Optical flow-based control for micro air vehicles
an efficient data-driven incremental nonlinear dynamic inversion approach

Ho, Hann Woei; Zhou, Ye; Feng, Yiting; de Croon, Guido C.H.E.

DOI
10.1007/s10514-024-10174-4

Publication date
2024

Document Version
Final published version

Published in
Autonomous Robots

Autonomous Robots (2024) 48:22
https://doi.org/10.1007/s10514-024-10174-4

Optical flow-based control for micro air vehicles: an efficient data-driven incremental nonlinear dynamic inversion approach

Hann Woei Ho[1] · Ye Zhou[1] · Yiting Feng[1] · Guido C. H. E. de Croon[2]

Received: 14 January 2024 / Accepted: 31 August 2024

Abstract

This paper proposes an innovative approach for optical flow-based control of micro air vehicles (MAVs), addressing challenges inherent in the nonlinearity of optical flow observables. The proposed incremental nonlinear dynamic inversion (INDI) control scheme employs an efficient data-driven approach to directly estimate the inverse of the time-varying INDI control effectiveness in real-time. This method eliminates the constant effectiveness assumption typically made by traditional INDI methods and reduces the computational burden associated with inverting this variable at each time step. It effectively handles rapidly changing system dynamics, often encountered in optical flow-based control, particularly height-dependent control variables. Stability analysis of the proposed control scheme is conducted, and its robustness and efficiency are demonstrated through both numerical simulations and real-world flight tests. These tests include multiple landings of an MAV on a static, flat surface with several different tracking setpoints, as well as hovering and landings on moving and undulating surfaces. Despite the challenges posed by noisy optical flow estimates and lateral or vertical movements of the landing surfaces, the MAV successfully tracks or lands on the surface with an exponential decay of both height and vertical velocity almost simultaneously, aligning with the desired performance.

Keywords Optical flow · Incremental nonlinear dynamic inversion · Data-driven approach · Micro air vehicles

1 Introduction

Recent advancements in small flying robot development have expanded their potential for diverse indoor applications, leveraging their adeptness at navigating confined spaces and minimizing the risk of harm when operating in close proximity to users (Soria, 2022; Fridovich-Keil et al., 2020; Curtis et al., 2023). However, achieving full autonomy for these robots remains a challenge, primarily due to their constrained payload capacities and computational resources.

Extensive research efforts have been devoted to designing solutions inspired by biological systems, particularly tiny flying insects that face similar limitations yet are capable of executing intricate tasks. Biologists believe that these insects heavily rely on visual information, such as optical flow cues, for navigation and control (Dong et al., 2023; Mehdi Yadipour et al., 2023; Collett, 2002).

In light of the computational efficiency of optical flow, its use in micro air vehicles (MAVs) has garnered increasing interest. Optical flow, which captures the apparent motion of features in a visual scene, offers rich environmental information that is crucial for tasks like hovering, landing, and obstacle avoidance (Yu et al., 2022; Mahlknecht et al., 2022; Falanga et al., 2020). Its low computational demands and adaptability to GPS-denied environments make optical flow methods well-suited for MAV navigation and control.

However, employing optical flow methods introduces significant control challenges. The highly nonlinear nature of optical flow observables, which are represented as ratios

✉ Ye Zhou
zhouye@usm.my

Hann Woei Ho
aehannwoei@usm.my

Yiting Feng
fengyt@student.usm.my

Guido C. H. E. de Croon
G.C.H.E.deCroon@tudelft.nl

[1] School of Aerospace Engineering, Engineering Campus, Universiti Sains Malaysia, 14300 Nibong Tebal, Pulau Pinang, Malaysia

[2] Faculty of Aerospace Engineering, Delft University of Technology, Kluyverweg 1, 2629HS Delft, Zuid Holland, The Netherlands

Published online: 06 October 2024

of velocities to MAV height, complicates the controller design. Moreover, the estimation of optical flow observables is susceptible to noise and error sources, such as sensor noise, motion blur, and lighting changes, further intensifying control complexities. Most approaches resort to proportional-integral-derivative (PID) controllers to regulate the optical flow to a desired constant (Ruffier & Franceschini, 2015; Herissé et al., 2011). However, these methods are typically tailored for systems with linear dynamics. The stability and performance of employing such linear controllers in a nonlinear system can raise concerns, as they may only function effectively within specific operating points, such as within a certain range of heights. Nonlinear control strategies, such as PID scheduling (Kendoul, 2014; Cesetti et al., 2010), also offer a means to address optical flow-based control nonlinearities. Nevertheless, they demand intricate parameter tuning to achieve optimal performance. Furthermore, employing PID scheduling in optical flow-based control for MAVs can be problematic, as it relies heavily on accurate knowledge of system dynamics and parameters, which are often unknown or uncertain in real-world applications. Additionally, the performance of PID scheduling can be sensitive to parameter variations and model errors. This sensitivity may lead to degraded performance and system instability when using optical flow observables as the control input (Ho et al., 2018).

Alternatively, adaptive approaches, such as bio-inspired strategies and machine learning (Guido CHE et al., 2022; O'Connell et al., 2022; Zhou, 2023), present potential solutions by leveraging real-time feedback to adapt strategies and enhance control performance, especially in dealing with nonlinearities and uncertainties. However, the application of these adaptive methods encounters notable challenges, particularly when considered for on-board processing within MAVs. The complexity of MAV systems, coupled with their limited computational capacity, poses substantial obstacles to their successful implementation. In addition, these methods often necessitate pre-training to generate stable control strategies, which is often conducted in simulation, introducing additional issues, such as sim-to-real mismatch. On the other hand, various nonlinear control methods have been serving as valuable candidates for addressing nonlinear flight control challenges. Among them, nonlinear dynamic inversion (NDI) is a model-based controller that establishes a direct relationship between output and input. NDI employs linear methods to design a virtual input and then solves for the physical input through nonlinear inversion. This method does not impose a heavy computational load or require pre-training of the controller. However, it relies on the assumption of complete accuracy in the model, a condition seldom met in real-world applications due to model simplification and external disturbances.

In light of this, incremental nonlinear dynamic inversion (INDI) was introduced to enhance adaptability and robustness while overcoming the limitations associated with traditional methods and the assumed accuracy of the model, effectively mitigating the impact of model inaccuracies through real-time sensor measurements in flight control (Smith, 1998; Bacon et al., 2001; Sieberling et al., 2010). The INDI method has demonstrated its applicability to nonlinear systems with unknown models and nonlinear outputs (Zhou et al., 2021). Functioning as a nonlinear control method, INDI utilizes the inverse of system dynamics to cancel the nonlinearities in the system. This utilization of the inverse enables INDI to generate a linear control input for stabilizing the system. Moreover, INDI exhibits robustness against parameter uncertainties and model errors, making it an effective tool for addressing nonlinearities in optical flow-based control for MAVs. Additionally, INDI stands out as a computationally efficient method that can be implemented in real-time, without the need for pre-training in either simulation or real-world scenarios. This characteristic renders it a practical and versatile choice for various applications.

The current implementation of INDI relies on a constant or derived control effectiveness matrix with fixed coefficients, determined by the system dynamics (Steffensen et al., 2022; Smeur et al., 2018). However, in the context of optical flow stabilization, landing, or tracking in MAVs, the system dynamics are intricately tied to the height of the vehicle above the ground. As the height of the vehicle changes, the dynamics of the system undergo corresponding changes. Consequently, the constant nature of the G matrix becomes inappropriate in this context, necessitating a dynamic approach. To overcome this challenge, it would be advantageous to develop an INDI method that incorporates a time-varying G matrix. This adaptation would empower the control system to respond dynamically to varying system dynamics, thereby enhancing overall performance and stability.

The online identification of the time-varying G matrix, followed by its inversion at every time step, emerges as a crucial aspect of INDI-based control for calculating the control increments. However, the computational intensity of this process, which involves calculating the G matrix and its inverse, poses challenges, especially in real-time applications.

To address this, we propose a novel INDI control scheme that directly identifies its inverse, denoted as $G^{\dagger} = G^{-1}$. This innovation aims to reduce the computational burden associated with control increment calculations, resulting in a significant reduction in computation time. Furthermore, this approach eliminates errors introduced by the inversion calculation, ultimately improving the accuracy of the control system.

The main contribution of this paper lies in the advancement of optical flow-based control for MAVs through the

development of an innovative data-driven INDI approach. Unlike traditional INDI implementations that rely on the constant control effectiveness assumption, our proposed approach introduces a time-varying matrix, addressing the output nonlinearities inherent in optical flow-based control. This adaptation enhances the system's responsiveness to changing conditions, eliminating the need to estimate or measure distances through additional sensors. Furthermore, we introduce an efficient data-driven approach within the INDI framework, directly identifying its inverse matrix $G^{\dagger} = G^{-1}$, reducing computational demands and enhancing control system accuracy. Through a stability analysis and a comprehensive evaluation via numerical simulations and real-world flight tests, including scenarios with noisy optical flow estimates and dynamic landing surfaces, we validate the robustness and efficacy of the proposed method. The proposed INDI method, featuring the direct identification of the $G^{\dagger}$ matrix, presents a computationally efficient approach with the potential to enhance the performance and efficiency of INDI-based control in MAVs. This paper is an expanded and revised version of the conference paper (Ho & Zhou, 2023), incorporating new sections covering comprehensive method derivation, stability, and robustness analysis, as well as presenting additional real-world flight tests and performance comparisons.

2 Problem formulation

This section describes the dynamics of an MAV during the landing process onto a moving platform and its observed optical flow signals.

2.1 Position dynamics of an MAV

The body reference frame of the MAV ($O^b X^b Y^b Z^b$) and the inertial reference frame ($O^I X^I Y^I Z^I$) are defined and depicted as in Fig. 1. A camera, aligned with the body reference frame, is attached to the MAV, looking downward. The position $\boldsymbol{d}$ and velocity $\boldsymbol{v}$ of the MAV are governed by Newton's second law:

$$\dot{\boldsymbol{d}} = \boldsymbol{v}, \tag{1}$$

$$\dot{\boldsymbol{v}} = \boldsymbol{g} + \frac{1}{M}\boldsymbol{T}(T, \boldsymbol{\omega}) + \frac{1}{M}\boldsymbol{\delta}(\boldsymbol{v}, \boldsymbol{\epsilon}), \tag{2}$$

where $\boldsymbol{g}$ is gravity, M is the mass, $\boldsymbol{T}$ is the thrust generated by the rotors, dependent on total thrust in Z_b direction T and attitude in Euler angles $\boldsymbol{\omega} = [\theta, \phi, \psi]$. The term $\boldsymbol{\delta}$ represents the aerodynamic forces due to velocity $\boldsymbol{v}$ and other factors $\boldsymbol{\epsilon}$, such as wind, acting on the MAV.

In the tracking moving platform and landing tasks, the changes in yaw angle ψ can be assumed small, and the initial MAV position can be placed so that X^b remains within the inertial $O^I Y^I Z^I$ plane, allowing us to neglect ψ in the MAV's position dynamics. The position dynamics simplified from Eq. (2) becomes

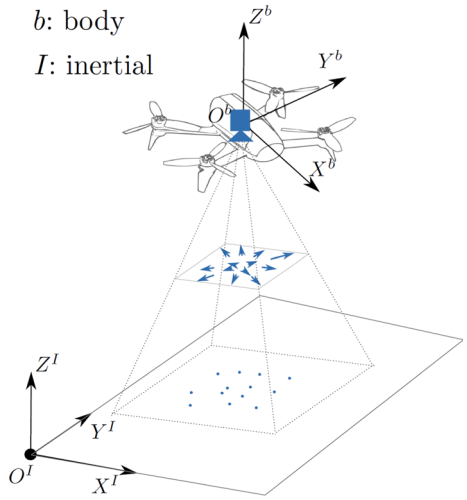

Fig. 1 MAV body ($O^b X^b Y^b Z^b$) and inertial ($O^I X^I Y^I Z^I$) reference frames. O^b is fixed at the center of gravity of the MAV, X^b points forward, Y^b is port, and Z^b points upward. The inertial reference frame is located on the ground and follows the North-East-Up ($X^I - Y^I - Z^I$) system

$$\begin{aligned}\dot{\boldsymbol{v}} &= \boldsymbol{g} + \frac{1}{M}\boldsymbol{T}(\theta, \phi, T) + \frac{1}{M}\boldsymbol{\delta}(\boldsymbol{v}, \boldsymbol{\epsilon}) \\ &= \begin{bmatrix} 0 \\ 0 \\ -g \end{bmatrix} + \frac{1}{M}\begin{bmatrix} \cos\phi \sin\theta \\ -\sin\phi \\ \cos\phi\cos\theta \end{bmatrix} T + \frac{1}{M}\boldsymbol{\delta}(\boldsymbol{v}, \boldsymbol{\epsilon}).\end{aligned} \tag{3}$$

The attitude of the MAV is stabilized through an inner control loop, which is designed to be accurate and responsive in its operation. And thus, the input of the position control loop is $\boldsymbol{u} = [\theta, \phi, T]^T$.

