

# Robust Output Feedback Control with Predefined State Boundaries for Multi-Rotor Systems

## SUPPLEMENTARY MATERIAL

Gerardo Flores, *Senior Member, IEEE*, Almuatazbella M. Boker, *Member, IEEE*, Mohammad Al Janaideh, *Member, IEEE*, and Mark W. Spong, *Life Fellow, IEEE*

**Abstract**—This supplementary material provides additional content to support the main paper, including mathematical derivations, simulation results, performance graphs, and other complementary information. These materials offer further insight into the proposed control strategy and help validate the effectiveness of the approach presented in the main manuscript.

### I. VARIABLE DEFINITIONS

The system states are:

- $p = (p_x, p_y, p_z)^\top \in \mathbb{R}^3$ : the position vector of the system's origin expressed in the inertial frame  $\mathcal{I}$ .
- $v = (v_x, v_y, v_z)^\top \in \mathbb{R}^3$ : the system's translational velocity expressed in the inertial frame  $\mathcal{I}$ .
- $R = \begin{pmatrix} c_\theta c_\psi & s_\phi s_\theta c_\psi - c_\phi s_\psi & c_\phi s_\theta c_\psi + s_\phi s_\psi \\ c_\theta s_\psi & s_\phi s_\theta s_\psi + c_\phi c_\psi & c_\phi s_\theta s_\psi - s_\phi c_\psi \\ -s_\theta & s_\phi c_\theta & c_\phi c_\theta \end{pmatrix}$  is the rotation matrix from the body-fixed frame  $\mathcal{B}$  to the inertial frame  $\mathcal{I}$ , [1]. Here,  $c$  and  $s$  denote the trigonometric functions cosine and sine, respectively, where  $c_\theta = \cos \theta$ ,  $s_\theta = \sin \theta$ , and similarly for  $\phi$  and  $\psi$ . Notice that  $R \in \text{SO}(3)$ .
- $\Theta = (\phi, \theta, \psi)^\top$  are the Euler's angles.
- $\Upsilon = (\dot{\phi}, \dot{\theta}, \dot{\psi})^\top$  is the first-time derivative of the Euler angles

The control inputs are:

- $T \in \mathbb{R}_{\geq 0}$ : the total thrust exerted by all the rotors.
- $\tau = (u_{\tau_x}, u_{\tau_y}, u_{\tau_z})^\top \in \mathbb{R}^3$ : the control torque generated by the difference of individual thrusts of each rotor.

The aerial system's parameters are:

- $m$ : the aerial system's mass.
- $J = \text{diag}(I_x, I_y, I_z)$ : the aerial system's moment of inertia matrix in the body frame  $\mathcal{B}$ .
- $g$ : the gravity constant.
- $e_z = (0, 0, 1)^\top \in \mathbb{R}^3$

Gerardo Flores is with the RAPTOR Lab, School of Engineering, College of Arts and Sciences, Texas A&M International University, Laredo, TX 78041 USA (Corresponding author e-mail: gerardo.flores@tamu.edu).

Almuatazbella M. Boker is with the Bradley Department of Electrical and Computer Engineering, Virginia Tech, Blacksburg, VA 24060, USA. (e-mail: boker@vt.edu).

Mohammad Al Janaideh is with the School of Engineering, University of Guelph, Guelph, ON N1G 2W1, Canada. (e-mail: maljanai@uoguelph.ca).

Mark Spong is with the Department of Systems Engineering, University of Texas at Dallas, Richardson, TX 75080 USA (e-mail: mspong@utdallas.edu).

The unknown terms in system  $\Sigma$  are:

- $\Delta^v(t) \in \mathbb{R}^3$ : the unknown force due to unmodeled dynamics and exogenous disturbances.
- $\Delta^\Upsilon(t) \in \mathbb{R}^3$ : the unknown torque due to unmodeled dynamics and exogenous disturbances.

$$\text{Also, } f(\Upsilon) = \left( \dot{\theta} \psi \left[ \frac{I_y - I_z}{I_x} \right] \quad \dot{\phi} \psi \left[ \frac{I_z - I_x}{I_y} \right] \quad \dot{\phi} \dot{\theta} \left[ \frac{I_x - I_y}{I_z} \right] \right)^\top.$$

### II. COMPUTATION OF THE DESIRED EULER ANGLES

The primary control objective is to regulate the position of the drone in three-dimensional space by designing the desired Euler angles and the thrust magnitude as *virtual control input vector*. Specifically, this involves leveraging the SE(3) system  $\Sigma$  to strategically modify the rotational matrix  $R$  and thrust  $T$  to achieve the desired positioning dynamics. By appropriately designing these variables, the proposed approach ensures precise and stable control of the drone's position while addressing the inherent coupling between its rotational and translational dynamics. The attitude dynamics will be directly controlled by the control input  $\tau$ .

