Uncertainty Modeling Enabled Meta Adaptive Control for Aerial Manipulators

Gao, Shiqi; Hong, Haichao; Sun, Sihao; Luo, Lingkun; Hu, Shiqiang

DOI
10.2514/1.G008068

Publication date
2024

Document Version
Final published version

Published in
Journal of Guidance, Control, and Dynamics

JOURNAL OF GUIDANCE, CONTROL, AND DYNAMICS
Vol. 47, No. 10, October 2024

Technical Notes

Uncertainty Modeling Enabled Meta Adaptive Control for Aerial Manipulators

Shiqi Gao*
Shanghai Jiao Tong University, 200240 Shanghai, People's Republic of China
Haichao Hong†
Shanghai Jiao Tong University, 200240 Shanghai, People's Republic of China
Sihao Sun‡
Delft University of Technology, 2628 CD Delft, The Netherlands
Lingkun Luo§
Shanghai Jiao Tong University, 200240 Shanghai, People's Republic of China
and
Shiqiang Hu¶
Shanghai Jiao Tong University, 200240 Shanghai, People's Republic of China

https://doi.org/10.2514/1.G008068

I. Introduction

THE interaction of aerial robots with the environment has evolved from passively perceiving [1] to actively grasping external objects with a manipulator [2]. However, the uncertainties caused by the unknown payload inevitably pose challenges to the precise control of the aerial manipulator. This paper focuses on analyzing the uncertainties associated with an aerial manipulator grasping an unknown payload and deriving a unified disturbance model through a linear transformation of the system dynamics. Utilizing this disturbance model allows for more precise compensation, enhancing trajectory tracking performance through meta-adaptive control. The motivation for this research lies in identifying uncertainty, distinguishing between known states and unknown parameters, and ultimately improving controller performance by estimating these uncertain parameters. In the real world, the ability to handle unknown disturbances is crucial in robotic manipulation, particularly for aerial platforms, where stability is more vulnerable compared to fixed-base manipulators.

Dynamic analysis is the premise of the controller design. In current research, the perspectives regarding the dynamic relationship between the aerial platform and the manipulator diverge into decentralized and centralized approaches. The decentralized method treats the aerial vehicle and the manipulator as two independent subsystems [3]. This approach assumes that the motion of the manipulator can hardly influence the translation and rotation dynamics of the aerial vehicle, which leads to two strategies for satisfying this assumption. On one hand, the coupling effect is considered an additional minor disturbance [4]. However, the rapid movement of the aerial manipulator may amplify the coupling effect, potentially violating the aforementioned assumption. Alternatively, the coupling effect can be neglected by using a lightweight robotic arm [5]. Nonetheless, reducing the arm's strength may limit its range of motion when handling objects of equal weight [6,7]. As a result, the decentralized method performs well merely in quasi-static and light-load scenarios. In contrast, the centralized approach views the system as a whole, taking into account the coupling effect within the inertia matrix [8]. The system's dynamics can be established using either the Euler–Lagrangian [9] or Newton–Euler recursive formulation [10]. This approach, free from the need to address additional internal disturbances, allows for more stable and accurate tracking performance [11]. However, the dynamic model matrices of the whole system tend to be more complex than those of the decentralized system.

Estimating the uncertainty is the primary objective of the controller design to adapt to unknown environmental disturbances. Current approaches can adapt to uncertainty either before the actual task through offline methods or during the control mission using online strategies.

Building upon the first concept, the system and parameter identification leverage the collected flight data to approximate either the unknown dynamics or parameters. In the context of system identification, the mechanism responsible for generating forces and moments in rotors is identified from pulse-width modulation (PWM) signals on the test stand. This enables more precise execution of grasping and transporting tasks [12]. In another study, the linear dynamics of the system near the hovering state are estimated via frequency-domain identification. Subsequently, a robust H-infinity controller with adaptive auxiliary components is designed to counteract disturbances caused by grasping loads [13]. However, when the states of the system move beyond the well-fit region, the offline-identified model may encounter malfunctions. In terms of parameter identification, the least-squares method is applied to estimate the load mass. This estimated value is then used to enhance the proportional integration differentiation (PID) controller [14]. Nonetheless, specific types of movements are required to gather state observations for parameter estimation, which may limit the flexibility of mission trajectories.

The second thought delves into more advanced adaptive control techniques. For instance, a sliding-mode adaptive controller has been devised to enable a planar aerial manipulator to adapt to the unknown load mass and thrust gain [15]. Moreover, a similar controller has been further refined by incorporating real-time estimates of the load mass and inertia [16]. However, both the simplifications in dynamics and the limited number of payload parameters can lead to an imprecise model and tracking errors. In addition, the model reference adaptive control (MRAC) method aligns the behavior of the system with uncertainty to that of a known reference model. Within the hierarchical control framework of a quad-tilt-wing unmanned aerial vehicle (UAV) in [17], a quadrotor with a two-link robotic arm in [18] is decomposed into translational and rotational dynamics. The unknown mass and inertia are addressed using an outer loop model reference adaptive controller and an inner loop nonlinear adaptive controller, respectively. Nevertheless, the disregarded coupling effect suggests the need for a more comprehensive description of dynamics. Furthermore, the $\mathcal{L}_1$ adaptive control technique has shown promise in

Received 6 November 2023; accepted for publication 20 May 2024; published online 26 July 2024.

*Ph.D. Candidate, School of Aeronautics and Astronautics.
†Associate Professor, School of Aeronautics and Astronautics. Senior Member AIAA.
‡Postdoctoral Researcher, Department of Cognitive Robotics.
§Research Assistant, School of Aeronautics and Astronautics.
¶Professor, School of Aeronautics and Astronautics; sqhu@sjtu.edu.cn (Corresponding Author).

achieving quicker adaptation with a robustness guarantee [19,20]. For quadrotors, the architecture for $\mathcal{L}_1$ augmentation of a geometric controller for both the translational and rotational dynamics [21] is proposed, which successfully compensates the time- and state-dependent uncertainties. This control architecture is further improved with performance guarantees, which ensure the states of the closed-loop system are bounded around the desired trajectory [22]. In the domain of aerial manipulators, the technique of incorporating an augmented $\mathcal{L}_1$ adaptive law into the control loop for its rotational dynamics [6] and vertical dynamics [23] is also adopted to achieve a more rapid tracking response under the unknown nonlinear forces and moments. While the response performance has improved, there is room for further refinement of the matched and mismatched disturbance models to provide more precise compensation.

As the complexity of the disturbance increases, data-driven augmented control techniques come to the rescue, tackling the formidable nonlinearity. These approaches utilize both vanilla neural networks [24,25] and Gaussian processes [26] to provide approximations of the intricate aerodynamic effects. Notably, these methods lead to performance improvements in both the baseline geometric adaptive and model predictive controllers. Simultaneously, advanced learning techniques, such as meta-learning [27], have been integrated into the adaptive control framework to effectively combat wind disturbances in quadrotors [28,29]. This upgraded learning strategy empowers the controller with more precise nonlinear features, allowing for finer compensation. This advanced approach motivates the research work in this study—analytically investing in the dynamics and the role of the uncertainties.

This study focuses on the uncertainty analysis and the controller design for aerial manipulators, aimed at compensating for disturbances. To achieve a more precise description of the dynamics, we first consider the payload as an integral part of the system, namely, an extension of the manipulator. The uncertainty of the payload is characterized by parameters such as mass, inertia, mass center position, and the corresponding deviation angle, ensuring a comprehensive consideration. Then, we establish the centralized Euler–Lagrangian dynamic model for the entire aerial manipulator system. Secondly, the analytical formulation of the uncertainty is deduced by leveraging the linearly parameterizable dynamics property of the aerial manipulator. Moreover, the structure of the unknown disturbance unveils its underlying mechanism, which guides the design of the compensation using a *kinematic regressor* and a *parameter vector*. To validate the proposed control schemes, we estimate the compensation and integrate it into a meta-adaptive control framework. The kinematic regressor is approximated via neural networks to bypass the complexities of symbolic derivation. Additionally, we employ the model-agnostic meta-learning (MAML) strategy to expedite training convergence and enhance the adaptability of the controller to various payload configurations. Finally, we demonstrate the Lyapunov stability of the closed-loop system. In simulation, we observe improved tracking performance compared to other baseline controllers.

The rest of this paper is organized as follows: Sec. II provides a comprehensive explanation of the analytical procedures related to the dynamics of the unknown disturbance in the aerial manipulator. Section III introduces the implementation of the meta-adaptive control framework, which is supported by the derived precise compensation. Section IV demonstrates the validation of improved trajectory tracking performance through numerical simulations under various tasks and payload settings. Finally, Sec. V offers the conclusions drawn from this study.

II. Uncertain Disturbance Model for Aerial Manipulator

In this section, we introduce a universal linear transformation of the unknown disturbance relative to inertia parameters. Specifically, the unknown disturbance is regarded as the product of a kinematic regressor matrix associated with known states and a vector with all uncertain parameters. This decomposition enhances the precision of disturbance counteraction through the implementation of a structurally similar compensation strategy.

A. Origin of Uncertain Disturbance

Preliminarily, we consider systems that conform to Lagrangian dynamics, which are subject to an unknown disturbance f_a.

$$M(q)\ddot{q} + C(q,\dot{q})\dot{q} + G(q) + f_a = \tau \tag{1}$$

where $q \in \mathbb{R}^n$ is the state vector including n generalized coordinates of the system; $\tau \in \mathbb{R}^n$ is the vector of applied generalized force and torque; $M(q) \in \mathbb{R}^{n\times n}$ is the inertia matrix, which is normally symmetric positive definite; $C(q,\dot{q}) \in \mathbb{R}^{n\times n}$ is the Coriolis matrix; $C(q,\dot{q})\dot{q} \in \mathbb{R}^n$ represents the centripetal and Coriolis force; and $G(q) \in \mathbb{R}^n$ denotes the gravitational force.

Remark 1: In Sec. II.A, to emphasize the capability of the omniscient controller in perfectly counteracting disturbances, we temporarily assume all uncertainties are incorporated into f_a, and the true dynamic matrices M, C, and G in Eq. (1) can be precisely identified and utilized by both omniscient and actual controllers. As for the specific payload parametric uncertainty discussed in Sec. II.B, we also conform both control laws to similar structures, i.e., the known dynamics plus the unknown disturbances. So, in this case, the known dynamics will become nominal ones $\hat{M}$, $\hat{C}$, and $\hat{G}$ and unknown disturbances will turn into errors between the actual and nominal dynamics.

To follow a desired trajectory q_d, several variables relevant to errors are usually defined, including the tracking error $\tilde{q}$, the sliding surface s, and the virtual reference trajectory q_r.

$$\begin{aligned} \tilde{q} &= q - q_d \\ s &= \dot{\tilde{q}} + \Lambda\tilde{q} \\ \dot{q}_r &= \dot{q} - s = \dot{\tilde{q}}_d - \Lambda\tilde{q} \end{aligned} \tag{2}$$

where Λ is a constant positive definite gain matrix.

Recalling the principle of *certainty equivalence* [30], the designed control law aims to achieve two goals: 1) offset the dynamics by nominal terms, and 2) ensure stability with a negative gain term. In an ideal scenario, the *omniscient controller* τ_d^o would perfectly achieve these objectives by possessing knowledge of the ideal compensation to precisely counteract the disturbance f_a almost like cheating. However, the *actual controller* τ_d^a can never achieve such perfection, as it relies on an approximated compensation $\hat{f}_a$ with errors. Now, we will explore the impact of these discrepancies on closed-loop stability. The omniscient and actual control laws are denoted as follows:

$$\begin{aligned} \tau_d^o &= M\ddot{q}_r + C\dot{q}_r + G + f_a - Ks \\ \tau_d^a &= M\ddot{q}_r + C\dot{q}_r + G + \hat{f}_a - Ks \end{aligned} \tag{3}$$

where K is a positive definite gain matrix. Then, substitute both control laws into Eq. (1) to obtain the closed-loop dynamics:

$$M\dot{s} + (C+K)s = 0 \tag{4a}$$

$$M\dot{s} + (C+K)s = \tilde{f}_a \tag{4b}$$

where the disturbance estimation error is $\tilde{f}_a = \hat{f}_a - f_a$. Select a Lyapunov candidate function $V = (1/2)s^T Ms$ and notice the skew-symmetric property for $\dot{M} - 2C$ of typical systems like manipulators [31], then $\dot{V} = -s^T Ks < 0$ can be deduced to guarantee the stability of the omniscient controller. However, the nonzero residue $\tilde{f}_a$ will degrade the convergence of the tracking errors compared to the former, which motivates us to investigate its specific structure for a more accurate disturbance estimation.

B. Uncertain Disturbance Model for Manipulator

Before tackling the complex aerial manipulator system, we begin with one of its simpler subsystems to establish a unified rule. Consider a two-link planar manipulator with an unknown payload, as illustrated in Fig. 1. The inertial frame $\{e\}$ is established with the base

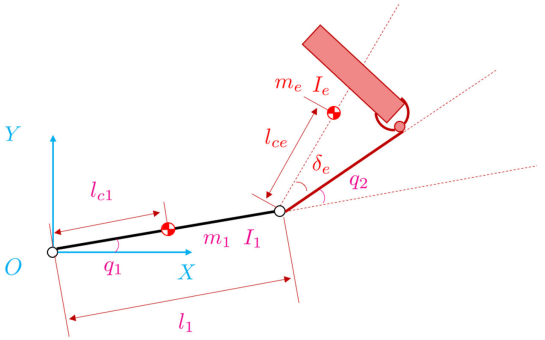

Fig. 1 Two-link manipulator carrying an unknown load.

joint serving as the origin O. In addition to the state and control input vectors introduced previously, the known parameters of the first link include its mass m_1, moment of inertia I_1, link length l_1, and length from the mass center to the base joint l_{c1}. To simplify the analysis, we assume the payload is rigidly attached. The second link is thus treated as an augmented link characterized by four unknown parameters: mass m_e, moment of inertia I_e, the distance from the mass center to the second joint l_{ce}, and the deviation angle δ_e relative to the original orientation of the second link. For brevity, denote all the unknown parameters as $\Theta^u = \{m_e, I_e, l_{ce}, \delta_e\}$.