2.2 The observed optical flow signal

The downward-facing camera attached to the MAV perceives the optical flow used for the landing on a moving platform. To elucidate control principles, this paper focuses on tracking and landing on a flat platform with slow translational degrees of freedom. Its position and velocity vectors are $\boldsymbol{d}^r$ and $\boldsymbol{v}^r$, respectively, where the superscript r signifies the reference platform. During movements, the camera captures the optical flow, representing the velocity of features on the platform beneath it. The optical flow signals are proportional to the ratio of relative velocity to the height above the platform $d_z - d_z^r$. In this configuration, the camera is able to capture the three-dimensional flow field of the features beneath it. And thus, the output vector of this problem is denoted as

$\boldsymbol{y} = [\vartheta_x, \vartheta_y, \vartheta_z]^T$, providing valuable information for applications, such as autonomous landing and tracking in MAVs.

The optical flow in the X- and Y-axes, denoting the lateral flow of features, describes the velocity of the features in the image plane. These measurements can be formulated as the ratio of the relative lateral velocity to the height above the platform:

$$\vartheta_x = c_x \frac{v_x - v_x^r}{d_z - d_z^r}, \tag{4}$$

$$\vartheta_y = c_y \frac{v_y - v_y^r}{d_z - d_z^r}, \tag{5}$$

where c_x and c_y are unknown proportionality constants. These constants, intrinsic to the optical flow measurement system, are determined by various factors, such as the resolution, image acquisition rate, and camera focal length. The above equations indicate that the position of the MAV along the X- and Y-axes is not directly observable from lateral flows. Nevertheless, these flows can be utilized to stabilize the position by maintaining zero lateral flows or controlling the velocity with an appropriate constant (Ho et al., 2018; Zhou et al., 2021).

As the MAV approaches the surface, the captured optical flow exhibits a divergent pattern corresponding to the flow divergence ϑ_z. This divergence represents the velocity of the features in the image plane along the line of sight and can be formulated as

$$\vartheta_z = c_z \frac{v_z - v_z^r}{d_z - d_z^r}, \tag{6}$$

where c_z is an unknown positive constant. When the MAV is in the process of approaching the platform, $v_z - v_z^r$ is negative, $d_z - d_z^r$ is consistently positive, resulting in a negative value for ϑ_z. The constant flow divergence approach (Zhou et al., 2021) has been employed to control the vertical dynamics of MAVs by maintaining the flow divergence as a negative constant for landing, i.e., $\vartheta_z = \vartheta_z^*$. Consequently, the time response of the distance $d_z - d_z^r$ experiences an exponential decrease, ultimately reaching zero.

3 Data-driven incremental nonlinear dynamic inversion

As depicted in the previous section, the system dynamics described by Eq. (3) are input non-affine, involve unknown model parameters, and lack direct measurability of the full states. In response to these challenges, this paper proposes a data-driven incremental nonlinear dynamic inversion (INDI) method to effectively control the MAV for tracking and landing on a moving platform with optical flows.

3.1 Reformulation of optical flow tracking control problems

The INDI method can be derived from the system represented by Eqs. (1, 3–6) or in a more general form for nonlinear systems:

$$\dot{\boldsymbol{x}} = \boldsymbol{f}(\boldsymbol{x}, \boldsymbol{u}), \tag{7}$$

$$\boldsymbol{y} = \boldsymbol{h}(\boldsymbol{x}), \tag{8}$$

where $\boldsymbol{x} \in \mathcal{R}^n$ is the system state, $\boldsymbol{u} \in \mathcal{R}^m$ is the system input, $\boldsymbol{y} \in \mathcal{R}^p$ is the system output or the observations, the functions $\boldsymbol{f} : \mathcal{R}^{n+m} \to \mathcal{R}^n$ and $\boldsymbol{h} : \mathcal{R}^n \to \mathcal{R}^p$ are smooth vector fields. If $p < m$ or $p > m$, the control problem becomes over-determined or under-determined, respectively. Control allocation techniques or weighted least square methods can be employed to address these problems. For both theoretical derivations and practical applications in this paper, it is assumed that the numbers of inputs and outputs are equal, i.e., $p = m$.

In the optical flow tracking problems under consideration, the observation encompasses both the system and platform dynamics, as indicated in Eqs. (4–6). Consequently, the state space must be augmented to incorporate the states of both the system and the reference platform, forming $[\boldsymbol{d}^T, \boldsymbol{v}^T, \boldsymbol{d}^{rT}, \boldsymbol{v}^{rT}]^T$. On the other hand, biological systems often leverage relative measurements for navigation and control. For example, many animals rely on relative motion cues in their visual system to track moving targets and navigate their environment. In alignment with these biological systems, the formulation of this control task focuses on the relative position and velocity $[(\boldsymbol{d}-\boldsymbol{d}^r)^T, (\boldsymbol{v}-\boldsymbol{v}^r)^T]^T$, thereby reducing the dimensionality of the problem. Furthermore, the relative position in the X- and Y-axes is not observable, as previously mentioned. The state of interest can be further reduced and defined as $\boldsymbol{x} = [\tilde{d}_z, \tilde{v}_x, \tilde{v}_y, \tilde{v}_z]^T = [d_z - d_z^r, v_x - v_x^r, v_y - v_y^r, v_z - v_z^r]^T$. Consequently, the dynamics of the relative position can be reformulated as follows:

$$\dot{\tilde{d}}_z = \tilde{v}_z, \tag{9}$$

$$\dot{\tilde{v}}_x = \frac{1}{M}\cos\phi\sin\theta T + \frac{1}{M}\delta_x - \dot{v}_x^r, \tag{10}$$

$$\dot{\tilde{v}}_y = -\frac{1}{M}\sin\phi T + \frac{1}{M}\delta_y - \dot{v}_y^r, \tag{11}$$

$$\dot{\tilde{v}}_z = \frac{1}{M}\cos\phi\cos\theta T - g + \frac{1}{M}\delta_z - \dot{v}_z^r. \tag{12}$$

The height above the platform is assumed to be $\tilde{d}_z > c_d$, where c_d is a positive threshold below which the system shuts down.

3.2 Incremental nonlinear dynamic inversion

We will first introduce the concept of nonlinear dynamic inversion (INDI), upon which our data-driven INDI approach is founded. To enhance clarity and understanding, we will begin by considering a nonlinear input-affine system, described as follows:

$$\dot{\boldsymbol{x}} = \boldsymbol{f}(\boldsymbol{x}) + \boldsymbol{g}(\boldsymbol{x})\boldsymbol{u}, \tag{13}$$

$$\boldsymbol{y} = \boldsymbol{h}(\boldsymbol{x}), \tag{14}$$

where $\boldsymbol{f} : \mathcal{R}^n \rightarrow \mathcal{R}^n$ and $\boldsymbol{h} : \mathcal{R}^n \rightarrow \mathcal{R}^p$ are smooth vector fields, and $\boldsymbol{g} : \mathcal{R}^n \rightarrow \mathcal{R}^{n\times m}$, whose columns are smooth vector fields. And the initial step involves input–output linearization, achieved by differentiating the output until the input-affine form appears (Isidori, 2013; Sieberling et al., 2010; Wang et al., 2019). The first-order derivative of the lth output is

$$\begin{aligned}\dot{y}_l = \frac{dh_l(\boldsymbol{x})}{dt} &= \frac{\partial h_l(\boldsymbol{x})}{\partial \boldsymbol{x}}\frac{d\boldsymbol{x}}{dt} \\ &= \nabla h_l(\boldsymbol{x})[\boldsymbol{f}(\boldsymbol{x}) + \boldsymbol{g}(\boldsymbol{x})\boldsymbol{u}] \\ &= L_f h_l(\boldsymbol{x}) + L_g h_l(\boldsymbol{x})\boldsymbol{u},\end{aligned} \tag{15}$$

where ∇ is Jacobian operator, and $L_f h_l(\boldsymbol{x}) = \nabla h_l(\boldsymbol{x})\boldsymbol{f}(\boldsymbol{x})$ denotes the first-order Lie derivative along the vector field $\boldsymbol{f}(\boldsymbol{x})$ (Slotine et al., 1991). If the term $L_g h_l(\boldsymbol{x})$ is non-zero, the first-order differentiation of the output $\dot{y}_l$, as in Eq. (15), is input-affine. However, if $L_g h_l(\boldsymbol{x})$ is zero, indicating a non-affine input, further differentiations are performed until the Lie derivative with respect to $\boldsymbol{g}(\boldsymbol{x})$ becomes nonzero:

$$y_l^{(r_l)} = L_f^{r_l} h_l(\boldsymbol{x}) + L_g L_f^{r_l-1} h_l(\boldsymbol{x})\boldsymbol{u}, \tag{16}$$

where r_l is the relative degree for lth output, $r_l \geq 1$. The total relative degree $r = r_1 + r_2 + \cdots + r_p$ is no more than the number of the states n. Consequently, the total relative degree of the system exists within the range of $r \in [p, n]$.

In the case where $r = n$, each output and its first $r_l - 1$ derivatives contribute to a transformation from the original x coordinate system to the new ξ coordinate system:

$$\boldsymbol{\xi} = \boldsymbol{\Gamma}_1(\boldsymbol{x}). \tag{17}$$

As the new system states $\boldsymbol{\xi}$ are smooth and invertible functions of $\boldsymbol{x}$, this transformation qualifies as a diffeomorphism within the defined domain. And the new system of $\boldsymbol{\xi}$ can be written in the controllability canonical form, also known as the companion form:

$$\begin{bmatrix} \dot{\xi}_1 \\ \dot{\xi}_2 \\ \vdots \\ \dot{\xi}_{r_1} \\ \dot{\xi}_{r_1+1} \\ \vdots \\ \dot{\xi}_r \end{bmatrix} = \begin{bmatrix} \xi_2 \\ \xi_3 \\ \vdots \\ L_f^{r_1} h_1(\boldsymbol{x}) \\ \xi_{r_1+2} \\ \vdots \\ L_f^{r_p} h_p(\boldsymbol{x}) \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ \vdots \\ L_g L_f^{r_1-1} h_1(\boldsymbol{x}) \\ 0 \\ \vdots \\ L_g L_f^{r_p-1} h_p(\boldsymbol{x}) \end{bmatrix} \boldsymbol{u}. \tag{18}$$

The system dynamics in the controllability canonical form can be more concisely represented as

$$\boldsymbol{y}^{(\boldsymbol{r})} = L_f^{\boldsymbol{r}} \boldsymbol{h}(\boldsymbol{x}) + L_g L_f^{\boldsymbol{r}-1} \boldsymbol{h}(\boldsymbol{x})\boldsymbol{u}. \tag{19}$$

And the output $\boldsymbol{y}$ is linear to the new state $\boldsymbol{\xi}$ as shown in Eq. (18). Utilizing this controllability canonical form above, the cancellation of nonlinearities and the imposition of desired linear dynamics can be easily implemented (Slotine et al., 1991).

Note that when $r < n$, it indicates the presence of $n - r$ internal dynamics, which are not directly observable from the input–output linearization. In such cases, it is always feasible to identify $n - r$ additional functions $\boldsymbol{\Gamma}_2(\boldsymbol{x})$, where $\boldsymbol{\Gamma}_2 : \mathcal{R}^n \rightarrow \mathcal{R}^{n-r}$, so that the construction of the transformation $\boldsymbol{\Gamma} = [\boldsymbol{\Gamma}_1; \boldsymbol{\Gamma}_2]$ ensures a diffeomorphism within the defined domain (Isidori, 2013). The selection of these additional functions in the defined domain is guided by the condition

$$L_{g_j} \Gamma_i(\boldsymbol{x}) = 0 \; \forall \; i \in [r+1, n], \;\; j \in [1, m]. \tag{20}$$

Subsequently, the transformation to the new $\boldsymbol{\zeta}$ coordinate system can be defined as

$$\boldsymbol{\zeta} = \begin{bmatrix} \boldsymbol{\xi} \\ \boldsymbol{\eta} \end{bmatrix} = \boldsymbol{\Gamma}(\boldsymbol{x}) = \begin{bmatrix} \boldsymbol{\Gamma}_1(\boldsymbol{x}) \\ \boldsymbol{\Gamma}_2(\boldsymbol{x}) \end{bmatrix}. \tag{21}$$

For systems written in the controllability canonical form as in Eq. (19), a virtual control input can be introduced to cancel the nonlinearities, yielding a linear input–output relation in the outer loop:

$$\boldsymbol{y}^{(\boldsymbol{r})} = \boldsymbol{v}, \tag{22}$$

which is separated from the nonlinear inner loop

$$\boldsymbol{v} = L_f^{\boldsymbol{r}} \boldsymbol{h}(\boldsymbol{x}) + L_g L_f^{\boldsymbol{r}-1} \boldsymbol{h}(\boldsymbol{x})\boldsymbol{u}. \tag{23}$$

Solving for the physical control input $\boldsymbol{u}$ involves inverting Eq. (23), as follows:

$$\boldsymbol{u} = L_g L_f^{\boldsymbol{r}-1} \boldsymbol{h}(\boldsymbol{x})^{-1} [\boldsymbol{v} - L_f^{\boldsymbol{r}} \boldsymbol{h}(\boldsymbol{x})]. \tag{24}$$

The solution for the virtual input $\boldsymbol{v}$ can be determined using linear feedback control techniques, such as polynomials, to place all the roots strictly in the left-half complex plane, ensuring exponentially stable dynamics.

