www.editada.org

International Journal of Combinatorial Optimization Problems and Informatics, 17(1), Jan-April 2026, 132-155. ISSN: 2007-1558. https://doi.org/10.61467/2007.1558.2026.v17i1.1230

Identification and control of a Tail-Sitter Unmanned Aerial Vehicle

Juan Carlos Álvarez González[1], Luis Enrique Ramos Velasco[2], Mario Alejandro Vega Navarrete[1], Carlos Roberto Domínguez Mayorga[3], Francisco Marroquín Gutiérrez[1], Pedro José Argumedo Teuffer[1].*
Universidad Politécnica Metropolitana de Hidalgo, 43860, Tolcayuca, Hidalgo, México
[2] Consejo de Ciencia, Tecnología e Innovación de Hidalgo, de Pachuca, 43830, San Agustín Tlaxiaca, Hidalgo, México
[3] Universidad Politécnica de Pachuca, 43830, Zempoala, Hidalgo, México
E-mails
182220022@upmh.edu.mx, lramosvelasco@gmail.com, mvega@upmh.edu.mx, carlos.dominguez@upp.edu.mx, fmarroquin@upmh.edu.mx, pargumedo@upmh.edu.mx.

Abstract. Unmanned aerial vehicles (UAVs) are increasingly used in fields such as military, search and rescue, and surveillance operations. Recent efforts have sought to combine the advantages of fixed-wing and rotor-wing aircraft. Tail-sitter UAVs are hybrid platforms that integrate rotor-wing vertical takeoff and landing with fixed-wing forward flight capabilities. They operate in three modes: vertical, horizontal, and transition. Depending on the transition mechanism, tail-sitters can be classified into mono thrust transitioning (MTT), collective thrust transitioning (CTT), and differential thrust transitioning (DTT). Despite their promise, UAVs face challenges in aerodynamic design and flight control, particularly regarding instability and sensitivity to disturbances. To address these issues, we propose an approach based on artificial intelligence and adaptive control. Specifically, we use neural networks with tunable parameters, such as inputs, weights and transfer functions, to enhance system robustness across operating conditions.
Keywords: Tail-sitter UAV; Neural networks; Adaptive control.

Article Info
Received Sep 14, 2025
Accepted November 14, 2025

1 Introduction

Unmanned aerial vehicles (UAVs) have become valuable tools across a wide range of applications, including military, search and rescue, entertainment, and surveillance operations. In recent years, efforts have intensified to combine the benefits of rotor-wing and fixed-wing platforms. Tail-sitter UAVs represent one such hybrid approach, successfully integrating vertical takeoff and landing (VTOL) with efficient forward flight. Tail-sitters operate in three flight modes: vertical, horizontal and transition (Saeed et al., 2018).

Despite recent progress, significant challenges remain in the areas of aerodynamic design and control. Tail-sitters are particularly susceptible to instabilities and disturbances. In transition flight, rapid changes in aerodynamic forces, control effectiveness, and flow separation introduce strong nonlinearities that complicate controller design. Even in vertical and horizontal modes, asymmetric loading and cross-coupling effects can degrade stability if not properly compensated. Even when modeled with simplified linear systems (Saeed et al., 2018) or addressed via nonlinear control strategies (Wang et al., 2015; Navarrete et al., 3 2018), conventional approaches may lack sufficient robustness under off-nominal conditions.

In this work, we propose an intelligent control approach for tail-sitter UAVs that includes online identification of the system dynamics in each flight mode. Our control architecture utilizes an artificial wavelet neural network (WNN) (Ramos et al., 2016), in which daughter wavelets are used as activation functions. This network structure enhances nonlinear approximation capability, improving the system's ability to estimate the dynamic behavior and compensate for disturbances, enhancing flight stability.

Among the most widely used control schemes, proportional-integral-derivative (PID) controllers remain popular due to their simplicity, explicit tuning procedures, and availability in commercial off-the-shelf systems. However, traditional PID controllers are designed for linear systems and exhibit limited robustness and convergence in the presence of nonlinearities (Åström &

Hägglund, 2009). Complex phenomena such as dead time, hard nonlinearities, or high-frequency dynamics pose particular difficulties. Moreover, PID tuning for nonlinear plants often requires explicit and nontrivial procedures.

The main objective of a PID controller is to minimize the error signal, defined as the difference between the reference and the output signals. A wide range of analytical and experimental methods are available to tune the PID gains (Åström & Hägglund, 2009). Among them, auto-tuning and self-tuning approaches have been developed to update controller gains online (Åström & Wittenmark, 2011; Islas et al., 2011), with WNN-based plant identification serving as the tuning basis.

Here, we extend the WNN-based identification and control methodology to multi-input multi-output (MIMO) systems using Morlet wavelets. Our main contribution is a novel identification and adaptive control scheme that enables autonomous operation of a tail-sitter UAV in longitudinal flight. The proposed approach includes two key features. First, adaptive wavelet PID (AWPID) gains are updated online based on real-time system identification. Second, both identification and control modules are designed for MIMO systems with unknown dynamics, enabling direct application without prior experimental modeling. We evaluate the method via numerical simulation using experimental parameters for a tail-sitter UAV (Edwards, 2014). Results show satisfactory performance in both trajectory tracking and disturbance rejection, demonstrating our method's effectiveness and applicability.

The rest of the papers is organized as follows. Section II presents a classification of hybrid UAVs. Section III presents the mathematical model. Section IV describes the WNN-based system identification and the control scheme. Section V presents numerical simulations results for a tail-sitter UAV based on experimental data.

2 Hybrid Unmanned Aerial Vehicles

Hybrid UAVs are typically classified into two types: convertible aircraft and tail-sitter aircraft. Convertible designs maintain a fixed structural orientation across all flight modes, relying on dedicated mechanisms to perform mode transitions. In contrast, tail-sitter designs take off and land vertically on their tails, with the entire airframe tilting to achieve forward cruise flight. Both types of aircraft can be further subdivided based on their transition mechanisms and airframe configurations. In what follows, we analyze the main subtypes, highlight their design features, discuss their respective advantages and disadvantages, and introduce representative examples.

Fig. 1. Classification of hybrid UAVs.

2.1 Convertible aircraft

Convertible UAVs employ various mechanisms to achieve transitions between vertical and horizontal flight. These platforms can be divided into four subtypes: tilt-rotor, tilt-wing, rotor-wing and dual-system, as shown in the first row of **Error! Reference source not found.**. Each subtype is described below:

1. Tilt-rotor
   A tilt-rotor UAV employs multiple rotors mounted on tilting shafts or nacelles. During the transition from hover to cruise, some or all of the rotors tilt towards the flight direction to generate thrust until cruising flight is achieved.
2. Tilt-wing
   A tilt-wing UAV employs a wing structure that tilts along with the rotors. During transition, the entire wing rotates to align with the forward direction, enabling smooth conversion between hover and cruise flight.
3. Rotor-wing

A rotor-wing or stop-rotor UAV employs a rotor that generates vertical lift during takeoff and hover. During cruise flight, the rotor stops and functions as a fixed wing, eliminating the need for separate lifting surfaces.

4. Dual system
   A dual-system UAV employs two independent propulsion systems: vertical thrust is provided by fixed rotors, and forward propulsion by dedicated propellers. Since no mechanical tilting is required, this configuration simplifies design and enhances reliability. However, the vertical rotors may introduce aerodynamic drag during cruise.

2.2 Tail-sitter aircraft

Tail-sitter UAVs combine the advantages of VTOL aircraft with those of fixed-wing vehicles. Their operation is typically divided into three flight modes: vertical, horizontal and transition. These platforms can be classified into three main subtypes based on their transition mechanism: mono thrust transitioning (MTT), collective thrust transitioning (CTT) and differential thrust transitioning (DTT), as shown in the second row of **Error! Reference source not found.**.

1. Mono Thrust Transitioning (MTT)
   A MTT tail-sitter UAV employs a single rotor, typically located either at the nose or rear of the fuselage, to generate thrust. Transition between hover and cruise flight is achieved via thrust vectoring mechanisms such as ducted fan vanes, cyclic or variable blade pitch propeller, or swashplates.
2. Collective Thrust Transitioning (CTT)
   A CTT tail-sitter UAV employs one or more fixed-pitch blade rotors. Flight mode transitions are achieved through coordinated control surface deflection and collective adjustment of rotor thrust. While the use of multiple rotors improves control authority and may outperform MTT configurations, only a few CTT platforms have reached notable levels of development. Remarkable examples include the T-Wing prototype developed by the University of Sydney and the VD200, developed by Chengdu Aircraft Research and Design Institute (CARDI) in China.
3. Differential Thrust Transitioning (DTT)
   A DTT tail-sitter UAV employs multiple rotors distributed above and below the horizontal plane of the airframe. During transition, differential change in rotor thrust produces pitching torque, enabling smooth mode conversion between vertical and forward flight.

3 Mathematical Model of the Tail-Sitter

This section presents the nonlinear dynamic model of the tail-sitter UAV. The model is used exclusively for numerical simulation in MATLAB and is not integrated into any control architecture. The vehicle under consideration is the V-bat vehicle, whose dynamics are described using a 12 state-space representation, defined by the following equations of motion:

$$\dot{P}n = u(\cos\theta\cos\psi) + v(\sin\phi\sin\theta\cos\psi) + w(\cos\phi\sin\theta\cos\psi + \sin\phi\sin\psi) \quad (1)$$

$$\dot{P}e = u(\cos\theta\sin\psi) + v(\sin\phi\sin\theta\sin\psi) + w(\cos\phi\sin\theta\sin\psi - \sin\phi\cos\psi) \quad (2)$$

$$\dot{P}d = u(-\sin\theta) + v(\sin\phi\cos\theta) + w(\cos\phi\cos\theta) \quad (3)$$

$$\dot{u} = rv - qw + \frac{Fx}{m} \quad (4)$$

$$\dot{v} = -ru + pw + \frac{Fy}{m} \quad (5)$$

$$\dot{w} = qu - pv + \frac{Fz}{m} \quad (6)$$

$$\dot{p} = \Gamma_1 pq + \Gamma_3 qr + \Gamma_3 l + \Gamma_4 n \quad (7)$$

$$\dot{q} = \Gamma_5 pr - \Gamma_6(p^2 - r^2) + \frac{m}{I_{yy}} \quad (8)$$

$$\dot{r} = \Gamma_7 pq + \Gamma_1 qr + \Gamma_4 l + \Gamma_8 n \quad (9)$$

$$\dot{\phi} = p + q\sin\psi\tan(\theta) + r\cos\phi\tan\theta \tag{10}$$

$$\dot{\theta} = q\cos\phi - r\sin\phi \tag{11}$$

$$\dot{\psi} = q\frac{\sin\phi}{\cos\theta} + r\frac{\cos\phi}{\cos\theta} \tag{12}$$

