Online Dynamic Model Calibration for Reliable Control of Quadrotor Based on Factor Graph Optimization

PEIWEN YANG
Hong Kong Polytechnic University, Hong Kong, China

WEISONG WEN, Member, IEEE
Hong Kong Polytechnic University, Hong Kong, China

SHIYU BAI, Member, IEEE
Hong Kong Polytechnic University, Hong Kong, China

JIAHAO HU
Hong Kong Polytechnic University, Hong Kong, China

***Abstract*—Precise dynamic model calibration is essential in achieving reliable control of unmanned aerial vehicles (UAV). However, the existing methods tend to use simplified dynamic models and cannot adapt to variations of the dynamic model. To fill this gap, this paper explores an online dynamic model calibration (ODMC) method for the quadrotor based on factor graph optimization (FGO). First, the dynamic model is derived with more error terms, such as the *center of gravity (CoG)*, *aerodynamic drag*, *horizontal velocity-induced thrust (HVT) disturbance*, and *viscous effect*. Moreover, the IMU-body extrinsic parameters and the gravity vector are also identified online. Second, the pose is estimated based on the visual-inertial odometry (VIO) together with the dynamic model parameters. The proposed method is validated using open-source and self-collected datasets. The evaluation demonstrates that the ODMC performs even better than the state-of-the-art data-driven methods (DDM) in force and torque prediction when the maximum velocity is less than 9 m/s. Furthermore, the effectiveness of the proposed method is validated by applying the estimated model in the differential-flatness-based controller (DFBC). Finally, we open-source code and datasets at https://github.com/RoboticsPolyu/QUAD_DYN to benefit the research community.**

Authors' addresses: the authors are with the Department of Aeronautical and Aviation Engineering, Hong Kong Polytechnic University, Hong Kong (e-mail: peiwen1.yang@connect.polyu.hk; welson.wen@polyu.edu.hk; shiyu.bai@polyu.edu.hk; jiaha.hu@polyu.edu.hk; hc.man@polyu.edu.hk). (Corresponding author: Weisong Wen.)

LIST OF SYMBOLS

Symbol	Remarks
$\{I\}$, $\{w\}$, and $\{b\}$	The IMU frame $\{I\}$, the world frame $\{w\}$, and the body frame $\{b\}$
$\mathbf{x}_I = \{\mathbf{p}_I^w, \mathbf{R}_I^w, \mathbf{v}_I^w, \boldsymbol{\omega}_I\}$	$\{I\}$'s state. $\mathbf{p}_I^w \in \mathbb{R}^3$, $\mathbf{R}_I^w \in \mathbb{R}^{3\times3}$ are translation vector and rotation matrix from $\{I\}$ to $\{w\}$, $\mathbf{v}_I^w \in \mathbb{R}^3$ is IMU velocity in $\{w\}$, and $\boldsymbol{\omega}_I \in \mathbb{R}^3$ is angular velocity in $\{I\}$
$\mathbf{T}_b, \mathbf{M}_b$	Resultant thrust $\mathbf{T}_b \in \mathbb{R}^3$ and resultant torque $\mathbf{M}_b \in \mathbb{R}^3$ in $\{b\}$
$\mathbf{P}$	Covariance matrix
$\mathbf{d} = \left[d^1, d^2, d^3\right]^{\mathrm{T}}$	$\mathbf{d} \in \mathbb{R}^3$, the drag coefficients
$\mathbf{b} = \left[b^1, b^2, b^3\right]^{\mathrm{T}}$	$\mathbf{b} \in \mathbb{R}^3$, the viscous coefficients
$\mathbf{I}_b = \mathrm{diag}\left(\left[I_b^1, I_b^2, I_b^3\right]^{\mathrm{T}}\right)$	$\mathbf{I}_b \in \mathbb{R}^{3\times3}$, the moment of inertia
$\mathbf{p}_M^b \in \mathbb{R}^3$	Center of gravity (CoG)'s position in $\{b\}$
$\mathbf{R}_G^w \in \mathbb{R}^{3\times3}$	Gravity's rotation matrix in $\{w\}$
k_h	Horizontal velocity-induced thrust (HVT) coefficient
g	Gravity constant
m_b	UAV's mass
k_f	Rotor's aerodynamic thrust coefficient
k_m	Rotor's aerodynamic torque coefficient
$\mathbf{p}_{r_j}^b \in \mathbb{R}^3$	j-th rotor's position in $\{b\}$
$\mathbf{x}_e = \left\{\mathbf{p}_b^I, \mathbf{R}_b^I\right\}$	Extrinsic parameters, $\mathbf{p}_b^I$ is the position of $\{b\}$ in $\{I\}$, $\mathbf{R}_b^I$ is the rotation from $\{b\}$ to $\{I\}$
ω_{r_j}	Rotating velocity of the j-th rotor

I. INTRODUCTION

As urban scenarios increasingly adopt unmanned aerial vehicle (UAV) applications, reliability has become a paramount concern [1]. A higher precision method for online identification of dynamic models is crucial for the reliability of UAV control. However, the precise identification of the UAV dynamic model is challenging. One of the primary issues is the presence of unacceptable model errors using the simplified dynamic model in the existing work. For example, the aerodynamic drag [2, 3], which is usually unmodelled, has a significant effect on the accuracy of quadrotor trajectory tracking. In addition, for certain UAV applications, such as UAV delivery, the dynamic model varies [4-6]. This variability is subject to changes in the UAV's payload, which can result in the gravity center and geometric center not coinciding. In short, the existing dynamic model identification methods restricted the adaptability of UAVs in changing environments or unforeseen circumstances. To address these challenges, an online dynamic model calibration (ODMC) method of quadrotors for reliability is highly expected by incorporating more dynamic effects.

Why do we need an explicit dynamic model of UAV? First, the dynamic model can compensate for the positioning after faults of external sensors occur [7]. For example, the dynamic model can provide effective information to compensate for positioning, during the GNSS signal blockage [8-10]. The works in [11, 12] exploited a relative motion constraint combining the quadrotor dynamic in a pre-integrated residual, implementing a visual-inertial-dynamic odometry system. With the help of the relative constraints coming from the quadrotor dynamic model, the positioning performance is improved. However, this method relied on the accuracy of the quadrotor dynamic model. Secondly, a precise dynamic model is vital in model predictive control (MPC) [13-16], disturbance observer-based control [17], and fault-tolerant control methods [16, 18, 19] for achieving more agile and safer maneuvers. The accuracy of these control methods depends on the accuracy of the dynamics model. Finally, the state-of-the-art data-driven methods (DDM) [12, 20, 21] revealed that the DDM based on a prior quadrotor dynamic model surpassed pure learning methods in autonomous racing. For example, in [21], deep reinforcement learning controllers adopted a dynamics model that combines the thrust and torque of the explicit dynamics model and the predicted residuals. In short, a high-accuracy quadrotor dynamic model plays an essential role in safety-assured and high-performance UAV positioning and control.

Existing works on UAV dynamic model calibration: on the one hand, flight-independent dynamic model calibration depends on specific calibration instruments and equipment [22-25]. On the other hand, the quadrotor's dynamic model can be inferred from flight data [26, 27]. In [7], the aerodynamics of small fixed-wing drones can be identified using inertial measurements from flight data for model-based navigation. However, these offline methods cannot infer dynamic parameters in real-time control systems. Therefore, in recent years, some works handled online dynamic model calibration based on the extended Kalman filter (EKF) [28, 29] or unknown wind estimation by particle filter [30]. In [28], an online quadrotor identification algorithm tightly fused visual, inertial, and aerodynamic information within a multi-state constraint Kalman filter (MSCKF). However, those methods simplified the dynamic models, which assumed that thrust is proportional to the square of rotor velocity [31] and didn't consider other mechanical effects such as drag, horizontal velocity-induced thrust (HVT), center of gravity (CoG) drift, viscous effect, etc. For higher accuracy, many works have considered other dynamic effects. For instance, the displacement of the CoG in the body frame was calibrated for the more precise torque model [32]. Moreover, the drag force and drag torque were modeled, and parameters of quadrotor dynamics were identified by a wind tunnel experiment [33]. Additionally, some works [34-36] estimated external forces based on simplified models. In addition to EKF, the recent achievements in factor graph optimization (FGO) for mobile robotics state estimation [37-40] have demonstrated its capability to reduce nonlinear errors. Therefore, FGO is a promising approach for highly nonlinear dynamic calibration. Diverging from explicit model-centric methodologies, some prevalent data-driven methods [21], such as NeuroBEM [20] and HDVIO [12], were proposed to learn the thrust and torque of a quadrotor, which combined a first-principle model with a data-driven component. Nevertheless, those data-driven approaches still relied on a prior dynamic model. In addition, these DDMs lack the capability for parameter updates in real-time flight. Therefore, further research is highly expected on online precise dynamic model calibration.

As illustrated in Fig 1, this paper proposes an online dynamic model calibration framework based on FGO for reliable control. First, the proposed factor graph combines visual inertia odometry (VIO) factors [41, 42] and online dynamic model factors (ODMF) for state estimation. In addition, actuator model calibration outputs the actuator's aerodynamic torque coefficient. The proposed ODMC is evaluated by Blackbird UAV datasets [43], NeuroBEM UAV datasets [20], and self-collected datasets. Finally, the dynamic parameters are used in (1) force and torque prediction; and (2) the differential-flatness-based controller (DFBC) [44]; to verify the efficiency of the identified dynamic model. The experimental results demonstrate the improved trajectory tracking performance after compensating for aerodynamic drag, CoG, and HVT by comparing with existing methods.

Specifically, the main contributions of this paper include:

1) *An online quadrotor dynamic model calibration framework based on FGO is proposed*. The improved dynamic model considers more effects, such as the *center of gravity (CoG), aerodynamic drag, horizontal velocity-induced thrust (HVT), and viscous effect*. In addition, the dynamic model parameters are optimized together with the quadrotor's pose, gravity rotation, and extrinsic parameters.
2) *An actuator model is proposed that can map motor input signals into the rotor's rotational velocity*. This paper extracts actuator's aerodynamic torque coefficient, eliminating the issue that cannot be observed jointly with the z-axis inertia of the UAV's moment.
3) *Extensive validation is conducted, which includes thrust prediction, torque prediction, and trajectory tracking experiments.* Calibration experiments on open-source datasets and ours are conducted to validate the accuracy of thrust and torque prediction. The torque can still be accurately predicted after changing the center of gravity. Then, *the quadrotor dynamic model is applied to the controller* to verify the trajectory-tracking performance.

The structure of the paper is as follows. In section II, the dynamic modeling of the quadrotor is given. Section

III introduces the proposed ODMC method. In Section VI, experiments are conducted to evaluate the performance of the proposed method. Section V provides a detailed discussion of the results. Finally, the conclusion and future work are summarized in Section VI. Moreover, the Jacobian matrices are listed in Appendix B.

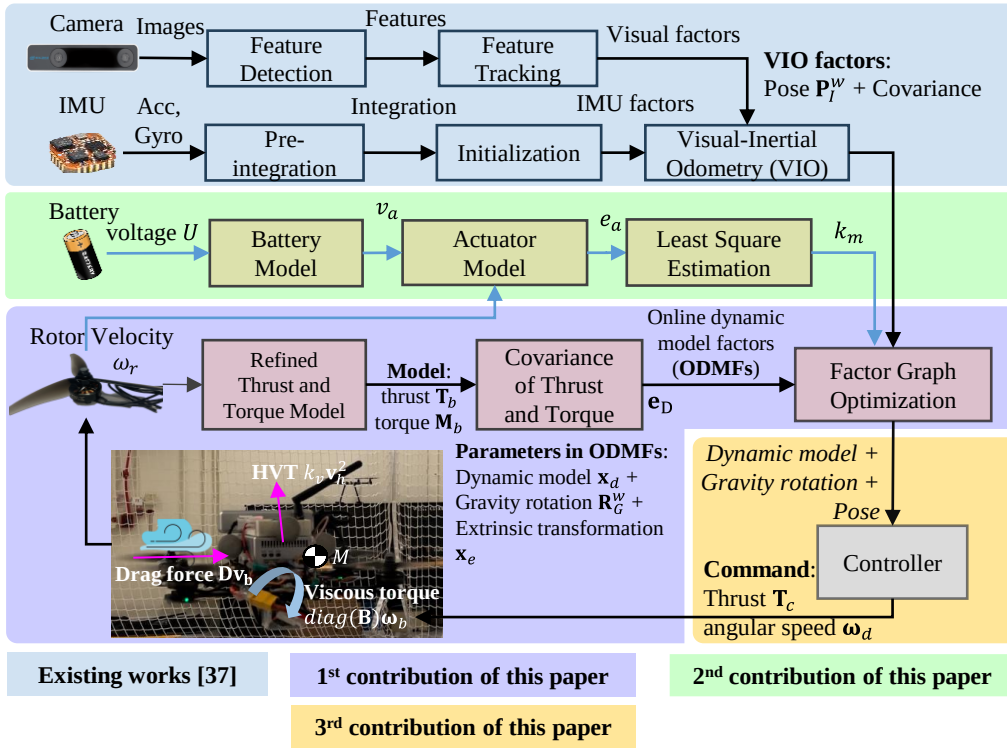


Fig. 1. Overview of the proposed online dynamic model calibration method.