Remark 1 Note that NDI is not confined to input-affine systems. In instances involving non-affine systems, besides the standard state transformation, such as in Eq. (21), it is necessary to determine an input transformation $\boldsymbol{\mu} = \Upsilon(\boldsymbol{x}, \boldsymbol{u})$ to transform the nonlinear system dynamics into an equivalent controllability canonical form. It is important to emphasize that this process demands a precise dynamic model, and the formulation of the input transformation is a case-by-case design. The NDI approach leverages mathematical modeling and nonlinear cancellation techniques to cope with nonlinear dynamics. However, its applicability relies on the availability of precise and time-invariant models, conditions that are often impractical in real-world systems.

To address inherent uncertainties and model mismatches, incremental NDI (INDI) was proposed as a more robust and adaptive alternative. INDI utilizes an online identified incremental model, calculating control input increments based on desired virtual input increments, thereby eliminating the strict reliance on an accurate system model. Furthermore, the INDI method is derived from the general nonlinear system in Eq. (7), approximating system dynamics around the condition at time t_0 using a first-order Taylor series expansion:

$$\dot{\boldsymbol{x}}(t) \approx \dot{\boldsymbol{x}}_0 + F(\boldsymbol{x}_0, \boldsymbol{u}_0)\Delta\boldsymbol{x}(t) + G(\boldsymbol{x}_0, \boldsymbol{u}_0)\Delta\boldsymbol{u}(t), \tag{25}$$

where $\Delta\boldsymbol{x}(t) = \boldsymbol{x}(t) - \boldsymbol{x}_0$, $\Delta\boldsymbol{u}(t) = \boldsymbol{u}(t) - \boldsymbol{u}_0$, $F(\boldsymbol{x}_0, \boldsymbol{u}_0) = \frac{\partial \boldsymbol{f}(\boldsymbol{x},\boldsymbol{u})}{\partial \boldsymbol{x}}|_{\boldsymbol{x}_0,\boldsymbol{u}_0} \in \mathcal{R}^{n\times n}$ is the system matrix of the linearized model, and $G(\boldsymbol{x}_0, \boldsymbol{u}_0) = \frac{\partial \boldsymbol{f}(\boldsymbol{x},\boldsymbol{u})}{\partial \boldsymbol{u}}|_{\boldsymbol{x}_0,\boldsymbol{u}_0} \in \mathcal{R}^{n\times m}$ is the control effectiveness matrix. This equation can be simplified with a time-scale separation assumption when the control input changes significantly faster than the system state. And assuming a high sampling frequency and instantaneous control effects (Ronald et al., 2018), Eq. (25) becomes:

$$\dot{\boldsymbol{x}}(t) \approx \dot{\boldsymbol{x}}_0 + G(\boldsymbol{x}_0, \boldsymbol{u}_0)\Delta\boldsymbol{u}(t). \tag{26}$$

With the above formulation, the linear outer loop is established by replacing the differentiated state with the virtual input $\dot{\boldsymbol{x}} = \boldsymbol{v}$. This system dynamics, featuring a full-state feedback problem, $n = p = m$, and a relative degree of 1 for each output, $r_l = 1$, is common in INDI applications. In the inner loop, the control increment can be determined by inverting Eq. (26) as

$$\Delta\boldsymbol{u}(t) = G(\boldsymbol{x}_0, \boldsymbol{u}_0)^{-1}(\boldsymbol{v} - \dot{\boldsymbol{x}}_0). \tag{27}$$

Since the incremental model is linear and usually much simpler compared to the global nonlinear model, the parameters of the matrix G, if unknown, can be identified online from the measurement or estimation of the differentiated state $\dot{\boldsymbol{x}}$ using least square (LS) techniques. Similarly, the INDI method is applicable to systems, where $\boldsymbol{y} \neq \boldsymbol{x}$ when their relative degrees $\boldsymbol{r}$ are known. In other words, the model structure of the input–output linearization, as in Eq. (19), is known, but the exact mathematical expressions for $L_f^{\boldsymbol{r}}\boldsymbol{h}(\boldsymbol{x})$ and $L_g L_f^{\boldsymbol{r}-1}\boldsymbol{h}(\boldsymbol{x})$ can be unknown.

Moreover, through the incremental model, the application of this approach extends to more general system formulations as in Eqs. (7) and (8). By differentiating each output until the input explicitly appears, the direct input–output relation of the model is derived. In the broader context of Eq. (19), where the input–output relation can be nonlinear and non-affine in the input, represented as

$$\boldsymbol{y}^{(\boldsymbol{r})} = \boldsymbol{\alpha}(\boldsymbol{x}, \boldsymbol{u}). \tag{28}$$

Utilizing the first-order Taylor series expansion and the time-scale separation principle, the input–output relation around any time instance t_0 can be linearly approximated as follows:

$$\boldsymbol{y}^{(\boldsymbol{r})}(t) \approx \boldsymbol{y}_0^{(\boldsymbol{r})} + G(\boldsymbol{x}_0, \boldsymbol{u}_0)\Delta\boldsymbol{u}(t), \tag{29}$$

where $G(\boldsymbol{x}_0, \boldsymbol{u}_0) = \frac{\partial \boldsymbol{\alpha}(\boldsymbol{x},\boldsymbol{u})}{\partial \boldsymbol{u}}|_{\boldsymbol{x}_0,\boldsymbol{u}_0} \in \mathcal{R}^{p\times m}$. Note that, for a non-affine input–output relation as shown in Eq. (28), an explicit and exact input transformation is unnecessary when using the local linearized model. Similarly, by replacing the differentiated output with the virtual input $\boldsymbol{y}^{(\boldsymbol{r})} = \boldsymbol{v}$, the control increment can be determined by inverting Eq. (29):

$$\Delta\boldsymbol{u}(t) = G(\boldsymbol{x}_0, \boldsymbol{u}_0)^{-1}[\boldsymbol{v} - \boldsymbol{y}_0^{(\boldsymbol{r})}]. \tag{30}$$

It is important to note that if $r < n$, the internal dynamics must also be stable or bounded to ensure effective controller design.

3.3 Incremental nonlinear dynamic inversion for optical flow tracking control

Given that the observation $\boldsymbol{y}$ exhibits nonlinearity in the state $\boldsymbol{x}$, the initial task involves establishing the input–output relationship by iteratively differentiating the output until the input emerges explicitly. The resulting expressions for the derivatives are provided below:

$$\dot{\vartheta}_x = c_x \left[\frac{\dot{\tilde{v}}_x}{\tilde{d}_z} - \frac{\tilde{v}_x \tilde{v}_z}{\tilde{d}_z^2} \right] = c_x \left[\frac{(\cos\phi \sin\theta T + \delta_x - M\dot{v}_x^r)}{M\tilde{d}_z} - \vartheta_x \vartheta_z \right], \tag{31}$$

$$\dot{\vartheta}_y = c_y \left[\frac{\dot{\tilde{v}}_y}{\tilde{d}_z} - \frac{\tilde{v}_y \tilde{v}_z}{\tilde{d}_z^2} \right] = c_y \left[\frac{(-\sin\phi T + \delta_y - M\dot{v}_y^r)}{M\tilde{d}_z} - \vartheta_y \vartheta_z \right], \tag{32}$$

$$\dot{\vartheta}_z = c_z \left[\frac{\dot{\tilde{v}}_z}{\tilde{d}_z} - \frac{\tilde{v}_z^2}{\tilde{d}_z^2} \right] = c_z \left[\frac{(\cos\phi \cos\theta T - Mg + \delta_z - M\dot{v}_z^r)}{M\tilde{d}_z} - \vartheta_z^2 \right]. \tag{33}$$

These derived equations can be seen as functions of new state $\boldsymbol{\zeta}$ instead of the original state $\boldsymbol{x}$, which is not directly observable. These three equations indicate that we have $\boldsymbol{\xi} = \boldsymbol{y}$ with a relative degree $r = 3$, and there is one internal dynamic. Complying with the condition in Eq. (20), we can choose $\eta = \tilde{d}_z$ to establish a diffeomorphism $\boldsymbol{\zeta} = \boldsymbol{\Gamma}(\boldsymbol{x})$ as in Eq. (21). Consequently, the input–output relationship is concisely represented by

$$\dot{\boldsymbol{y}} = \boldsymbol{\alpha}(\boldsymbol{\zeta}, \boldsymbol{u}, \boldsymbol{\delta}, \dot{\boldsymbol{v}}^r), \tag{34}$$

which is nonlinear and non-affine in the input. Taking the first-order Taylor series expansion results in

$$\begin{aligned} \dot{\boldsymbol{y}} \approx\ & \dot{\boldsymbol{y}}_0 + \frac{\partial \boldsymbol{\alpha}(\boldsymbol{\zeta}, \boldsymbol{u}, \boldsymbol{\delta}, \dot{\boldsymbol{v}}^r)}{\partial \boldsymbol{\zeta}} |_{\boldsymbol{\zeta}_0, \boldsymbol{u}_0, \boldsymbol{\delta}_0, \dot{\boldsymbol{v}}_0^r} (\boldsymbol{\zeta} - \boldsymbol{\zeta}_0) \\ & + \frac{\partial \boldsymbol{\alpha}(\boldsymbol{\zeta}, \boldsymbol{u}, \boldsymbol{\delta}, \dot{\boldsymbol{v}}^r)}{\partial \boldsymbol{u}} |_{\boldsymbol{\zeta}_0, \boldsymbol{u}_0, \boldsymbol{\delta}_0, \dot{\boldsymbol{v}}_0^r} (\boldsymbol{u} - \boldsymbol{u}_0) \\ & + \frac{\partial \boldsymbol{\alpha}(\boldsymbol{\zeta}, \boldsymbol{u}, \boldsymbol{\delta}, \dot{\boldsymbol{v}}^r)}{\partial \boldsymbol{\delta}} |_{\boldsymbol{\zeta}_0, \boldsymbol{u}_0, \boldsymbol{\delta}_0, \dot{\boldsymbol{v}}_0^r} (\boldsymbol{\delta} - \boldsymbol{\delta}_0) \\ & + \frac{\partial \boldsymbol{\alpha}(\boldsymbol{\zeta}, \boldsymbol{u}, \boldsymbol{\delta}, \dot{\boldsymbol{v}}^r)}{\partial \dot{\boldsymbol{v}}^r} |_{\boldsymbol{\zeta}_0, \boldsymbol{u}_0, \boldsymbol{\delta}_0, \dot{\boldsymbol{v}}_0^r} (\dot{\boldsymbol{v}}^r - \dot{\boldsymbol{v}}_0^r), \end{aligned} \tag{35}$$

where the second or higher-order terms have been neglected in this equation.

Further simplification is achieved by employing the time-scale separation assumption. It is important to note that, due to the output nonlinearity of optical flow signals, this assumption is applicable only when $1/\tilde{d}_z$ remains bounded, as proved in our previous work (Zhou et al., 2021). The practical application of this assumption involves setting a threshold for the height $\tilde{d}_z$, as mentioned in Sect. 3.1. The convergent property of this internal dynamics $\eta = \tilde{d}_z$ can be guaranteed in hovering and landing tasks by maintaining the flow divergence as zero or a negative constant. For the tasks with unbounded $1/\tilde{d}_z$ or other control objectives directly influencing height-related dynamics, one can employ an additional state estimator (Ho et al., 2017) or an extended INDI (EINDI) method (Zhou et al., 2021).

Additionally, it is assumed that aerodynamic forces $\boldsymbol{\delta}$ and platform dynamics $\dot{\boldsymbol{v}}^r$ change at a slower pace than the system dynamics, allowing for a more efficient focus on identifying the direct input–output relationship and generating the control input. Under these assumptions, Eq. (35) is further simplified to

$$\dot{\boldsymbol{y}} \approx \dot{\boldsymbol{y}}_0 + G(\boldsymbol{\zeta}_0, \boldsymbol{u}_0)(\boldsymbol{u} - \boldsymbol{u}_0), \tag{36}$$

where $\boldsymbol{u} = [\theta, \phi, T]^T$ and

$$G(\boldsymbol{\zeta}, \boldsymbol{u}) = \frac{1}{M\tilde{d}_z} \cdot \begin{bmatrix} c_x \cos\phi \cos\theta T & -c_x \sin\phi \sin\theta T & c_x \cos\phi \sin\theta \\ 0 & -c_y \cos\phi T & -c_y \sin\phi \\ -c_z \cos\phi \sin\theta T & -c_z \sin\phi \cos\theta T & c_z \cos\phi \cos\theta \end{bmatrix}. \tag{37}$$

To address the nonlinear control problem described above, we introduce a virtual control input to replace the differentiated output $\dot{\boldsymbol{y}} = \boldsymbol{v}$, and the control increment can be determined by inverting Eq. (36):

$$\boldsymbol{u} - \boldsymbol{u}_0 = G(\boldsymbol{\zeta}_0, \boldsymbol{u}_0)^{-1}(\boldsymbol{v} - \dot{\boldsymbol{y}}_0). \tag{38}$$

In traditional MAV control scenarios, the matrix G is often derived, and its unknown or unobservable parameters are typically assumed to be constants. Although unknown, the identification of this G matrix is straightforward under the constant assumption. However, in optical flow-based control problems, as indicated in Eq. (37), the matrix G not only depends on unknown parameters c_x, c_y, and c_z, but it also exhibits nonlinear variation with the distance above the platform $\tilde{d}_z$, which is not directly measurable using optical flow sensors. Consequently, measuring all relevant variables and calculating matrix G from Eq. (37) at each time step is difficult. Alternatively, another approach to determine the control increment is to identify G online using the recursive least square (RLS) approach from input and output data with the time sequence, which can be rewritten as

$$\dot{\boldsymbol{y}}_k - \dot{\boldsymbol{y}}_{k-1} \approx G_{k-1} \cdot (\boldsymbol{u}_k - \boldsymbol{u}_{k-1}). \tag{39}$$

In this case, the explicit representation of the time-varying matrix G in Eq. (37) is unnecessary, but it needs to be identified and inverted at every time step.