The external forces acting along the airframe axes are denoted by (fx, fy, fz) and the external moments by (l, m, n). The dynamic coupling coefficients Γ_1 through Γ_8 are defined as:

$$\Gamma = I_{xx}I_{zz} - I_{xx}^2 \qquad \Gamma_3 = \frac{I_{zz}}{\Gamma} \qquad \Gamma_6 = \frac{I_{xz}}{I_{yy}}$$

$$\Gamma_1 = \frac{\left(I_{xz}\left(I_{xx} - I_{yy} + I_{zz}\right)\right)}{\Gamma} \qquad \Gamma_4 = \frac{I_{xz}}{\Gamma} \qquad \Gamma_7 = \frac{\left(I_{xx} - \left(I_{yy}\right)I_{xx}\right) + I_{xz}^2}{\Gamma}$$

$$\Gamma_2 = \frac{I_{zz}\left(I_{zz} - I_{yy}\right) + I_{xz}^2}{\Gamma} \qquad \Gamma_5 = \frac{(I_{zz} - I_{xx})}{I_{yy}} \qquad \Gamma_8 = \frac{I_{xx}}{\Gamma}$$

According to (Edwards, 2014), the external forces and moments are defined as:

$$\mathrm{fx} = -\mathrm{mg}\sin\theta + \frac{1}{2}\rho V_a^2 \mathrm{S}\left(C_X(\alpha) + C_{X_q}(\alpha)\frac{c}{2V_a}q + C_{X_{\delta_e}}(\alpha)\delta_e\right) + \frac{1}{2}\rho S_{prop}C_{prop}((k_{motor}\delta_t)^2 - V_a^2) \tag{13}$$

$$fy = mg\cos\theta\sin\phi + \frac{1}{2}\rho V_a^2 S\left(C_{Y_0} + C_{Y_\beta}\beta + C_{Y_p}\frac{b}{2V_a}p + C_{Y_r}\frac{b}{2V_a}r + C_{Y_{\delta_a}}\delta_a + C_{Y_{\delta_r}}\delta_r\right) \tag{14}$$

$$fz = mg\cos\theta\cos\phi + \frac{1}{2}\rho V_a^2 S\left(C_Z(\alpha) + C_{Z_q}(\alpha)\frac{c}{2V_a}q + C_{Z_{\delta_e}}(\alpha)\delta_e\right) \tag{15}$$

$$l = \frac{1}{2}\rho V_a^2 S\left(b\left[C_{l_0} + C_{l_\beta}\beta + C_{l_p}\frac{b}{2V_a}p + C_{l_r}\frac{b}{2V_a}r + C_{l_{\delta_a}}\delta_a + C_{l_{\delta_r}}\delta_r\right]\right) + \frac{1}{2}\rho S_{prop}C_{prop}\left(-k_{T_p}(k_\omega\delta_t)^2\right) \tag{16}$$

$$m = \frac{1}{2}\rho V_a^2 S\left(c\left[C_{m_0} + C_{m_\alpha}\alpha + C_{m_q}\frac{c}{2V_a}q + C_{m_{\delta_e}}\delta_e\right]\right) \tag{17}$$

$$n = \frac{1}{2}\rho V_a^2 S\left(b\left[C_{n_0} + C_{n_\beta}\beta + C_{n_p}\frac{b}{2V_a}p + C_{n_r}\frac{b}{2V_a}r + C_{n_{\delta_a}}\delta_a + C_{n_{\delta_r}}\delta_r\right]\right) \tag{18}$$

The aerodynamic force and moment coefficients are computed using the following relations:

$$C_X(\alpha) = -C_D(\alpha)\cos\alpha + C_L(\alpha)\sin\alpha \qquad C_{X_q}(\alpha) = -C_{D_q}\cos\alpha + C_{L_q}\sin\alpha \qquad C_{X_\delta}(\alpha) = -C_{D_{\delta_e}}\cos\alpha + C_{L_{\delta_e}}\sin\alpha$$

$$C_Z\alpha = -C_D(\alpha)\sin\alpha + C_L(\alpha)\cos\alpha \qquad C_{Z_q}(\alpha) = -C_{D_q}\sin\alpha + C_{L_q}\cos\alpha \qquad C_{Z_{\delta_e}}(\alpha) = -C_{D_{\delta_e}}\sin\alpha + C_{L_{\delta_e}}\cos\alpha$$

The aerodynamic parameters used in this mathematical model were taken directly from the system identification results reported by Edwards (2014), who performed an extensive experimental characterization of the V-bat, including wind-tunnel tests and flight-data regression. These parameters were selected because they represent one of the most complete and publicly available datasets for a tail-sitter configuration. Additionally, several subsequent studies on hybrid UAVs have adopted or validated similar aerodynamic trends for vehicles of comparable geometry and operational regimes, confirming the consistency of the coefficients used (e.g., nonlinear lift–drag behavior with angle of attack, control-surface effectiveness, and cross-coupling moments). Aerodynamic forces are part of a system of forces that also includes the aircraft's weight and thrust, all of which

influence and interact with an aircraft's performance. Among the main aspects of this performance, stability and control are key factor for fulfilling the aircraft's assigned mission profile. For this reason, the coefficients of aerodynamic forces are included in the equations mentioned above. The equations presented in this document, in general terms, are mentioned in specialized literature on aircraft flight dynamics, such as Lipera (2000, 2001) and Beard & Maclain (2012).

Since this work focuses on evaluating the performance of the proposed identification and control strategy rather than refining aerodynamic modeling, the Edwards model provides a reliable and experimentally supported baseline. This ensures that the numerical simulations reflect realistic dynamic behavior while avoiding the need for additional system identification campaigns.

4 Methodology

Based on previous work Álvarez et al. (2019a) and Álvarez et al. (2019b), it has been shown that the Morlet wavelet provides superior performance in both system identification and adaptive PIF control. In those studies, an online wavelet-based identification was successfully integrated with a self-tuning PID structure.

4.1 Identification and Control Scheme

In this work, the identification and intelligent control of the tail-sitter are carried out following the architecture shown in **Error! Reference source not found.**.

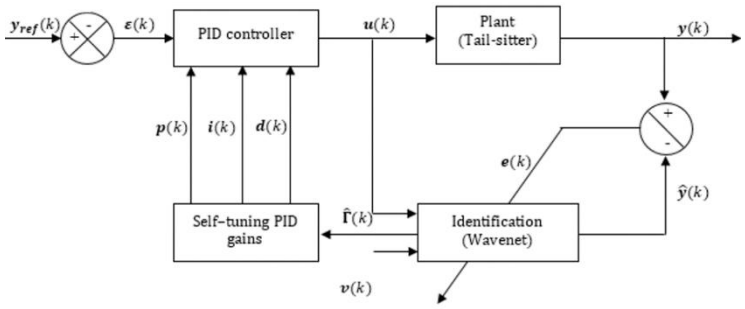

Fig. 2. Block diagram of the proposed system showing the WNN identification module, the adaptive PID controller, and the feedback used to update gains in real time.

The methodology is structured in three stages: inverse model identification, PID control design, and online self-tuning of feedback gains. These are detailed in the subsections below.

4.2 System Identification Using WNN

The wavelet neural network architecture (WNN), shown in Fig. 3, approximates the desired output signal $y(t)$ by constructing a linear combination of daughter wavelets $h_{a,b}(t)$, which are generated from a mother wavelet $h(t)$ by a dilation $a$and a translation b:

$$h_{a,b}(t) = h\left(\frac{t-b}{a}\right) \tag{19}$$

with the translation and dilatation factors $a > 0$, $b \in R$, in that order.

The network output, denoted by $\hat{y}_l(t)$, is obtained using an infinite impulse response (IIR) structure:

$$\hat{y}_l(k) = \sum_{q=1}^{p}\sum_{l=1}^{M} c_{i,l}\, z_i(k-l)u_q(k) + \sum_{j=1}^{N} d_{i,j}\, \hat{y}_l(k-j)v(k) \tag{20}$$

Where the internal signal is defined as:

$$z_i(k) = \sum_{l=1}^{L} w_{i,l}\, \psi_l(k) \tag{21}$$

Here, L is the number of daughter wavelets, $w_{i,l}$ are neuron weights, $c_{i,j}$ and $d_{i,j}$ are the forward and backward coefficients of the IIR filter, respectively. The network is trained using a least-mean-square (LMS) algorithm to minimize a convex cost function E, defined by:

$$E = \left[E_1 E_2 \dots E_i \dots E_p\right]^T \tag{22}$$

For a specific output i, E_i is defined as:

$$E_i = \frac{1}{2}\sum_{k=1}^{T} e_i^2\,(k) \tag{23}$$

Where the estimation error $e_i(k)$ is the difference between the IIR filter output signal $\widehat{y_i(k)}$ and the actual system output signal $y_i(k)$.

$$e_i(k) = y_i(k) - \hat{y}_i(k) \tag{24}$$

The objective is to minimize $E(w_k, a_k, b_k)$ by adjusting the network parameters w_k, a_k, b_k, c_k and d_k, where $k = 1,2,\dots,K$. This involves computing the gradients:

$$\frac{\delta E}{\delta w_k}, \quad \frac{\delta E}{\delta a_k}, \quad \frac{\delta E}{\delta b_k}, \quad \frac{\delta E}{\delta c_k}, \quad \frac{\delta E}{\delta d_k} \tag{25}$$

The increments of each coefficient are the negatives of their gradients,

$$\Delta w = -\frac{\delta E}{\delta w}, \quad \Delta a = -\frac{\delta E}{\delta a}, \quad \Delta b = -\frac{\delta E}{\delta b}, \quad \Delta c = -\frac{\delta E}{\delta c}, \quad \Delta d = -\frac{\delta E}{\delta d} \tag{26}$$

Thus the coefficients w, a and b of the WNN are updated according to the rules

$$w(t+1) = w(t) + \mu w \Delta w \tag{27}$$
$$a(t+1) = a(t) + \mu a \Delta a \tag{28}$$
$$b(t+1) = b(t) + \mu b \Delta b \tag{29}$$
$$c(t+1) = c(t) + \mu c \Delta c \tag{30}$$
$$d(t+1) = d(t) + \mu d \Delta d \tag{31}$$

For the implementation used in this work, the WNN employs a total of three daughter wavelets (L = 3), which correspond to three active neurons in the hidden layer. This number was selected because it provided a good balance between approximation capability and computational efficiency in preliminary tests, and no formal guidelines are available in the literature regarding the optimal number of wavelets for this class of UAV models.

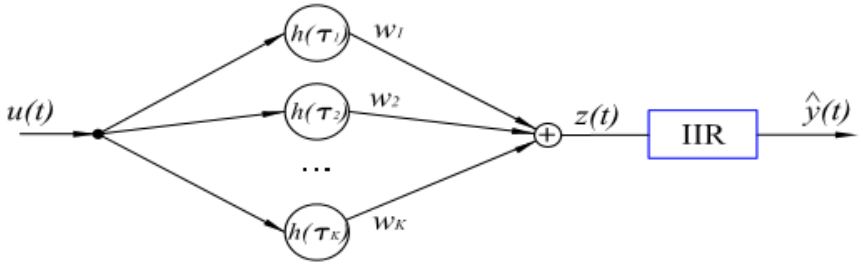

Fig. 3. Structure of the wavelet neural network with daughter wavelets and an IIR filter to represent nonlinear dynamic behavior.