II. MODELING

Multi-rotor UAVs rely on the aerodynamic thrust and torque generated by their rotors to achieve motion control [45]. The rotor's thrust is proportional to the square of the rotational velocity of the rotor [46, 47]. This paper adopts an improved dynamic model of the quadrotor, which additionally considers dynamic effects, such as CoG, HVT, aerodynamic drag, viscous effect, and actuator dynamics [48].

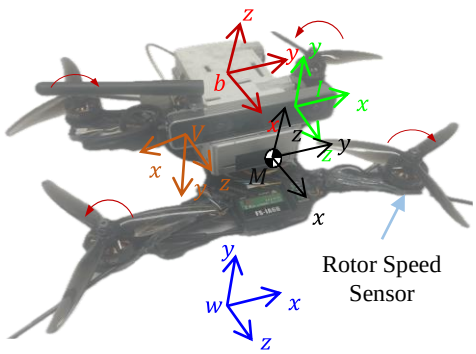


Fig. 2. The frame of the quadrotor: body (geometric center) frame $\{b\}$, IMU frame $\{I\}$, camera frame $\{V\}$, CoG frame $\{M\}$, and world frame $\{w\}$.

As shown in Fig 2, all frames are defined, and the CoG frame and the body frame have the same attitude. *The variables* $\boldsymbol{\chi} = \{\mathbf{x}_I^1, \ldots, \mathbf{x}_I^N, \mathbf{x}_d, \mathbf{x}_e\}$ *to be estimated include the following:*

1) N quadrotor's states $\mathbf{x}_I^i, i \in \{1, 2, \ldots, N\}$ in a fixed time window, where $\mathbf{x}_I = \{\mathbf{p}_I^w, \mathbf{R}_I^w, \mathbf{v}_I^w, \boldsymbol{\omega}_I\}$ is $\{I\}$'s state. Besides, $\mathbf{p}_I^w$, $\mathbf{R}_I^w$ are translation and rotation from $\{I\}$ to $\{w\}$, $\mathbf{v}_I^w$ is IMU velocity in $\{w\}$, and $\boldsymbol{\omega}_I = \left[\omega_I^1, \omega_I^2, \omega_I^3\right]^T$ is angular velocity in $\{I\}$. Similarly, replacing the subscript i with b represents $\{b\}$'s state in $\{w\}$. In this paper, we use both 3-dimensional (3D) rotation matrices $\mathbf{R} \in SO(3)$ and Lie algebra $\boldsymbol{\theta} \in \mathfrak{so}(3)$ to represent rotation.

2) The parameters of the dynamic model are $\mathbf{x}_d = \{k_f, k_m, \mathbf{I}_b, \mathbf{d}, \mathbf{b}, \mathbf{p}_M^b, \mathbf{R}_G^w, k_h\}$. k_f is the thrust coefficient and k_m is the aerodynamic torque coefficient. $\mathbf{I}_b = \mathrm{diag}\left(\left[I_b^1, I_b^2, I_b^3\right]^T\right)$ is a 3D diagonal matrix, representing the moment of inertia. $\mathbf{d} = \left[d^1, d^2, d^3\right]^T$ is a 3D vector, representing the drag coefficients [49], $\mathbf{b} = \left[b^1, b^2, b^3\right]^T$ is a 3D vector, representing the viscous coefficients [49]. $\mathbf{p}_M^b$ is the CoG's position in $\{b\}$. $\mathbf{R}_G^w$ is the gravity's rotation matrix in $\{w\}$. k_h is the HVT coefficient [44, 49].

3) The extrinsic parameters $\mathbf{x}_e = \left\{\mathbf{p}_b^I,\ \mathbf{R}_b^I\right\}$ also need to be calibrated. $\mathbf{p}_b^I$ is the position of $\{b\}$ in $\{I\}$ and $\mathbf{R}_b^I$ is the rotation from $\{b\}$ to $\{I\}$.

Notably, diag() is an operator that creates a diagonal matrix from a given vector.

A. Improved dynamic model

The dynamic model [44, 49] $\mathbf{x}_b = \mathrm{f}\left(\mathbf{x}_b, \mathbf{u}_b\right)$ of the quadrotor is as follows:

$$\begin{aligned}
\dot{\mathbf{p}}_b^w &= \mathbf{v}_b^w \\
\dot{\mathbf{R}}_b^w &= \mathbf{R}_b^w \boldsymbol{\omega}_b^{\times} \\
\dot{\mathbf{v}}_b^w &= -\mathbf{R}_G^w \mathbf{e}_3 g + \frac{\mathbf{R}_b^w \mathbf{T}_b}{\mathrm{m}_b} + \frac{\mathbf{R}_b^w \mathrm{diag}(\mathbf{d}) \mathbf{R}_w^b \mathbf{v}_b^w}{\mathrm{m}_b} + \mathbf{H}_{VT} \\
\dot{\boldsymbol{\omega}}_b &= \mathbf{I}_b^{-1}\left(-\boldsymbol{\omega}_b^{\times}\mathbf{I}_b\boldsymbol{\omega}_b + \mathbf{M}_b + \mathrm{diag}(\mathbf{b})\boldsymbol{\omega}_b\right)
\end{aligned} \tag{1}$$

where $\mathbf{R}_w^b$ is the inverse matrix of $\mathbf{R}_w^b$, m_b is the weight, $\mathbf{T}_b$ and $\mathbf{M}_b$ are 3D vectors, representing resultant thrust and resultant torque, respectively, g is the gravity constant, and $\mathbf{e}_3$ is $\begin{bmatrix}0 & 0 & 1\end{bmatrix}^{\mathrm{T}}$. The skew-symmetric matrix $\boldsymbol{\omega}_b^{\times}$ is as follows:

$$\boldsymbol{\omega}_b^{\times} = \begin{bmatrix} 0 & -\omega_b^3 & \omega_b^2 \\ \omega_b^3 & 0 & -\omega_b^1 \\ -\omega_b^2 & \omega_b^1 & 0 \end{bmatrix} \tag{2}$$

The torque term $\mathrm{diag}(\mathbf{b})\boldsymbol{\omega}_b$ in (1) represents viscous torque [49], and $\mathbf{H}_{VT}$ acts as a quadratic velocity-dependent input disturbance [44, 49], defined as:

$$\mathbf{H}_{VT} = k_h\left(v_{b_x}^w v_{b_x}^w + v_{b_y}^w v_{b_y}^w\right)\mathbf{e}_3 \tag{3}$$

B. Aerodynamics

The thrust $\mathbf{T}_b$ [50] generated by all rotors is as follows:

$$\mathbf{T}_b = \begin{bmatrix} 0 & 0 & k_f \sum_{j=1}^{4} \omega_{r_j}^2 \end{bmatrix}^T \tag{4}$$

where k_f is the thrust coefficient, and ω_{r_j} is the rotating velocity of the j-th rotor.

In addition, the resultant torque $\mathbf{M}_b$ [50] is as follows:

$$\mathbf{M}_b = \sum_{i=1}^{4}\left\{\mathbf{T}_b^j \times \mathbf{r}_i + (-1)^j k_m \mathbf{e}_3 \omega_{r_j}^2\right\} \tag{5}$$

where $\mathbf{r}_j$ is the position of i-th rotor in the body frame. In general, $\mathbf{r}_j$ is particular to thrust $\mathbf{T}_b^j$, and the third element of $\mathbf{r}_j$ is 0.

According to (4) and (5), the resultant thrust-torque of the quadrotor model is as follows:

$$\mathrm{g}(\mathbf{u}) = \begin{bmatrix} \mathbf{T}_b \\ \mathbf{M}_b \end{bmatrix} = \mathbf{K}\mathbf{u},$$

$$\mathbf{K} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ k_f & k_f & k_f & k_f \\ -k_f \mathrm{l}_y & k_f \mathrm{l}_y & k_f \mathrm{l}_y & -k_f \mathrm{l}_y \\ -k_f \mathrm{l}_x & -k_f \mathrm{l}_x & k_f \mathrm{l}_x & k_f \mathrm{l}_x \\ k_m & -k_m & k_m & -k_m \end{bmatrix}, \mathbf{u} = \begin{bmatrix} \omega_{r_1}^2 \\ \omega_{r_2}^2 \\ \omega_{r_3}^2 \\ \omega_{r_4}^2 \end{bmatrix} \tag{6}$$

where l_x and l_y are the length and width of every arm of the quadrotor.

C. CoG effect

There is a translation $\mathbf{p}_M^b$ between the body center and the mass center. The position of rotors at the CoG frame is redefined as $\hat{\mathbf{r}}_j$, which is as follows:

$$\hat{\mathbf{r}}_j = \mathbf{r}_j - \mathbf{p}_M^b \tag{7}$$

Therefore, the torque model needs to be re-constructed as follows:

$$\begin{aligned}
\hat{\mathbf{M}}_b &= \sum_{j=1}^{4}\left\{\mathbf{T}_b^j \times \left(\mathbf{r}_j - \mathbf{p}_M^b\right) + (-1)^j k_m \mathbf{e}_3 \omega_{r_j}^2\right\} \\
&= \mathbf{M}_b - \sum\nolimits_{j=1}^{4} \mathbf{T}_b^j \times \mathbf{p}_M^b \\
&= \mathbf{M}_b - \mathbf{C}_M \mathbf{p}_M^b
\end{aligned} \tag{8}$$

Therefore, according to (1) and (8), angular velocity dynamics become,

$$\dot{\boldsymbol{\omega}}_b = \mathbf{I}_b^{-1}\left(-\boldsymbol{\omega}_b^{\times}\mathbf{I}_b\boldsymbol{\omega}_b + \mathbf{M}_b + \mathrm{diag}(\mathbf{b})\boldsymbol{\omega}_b - \mathbf{C}_M\mathbf{p}_M^b\right) \tag{9}$$

where the term $\mathbf{C}_M\mathbf{p}_M^b$ is the torque caused by the CoG effect and the $\mathbf{C}_M$ is as follows:

$$\mathbf{C}_M = \begin{bmatrix} 0 & -k_f \sum_{j=1}^{4} \omega_{r_j}^2 & 0 \\ k_f \sum_{j=1}^{4} \omega_{r_j}^2 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \tag{10}$$

D. Actuator model

As shown in Fig 3, the pipeline of the actuator model includes two parts: the ESC model and the motor model. The actuator is controlled by the control input U_i from the controller, which converts electrical energy into mechanical energy. The ESC model converts the input voltage v_s into the analog voltage v_a applied to the motor. The battery voltage can be obtained by a voltmeter in real time. The ESC model cannot be directly derived, but we present an accurate model through experimental data. In addition, the motor model maps the analog voltage v_a into the rotor speed ω_r, in which motor dynamics include electrodynamics and mechanical dynamics.

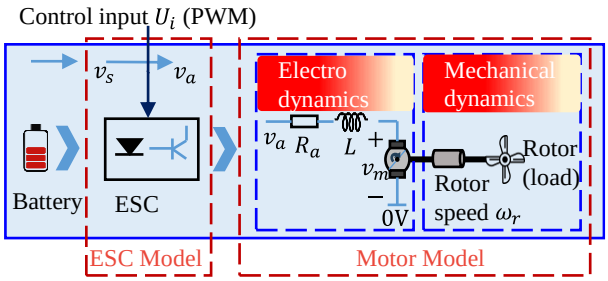


Fig. 3. The actuator model includes the electronic speed control (ESC) model and the motor model.

Electrodynamics and mechanical dynamics models are already well-established in [48]. A unified motor dynamic model in [48] is as follows:

$$\dot{\omega}_r = S_0 v_a^i - S_1 \omega_r - S_2 \omega_r^2 \tag{11}$$

where $S_0 = J^{-1} k_\Phi \Phi R_a^{-1}$, $S_1 = J^{-1}\left(k_\Phi \Phi R_a^{-1} k_\Phi \Phi - F\right)$, and $S_2 = J^{-1} k_m$. J is the rotor moment of inertia, v_a is analog voltage, R_a is resistance, and Φ is magnetic flux its coefficient is k_Φ, F is the viscous friction coefficient, and k_m is the aerodynamic torque coefficient. In general, as the electrical dynamic of the motor is much faster than the mechanical dynamic, the whole motor dynamic is assumed to behave as a first-order system, that is, $dI/dt \approx 0$. Therefore, inductance L is ignored in (11). In the calibration process, S_0, S_1, S_2 are regarded as a unit to be estimated.