Now, in the augmented system, the unknown parameters will introduce uncertainties into the actual dynamic matrices M, C, and G. Hence, the unknown disturbances are implicitly contained in the left-hand side of Eq. (5a). To maintain the control law structure similar to that in Sec. II.A, we design the omniscient controller by the known nominal dynamics $\hat{M}, \hat{C}$, and $\hat{G}$ and the ideal compensation f_a. In this case, the system dynamics and the omniscient control law can be expressed as follows:

$$M(q)\ddot{q} + C(q,\dot{q})\dot{q} + G(q) = \tau \tag{5a}$$

$$\tau_d^o = \hat{M}\ddot{q}_r + \hat{C}\dot{q}_r + \hat{G} + f_a - Ks \tag{5b}$$

$$\tau_d^a = \hat{M}\ddot{q}_r + \hat{C}\dot{q}_r + \hat{G} + \hat{f}_a - Ks \tag{5c}$$

Therefore, by substituting omniscient control law (5b) into the system dynamics equation (5a), the ideal compensation f_a can be derived as follows to achieve the ideally convergent error dynamics like Eq. (4a):

$$f_a = -(\tilde{M}\ddot{q}_r + \tilde{C}\dot{q}_r + \tilde{G}) \tag{6}$$

where $\tilde{M} = \hat{M} - M$ represents the estimation error between the nominal and actual dynamics, while $\tilde{C}$ and $\tilde{G}$ are defined similarly. Now, we will demonstrate that f_a can be expressed as the product of a kinematic regressor matrix and a parameter vector, which are associated with known states and unknown parameters, respectively. These can be deduced from Proposition 1.

Proposition 1: The general Lagrangian systems linear to dynamic parameters [32] (i.e., mass, inertia, length, etc.) ensure the transformation of the dynamic equation into a linear form, i.e., $M\ddot{q} + C\dot{q} + G = Y\Theta + b$. If a vector Θ containing all combination bases of dynamic parameters is found, the analytical expression of the kinematic regressor matrix Y can be subsequently determined.

Remark 2: For general Lagrangian systems, the relationship between dynamic parameters and state variables involves linear operations, including addition and scalar multiplication. The essence of matrix operations is linear operations. Therefore, for the left-hand side of the equations of motion $M\ddot{q} + C\dot{q} + G$, all dynamic parameters and their combinations can be grouped into a vector. Employing the principles of matrix operations, the left-hand side can be rearranged into an equivalent form: $Y\Theta + b$.

Based on all the found combination bases in the parameter vector Θ, the dynamics equation (5a) can be linearly transformed as

$$M(q)\ddot{q} + C(q,\dot{q})\dot{q} + G(q) = Y(q,\dot{q},\ddot{q}) \cdot \Theta(m_e, I_e, l_{ce}, \delta_e) + b(q,\ddot{q}) \tag{7a}$$

$$\Theta(m_e, I_e, \delta_e, l_{ce}) = \left[I_e,\ m_e,\ m_e l_{ce}^2,\ m_e l_{ce}\sin(\delta_e),\ m_e l_{ce}\cos(\delta_e) \right]^T \tag{7b}$$

where $Y(q,\dot{q},\ddot{q}) \in \mathbb{R}^{2\times 5}$ is the kinematic regressor matrix, $\Theta(m_e, I_e, \delta_e, l_{ce}) \in \mathbb{R}^{5\times 1}$ is the parameter vector, and $b(q,\ddot{q}) \in \mathbb{R}^{2\times 1}$ is the kinematic bias vector related to remaining dynamics. To derive the ideal compensation, first mark the states $\ddot{q}^*$ and $\dot{q}^*$ in the left side of Eq. (7a) as $M(q)\ddot{q}^* + C(q,\dot{q})\dot{q}^* + G(q)$. Then, replace the $\ddot{q}^*$ and $\dot{q}^*$ with $\ddot{q}_r$ and $\dot{q}_r$, respectively, to derive $M(q)\ddot{q}_r + C(q,\dot{q})\dot{q}_r + G(q)$, which can be further linearly transformed as Eq. (8). Accordingly, the regressor matrix $Y(q,\dot{q},\ddot{q})$ will be changed into $Y'(q,\dot{q},\dot{q}_r,\ddot{q}_r)$. This transformation process and the specific expressions of the related matrices are provided in Appendix A.

$$M(q)\ddot{q}_r + C(q,\dot{q})\dot{q}_r + G(q) = Y'(q,\dot{q},\dot{q}_r,\ddot{q}_r) \cdot \Theta(m_e, I_e, l_{ce}, \delta_e) + b(q,\ddot{q}) \tag{8}$$

Next, replace the true parameters m_e, I_e, l_{ce}, and δ_e with estimated parameters $\hat{m}_e, \hat{I}_e, \hat{l}_{ce}$, and $\hat{\delta}_e$ to get Eq. (9).

$$\hat{M}(q)\ddot{q}_r + \hat{C}(q,\dot{q})\dot{q}_r + \hat{G}(q) = Y'(q,\dot{q},\dot{q}_r,\ddot{q}_r) \cdot \hat{\Theta}(\hat{m}_e, \hat{I}_e, \hat{l}_{ce}, \hat{\delta}_e) + b(q,\ddot{q}) \tag{9}$$

Finally, the ideal compensation defined by Eq. (6) can be derived from Eq. (8) minus Eq. (9), where the parameter estimation error is $\tilde{m}_e = \hat{m}_e - m_e$ and so on for the rest.

$$f_a = -Y'(q,\dot{q},\dot{q}_r,\ddot{q}_r) \cdot \tilde{\Theta}(\tilde{m}_e, \tilde{I}_e, \tilde{l}_{ce}, \tilde{\delta}_e) \tag{10}$$

As Eq. (10) shows, the uncertain disturbance (or ideal compensation) can be decoupled into a regressor matrix related to the states of the system and a vector representing the parameter estimation errors. More intuitively, we can draw a physical interpretation that the uncertainty force and moment can be written as a function of the errors between estimated values and actual values of the physical terms (e.g., the payload mass, inertia), which reveals the physical source of the disturbance.

C. Uncertain Disturbance Model for Aerial Manipulator

Now, in the case of the aerial manipulator with an unknown payload, we can derive its uncertain disturbance model by following procedures similar to those employed for the manipulator.

1. *Lagrangian Dynamics*

Consider a quadrotor with a two-DOF manipulator, as shown in Fig. 2. The inertial frame is represented by $\{e\}$, while the body frame $\{b\}$ originates at the center of mass of the quadrotor. The link frame of

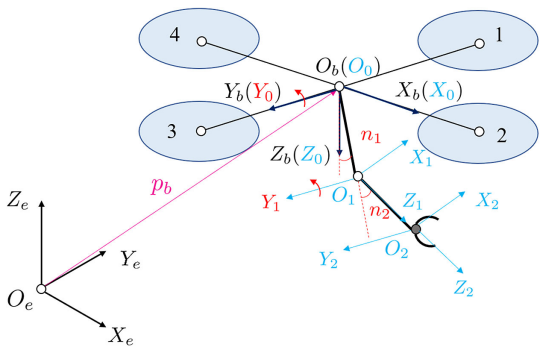

Fig. 2 Illustration of a quadrotor with a 2-DOF manipulator.

the manipulator $\{i\}$ is defined by the Denavit–Hartenberg convention [33], with its base origin O_0 coinciding with O_b. The position of O_b relative to the frame $\{e\}$ is denoted by $p_b = [x, y, z]^T$. The orientation of the quadrotor, described by roll-pitch-yaw Euler angles, is represented as $\Phi_b = [\phi, \theta, \psi]^T$. The joint angles of the manipulator are indicated by $n_m = [n_1, n_2]^T$. Then, the generalized coordinates of the system are expressed as $q = [p_b^T, \Phi_b^T, n_m^T]^T$.

Before attaching the unknown payload to the aerial manipulator, it is essential to establish the Lagrangian dynamics, determine the specific form of dynamic matrices, and investigate their relationships with the inertia parameters. Remember that the Lagrangian dynamics is characterized by Lagrangian $\mathcal{L} = \mathcal{K} - \mathcal{U}$, where $\mathcal{K}$ and $\mathcal{U}$ represent the total kinematic and potential energy of the system, respectively. Subsequently, the Euler–Lagrange equation can be formulated as

$$\frac{d}{dt}\frac{\partial \mathcal{L}}{\partial \dot{q}} - \frac{\partial \mathcal{L}}{\partial q} = \tau \tag{11}$$

where q represents generalized coordinates and τ denotes generalized forces. The total kinematic energy of the aerial manipulator comprises two aspects: the quadrotor part $\mathcal{K}_b$ and link parts $\mathcal{K}_{l_i}$, where $i \in \{1, \ldots, n\}$ denotes the index for each segment of the manipulator. Note that the kinematic energy includes both translational and rotational components. Therefore, the kinematic energy of the aerial manipulator can be expressed as follows:

$$\begin{aligned}
\mathcal{K} &= \mathcal{K}_b + \sum_{i=1}^{n} \mathcal{K}_{l_i} \\
\mathcal{K}_b &= \frac{1}{2} m_b \dot{p}_b^T \dot{p}_b + \frac{1}{2} \omega_b^T R_b H_b R_b^T \omega_b \\
\mathcal{K}_{l_i} &= \frac{1}{2} m_{l_i} \dot{p}_{l_i}^T \dot{p}_{l_i} + \frac{1}{2} \omega_{l_i}^T R_b R_{l_i}^b H_{l_i} R_b^{l_i} R_b^T \omega_{l_i}
\end{aligned} \tag{12}$$

where m_b and m_{l_i}, and H_b and H_{l_i} represent the mass and inertia matrix of the quadrotor and ith link of the manipulator, respectively. R_b and $R_{l_i}^b$ denote the rotation matrix from reference frames $\{b\}$ to $\{e\}$ and $\{i\}$ to $\{b\}$, respectively. The absolute translational and rotational velocities relative to $\{e\}$ are denoted by $\dot{p}_b$, $\dot{p}_{l_i}$, ω_b, and ω_{l_i}. Note that the absolute rotational velocity satisfies the transformation rules $\omega_b = T_b \dot{\Phi}_b$, where $\dot{\Phi}_b$ represents the Euler angle rate. Here, T_b is the transformation matrix satisfying $T_b = R_b Q$, and Q facilitates the mapping of $\omega_b^b = Q\dot{\Phi}_b$ from $\dot{\Phi}_b$ to the rotational velocity ω_b^b with respect to $\{b\}$. More detailed explanations of T_b and Q can be referred to in Appendix B.

The total potential energy $\mathcal{U}$ can be derived using the analogous procedures and formulated as

$$\begin{aligned}
\mathcal{U} &= \mathcal{U}_b + \sum_{i=1}^{n} \mathcal{U}_{l_i} \\
\mathcal{U}_b &= m_b g e_3^T p_b \\
\mathcal{U}_{l_i} &= m_{l_i} g e_3^T \left(p_b + R_b p_{bl_i}^b \right)
\end{aligned} \tag{13}$$

where $e_3 = [0, 0, 1]^T$ serves as a basis vector, and $p_{bl_i}^b$ denotes the center of mass for the ith link relative to frame $\{b\}$. Then, the Lagrangian dynamics of the aerial manipulator can be derived in the same form as Eq. (5a), i.e., $M(q)\ddot{q} + C(q, \dot{q})\dot{q} + G(q) = \tau$, by substituting the specific energy expressions of $\mathcal{K}$ and $\mathcal{U}$ into the Lagrangian $\mathcal{L}$. To be more precise, the inertia matrix $M \in \mathbb{R}^{n\times n}$ can be represented in the following 3×3 block form:

$$\begin{aligned}
M_{11} &= \left(m_b + \sum_{i=1}^{n} m_{l_i} \right) I_3 \\
M_{22} &= Q^T H_b Q + \sum_{i=1}^{n} \Big[m_{l_i} T_b^T S(R_b p_{bl_i}^b)^T S(R_b p_{bl_i}^b) T_b \\
&\qquad + Q^T R_{l_i}^b H_{l_i} R_b^{l_i} Q \Big] \\
M_{33} &= \sum_{i=1}^{n} \left(m_{l_i} J_P^{(l_i)^T} J_P^{(l_i)} + J_O^{(l_i)^T} R_{l_i}^b H_{l_i} R_b^{l_i} J_O^{(l_i)} \right) \\
M_{12} &= M_{21}^T = -\sum_{i=1}^{n} \Big[m_{l_i} S(R_b p_{bl_i}^b) T_b \Big] \\
M_{13} &= M_{31}^T = \sum_{i=1}^{n} \left(m_{l_i} R_b J_P^{(l_i)} \right) \\
M_{23} &= M_{32}^T = \sum_{i=1}^{n} \Big[Q^T R_{l_i}^b H_{l_i} R_b^{l_i} J_O^{(l_i)} - m_{l_i} T_b^T S(R_b p_{bl_i}^b)^T R_b J_P^{(l_i)} \Big]
\end{aligned} \tag{14}$$

where $I_3 \in \mathbb{R}^{3\times3}$ represents an identity matrix, and $J_P^{(l_i)}$ and $J_O^{(l_i)}$ denote the linear and angular velocity Jacobian matrices of the manipulator, respectively. The function $S(\cdot)$ represents the skew-symmetric operator.

The Coriolis matrix, $C \in \mathbb{R}^{n\times n}$, is computed from M, with each element c_{ij} located in the ith row and jth column.

$$c_{kj} = \sum_{i=1}^{n} \frac{1}{2} \left(\frac{\partial m_{kj}}{\partial q_i} + \frac{\partial m_{ki}}{\partial q_j} - \frac{\partial m_{ij}}{\partial q_k} \right) \dot{q}_i \tag{15}$$

where m_{ij} represents the element in M at the corresponding position. Thus, it can be safely concluded that C is determined by M from Eq. (15). Therefore, the combinations of inertia parameters in C, including mass, inertia, and length, are entirely inherited from M, due to the linear operation property of differentiation with respect to q.

The element in the gravitational vector can be computed as

$$g_i = \frac{\partial \mathcal{U}}{\partial q_i} \tag{16}$$

Similarly to the analysis of C, the combinations of inertia parameters in G are consequently inherited from those in $\mathcal{U}$.

Now, consider the aerial manipulator with an unknown payload attached to its end effector, akin to the manipulator studied in Sec. II. As shown in Fig. 3, the initially known inertia parameters of the second link become contaminated by the payload, resulting in an unknown parameter set $\Theta^u = \{m_e, H_{le}, l_{ce}, \delta_e\}$. Here, m_e, l_{ce}, $H_{le} = \mathrm{diag}(I_{xe}^{l_{ce}}, I_{ye}^{l_{ce}}, I_{ze}^{l_{ce}})$ represent the mass, center of mass, and the inertia matrix for the augmented second link, while δ_e denotes the corresponding deviation angle.