To leverage the advantages of the SE(3) system  $\Sigma$  and the double-integrator structure of the attitude subsystem, this section derives the desired Euler angles and the corresponding thrust magnitude for control. For that, notice that the control input vector  $-\frac{T}{m}Re_z$  has the form,

$$\begin{aligned} \underbrace{-\frac{T}{m}Re_z}_{\text{virtual control } u} &= - \begin{pmatrix} c_\theta c_\psi & s_\phi s_\theta c_\psi - c_\phi s_\psi & c_\phi s_\theta c_\psi + s_\phi s_\psi \\ c_\theta s_\psi & s_\phi s_\theta s_\psi + c_\phi c_\psi & c_\phi s_\theta s_\psi - s_\phi c_\psi \\ -s_\theta & s_\phi c_\theta & c_\phi c_\theta \end{pmatrix} \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} \frac{T}{m} \\ &= - \underbrace{\begin{pmatrix} c_\phi s_\theta c_\psi + s_\phi s_\psi \\ c_\phi s_\theta s_\psi - s_\phi c_\psi \\ c_\phi c_\theta \end{pmatrix}}_{\text{Direction } R(3)} \cdot \underbrace{\frac{T}{m}}_{\text{magnitude}}. \end{aligned} \quad (1)$$

where  $R = (R(1), R(2), R(3))$  and each  $R(i) \in \mathbb{R}^3$  for  $i = \{1, 2, 3\}$  is one of the three columns of the rotational matrix  $R$ . Notice that (1) clearly states that to modify the positioning dynamics, one must strategically modify the rotational matrix  $R$  and thrust  $T$ .

Designing the desired rotational matrix  $R_d$  is crucial for achieving the control objective of regulating the drone's translational dynamics. Specifically,  $R_d$  defines the orientation required to generate the control vector  $u = -\frac{T}{m}Re_z$ , which directly affects the drone's position. This matrix enables the control system to decouple and manage the inherent coupling between the attitude and position dynamics. To

formalize this, (1) is defined as the *virtual control input*  $u = (u_1, u_2, u_3)^\top$ , which partially depends on the attitude states. From this equation, it is evident that the direction of the virtual control vector  $u$  is defined by  $-R(3)$ , while its magnitude is  $\|u\| = \frac{T}{m}$ . The thrust  $T$  is then computed as:

$$T = m\|u\|. \quad (2)$$

This design aligns the drone's orientation with the required thrust direction while stabilizing its position. The following section describes the specific methodology for constructing  $R_d$ .

*1) Design of  $R_d$ :* As established in (1), the control vector  $u$  defines the desired thrust direction. Consequently, the third column of the desired rotation matrix,  $R_d(3)$ , is aligned with the normalized direction of  $u$ . This design ensures that the drone's thrust aligns with the control input necessary to achieve the desired translational dynamics. Thus, the goal is to find each of the orthonormal vectors  $R_d(1)$ ,  $R_d(2)$ , and  $R_d(3)$  that compose  $R_d = (R_d(1), R_d(2), R_d(3))$ . First, let us find  $R_d(3)$ . From (1) one can easily deduce that,

$$R_d(3) = -\frac{u}{\|u\|}. \quad (3)$$

where we assume that  $\|u\| \neq 0$ .

After selecting  $R_d(3)$  based on the direction of  $u$ , the rotational matrix  $R_d$  retains two degrees of freedom, corresponding to the remaining orthonormal vectors  $R_d(1)$  and  $R_d(2)$ . These vectors can be chosen to satisfy the orthonormality conditions and to define the desired heading direction, as described below.

To determine the desired rotation matrix  $R_d$ , we begin by selecting the desired direction of the  $x$ -body axis,  $x_d^B \in \mathbb{R}^3$ , which specifies the aerial robot's heading direction in the plane normal to  $R_d(3)$ . It is a standard result that the vectors  $R_d(3)$  and  $x_d^B$  are not parallel, ensuring the cross product  $R_d(3) \times x_d^B \neq 0$  due to the nature of the vectors involved. Using this property, we compute  $R_d(2)$  by projecting  $x_d^B$  onto the plane normal to  $R_d(3)$ :

$$R_d(2) = \frac{R_d(3) \times x_d^B}{\|R_d(3) \times x_d^B\|}. \quad (4)$$

Subsequently, the last orthonormal vector  $R_d(1)$  is determined using the right-hand rule:

$$R_d(1) = R_d(2) \times R_d(3). \quad (5)$$

This results in the desired rotational matrix  $R_d$ , defined as:

$$R_d = (R_d(1), R_d(2), R_d(3)). \quad (6)$$

Figure 1 illustrates these computations, highlighting the relationships between the heading vector  $x_d^B$  and the components of  $R_d$ .