Inspired by the ESC model of work [21], an ESC model based on real data tests is proposed,

$$v_a^i = v_s^i \sqrt{U_i - U_a} + U_i - U_b \tag{12}$$

where U_i is the control input of ESC at the timestamp i, v_s^i is the battery voltage, U_a and U_b are bias variables. The control input U_i is the stepped pulse-width modulation (PWM) signals from the controller.

Then, based on (11) and (12), the residuals of the actuator model are as follows:

$$\begin{aligned} r_a^i = {} & S_0\left(v_s^i \sqrt{U_i - U_a} + U_i - U_b\right) - S_1 \omega_{r_i} \\ & - S_2 \omega_{r_i}^2 - \frac{\left(\omega_{i+1} - \omega_i\right)}{\Delta t} \end{aligned} \tag{13}$$

where Δt is the actuator's control and measurement cycle.

The actuator model (13) can be estimated by the least squares method (LSM) [48].

III. METHODOLOGY

A. Covariance of thrust and torque

We assume an explicit noise model of the resultant thrust model with additive Gaussian white noise assumption:

$$\begin{aligned} \mathbf{T}_b &= \begin{bmatrix} n_x & n_y & \sum_{j=1}^{4} T_z^j \end{bmatrix}^{\mathrm{T}}, \\ T_z^j &= k_f \omega_{r_j}^2 + n_z, n_z \sim \mathcal{N}\left(0, \sigma_z^2\right), j = 1,2,3,4 \end{aligned} \tag{14}$$

where σ_z^2 is the variance of the thrust noise n_z of j-th actuator. n_x and n_y are unknown thrust disturbances in the x-axis and y-axis directions in $\{b\}$, and T_z^j is the thrust of j-th actuators. Also, we assume that $n_x \sim \mathcal{N}\left(0, \sigma_x^2\right)$ and $n_y \sim \mathcal{N}\left(0, \sigma_y^2\right)$ are Gaussian white noise.

Based on (6) and (14), the thrust and torque noise model can be derived as follows:

$$\begin{aligned} \hat{\mathbf{T}}_{b_i} &= \mathbf{T}_{b_i} + \mathbf{n}_T, \mathbf{n}_T \sim N\left(0, \mathbf{P}_T\right) \\ \hat{\mathbf{M}}_{b_i} &= \mathbf{M}_{b_i} + \mathbf{n}_M, \mathbf{n}_M \sim N\left(0, \mathbf{P}_M\right) \end{aligned} \tag{15}$$

where $\hat{\mathbf{T}}_{b_i}$ and $\hat{\mathbf{M}}_{b_i}$ are virtual measurements according to (4) and (5), $\mathbf{P}_T = \mathrm{diag}\left(\left[\sigma_x^2, \sigma_y^2, 4\sigma_z^2\right]^{\mathrm{T}}\right)$ and $\mathbf{P}_M = 4\mathrm{diag}\left(\left[\mathrm{l}_y^2, \mathrm{l}_x^2, k_m^2 / k_f^2\right]^{\mathrm{T}}\right)\sigma_z^2$ according to (6) and (14).

B. Error model of ODMF

The continuous-time stochastic differential equation is modeled as,

$$\begin{aligned} d\mathbf{x}(t) &= \mathrm{f}\left(\mathbf{x}(t), \mathbf{u}^*(t)\right)dt + \mathrm{g}\left(\mathbf{x}(t), \mathbf{u}^*(t)\right)d\mathbf{W}(t), \\ \mathbf{u}^* &= \left[\hat{\mathbf{T}}(t) \quad \hat{\boldsymbol{\omega}}(t) \quad \hat{\mathbf{M}}(t)\right]^{\mathrm{T}} \end{aligned} \tag{16}$$

where $\mathrm{g}\left(\mathbf{x}(t), \mathbf{u}^*(t)\right)$ modulates the stochastic process perturbation.

According to (15) and the gyroscope measurement noise variance is σ_g^2, we obtain:

$$\mathbf{u}_i^* \sim \mathcal{N}\left(\boldsymbol{\mu}_{\mathbf{u}_i^*}, \boldsymbol{\Sigma}_{\mathbf{u}_i^*}\right), \boldsymbol{\Sigma}_{\mathbf{u}_i^*} = \begin{bmatrix} \mathbf{P}_T & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \sigma_g^2 \mathbf{I}_3 & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{P}_M \end{bmatrix} \tag{17}$$

Given two time instants that correspond to body frames b_i and b_{i+1}, position, rotation, velocity, and angular velocity can be propagated by virtual measurements $\mathbf{u}^*$ during time interval $[t_i, t_{i+1}]$ in the body frame:

$$\begin{aligned} \mathbf{p}_{b_{i+1}}^w &= \mathbf{v}_{b_i}^w \Delta \mathrm{t} + \mathbf{p}_{b_i}^w - 0.5\mathbf{R}_G^w \mathbf{e}_3 \mathrm{g} \Delta \mathrm{t}^2 \\ &\quad + 0.5\mathbf{R}_{b_i}^w \left(\hat{\mathbf{T}}_{b_i} - \mathbf{n}_T\right)\Delta \mathrm{t}^2 \\ \mathbf{R}_{b_{i+1}}^w &= \mathbf{R}_{b_i}^w \left(\mathbf{I} + \left(\hat{\boldsymbol{\omega}}_{b_i} - \mathbf{n}_g\right)^{\times} \Delta \mathrm{t}\right) \\ \mathbf{v}_{b_{i+1}}^w &= \mathbf{v}_{b_i}^w - \mathbf{R}_{\mathrm{G}}^{\mathrm{w}} \mathbf{e}_3 \mathrm{g} \Delta \mathrm{t} + \mathbf{R}_{b_i}^w \left(\hat{\mathbf{T}}_{b_i} - \mathbf{n}_T\right) / \mathrm{m}_b \, \Delta \mathrm{t} \\ &\quad + \left(\mathbf{R}_{b_i}^w \mathrm{diag}\left(\mathbf{d}\right) \mathbf{R}_w^{b_i} \mathbf{v}_{b_i}^w / \mathrm{m}_b + \mathbf{H}_{VT}\right)\Delta \mathrm{t} \\ \boldsymbol{\omega}_{b_{i+1}} &= \boldsymbol{\omega}_{b_i} + \mathbf{I}_{\mathrm{b}}^{-1}\left(\hat{\mathbf{M}}_{b_i} - \mathbf{n}_M + \boldsymbol{\omega}_{b_i} \times \mathbf{I}_{\mathrm{b}} \boldsymbol{\omega}_{b_i}\right)\Delta \mathrm{t} \\ &\quad + \mathbf{I}_{\mathrm{b}}^{-1}\left(\mathrm{diag}\left(\mathbf{b}\right)\boldsymbol{\omega}_{b_i} - \mathbf{C}_M^i \mathbf{p}_m^b\right)\Delta \mathrm{t} \end{aligned} \tag{18}$$

where $\Delta \mathrm{t}$ is the period of states. $\mathrm{Log}(*)$ and $\mathrm{Exp}(*)$ represent the Logarithmic map and Exponential map of Lie theory, respectively.

We assume that the uncertainty of the error mainly depends on the virtual measurements $\mathbf{u}_i^*$. To facilitate the calculation of uncertainty, we deform the error model (18) by isolating the virtual measurements $\mathbf{u}_i^*$. The dynamic model factor's error function $\mathbf{r}_D = \left[\bar{\mathbf{e}}_p^{\mathrm{T}}, \bar{\mathbf{e}}_\theta^{\mathrm{T}}, \bar{\mathbf{e}}_v^{\mathrm{T}}, \bar{\mathbf{e}}_\omega^{\mathrm{T}}\right]^{\mathrm{T}}$ can be derived from the error model (18) as follows:

$$\begin{aligned}
\bar{\mathbf{e}}_p &= \mathbf{R}_w^{b_i}\mathrm{m}_b\left(\mathbf{p}_{b_{i+1}}^w - \mathbf{v}_{b_i}^w\Delta\mathrm{t} + 0.5\mathbf{R}_G^w\mathbf{e}_3\mathrm{g}\Delta\mathrm{t}^2 - \mathbf{p}_{b_i}^w\right)\\
&\quad -0.5\left(\hat{\mathbf{T}}_{b_i} - \mathbf{n}_T\right)\Delta\mathrm{t}^2\\
\bar{\mathbf{e}}_\theta &= \mathrm{Log}\left(\mathbf{R}_w^{b_i}\mathbf{R}_{b_{i+1}}^w\right) - \left(\hat{\boldsymbol{\omega}}_{b_i} - \mathbf{n}_g\right)\Delta\mathrm{t}\\
\bar{\mathbf{e}}_v &= \mathbf{R}_w^{b_i}\mathrm{m}_b\left(\mathbf{v}_{b_{i+1}}^w - \mathbf{v}_{b_i}^w + \mathbf{R}_{\mathrm{G}}^{\mathrm{w}}\mathbf{e}_3\mathrm{g}\Delta\mathrm{t}\right)\\
&\quad -\left(\hat{\mathbf{T}}_{b_i} - \mathbf{n}_T + \mathrm{diag}\left(\mathbf{d}\right)\mathbf{R}_w^{b_i}\mathbf{v}_{b_i}^w + \mathbf{H}_{VT}\right)\Delta\mathrm{t}\\
\bar{\mathbf{e}}_\omega &= \mathbf{I}_b\left(\boldsymbol{\omega}_{b_{i+1}} - \boldsymbol{\omega}_{b_i}\right) + \boldsymbol{\omega}_{b_i}\times\mathbf{I}_b\boldsymbol{\omega}_{b_i}\Delta\mathrm{t}\\
&\quad -\left(\mathrm{diag}\left(\mathbf{b}\right)\boldsymbol{\omega}_{b_i} + \hat{\mathbf{M}}_{b_i} - \mathbf{n}_M - \mathbf{C}_M^i\mathbf{p}_m^b\right)\Delta\mathrm{t}
\end{aligned} \tag{19}$$

The dynamic model factor's residual function can be converted into $\mathbf{r}_D = \mathbf{A}_i + \mathbf{B}_i\mathbf{u}_i^* + \mathbf{C}_i\mathbf{w}_i$. Furthermore, the covariance matrix $\mathbf{P}^D$ of $\mathbf{r}_D$ and its proof are supplied in Appendix A.

The extrinsic transformation from the IMU frame $\mathbf{x}_b$ to the body frame $\mathbf{x}_I$ is as follows:

$$\begin{aligned}
\mathbf{p}_b^w &= \mathbf{p}_I^w + \mathbf{R}_I^w\mathbf{p}_b^I\\
\mathbf{R}_b^w &= \mathbf{R}_I^w\mathbf{R}_b^I\\
\mathbf{v}_b^w &= \mathbf{v}_I^w + \mathbf{R}_I^w\left(\hat{\boldsymbol{\omega}}_I - \mathbf{b}_g\right)^\times\mathbf{p}_b^I\\
\boldsymbol{\omega}_b &= \left(\mathbf{R}_b^I\right)^T\left(\hat{\boldsymbol{\omega}}_I - \mathbf{b}_g\right)
\end{aligned} \tag{20}$$

where $\hat{\boldsymbol{\omega}}_I$ is the angular velocity measurement, and $\mathbf{b}_g$ is the estimated IMU gyroscope bias.

The Jacobian matrices $\dfrac{\partial\mathbf{r}_D}{\partial\boldsymbol{\chi}}$ are listed in Appendix B.

C. ODMC based on FGO

We adopt VIO output $\hat{\mathbf{x}}_{I_i}$ as the VIO factor from the work [41]. $\hat{\mathbf{z}}_i^D$ is rotor velocity measurement at the timestamp i. $\mathrm{r}_D\left(\hat{\mathbf{z}}_i^D, \boldsymbol{\chi}\right)$ is the dynamic model residual. Above all, the optimization problem can be summarized as follows:

$$\min_{\boldsymbol{\chi}}\left\{\begin{aligned}
&\left(\mathbf{r}_p - \mathbf{H}_p\boldsymbol{\chi}\right)^2 + \sum\nolimits_{k\in I}\mathrm{r}_I\left(\hat{\mathbf{x}}_{I_i}, \boldsymbol{\chi}\right)_{\mathbf{P}_{vio}}^2\\
&+\sum\nolimits_{i=0}^{N-1}\left(\mathrm{r}_D\left(\hat{\mathbf{z}}_i^D, \boldsymbol{\chi}\right)_{\mathbf{P}_i^D}^2\right)
\end{aligned}\right\} \tag{21}$$

where $\mathbf{r}_{\mathbf{P}}^2 = \mathbf{r}^T\mathbf{P}^{-1}\mathbf{r}$ is the Mahalanobis norm, r_I is the VIO factor residual and its covariance matrix is $\mathbf{P}_{vio}$, and $\mathbf{r}_p$ is the prior residual [42, 51]. $\mathbf{P}_i^D$ is supplied in Appendix A.