To recursively identify the G matrix, the input information and output target of the model identification in Eq. (39) can be written as $\Delta \boldsymbol{u}_k = \boldsymbol{u}_k - \boldsymbol{u}_{k-1}$ and $\Delta \dot{\boldsymbol{y}}_k = \dot{\boldsymbol{y}}_k - \dot{\boldsymbol{y}}_{k-1}$,

respectively. The elements of the G matrix are typically identified row by row using the RLS method. Given that all rows share the same covariance matrix, they can also be identified simultaneously. The RLS approach utilized in this paper is as follows (Haykin, 2002; Zhou et al., 2021):

$$\boldsymbol{\tau}_k = \Delta \dot{\boldsymbol{y}}_k - \widehat{\Delta \dot{\boldsymbol{y}}}_k, \tag{40}$$

$$\hat{G}_k^T = \hat{G}_{k-1}^T + \frac{\mathfrak{C}_{k-1}\Delta \boldsymbol{u}_k}{\gamma + \Delta \boldsymbol{u}_k^T \mathfrak{C}_{k-1}\Delta \boldsymbol{u}_k}\boldsymbol{\tau}_k, \tag{41}$$

$$\mathfrak{C}_k = \frac{1}{\gamma}\left(\mathfrak{C}_{k-1} - \frac{\mathfrak{C}_{k-1}\Delta \boldsymbol{u}_k \Delta \boldsymbol{u}_k^T \mathfrak{C}_{k-1}}{\gamma + \Delta \boldsymbol{u}_k^T \mathfrak{C}_{k-1}\Delta \boldsymbol{u}_k}\right), \tag{42}$$

where $\boldsymbol{\tau}_k \in \mathcal{R}^p$ is the prediction error, also known as the innovation, $\mathfrak{C}_k \in \mathcal{R}^{m\times m}$ is the estimation covariance matrix, and $\gamma \in [0, 1]$ is the forgetting factor in the RLS approach for the incremental model identification. For faster-changing time-variant model parameters, a smaller value of the forgetting factor is typically chosen. This allows the algorithm to give more weight to recent data, thereby quickly adapting to changes in the system dynamics. Conversely, a larger forgetting factor is suitable for more stable systems, as it smooths out short-term fluctuations and relies more on historical data for parameter estimation.

Remark 2 While the inputs of the outer loop position control are Euler angles, and total thrust, $\boldsymbol{u} = [\theta, \phi, T]^T$, which do not strictly form an input-affine system, it is feasible to employ a nonlinear solver to determine the generated thrust vector $\boldsymbol{\mu} = [T_x, T_y, T_z]^T$ with Eq. (3) without involving the state $\boldsymbol{x} = [\tilde{d}_z, \tilde{v}_x, \tilde{v}_y, \tilde{v}_z]^T$ in the input transformation. Consequently, the system dynamics can be viewed as affine in the thrust vector $\boldsymbol{\mu}$, providing a more direct framework for control design and analysis. This implies that, in this specific problem, the input transformation $\boldsymbol{\mu} = \Upsilon(\boldsymbol{u})$ can be performed independently without affecting the state dynamics.

The optical flow dynamics described in Eqs. (9–12) can be reformulated to be an input-affine system with input transformation, converting the Euler angles and total thrust $\boldsymbol{u} = [\theta, \phi, T]^T$ into the external thrust forces in the X-, Y-, and Z-axes, $\boldsymbol{\mu} = [T_x, T_y, T_z]^T$, as shown in Eq. (3). With a targeted thrust force vector, the nonlinear inversion, $\boldsymbol{u} = \Upsilon^{-1}(\boldsymbol{\mu})$, can be inserted into Eq. (38) to enable the determination of thrust and Euler angle commands (Smeur et al., 2018). This is a practical approach to finding attitude and thrust commands for a desired acceleration without linearizing the input function. However, this approach requires additional calculations and sacrifices some of the generality inherent to the INDI approach. Therefore, in the implementation, we will directly utilize the linearized input in terms of attitude and thrust.

3.4 Inverted G matrix identification

To alleviate the computational burden associated with the data-driven approach, this paper proposes a direct identification of the inverted G matrix, denoted as $G^\dagger = G^{-1}$, from input and output data with a time sequence $t_k, t_{k-1}, t_{k-2}, \cdots$ as follows:

$$\boldsymbol{u}_k - \boldsymbol{u}_{k-1} \approx G_{k-1}^\dagger \cdot (\dot{\boldsymbol{y}}_k - \dot{\boldsymbol{y}}_{k-1}). \tag{43}$$

The parameters in matrix $G^\dagger \in \mathcal{R}^{m\times p}$ can be identified using the RLS approach. The control command can then be directly calculated with the identified matrix $\widehat{G}^\dagger$:

$$\boldsymbol{u} = \boldsymbol{u}_k + \widehat{G}_{k-1}^\dagger \cdot (\boldsymbol{v}_k - \dot{\boldsymbol{y}}_k). \tag{44}$$

The virtual input $\boldsymbol{v}_k$ at the current time t_k is designed as a feedback linear control with constant proportional, integral, and derivative feedback gains:

$$\boldsymbol{v}_k = -K_p(\boldsymbol{y}_k - \boldsymbol{\vartheta}^*) - K_i \int_0^t (\boldsymbol{y}_k - \boldsymbol{\vartheta}^*)dt - K_d\ \dot{\boldsymbol{y}}_k, \tag{45}$$

where $\boldsymbol{\vartheta}^*$ is a constant vector of the desired optical flow. In the tracking platform task, we will have $\boldsymbol{\vartheta}^* = \boldsymbol{0}$, and in the landing on a moving platform task, we will have $\boldsymbol{\vartheta}^* = [0\,0\,\vartheta_z^*]^T$. The desired flow divergence ϑ_z^* for landing is a negative constant, and by following the constant flow divergence, the MAV will have a soft landing on the platform with the time response of $\tilde{d}_z$ (Zhou et al., 2021):

$$\tilde{d}_z(t) = \tilde{d}_{z_0} e^{\frac{\vartheta_z^*}{c_z}t}, \tag{46}$$

where $\tilde{d}_{z_0}$ is the initial height above the landing surface.

In data-based INDI methods, where the matrix G is unknown and time-varying, the control command can be calculated either by first identifying G and then calculating its inversion in the traditional way, or by directly identifying the inverse matrix $G^\dagger$ using our proposed algorithm. Both the RLS identification step and the matrix inversion step require $\mathcal{O}(p^3)$ computational effort. The approach with direct identification of $G^\dagger$ has a lower computation load compared to the traditional method because the inversion step is no longer necessary. This approach results in a more efficient INDI control implementation.

The proposed method of directly identifying the inverse matrix $G^\dagger$ also has advantages in terms of accuracy and stability, especially for optical flow-based control. The matrix G increases to very large values and changes very quickly when the vehicle is approaching the platform, as it is proportional to $1/\tilde{d}_z$. This can result in difficulties for the control system to accurately estimate the matrix G. On the other hand, the inverse matrix $G^\dagger$ tends to converge as the vehicle

approaches the platform, making the updating process more straightforward and precise. Furthermore, by eliminating the inversion step, the proposed method reduces the chances of numerical errors and improves the stability of the control system, especially when dealing with the possibility of ill-conditioned matrices. In conclusion, the direct identification of the inverse matrix $G^{\dagger}$ offers both computational efficiency and improved accuracy and stability of the data-driven INDI approach.

3.5 Stability and robustness analysis

This subsection undertakes an analysis of the stability and robustness of the proposed INDI approach with online identification of the incremental model. Specifically, it focuses on the scenario where the inverse matrix $G^{\dagger}$ introduces errors in estimating the inverted G matrix when employing LS techniques. To broaden the analysis and enhance its applicability, we consider a more general condition involving model uncertainties, which provides valuable insights into the robustness of the INDI approach, evaluating its stability under circumstances characterized by both estimation errors and uncertainties in the underlying model.

For a clearer illustration, we will examine a more common case involving the following affine nonlinear system:

$$\begin{cases} \dot{\boldsymbol{x}} = \boldsymbol{f}(\boldsymbol{x}) + \boldsymbol{g}(\boldsymbol{x})\boldsymbol{u} + \boldsymbol{\sigma} \\ \boldsymbol{y} = \boldsymbol{h}(\boldsymbol{x}) \end{cases} \tag{47}$$

where $\boldsymbol{x}$ is the state vector, $\boldsymbol{u}$ is the input vector, $\boldsymbol{y}$ is the output vector, and $\boldsymbol{\sigma}$ is the uncertain external disturbance. Applying a virtual control input $\boldsymbol{v}$, the traditional NDI control law can be designed as:

$$\boldsymbol{u} = \boldsymbol{g}^{-1}(\boldsymbol{x})[\boldsymbol{v} - \boldsymbol{f}(\boldsymbol{x})]. \tag{48}$$

In addition, model uncertainties $\Delta \boldsymbol{f}(\boldsymbol{x})$ and $\Delta \boldsymbol{g}(\boldsymbol{x})$ will be considered, and the system dynamics can be written as

$$\dot{\boldsymbol{x}} = \boldsymbol{f}(\boldsymbol{x}) + \Delta \boldsymbol{f}(\boldsymbol{x}) + [\boldsymbol{g}(\boldsymbol{x}) + \Delta \boldsymbol{g}(\boldsymbol{x})]\boldsymbol{u} + \boldsymbol{\sigma}. \tag{49}$$

Substituting Eq. (48) into Eq. (49) yields

$$\begin{aligned} \dot{\boldsymbol{x}} = {} & \Delta \boldsymbol{f}(\boldsymbol{x}) + \Delta \boldsymbol{g}(\boldsymbol{x})\boldsymbol{g}^{-1}(\boldsymbol{x})\boldsymbol{f}(\boldsymbol{x}) \\ & + [\boldsymbol{I}_{n\times n} + \Delta \boldsymbol{g}(\boldsymbol{x})\boldsymbol{g}^{-1}(\boldsymbol{x})]\boldsymbol{v} + \boldsymbol{\sigma}. \end{aligned} \tag{50}$$

The desired dynamic equation $\dot{\boldsymbol{x}} = \boldsymbol{v}$ can be guaranteed only when $\Delta \boldsymbol{f}(\boldsymbol{x}) = \Delta \boldsymbol{g}(\boldsymbol{x}) = \boldsymbol{0}$ and $\boldsymbol{\sigma} = \boldsymbol{0}$. Therefore, the traditional NDI relies on an accurate system model, and any deviations due to external disturbances, parameter perturbations, or model uncertainties can degrade the robustness of the real system.

3.5.1 Stability analysis

A detailed stability analysis of a closed-loop system employing INDI control has been conducted, considering the perturbations arising from omitted terms of $\Delta \boldsymbol{x}$ and the higher-order terms of $\Delta \boldsymbol{x}$ and $\Delta \boldsymbol{u}$, employing Lyapunov methods and the nonlinear system perturbation theory (Wang et al., 2019). In the context of this paper, we will omit these terms and focus on a direct analysis of the stability of the INDI control considering the influence of external disturbances, incorporating an outer-loop feedback control law with fractional power terms. INDI control utilizes the first-order Taylor series expansion of the state equation around the condition at $(\boldsymbol{x}_0, \boldsymbol{u}_0)$ as in Eq. (25), and rewritten by considering external disturbances as below:

$$\dot{\boldsymbol{x}} = \dot{\boldsymbol{x}}_0 + F(\boldsymbol{x}_0, \boldsymbol{u}_0)\Delta \boldsymbol{x} + \boldsymbol{g}(\boldsymbol{x}_0)\Delta \boldsymbol{u} + \boldsymbol{\sigma}, \tag{51}$$

$$\dot{\boldsymbol{y}} = \nabla \boldsymbol{h}(\boldsymbol{x})[\dot{\boldsymbol{x}}_0 + F(\boldsymbol{x}_0, \boldsymbol{u}_0)\Delta \boldsymbol{x} + \boldsymbol{g}(\boldsymbol{x}_0)\Delta \boldsymbol{u} + \boldsymbol{\sigma}], \tag{52}$$

where $F(\boldsymbol{x}_0, \boldsymbol{u}_0) = \frac{\partial[\boldsymbol{f}(\boldsymbol{x}) + \boldsymbol{g}(\boldsymbol{x})\boldsymbol{u})]}{\partial \boldsymbol{x}}|_{\boldsymbol{x}_0, \boldsymbol{u}_0}$.