Similarly, the initial values of the parameters w, a, b, c and d were assigned using random initialization, since existing studies do not provide theoretical criteria for choosing these values in WNN-based identification. The learning rates $\mu w, \mu a, \mu b, \mu c, \mu d$ were tuned empirically through trial-and-error, following common practice in adaptive wavelet networks, until stable and sufficiently fast convergence was observed. These details are included to enhance the reproducibility of the identification procedure.

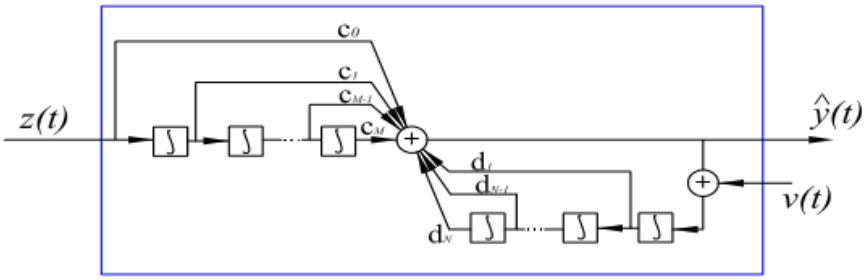

Fig. 4. Diagram of the IIR filter used inside the WNN to provide dynamic memory and capture temporal dependencies.

4.3 Adaptive control

The adaptive PID control module receives the error signal e, which is the difference between the reference signal and the system output, along with two delayed error signals $e1$, $e–2$ and finally the identification error signal ei. The control law is expressed as:

$$u = u + 1 + \big(kp * (e - e - 1)\big) + (ki * e) + \big(kd * (e - 2 * e - 1 + e - 2)\big) \quad (32)$$

To ensure the controller remains adaptive under varying conditions, a persistent signal Γ is included to update the gains. The self-tuning rules are:

$$kp = kp + \mu kp * (e - e - 1) \quad (33)$$
$$ki = ki + \mu ki * e \quad (34)$$
$$kd = kd + \mu kd * (e - 2 * e - 1 + e - 2) \quad (35)$$

Here, μkp, μki and μkd are the PID controller learning rates, which are distinct from the neural network learning rates used in the identification module.

Given the highly coupled and rapidly varying dynamics of the tail-sitter UAV—particularly during transition between vertical and horizontal flight—the initial PID gains must be chosen to guarantee at least marginal stability at startup. For this reason, the gains kp, ki and kd are initialized with any set of values that ensures initial closed-loop stability, since inappropriate starting gains can drive the system to instability within very short time intervals.

Because the literature provides no formal guidelines for selecting initial gains in adaptive PID schemes for tail-sitters, these values are chosen empirically, ensuring that control signals remain bounded and that the aircraft maintains stable attitude in hover. The learning rates μkp, μki and μkd re also tuned by trial-and-error, with the constraint that their magnitude must be small enough to prevent destabilizing gain variations during the transient response.

This initialization strategy minimizes undesirable overshoot or oscillations in the early stages of adaptation, while still allowing the controller to converge toward more effective gain values as the WNN-based identification improves its estimation of the system dynamics.

5 Results

Initial stages of this work focused on simulating the nonlinear model of the V-bat aircraft and implementing a wavelet neural network for offline identification. Then, a Morlet wavelet was adopted, and the first version of the controller was implemented.

This section presents the simulation results of the proposed identification and adaptive control scheme applied to the tail-sitter UAV. We consider the three phases of operation simulated using the nonlinear model developed before.

5.1 Online identification

Figures 5 to 8 illustrate the online identification of position, linear velocity, angular velocity and angular position, respectively.

Figure 5 shows the online identification of the inertial position components Pn, Pe, and Pd, corresponding to north, east, and down displacements, during simulated flight. The plots compare the ground truth position signals with the neural network estimates (NN). The results demonstrate excellent agreement in the north and down directions, while a small transient error is observed in the east direction around t = 6 seconds, possibly due to dynamic excitation or a brief loss of model generalization. Overall, the neural network successfully tracks the position trajectories across all axes.

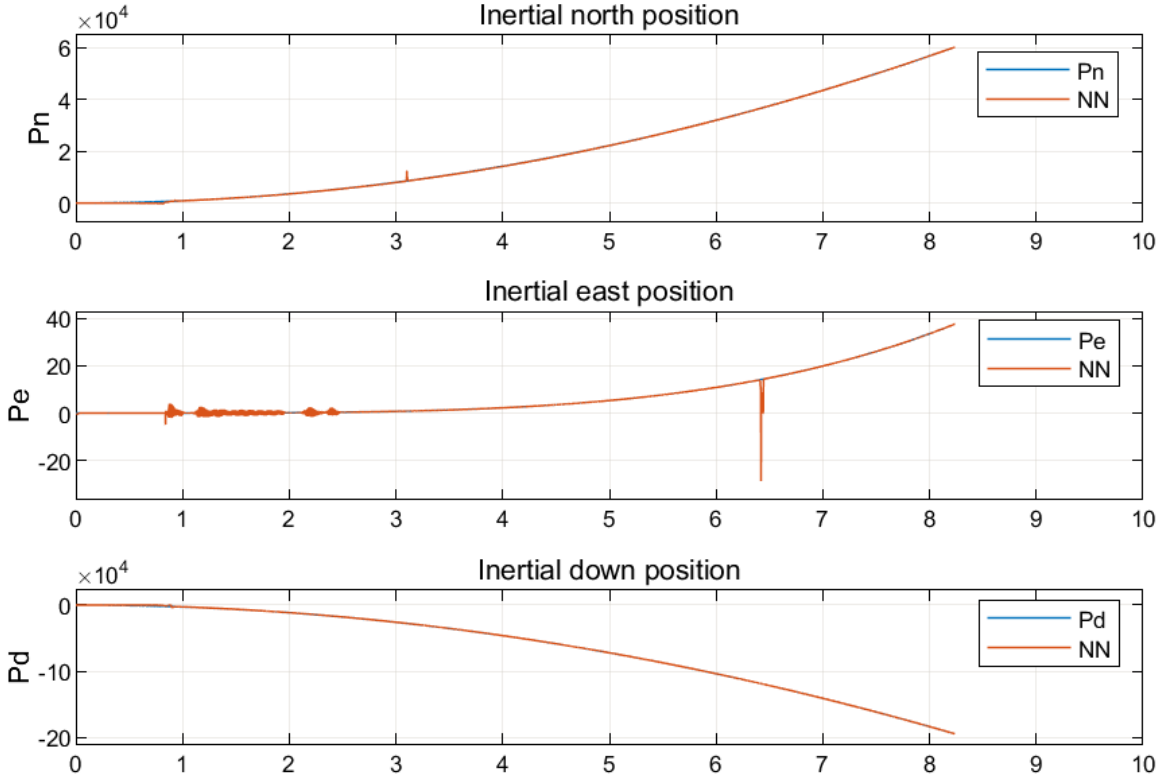

Fig. 5. Comparison between true and estimated inertial positions; the WNN tracks the motion with small transient errors.

Figure 6 presents the online identification of the linear velocity components u, v, and w, corresponding to motion along the body-frame x, y, and z axes, respectively. The neural network estimates closely match the true velocities across all channels, despite brief transients in u and v below and near t = 1 second. The network captures both the general trends and dynamic curvature of the trajectories, particularly the peak in u near t = 5 seconds and the steady descent in w. These results confirm the network's ability to track time-varying linear motion under realistic nonlinear dynamics.

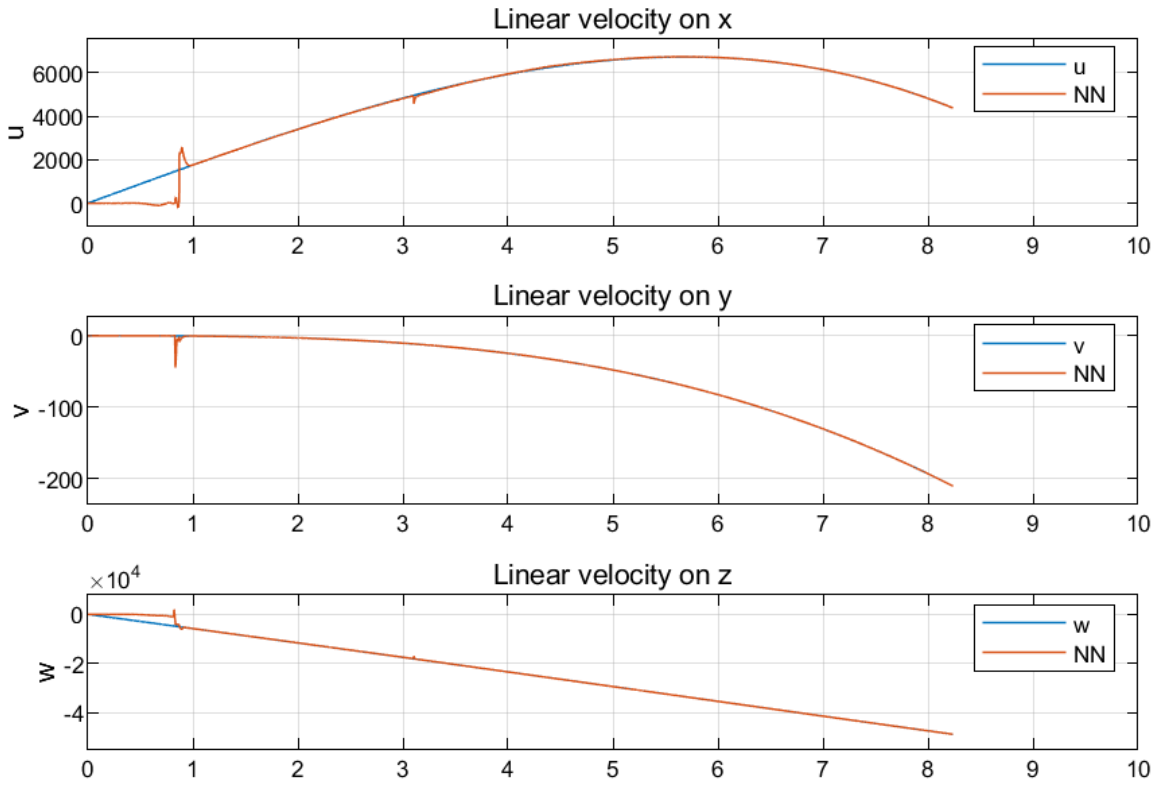

Fig. 6. Estimated body-frame velocities showing good tracking after initial adaptation transients.

Figure 7 displays the online identification of angular velocity components p, q, and r, corresponding to rotations about the body-frame x, y, and z axes, respectively. The wavelet neural network effectively tracks rapid changes in angular velocity during critical transition periods, particularly near t = 1, t=2 and t = 8 seconds. Despite brief transients, the estimated trajectories closely follow the true dynamics. These results highlight the network's capacity to capture high-frequency rotational behavior, essential for accurate modeling of tail-sitter attitude dynamics.