As shown in Fig 4, we construct a factor graph, in which VIO factors and online dynamic model factors (ODMFs) are fused based on incremental FGO. Duration the optimization, variables that exceed the time window will be marginalized. After the convergence of dynamic parameters, the controller can be compensated in real time. Above all, we derive the residual, covariance, and Jacobian matrices of the ODMF.

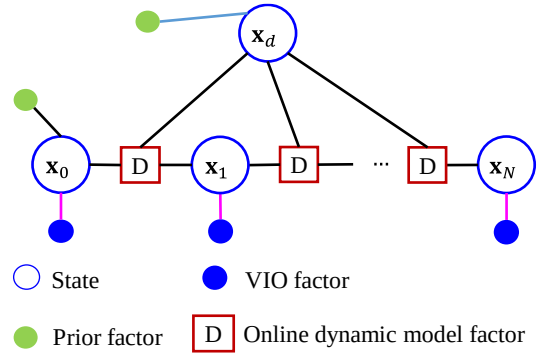


Fig. 4. The factor graph of online dynamic model calibration.

D. Controller

The DFBC [44] is adopted to verify the dynamic parameters. By adding more dynamics terms, the controller law is improved as follows:

$$\begin{aligned}
\mathbf{T}_c &= \mathrm{m}_b\mathbf{a}_{ref} + \mathbf{K}_p\mathrm{m}_b\left(\mathbf{p}_d^w - \mathbf{p}_b^w\right)\\
&\quad +\mathbf{K}_v\mathrm{m}_b\left(\mathbf{v}_d^w - \mathbf{v}_b^w\right) - \mathbf{R}_b^w\mathrm{diag}\left(\mathbf{d}\right)\mathbf{R}_w^b\mathbf{v}_b^w - \mathbf{K}_{VT}
\end{aligned} \tag{22}$$

$$\boldsymbol{\omega}_d = \mathbf{K}_R\mathrm{Logmap}\left(\mathbf{R}_w^b\mathbf{R}_d^w\right) + \boldsymbol{\omega}_{ref} \tag{23}$$

where $\mathbf{K}_p$, $\mathbf{K}_v$, and $\mathbf{K}_R$ are constant diagonal matrices, $\mathbf{p}_d^w$, $\mathbf{v}_d^w$, and $\mathbf{R}_d^w$ are the desired target, $\mathbf{a}_{ref}$ and $\boldsymbol{\omega}_{ref}$ are the acceleration and angular velocity from trajectory reference, $\mathbf{T}_c$ and $\boldsymbol{\omega}_d$ are control input. All the desired targets and references are generated by work [52].

Inspired by the incremental nonlinear dynamic inversion (INDI) controller [16, 44], the compensated angular velocity command (23) becomes:

$$\boldsymbol{\omega}_d^* = \boldsymbol{\omega}_d + \mathbf{K}_M^{-1}\left(\mathrm{diag}\left(\mathbf{b}\right)\boldsymbol{\omega}_{b_i} - \mathbf{C}_M^i\mathbf{p}_m^b\right) \tag{24}$$

where $\mathbf{K}_M$ is the angular velocity controller gain.

IV. EXPERIMENTS

A. Open-source datasets validation

1) Illustration of the abbreviations of different methods

ODMC, ODMC - no drag, ODMC - no 'B', ODMC - no HVT, and ***ODMC - no CoG*** represent the proposed **o**nline **d**ynamic **m**odel **c**alibration method, ODMC without aerodynamic drag term, ODMC without viscous torque term, ODMC without HVT term, and ODMC without CoG-induced torque, respectively.

***OMDC + NEE*:** the proposed **o**nline **d**ynamic **m**odel **c**alibration method on datasets with added Gaussian noise, namely **n**oise **e**xpansion **e**xperiments (NEE).

2) Force and torque prediction

This paper selects two widely used datasets Blackbird [43] and NeuroBEM [20] to evaluate the force and torque prediction performance. In addition, ground truth (GT)

data is generated using a motion capture system, which can provide centimeter-level positioning accuracy.

Force and torque prediction: The root mean squared error (RMSE) of the predicted force and torque is listed in Table I. Table I shows that the prediction accuracy of the proposed **ODMC** is comparable to **BEM+NN** and **HDVIO** methods from the NeuroBEM dataset experiments. Although data-driven methods (DDM), such as HDVIO, have a slight advantage, online calibration does not rely on datasets and can be applied to the unseen flight environment. The average time for the online calibration process is 3.4 milliseconds (approximately 294 Hz) on an NVIDIA Jetson TX2. However, the HDVIO inference runs at around 180 Hz on the same platform [12]. The proposed method and data-driven methods can both exceed the required 100 Hz state-update rate for agile flight. While data-driven methods are time-consuming, they leverage GPUs, thereby alleviating some of the burden on CPU resources.

In addition, Fig 5 and Fig 6 show the force, torque, aerodynamic drag, and HVT prediction of different datasets. *What does the dynamic model prediction precision affect?* On the one hand, **ODMC - no ' B '** reveals that viscous torque is the main component of classical dynamic unmodelled torque. Because, if viscous torque is ignored, the x-axis and y-axis torque prediction RMSE is greater than **ODMC** by 2 times for the NeuroBEM. As for the Blackbird, the viscous torque $\mathrm{diag}(\mathbf{b})\boldsymbol{\omega}_b$ is negligible because the angular velocity amplitude values in this dataset are much smaller than those in the NeuroBEM. Additionally, the z-axis torque is not influenced by viscous torque. On the other hand, **ODMC - no drag** reveals that aerodynamic drag is the main component of classical dynamic unmodelled force. Compared to **ODMC**, **ODMC - no drag**'s x-axis and y-axis ext. force RMSE increases by nearly 3 times, and the z-axis ext. force RMSE increases by 9 times. As for the dataset Blackbird, the predictive performance exhibits the same pattern. Our dataset calibration experiments also indicate that the proposed **ODMC** can achieve accurate force and torque prediction. Furthermore, the explicit model methods are comparable to data-driven methods (**DDM**) proposed by other scholars.

***HVT*:** As for **ODMC - no HVT** in Table I, without considering HVT in the dataset Blackbird, the RMSE of z-axis force prediction RMSE increases by 15%. Although the error increment is small, it still has a crucial impact on the high-speed control of UAVs. As for the HVT in the dataset NeuroBEM, the prediction RMSE of the z-axis force increases by 0.015 N.

***NEE*:** To investigate the impact of positioning noise, we add Gaussian noise into the position. When Gaussian noise with a mean square error of 0.05 m ($\sigma = 0.05m$) is added to the position values of the x, y, and z axes, the RMSE of the force prediction on the z-axis of the body frame increases by 50%. In addition, the RMSE of the force prediction on the x-O-y plane of the body frame increases by 1.32 times. Furthermore, Expansion noise experiments with Gaussian noise $\sigma = 0.10m$ is also tested.

Dynamic model parameters convergence: The time window is set to 1 second in the increment smoother. As illustrated in Fig 7, all dynamic model parameters $\mathbf{x}_d = \left\{k_f, \mathbf{I}_b, \mathbf{d}, \mathbf{b}, \mathbf{R}_G^w, k_h\right\}$ converge to a constant value within 9 seconds. Besides, d^3 is observable due to the body's z-axis velocity being greater than zero. Because k_m and I_b^3 cannot be identified jointly, we set k_m as a constant value.

Table I

The RMSE of predicted force and torque

Datasets	Method	Ext. Force (N)			Torque (N*m)		
		x-axis	y-axis	z-axis	x-axis	y-axis	z-axis
Blackbird [43]	**ODMC (Ours)**	**0.066**	0.053	0.059	**0.021**	**0.033**	**0.0071**
	ODMC - no ' B '	0.068	**0.051**	**0.056**	0.025	0.035	0.0088
	ODMC - no drag	0.540	0.460	0.380	0.027	0.050	0.0092
	ODMC - no HVT	0.068	0.053	0.068	0.022	0.033	0.0071
NeuroBEM [20]	**PolyFit [26]**	0.453		0.832	0.027		0.008
	BEM+NN [20]	**0.204**		0.504	**0.014**		**0.004**
	HDVIO [12]	0.402		0.672	-		-
	ODMC	0.185	0.188	**0.459**	0.0142	0.0232	0.0052
	ODMC - no ' B '	0.196	0.212	0.616	0.0640	0.0800	0.0052
	ODMC - no drag	1.080	1.160	1.792	0.0160	0.0272	0.0052
	ODMC - no HVT	0.196	0.212	0.540	0.0164	0.0276	0.0052
	ODMC + NEE ($\sigma = 0.05$ *m*)	0.340	0.356	0.716	0.0164	0.0272	0.0052
	ODMC + NEE ($\sigma = 0.10$ *m*)	0.988	0.808	0.864	0.0164	0.0276	0.0052
Ours	**ODMC**	0.048	0.060	0.139	0.0112	0.0138	0.007
	ODMC - no CoG	0.048	0.063	0.143	0.0112	0.0510	0.010

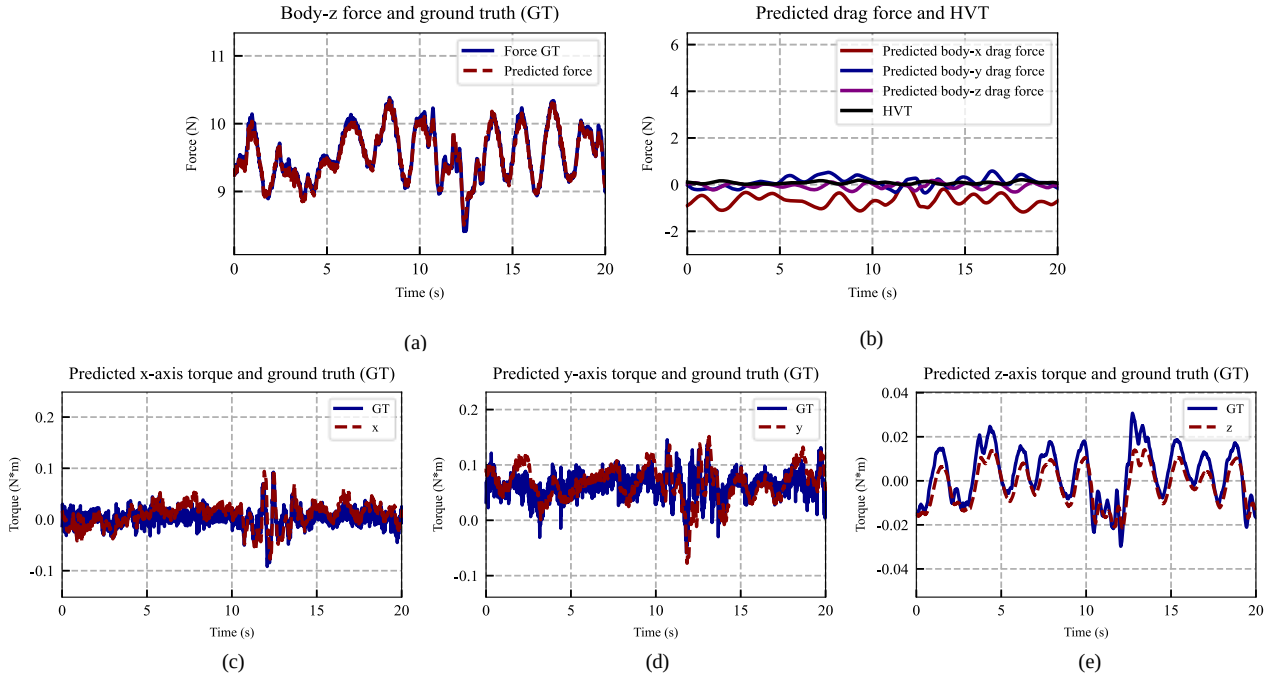


Fig. 5. **Blackbird**: (a) Predicted force in body x-axis and force ground truth. (b) Predicted drag force and HVT $\mathbf{H}_{VT}$. (c) Predicted torque in body x-axis and ground truth. (d) Predicted torque in body y-axis and ground truth. (e) Predicted torque in body z-axis and ground truth.