According to the time scale separation principle, if a fast control input is performed, the change rate of $\boldsymbol{x}$ can be considered much slower than $\boldsymbol{u}$, i.e., $\Delta \boldsymbol{x} \ll \Delta \boldsymbol{u}$. And we can also reasonably assume that $\boldsymbol{x} \approx \boldsymbol{x}_0$ in a small time increment, resulting in

$$\dot{\boldsymbol{y}} = \nabla \boldsymbol{h}(\boldsymbol{x})[\dot{\boldsymbol{x}}_0 + \boldsymbol{g}(\boldsymbol{x}_0)\Delta \boldsymbol{u} + \boldsymbol{\sigma}]. \tag{53}$$

Subsequently, the incremental control law is designed as

$$\Delta \boldsymbol{u} = [\nabla \boldsymbol{h}(\boldsymbol{x})B_0]^{-1}[\boldsymbol{v} - \nabla \boldsymbol{h}(\boldsymbol{x})(\dot{\boldsymbol{x}}_0 + \hat{\boldsymbol{\sigma}})], \tag{54}$$

where $\hat{\boldsymbol{\sigma}}$ is the estimation of disturbances, and $\tilde{\boldsymbol{\sigma}} = \boldsymbol{\sigma} - \hat{\boldsymbol{\sigma}}$ is the estimation error. The external disturbance $\boldsymbol{\sigma}$ can be compensated by the feedforward signal $\hat{\boldsymbol{\sigma}}$ from a disturbance observer. Due to the convergence property of the disturbance observer, $\|\tilde{\boldsymbol{\sigma}}\|$ is bounded. Thus, Eq. (52) can be transformed into the closed-loop dynamics as

$$\dot{\boldsymbol{y}} = \boldsymbol{v} + \breve{\boldsymbol{\sigma}}(t), \tag{55}$$

where $\breve{\boldsymbol{\sigma}}(t) = \nabla \boldsymbol{h}(\boldsymbol{x})\tilde{\boldsymbol{\sigma}}$ is a bounded disturbance that will converge to zero asymptotically by using a disturbance observer.

Remark 3 The disturbance observer ensures effective feedforward compensation of unknown external disturbances, resulting in a bounded estimation error term in the closed-loop system. Notably, the stability of the proposed controller is maintained even without an observer, provided that the disturbances affecting the original system are bounded.

Remark 4 In the NDI control scheme, a PID controller is commonly applied to feedback linear control. However,

the selection of control parameters requires a certain level of experience and prior information. To ensure the anti-disturbance capability and convergence rate of the feedback control, a feedback control law with fractional power terms can be used to illustrate the control stability.

To formulate a feedback control law for the outer loop of the INDI control, we define the tracking error as $\boldsymbol{e} = \boldsymbol{y} - \boldsymbol{y}_d$ and the feedback gain matrix as $\boldsymbol{K} = \text{diag}(k_1, k_2, k_3) \in \mathbb{R}^{3\times3}$. In addition, the fractional power term $\lfloor \boldsymbol{e} \rceil^{\varrho} \in \mathbb{R}^3$ is introduced and defined as $\lfloor \boldsymbol{e} \rceil^{\varrho} = \left[|e_1|^{\varrho}\,\text{sign}(e_1) \;\; |e_2|^{\varrho}\,\text{sign}(e_2) \;\; |e_3|^{\varrho}\,\text{sign}(e_3)\right]^{\top}$. The fractional exponent ϱ in this control design raises each element of the vector $\boldsymbol{e}$ to the power of ϱ.

Theorem 1 *Assume that there exists a positive constant ι to satisfy $\|\check{\boldsymbol{\sigma}}(t)\| \leq \iota$. For a nonlinear system in the form of Eq. (55), design a non-smooth controller as $\boldsymbol{v} = -\boldsymbol{K}\lfloor \boldsymbol{e} \rceil^{\varrho} + \dot{\boldsymbol{y}}_d$, then the tracking error $\boldsymbol{e}$ will converge to a small region ($\Omega = \{\boldsymbol{e} : |e_i| \leq (\frac{c_i+\iota}{k_i})^{1/\varrho}\}$) in the neighborhood of origin equilibrium point within a finite time, where $k_i > 0, 0 < \varrho < 1$, c_i is an arbitrary positive constant.*

Proof Consider a candidate Lyapunov function as $V(\boldsymbol{e}) = \frac{1}{2}\boldsymbol{e}^{\top}\boldsymbol{e}$, we will have its derivative along trajectories of the system as follows:

$$\begin{aligned}
\dot{V}(\boldsymbol{e}) &= \boldsymbol{e}^{\top}(-\boldsymbol{K}\lfloor \boldsymbol{e} \rceil^{\varrho} + \check{\boldsymbol{\sigma}}) \\
&= \begin{pmatrix} e_1 & e_2 & e_3 \end{pmatrix} \begin{pmatrix} -k_1|e_1|^{\varrho}\text{sign}(e_1) + \check{\sigma}_1 \\ -k_2|e_2|^{\varrho}\text{sign}(e_2) + \check{\sigma}_2 \\ -k_3|e_3|^{\varrho}\text{sign}(e_3) + \check{\sigma}_3 \end{pmatrix} \\
&= \sum_{i=1}^{3}(-k_i|e_i|^{\varrho+1} + e_i\check{\sigma}_i) \\
&\leq -\sum_{i=1}^{3}(k_i|e_i|^{\varrho} - \iota)|e_i|,
\end{aligned} \tag{56}$$

where the disturbance is bounded and satisfies $\|\check{\boldsymbol{\sigma}}(t)\| \leq \iota$. When $|e_i| > (\frac{c_i+\iota}{k_i})^{1/\varrho}$, we will have

$$\dot{V}(\boldsymbol{e}) \leq -\sum_{i=1}^{3} c_i(\frac{c_i+\iota}{k_i})^{1/\varrho} < 0. \tag{57}$$

According to Lyapunov's direct method, the tracking error $\boldsymbol{e}$ is convergent and eventually converges to the corresponding compact set $\Omega = \{\boldsymbol{e} : |e_i| \leq (\frac{c_i+\iota}{k_i})^{1/\varrho}\}$. The proof of Theorem 1 is completed. □

Remark 5 The non-smooth control can be simplified as feedback control with constant proportional gains when $\varrho = 1$, whereby the stability of classical PID control can also be verified based on Theorem 1.

3.5.2 Robustness to model uncertainties

While the direct identification of the inverse matrix $G^{\dagger}$ offers increased efficiency and reduces computational errors in the inversion step of the traditional INDI approach, it introduces estimation errors of the inverted G matrix due to the errors-in-variables property when utilizing LS techniques. This estimation error becomes a source of model uncertainties in the incremental model identification process. Therefore, we explore a more general condition encompassing a broader range of model uncertainties. This evaluation assesses the stability and robustness of the approach in situations marked by both estimation errors and uncertainties in the underlying model.

The system dynamics with model uncertainty $\Delta\boldsymbol{g}$ can be written as

$$\dot{\boldsymbol{x}} \approx \dot{\boldsymbol{x}}_0 + [\boldsymbol{g}(\boldsymbol{x}_0) + \Delta\boldsymbol{g}(\boldsymbol{x}_0)]\Delta\boldsymbol{u} + \boldsymbol{\sigma}. \tag{58}$$

Substituting Eq. (54) into Eq. (58) results in a closed-loop system:

$$\begin{aligned}
\dot{\boldsymbol{x}} &\approx -\Delta\boldsymbol{g}(\boldsymbol{x}_0)\boldsymbol{g}(\boldsymbol{x}_0)^{-1}\dot{\boldsymbol{x}}_0 \\
&\quad +[\boldsymbol{I}_{n\times n} + \Delta\boldsymbol{g}(\boldsymbol{x})\boldsymbol{g}(\boldsymbol{x})^{-1}][\nabla\boldsymbol{h}(\boldsymbol{x})^{-1}\boldsymbol{v}] + \boldsymbol{\sigma} \\
&= -\boldsymbol{B}\dot{\boldsymbol{x}}_0 + (\boldsymbol{I}_{n\times n} + \boldsymbol{B})[\nabla\boldsymbol{h}(\boldsymbol{x})^{-1}\boldsymbol{v}] + \boldsymbol{\sigma}, \qquad (59) \\
\dot{\boldsymbol{y}} &= -\boldsymbol{B}\dot{\boldsymbol{y}}_0 + (\boldsymbol{I}_{n\times n} + \boldsymbol{B})\boldsymbol{v} + \boldsymbol{\sigma}, \qquad (60)
\end{aligned}$$

where $\boldsymbol{B} = \Delta\boldsymbol{g}(\boldsymbol{x}_0)\boldsymbol{g}(\boldsymbol{x}_0)^{-1}$.

Considering $\boldsymbol{y}_d$ as a constant vector, introducing the virtual feedback control $\boldsymbol{v} = -\boldsymbol{K}(\boldsymbol{y} - \boldsymbol{y}_d)$ yields

$$\dot{\boldsymbol{e}} = -\boldsymbol{B}\dot{\boldsymbol{e}} - (\boldsymbol{I}_{n\times n} + \boldsymbol{B})\boldsymbol{K}\boldsymbol{e} + \boldsymbol{\sigma}, \tag{61}$$

or

$$\dot{\boldsymbol{e}} = -\boldsymbol{K}\boldsymbol{e} + (\boldsymbol{I}_{n\times n} + \boldsymbol{B})^{-1}\boldsymbol{\sigma}. \tag{62}$$

Taking the Laplace transformation for both sides of Eq. (61) results in

$$s\boldsymbol{I}_{n\times n}\boldsymbol{y} = -s\boldsymbol{B}\boldsymbol{y} - (\boldsymbol{I}_{n\times n} + \boldsymbol{B})\boldsymbol{K}(\boldsymbol{y} - \boldsymbol{y}_d), \tag{63}$$

where the term $\boldsymbol{\sigma}$ is omitted, as the bounded disturbance does not alter the input–output relation. Thus, the transfer function of the closed-loop system in the presence of model uncertainties can be written as:

$$\frac{\boldsymbol{y}(s)}{\boldsymbol{y}_d(s)} = \frac{\boldsymbol{K}}{s\boldsymbol{I}_{n\times n} + \boldsymbol{K}}, \tag{64}$$

$$\frac{y_i(s)}{y_{d_i}(s)} = \frac{k_i}{s + k_i}, i = 1, 2, \ldots, n. \tag{65}$$

The model uncertainty term $\Delta \boldsymbol{g}$ or $\boldsymbol{B}$ does not appear in the transfer function of the closed-loop system. It shows that the presence of model uncertainty will not change the transfer function of the closed-loop system, indicating that the transition relationship between input and output of the system remains the same. Therefore, the stability characteristics of the closed-loop system, in the presence of model uncertainty, persist and can be guaranteed under the INDI control scheme.

Remark 6 Since the control input of the optical flow-based control dynamics can be transformed to the thrust vector $\boldsymbol{\mu} = \Upsilon(\boldsymbol{u})$, the original dynamic system can be regarded as affine in the input. Hence, the above robustness analysis based on the affine nonlinear system is also applicable to the presented MAV optical flow-based control system.

For the non-affine nonlinear system as in Eq. (29), the nonlinear term $\boldsymbol{g}(\boldsymbol{x}_0)$ in Eq. (58) can be modified as $G(\boldsymbol{x}_0, \boldsymbol{u}_0)$ presented by Eq. (37). Due to the involved incremental technique, the update of incremental control has no effect on $\boldsymbol{u}_0$, which is associated with $G(\boldsymbol{x}_0, \boldsymbol{u}_0)$. Thus, a similar analysis as the aforementioned affine system can account for the robustness of INDI in the case of a non-affine nonlinear system.

4 Numerical simulations

In this section, the feasibility and performance of the proposed optical flow-based control method will be demonstrated through numerical simulations, specifically addressing three-dimensional tracking scenarios. A comparative analysis is conducted between the INDI control with (1) the conventional approach involving the identification of the G matrix and its inverse computation at every time step and (2) the proposed method that directly identifies the inverse matrix $G^{\dagger}$. The simulations involve an MAV with a mass of $M = 1.2\ kg$, proportionality constants c_x, c_y, c_z set to 1, and an initial height of 3 m. To ensure a fair comparison, the initial parameters and the INDI outer loop implementations remain the same. The outer loop virtual input employs a linear feedback control with a constant proportional gain $K_p = 1$. All simulations are conducted in the MATLAB environment using a laptop computer equipped with an i7-8565U processor, operating at 1.8 GHz, and having 40 GB of RAM.