Once we have a desired rotational matrix  $R_d$ , we use the following relation to find the vector of desired Euler angles

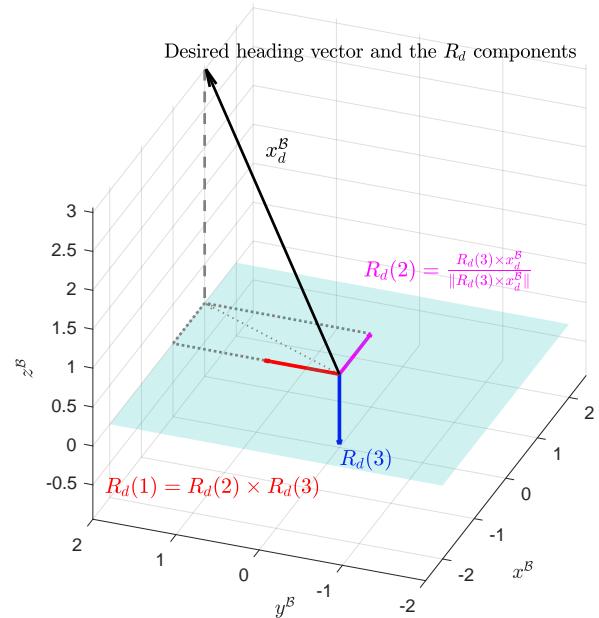


Fig. 1: The heading vector  $x_d^B$  and the vector elements of  $R_d$  described in section II.

$$\Theta^d = (\phi^d, \theta^d, \psi^d)^\top, [2]:$$

$$\begin{aligned} \phi^d &= \arctan \left( \frac{R_d(3, 2)}{R_d(3, 3)} \right) \\ \theta^d &= \arcsin (-R_d(3, 1)), \\ \psi^d &= \arctan \left( \frac{R_d(2, 1)}{R_d(1, 1)} \right). \end{aligned} \quad (7)$$

### III. ADDITIONAL MATHEMATICAL RESULTS

**Lemma 1.** *The inequality*

$$\frac{1}{\varphi^2 - x^2} - \frac{1}{2(\varphi^2 - x^2)^2} \geq 0 \quad (8)$$

holds for all  $x \in \left[ -\frac{\sqrt{2\varphi^2 - 1}}{\sqrt{2}}, \frac{\sqrt{2\varphi^2 - 1}}{\sqrt{2}} \right]$ , with  $\varphi > \frac{1}{\sqrt{2}}$ .

*Proof.* Let us rewrite (8) as  $\frac{2\varphi^2 - 2x^2 - 1}{(\varphi^2 - x^2)^2} \geq 0$ , where the equivalent inequality  $2\varphi^2 - 2x^2 - 1 \geq 0$  can easily be solved. From the solutions, the inequality holds as long as

$$\varphi > \frac{1}{\sqrt{2}}, \text{ and } -\frac{\sqrt{2\varphi^2 - 1}}{\sqrt{2}} \leq x \leq \frac{\sqrt{2\varphi^2 - 1}}{\sqrt{2}}. \quad (9)$$

□

### IV. SIMULATION RESULTS

This section provides a comprehensive simulation of the proposed approach using MATLAB Simulink. For illustrative purposes, we simulated a quadrotor aircraft, a common topic in recent studies such as [3] and [4]. The simulation experiments utilized the ODE5 (Dormand-Prince) solver with a fixed step size of 0.001 seconds to ensure high accuracy and stability.