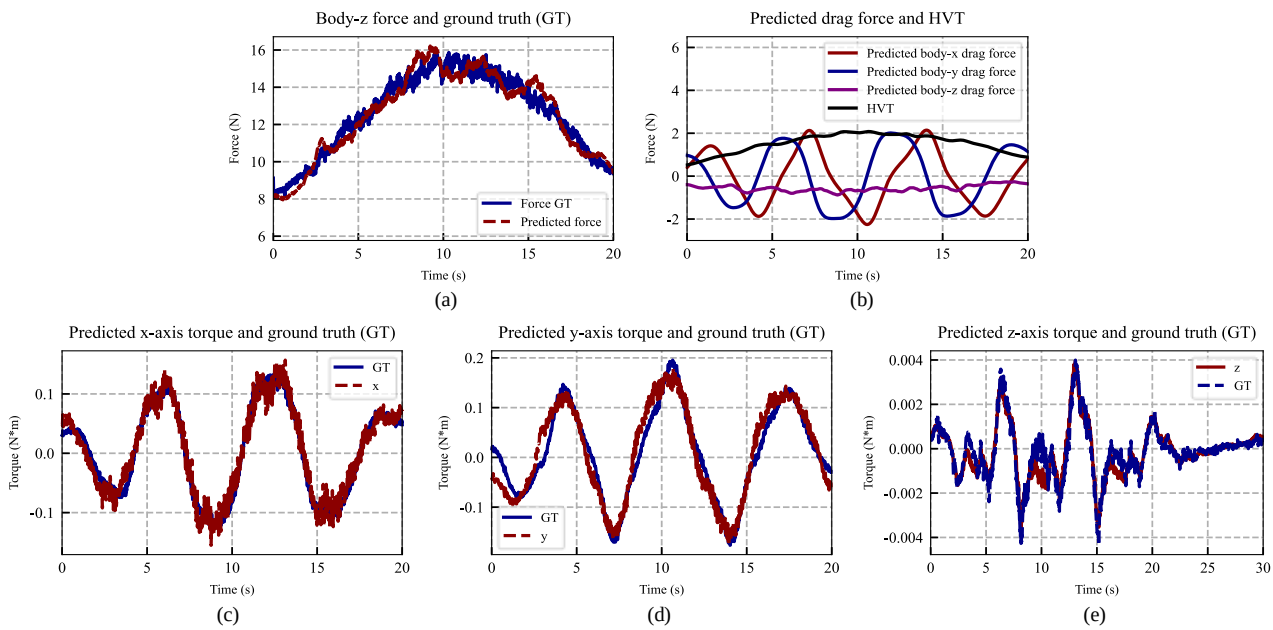


Fig. 6. **NeuroBEM:** (a) Predicted force in body x-axis and force ground truth (maximum velocity is 8.8 m/s). (b) Predicted drag force and HVT H_{VT} . (c) Predicted torque in body x-axis and ground truth. (d) Predicted torque in body y-axis and ground truth. (e) Predicted torque in body z-axis and ground truth.

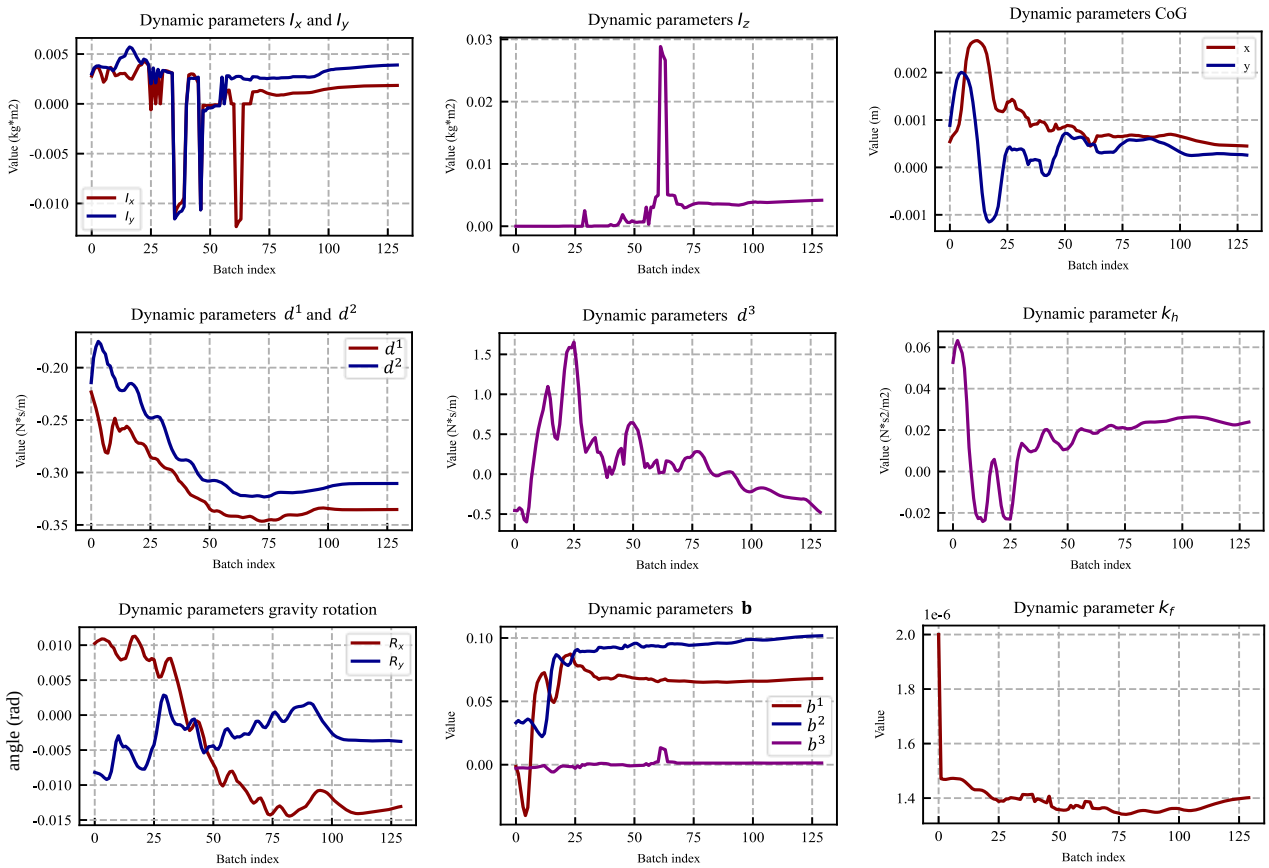
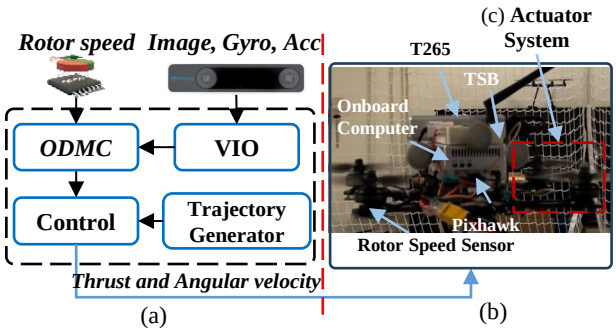


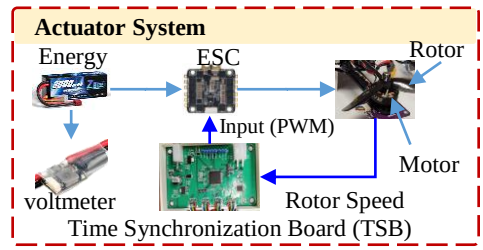
Fig. 7. **NeuroBEM**: Dynamic model parameters optimization process (maximum velocity is 8.8 m/s).

B. Self-dataset experiments

1) Experiment setup

We have developed a scalable quadrotor equipped with an IMU, camera, and four rotor velocity sensors. The quadrotor's framework is illustrated in Fig 8. To benefit the research community, we will open-source the hardware, code, and datasets. The UAV performs an octo-flight and the trajectory is generated by [52].





(c)

Fig. 8. Overview of the quadrotor system. (a) The framework of proposed online calibration and control. (b) The developed quadrotor. (c) The actuator system.

2) Actuator calibration

Why do we need to calibrate the actuator? The aerodynamic torque coefficient k_m and the z-axis value of the moment of inertia cannot be observed jointly. The reason is the following: According to (5) and (19), the torque's residual is $\bar{\mathbf{e}}_\omega = \mathbf{I}_b \Delta\boldsymbol{\omega}_{b_i} + \boldsymbol{\omega}_{b_i} \times \mathbf{I}_b \boldsymbol{\omega}_{b_i} \Delta t$

$$- \left(\operatorname{diag}(\mathbf{b}) \boldsymbol{\omega}_{b_i} + \sum_{j=1}^{4} \left\{ \mathbf{T}_b^j \times \mathbf{r}_j + (-1)^j k_m \mathbf{e}_3 \omega_{r_j}^2 \right\} - \mathbf{C}_M^i \mathbf{p}_m^b \right) \Delta t$$

, where $\Delta\boldsymbol{\omega}_{b_i} = \boldsymbol{\omega}_{b_{i+1}} - \boldsymbol{\omega}_{b_i}$. Because the first two elements

of the thrust vector $\mathbf{T}_b^j$ are close to 0, the third element of $\mathbf{T}_b^j \times \mathbf{r}_j$ is zero. In other words, torque around the z-axis (yaw torque) is only generated by differential aerodynamic torque among the four rotating propellers. According to (10), the third element of $\mathbf{C}_M^i \mathbf{p}_m^b$ is also zero. Finally, the third element of $\bar{\mathbf{e}}_\omega$ is

$$I_b^3 \Delta\omega_{b_i}^3 + \Delta t \omega_{b_i}^1 \omega_{b_i}^2 \left(I_b^2 - I_b^1\right) - k_m \Delta t \sum_{j=1}^{4} (-1)^j \omega_{r_j}^2 - \Delta t b^3 \omega_{b_i}^3$$

$$\approx I_b^3 \Delta\omega_{b_i}^3 - k_m \Delta t \sum_{j=1}^{4} (-1)^j \omega_{r_j}^2 = 0 \ , \quad \text{because} \quad I_b^2 - I_b^1 \approx 0$$

and $\Delta t b^3 \omega_{b_i}^3 \approx 0$. In summary, I_b^3 and k_m cannot be jointly estimated due to the linear dependence between $\Delta\omega_{b_i}^3$ and $\Delta t \sum_{j=1}^{4} (-1)^j \omega_{r_j}^2$, despite multiple constraints given by $I_b^3 \Delta\omega_{b_i}^3 - k_m \Delta t \sum_{j=1}^{4} (-1)^j \omega_{r_j}^2 = 0$. Therefore, we need to calibrate k_m independently.

Calibration process: The TSB sends stepped pulse-width modulation (PWM) signals to the motor's ESC. At the same time, TSB can measure the rotor velocity in revolutions per minute (RPM). The UAV's power module can measure the battery voltage. All data is collected by the onboard computer.

The calibration results are illustrated on Fig 9. PWM starts at 1100 with a step value of 50 and continues until 1550. Each PWM value is maintained for 3 seconds. As illustrated in Fig 9-(a), the rotor speed measurements are collected, induced by the stepped PWM signals. The response phase represents dynamical behavior in which the motor's velocity increases to a maximum value. The steady state represents the dynamic process that does not change or change only slightly over time. The blue line represents the RPM difference between adjacent frames. The purple line represents the motor velocity prediction error based on the calibration results. The rotor velocity prediction variances for both the response phase and steady state are equal, indicating consistent prediction accuracy under static and dynamic conditions. The calibrated k_m is 0.000573 $\mathrm{N \cdot m / RPM^2}$. In addition, ESC model parameters $U_a = 760.680$ RPM and $U_b = 1255.490$ RPM .

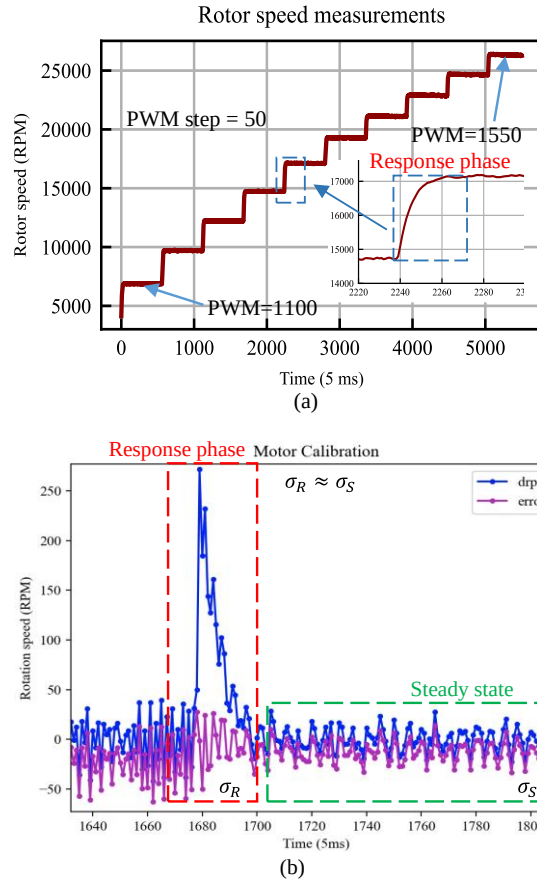


Fig. 9. (a) The rotor speed measurements induced by stepped PWM signals. (b) Actuator calibration results. σ_R and σ_S represent the response phase's variance and the steady state's variance, respectively.