Figures 2 and 3 illustrate the performance of the INDI control with the two approaches and during the maneuver of an MAV landing on a moving platform, with a constant flow divergence setpoint $\vartheta_z^* = -0.1\ rad/s$. The reference platform undergoes three-dimensional translational movement in a sinusoidal pattern, as indicated by the dashed line in the first row of both figures. In the first approach, as shown in

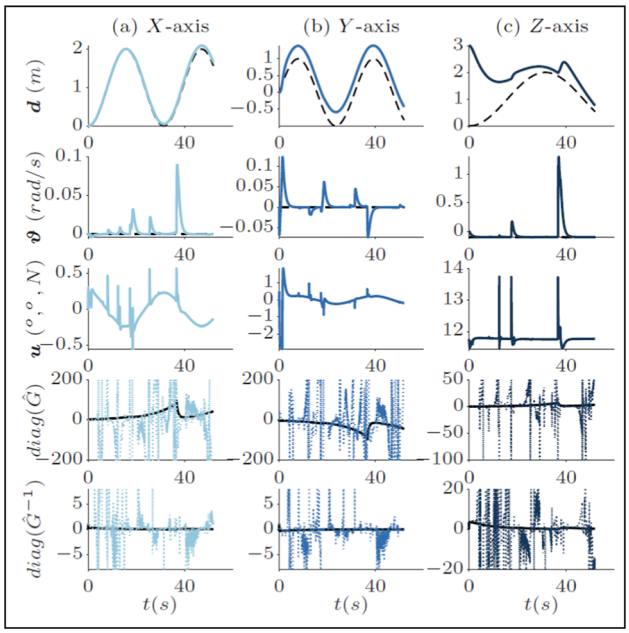

Fig. 2 The MAV executes a landing maneuver on a moving platform with INDI control, wherein the G matrix is identified and inverted at each step. Each column in the figure depicts the positions of the MAV $\boldsymbol{d}$ (the dashed line represents the position of the platform $\boldsymbol{d}^r$), optical flows $\boldsymbol{\vartheta}$ (the dashed line represents the tracked setpoint $\boldsymbol{\vartheta}^* = [0, 0, -0.1]$), control inputs $\boldsymbol{u} = [\theta, \phi, T]^T$, the diagonal elements of the identified matrix $\hat{G}$ (the solid line indicates the true value), and the diagonal elements of the calculated matrix inversion $\hat{G}^{-1}$ (the solid line indicates the true value) for the (**a**) X-axis, (**b**) Y-axis, and (**c**) Z-axis

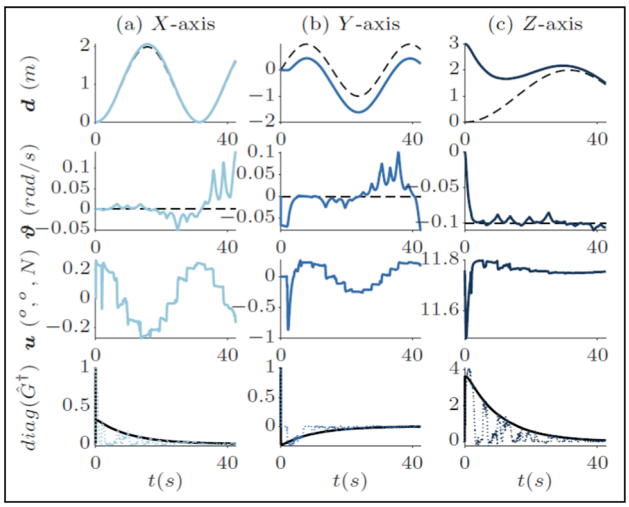

Fig. 3 The MAV executes a landing maneuver on a mobile platform under INDI control, directly identifying the inverse matrix $G^{\dagger}$. Each column in the figure depicts the positions of the MAV $\boldsymbol{d}$ (the dashed line represents the position of the platform $\boldsymbol{d}^r$), optical flows $\boldsymbol{\vartheta}$ (the dashed line represents the tracked setpoint $\boldsymbol{\vartheta}^* = [0, 0, -0.1]$), control inputs $\boldsymbol{u} = [\theta, \phi, T]^T$, and diagonal elements of the inverse matrix $\hat{G}^{\dagger}$ (the solid line represents the true value) for the (**a**) X-axis, (**b**) Y-axis, and (**c**) Z-axis

Fig. 2, the matrix G is identified using the RLS technique, and its inverse is calculated at each time step to generate control inputs according to Eq. (44). Considering the small values of the non-diagonal elements due to the small pitch angle θ and roll angle ϕ, as described in Eq. (37), the figures only depict the diagonal elements of the identified matrix $\hat{G}$ and its inversion $\hat{G}^{-1}$ with the dotted lines. And the solid lines represent the true values of the matrix G diagonal elements corresponding to motion in the X-, Y-, and Z-axes, calculated from Eq. (37).

As the MAV approaches the platform, the observed optical flow becomes highly sensitive to even minor input variations, leading to a rapid increase in the diagonal elements of the matrix G. Timely updates to the estimated G matrix are crucial to adapting the control law for a smooth landing. Failure to do so will result in degraded control performance, as depicted in Fig. 2. In this optical flow-based control task, especially when the height $\tilde{d}_z$ is small, the conventional method of identifying and inverting the matrix G is sensitive to the choice of the forgetting factor γ in the RLS approach. The forgetting factor needs to be chosen or tuned to a relatively small value to allow the fast change of the G matrix. To expedite parameter updates, a value of $\gamma = 0.8$ is employed in the simulation. However, this choice may introduce oscillations in the estimation. Conversely, larger values of γ impede the update rate of matrix G, resulting in worsened performance than presented in Fig. 2. Another practical approach involves setting a higher threshold for the height $\tilde{d}_z$ and shutting down the system when it falls below the threshold. However, this approach necessitates additional sensing and operations.

On the other hand, the inverse matrix $G^{\dagger}$, which is proportional to the height $\tilde{d}_z$, converges as the MAV approaches the landing platform, as depicted in Fig. 3. When the height $\tilde{d}_z$ is small, the directly identified $G^{\dagger}$ decreases correspondingly to small values, generating small control increments for a smooth landing of the MAV. In this scenario, increasing the forgetting factor γ may achieve a more stable parameter update than illustrated in the figure. Although both the matrix G and its inverse matrix $G^{\dagger}$ are time-varying, the direct identification of $G^{\dagger}$ is less sensitive to the choice of the forgetting factor in the RLS identification process and the threshold for the height $\tilde{d}_z$.

Furthermore, the proposed INDI control demonstrates computational efficiency, evidenced by a reduction in CPU time for a 50-second simulation. Specifically, the traditional INDI control with an inversion step requires $0.1062s$, whereas the proposed INDI control, which involves the direct identification of the inverse matrix $G^{\dagger}$, only needs $0.0681s$. This result is averaged over 20 simulations.

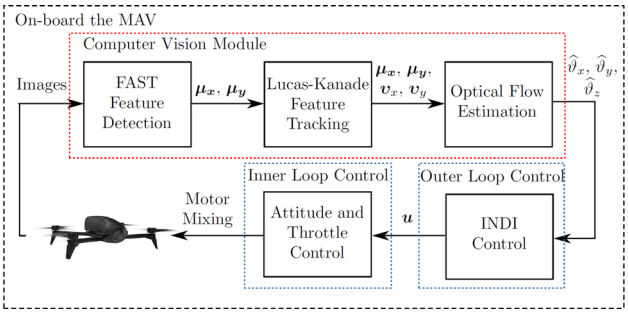

Fig. 4 Flight test setup using an MAV (Parrot Bebop 2) with an open source autopilot (Paparazzi). All the computer vision and control algorithms run onboard the MAV. In the vision module, FAST detects corners ($\boldsymbol{\mu_x}, \boldsymbol{\mu_y}$) in each image captured by the downward camera. Then, Lucas-Kanade tracks these corners in the next image and computes the flow vectors ($\boldsymbol{v_x}$, $\boldsymbol{v_y}$). Lastly, optical flow in the X^b-, Y^b-, Z^b-axes ($\widehat{\boldsymbol{\vartheta}}$) as defined in this paper are estimated. In the outer loop control module, the INDI controller receives $\widehat{\boldsymbol{\vartheta}}$ as the input and commands the attitude and throttle of the MAV ($\boldsymbol{u}$)

5 Flight tests

In this section, the experimental validation of the proposed INDI control scheme implemented on an MAV is presented. Several flight tests were conducted in an indoor environment under various challenging conditions: (1) landings with varying control parameters on a static and flat landing surface, and (2) hovering and landings over moving and undulating landing surfaces. The experimental setup and the results of all the flight tests are presented and discussed in this section.[1]

5.1 Flight test setup

The experimental setup employs a Parrot Bebop 2 MAV integrated with the Paparazzi autopilot software,[2] as illustrated in Fig. 4. Standard sensors, including an inertia measurement unit, a sonar sensor, and two cameras (front and bottom) are equipped on the MAV. The bottom camera (facing downward) is of particular interest for hovering and landing purposes. Throughout the tests, the sonar sensor was activated to record height data, which facilitates the evaluation of landing performance.

In the autopilot software, two modules were specifically created for the flight tests: the computer vision and the outer loop control, as depicted in Fig. 4. The computer vision module processes the images captured from the camera during the flight tests and implements computer vision algorithms, including the Features from Accelerated Segment Test (FAST) algorithm (Rosten & Drummond, 2006) for feature detection and the Lucas-Kanade tracker (Bouquet, 2000) for feature tracking.

[1] Video demos can be found at https://youtu.be/QX-9Hkagwr4

[2] Paparazzi Autopilot: http://wiki.paparazziuav.org

By taking the average of the flow vectors in the X^b- and Y^b-axes ($\boldsymbol{v}_x$, $\boldsymbol{v}_y$) obtained from the feature tracking, optical flow $\widehat{\vartheta}_x$ and $\widehat{\vartheta}_y$ can be computed. To estimate flow divergence $\widehat{\vartheta}_z$, the approach in Ho et al. (2018) utilizing Eq. (66) is employed. This involves computing the image distances of every two features between consecutive images κ, and then computing the average ratio of the change in the image distances to its previous image distance with a known time interval Δt, resulting in the estimation of flow divergence $\widehat{\vartheta}_z$ as follows:

$$\widehat{\vartheta}_z = \frac{1}{M} \cdot \frac{1}{\Delta t} \sum_{i=1}^{M} \left[\frac{\kappa_{(t-\Delta t),i} - \kappa_{t,i}}{\kappa_{(t-\Delta t),i}} \right], \tag{66}$$

where M is the total number of tracked corners. A low-pass filter was introduced to minimize the estimation noise.

The outer loop control module, which implements the INDI control scheme detailed in Sect. 3, receives output from the computer vision module and controls the attitudes and throttle of the MAV. The flight tests were started with manual take-off to a specified position and then switched to automatic mode, where only the required modules executed the designated tests. The height measurement from the sonar sensor was recorded for performance validation and analysis purposes, but not utilized in the control process.

5.2 Static and flat landing surface

Multiple landing tests were conducted for the MAV on a static and flat landing surface. The proposed INDI controller was employed to track three different setpoints of flow divergence, i.e., $-0.1\ rad/s$, $-0.2\ rad/s$, and $-0.3\ rad/s$. Figures 5 and 6 present the results of the flight tests, where the initial conditions were varied to assess the controller's robustness. The former set of tests initiated from a nearly static or hovering state ($v_{z_0} \approx 0$), while the latter began with pre-existing motion ($v_{z_0} \neq 0$). As the setpoint value increases, the landing velocity also increases, making the tracking task more demanding for the controller as it must efficiently accommodate the rapidly changing velocity of the MAV.