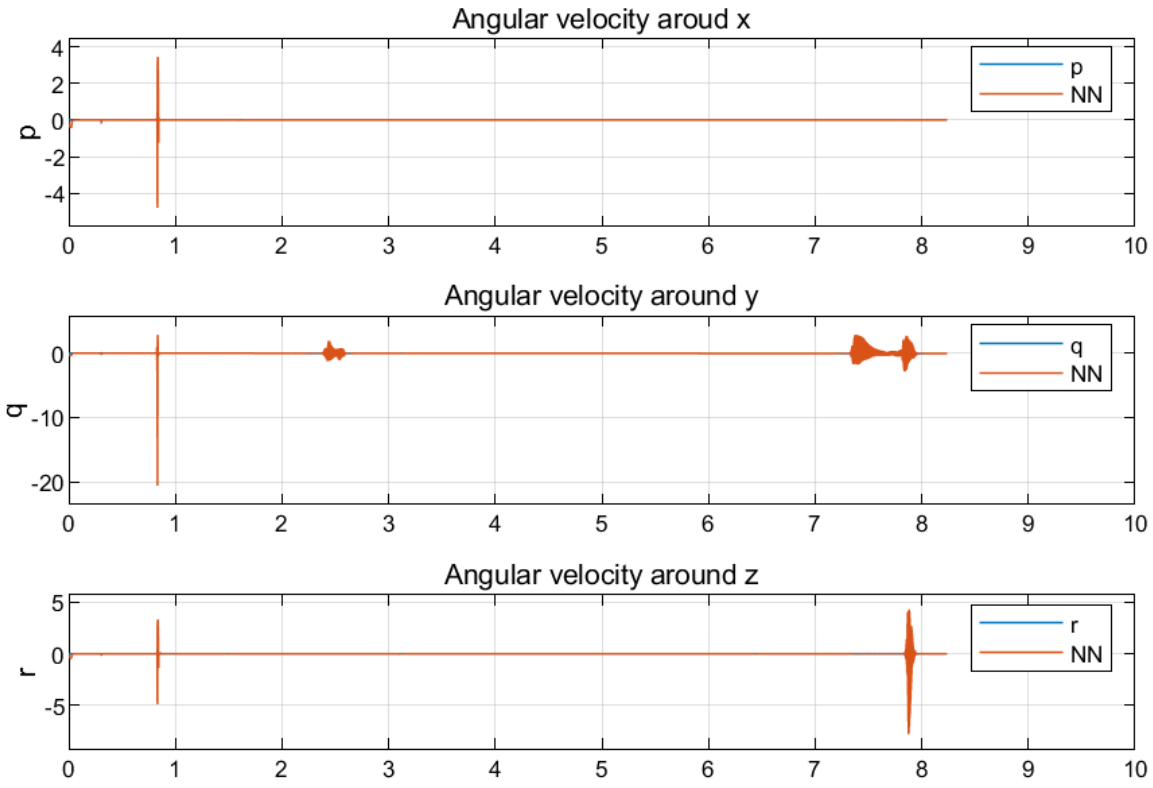

Fig. 7. WNN estimation of rotational rates with brief transients followed by accurate tracking.

Figure 8 illustrates the online identification of angular positions ϕ, θ, and ψ, corresponding to roll, pitch, and yaw angles about the body-frame x, y, and z axes, respectively. The neural network exhibits transient estimation errors before and up to t = 1 second, especially for the roll and pitch angles. Similar deviations are observed in yaw near t=8 seconds. Despite brief transients, the estimated trajectories closely follow the true dynamics. These results confirm the network's ability to estimate nonlinear attitude dynamics, with space for improved robustness.

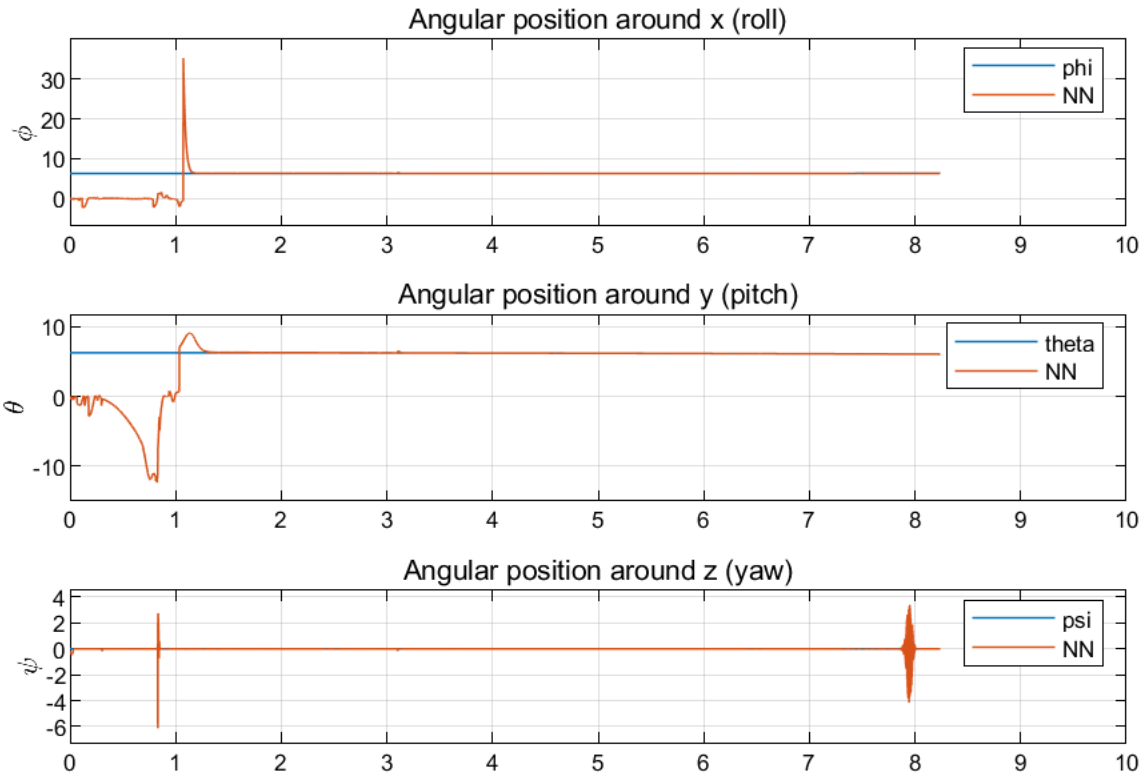

Fig. 8. Estimated roll, pitch, and yaw angles; early spikes reduce as the network stabilizes.

Once the neural network demonstrates sufficient identification accuracy, the control signals are activated.

5.2 Identification and intelligent control

This study focuses on identification and control during three phases of flight: take-off, cruise and landing phases. Transitions between these phases are highly unstable and are left for future work. The control signals are applied through the actuator inputs $\delta_a, \delta_e, \delta_t$ and δ_r. For each input, a PID controller is implemented with initial gains chosen empirically based on system response. Once active, the program adjusts the gains adaptively.

5.3 Take-off phase

The take-off simulation begins with the aircraft positioned on the ground, tilted vertically. The initial conditions are shown in Table 1.

Table 1. Initial conditions for identification and control of the tail-sitter during take-off phase.

Initial conditions of the tail-sitter during take-off phase											
P_n	P_e	P_d	u	v	w	p	q	r	ϕ	θ	ψ
0	0	0	0	0	0	0	0	0	0	$\frac{\pi}{2}$	0

The initial PID gains for each control signal are provided in Table 2.

Table 2. Initial gains of the PID controller for take-off phase.

Initial gains of the PID controller for take-off phase.

Initial gains of the PID controller for take-off phase.

Gain	K_p	K_i	K_d
Control Signal u_1	65	-9	-9
Control Signal u_2	1	0.1	0.1
Control Signal u_3	2	0.5	0.1
Control Signal u_4	0.1	0.1	0.1

Figure 9 shows the position control performance during the take-off phase, including the neural network identification of the inertial positions Pn, Pe, and Pd along the north, east, and down axes, respectively. Although the reference positions remain fixed at zero throughout the simulation, the neural network initially overshoots in all three axes, reaching peak values near 200 units before returning and stabilizing near zero. These oscillations indicate that the controller is attempting to track a stationary target but is influenced by neural network transients and possible noise in the adaptation process. The repeated spike near $t = 10^{-6}$ seconds further highlights a numerical instability that should be mitigated in future tuning or filtering stages.

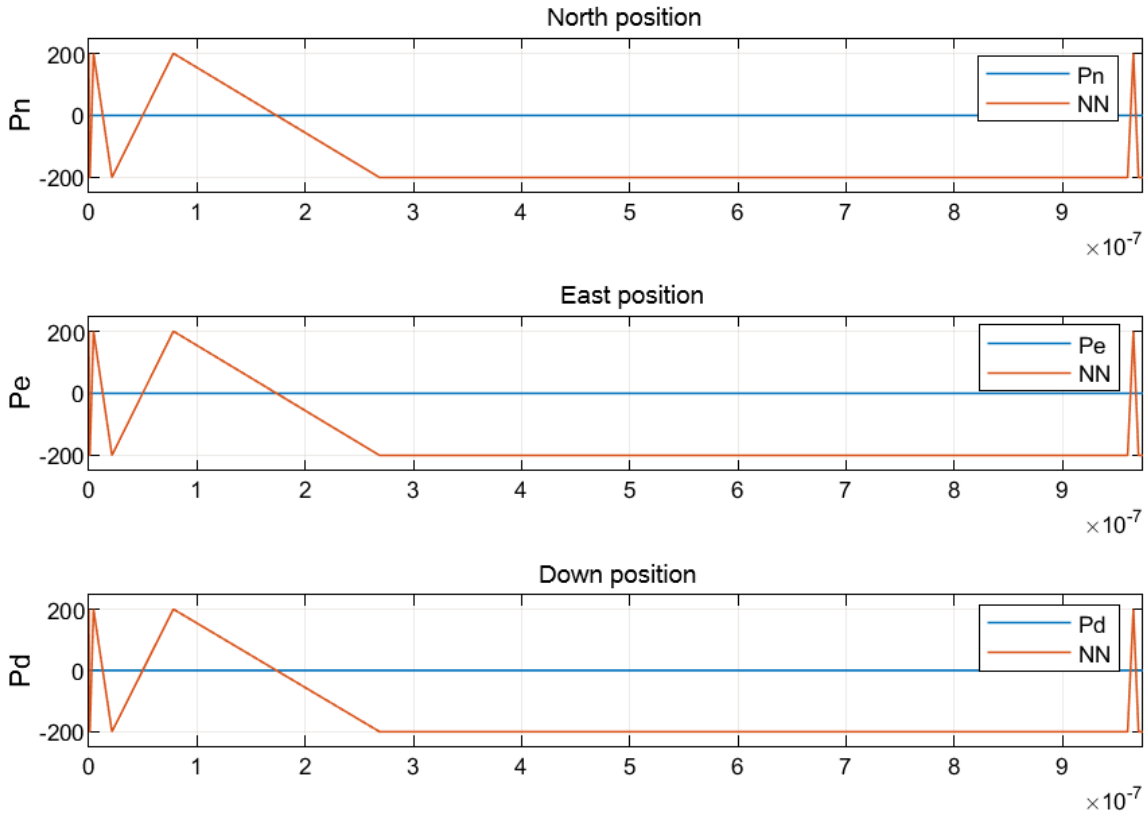

Fig. 9. Tracking of inertial position during take-off; initial oscillations decrease as the controller adapts.