2. *Uncertain Disturbance Model*

To investigate the disturbance structure of the aerial manipulator with an unknown payload, we will reformulate the equations of motion into a linear form with respect to a parameter vector. Guided

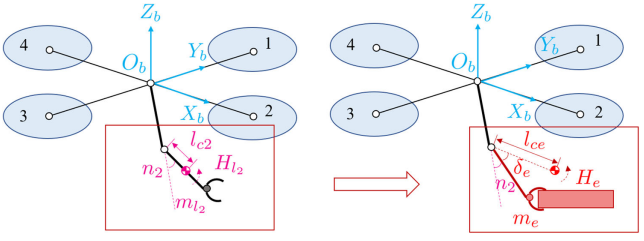

Fig. 3 A quadrotor with a 2-DOF manipulator carrying an unknown payload.

by Proposition 1, this parameter vector includes all combinations of unknown parameters. Furthermore, leveraging the specific expressions and the linearity of dynamic matrices in Eqs. (14–16) concerning inertia parameters, we only need to extract the combinations of unknown parameters from the basic terms in M. These are the terms affected by the unknown parameters in Eq. (14).

$$
\begin{aligned}
&m_{l_2} \to m_e \\
&H_{l_2} = \begin{bmatrix} I_x^{l_{c2}} & 0 & 0 \\ 0 & I_y^{l_{c2}} & 0 \\ 0 & 0 & I_z^{l_{c2}} \end{bmatrix} \to \begin{bmatrix} I_{xe}^{l_{ce}} & 0 & 0 \\ 0 & I_{ye}^{l_{ce}} & 0 \\ 0 & 0 & I_{ze}^{l_{ce}} \end{bmatrix} \\
&p_{bl_2}^b = R_1^b \cdot p_{12}^1 + R_2^b \cdot p_{bl_2}^2 \to R_1^0 \cdot p_{12}^1 + R_1^0 \cdot R_2^1(\delta_e) \cdot p_{bl_2}^2(l_{ce}) \\
&R_{l_2}^b = R_1^b \cdot R_2^1 \to R_1^0 \cdot R_2^1(\delta_e) \\
&J_P^{(l_2)} = \begin{bmatrix} z_0^b \times p_{bl_2}^b & z_1^b \times \left(R_2^b \cdot p_{bl_2}^2\right) \end{bmatrix} \\
&\quad \to \begin{bmatrix} z_0^b \times p_{bl_2}^b(\delta_e, l_{ce}) & z_1^b \times \left(R_1^0 \cdot R_2^1(\delta_e) \cdot p_{bl_2}^2(l_{ce})\right) \end{bmatrix}
\end{aligned} \tag{17}
$$

where $R_2^1(\delta_e)$ denotes the rotation matrix from frame $\{2\}$ to $\{1\}$, while $p_{12}^1 = [0, 0, l_1]^T$ and $p_{bl_2}^2(l_{ce}) = [0, 0, l_{ce}]^T$ represent the lengths of the first and second augmented links, respectively, measured in frames $\{1\}$ and $\{2\}$. Their specific forms are provided below:

$$
\begin{aligned}
R_2^1(\delta_e) &= \begin{bmatrix} \cos(n_2 + \delta_e) & 0 & \sin(n_2 + \delta_e) \\ 0 & 1 & 0 \\ -\sin(n_2 + \delta_e) & 0 & \cos(n_2 + \delta_e) \end{bmatrix} \\
p_{bl_2}^2 &= \begin{bmatrix} 0, 0, l_{ce} \end{bmatrix}^T
\end{aligned} \tag{18}
$$

Based on the source of the parametric uncertainties in Eq. (17), we can summarize all the patterns of unknown parameters in Table 1. Furthermore, we can simplify the analysis by transforming these uncertain patterns into simpler analysis targets, leveraging the properties of skew matrices as demonstrated in Eq. (19), where $S(\cdot)$ represents the skew-symmetric operator, while x and R denote the input vector and rotation matrix, respectively. It is worth noting that, for simplification purposes, we temporarily disregard the coefficient m_{l_2}, which will be reintroduced in the final parameter vector.

$$
\begin{aligned}
S(x) + S(x)^T &= 0 \\
S(Rx) &= R \cdot S(x) \cdot R^T
\end{aligned} \tag{19}
$$

The analysis targets prompt us to focus on fundamental terms such as $S(p_{bl_2}^b)$, $J_P^{(l_2)}$, and $R_2^l H_{l_2} R_l^2$. Then, our objective shifts to identifying all basic combinations of unknown parameters within these analysis targets.

Table 1 Analysis of all uncertain patterns in the inertia matrix M

Block components	Uncertain patterns	Analysis targets
M_{11}	m_{l_2}	m_{l_2}
M_{22}	$m_{l_2} S(R_b p_{bl_2}^b)^T S(R_b p_{bl_2}^b)$	$S(p_{bl_2}^b)^T S(p_{bl_2}^b)$
$M_{22}, M_{33}, M_{23}, M_{32}$	$m_{l_2} R_{l_2}^b H_{l_2} R_b^{l_2}$	$R_2^1 H_{l_2} R_1^2$
M_{33}	$m_{l_2} J_P^{(l_2)T} J_P^{(l_2)}$	$J_P^{(l_2)T} J_P^{(l_2)}$
M_{12}, M_{21}	$m_{l_2} S(R_b p_{bl_2}^b)$	$S(p_{bl_2}^b)$
M_{13}, M_{31}	$m_{l_2} J_P^{(l_2)}$	$J_P^{(l_2)}$
M_{23}, M_{32}	$m_{l_2} S(R_b p_{bl_i}^b)^T R_b J_P^{(l_2)}$	$S(p_{bl_2}^b) J_P^{(l_2)}$

Firstly, identify the unknown basis within $S(p_{bl_2}^b)$ and its input $p_{bl_2}^b(\delta_e, l_{ce})$.

$$
\begin{aligned}
p_{bl_2}^b(\delta_e, l_{ce}) &= R_1^0 \cdot R_2^1(\delta_e) \cdot p_{bl_2}^2(l_{ce}) + R_1^0 \cdot p_{12}^1 \\
&= R_1^0 \cdot P_1 \cdot B_1 + b_1 \\
&= P_2 \cdot B_1 + b_1
\end{aligned} \tag{20}
$$

where $B_1 = [l_{ce} \sin(\delta_e),\ l_{ce} \cos(\delta_e)]^T$ represents the unknown basis for $p_{bl_2}^b(\delta_e, l_{ce})$, P_1 and $P_2 = R_1^0 P_1$ denote the corresponding regressor matrices, and $b_1 = R_1^0 \cdot p_{12}^1$ represents a residual bias vector. Note that for a given input vector $p_{bl_2}^b(\delta_e, l_{ce}) = [a_1, a_2, a_3]^T$, its skew-symmetric matrix is defined as

$$
S(p_{bl_2}^b) = \begin{bmatrix} 0 & -a_3 & a_2 \\ a_3 & 0 & -a_1 \\ -a_2 & a_1 & 0 \end{bmatrix} \tag{21}
$$

As Eq. (20) suggests, each element a_i in $S(p_{bl_2}^b)$ is derived from the vector B_1 with unknown parameters. Therefore, the unknown basis for $S(p_{bl_2}^b)$ is

$$
V_1\{l_{ce} \sin(\delta_e),\ l_{ce} \cos(\delta_e)\} \tag{22}
$$

Then, we can calculate the product term $S(p_{bl_2}^b)^T S(p_{bl_2}^b)$.

$$
S(p_{bl_2}^b)^T S(p_{bl_2}^b) = \begin{bmatrix} a_2^2 + a_3^2 & -a_1 a_2 & -a_1 a_3 \\ -a_1 a_2 & a_1^2 + a_3^2 & -a_2 a_3 \\ -a_1 a_3 & -a_2 a_3 & a_1^2 + a_2^2 \end{bmatrix} \tag{23}
$$

Utilizing the expanded form of $p_{bl_2}^b$ to obtain the specific structure of a_i, we can derive that each element within matrix $S(p_{bl_2}^b)^T S(p_{bl_2}^b)$ can be constructed by the following unknown basis (refer to Appendix C for details).

$$
V_1 \cup V_2\{l_{ce}^2 \cos^2(\delta_e),\ l_{ce}^2 \sin^2(\delta_e),\ l_{ce}^2 \cos(\delta_e) \sin(\delta_e)\} \cup V_3\{l_{ce}^2\} \tag{24}
$$

Secondly, identify the unknown basis related to $J_P^{(l_2)}$ using the similar steps as for $S(p_{bl_2}^b)$. Retain the notations in Eq. (20) and recall that the cross product of two vectors can be represented as a matrix multiplication by introducing a skew matrix, i.e., $a \times b = S(a) \cdot b$. Thus, $J_P^{(l_2)}$ can be expressed as

$$
\begin{aligned}
J_P^{(l_2)} &= \begin{bmatrix} z_0^b \times p_{bl_2}^b(\delta_e, l_{ce}) & z_1^b \times \left(R_1^0 \cdot R_2^1(\delta_e) \cdot p_{b_2}^2(l_{ce})\right) \end{bmatrix} \\
&= \begin{bmatrix} S(z_0^b) \cdot (P_2 \cdot B_1 + b_1) & S(z_1^b) \cdot (P_2 \cdot B_1) \end{bmatrix} \\
&= \begin{bmatrix} P_3 \cdot B_1 + b_2 & P_4 \cdot B_1 \end{bmatrix}
\end{aligned} \tag{25}
$$

where $P_3 = S(z_0^b) P_2$ and $P_4 = S(z_1^b) P_2$ are the respective regressor matrices, and $b_2 = S(z_0^b) b_1$ is a bias vector relevant only to known states. As a result, the unknown basis of $J_P^{(l_2)}$ is equivalent to V_1. Before analyzing $J_P^{(l_2)T} J_P^{(l_2)}$, it is worth noting that the two rotational axes of joints are parallel, hence, $z_1^b = z_0^b$, or $P_3 = P_4$. Another property ${P_3}^T P_3 = I_2$ can be verified, where $I_2 \in \mathbb{R}^{2\times 2}$ denotes the identity matrix. Consequently, the product $J_P^{(l_2)T} J_P^{(l_2)}$ can be expressed as

$$
\begin{aligned}
J_P^{(l_2)T} J_P^{(l_2)} &= \begin{bmatrix} B_1^T \cdot P_3^T + b_2^T \\ B_1^T \cdot P_4^T \end{bmatrix} \cdot \begin{bmatrix} P_3 \cdot B_1 + b_2 & P_4 \cdot B_1 \end{bmatrix} \\
&= \begin{bmatrix} B_1^T B_1 + B_1^T P_3^T b_2 + b_2^T P_3 B_1 + b_2^T b_2 & B_1^T B_1 + b_2^T P_4 B_1 \\ B_1^T B_1 + B_1^T P_4^T b_2 & B_1^T B_1 \end{bmatrix}
\end{aligned} \tag{26}
$$

Notice that $B_1^T B_1 = l_{ce}^2$, and all terms in Eq. (27) are derived from either B_1 or $B_1^T B_1$. Thus, the unknown basis of $J_P^{(l_2)T} J_P^{(l_2)}$ should be $V_1 \cup V_3$.

As for the cross-term $S(p_{bl_2}^b) J_P^{(l_2)}$, each element should be analyzed. Therefore, the expanded form of $J_P^{(l_2)}$ can be given as follows:

$$J_P^{(l_2)} = \begin{bmatrix} c_{11} & c_{12} \\ 0 & 0 \\ c_{31} & c_{32} \end{bmatrix} \tag{27}$$

Then, the product term $S(p_{bl_2}^b) J_P^{(l_2)}$ can be represented as follows:

$$\begin{aligned} S(p_{bl_2}^b) J_P^{(l_2)} &= \begin{bmatrix} 0 & -a_3 & a_2 \\ a_3 & 0 & -a_1 \\ -a_2 & a_1 & 0 \end{bmatrix} \cdot \begin{bmatrix} c_{11} & c_{12} \\ 0 & 0 \\ c_{31} & c_{32} \end{bmatrix} \\ &= \begin{bmatrix} 0 & 0 \\ a_3 c_{11} - a_1 c_{31} & a_3 c_{12} - a_1 c_{32} \\ 0 & 0 \end{bmatrix} \end{aligned} \tag{28}$$

where a_2 is equal to zero. We can verify that the specific structure of c_{ij} can also be constructed by the unknown basis V_1. Therefore, both $a_3 c_{11} - a_1 c_{31}$ and $a_3 c_{12} - a_1 c_{32}$ can be represented using the unknown basis $V_1 \cup V_3$, as can the product term $S(p_{bl_2}^b) J_P^{(l_2)}$. Further details can be found in Appendix C.

Thirdly, by directly substituting Eqs. (17) and (18) to calculate the product, we can determine the unknown basis of $R_2^1 H_{l_2} R_1^2$ for each element. Please refer to Appendix C for its specific expression. Then, the unknown basis of $R_2^1 H_{l_2} R_1^2$ can be assembled as follows:

$$V_4 \left\{ \left\{ \sin^2(\delta_e),\ \cos^2(\delta_e),\ \sin(\delta_e)\cos(\delta_e) \right\} \times \left\{ I_{xe}^{l_{ce}},\ I_{ze}^{l_{ce}} \right\} \cup \left\{ I_{ye}^{l_{ce}} \right\} \right\} \tag{29}$$

where "×" denotes the Cartesian product of two sets.