3) Force and torque prediction

The force and torque prediction results are listed in Table I. The results show that the prediction's performance is consistent with the open-source dataset. Based on our datasets, the results demonstrate that the RMSE of force and torque predictions are approximately 0.138 N and 0.014 $\mathrm{N \cdot m}$, respectively. Then, the CoG parameter is adjusted by shifting the onboard computer 1.0 cm from the center position. As listed in Table I, without considering CoG-induced torque, the RMSE of y-axis torque prediction is severely distorted. Furthermore, failing to identify the CoG parameters results in a completely inaccurate moment of inertia.

C. Control Experiments

As shown in Fig 10, the compensated control by the calibrated dynamic model greatly improves the trajectory tracking accuracy of UAVs on the x-O-y plane. After compensation, the RMSE of x, y, and z's position tracking error is 3.4 cm, 3.2 cm, and 1.2 cm, respectively.

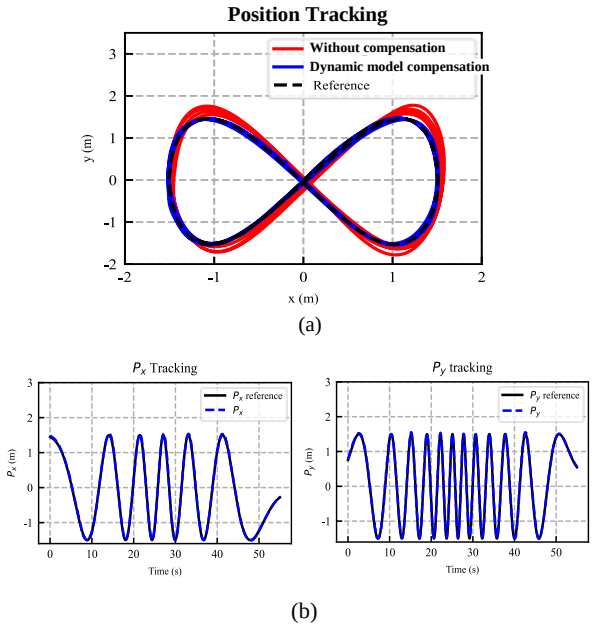
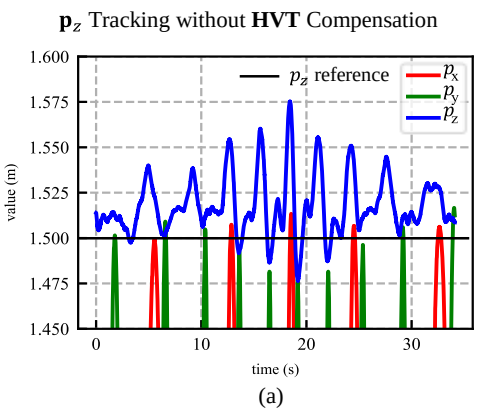


Fig. 10. Comparison of octo-flight position tracking performance between controller based on dynamic model compensation and uncompensated controller (maximum velocity is 5 m/s). (a) Position tracking comparison with dynamic model compensation and without compensation. (b) X-axis and y-axis position tracking curves with dynamic model compensation.

HVT impact: Besides, to verify the impact of HVT, we compare the performance of octo-flight position tracking with HVT compensation. As illustrated in Fig 11-(a), without HVT compensation, the z-axis tracking experiences a severe vibration due to the HVT effect. Because the force generated by the horizontal velocity is superimposed on the controlled thrust. However, with the help of HVT compensation, the RMSE of the z-axis position tracking significantly decreases from 2.4 cm to 1.2 cm. The maximum absolute value of errors has decreased from 7.5 cm to 4.1 cm.



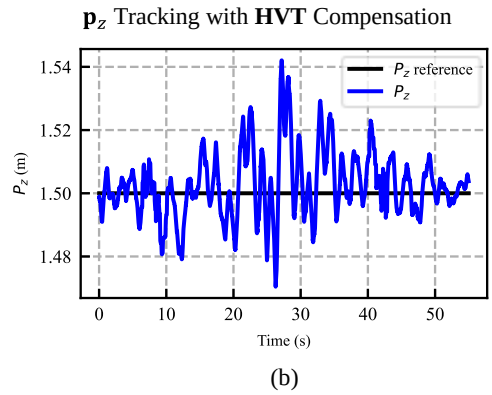


Fig. 11. V-shape position tracking curve for HVT evaluation. (a) Tracking accuracy of position without HVT compensation. (b) Tracking accuracy of z-axis position with HVT compensation (maximum velocity is 3.5 m/s).

V. DISCUSSION

ODMC versus DDM: (a) **ODMC can run online, but DDM is a black box relying on training data:** The HDVIO network only used a buffer of collective thrust and gyroscope measurements as input. However, the force is coupled with velocity. Furthermore, some hidden variables, e.g., CoG, may change during the flight. Current networks cannot update online efficiently. Conversely, the first principal model derived from physical models has stronger universality. (b) **ODMC relies on the ideal fluid assumption**: The flight scenarios are simple for low-velocity datasets below 9 m/s, in which air behaves as an explicit model. However, when dealing with complex aerodynamics, such as turbulent flow, typhoons, or interaction flow, describing the model becomes challenging. This is a bottleneck of the first principal model.

VI. CONCLUSION AND FUTURE WORK

This paper proposes an online dynamic model identification method based on FGO. A factor graph is designed to perform state, extrinsic parameters, and dynamical parameters calibration. A quadrotor is applied to perform data collection, calibration, and control experiments. In addition, the proposed online calibration framework is evaluated by open-source datasets and our self-collected datasets. The results demonstrate that the proposed method effectively achieves accurate and robust online identification of dynamical parameters. In addition, the calibrated dynamic model can improve the trajectory tracking performance of the UAVs. In the future, we plan to extend the work to outdoor scenarios.

ACKNOWLEDGMENT

This paper is funded by the MEITUAN ACADEMY OF ROBOTICS SHENZHEN under the project "*Vision Aided GNSS-RTK Positioning for UAV System in Urban*

Canyons (ZGHQ)". This paper is also funded by the PolyU Research Institute for Advanced Manufacturing (RIAM) under the project "*Unmanned Aerial Vehicle Aided High Accuracy Addictive Manufacturing for Carbon Fiber Reinforced Thermoplastic Composites Material (CD8S)*".

APPENDIX A

The conversion of $\mathbf{r}_D$ is as follows:

$$\mathbf{r}_D = \mathbf{A}_i + \mathbf{B}_i\mathbf{u}_i^* + \mathbf{C}_i\mathbf{w}_i,$$
$$\mathbf{C}_i \approx \begin{bmatrix} 0.5\Delta t^2\mathbf{I}_3 & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \Delta t\mathbf{I}_3 & \mathbf{0} \\ \Delta t\mathbf{I}_3 & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \Delta t\mathbf{I}_3 \end{bmatrix}, \mathbf{w}_i = \begin{bmatrix} \mathbf{n}_T \\ \mathbf{n}_g \\ \mathbf{n}_M \end{bmatrix} \tag{25}$$

Thus, the covariance matrix $\mathbf{P}^D$ of $\mathbf{r}_D$ is as follows:

$$\mathbf{P}^D = \mathbf{C}_i\boldsymbol{\Sigma}_{\mathbf{u}_i^*}\mathbf{C}_i^{\mathrm{T}} = \begin{bmatrix} 0.25\mathbf{P}_T\Delta t^4 & \mathbf{0} & 0.5\mathbf{P}_T\Delta t^3 & \mathbf{0} \\ \mathbf{0} & \sigma_g^2\mathbf{I}_3\Delta t^2 & \mathbf{0} & \mathbf{0} \\ 0.5\mathbf{P}_T\Delta t^3 & \mathbf{0} & \mathbf{P}_T\Delta t^2 & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{P}_M\Delta t^2 \end{bmatrix} \tag{26}$$

APPENDIX B

The Jacobian matrices of error $\mathbf{r}_D$ concerning state $\mathbf{x}_{I_i}$ and state $\mathbf{x}_{I_{i+1}}$ are as follows:

$$\begin{aligned} \frac{\partial \mathbf{r}_D}{\partial \mathbf{x}_{I_i}} &= \frac{\partial \mathbf{r}_D}{\partial \mathbf{x}_{b_i}}\frac{\partial \mathbf{x}_{b_i}}{\partial \mathbf{x}_{I_i}} = \frac{\partial \mathbf{r}_D}{\partial \mathbf{x}_{b_i}}\frac{\partial \mathbf{x}_b}{\partial \mathbf{x}_I} \\ \frac{\partial \mathbf{r}_D}{\partial \mathbf{x}_{I_{i+1}}} &= \frac{\partial \mathbf{r}_D}{\partial \mathbf{x}_{b_{i+1}}}\frac{\partial \mathbf{x}_{b_{i+1}}}{\partial \mathbf{x}_{I_{i+1}}} = \frac{\partial \mathbf{r}_D}{\partial \mathbf{x}_{b_{i+1}}}\frac{\partial \mathbf{x}_b}{\partial \mathbf{x}_I} \end{aligned} \tag{27}$$

The Jacobian matrix of error $\mathbf{r}_D$ concerning state $\mathbf{x}_{b_i}$ is as follows:

$$\frac{\partial \mathbf{r}_D}{\partial \mathbf{x}_{b_i}} = \begin{bmatrix} -\mathbf{R}_w^{b_i}\mathrm{m}_b & \left(\mathbf{R}_w^{b_i}\mathbf{p}_p\right)^{\times} & -\mathbf{R}_w^{b_i}\mathrm{m}_b\Delta t & 0 \\ 0 & -\mathbf{R}_w^{b_{i+1}}\mathbf{R}_{b_i}^w & 0 & -\mathbf{I}_{3\times 3}\Delta t \\ 0 & \frac{\partial \bar{\mathbf{e}}_{\mathrm{v}}}{\partial \boldsymbol{\theta}_{b_i}^w} & \frac{\partial \bar{\mathbf{e}}_{\mathrm{v}}}{\partial \mathbf{v}_{b_i}^w} & 0 \\ 0 & 0 & 0 & \frac{\partial \bar{\mathbf{e}}_{\omega}}{\partial \boldsymbol{\omega}_{b_i}} \end{bmatrix} \tag{28}$$

where $\mathbf{p}_p$, $\frac{\partial \bar{\mathbf{e}}_{\mathrm{v}}}{\partial \boldsymbol{\theta}_{b_i}^w}$, $\frac{\partial \bar{\mathbf{e}}_{\mathrm{v}}}{\partial \mathbf{v}_{b_i}^w}$ and $\frac{\partial \bar{\mathbf{e}}_{\omega}}{\partial \boldsymbol{\omega}_{b_i}}$ are as follows:

$$\begin{aligned} \mathbf{p}_p &= \mathrm{m}_b\left(\mathbf{p}_{b_{i+1}}^w - \mathbf{v}_{b_i}^w\Delta t + 0.5\mathbf{R}_G^w\mathbf{e}_3\mathrm{g}\Delta t^2 - \mathbf{p}_{b_i}^w\right) \\ \frac{\partial \bar{\mathbf{e}}_{\mathrm{v}}}{\partial \boldsymbol{\theta}_{b_i}^w} &= \left(\mathbf{R}_w^{b_i}\mathbf{p}_v\right)^{\times} - \mathrm{diag}\left(\mathbf{D}\right)\left(\mathbf{R}_w^{b_i}\mathbf{v}_{b_i}^w\right)^{\times}\Delta t \\ \mathbf{p}_v &= \mathrm{m}_{\mathrm{b}}\left(\mathbf{v}_{b_{i+1}}^w - \mathbf{v}_{b_i}^w + \mathbf{R}_G^w\mathbf{e}_3\mathrm{g}\Delta t\right) \end{aligned} \tag{29}$$

$$\frac{\partial \bar{\mathbf{e}}_{\mathrm{v}}}{\partial \mathbf{v}_{b_i}^w} = -\mathbf{R}_w^{b_i}\mathrm{m}_{\mathrm{b}} - \mathrm{diag}\left(\mathbf{D}\right)\mathbf{R}_w^{b_i}\Delta t$$

$$\frac{\partial \bar{\mathbf{e}}_{\omega}}{\partial \boldsymbol{\omega}_{b_i}} = \begin{bmatrix} 0 & \left(I_b^3 - I_b^2\right)\omega_3 & \left(I_b^3 - I_b^2\right)\omega_2 \\ \left(I_b^1 - I_b^3\right)\omega_3 & 0 & \left(I_b^1 - I_b^3\right)\omega_1 \\ \left(I_b^2 - I_b^1\right)\omega_2 & \left(I_b^2 - I_b^1\right)\omega_1 & 0 \end{bmatrix}\Delta t - \mathbf{I}_b$$

where $\mathbf{R}_{b_i}^w = \mathrm{Expmap}\left(\boldsymbol{\theta}_{b_i}^w\right)$.