Figure 10 presents the velocity identification and control results for the take-off phase, specifically for the body-frame linear velocities u, v, and w along the x, y, and z axes. The desired velocities remain constant at zero, while the neural network estimates exhibit pronounced transient excursions reaching approximately ± 200 units in each axis. These initial overshoots likely stem from the network's attempt to adapt its parameters in real time and are not reflective of the physical model, which remains at rest. The sharp spike observed near $t = 10^{-6}$ seconds suggests either numerical error accumulation or insufficient regularization in the identification process.

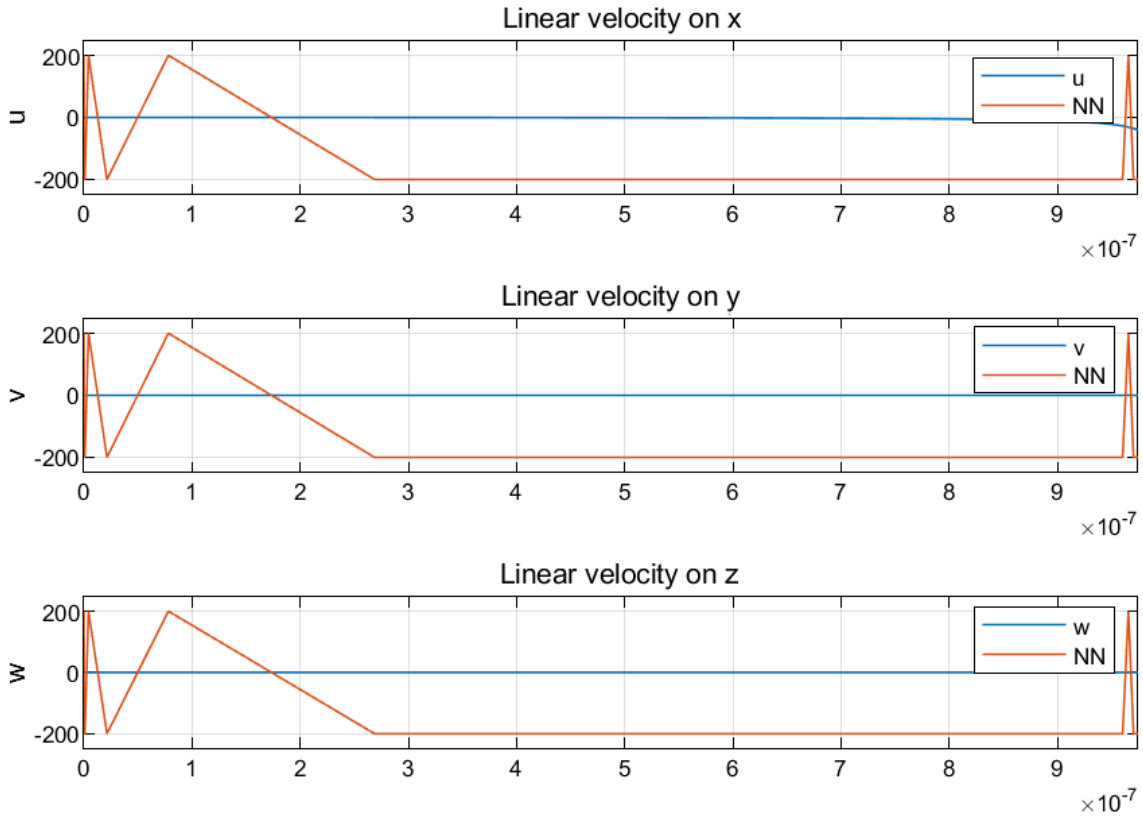

Fig. 10. Linear velocity estimates during take-off showing large initial deviations and later stabilization.

Figure 11 displays the identification and control of the angular velocities p, q, and r around the body axes x, y, and z, respectively, during the take-off phase. The desired angular velocities are fixed at zero, as the platform remains nominally static. Nevertheless, the neural network output exhibits sharp transients and oscillatory artifacts peaking around ± 200 units shortly after initialization and again near $t = 10^{-6}$ seconds. These fluctuations are identification errors from the neural network, not present in the true model, and suggest limited generalization during the learning stage.

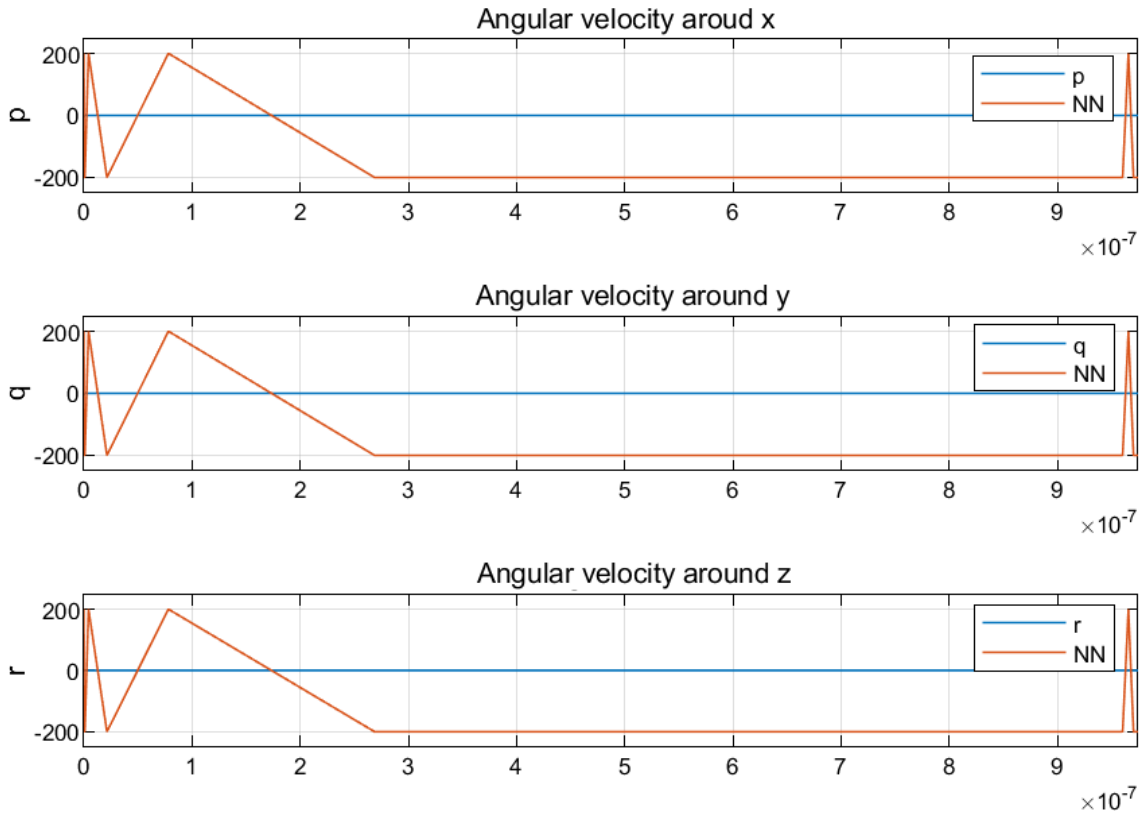

Fig. 11. Rotational velocity tracking with strong initial transients and improved mid-interval behavior.

5.4 Cruise flight phase

Cruise simulation is conducted with the UAV in forward flight at a speed of $7.2m/s$, with an angle of attack α of $1,386°$. Initial conditions are listed in Table 3.

Table 3. Initial conditions for tail-sitter during cruise flight

Initial conditions for tail-sitter during cruise flight											
P_n	P_e	P_d	u	v	w	p	q	r	ϕ	θ	ψ
0	0	10	7.2	0	0	0	0	0	0	0	0

The initial PID gains for cruise are shown in Table 4.

Table 4. Initial gains of the PID controller for cruise flight phase.

Initial gains of the PID controller for cruise flight phase			
Gain	K_p	K_i	K_d
Control Signal u_1	0.1	0.01	0.12
Control Signal u_2	0.0001	0.0000001	0.00001
Control Signal u_3	0.5	0.05	0.15
Control Signal u_4	0.0001	0.0000001	0.000001

Figure 12 illustrates the identification and control of the inertial positions Pn, Pe, and Pd along the x, y, and z axes, respectively, during the cruise phase. The reference trajectory shows a steady increase along all three axes, with smooth motion particularly evident in Pn. The neural network output follows the desired position trends but displays transient overshoots and small-amplitude oscillations, especially near t = 0 and around t = 3 seconds. These deviations reflect identification inaccuracies from the neural model, although the global tracking remains acceptable. Notably, the east and down axes show more pronounced artifacts, particularly sharp spikes, indicating greater sensitivity to learning error along those directions.

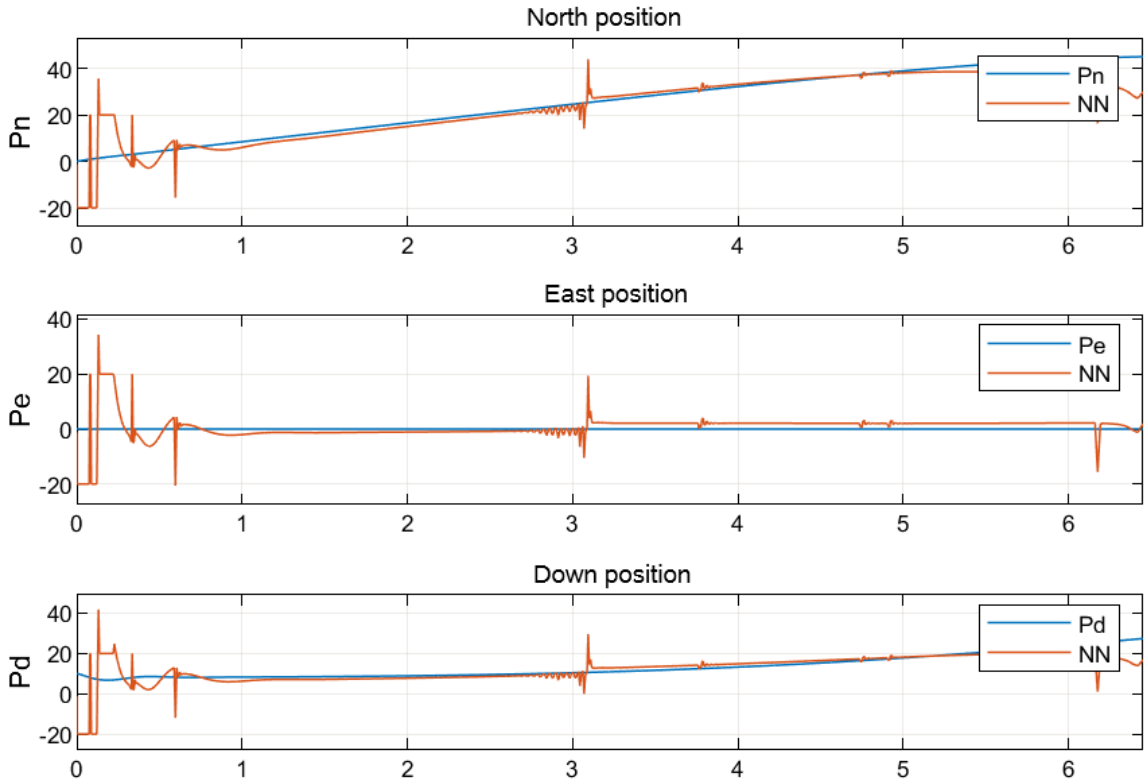

Fig. 12. Cruise-phase position tracking with smooth trends and small oscillations in some axes.