Recall that the matrix C is derived from M, and $\mathcal{U}$ contains $p_{bl_2}^b$. Therefore, C will share the same unknown basis with M, and the unknown basis for G is V_1. Finally, with the guidance of Proposition 1, we can derive that the Lagrangian dynamics of the aerial manipulator with a payload can be reformulated into an equivalent linear form with respect to its dynamic parameters.

$$M(q)\ddot{q} + C(q,\dot{q})\dot{q} + G(q) = Y(q,\dot{q},\ddot{q}) \cdot \Theta + b \tag{30}$$

where Θ represents the parameter vector composed of all 14 unknown bases from the sets V_1, V_2, V_3, V_4 and $m_{1,2}$. For the case of aerial manipulators, the increased complexity of the M and C matrices impedes the specific analytical expression for the Y matrix. However, Y does exist and is fully determined given Θ. Similarly to the derivation procedures of the manipulator from Eqs. (8–10), the uncertain disturbance (or ideal compensation) of the aerial manipulator with a payload will share the same form.

$$f_a = -Y'(q,\dot{q},\dot{q}_r,\ddot{q}_r) \cdot \tilde{\Theta} \tag{31}$$

Therefore, for the aerial manipulator with a payload, the uncertain disturbance (or ideal compensation) can also be regarded as the product of a kinematic regressor matrix relevant to the states and commands, and a parameter vector stemming from the errors in estimating unknown parameters.

III. Meta Adaptive Control for Aerial Manipulator

In this section, we develop a meta-adaptive control leveraging the structure of the ideal compensation derived in Sec. II, aimed at improving trajectory tracking precision for the aerial manipulator. Subsequently, the stability of the closed-loop system is guaranteed by Lyapunov analysis.

A. Meta-Adaptive Control

The meta-adaptive control is implemented in two steps similar to [29]. Firstly, offline meta-learning is employed to train an approximator Γ of the kinematic regressor in a data-efficient way. Secondly, the online composite adaptive control law is devised to estimate the unknown parameters a.

1. *Meta-Learning and Adaptation Goal*

The specific analytical form of the regressor matrix in Eq. (31) is hard to calculate for the entire aerial manipulator system in practice, despite its theoretical existence. Furthermore, the regressor matrix is essential to implement a controller capable of direct compensation for disturbances at the system level. Therefore, we employ neural networks to approximate the intractable regressor matrix, enabling the controller to counteract unknown disturbances with acceptable errors. Following the structure of the ideal compensation in Eq. (31), the parameterized compensation can be expressed as

$$f_a = Y'(q,\dot{q},\dot{q}_r,\ddot{q}_r) \cdot \tilde{\Theta} \approx \Gamma(q,\dot{q},\ddot{q};W) \cdot a \tag{32}$$

where $\Gamma : \mathbb{R}^n \times \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}^{n \times m}$ denotes a neural network to approximate the kinematic regressor parameterized by weights W, and $a \in \mathbb{R}^m$ represents corresponding parameters. Note that for a specific task, the state q at time k is actually affected by q_d at time $k-1$, so the task-relevant information is indirectly contained in the state.

Denote the maximum representation error $\bar{d}$ of the approximation as

$$\bar{d} = \min_a \max_{q,\dot{q},\ddot{q} \in S} \left\| Y'(q,\dot{q},\dot{q}_r,\ddot{q}_r) \cdot \tilde{\Theta} - \Gamma(q,\dot{q},\ddot{q};W) \cdot a \right\| \tag{33}$$

whose existence is guaranteed under the assumption of the bounded Lipschitz constant of f_a and the bounded training error in the feasible domain S of states.

2. *Offline Meta-Learning*

The precollect data $D = \{D_1, D_2, \ldots, D_N\}$ in N different grasping and tracking conditions $\{e_1, e_2, \ldots, e_N\}$ are required to train the approximator Γ. Each subset $D_i (i = 1, 2, \ldots, N)$ contains M pairs of (x_j^i, y_j^i) sampled from the trajectory T_i, where $x_j^i = [q_j^i, \dot{q}_j^i, \ddot{q}_j^i]$ is the sampled state, $y_j^i = -(f_a)_j^i$ is the label for compensation force and $j = 1, 2, \ldots, M$ is the time stamp.

The training objective is quantified by the loss function. On the one hand, the regression loss contributes to optimal weights W^* for precise approximation of f_a. On the other hand, the regulation loss constrains the approximator Γ for domain invariance. Then, the optimization problem can be defined as

$$\begin{aligned} W^* = \max_h \min_{\Gamma,a} & \sum_{i=1}^{N} \sum_{j=1}^{M} \left\| \Gamma(x_j^i; W) \cdot a_i - y_j^i \right\|^2 \\ & + \alpha \cdot \log\left(h\left(\Gamma(x_j^i; W)\right)^T \cdot e_i \right) \end{aligned} \tag{34}$$

where h represents another neural network functioning as a classifier to discriminate among various grasping conditions using one-hot labels $\{e_1, e_2, \ldots, e_N\}$. This network is adversarially trained to ensure domain invariance across scenarios with different payload settings but the identical desired trajectory. Note that conditions with the same desired trajectory share the same label. Therefore, the learned approximator can capture the independent influence of system states on the disturbance from the trainset and will exhibit improved generalization to the unseen testing scenarios.

Now, the optimization problem (34) can be solved by the following meta-learning algorithm, where Γ, h are updated by gradient descent in the outer loop and a is updated by least-square in the inner loop. In a training epoch, each D_i will be divided into two nonoverlapped subsets D_i^a and D_i^t for adapting a and training Γ, respectively. The main processes include the following: 1) adaptation step solves a_i^* in

condition e_i on dataset D_i^a in the inner loop; 2) the training step updates the parameters of the approximator Γ on D_i^t using solved a_i^*; 3) the regulation step updates the discriminator h on D_i^t. It is worth noting that normalizing a^* and the spectral normalization for Γ will benefit the robustness and Lipschitz property for control design. More details of the training procedures can be found in [29].

3. Online Adaptive Control

The offline meta-learning process yields a precise approximator with optimal parameters $\Gamma(W^*)$, effectively minimizing the prediction error of f_a by solving a^* through the least-square method. However, the control objective also involves minimizing the tracking error, thereby requiring a composite adaptive law to update a. Thus, the control system is divided into two parts: a *nonlinear control law* τ_d and an *adaptive law* $\hat{a}$. For brevity, the independent variables in matrices and vectors are omitted.

To formulate the control problem, we adopt the commonly used tracking errors defined in Eq. (2) and consider the dynamics of the aerial manipulator with the same form as Eq. (5a). Then, the nonlinear control law augmented by the approximated compensation can be formulated as

$$\tau_d = \hat{M}\ddot{q}_r + \hat{C}\dot{q}_r + \hat{G} - Ks - \Gamma\hat{a} \tag{35}$$

where $\hat{M}\ddot{q}_r + \hat{C}\dot{q}_r + \hat{G}$ represents the nominal dynamics, K is a positive definite gain matrix, and $\hat{a}$ denotes the estimated parameters. If the representation error is denoted as $d(q, \dot{q}, \ddot{q}) = Y'(q, \dot{q}, \dot{q}_r, \ddot{q}_r) \cdot \tilde{\Theta} - \Gamma(q, \dot{q}, \ddot{q}; W) \cdot a$, then the closed-loop dynamics can be derived by substituting Eq. (35) into Eq. (5a):

$$M\dot{s} + (C + K)s + \Gamma\tilde{a} = d \tag{36}$$

The adaptive law contains a prediction error term and a tracking error term and can be expressed as

$$\dot{\hat{a}} = -P\Gamma^T(\Gamma\hat{a} - \hat{y}) + P\Gamma^T s \tag{37a}$$

$$\dot{P} = -P\Gamma^T\Gamma P \tag{37b}$$

The prediction error term is analogous to an online least-square estimator, i.e., the first part in Eq. (37a) with an evolved covariance matrix P. This term contributes to reducing the cumulative error between the estimated compensation $\Gamma\hat{a}$ and the measured compensation $\hat{y}$ via finite difference. In addition, the tracking error term ensures the stability of the closed-loop system, i.e., the second component in Eq. (37a) relevant to the tracking error s.

B. Lyapunov Stability Analysis

Now, the stability analysis of the proposed control scheme will be verified by the Lyapunov theory.

Theorem 1: Given the dynamics of the aerial manipulator in Eq. (5a), along with the nonlinear control law (35) and the adaptive law (37), the norm of the augmented tracking error $[s^T, \tilde{a}^T]^T$ will exponentially converge within the bound specified in Eq. (38) at a rate α.

$$\lim_{t\to\infty}\left\|\begin{bmatrix} s \\ \tilde{a}\end{bmatrix}\right\| \leqslant \frac{1}{\alpha \min[\lambda_{\min}(M), \lambda_{\min}(P^{-1})]}\left(\sup_t\|d\| + \sup_t\|\Gamma^T\epsilon\|\right) \tag{38}$$

where $\lambda_{\min/\max}(\cdot)$ represents the minimum (maximum) eigenvalue of the matrix and α is a constant and ϵ is defined as $\hat{y} - \Gamma a$.

Proof: Combine the closed-loop dynamics of both trajectory tracking Eq. (36) and parameter estimation Eq. (37) into a more compact form:

$$\begin{bmatrix} M & 0 \\ 0 & P^{-1}\end{bmatrix}\cdot\begin{bmatrix}\dot{s} \\ \dot{\tilde{a}}\end{bmatrix} + \begin{bmatrix} C+K & \Gamma \\ -\Gamma^T & \Gamma^T\Gamma\end{bmatrix}\cdot\begin{bmatrix} s \\ \tilde{a}\end{bmatrix} = \begin{bmatrix} d \\ \Gamma^T\epsilon\end{bmatrix} \tag{39}$$

Select the Lyapunov candidate function $\mathcal{V}$ as

$$\mathcal{V} = \begin{bmatrix} s \\ \tilde{a}\end{bmatrix}^T\begin{bmatrix} M & 0 \\ 0 & P^{-1}\end{bmatrix}\begin{bmatrix} s \\ \tilde{a}\end{bmatrix} \tag{40}$$

To simplify the symbolic notations, define $\mathcal{V} = x^T\mathcal{M}x$, where $x = [s^T, \tilde{a}^T]^T$, then we can derive

$$\mathcal{M} = \begin{bmatrix} M & 0 \\ 0 & P^{-1}\end{bmatrix} \tag{41}$$

Equation (39) can be reformulated as

$$\mathcal{M}\dot{x} = -\begin{bmatrix} C+K & \Gamma \\ -\Gamma^T & \Gamma^T\Gamma\end{bmatrix}\cdot x + \begin{bmatrix} d \\ \Gamma^T\epsilon\end{bmatrix} \tag{42}$$

Now, we can compute $\dot{\mathcal{V}}$ using Eq. (42).

$$\begin{aligned}\dot{\mathcal{V}} &= 2x^T\mathcal{M}\dot{x} + x^T\dot{\mathcal{M}}x \\ &= -2\begin{bmatrix} s \\ \tilde{a}\end{bmatrix}^T\cdot\begin{bmatrix} C+K & \Gamma \\ -\Gamma^T & \Gamma^T\Gamma\end{bmatrix}\cdot\begin{bmatrix} s \\ \tilde{a}\end{bmatrix} + 2\begin{bmatrix} s \\ \tilde{a}\end{bmatrix}^T\cdot\begin{bmatrix} d \\ \Gamma^T\epsilon\end{bmatrix} \\ &\quad + \begin{bmatrix} s \\ \tilde{a}\end{bmatrix}^T\cdot\begin{bmatrix}\dot{M} & 0 \\ 0 & \dot{P}^{-1}\end{bmatrix}\cdot\begin{bmatrix} s \\ \tilde{a}\end{bmatrix} \\ &= -2\begin{bmatrix} s \\ \tilde{a}\end{bmatrix}^T\cdot\begin{bmatrix} K & 0 \\ 0 & 1/2\Gamma^T\Gamma\end{bmatrix}\cdot\begin{bmatrix} s \\ \tilde{a}\end{bmatrix}^T + 2\begin{bmatrix} s \\ \tilde{a}\end{bmatrix}^T\cdot\begin{bmatrix} d \\ \Gamma^T\epsilon\end{bmatrix}\end{aligned} \tag{43}$$

Remark 3: The derivation of Eq. (43) employs two simplification techniques: 1) the skew-symmetric property of matrix $\dot{M} - 2C$ renders the corresponding quadratic terms to zero; 2) for the first term in Eq. (43), the off-diagonal elements of the quadratic form matrix can be eliminated because of $s^T\Gamma\tilde{a} - \tilde{a}^T\Gamma^T s = 0$. 3) Notice $\dot{P}^{-1} = -P^{-1}\dot{P}P^{-1}$.

As K, $\Gamma^T\Gamma$, M and P^{-1} are uniformly positive definite and uniformly bounded [29], there exists some $\alpha > 0$ to bound the first diagonal block matrix in the simplified result of Eq. (43), where α is determined by the eigenvalues of the block matrices as presented in Eq. (44). In addition, the second term can be bounded using the Cauchy-Schwartz inequality, along with the definition of the upper bound Eq. (46) for the disturbance term. Then, we can derive the following inequality for $\dot{\mathcal{V}}$:

$$\begin{aligned}\dot{\mathcal{V}} &\leqslant -2\alpha\begin{bmatrix} s \\ \tilde{a}\end{bmatrix}^T\cdot\begin{bmatrix} M & 0 \\ 0 & P^{-1}\end{bmatrix}\cdot\begin{bmatrix} s \\ \tilde{a}\end{bmatrix}^T + 2\begin{bmatrix} s \\ \tilde{a}\end{bmatrix}^T\cdot\begin{bmatrix} d \\ \Gamma^T\epsilon\end{bmatrix} \\ &\leqslant -2\alpha\mathcal{V} + 2\left\|\begin{bmatrix} s \\ \tilde{a}\end{bmatrix}\right\|\cdot\left\|\begin{bmatrix} d \\ \Gamma^T\epsilon\end{bmatrix}\right\| \\ &\leqslant -2\alpha\mathcal{V} + 2\sqrt{\frac{\mathcal{V}}{\lambda_{\min}(\mathcal{M})}}D\end{aligned} \tag{44}$$

where

$$\alpha = \min\left(\frac{\lambda_{\min}(K)}{\lambda_{\max}(M)}, \frac{1/2\lambda_{\min}(\Gamma^T\Gamma)}{\lambda_{\max}(P^{-1})}\right) \tag{45}$$

and

$$D = \sup_t\left\|\begin{bmatrix} d \\ \Gamma^T\epsilon\end{bmatrix}\right\| \tag{46}$$

Introduce a transformation $\mathcal{W} = \sqrt{\mathcal{V}}$, from which we can derive $\mathcal{V} = \mathcal{W}^2$ and $\dot{\mathcal{V}} = 2\dot{\mathcal{W}}\mathcal{W}$. Then, the inequality (44) can be reformulated using these equations.

$$\begin{aligned}\dot{\mathcal{V}} &\leqslant -2\alpha\mathcal{V} + 2\sqrt{\frac{\mathcal{V}}{\lambda_{\min}(\mathcal{M})}}D \\ \dot{\mathcal{W}} &\leqslant -\alpha\mathcal{W} + \frac{D}{\sqrt{\lambda_{\min}(\mathcal{M})}}\end{aligned} \tag{47}$$

Applying the comparison lemma [34], we can derive

$$\sqrt{\mathcal{V}} = \mathcal{W} \leqslant e^{-\alpha t}\left(\mathcal{W}(0) - \frac{D}{\alpha\sqrt{\lambda_{\min}(\mathcal{M})}}\right) + \frac{D}{\alpha\sqrt{\lambda_{\min}(\mathcal{M})}} \tag{48}$$

By utilizing the relationship between a quadratic term and its eigenvalues, $\lambda_{\min}(\mathcal{M})\|x\|^2 \leqslant \mathcal{V}$, the augmented tracking error x will be bounded.

$$\begin{aligned}\sqrt{\lambda_{\min}(\mathcal{M})}\|x\| \leqslant \sqrt{\mathcal{V}} &\leqslant e^{-\alpha t}\left(\mathcal{W}(0) - \frac{D}{\alpha\sqrt{\lambda_{\min}(\mathcal{M})}}\right) + \frac{D}{\alpha\sqrt{\lambda_{\min}(\mathcal{M})}} \\ \|x\| &\leqslant e^{-\alpha t}\left(\frac{\mathcal{W}(0)}{\sqrt{\lambda_{\min}(\mathcal{M})}} - \frac{D}{\alpha\lambda_{\min}(\mathcal{M})}\right) + \frac{D}{\alpha\lambda_{\min}(\mathcal{M})}\end{aligned} \tag{49}$$

where $\mathcal{W}(0)$ represents the initial value of $\mathcal{W}$ at time $t = 0$. Finally, by taking the limit as $t \to \infty$ on both sides, Theorem 1 can be proven.