The Jacobian matrix of error $\mathbf{r}_D$ concerning state $\mathbf{x}_{b_{i+1}}$ is as follows:

$$\frac{\partial \mathbf{r}_D}{\partial \mathbf{x}_{b_{i+1}}} = \begin{bmatrix} \mathbf{R}_w^{b_i}\mathrm{m}_b & 0 & 0 & 0 \\ 0 & \mathbf{I}_{3\times 3} & 0 & 0 \\ 0 & 0 & \mathbf{R}_w^{b_i}\mathrm{m}_{\mathrm{b}} & 0 \\ 0 & 0 & 0 & \mathbf{I}_b \end{bmatrix} \tag{30}$$

The Jacobian matrix of $\mathbf{x}_b$ concerning $\mathbf{x}_I$ is as follows:

$$\frac{\partial \mathbf{x}_b}{\partial \mathbf{x}_I} = \begin{bmatrix} \mathbf{I}_{3\times 3} & \left(\mathbf{P}_{\boldsymbol{b}}^{\boldsymbol{I}}\right)^{\times} & 0 & 0 \\ 0 & \mathbf{R}_I^b & 0 & 0 \\ 0 & \left\{\left(\hat{\boldsymbol{\omega}}_{\boldsymbol{I}_i} - \mathbf{b}_{\boldsymbol{g}}^{\boldsymbol{i}}\right)^{\times}\mathbf{P}_{\boldsymbol{b}}^{\boldsymbol{I}}\right\}^{\times} & \mathbf{I}_{3\times 3} & \mathbf{R}_I^w\left(\mathbf{P}_b^I\right)^{\times} \\ 0 & 0 & 0 & -\mathbf{R}_I^b \end{bmatrix} \tag{31}$$

The Jacobian matrices of error $\mathbf{r}_D$ concerning extrinsic parameters $\mathbf{x}_e$ are as follows:

$$\begin{aligned} \frac{\partial \mathbf{r}_D}{\partial \mathbf{p}_b^I} &= \frac{\partial \mathbf{r}_D}{\partial \mathbf{x}_{b_i}}\frac{\partial \mathbf{x}_b}{\partial \mathbf{p}_b^I} + \frac{\partial \mathbf{r}_D}{\partial \mathbf{x}_{b_{i+1}}}\frac{\partial \mathbf{x}_b}{\partial \mathbf{p}_b^I} \\ \frac{\partial \mathbf{r}_D}{\partial \boldsymbol{\theta}_b^I} &= \frac{\partial \mathbf{r}_D}{\partial \mathbf{x}_{b_i}}\frac{\partial \mathbf{x}_b}{\partial \boldsymbol{\theta}_b^I} + \frac{\partial \mathbf{r}_D}{\partial \mathbf{x}_{b_{i+1}}}\frac{\partial \mathbf{x}_b}{\partial \boldsymbol{\theta}_b^I} \\ \frac{\partial \mathbf{x}_b}{\partial \boldsymbol{\theta}_b^I} &= \begin{bmatrix} 0 & \mathbf{I}_{3\times 3} & 0 & \left\{\mathbf{R}_I^b\left(\hat{\boldsymbol{\omega}}_{I_i} - \mathbf{b}_g\right)\right\}^{\times} \end{bmatrix}^{\mathrm{T}} \\ \frac{\partial \mathbf{x}_b}{\partial \mathbf{p}_b^I} &= \begin{bmatrix} \mathbf{R}_I^w & 0 & \mathbf{R}_I^w\left(\hat{\boldsymbol{\omega}}_{\boldsymbol{I}_i} - \mathbf{b}_g\right)^{\times} & 0 \end{bmatrix}^{\mathrm{T}} \end{aligned} \tag{32}$$

where $\mathbf{R}_b^I = \mathrm{Expmap}\left(\boldsymbol{\theta}_b^I\right)$.

The Jacobian matrix of error $\mathbf{r}_D$ concerning dynamical parameters $\mathbf{x}_d$ is as follows:

$$\frac{\partial \mathbf{r}_D}{\partial \mathbf{x}_d} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ \frac{\partial \bar{\mathbf{e}}_{\mathrm{v}}}{\partial k_f} & 0 & 0 & \frac{\partial \bar{\mathbf{e}}_{\mathrm{v}}}{\partial \mathbf{d}} & 0 & 0 & \frac{\partial \bar{\mathbf{e}}_{\mathrm{v}}}{\partial \boldsymbol{\theta}_G^w} & \frac{\partial \bar{\mathbf{e}}_{\mathrm{v}}}{\partial k_h} \\ \frac{\partial \bar{\mathbf{e}}_{\omega}}{\partial k_f} & \frac{\partial \bar{\mathbf{e}}_{\omega}}{\partial k_m} & \frac{\partial \bar{\mathbf{e}}_{\omega}}{\partial \mathbf{I}_b} & 0 & \frac{\partial \bar{\mathbf{e}}_{\omega}}{\partial \mathbf{b}} & \frac{\partial \bar{\mathbf{e}}_{\omega}}{\partial \mathbf{p}_M^b} & 0 & 0 \end{bmatrix} \tag{33}$$

where:

$$\frac{\partial \bar{\mathbf{e}}_\omega}{\partial k_f} = \sum_{j=1}^{4} \left\{ \mathbf{T}_b^j \times \mathbf{r}_j + (-1)^j k_m \begin{bmatrix} 0 \\ 0 \\ \omega_{r_j}^2 \end{bmatrix} \right\}$$

$$\frac{\partial \bar{\mathbf{e}}_\omega}{\partial k_m} = \sum_{j=1}^{4} \left\{ (-1)^i k_f \begin{bmatrix} 0 \\ 0 \\ \omega_{r_j}^2 \end{bmatrix} \right\}$$

$$\frac{\partial \bar{\mathbf{e}}_\omega}{\partial \mathbf{I}_b} = \operatorname{diag}\left(\boldsymbol{\omega}_{b_{i+1}} - \boldsymbol{\omega}_{b_i}\right) + \mathbf{J}_{IB}$$

$$\mathbf{J}_{IB} = \begin{bmatrix} 0 & -\omega_{b_i}^2 \omega_{b_i}^3 & \omega_{b_i}^2 \omega_{b_i}^3 \\ \omega_{b_i}^1 \omega_{b_i}^3 & 0 & -\omega_{b_i}^1 \omega_{b_i}^3 \\ -\omega_{b_i}^1 \omega_{b_i}^2 & \omega_{b_i}^1 \omega_{b_i}^2 & 0 \end{bmatrix} \tag{34}$$

$$\frac{\partial \bar{\mathbf{e}}_v}{\partial k_f} = \begin{bmatrix} 0 & 0 & \sum_{i=1}^{4} \omega_{r_j}^2 \end{bmatrix}^{\mathrm{T}}$$

$$\frac{\partial \bar{\mathbf{e}}_v}{\partial \mathbf{d}} = -\operatorname{diag}\left(\mathbf{R}_w^{b_i} \mathbf{v}_{b_i}^w\right) \Delta \mathrm{t}$$

$$\frac{\partial \bar{\mathbf{e}}_\omega}{\partial \mathbf{b}} = -\operatorname{diag}\left(\boldsymbol{\omega}_{b_i}\right) \Delta \mathrm{t}$$

$$\frac{\partial \bar{\mathbf{e}}_v}{\partial \boldsymbol{\theta}_G^w} = \mathbf{R}_w^{b_i} \mathrm{m}_b \mathbf{R}_G^w \left(\mathbf{e}_3 \mathrm{g}\right)^\times$$

$$\frac{\partial \bar{\mathbf{e}}_\omega}{\partial \mathbf{p}_M^b} = -\mathbf{C}_M^i \Delta \mathrm{t}$$

$$\frac{\partial \bar{\mathbf{e}}_v}{\partial k_h} = \left(\mathbf{v}_{b_x}^w \mathbf{v}_{b_x}^w + \mathbf{v}_{b_y}^w \mathbf{v}_{b_y}^w\right) \mathbf{e}_3$$

REFERENCES

[1] Q. Xu, Z. Wang, and Z. Zhen, "Information fusion estimation-based path following control of quadrotor UAVs subjected to Gaussian random disturbance," *ISA Trans*, vol. 99, pp. 84-94, Apr 2020, doi: 10.1016/j.isatra.2019.10.003.

[2] J.-M. Kai, G. Allibert, M.-D. Hua, and T. Hamel, "Nonlinear feedback control of quadrotors exploiting first-order drag effects," presented at the IFAC-PapersOnLine, 2017.

[3] H. Kuang and X. Liu, "Convergence-Guaranteed Trajectory Optimization for Quadrotors Subject to Aerodynamic Drag," *IEEE Transactions on Aerospace and Electronic Systems*, pp. 1-12, 2024, doi: 10.1109/taes.2024.3414958.

[4] H. Cao, Y. Li, C. Liu, and S. Zhao, "ESO-Based Robust and High-Precision Tracking Control for Aerial Manipulation," *IEEE Transactions on Automation Science and Engineering*, pp. 1-17, 2023, doi: 10.1109/tase.2023.3260874.

[5] K. Zhang *et al.*, "Aerial additive manufacturing with multiple autonomous robots," *Nature*, vol. 609, no. 7928, pp. 709-717, Sep 2022, doi: 10.1038/s41586-022-04988-4.

[6] S. Kirchgeorg, E. Aucone, F. Wenk, and S. Mintchev, "Design, Modeling, and Control of AVOCADO: A Multimodal Aerial-Tethered Robot for Tree Canopy Exploration," *IEEE Transactions on Robotics*, vol. 40, pp. 592-605, 2024, doi: 10.1109/tro.2023.3334630.

[7] A. Sharma, G. F. Laupré, and J. Skaloud, "Identifying Aerodynamics of Small Fixed-Wing Drones Using Inertial Measurements for Model-Based Navigation," *NAVIGATION: Journal of the Institute of Navigation*, vol. 70, no. 4, 2023, doi: 10.33012/navi.611.

[8] L.-T. Hsu, "Analysis and modeling GPS NLOS effect in highly urbanized area," *GPS Solutions*, vol. 22, no. 1, 2017, doi: 10.1007/s10291-017-0667-9.

[9] W. W. Wen and L.-T. Hsu, "3D LiDAR Aided GNSS NLOS Mitigation in Urban Canyons," *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 10, pp. 18224-18236, 2022, doi: 10.1109/tits.2022.3167710.

[10] X. Bai, W. Wen, and L.-T. Hsu, "Time-Correlated Window-Carrier-Phase-Aided GNSS Positioning Using Factor Graph Optimization for Urban Positioning," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 58, no. 4, pp. 3370-3384, 2022, doi: 10.1109/taes.2022.3149730.

[11] B. Nisar, P. Foehn, D. Falanga, and D. Scaramuzza, "VIMO: Simultaneous Visual Inertial Model-based Odometry and Force Estimation," presented at the Robotics: Science and Systems Conference (RSS), 2019.

[12] C. Giovanni, B. Leonard, and S. Davide, "HDVIO: Improving Localization and Disturbance Estimation with Hybrid Dynamics VIO," presented at the Robotics: Science and Systems conference (RSS), Daegu, 2023

[13] D. Hanover, P. Foehn, S. Sun, E. Kaufmann, and D. Scaramuzza, "Performance, Precision, and Payloads: Adaptive Nonlinear MPC for Quadrotors," *IEEE Robotics and Automation Letters*, vol. 7, no. 2, pp. 690-697, 2022, doi: 10.1109/lra.2021.3131690.

[14] P. Foehn, A. Romero, and D. Scaramuzza, "Time-optimal planning for quadrotor waypoint flight," *SCIENCE ROBOTICS*, 2021.

[15] A. Romero, R. Penicka, and D. Scaramuzza, "Time-Optimal Online Replanning for Agile Quadrotor Flight," *IEEE Robotics and Automation Letters*, vol. 7, no. 3, pp. 7730-7737, 2022, doi: 10.1109/lra.2022.3185772.