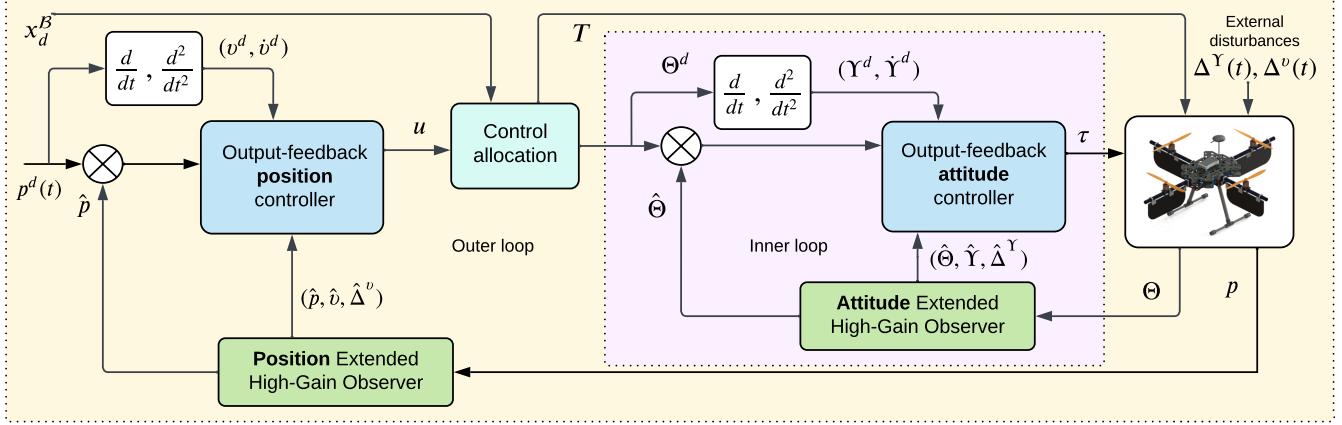


Fig. 2: Feedback control structure. Notice that the controller only use the available outputs ( $p, \Theta$ ) from the multi-rotor system.

TABLE I: Quadrotor parameters for the simulation.

Parameter	Value	Unit
$J$	$\text{diag}(0.0820, 0.0845, 0.1377)$	$\text{kgm}^2$
$m$	4.34	kg
$g$	9.8	$\text{m/s}^2$

#### A. Simulation conditions

The quadrotor parameters for the simulation were taken from [5] and are shown in Table I. The quadrotor is assumed to be influenced by external disturbances affecting the aircraft's orientation and position dynamics. The position disturbances are modeled as follows:

$$\begin{aligned}\Delta_1^v &= \vartheta_1 \sin(t) \\ \Delta_2^v &= \exp\left(-\frac{(t-b)^2}{2c^2}\right) \left( \frac{a \cos(r)}{\sqrt{\varphi^2(\sin(r))^2 + (\cos(r))^2}} \right) \\ \Delta_3^v &= \vartheta_3 \cos(t)\end{aligned}\quad (10)$$

where  $r = \omega t + 3.15\pi$ ,  $a = 40$ ,  $\omega = 3$ ,  $\varphi = 0.09$ ,  $c = 0.4$ ,  $b = 3.51$ ,  $\vartheta_1 = 1.8$ ,  $\vartheta_2 = 0.5$ ,  $\vartheta_3 = 1.4$ ,  $\vartheta_4 = 2.3$ . The disturbance  $\Delta^Y(t)$  for the orientation dynamics is modeled as follows:

$$\begin{aligned}\Delta_1^Y &= \vartheta_1 \cos(t) \\ \Delta_2^Y &= \vartheta_2 \sin(t) + \vartheta_2 \\ \Delta_3^Y &= \vartheta_3 \sin(t).\end{aligned}\quad (11)$$

Such disturbances are illustrated in Fig. 4(b).

The parameters of the proposed observers are depicted in Table II.

TABLE II: Observer parameters.

Position Observer			
Parameter	Value	Parameter	Value
$\epsilon_\alpha$	0.01	$\hat{\alpha}_2$	146
$\hat{\alpha}_1$	21	$\hat{\alpha}_3$	336
Attitude Observer			
Parameter	Value	Parameter	Value
$\epsilon_\beta$	0.05	$\hat{\beta}_2$	47
$\hat{\beta}_1$	12	$\hat{\beta}_3$	60

The control parameters for the position and attitude control parts are described in Tables III and IV, respectively.

TABLE III: Position control gains.

Parameter	Value	Parameter	Value
$k_1$	10	$\sigma_1$	8
$k_2$	10	$\sigma_2$	3
$k_3$	5	$\sigma_3$	4
$l_1^p$	0.097	$m_1^v$	0.485
$l_2^p$	1.746	$m_2^v$	1.455
$l_3^p$	1.746	$m_3^v$	1.164

TABLE IV: Attitude control gains.

Parameter	Value	Parameter	Value	Parameter	Value
$l_1^\Theta$	1.4	$\kappa_1$	2	$\mu_1$	7
$l_2^\Theta$	1.4	$\kappa_2$	10	$\mu_2$	7
$l_3^\Theta$	2.1	$\kappa_3$	3	$\mu_3$	7

All the system's initial conditions are zero except for the rotational matrix, which is  $R(0) = R_z(10)R_y(8.55)R_x(9.28)$  equivalently to  $\Theta = (0.1745, 0.1491, 0.1618)^\top$  radians. The initial conditions for all the observer states are also zero.