IV. Experiments

We implemented and evaluated the performance of the meta-adaptive controller using MATLAB R2023a and SIMULINKS, and trained the approximated kinematic regressor with PyTorch 1.8. Our experiments aim to validate the improved track performance and the robustness of the meta-adaptive controller across various desired commands and payload settings. The first experiment involves tracking two types of trajectories. One is randomly generated by Minimum Snap [35] (abbreviated as MinSnap), while another is a figure-8 trajectory. In the second experiment, the uncertain parameters of the payload range from the seen to the unseen training dataset. All these experiments are compared with the baseline, including a PID and a conventional adaptive controller [36].

A. Dataset Generation and Network Training

The processes of training the kinematic regressor Γ are critical for the well-performed meta-adaptive controller. Now, we provide the details of the experiment settings.

The reference trajectories and uncertainties are specified for preparation. The desired trajectories consist of two components. For the quadrotor, the reference trajectory involves time polynomials $x_d(t), y_d(t), z_d(t), \psi_d(t)$ solved by Minimum Snap [35] with randomly generated waypoints and the subsequently determined attitude angle $\phi_d(t), \theta_d(t)$. For the manipulator, the desired command is represented by the time domain step response of the first-order inertial element $n_{1d}(t), n_{2d}(t)$. Additionally, we introduce all unknown parameters of the payload $\Theta^u = \{m_e, H_{le}, l_{ce}, \delta_e\}$ to the aerial manipulator that shared the same inertia parameters as that in [36]. Since the payload mass will impact all the uncertain patterns in Table 1 and contribute the most to the uncertainty, we characterize the uncertainty with various payload masses and other unknown parameters fixed. To be more specific, we use a linear range of $m_o \in \{0.1,\ 0.2,\ 0.3,\ 0.4,\ 0.5\}$ kg and fix the others to $I^{l_o}_{xe} = I^{l_o}_{ye} = I^{l_o}_{ze} = 0.001\ \text{kg}\cdot\text{m}^2$, $l_o = 0.05$ m, and $\delta_e = 25°$ for generating training dataset. Note that the augmented inertia parameters of the second link are obtained by adding its previous setting to the payload, i.e., $m_e = m_2 + m_o$, etc. Moreover, considering the sensor noise and errors [37], we simulate the measurement noise for accelerations as Gaussian white noise with a variance of 0.2.

Next, we will introduce the dataset generation and details about the neural network. The training data are sampled from the trajectories guided by the baseline controller [36]. The compensation for disturbances is directly estimated from the integration of the error feedback $\hat{f}_a = -\int K_a s \cdot \mathrm{d}t$, where K_a is a diagonal gain matrix. We simulate fifty 10-second-long trajectories for each m_o and collected 120 samples for each trajectory. The inputs for the network consist of ϕ, θ, ψ, n_1, and n_2 selected from q, which aims for positional error compensation along the x, y, z directions, based on the form of M_{11}. The velocity $\dot{q}$ and commands $\dot{q}_d$ and $\ddot{q}_d$ are included as well. Then, we model each row of the kinematic regressor Γ using a multilayer perceptron (MLP), which contains 20-30-20 hidden nodes across its three layers. The network is trained by the aforementioned procedures and inserted into the meta-adaptive control law in Eqs. (35) and (37).

B. Trajectory Tracking Performance

In this experiment, we utilize the trained meta-adaptive controller to track two kinds of desired trajectories, namely, the randomly generated MinSnap and figure-8. To account for the uncertainty, we securely set the payload mass m_o to 0.1 kg within the range of the training dataset, and other parameters the same as those in Sec. VI.A. Note that the sequential waypoints required to generate the MinSnap trajectory are entirely random. Hence, both MinSnap and figure-8 trajectories for testing have not been seen for the networks. Furthermore, we evaluate the tracking performances of PID, baseline adaptive, and meta-adaptive controllers using the gain matrices specified in Table 2. We implement the PID controller by employing the PID-form combination of errors to estimate the disturbance compensation, denoted as $\hat{f}_a = -(K_p\tilde{q} + \int K_i\tilde{q} + K_d\dot{\tilde{q}}) \cdot \mathrm{d}t$, since the system will diverge under a naive PID control law.

Figures 4 and 5 display the trajectories in both 3D and 2D versions. The meta-adaptive controller (red) outperforms both the PID controller (pink) and the baseline adaptive controller (blue) in tracking the MinSnap trajectory (green). As shown in Figs. 6 and 7, the proposed controller effectively reduces the tracking error in the x, y, z channels by accurately predicting disturbances. This validates the network's precise approximation and the stability ensured by the online adaptive law. According to Table 3, the meta-adaptive controller shows the lowest root-mean-squared error (RMSE). Similar conclusions can be inferred from the results of figure-8 trajectory tracking. The tracking performance is illustrated in Figs. 8–10, while the disturbance

Table 2 List of gain matrices

Controller	Gain
PID	$K = 18I$ $K_p = 3I$ $K_i = 3I$ $K_d = 2I$
Baseline adaptive	$K = 18I$ $K_a = 8I$
Meta adaptive	$K = 18I$

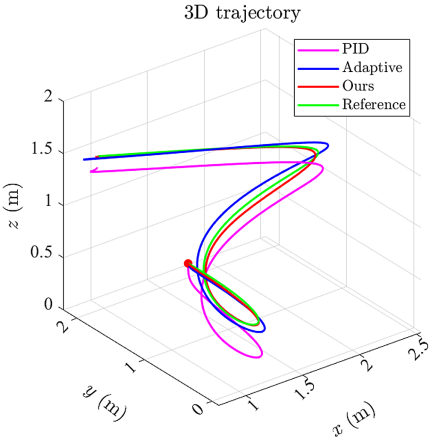

Fig. 4 MinSnap trajectory tracking in 3D view.

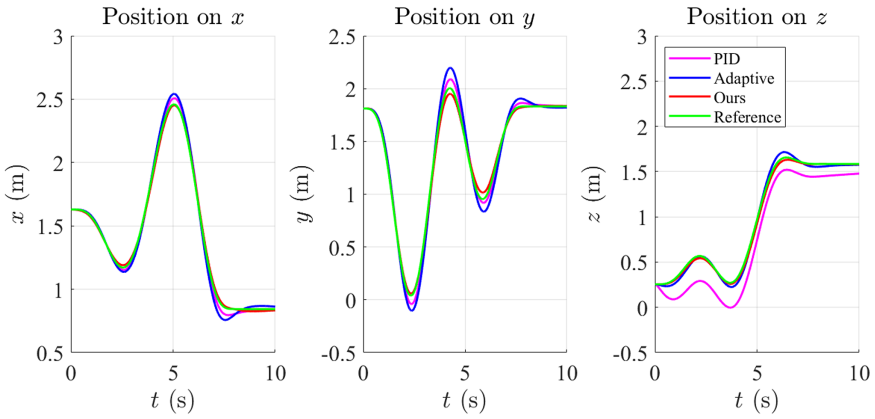

Fig. 5 MinSnap trajectory tracking in each channel.

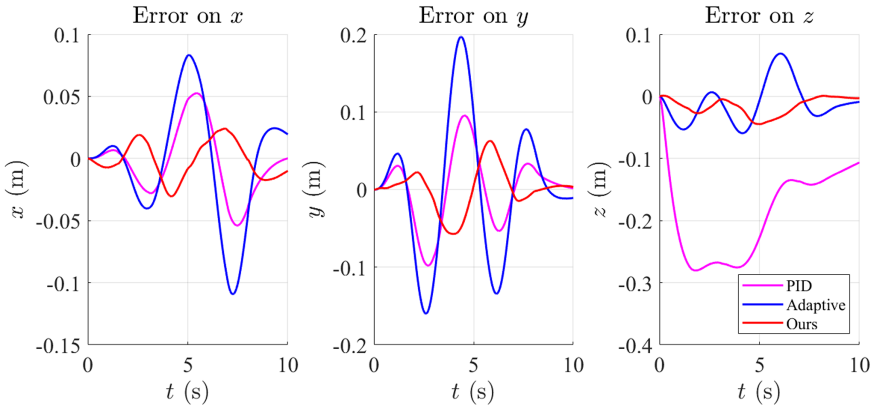

Fig. 6 MinSnap trajectory tracking error in each channel.

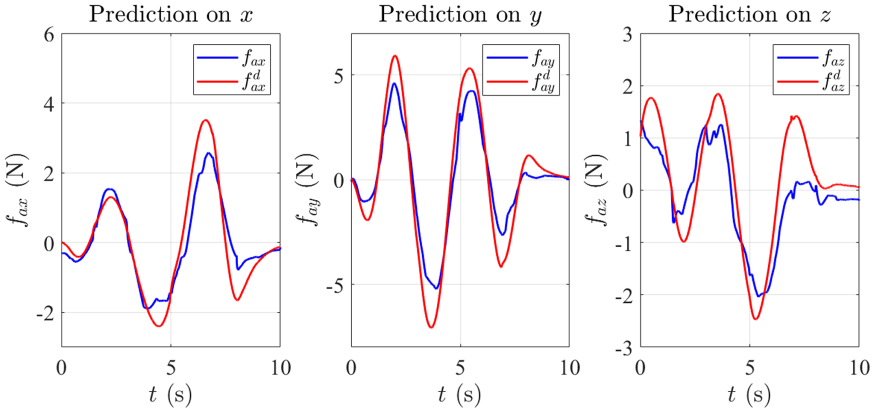

Fig. 7 MinSnap disturbance prediction in each channel.

Table 3 Tracking RMSE within training dataset mass distribution

	MinSnap RMSE, m			Figure-8 RMSE, m		
Controller	e_x	e_y	e_z	e_x	e_y	e_z
PID	0.0275	0.0461	0.2024	0.1493	0.1470	0.1854
Baseline adaptive	0.0471	0.0885	0.0352	0.1353	0.0486	0.0193
Meta adaptive	**0.0147**	**0.0276**	**0.0211**	**0.0610**	**0.0286**	**0.0179**

The lowest RMSE is highlighted in each column.

predictions are given in Fig. 11. Additionally, relevant statistics are listed in Table 3. Regarding the results of the PID controller, the additional proportional and differentiating terms contribute to a slight improvement compared to the baseline adaptive controller in the horizontal direction when there are fewer disturbances. However, the convergence-guaranteed adaptive law ensures system stability in the vertical direction, especially under major disturbances.

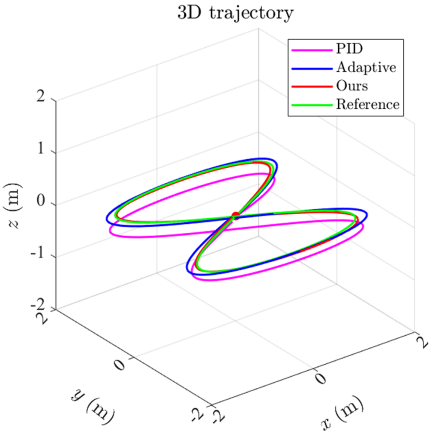

Fig. 8 Figure-8 trajectory tracking in 3D view.

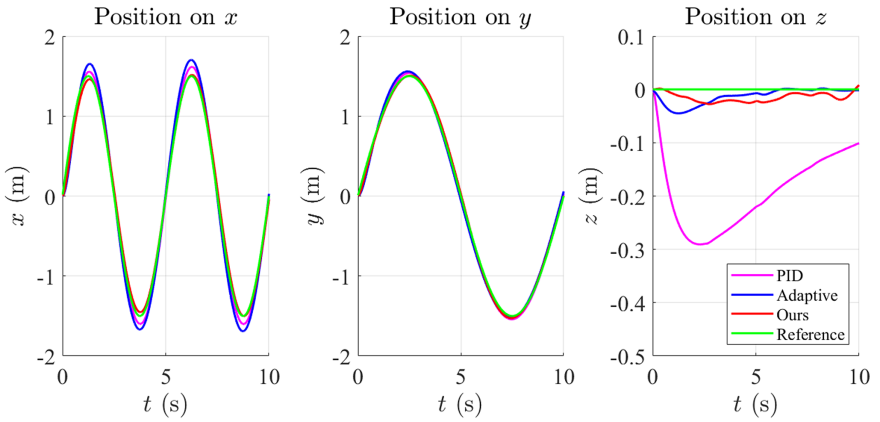

Fig. 9 Figure-8 trajectory tracking in each channel.