[16] F. Nan, S. Sun, P. Foehn, and D. Scaramuzza, "Nonlinear MPC for Quadrotor Fault-Tolerant Control," *IEEE Robotics and Automation Letters*, vol. 7, no. 2, pp. 5047-5054, 2022, doi: 10.1109/lra.2022.3154033.

[17] Z. Huang and M. Chen, "Augmented Disturbance Observer-Based Appointed-Time Tracking Control of UAVs Under Exogenous Disturbance," *IEEE Transactions on Intelligent Vehicles*, vol. 9, no. 1, pp. 2822-2835, 2024, doi: 10.1109/tiv.2023.3303348.

[18] W. Jung and H. Bang, "Fault and Failure Tolerant Model Predictive Control of Quadrotor UAV," *International Journal of Aeronautical and Space Sciences*, vol. 22, no. 3, pp. 663-675, 2021, doi: 10.1007/s42405-020-00331-1.

[19] C. Ke, K.-Y. Cai, and Q. Quan, "Uniform Passive Fault-Tolerant Control of a Quadcopter With One, Two, or Three Rotor Failure," *IEEE Transactions on Robotics*, vol. 39, no. 6, pp. 4297-4311, 2023, doi: 10.1109/tro.2023.3297048.

[20] L. Bauersfeld, E. Kaufmann, P. Foehn, S. Sun, and D. Scaramuzza, "NeuroBEM Hybrid Aerodynamic Quadrotor Model," *Robotics: Science and Systems*, 2021.

[21] E. Kaufmann, L. Bauersfeld, A. Loquercio, M. Muller, V. Koltun, and D. Scaramuzza, "Champion-level drone racing using deep reinforcement learning," *Nature*, vol. 620, no. 7976, pp. 982-987, Aug 2023, doi: 10.1038/s41586-023-06419-4.

[22] Y. Qi, Y. Zhu, J. Wang, J. Shan, and H. H. T. Liu, "MUDE-based control of quadrotor for accurate attitude tracking," *Control Engineering Practice*, vol. 108, 2021, doi: 10.1016/j.conengprac.2020.104721.

[23] P. K. Swee, L. K. Li, W. Fei, C. M. Ben, and L. H. Tong, "Explicit model identification and control of a micro aerial vehicle," presented at the International Conference on Unmanned Aircraft Systems (ICUAS), 2014.

[24] R. Ding, S. Bai, K. Dong, and P. Chirarattananon, "Aerodynamic effect for collision-free reactive navigation of

a small quadcopter," *npj Robotics*, vol. 1, no. 1, 2023, doi: 10.1038/s44182-023-00002-9.

[25] H. Bu, H. Wu, C. Bertin, Y. Fang, and S. Zhong, "Aerodynamic and acoustic measurements of dual small-scale propellers," *Journal of Sound and Vibration*, vol. 511, 2021, doi: 10.1016/j.jsv.2021.116330.

[26] S. Sun, C. C. de Visser, and Q. Chu, "Quadrotor Gray-Box Model Identification from High-Speed Flight Data," *Journal of Aircraft*, vol. 56, no. 2, pp. 645-661, 2019, doi: 10.2514/1.C035135.

[27] D. Six, S. Briot, J. Erskine, and A. Chriette, "Identification of the Propeller Coefficients and Dynamic Parameters of a Hovering Quadrotor From Flight Data," *IEEE Robotics and Automation Letters*, vol. 5, no. 2, pp. 1063-1070, 2020, doi: 10.1109/lra.2020.2966393.

[28] C. Chen, Y. Yang, P. Geneva, W. Lee, and G. Huang, "Visual-Inertial-Aided Online MAV System Identification," presented at the 2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), 2022.

[29] A. Shastry and D. A. Paley, "UAV State and Parameter Estimation in Wind Using Calibration Trajectories Optimized for Observability," *IEEE Control Systems Letters*, vol. 5, no. 5, pp. 1801-1806, 2021, doi: 10.1109/lcsys.2020.3044491.

[30] Z. Xing, Y. Zhang, and C.-Y. Su, "Active Wind Rejection Control for a Quadrotor UAV Against Unknown Winds," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 59, no. 6, pp. 8956-8968, 2023, doi: 10.1109/taes.2023.3315254.

[31] P. Lyu, J. Lai, J. Liu, H. H. T. Liu, and Q. Zhang, "A Thrust Model Aided Fault Diagnosis Method for the Altitude Estimation of a Quadrotor," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 54, no. 2, pp. 1008-1019, 2018, doi: 10.1109/taes.2017.2773262.

[32] B. h. Christoph, B. Christian, H.-S. Alexander, and W. Stephan, "Combined System Identification and State Estimation for a Quadrotor UAV," presented at the ICRA, 2021.

[33] H. Jeon, J. Song, H. Lee, and Y. Eun, "Modeling Quadrotor Dynamics in a Wind Field," *IEEE/ASME Transactions on Mechatronics*, vol. 26, no. 3, pp. 1401-1411, 2021, doi: 10.1109/tmech.2020.3019831.

[34] Z. Ding, T. Yang, K. Zhang, C. Xu, and F. Gao, "VID-Fusion: Robust Visual-Inertial-Dynamics Odometry for Accurate External Force Estimation," presented at the 2021 IEEE International Conference on Robotics and Automation (ICRA), 2021.

[35] B. Nisar, P. Foehn, D. Falanga, and D. Scaramuzza, "VIMO: Simultaneous Visual Inertial Model-Based Odometry and Force Estimation," *IEEE Robotics and Automation Letters*, vol. 4, no. 3, pp. 2785-2792, 2019, doi: 10.1109/lra.2019.2918689.

[36] A. Papadimitriou, H. Jafari, S. S. Mansouri, and G. Nikolakopoulos, "External force estimation and disturbance rejection for Micro Aerial Vehicles," *Expert Systems with Applications*, vol. 200, 2022, doi: 10.1016/j.eswa.2022.116883.

[37] S. Bai, J. Lai, P. Lyu, B. Ji, B. Wang, and X. Sun, "A Novel Plug-and-Play Factor Graph Method for Asynchronous Absolute/Relative Measurements Fusion in Multisensor Positioning," *IEEE Transactions on Industrial Electronics*, vol. 70, no. 1, pp. 940-950, 2023, doi: 10.1109/tie.2022.3150077.

[38] W. Wen, X. Bai, Y. C. Kan, and L.-T. Hsu, "Tightly Coupled GNSS/INS Integration via Factor Graph and Aided by Fish-Eye Camera," *IEEE Transactions on Vehicular Technology*, vol. 68, no. 11, pp. 10651-10662, 2019, doi: 10.1109/tvt.2019.2944680.

[39] W. Wen, T. Pfeifer, X. Bai, and L. T. Hsu, "Factor graph optimization for GNSS/INS integration: A comparison with the extended Kalman filter," *Navigation*, vol. 68, no. 2, pp. 315-331, 2021, doi: 10.1002/navi.421.

[40] P. Yang and W. Wen, "Tightly Joining Positioning and Control for Trustworthy Unmanned Aerial Vehicles Based on Factor Graph Optimization in Urban Transportation," presented at the 2023 IEEE 26th International Conference on Intelligent Transportation Systems (ITSC), 2023.

[41] T. Qin, P. Li, and S. Shen, "VINS Mono A robot and Versatile Monocular Visual-Inertial State Estimator," 2017.

[42] M. Kaess, H. Johannsson, R. Roberts, V. Ila, J. J. Leonard, and F. Dellaert, "iSAM2: Incremental smoothing and mapping using the Bayes tree," *The International Journal of Robotics Research*, vol. 31, no. 2, pp. 216-235, 2011, doi: 10.1177/0278364911430419.

[43] A. Amado, G. Winter, M. Varun, S.-M. Thomas, and K. Sertac, "The Blackbird UAV Dataset," *The International Journal of Robotics Research*, 2016, doi: 10.1177/ToBeAssigned.

[44] S. Sun, A. Romero, P. Foehn, E. Kaufmann, and D. Scaramuzza, "A Comparative Study of Nonlinear MPC and Differential-Flatness-Based Control for Quadrotor Agile Flight," *IEEE Transactions on Robotics*, vol. 38, no. 6, pp. 3357-3373, 2022, doi: 10.1109/tro.2022.3177279.

[45] B. Rubí, R. Pérez, and B. Morcego, "A Survey of Path Following Control Strategies for UAVs Focused on Quadrotors," *Journal of Intelligent & Robotic Systems*, vol. 98, no. 2, pp. 241-265, 2019, doi: 10.1007/s10846-019-01085-z.

[46] H. Huang, M. G. Hoffmann, W. L. Steven, and J. C. Tomlin, "Aerodynamics_and_control_of_autonomous_quadrotor_helicopters_in_aggressive_maneuvering," presented at the IEEE International Conference on Robotics and Automation, 2009.

[47] W. Lei and C. Li, "On-line aerodynamic identification of quadrotor and its application to tracking control," *IET Control Theory & Applications*, vol. 11, no. 17, pp. 3097-3106, 2017, doi: 10.1049/iet-cta.2017.0664.

[48] C.-C. Peng and C.-Y. Su, "Modeling and Parameter Identification of a Cooling Fan for Online Monitoring," *IEEE Transactions on Instrumentation and Measurement*, vol. 70, pp. 1-14, 2021, doi: 10.1109/tim.2021.3104375.

[49] M. Faessler, A. Franchi, and D. Scaramuzza, "Differential Flatness of Quadrotor Dynamics Subject to Rotor Drag for Accurate Tracking of High-Speed Trajectories," *IEEE Robotics and Automation Letters*, vol. 3, no. 2, pp. 620-626, 2018, doi: 10.1109/lra.2017.2776353.

[50] P.-J. Bristeau, P. Martin, E. Salaun, and N. Petit, "The role of propeller aerodynamics in the model of a quadrotor UAV," presented at the Proceedings of the European Control Conference 2009, 2009.

[51] M. Kaess, A. Ranganathan, and F. Dellaert, "iSAM: Incremental Smoothing and Mapping," *IEEE Transactions on Robotics*, vol. 24, no. 6, pp. 1365-1378, 2008, doi: 10.1109/tro.2008.2006706.

[52] D. Mellinger and V. Kumar, "Minimum Snap Trajectory Generation and Control for Quadrotors," presented at the IEEE International Conference on Robotics and Automation, 2011.

Peiwen Yang received the M.S. degree in the School of Information and Electronics, Beijing Institute of Technology, Beijing, China, in 2019. He is currently a Ph.D. student in the Department of Aeronautical and Aviation Engineering, at the Hong Kong Polytechnic University. His current research interests include aerial vehicle control, computer vision, and robotics.

Weisong Wen (Member, IEEE) received a BEng degree in Mechanical Engineering from Beijing Information Science and Technology University (BISTU), Beijing, China, in 2015, and an MEng degree in Mechanical Engineering from the China Agricultural University, in 2017. After that, he received a PhD degree in Mechanical Engineering from The Hong Kong Polytechnic University (PolyU), in 2020. He was also a visiting PhD student with the Faculty of Engineering, University of California, Berkeley (UC Berkeley) in 2018. Before joining PolyU as an Assistant Professor in 2023, he was a Research Assistant Professor at AAE of PolyU since 2021. He has published 30 SCI papers and 40 conference papers in the field of GNSS (ION GNSS+) and navigation for Robotic systems (IEEE ICRA, IEEE ITSC), such as autonomous driving vehicles. He won the innovation award from TechConnect 2021, the Best Presentation Award from the Institute of Navigation (ION) in 2020, and the First Prize in the Hong Kong Section in Qianhai-Guangdong-Macao Youth Innovation and Entrepreneurship Competition in 2019 based on his research achievements in the 3D LiDAR aided GNSS positioning for robotics navigation in urban canyons. The developed 3D LiDAR-aided GNSS positioning method has been reported by top magazines such as Inside GNSS and has attracted industry recognition with remarkable knowledge transfer.

Shiyu Bai (Member, IEEE) was born in Xuzhou, Jiangsu, China. He received a Ph.D. degree in navigation, guidance, and control from Nanjing University of Aeronautics and Astronautics, Nanjing, China, in 2022. He is currently a postdoctoral fellow in the Department of Aeronautical and Aviation Engineering at the Hong Kong Polytechnic University. His research interests include inertial navigation, multi-sensor fusion, indoor positioning, and vehicular positioning.

Jiahao Hu received his M.S. degree from the School of Mechanical and Electrical Engineering at Wuhan University of Technology, Wuhan, China, in 2023. He is currently a Project Administrative Assistant in the Department of Aeronautical and Aviation Engineering at The Hong Kong Polytechnic University, Hong Kong. His current research interests include navigation and positioning, computer vision, and robotics.