The desired position trajectory is an elliptical helix, which is given as follows together with its first two derivatives:

$$\begin{aligned}p^d &= (0.4t, 0.4 \sin(\pi t), -0.6 \cos(\pi t))^\top \\ \dot{p}^d &= v^d = (0.4, 0.4\pi \cos(\pi t), 0.6\pi \sin(\pi t))^\top \\ \ddot{p}^d &= \dot{v}^d = (0, -0.4\pi^2 \sin(\pi t), 0.6\pi^2 \cos(\pi t))^\top\end{aligned}\quad (12)$$

and it is depicted in red in Fig. 8.

#### B. Simulation results

First, let us study the behavior of position dynamics EHGO and the attitude dynamics EHGO. Those observers are responsible for estimating all the system states, including disturbances. The system states and their estimates are depicted in Fig. 3. In contrast, the disturbances and their estimates are depicted on the right-hand side of Fig. 4. As expected on this kind of observer, the EHGO can

respond overly to small perturbations or errors; this can be visualized during the transient response near  $t = 0$  in the orientation estimates of  $(\Theta, \Upsilon)$ . That phenomenon can be reduced by limiting the observer response using saturations and tuning the observer gains. In our case, we have not used saturations as proposed by Khalil in [6], except for the observer for the position disturbance  $\Delta^v(t)$  due to the excessive and aggressive disturbance that produces  $\Delta_2^v(t)$ ; see the middle plot of the right-hand side of Fig. 4. As suggested above, the estimated disturbances and their real values appear on the right-hand side of Fig. 4. Notice that all the time, the quadrotor is disturbed in all the states, and it is particularly affected during  $t = (2.5, 4.5)$  where a peak force of magnitude  $\pm 40N$  disturbs the quadrotor in the  $y$  axis.

Fig. 4a depicts the proposed output feedback control algorithm responses. On the left-hand side is the position control  $u$ , and on the right-hand side is the attitude control  $\tau$ . Notice the aggressive response in  $u_2$  during  $t = (2.5, 4.5)$  produced to compensate for the undesired effect of  $\Delta_2^v(t)$  in the quadrotor UAV. Notice that near  $t = 0$ , the attitude controller responds aggressively due to the distance between initial conditions and the desired position and orientation. Recall that the system under consideration is underactuated, and due to the inner/outer loop control architecture depicted in Fig. 2, the attitude controller should respond faster than the actual position controller, [7], [8]. That is why we can see a considerably more aggressive response in  $\tau$  than in  $u$ . The thrust computed as in (2) is depicted in Fig. 5.

Fig. 6 illustrates the attitude and position tracking errors under the proposed controller. The predefined bounds, specified by  $(l_i^p, m_i^v)$  for position and velocity tracking error states, and  $l_i^\Theta$  for angular velocity tracking error, are also shown. These bounds, established by the control engineer, provide critical performance limits for the system. As noted, the angular velocity tracking error was intentionally left unconstrained. Despite this, the bounds effectively guide the controller in rejecting disturbances and keeping the tracking error close to zero. Fig. 6 clearly demonstrates how these bounds contribute to robust performance, ensuring minimal error even in the presence of disturbances.

Finally, Fig. 7 demonstrates the proposed closed-loop system's behavior, showcasing the disturbance's observation error under different values of the high-gain observer parameter  $\epsilon$ . The graph shows the trajectories of the difference

$\hat{\Delta}_3^v - \Delta_3^v$ , where  $\Delta_3^v$  represents the disturbance affecting the altitude dynamics of the drone. Different values of  $\epsilon$  (ranging from 0.01 to 0.05) are plotted to highlight the effect of the observer gain on the disturbance estimation accuracy. The results indicate that lower values of  $\epsilon$  lead to faster convergence of the observation error under control, while higher values result in slower convergence but may reduce oscillations, reflecting the trade-off between responsiveness and stability in tuning the high-gain observer parameter  $\epsilon$ .