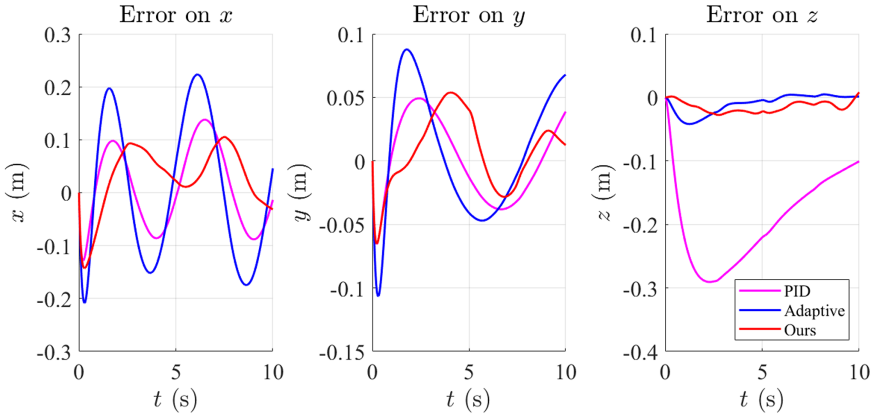

Fig. 10 Figure-8 trajectory tracking error in each channel.

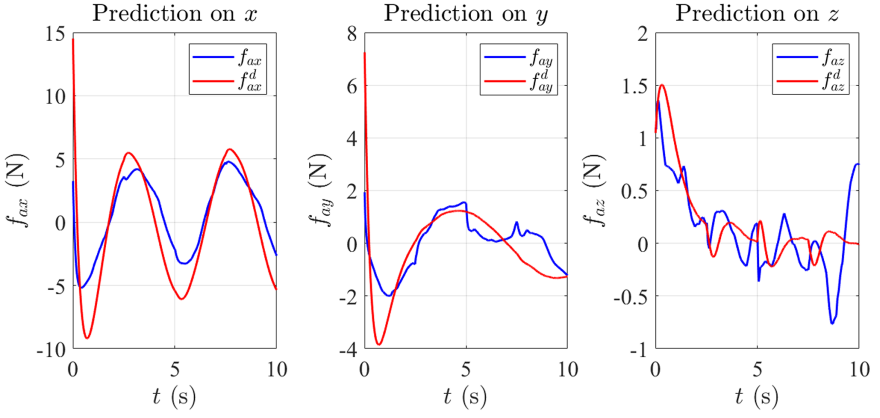

Fig. 11 Figure-8 disturbance prediction in each channel.

C. Robustness to Variant Uncertainty

In this experiment, we account for the significant impact of payload mass on unknown disturbances by characterizing uncertainty using a wider range of payload masses that extends outside the domain of the training dataset. Initially, we attempted m_o slightly beyond the training dataset as 0.6 kg while tracking both MinSnap and figure-8 trajectory. Akin to the first experiment, the improved tracking performance of the proposed controller for MinSnap trajectory can be observed in Figs. 12–14. The capability of the meta-adaptive controller to predict disturbances is verified in Fig. 15. In addition, the outcomes for the figure-8 tracking are depicted in Figs. 16–19. Our proposed controller achieves the lowest RMSE in all tracking directions, as shown in Table 4. It effectively reduces the RMSE by about 0.24 m in the vertical direction z compared with the PID controller, and about 0.02–0.11 m compared with the baseline adaptive controller. This preliminarily demonstrates the generalization capability of the proposed controller for scenarios involving unseen masses.

To further investigate the generalization ability of the proposed controller, we uniformly sampled the payload masses from 0.02 to 1.00 kg with 50 candidates and randomly generated 10 MinSnap trajectories at each sampled m_o. Then, we visualize the tracking RMSE surfaces in each x, y, z channel in Fig. 20. In this 3D plot, the horizontal axes represent different payload masses and simulation cases, respectively. The vertical gray plane divides the simulation scenarios into two regions: the left part belongs to the training distribution, and the right does not. The mean RMSE curves across all simulations at each m_o are projected to the front side, with the red curve representing the meta-adaptive controller, the pink one for the PID controller, and the blue one for the baseline adaptive controller.

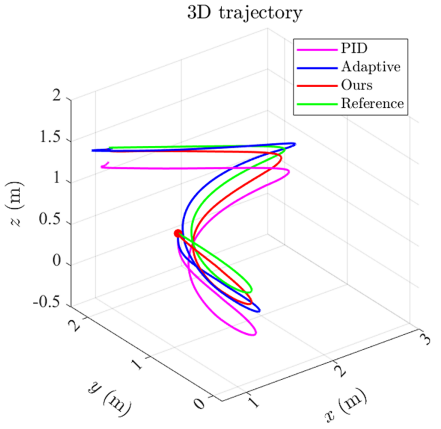

Fig. 12 MinSnap trajectory tracking in 3D view.

The results indicate that both the PID and the baseline adaptive controllers exhibit a larger RMSE at each channel (top surface). In contrast, the proposed controller not only achieves a smaller RMSE (lower surface) with less variance but also maintains stable performance even when the payload masses extend far beyond the distribution of the training dataset. This is further illustrated by the results in Fig. 21, which depicts the fluctuating range of RMSE, with the bold mean RMSE curve in the center bounded by its minimum and maximum values. However, for the z axis, the RMSEs of all controllers increase due to the gravity of the heavier payload. Note that the proposed controller consistently outperforms the other two benchmark controllers.

V. Conclusions

The uncertainty of the aerial manipulator with an unknown payload has been investigated to achieve more precise disturbance

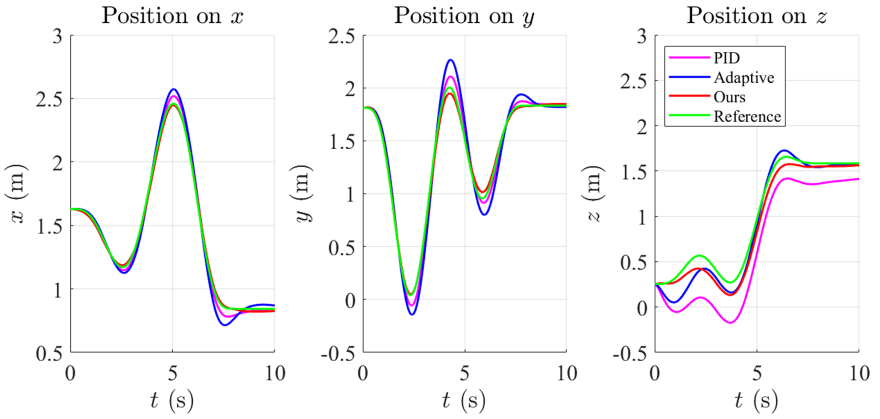

Fig. 13 MinSnap trajectory tracking in each channel.

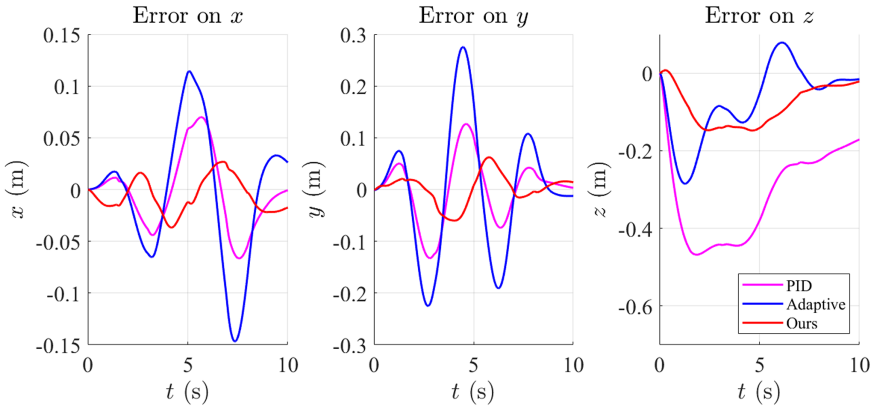

Fig. 14 MinSnap trajectory tracking error in each channel.

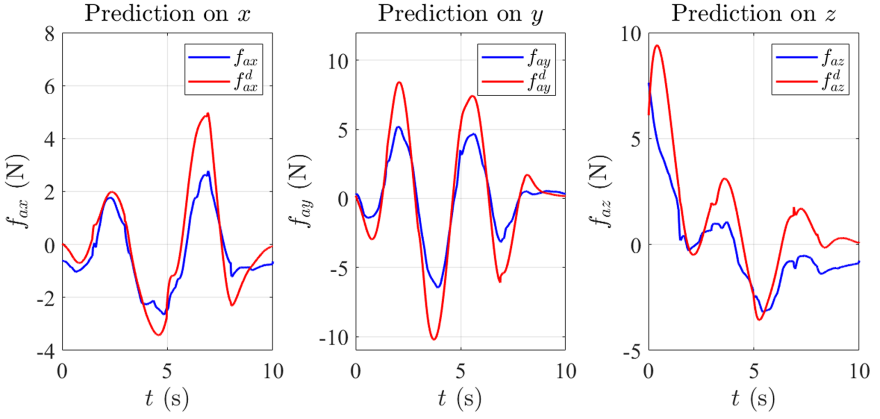

Fig. 15 MinSnap disturbance prediction in each channel.

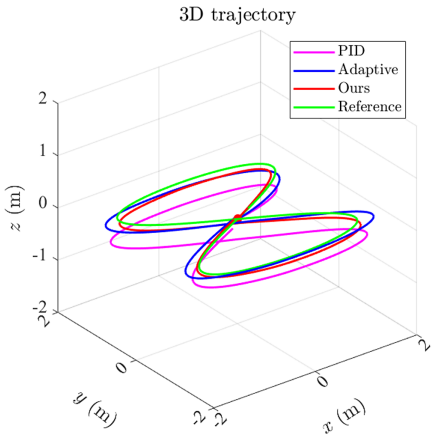

Fig. 16 Figure-8 trajectory tracking in 3D view.

Table 4 Tracking RMSE out of training dataset mass distribution

	MinSnap RMSE, m			Figure-8 RMSE, m		
Controller	e_x	e_y	e_z	e_x	e_y	e_z
PID	0.0360	0.0623	0.3340	0.1025	0.0401	0.3374
Adaptive	0.0641	0.1246	0.1189	0.1826	0.0597	0.1134
Meta adaptive	**0.0176**	**0.0292**	**0.0979**	**0.0724**	**0.0340**	**0.0959**

The lowest RMSE is highlighted in each column.

compensation. We consider both the center of mass position and the deviation angle of the payload, treating it as a component of the augmented system to ensure more accurate dynamic modeling. The linear parameterization technique is adopted to decompose the uncertainty into a kinematic regressor matrix and a parameter vector related to known states and unknown parameters, respectively, which facilitates

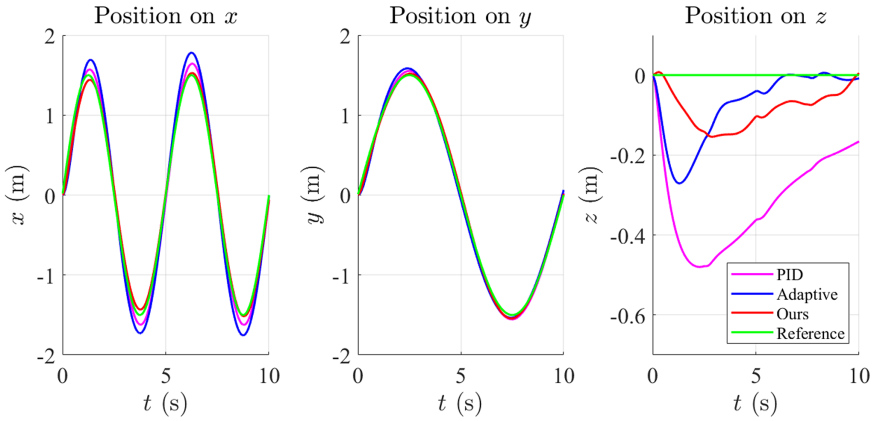

Fig. 17 Figure-8 trajectory tracking in each channel.

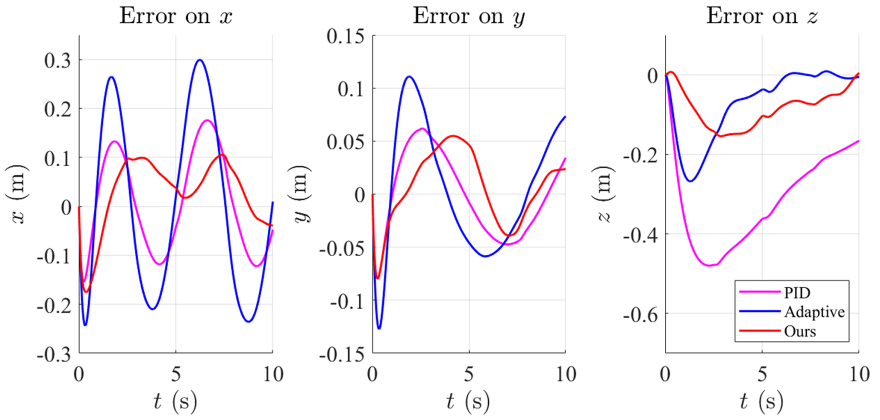

Fig. 18 Figure-8 trajectory tracking error in each channel.

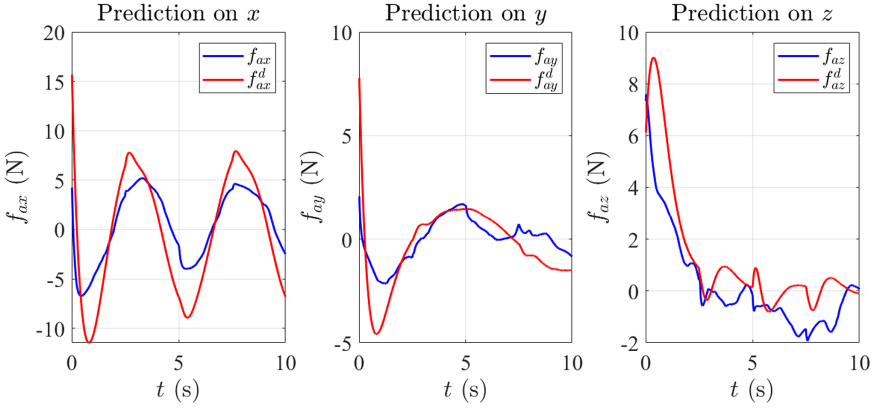

Fig. 19 Figure-8 disturbance prediction in each channel.