The results depicted in both Figs. 5 and 6 clearly demonstrate the ability of the controller to accurately track the designated setpoints. It also exhibits a quick response to even larger setpoints, indicating its robustness and efficiency. As a result, for all landings, both height and vertical velocity of the MAV decay exponentially to zero at almost the same time, as desired. Since this control implementation is designed specifically for vertical landings, the inverse matrix $\widehat{G}^{\dagger}$ contains only a single element. It is noteworthy that the identified $\widehat{G}^{\dagger}$ exhibits an exponential decay and convergence, analogous to the change in height. This observation supports the validity

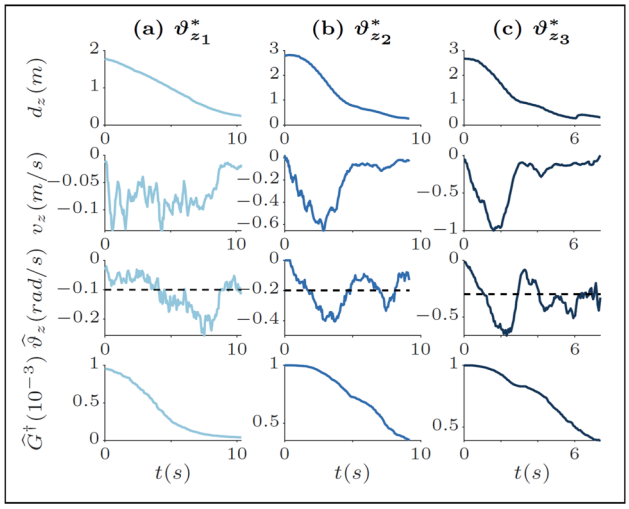

Fig. 5 INDI control from hovering ($v_{z_0} \approx 0$) **to landing:** by tracking different flow divergence setpoints: (**a**) $\vartheta^*_{z_1} = -0.1\ rad/s$, (**b**) $\vartheta^*_{z_2} = -0.2\ rad/s$, (**c**) $\vartheta^*_{z_3} = -0.3\ rad/s$. Each row presents the MAV height d_z, vertical velocity v_z, flow divergence $\widehat{\vartheta}_z$, and inverse matrix $\widehat{G}^{\dagger}$ for the respective flight test. All the tracked setpoints (ϑ^*_z) are plotted as dashed lines

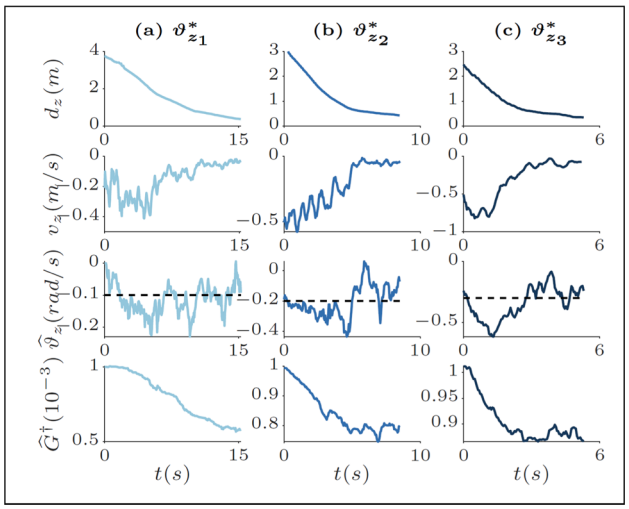

Fig. 6 INDI control from moving ($v_{z_0} \neq 0$) **to landing:** by tracking different flow divergence setpoints: (**a**) $\vartheta^*_{z_1} = -0.1\ rad/s$, (**b**) $\vartheta^*_{z_2} = -0.2\ rad/s$, (**c**) $\vartheta^*_{z_3} = -0.3\ rad/s$. Each row presents the MAV height d_z, vertical velocity v_z, flow divergence $\widehat{\vartheta}_z$, and inverse matrix $\widehat{G}^{\dagger}$ for the respective flight test. All the tracked setpoints (ϑ^*_z) are plotted as dashed lines

of the theoretical derivation of $G^{\dagger}$, which is a function of height.

To validate the superiority of the proposed INDI control scheme over commonly used controller for optical flow landing, landing tests were carried out by tracking a flow divergence setpoint of $\vartheta^*_{z_1} = -0.1\ rad/s$ with varying fixed control gains (ranging from small to large: 0.03, 0.1, and 0.5), using the control law outlined in Eq. (67):

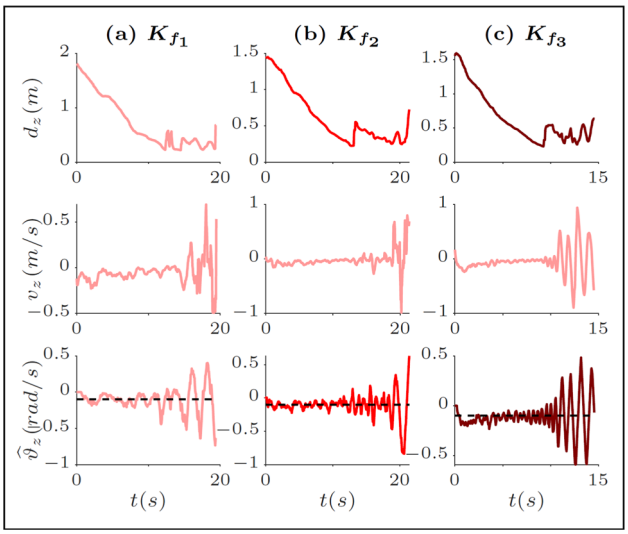

Fig. 7 Fixed-gain landing control: by tracking a flow divergence setpoint of $\vartheta^*_{z_1} = -0.1\ rad/s$ for different control gains: (**a**) $K_{f_1} = 0.03$, (**b**) $K_{f_2} = 0.1$, (**c**) $K_{f_3} = 0.5$. Each row presents the MAV height d_z, vertical velocity v_z, and flow divergence $\widehat{\vartheta}_z$ for the respective flight test.The tracked setpoints ($\vartheta^*_{z_1}$) are plotted as dashed lines

$$\boldsymbol{u} = -K_f(\boldsymbol{\vartheta}_z - \boldsymbol{\vartheta}^*_z), \tag{67}$$

K_f is the fixed gain used in the classical feedback control.

In Fig. 7, the landing test results with the classical fixed gain control are presented. Each of the three landings exhibits a consistent exponential delay in both height and vertical velocity, accompanied by oscillations at a specific height near the ground. A comparison of control gains, ranging from the smallest to the largest values, reveals a quicker onset of instability with larger gains. This observation suggests that opting for smaller gains might prevent oscillations during optical flow landing (Croon, 2016; Ho et al., 2018).

Table 1 presents the root mean square errors (RMSE) for tracking of a flow divergence of $\vartheta^*_{z_1}$ using fixed-gain control with different control gains and with the proposed INDI control under various initial conditions. In the case of fixed-gain control, it is observed that lowering the control gains results in higher RMSE, which indicates poorer tracking performance. This implies that adjusting the control gains for fixed-gain control, either by increasing or decreasing them, does not offer a compromise between stability and tracking performance.

Moreover, when compared to the proposed INDI approach, the RMSE for tracking the flow divergence is significantly smaller, which signifies superior performance. Interestingly, the RMSE for tracking landings initiated with initial motion ($v_{z_0} \neq 0$) is smaller than that for landings starting from a hovering state ($v_{z_0} \approx 0$). This observation suggests that, with a specific excitation at the commencement of the landing, the proposed INDI control scheme demonstrates enhanced performance.

5.3 Moving and undulating landing surface

Figure 8 illustrates the results of hovering and landing tests performed on a dynamically changing landing surface, featuring wave-like fluctuations. These flight tests aimed to validate the efficiency and robustness of the proposed INDI control scheme in handling lateral and vertical movements of the landing surface

The first test, presented in Fig. 8a, involved hovering the MAV over a surface undergoing lateral movement by utilizing the controller to regulate the flow divergence ($\vartheta^*_z = 0\ rad/s$) and optical flow in the Y^b-axis ($\vartheta^*_y = 0\ rad/s$). The results in this figure show that both setpoints were well tracked using the INDI controller even for a laterally moving surface, as demonstrated by the consistent height and unchanged position of the MAV relative to the surface feature, visible in the stacked image.

Besides the regulation task, Fig. 8b depicts the results of a landing test on a surface undergoing lateral movement, rather than hovering. The controller tracked a constant flow divergence setpoint, i.e., $\vartheta^*_z = -0.2\ rad/s$ (selected as the median value of the setpoints from previous landing tests in Sect. 5.2) for landing and maintaining the regulation of the optical flow in the Y^b-axis ($\vartheta^*_y = 0\ rad/s$) for following the moving surface. The results indicate that the controller was adapting the changes at the beginning, as the estimated $\widehat{\vartheta}_z$ shows a slower response leading to a minor height fluctuation at $t \approx 4\ s$. Eventually, the controller accurately tracked the setpoint ϑ^*_z and landed the MAV with an exponential decay of the height d_z.

To assess the controller under more challenging conditions, the last flight test stimulated the vertical movement of the landing surface by translating the MAV over an undulating terrain, as presented in Fig. 8c. The lateral movement of the MAV was induced by tracking a setpoint of optical

Table 1 Root mean square error (RSME) comparison between fixed-gain with various control gains K_f and INDI control at different initial vertical velocities v_{z_0} of the MAV

MAV landings by tracking $\vartheta^*_{z_1}$ with					
Control	Fixed-gain			INDI	
	$K_{f_1} = 0.03$	$K_{f_2} = 0.1$	$K_{f_3} = 0.5$	$v_{z_0} \approx 0$	$v_{z_0} \neq 0$
RSME	0.1694	0.1664	1.592	0.0567	0.0429

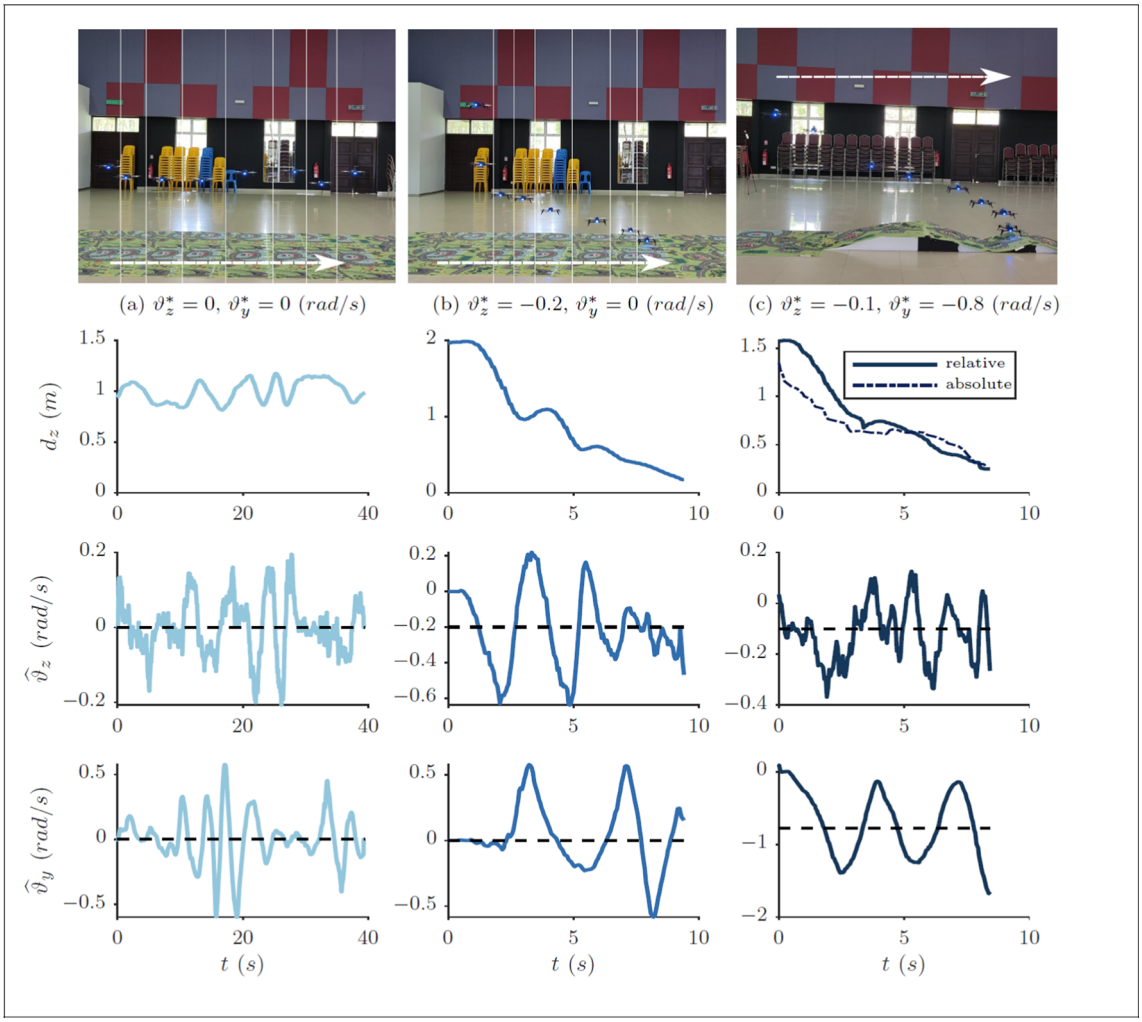

Fig. 8 MAV Hovering and Landings on a moving and undulating ground. The columns (**a**) and (**b**) depict the hovering (by tracking $\vartheta_z^* = 0\ rad/s$, $\vartheta_y^* = 0\ rad/s$) and landing (by tracking $\vartheta_z^* = -0.2\ rad/s$, $\vartheta_y^* = 0\ rad/s$) results of the MAV, respectively, above a moving and flat surface, while the column (**c**) displays the landing (by tracking $\vartheta_z^* = -0.1\ rad/s$) results of the MAV during translation (by tracking $\vartheta_y^* = -0.8\ rad/s$) over an undulating ground. Each column shows the stacked images of the flight scene, MAV height d_z, flow divergence $\widehat{\vartheta}_z$, and optical flow in the Y^b-axis $\widehat{\vartheta}_y$ for the respective flight test. All the tracked setpoints (ϑ_z^*, ϑ_y^*) are plotted as dashed lines

flow in the Y^b-axis ($\vartheta_y^* = -0.8\,rad/s$). To demonstrate that the MAV can safely land on the uneven terrain, a flow divergence setpoint $\vartheta_z^* = -0.1\,rad/s$ was chosen. Since the sonar measurement only provides the height relative to the landing surface, an ultra-wideband (UWB) tag attached to the MAV, along with four UWB anchors in the test area, measured the absolute height of the MAV using Double-Sided Two-Way Ranging (Ching et al., 2022). The absolute height allows us to analyze at which instance the MAV changing its velocity or even stopping. The results indicate that the controller adapted well to the translation over an undulating terrain as the tracking of the flow divergence ϑ_z^* is accurate leading to the exponential decay of the relative height, as desired. Additionally, the absolute height data revealed that the MAV was descending in a slow pace when it encountered the first peak ($t \approx 4\ s$). This is due to the fact that the MAV detected its movement closer to the landing surface at that instance. After it moved over the first peak, the MAV continued to descend and land safely on the next peak of the undulating surface while translating.