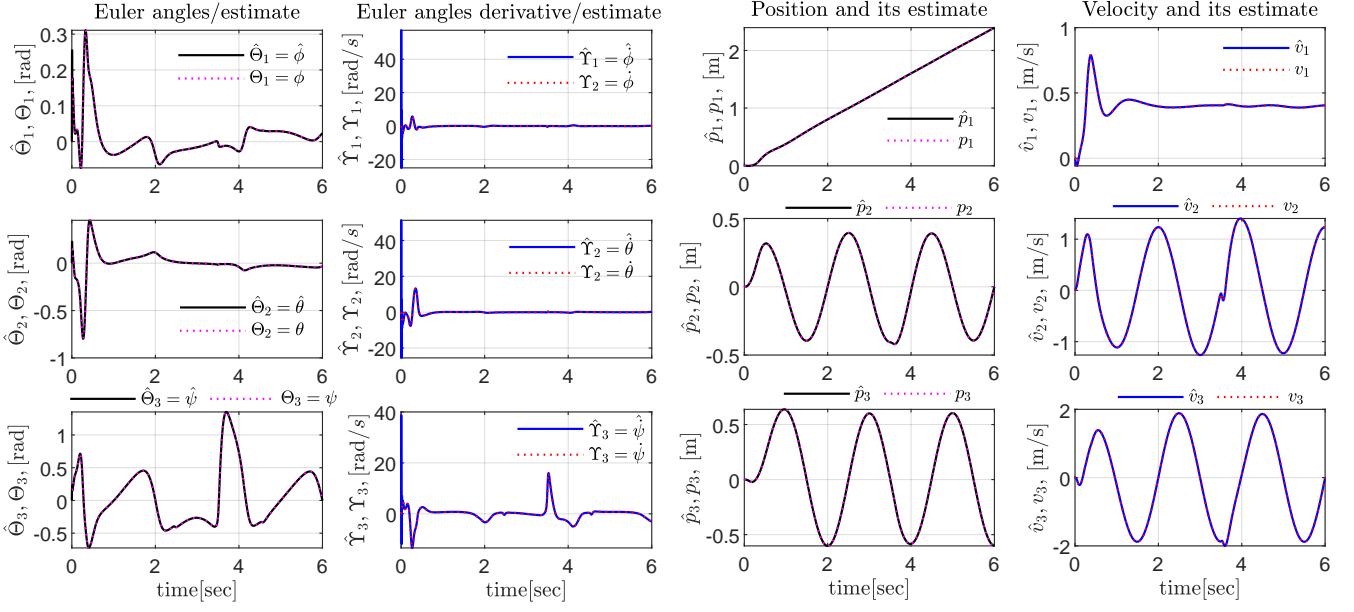
### C. Comparison

One of the most widely adopted controllers in autopilot systems is based on the Special Euclidean group  $SE(3)$ . Among these, the geometric controller proposed by T. Lee and colleagues [5] is particularly notable. This controller has demonstrated robustness in real UAV platforms. To evaluate its performance, we compare it with our proposed controller under the simulation conditions described above, using control parameters similar to those in [5].

Fig. 8 illustrates a 3D plot of an elliptical helix tracking the trajectory of the quadrotor UAV. Additionally, the right side of Fig. 8 displays the response of the geometric tracking control in  $SE(3)$  as presented in [5]. Despite the high efficiency of the geometric controller in many scenarios, our proposed controller demonstrates superior performance, particularly under aggressive perturbations.

## REFERENCES

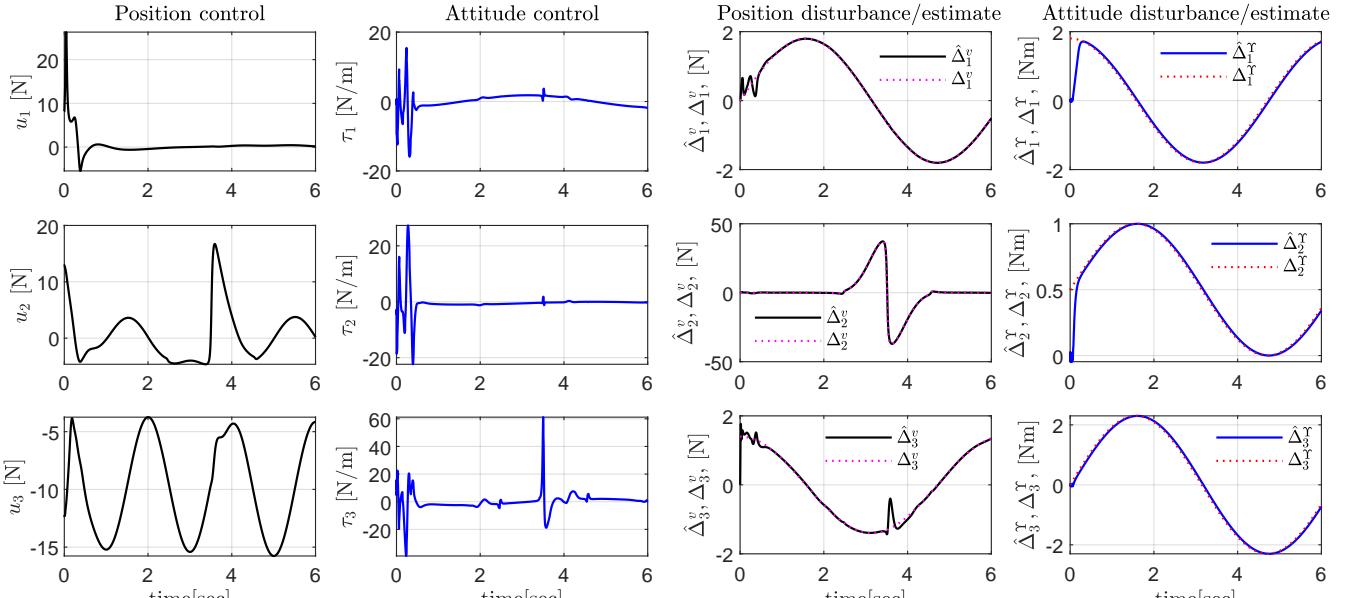
- [1] B. Etkin and L. Reid, *Dynamics of Flight Stability and Control*. John Wiley and Sons, 1996.
- [2] B. Stevens and F. Lewis, *Aircraft Control and Simulation*. John Wiley and Sons, 1992.