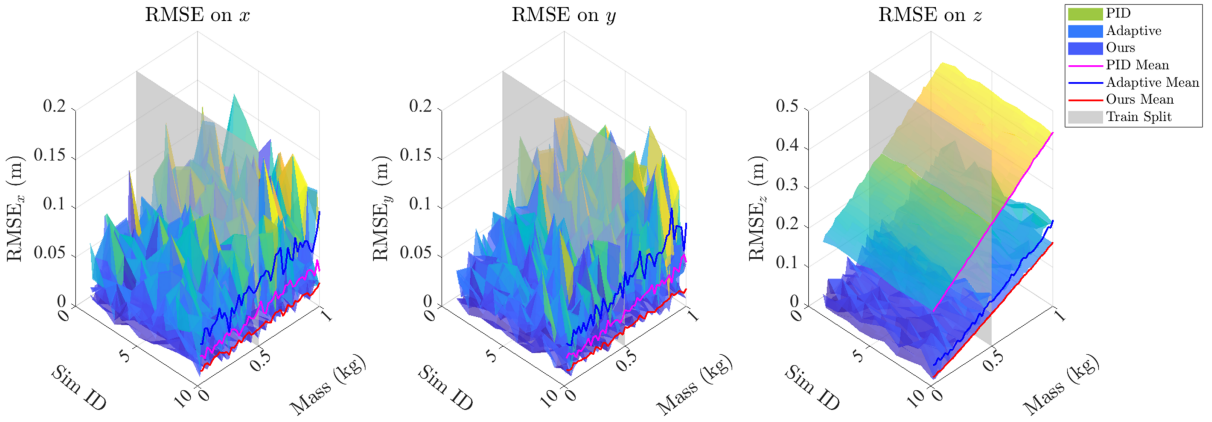

Fig. 20 MinSnap trajectory tracking error surface under variant masses.

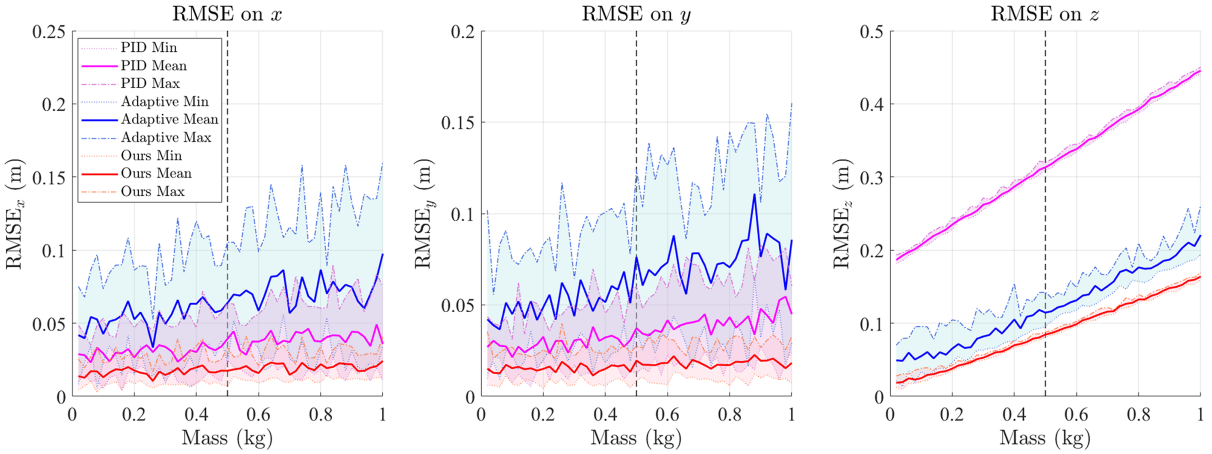

Fig. 21 MinSnap trajectory tracking error curve under variant masses.

designing the structure of the compensation. The effectiveness of the proposed disturbance compensation scheme has been verified. We integrate the derived compensation model into the meta-adaptive control framework, ensuring a proven Lyapunov stability guarantee. Comparative studies with PID and traditional adaptive controllers demonstrate lower tracking errors. These results validate that the proposed method effectively reduces tracking errors and rapidly stabilizes the system, regardless of the specific task trajectories and payload settings.

Acknowledgments

This work is supported by the National Natural Science Foundation of China (No. 62006152, No. 61773262, No. 12302060), the China Aviation Science Foundation (No. 2022Z071057002), the States Key Laboratory of Air Traffic Management System (No. SKLATM202301), and the Shanghai Rising-Star Program (Sailing Project, 23YF1420200).

Appendix A: Manipulator with an Unknown Payload

To clarify the transformation from Y to Y', we present the specific expressions of the variables involved in the intermediate derivation process. The expressions for the matrices M, C, and G in Eq. (7a) are as follows:

$$M = \begin{bmatrix} m_1 l_{c1}^2 + I_1 + I_e + m_e(l_1^2 + 2\cos(\delta_e - q_2) l_1 l_{ce} + l_{ce}^2) & I_e + m_e(l_{ce}^2 + l_1 \cos(\delta_e - q_2) l_{ce}) \\ I_e + m_e(l_{ce}^2 + l_1 \cos(\delta_e - q_2) l_{ce}) & I_e + m_e l_{ce}^2 \end{bmatrix} \tag{A1}$$

$$C = \begin{bmatrix} l_1 l_{ce} m_e \dot{q}_2 \sin(\delta_e - q_2) & l_1 l_{ce} m_e \dot{q}_1 \sin(\delta_e - q_2) + l_1 l_{ce} m_e \dot{q}_2 \sin(\delta_e - q_2) \\ -l_1 l_{ce} m_e \dot{q}_1 \sin(\delta_e - q_2) & 0 \end{bmatrix} \tag{A2}$$

$$G = \begin{bmatrix} g m_e (l_{ce} \cos(q_1 - \delta_e + q_2) + l_1 \cos(q_1)) + g l_{c1} m_1 \cos(q_1) \\ g l_{ce} m_e \cos(q_1 - \delta_e + q_2) \end{bmatrix} \tag{A3}$$

Then, with the parameter vector Θ as given in Eq. (7b), the expanded forms of Y and b in Eq. (7a) can be derived as follows.

$$
\begin{aligned}
y_{11} &= y_{13} = y_{21} = y_{23} = \ddot{q}_1 + \ddot{q}_2 \\
y_{12} &= l_1^2\ddot{q}_1 + gl_1\cos(q_1) \\
y_{22} &= 0 \\
y_{14} &= g\sin(q_1+q_2) + l_1(2\ddot{q}_1 + \ddot{q}_2)\sin(q_2) + 2l_1\dot{q}_1\dot{q}_2\cos(q_2) \\
&\quad + l_1\dot{q}_2^2\cos(q_2) \\
y_{15} &= g\cos(q_1+q_2) + l_1(2\ddot{q}_1 + \ddot{q}_2)\cos(q_2) - 2l_1\dot{q}_1\dot{q}_2\sin(q_2) \\
&\quad - l_1\dot{q}_2^2\sin(q_2) \\
y_{24} &= g\sin(q_1+q_2) + l_1\ddot{q}_1\sin(q_2) - l_1\dot{q}_1^2\cos(q_2) \\
y_{25} &= g\cos(q_1+q_2) + l_1\ddot{q}_1\cos(q_2) + l_1\dot{q}_1^2\sin(q_2)
\end{aligned}
\tag{A4}
$$

$$
b(q,\ddot{q})_{2\times 1} = \begin{bmatrix} (I_1 + m_1 l_{c1}^2)\ddot{q}_1 + m_1 g l_{c1}\cos(q_1) \\ 0 \end{bmatrix} \tag{A5}
$$

where y_{ij} is the element in the ith row and jth column of matrix $Y = (y_{ij}) \in \mathbb{R}^{2\times 5}$.

Finally, mark the states $\ddot{q}^*$ and $\dot{q}^*$ in the left side of Eq. (7a) as $M(q)\ddot{q}^* + C(q,\dot{q})\dot{q}^* + G(q)$, and replace them with $\ddot{q}_r$ and $\dot{q}_r$, respectively. Now, upon linear transformation of $M(q)\ddot{q}_r + C(q,\dot{q})\dot{q}_r + G(q)$, the new regressor matrix Y' will be derived.

$$
\begin{aligned}
y'_{11} &= y'_{13} = y'_{21} = y'_{23} = \ddot{q}_{r1} + \ddot{q}_{r2} \\
y'_{12} &= l_1^2\ddot{q}_{r1} + gl_1\cos(q_1) \\
y'_{22} &= 0 \\
y'_{14} &= g\sin(q_1+q_2) + l_1(2\ddot{q}_{r1} + \ddot{q}_{r2})\sin(q_2) \\
&\quad + l_1(\dot{q}_1\dot{q}_{r2} + \dot{q}_{r1}\dot{q}_2)\cos(q_2) + l_1\dot{q}_2\dot{q}_{r2}\cos(q_2) \\
y'_{15} &= g\cos(q_1+q_2) + l_1(2\ddot{q}_{r1} + \ddot{q}_{r2})\cos(q_2) \\
&\quad - l_1(\dot{q}_1\dot{q}_{r2} + \dot{q}_{r1}\dot{q}_2)\sin(q_2) - l_1\dot{q}_2\dot{q}_{r2}\sin(q_2) \\
y'_{24} &= g\sin(q_1+q_2) + l_1\ddot{q}_{r1}\sin(q_2) - l_1\dot{q}_1\dot{q}_{r1}\cos(q_2) \\
y'_{25} &= g\cos(q_1+q_2) + l_1\ddot{q}_{r1}\cos(q_2) + l_1\dot{q}_1\dot{q}_{r1}\sin(q_2)
\end{aligned}
\tag{A6}
$$

where y'_{ij} represents the element in the ith row and jth column of matrix $Y' = (y'_{ij}) \in \mathbb{R}^{2\times 5}$. The aforementioned steps are similar to the aerial manipulator.

Appendix B: Transformation Among Rotational Velocities

For aircraft, the rotational velocity measured in the body-fixed frame is defined as $\omega_b^b = [p, q, r]^T$, with p, q, r denoting the roll rate, pitch rate, and yaw rate, respectively. The Euler angle rates are noted as $\dot{\Phi}_b = [\dot{\phi}, \dot{\theta}, \dot{\psi}]^T$. Usually, the Euler angle rates $\dot{\Phi}_b$ can be converted into the rotational velocity ω_b^b using the following transformation [38]:

$$
\begin{bmatrix} p \\ q \\ r \end{bmatrix} = \underbrace{\begin{bmatrix} 1 & 0 & -\sin\theta \\ 0 & \cos\phi & \cos\theta\sin\phi \\ 0 & -\sin\phi & \cos\theta\cos\phi \end{bmatrix}}_{Q} \begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} \tag{B1}
$$

where ϕ, θ, and ψ represent the Euler angles of the aircraft. Note that the absolute rotational velocity ω_b can be transformed from the rotational velocity ω_b^b via the rotation matrix R_b. In other words, ω_b can be expressed as $\omega_b = R_b\omega_b^b$. Combing the above transformation $\omega_b^b = Q\dot{\Phi}_b$, we can derive the relationship of $\omega_b = R_bQ\dot{\Phi}_b$ and $T_b = R_bQ$ equivalently.

Appendix C: Aerial Manipulator with an Unknown Payload

Unknown Basis for $S(p_{bl_2}^b)$ ***and*** $S(p_{bl_2}^b)^T S(p_{bl_2}^b)$**:** The specific expression for $p_{bl_2}^b$ can be obtained by substituting Eq. (18) into Eq. (20). So does the expanded form of $S(p_{bl_2}^b)$. Then, we can find that the element a_i conforms to the following structure:

$$
\begin{aligned}
a_1 &= \lambda_{1\,s}\cdot[l_{ce}\sin(\delta_e)] + \lambda_{1c}\cdot[l_{ce}\cos(\delta_e)] + b_{a_1} \\
a_2 &= 0 \\
a_3 &= \lambda_{3\,s}\cdot[l_{ce}\sin(\delta_e)] + \lambda_{3c}\cdot[l_{ce}\cos(\delta_e)] + b_{a_3}
\end{aligned}
\tag{C1}
$$

Note that we focus on the unknown basis and abbreviate the multipliers as $\lambda_{1\,s}, \lambda_{1c}, \lambda_{3\,s}, \lambda_{3c}$ and the biases as b_{a_1}, b_{a_3}. By substituting Eq. (C1) into Eq. (23) and examining all unknown basis in the quadratic terms of a_i, we can determine the unknown basis for $S(p_{bl_2}^b)^T S(p_{bl_2}^b)$ as indicated in Eq. (24).

Unknown Basis for $J_P^{(l_2)}$ ***and*** $S(p_{bl_2}^b)J_P^{(l_2)}$**:** The raw expression for $J_P^{(l_2)}$ undergoes the following transformations: First, we substitute $p_{bl_2}^b$ using Eq. (20) and $p_{b_2}^2 = P_2 \cdot B_1$. Next, we convert the cross-product operation into matrix multiplication. Finally, by applying the distributive property of matrix multiplication, we arrive at the final result.

$$
\begin{aligned}
J_P^{(l_2)} &= \begin{bmatrix} z_0^b \times p_{bl_2}^b(\delta_e, l_{ce}) & z_1^b \times \left(R_1^0 \cdot R_2^1(\delta_e)\cdot p_{b_2}^2(l_{ce})\right) \end{bmatrix} \\
&= \begin{bmatrix} z_0^b \times (P_2\cdot B_1 + b_1) & z_1^b \times (P_2 \cdot B_1) \end{bmatrix} \\
&= \begin{bmatrix} S(z_0^b)\cdot(P_2\cdot B_1 + b_1) & S(z_1^b)\cdot(P_2\cdot B_1) \end{bmatrix} \\
&= \begin{bmatrix} S(z_0^b)P_2\cdot B_1 + S(z_0^b)b_1 & S(z_1^b)P_2\cdot B_1 \end{bmatrix} \\
&= \begin{bmatrix} P_3\cdot B_1 + b_2 & P_4\cdot B_1 \end{bmatrix}
\end{aligned}
\tag{C2}
$$

By substituting specific variables into Eq. (25), we can deduce that the expression for $J_P^{(l_2)}$ takes the form of Eq. (27), and each element c_{ij} therein exhibits the following structure:

$$
\begin{aligned}
c_{11} &= \lambda_{11\,s}\cdot[l_{ce}\sin(\delta_e)] + \lambda_{11c}\cdot[l_{ce}\cos(\delta_e)] + b_{c_{11}} \\
c_{12} &= \lambda_{12\,s}\cdot[l_{ce}\sin(\delta_e)] + \lambda_{12c}\cdot[l_{ce}\cos(\delta_e)] + b_{c_{12}} \\
c_{31} &= \lambda_{31\,s}\cdot[l_{ce}\sin(\delta_e)] + \lambda_{31c}\cdot[l_{ce}\cos(\delta_e)] + b_{c_{31}} \\
c_{32} &= \lambda_{32\,s}\cdot[l_{ce}\sin(\delta_e)] + \lambda_{32c}\cdot[l_{ce}\cos(\delta_e)] + b_{c_{32}}
\end{aligned}
\tag{C3}
$$

Therefore, according to the specific structures provided by Eq. (C1) and Eq. (C3), we can confirm that the unknown basis for $a_3c_{11} - a_1c_{31}$ and $a_3c_{12} - a_1c_{32}$ is $V_1 \cup V_3$.