6 Conclusion

This paper introduced a a novel data-driven incremental nonlinear dynamic inversion (INDI) method designed to tackle challenges in nonlinear optical flow-based control, particularly when dealing with rapidly changing and height-dependent system output dynamics. A noteworthy challenge is that the control effectiveness matrix G in the INDI for optical flow-based control is determined to be time-varying, correlating with the distance above the platform, a parameter

not directly measurable. As a result, an online identification of G from input and output data, followed by an inversion, at every time step was deemed necessary for calculating the control increments. To streamline this process and reduce the likelihood of numerical errors while enhancing the accuracy and stability of the data-driven INDI approach, we proposed the direct identification of the inverse matrix $G^{\dagger}$, which eliminates the need for a separate inversion step. The proposed method was applied and compared with the conventional approach in numerical simulations aimed at controlling an MAV during landing on a moving platform. The results revealed enhanced control performance and improved computational efficiency with the proposed method. The effectiveness of the proposed method was further substantiated through real-world flight tests, successfully demonstrating its capability to track and land on various surfaces with the desired height and velocity decay. In future work, emphasis will be placed on: (1) extending the incremental model to handle challenges, such as measurement noises, delays, scaleless system outputs, or fast dynamics, and (2) enhancing this efficient optical flow-based control method by integrating a directional guidance strategy that leverages additional visual cues from both frontal and downward perspectives, which enable precise navigation of small flying robots in complex, obstacle-dense environments.

Acknowledgements The first and corresponding authors would like to thank Malaysian Ministry of Higher Education (MOHE) for providing the Fundamental Research Grant Scheme (FRGS) (Grant number: FRGS/1/2020/TK0/USM/03/3) for conducting this research.

Author Contributions H.W.H. and Y.Z wrote the main manuscript text. Y.Z. performed the simulations. H.W.H. carried out the implementation and experiments. Y.Z. and Y.F. conducted theoretical analysis. H.W.H., Y.Z, and G.C.H.E.d.C discussed the results. G.C.H.E.d.C. reviewed the manuscript. H.W.H. and Y.Z administrated the project and acquired the funding support.

Declarations

Ethical statements Not applicable.

References

Bacon, B. J., Ostroff, A. J., & Joshi, S. M. (2001). Reconfigurable NDI controller using inertial sensor failure detection & isolation. *IEEE Transactions on Aerospace and Electronic Systems, 37*(4), 1373–1383.

Bouquet, J. Y. (2000). *Pyramidal implementation of the Lucas Kanade feature tracker*. Microprocessor Research Labs, Intel Corporation.

Cesetti, A., Frontoni, E., Mancini, A., Zingaretti, P., & Longhi, S. (2010). A vision-based guidance system for UAV navigation and safe landing using natural landmarks. *Journal of Intelligent and Robotic Systems, 57*, 233–257.

Ching, P. L., Tan, S. C., & Ho, H. W. (2022). Ultra-wideband localization and deep-learning-based plant monitoring using micro air vehicles. *Journal of Aerospace Information Systems, 19*(11), 717–728.

Collett, T. S. (2002). Insect vision: Controlling actions through optic flow. *Current Biology, 12*(18), R615–R617.

Curtis, Andrew G.., Strong, Billie, Steager, Edward, Yim, Mark, & Rubenstein, Michael. (2023). Autonomous 3D position control for a safe single motor micro aerial vehicle. *IEEE Robotics and Automation Letters, 8*(6), 3566–3573.

De Croon, G. C. H. E., Dupeyroux, J. J. G., De Wagter, C., Chatterjee, A., Olejnik, D. A., & Ruffier, F. (2022). Accommodating unobservability to control flight attitude with optic flow. *Nature, 610*(7932), 485–490.

de Croon, B., & G.C.H.E. (2016). Monocular distance estimation with optical flow maneuvers and efference copies: A stability-based strategy. *Bioinspiration & Biomimetics, 11*(1), 016004.

Dong, S., Lin, T., Nieh, J. C., & Tan, K. (2023). Social signal learning of the waggle dance in honey bees. *Science, 379*(6636), 1015–1018.

Falanga, D., Kleber, K., & Scaramuzza, D. (2020). Dynamic obstacle avoidance for quadrotors with event cameras. *Science Robotics, 5*(40), eaaz9712.

Fridovich-Keil, D., Bajcsy, A., Fisac, J. F., Herbert, S. L., Wang, S., Dragan, A. D., & Tomlin, C. J. (2020). Confidence-aware motion prediction for real-time collision avoidance1. *The International Journal of Robotics Research, 39*(2–3), 250–265.

Haykin, S. S. (2002). *Adaptive filter theory*. Pearson Education India.

Herissé, B., Hamel, T., Mahony, R., & Russotto, F.-X. (2011). Landing a VTOL unmanned aerial vehicle on a moving platform using optical flow. *IEEE Transactions on Robotics, 28*(1), 77–89.

Ho, H. W., & Zhou, Y. (2023). Incremental nonlinear dynamic inversion based optical flow control for flying robots: An efficient data-driven approach. In *Proceedings of robotics: Science and systems*, Daegu, Republic of Korea, July. https://doi.org/10.15607/RSS.2023.XIX.081.

Ho, H. W., de Croon, G. C. H. E., & Chu, Q. (2017). Distance and velocity estimation using optical flow from a monocular camera. *International Journal of Micro Air Vehicles, 9*(3), 198–208.

Ho, H. W., de Croon, G. C. H. E., van Kampen, E., Chu, Q. P., & Mulder, M. (2018). Adaptive gain control strategy for constant optical flow divergence landing. *IEEE Transactions on Robotics, 34*(2), 508–516.

Isidori, A. (2013). *Nonlinear control systems*. Berlin: Springer.

Kendoul, F. (2014). Four-dimensional guidance and control of movement using time-to-contact: Application to automated docking and landing of unmanned rotorcraft systems. *The International Journal of Robotics Research, 33*(2), 237–267.

Mahlknecht, F., Gehrig, D., Nash, J., Rockenbauer, F. M., Morrell, B., Delaune, J., & Scaramuzza, D. (2022). Exploring event camera-based odometry for planetary robots. *IEEE Robotics and Automation Letters, 7*(4), 8651–8658.

Mehdi Yadipour, Md., Billah, A., & Faruque, I. A. (2023). Optic flow enrichment via drosophila head and retina motions to support inflight position regulation. *Journal of Theoretical Biology, 562*, 111416.

O'Connell, M., Shi, G., Shi, X., Azizzadenesheli, K., Anandkumar, A., Yue, Y., & Chung, S.-J. (2022). Neural-fly enables rapid learning for agile flight in strong winds. *Science Robotics, 7*(66), eabm6597.

Rosten, E. & Drummond, T. (2006). Machine learning for high-speed corner detection. In *Computer Vision–ECCV 2006* (pp. 430–443). Springer.

Ruffier, F., & Franceschini, N. (2015). Optic flow regulation in unsteady environments: A tethered MAV achieves terrain following and targeted landing over a moving platform. *Journal of Intelligent & Robotic Systems, 79*, 275–293.

Sieberling, S., Chu, Q. P., & Mulder, J. A. (2010). Robust flight control using incremental nonlinear dynamic inversion and angular acceleration prediction. *Journal of Guidance, Control, and Dynamics, 33*(6), 1732–1742.

Slotine, Jean-Jacques E., Li, Weiping, et al. (1991). *Applied nonlinear control*. Englewood Cliffs: Prentice Hall.
Smeur, E. J. J., de Croon, G. C. H. E., & Chu, Q. (2018). Cascaded incremental nonlinear dynamic inversion for MAV disturbance rejection. *Control Engineering Practice, 73*, 79–90.
Smith, P. (1998). A simplified approach to nonlinear dynamic inversion based flight control. In *23rd atmospheric flight mechanics conference* (pp. 4461).
Soria, E. (2022). Swarms of flying robots in unknown environments. *Science Robotics, 7*(66), eabq2215.
Steffensen, Rasmus, Steinert, Agnes, & Smeur, Ewoud JJ. (2022). Nonlinear dynamic inversion with actuator dynamics: An incremental control perspective. *Journal of Guidance, Control, and Dynamics* (pp. 1–9).
van't Veld, R., Van Kampen, E.-J. &d Chu, Q. (2018). Stability and robustness analysis and improvements for incremental nonlinear dynamic inversion control. In *AIAA guidance. Navigation, and control conference*, p.1127.
Wang, X., Van Kampen, E. J., Chu, Q., & Peng, L. (2019). Stability analysis for incremental nonlinear dynamic inversion control. *Journal of Guidance, Control, and Dynamics, 42*(5), 1116–1129.
Yu, Z., Zardini, G., Censi, A., & Fuller, S. (2022). Visual confined-space navigation using an efficient learned bilinear optic flow approximation for insect-scale robots. In *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)* (pp 4250–4256). IEEE.
Zhou, Y. (2023). Efficient online globalized dual heuristic programming with an associated dual network. *IEEE Transactions on Neural Networks and Learning Systems, 34*(12), 10079–10090. https://doi.org/10.1109/TNNLS.2022.3164727
Zhou, Y., Ho, H. W., & Chu, Q. (2021). Extended incremental nonlinear dynamic inversion for optical flow control of micro air vehicles. *Aerospace Science and Technology, 116*, 106889.

Hann Woei Ho received his B.Eng. degree in Aerospace Engineering from Universiti Sains Malaysia, Malaysia in 2009 and M.S. degree in Aerospace Engineering (cum laude), specialized in Control and Simulation from Delft University of Technology, the Netherlands in 2012. He obtained his Ph.D. degree in Aerospace Engineering at the Micro Air Vehicle lab (MAV-lab) of Delft University of Technology in 2017, with the topic of autonomous landing of Micro Air Vehicles through bio-inspired monocular vision. His research interests include bio-inspired vision and control of MAVs, machine learning-based control strategies, state estimation, and MAV design. Currently, he is a Senior Lecturer and UAV lab manager at the School of Aerospace Engineering of Universiti Sains Malaysia, Malaysia.

Ye Zhou received the B.S. and M.S. degrees (Hons.) from the School of Mechanical and Electrical Engineering, Northwestern Polytechnical University, Xi'an, China, in 2010 and 2013, respectively, and the Ph.D. degree in control and simulation, from the Faculty of Aerospace Engineering, Delft University of Technology, Delft, The Netherlands, in 2018. Her Ph.D. research topic was online reinforcement learning control for aerospace systems. She was a Lecturer with the Faculty of Aerospace Engineering, Delft University of Technology, from 2017 to 2018. She is currently a Senior Lecturer with the School of Aerospace Engineering, Universiti Sains Malaysia. Her research interests lie in reinforcement learning, nonlinear control, adaptive control, intelligent control, guidance, and navigation.

Yiting Feng received his B.S. degree in Nanjing University of Aeronautics and Astronautics, Nanjing, China, in 2017, and his M.S. degree in Northwestern Polytechnical University, Xi'an, China, in 2020. He is currently pursuing a Ph.D. degree at the School of Aerospace Engineering, Universiti Sains Malaysia, Malaysia. His current research interests include reinforcement learning, nonlinear control, optimal control and their applications to aerial vehicles.

Guido C. H. E. de Croon received his M.Sc. and Ph.D. in the field of Artificial Intelligence (AI) at Maastricht University, the Netherlands. Currently, he is Full Professor at TU Delft and scientific lead of the Micro Air Vehicle lab (MAVLab) of Delft University of Technology. His research interest lies with computationally efficient, bio-inspired algorithms for autonomous, light-weight flying robots, with an emphasis on computer vision. His work has included fully autonomous flight of the 20-gram flapping wing drone "DelFly Explorer" and a swarm of tiny, 30-gram nano-copters able to explore unknown environments and localize gas leaks. Moreover, his group has made several advances in energy-efficient, low-latency neuromorphic sensing and processing for autonomous drones. Finally, his work has generated various new hypotheses on biological intelligence, including how honeybees actively evaluate distances with optical flow and how flying insects can estimate their flight attitude without using accelerometers.