Figure 13 shows the identification and control of the linear velocity components u, v, and w along the body-frame axes x, y, and z, respectively, during cruise flight. The neural network estimates exhibit good overall agreement with the model dynamics after the transient stage. However, in the initial seconds, strong overshoots and oscillations are visible, particularly in the u and v components, which indicate instability in the learning process. Small-amplitude ripples persist throughout the trajectory and there are brief disturbances around t = 3 seconds and near the end of the time window. Despite these artifacts, the NN output stabilizes and tracks the steady-state behavior of the simulated model reasonably well.

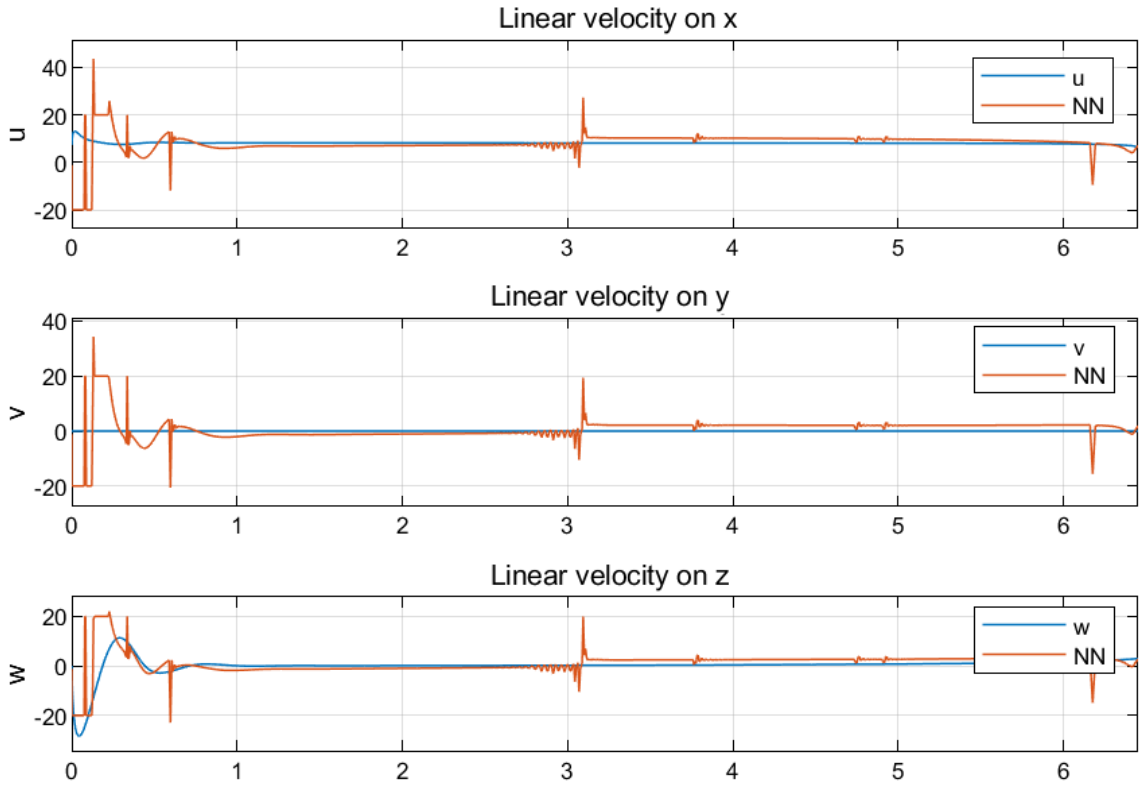

Fig. 13. Linear velocity estimation in cruise; the WNN converges after initial overshoots.

Figure 14 shows the identification and control of the angular velocity components p, q, and r about the body-frame axes x, y, and z during the cruise segment. The reference signals remain smooth throughout, whereas the neural network (NN) outputs exhibit high-frequency transients at the onset and brief disturbances near t = 3 seconds and the simulation endpoint. These perturbations reflect numerical instability in the online identification process rather than true angular dynamics, which are assumed constant. Despite these fluctuations, the NN recovers the steady-state behavior in all three channels and tracks the angular rates with low error outside the transient zones.

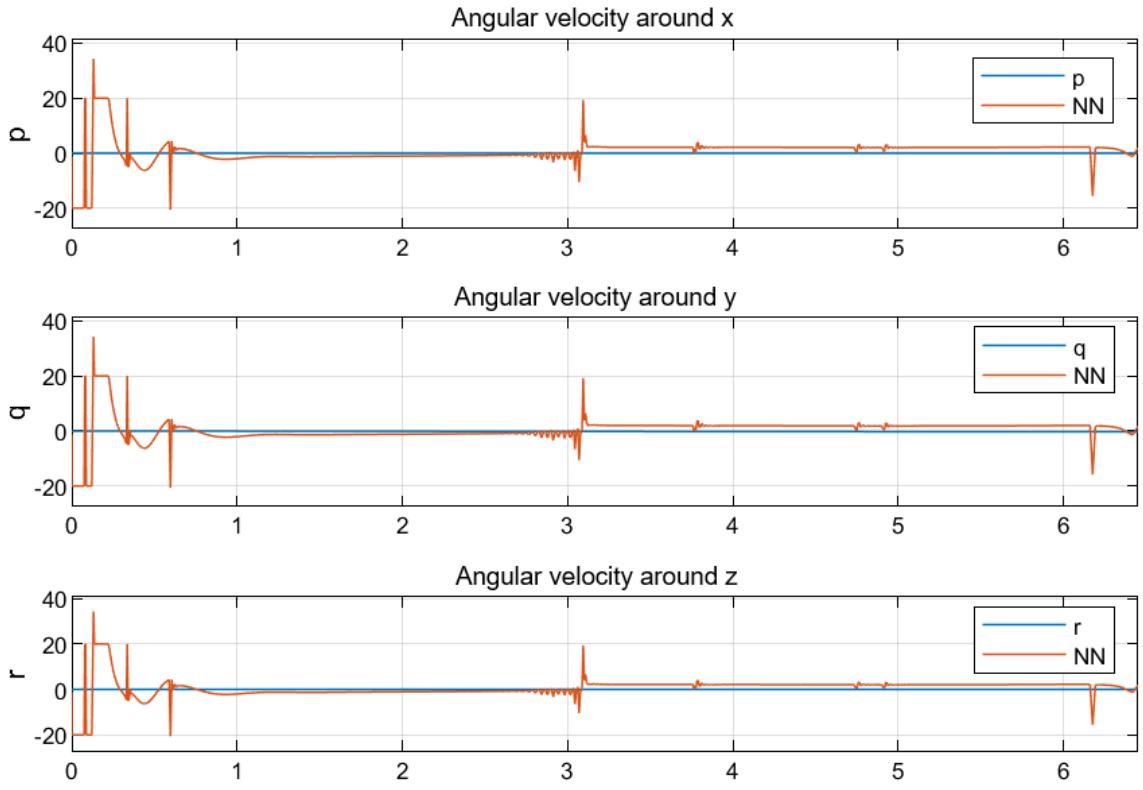

Fig. 14. Rotational velocity tracking during steady flight with minor disturbances.

Figure 15 presents the identification and control of the angular positions ϕ, θ, and ψ around the body axes x, y, and z, respectively. As with previous cases, the neural network (NN) output exhibits transients and abrupt errors during the initial seconds and at discrete instants near t = 3 and t = 6 seconds. These deviations do not correspond to actual changes in the angular states, which remain nearly constant throughout the simulation. The errors arise from numerical instability in the online estimation. Outside these transient regions, the NN tracks the reference signals with low steady-state error, recovering the correct roll, pitch, and yaw values during cruise.

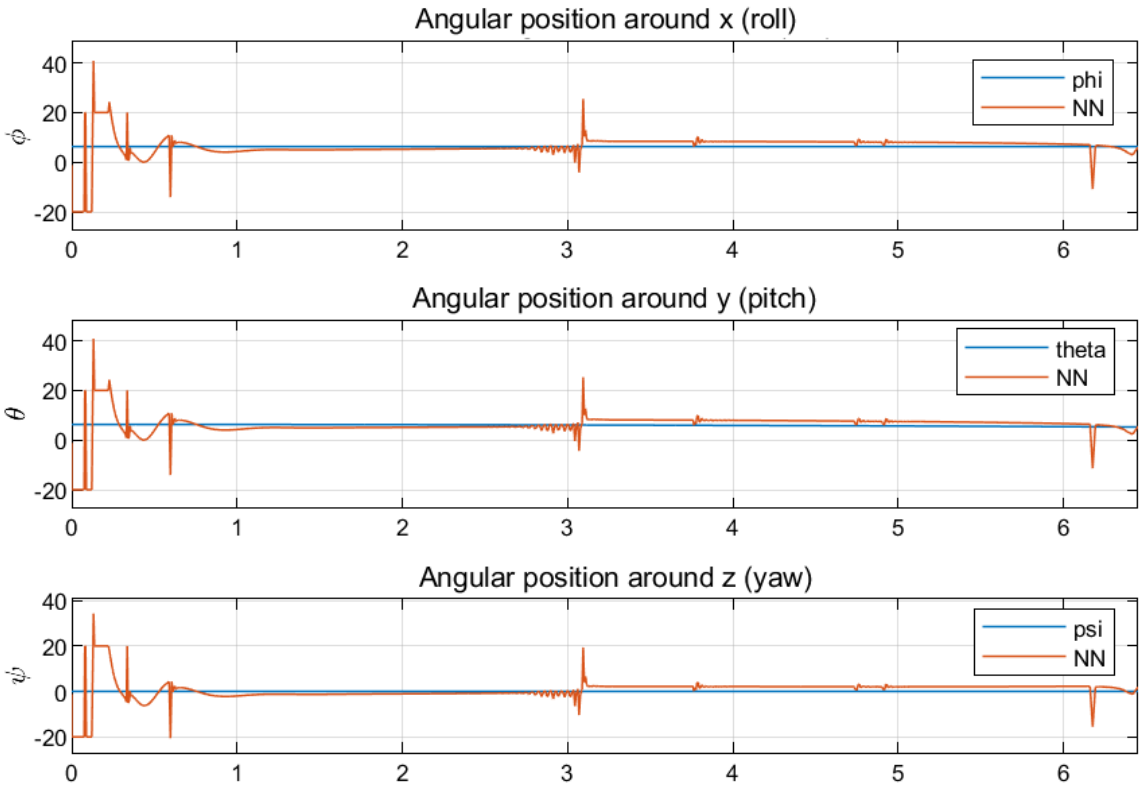

Fig. 15. Attitude tracking (roll, pitch, yaw) with brief deviations but accurate steady-state behavior.