Unknown Basis for $R_2^1H_{l_2}R_1^2$**:** The specific expression of the product term $R_2^1H_{l_2}R_1^2$ is as follows:

$$
R_2^1H_{l_2}R_1^2 = \begin{bmatrix} \cos^2(n_2+\delta_e)I_{xe}^{l_{ce}} + \sin^2(n_2+\delta_e)I_{ze}^{l_{ce}} & 0 & \sin(n_2+\delta_e)\cos(n_2+\delta_e)(I_{ze}^{l_{ce}} - I_{xe}^{l_{ce}}) \\ 0 & I_{ye}^{l_{ce}} & 0 \\ \sin(n_2+\delta_e)\cos(n_2+\delta_e)(I_{ze}^{l_{ce}} - I_{xe}^{l_{ce}}) & 0 & \sin^2(n_2+\delta_e)I_{xe}^{l_{ce}} + \cos^2(n_2+\delta_e)I_{ze}^{l_{ce}} \end{bmatrix} \tag{C4}
$$

References

[1] Du, R., and Cowlagi, R. V., "Interactive Sensing and Planning for a Quadrotor Vehicle in Partially Known Environments," *Journal of Guidance, Control, and Dynamics*, Vol. 42, No. 7, 2019, pp. 1601–1611. https://doi.org/10.2514/1.G003773

[2] Medhi, J. K., and McGhan, C. L., "Towards a Modular Architecture for Intelligent Aerial Manipulator Systems," *AIAA Information Systems-AIAA Infotech@ Aerospace*, AIAA Paper 2018-1631, 2018. https://doi.org/10.2514/6.2018-1631

[3] Cataldi, E., Muscio, G., Trujillo, M. A., Rodríguez, Y., Pierri, F., Antonelli, G., Caccavale, F., Viguria, A., Chiaverini, S., and Ollero, A., "Impedance Control of an Aerial-Manipulator: Preliminary Results," *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Inst. of Electrical and Electronics Engineers, New York, 2016, pp. 3848–3853. https://doi.org/10.1109/IROS.2016.7759566

[4] Meng, X., He, Y., Gu, F., Li, Q., and Han, J., "Dynamics Modeling and Simulation Analysis for Rotorcraft Aerial Manipulator System," *36th Chinese Control Conference (CCC)*, Inst. of Electrical and Electronics Engineers, New York, 2017, pp. 1156–1161. https://doi.org/10.23919/ChiCC.2017.8027504

[5] Bellicoso, C. D., Buonocore, L. R., Lippiello, V., and Siciliano, B., "Design, Modeling and Control of a 5-Dof Light-Weight Robot Arm for Aerial Manipulation," *23rd Mediterranean Conference on Control and Automation (MED)*, Inst. of Electrical and Electronics Engineers, New York, 2015, pp. 853–858. https://doi.org/10.1109/MED.2015.7158852

[6] Jones, R. M., Sun, D., Barsi Haberfeld, G., Lakshmanan, A., Marinho, T., and Hovakimyan, N., "Design and Control of a Small Aerial Manipulator for Indoor Environments," *AIAA Information Systems-AIAA Infotech@ Aerospace*, AIAA Paper 2017-1374, 2017. https://doi.org/10.2514/6.2017-1374

[7] Sumathy, V., Warier, R., and Ghose, D., "Design, Reachability Analysis, and Constrained Motion Planning for a Quadcopter Manipulator System," *AIAA Scitech 2022 Forum*, AIAA Paper 2022-0269, 2022. https://doi.org/10.2514/6.2022-0269

[8] Suarez, A., Jimenez-Cano, A. E., Vega, V. M., Heredia, G., Rodriguez-Castaño, A., and Ollero, A., "Design of a Lightweight Dual Arm System for Aerial Manipulation," *Mechatronics*, Vol. 50, April 2018, pp. 30–44. https://doi.org/10.1016/j.mechatronics.2018.01.005

[9] Lippiello, V., and Ruggiero, F., "Cartesian Impedance Control of a UAV with a Robotic Arm," *IFAC Proceedings*, Vol. 45, No. 22, 2012, pp. 704–709. https://doi.org/10.3182/20120905-3-HR-2030.00158

[10] Antonelli, G., and Cataldi, E., "Adaptive Control of Arm-Equipped Quadrotors. Theory and Simulations," *22nd Mediterranean Conference on Control and Automation*, Inst. of Electrical and Electronics Engineers, New York, 2014, pp. 1446–1451. https://doi.org/10.1109/MED.2014.6961579

[11] Samadikhoshkho, Z., Ghorbani, S., and Janabi-Sharifi, F., "Coupled Dynamic Modeling and Control of Aerial Continuum Manipulation Systems," *Applied Sciences*, Vol. 11, No. 19, 2021, p. 9108. https://doi.org/10.3390/app11199108

[12] Meng, X., He, Y., Gu, F., Yang, L., Dai, B., Liu, Z., and Han, J., "Design and Implementation of Rotor Aerial Manipulator System," *2016 IEEE International Conference on Robotics and Biomimetics (ROBIO)*, Inst. of Electrical and Electronics Engineers, New York, 2016, pp. 673–678. https://doi.org/10.1109/ROBIO.2016.7866400

[13] Zhang, Y., Fan, W., Xiang, C., Xu, B., Ai, T., Yuan, L., and Liu, Y., "An Innovative Aerial Manipulator with Tandem Ducted Fans: Modeling, Control, and Simulation," *Complexity*, Vol. 2020, Sept. 2020, pp. 1–24. https://doi.org/10.1155/2020/7923539

[14] Mellinger, D., Lindsey, Q., Shomin, M., and Kumar, V., "Design, Modeling, Estimation and Control for Aerial Grasping and Manipulation," *2011 IEEE/RSJ International Conference on Intelligent Robots and Systems*, Inst. of Electrical and Electronics Engineers, New York, 2011, pp. 2668–2673. https://doi.org/10.1109/IROS.2011.6094871

[15] Baraban, G., Sheckells, M., Kim, S., and Kobilarov, M., "Adaptive Parameter Estimation for Aerial Manipulation," *2020 American Control Conference (ACC)*, Inst. of Electrical and Electronics Engineers, New York, 2020, pp. 614–619. https://doi.org/10.23919/ACC45564.2020

[16] Lee, H., Kim, S., and Kim, H. J., "Control of an Aerial Manipulator Using On-Line Parameter Estimator for an Unknown Payload," *2015 IEEE International Conference on Automation Science and Engineering (CASE)*, Inst. of Electrical and Electronics Engineers, New York, 2015, pp. 316–321. https://doi.org/10.1109/CoASE.2015.7294098

[17] Yildiz, Y., Unel, M., and Demirel, A. E., "Nonlinear Hierarchical Control of a Quad Tilt-Wing Uav: An Adaptive Control Approach," *International Journal of Adaptive Control and Signal Processing*, Vol. 31, No. 9, 2017, pp. 1245–1264. https://doi.org/10.1002/acs.2759

[18] Yilmaz, E., Zaki, H., and Unel, M., "Nonlinear Adaptive Control of an Aerial Manipulation System," *18th European Control Conference (ECC)*, Inst. of Electrical and Electronics Engineers, New York, 2019, pp. 3916–3921. https://doi.org/10.23919/ECC.2019.8795709

[19] Jafarnejadsani, H., Sun, D., Lee, H., and Hovakimyan, N., "Optimized $\mathcal{L}_1$ Adaptive Controller for Trajectory Tracking of an Indoor Quadrotor," *Journal of Guidance, Control, and Dynamics*, Vol. 40, No. 6, 2017, pp. 1415–1427. https://doi.org/10.2514/1.G000566

[20] Furgiuele, T. C., and Parks, D. J., "Time-Delay Margin Tuning of a Quadrotor Adaptive Controller," *Journal of Guidance, Control, and Dynamics*, Vol. 46, No. 2, 2023, pp. 362–373. https://doi.org/10.2514/1.G007066

[21] Wu, Z., Cheng, S., Ackerman, K. A., Gahlawat, A., Lakshmanan, A., Zhao, P., and Hovakimyan, N., "L_1 Adaptive Augmentation for Geometric Tracking Control of Quadrotors," *2022 International Conference on Robotics and Automation (ICRA)*, Inst. of Electrical and Electronics Engineers, New York, 2022, pp. 1329–1336. https://doi.org/10.1109/ICRA46639.2022.9811946

[22] Wu, Z., Cheng, S., Zhao, P., Gahlawat, A., Ackerman, K. A., Lakshmanan, A., Yang, C., Yu, J., and Hovakimyan, N., "$\mathcal{L}_1$Quad: $\mathcal{L}_1$ Adaptive Augmentation of Geometric Control for Agile Quadrotors with Performance Guarantees," arXiv preprint arXiv: 2302.07208, 2023. https://doi.org/10.48550/arXiv.2302.07208

[23] Sun, D., Wan, N., Dai, W., Zhang, Y., and Hovakimyan, N., "Control Design for an Aerial Manipulator for Pick-and-Place Tasks," *AIAA Scitech 2019 Forum*, AIAA Paper 2019-1291, 2019. https://doi.org/10.2514/6.2019-1291

[24] Sánchez-Sánchez, C., and Izzo, D., "Real-Time Optimal Control via Deep Neural Networks: Study on Landing Problems," *Journal of Guidance, Control, and Dynamics*, Vol. 41, No. 5, 2018, pp. 1122–1135. https://doi.org/10.2514/1.G002357

[25] Bisheban, M., and Lee, T., "Geometric Adaptive Control with Neural Networks for a Quadrotor in Wind Fields," *IEEE Transactions on Control Systems Technology*, Vol. 29, No. 4, 2020, pp. 1533–1548. https://doi.org/10.1109/TCST.2020.3006184

[26] Cho, S., Shim, D. H., and Kim, J., "Gaussian Process-Based Visual Servoing Framework for an Aerial Parallel Manipulator," *AIAA Information Systems-AIAA Infotech@ Aerospace*, AIAA Paper 2017-1067, 2017. https://doi.org/10.2514/6.2017-1067

[27] Finn, C., Abbeel, P., and Levine, S., "Model-Agnostic Meta-Learning for Fast Adaptation of Deep Networks," *34th International Conference on Machine Learning*, PMLR, New York, 2017, pp. 1126–1135. https://doi.org/10.48550/arXiv.1703.03400

[28] Craig, W., Yeo, D., and Paley, D. A., "Geometric Attitude and Position Control of a Quadrotor in Wind," *Journal of Guidance, Control, and Dynamics*, Vol. 43, No. 5, 2020, pp. 870–883. https://doi.org/10.2514/1.G004710

[29] O'Connell, M., Shi, G., Shi, X., Azizzadenesheli, K., Anandkumar, A., Yue, Y., and Chung, S.-J., "Neural-Fly Enables Rapid Learning for Agile Flight in Strong Winds," *Science Robotics*, Vol. 7, No. 66, 2022, Paper eabm6597. https://doi.org/10.1126/scirobotics.abm6597

[30] Mania, H., Tu, S., and Recht, B., "Certainty Equivalence Is Efficient for Linear Quadratic Control," *Advances in Neural Information Processing Systems*, Vol. 32, Curran Assoc., New York, 2019, p. 2. https://doi.org/10.48550/arXiv.1902.07826

[31] Slotine, J.-J. E., and Li, W., "On the Adaptive Control of Robot Manipulators," *International Journal of Robotics Research*, Vol. 6, No. 3, 1987, pp. 49–59. https://doi.org/10.1177/027836498700600303

[32] Khosla, P. K., and Kanade, T., "Parameter Identification of Robot Dynamics," *1985 24th IEEE Conference on Decision and Control*, Inst. of Electrical and Electronics Engineers, New York, 1985, pp. 1754–1760. https://doi.org/10.1109/CDC.1985.268838

[33] Spong, M. W., Hutchinson, S., and Vidyasagar, M., *Robot Modeling and Control*, Wiley, New York, 2020, Chap. 3.2. https://doi.org/10.1109/MCS.2021.3122271

[34] Khalil, H., *Nonlinear Systems*, Macmillan Publishing Company, London, 1992, Chap. 3.4.

[35] Mellinger, D., and Kumar, V., "Minimum Snap Trajectory Generation and Control for Quadrotors," *2011 IEEE International Conference on Robotics and Automation*, Inst. of Electrical and Electronics Engineers, New York, 2011, pp. 2520–2525.
https://doi.org/10.1109/ICRA.2011.5980409

[36] Kim, S., Choi, S., and Kim, H. J., "Aerial Manipulation Using a Quadrotor with a Two Dof Robotic Arm," *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems*, Inst. of Electrical and Electronics Engineers, New York, 2013, pp. 4990–4995.
https://doi.org/10.1109/IROS.2013.6697077

[37] Xu, Y., Shmaliy, Y. S., Shen, T., Chen, D., Sun, M., and Zhuang, Y., "INS/UWB-Based Quadrotor Localization Under Colored Measurement Noise," *IEEE Sensors Journal*, Vol. 21, No. 5, 2020, pp. 6384–6392.
https://doi.org/10.1109/JSEN.2020.3038242

[38] Ducard, G. J., *Fault-Tolerant Flight Control and Guidance Systems: Practical Methods for Small Unmanned Aerial Vehicles*, Springer Science & Business Media, Heidelberg, 2009, Chap. 3.4.

D.Paley
Associate Editor