- [3] J. Lin, Z. Miao, Y. Wang, H. Wang, X. Wang, and R. Fierro, "Vision-based safety-critical landing control of quadrotors with external uncertainties and collision avoidance," *IEEE Transactions on Control Systems Technology*, vol. 32, no. 4, pp. 1310–1322, 2024.
- [4] T. Jin, J. Di, X. Wang, and H. Ji, "Safety barrier certificates for path integral control: Safety-critical control of quadrotors," *IEEE Robotics and Automation Letters*, vol. 8, no. 9, pp. 6006–6012, 2023.
- [5] T. Lee, M. Leok, and N. H. McClamroch, "Geometric tracking control of a quadrotor UAV on  $SE(3)$ ," in *49th IEEE Conference on Decision and Control (CDC)*, 2010, pp. 5420–5425.
- [6] H. Khalil, *Nonlinear Systems*. Upper Saddle River, NJ: Prentice Hall, 2002.
- [7] G. Flores and R. Lozano, "Lyapunov-based controller using singular perturbation theory: An application on a mini-UAV," in *2013 American Control Conference*, 2013, pp. 1596–1601.
- [8] S. Bertrand, N. Guénard, T. Hamel, H. Piet-Lahanier, and L. Eck, "A hierarchical controller for miniature vtol UAVs: Design and stability analysis using singular perturbation theory," *Control Engineering Practice*, vol. 19, no. 10, pp. 1099–1108, 2011.



