G.; Lu, L.; Perdikaris, P.; Wang, S.; Yang, L. Physics-informed machine learning. *Nat. Rev. Phys.* **2021**, *3*, 422–440. [\[CrossRef\]](#)
32. Arnold, F.; King, R. State-space modeling for control based on physics-informed neural networks. *Eng. Appl. Artif. Intell.* **2021**, *101*, 104195. [\[CrossRef\]](#)
33. Zobeiry, N.; Humfeld, K.D. A physics-informed machine learning approach for solving heat transfer equation in advanced manufacturing and engineering applications. *Eng. Appl. Artif. Intell.* **2021**, *101*, 104232. [\[CrossRef\]](#)
34. Shen, S.; Lu, H.; Sadoughi, M.; Hu, C.; Nemani, V.; Thelen, A.; Webster, K.; Darr, M.; Sidon, J.; Kenny, S. A physics-informed deep learning approach for bearing fault detection. *Eng. Appl. Artif. Intell.* **2021**, *103*, 104295. [\[CrossRef\]](#)

35. Liu, X.; Peng, W.; Gong, Z.; Zhou, W.; Yao, W. Temperature field inversion of heat-source systems via physics-informed neural networks. *Eng. Appl. Artif. Intell.* **2022**, *113*, 104902. [[CrossRef](#)]
36. Nascimento, R.G.; Fricke, K.; Viana, F.A. A tutorial on solving ordinary differential equations using Python and hybrid physics-informed neural network. *Eng. Appl. Artif. Intell.* **2020**, *96*, 103996. [[CrossRef](#)]
37. Ahnert, K.; Abel, M. Numerical differentiation of experimental data: Local versus global methods. *Comput. Phys. Commun.* **2007**, *177*, 764–774. [[CrossRef](#)]
38. Chartrand, R. Numerical Differentiation of Noisy, Nonsmooth Data. *Int. Sch. Res. Netw.* **2011**, *2011*, 1023–1033. [[CrossRef](#)]
39. Zhang, L.; Schaeffer, H. On the Convergence of the SINDy Algorithm. *Multiscale Model. Simul.* **2019**, *17*, 948–972. [[CrossRef](#)]
40. Gershenfeld, N.A. *The Nature of Mathematical Modeling*; Cambridge University Press: Cambridge, UK, 1999. [[CrossRef](#)]
41. Xue, R.; Dai, L.; Huo, D.; Xie, H.; Sun, Z.; Xia, Y. Compound tracking control based on MPC for quadrotors with disturbances. *J. Frankl. Inst.* **2022**, *359*, 7992–8013. [[CrossRef](#)]
42. Chen, H.; Allgöwer, F. A Quasi-Infinite Horizon Nonlinear Model Predictive Control Scheme with Guaranteed Stability. *Automatica* **1998**, *34*, 1205–1217. [[CrossRef](#)]
43. Althoff, M.; Stursberg, O.; Buss, M. Reachability analysis of nonlinear systems with uncertain parameters using conservative linearization. In Proceedings of the 47th IEEE Conference on Decision and Control, Cancun, Mexico, 9–11 December 2008; pp. 4042–4048. [[CrossRef](#)]
44. Sun, Z.; Xia, Y. Receding horizon tracking control of unicycle-type robots based on virtual structure. *Int. J. Robust Nonlinear Control* **2016**, *26*, 3900–3918. [[CrossRef](#)]
45. Sontag, E.D., Input to State Stability: Basic Concepts and Results. In *Nonlinear and Optimal Control Theory: Lectures Given at the C.I.M.E. Summer School Held in Cetraro, Italy June 19–29, 2004*; Springer: Berlin/Heidelberg, Germany, 2008; pp. 163–220. [[CrossRef](#)]
46. Sun, Z.; Dai, L.; Liu, K.; Xia, Y.; Johansson, K.H. Robust MPC for tracking constrained unicycle robots with additive disturbances. *Automatica* **2018**, *90*, 172–184. [[CrossRef](#)]
47. Cheng, Z.; Pei, H.; Li, S. Neural-Networks Control for Hover to High-Speed-Level-Flight Transition of Ducted Fan UAV With Provable Stability. *IEEE Access* **2020**, *8*, 100135–100151. [[CrossRef](#)]
48. Andersson, J.A.E.; Gillis, J.; Horn, G.; Rawlings, J.B.; Diehl, M. CasADi—A software framework for nonlinear optimization and optimal control. *Math. Program. Comput.* **2019**, *11*, 1–36. [[CrossRef](#)]
49. Wächter, A.; Biegler, L.T. On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming. *Math. Program.* **2006**, *106*, 25–57. [[CrossRef](#)]

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.