5.5 Landing phase

In the landing phase, the UAV begins at an altitude of $30m$ and inclined at 90^{o} mimicking the take-off process in reverse. Initial conditions are shown in Table 5.

Table 5. Initial conditions for tail-sitter during landing phase

Initial conditions for tail-sitter during landing phase											
P_n	P_e	P_d	u	v	w	p	q	r	ϕ	θ	ψ
30	0	0	5	0	0	0	0	0	0	$\frac{\pi}{2}$	0

Initial PID gains are shown in Table 6.

Table 6. Initial gains of the PID controller for landing phase.

Initial gains of the PID controller for landing phase			
Gain	K_p	K_i	K_d
Control Signal u_1	-3.5	-0.01	-0.001
Control Signal u_2	1	0.1	0.1
Control Signal u_3	1	0.09	1
Control Signal u_4	0.1	0.1	0.1

Figure 16 presents the identification and control of the position along the North Pn, East Pe, and Down Pd directions. In all three components, the neural network (NN) output initially overshoots the reference position at early times, then approaches the reference more closely during mid-time intervals and again deviates with a final overshoot near the end of the time horizon. This behavior highlights a characteristic transient mismatch at the beginning and end of the trajectory, while the NN achieves better accuracy in the central portion.

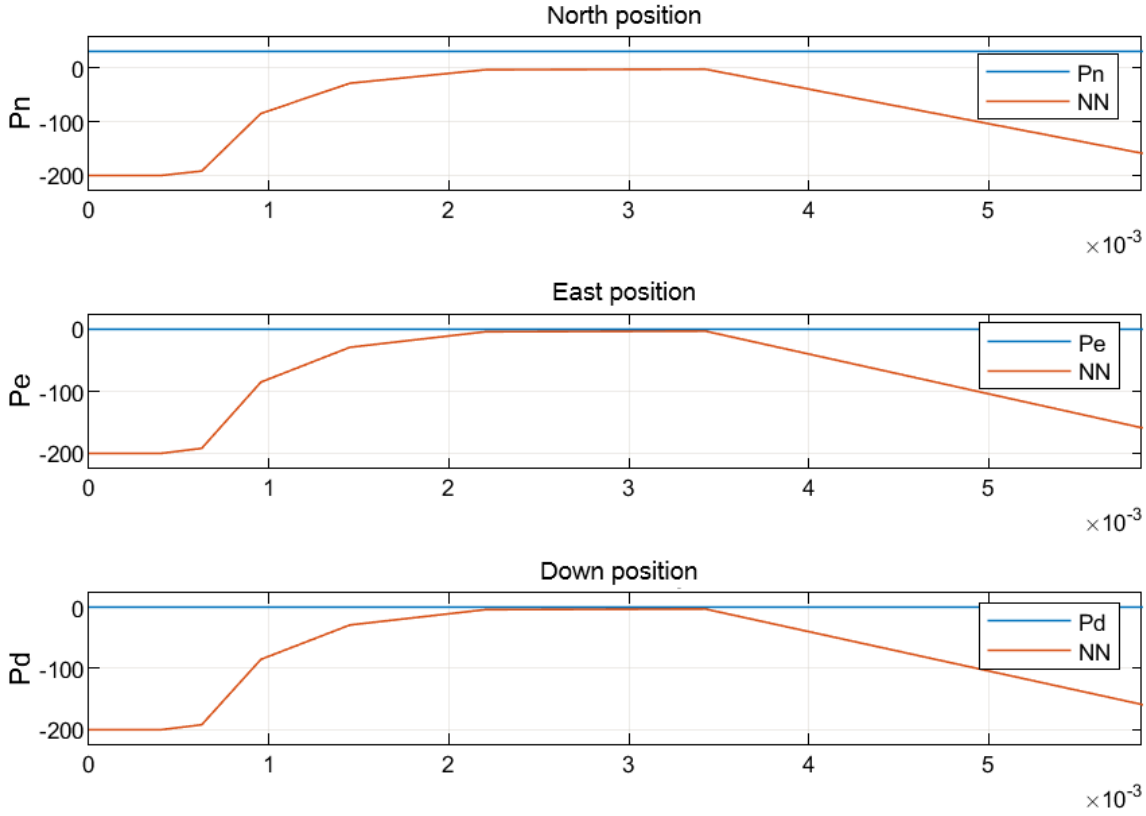

Fig. 16. Landing-phase position estimates with typical overshoots at start and end of the maneuver.

Figure 17 shows the identification and control of the linear velocity components u, v, and w along the x, y, and z axes, respectively. In all three plots, the neural network estimate exhibits an initial undershoot followed by a fast rise that overshoots the reference velocity. The approximation then stabilizes around the correct value during the middle segment of the time range, before deviating again at the end. This pattern is consistent across all axes and reflects a transient response with improved accuracy near the center of the time window and systematic errors near the boundaries.

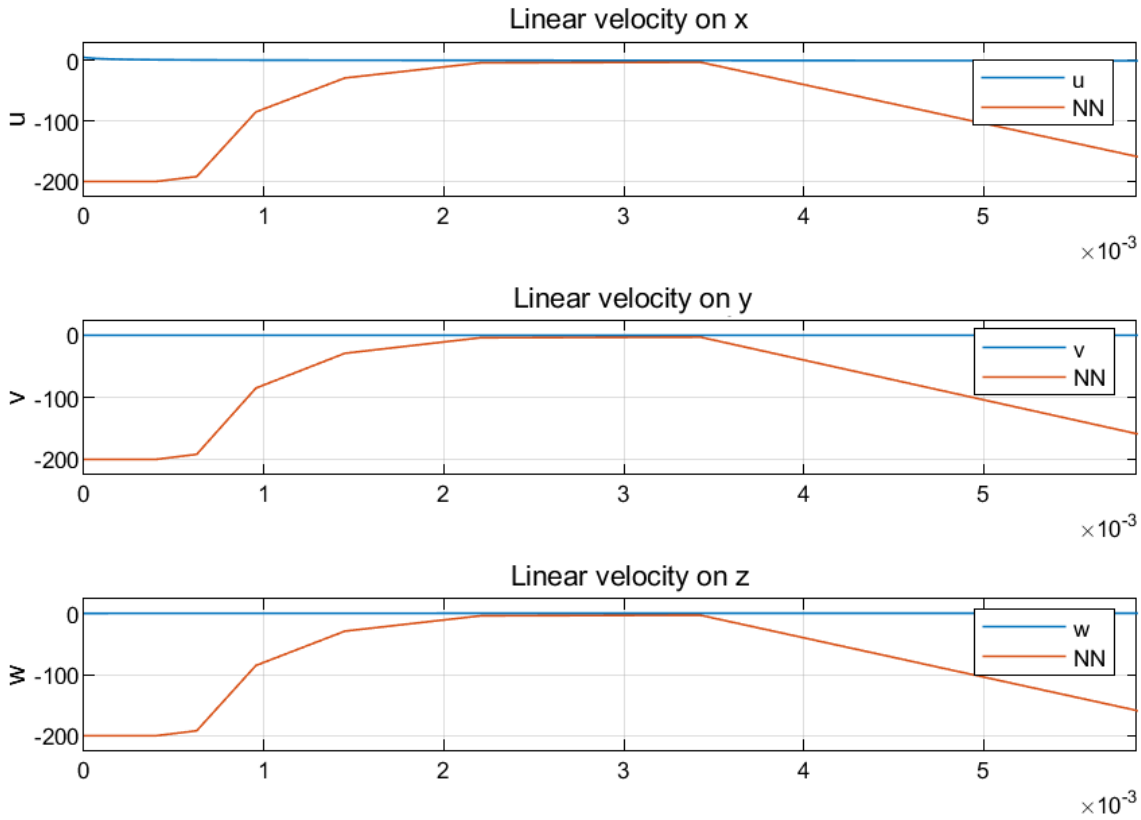

Fig. 17. Linear velocity tracking during descent; better performance in the mid-time region.

Figure 18 illustrates the identification and control of the angular velocity components p, q, and r around the x, y, and z axes, respectively. The neural network prediction displays significant undershoot at the start of the interval, followed by a closer approximation to the reference values during the middle portion of the time window. Toward the end, the approximation again departs from the reference with increasing error. This pattern is consistent across all three rotational axes, highlighting the neural network limited performance at boundary times and relatively improved alignment during steady-state behavior near the center.

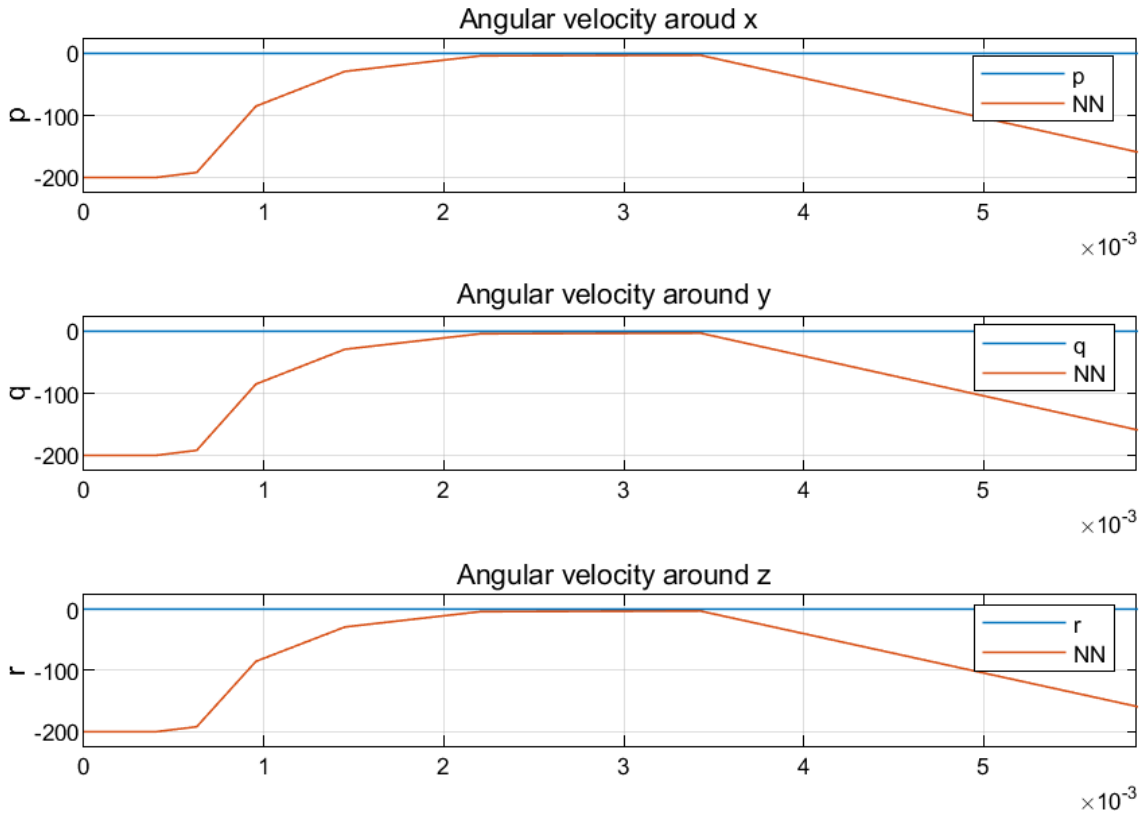

Fig. 18. Rotational velocity estimation showing boundary transients and stable mid-interval accuracy.