(a) Attitude vs attitude estimates.

(b) Position vs position estimates.

Fig. 3: The output for these observers is the position vector  $p$  and the angular position  $\Theta$ . (a) The convergence of the observed attitude states, as estimated by the attitude EHGO, is illustrated. (b) The convergence of the estimated position and velocity states, as obtained using the position EHGO, is also illustrated.



(a) Control algorithm.

(b) Disturbances vs disturbances estimates.

Fig. 4: (a) The proposed control algorithms  $u$  and  $\tau$ . The control term  $u_2$  adjusts in response to the aggressive disturbance  $\Delta^v(t)$  illustrated in (b). (b) On the left, the position perturbation  $\Delta^v(t)$  and its estimate  $\hat{\Delta}^v(t)$ , obtained using EHGO, are shown. Note the significant magnitude of the aggressive perturbation  $\Delta_2^v(t)$ . On the right, the attitude perturbation  $\Delta^r(t)$  estimated with the attitude dynamics EHGO is depicted.

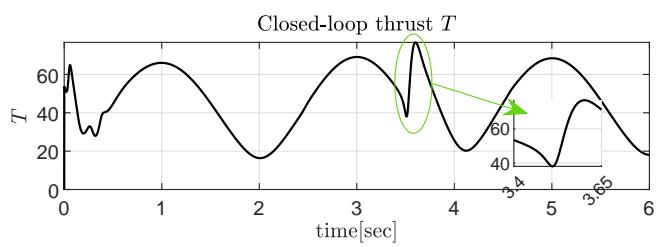
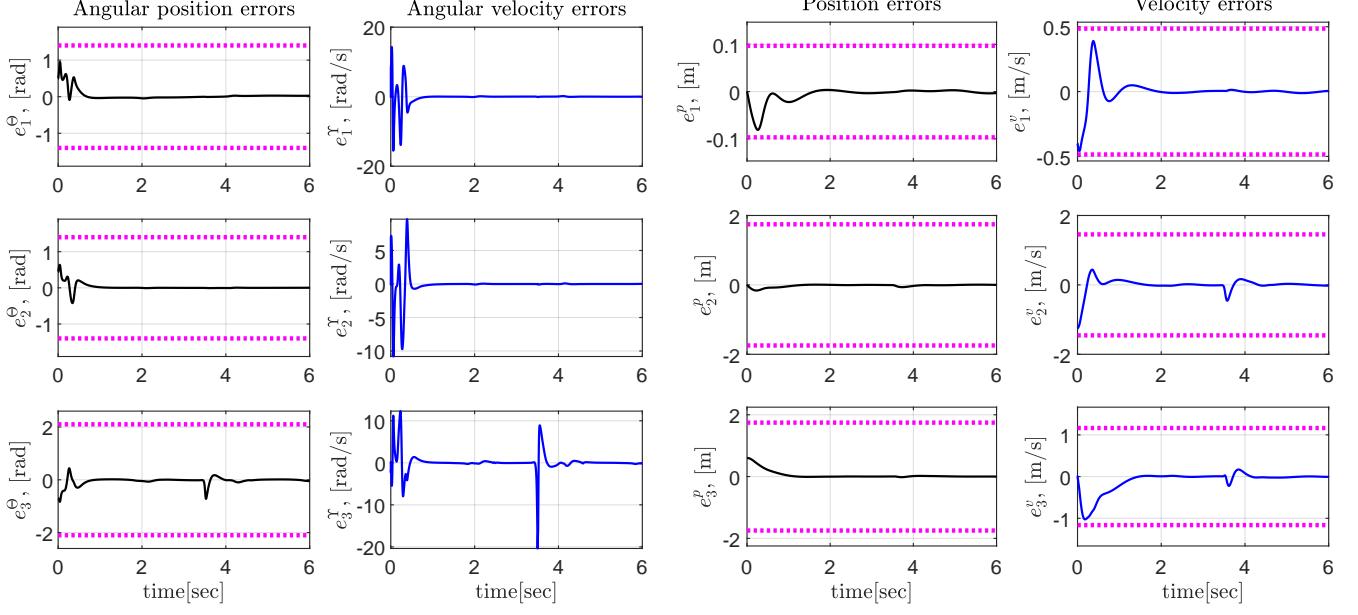


Fig. 5: The thrust computed by (2) is depicted. The aggressive, smooth response highlighted in the green circle corresponds to the disturbance  $\Delta_2^v$ .



(a) Attitude tracking errors.

(b) Position tracking errors.

Fig. 6: The position, velocity, and angular position errors exhibit rapid convergence with no noticeable transient responses or peaks, even in the presence of disturbances. Magenta dotted lines indicate the user-imposed constraints on position, velocity, and angular error states. (a) The attitude tracking errors are shown, with the angular velocity error intentionally unconstrained per the control algorithm. The angular position errors are bounded by the parameters  $l_i^\Theta$ , as detailed in Table IV, and are represented by magenta dotted lines. (b) The position and velocity errors respect the bounds  $l_i^p$  and  $m_i^v$ , respectively, as outlined in Table III and depicted by the magenta dotted lines. This observation is particularly evident in the initial phases of  $e_1^p$ ,  $e_1^v$ , and  $e_3^v$ .

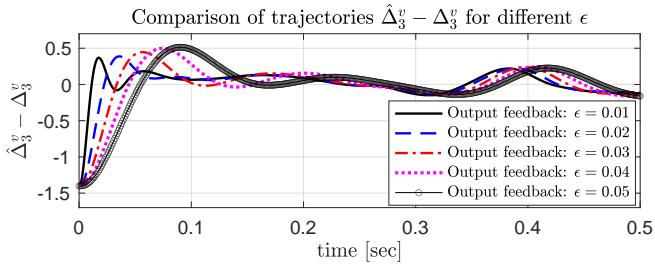


Fig. 7: Comparison of the trajectories of  $\hat{\Delta}_z^v - \Delta_z^v$  of the output feedback closed-loop system for different values of  $\epsilon$ .

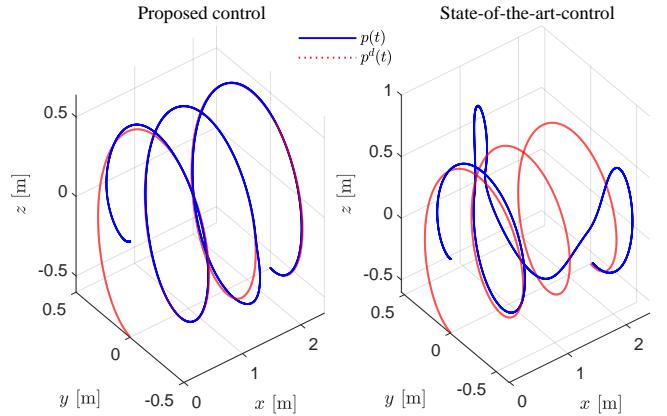


Fig. 8: The desired trajectory of an elliptical helix is in red. In the LHS, the tracking position trajectory is used using the proposed control. On the RHS, the position trajectory with geometric tracking control on  $SE(3)$  presented in [5]. For comparison, we used the same gains as in that work.