Table 7. Consolidated quantitative metrics for position and attitude tracking.

RMS Error for Figures 8, 9, 15, and 16

Figure	Variable	RMS
8	Roll	0.226
	Pitch	0.103
	Yaw	0.083
9	X	0.483
	Y	0.505
	Z	0.314
15	Roll	0.050
	Pitch	0.041
	Yaw	0.043
16	X	0.633
	Y	0.618
	Z	0.622

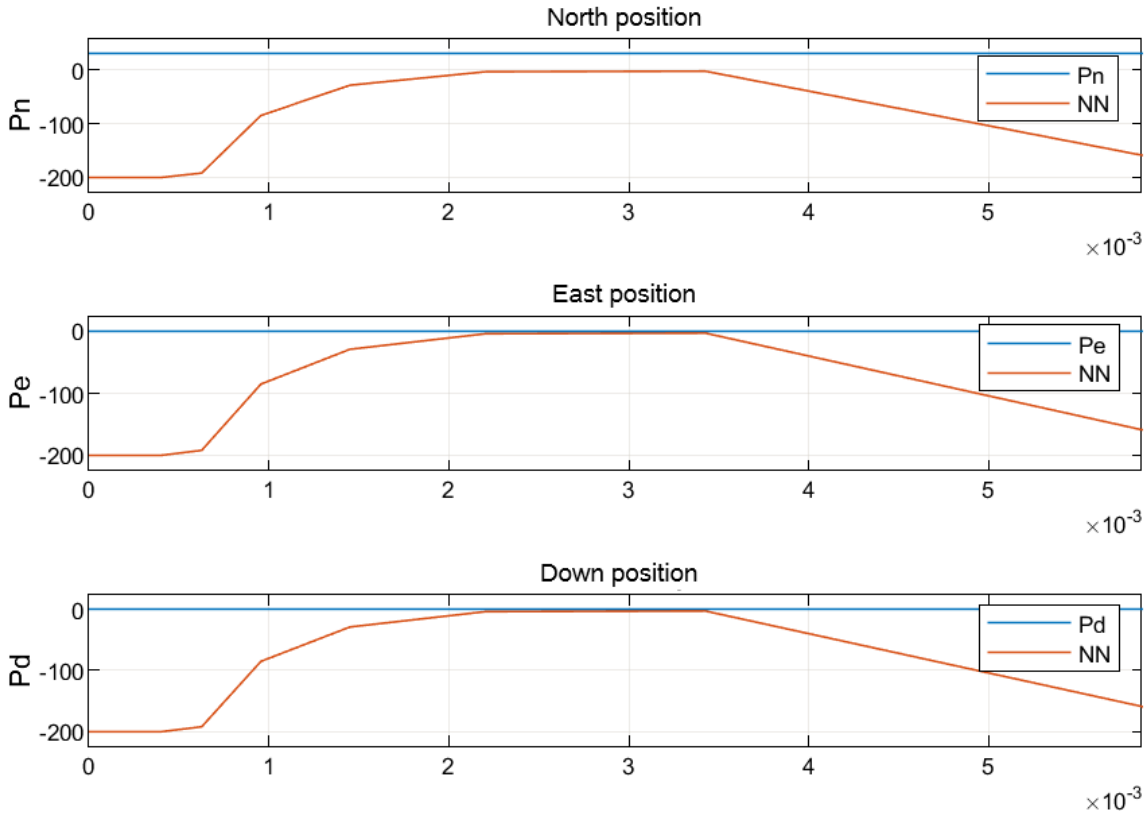

Fig. 19. Roll, pitch, and yaw tracking during landing with correct steady-state behavior.

To complement the qualitative analysis and provide an objective evaluation of the controller's performance, quantitative metrics were estimated from the curves shown in Figures 8, 9, 15, and 16. The reference and neural-network output signals were digitally extracted from the plots. The evaluated metrics RMS error, cover both position and attitude variables. The resulting values, summarized in Table 7, show that the neural network is able to track the trajectories adequately, with moderate RMS errors and consistent settling times across the different scenarios. These quantitative results reinforce the validity of the proposed controller and provide additional evidence of its performance under dynamic operating conditions.

The results show noticeable peaks and oscillations in several states—particularly during take-off and landing—and in some phases the transient errors persist for relatively long intervals. This behavior is mainly due to the simultaneous adaptation of the WNN-based identifier and the PID gains under highly nonlinear operating conditions. Because the IIR structure inside the WNN provides dynamic memory, early estimation errors may propagate for several time steps before decaying, which contributes to the persistence of transient deviations. In addition, the learning rates used in the online updates directly influence the speed and aggressiveness of parameter adaptation; higher values accelerate convergence but can amplify oscillations, while lower values improve stability but slow down compensation during abrupt maneuvers. The combination of these factors explains why the transients are more severe in phases with rapid attitude changes, such as take-off and landing.

Several strategies could be implemented to mitigate these effects. Low-pass filtering of the identification and control errors can reduce the impact of high-frequency fluctuations entering the IIR recursion. Slower adaptation rates at the beginning of each phase would allow the WNN to stabilize before applying more aggressive updates. Regularization mechanisms—such as bounding parameter updates or introducing decay terms—may also reduce the risk of persistent transients. These modifications can improve robustness while preserving the benefits of online learning.

6 Conclusions

This section presents the main conclusions of the study based on the results presented in the previous section. It is divided into two parts: the online identification of the tail-sitter UAV, and its intelligent identification and control.

6.1 Online identification

The online identification procedure was applied to the V-bat tail-sitter model using a neural network with Morlet wavelet activation functions, selected based on prior identification results. Compared to the T-wing platform, the V-bat demonstrated improved performance due to more accurate aerodynamic parameters obtained experimentally. The neural network output closely followed the true system behavior with minimal disturbances. Notably, the real signals remained stable even in the absence of control inputs.

These results support the conclusion that neural networks are effective tools for identifying systems with highly non-linear and inherently unstable dynamics. While challenges remain in tracking non-periodic or discontinuous signals, the use of IIR filters enables good approximations when the underlying dynamics are smooth or feature isolated abrupt transitions. Under these conditions, residual identification errors remain small.

However, the proposed identification approach still presents some limitations. First, the lack of formal guidelines for selecting the number of wavelets and learning rates implies that training performance may depend on empirical tuning. Second, the identification accuracy deteriorates when the system experiences strong discontinuities or rapid transients, as the network may require additional iterations to adapt. These limitations suggest the need for more systematic parameter selection methods and for mechanisms that enhance robustness during fast dynamic changes.

6.2 Intelligent identification and control of the tail-sitter

The final objective of this work was the implementation of control strategies for the previously identified system. Adaptive PID controllers, as described in Section 5, were integrated with the neural network model, and the combined performance was evaluated.

The results revealed limitations in the control of velocity signals during dynamic flight phases such as take-off and landing. These limitations are attributed to the rapid response required from both the network and the controller to simultaneously identify the system and tune control gains. However, in the cruise phase, where the system dynamics are smoother and involve gradual changes, the control signal proved effective in maintaining correct operation.

We therefore conclude that adaptive control based on online neural network identification is a promising approach for non-linear and fast-response systems, provided the dynamics remain smooth and do not involve abrupt or large transitions. This was clearly observed during cruise flight, where only moderate speed variation was needed and the system responded effectively.

Despite these positive results, several limitations of the proposed adaptive control scheme were identified. The controller exhibits reduced performance during highly transient phases, partly due to the sensitivity of the PID gain updates to noisy or rapidly changing error signals. Moreover, the absence of regularization mechanisms or filtering strategies can amplify transient oscillations during adaptation.

Future improvements could include incorporating regularization terms in the adaptation laws, applying low-pass filtering to the identification and control errors, or introducing slower adaptation rates during aggressive maneuvers to ensure stability. Additionally, hybrid schemes combining model-based elements with WNN-based adaptation could further enhance performance during abrupt transitions.

6.3 Future Work

Based on the findings and limitations identified in this study, several concrete directions for future work can be pursued. One important step is the experimental validation of the proposed identification and control strategies on a real tail-sitter platform. Such tests would provide essential insights into performance under realistic conditions, including sensor noise, actuator constraints, and environmental disturbances.

Another relevant extension is the incorporation of wind perturbation and turbulence models into the identification and control framework. This addition would allow evaluating the robustness of the proposed method in outdoor flight scenarios where aerodynamic disturbances play a significant role.

A further line of research involves comparing the WNN-based identification scheme with alternative approaches such as extended Kalman filtering, recursive least squares, Gaussian-process regression, or deep-learning-based models. These methods may offer complementary advantages in accuracy, convergence speed, or computational efficiency.

Finally, improvements to the adaptive control module could be explored through the introduction of regularization mechanisms to prevent parameter drift, the inclusion of real-time filtering of control and identification errors to reduce noise sensitivity, or the use of slower adaptation rates during aggressive maneuvers to enhance stability. Collectively, these research directions would strengthen the reliability and applicability of the proposed methodology for autonomous operation of tail-sitter UAVs in real-world environments.

7 References

Álvarez González, J. C., Vega Navarrete, M. A., Ramos Velasco, L. E., Domínguez Mayorga, C. R., García Rodríguez, R., & Parra Vega, V. (2019a). Identificación inteligente de un dron convertible. *Academia Journals Hidalgo 2019*.

Álvarez González, J. C., Vega Navarrete, M. A., Ramos Velasco, L. E., Domínguez Mayorga, C. R., García Rodríguez, R., & Parra Vega, V. (2019b). Identification and intelligent control of a tail-sitter UAV. In *Proceedings of the 3rd International Conference on Aeronautics*.

Åström, K. J., & Hägglund, T. (2009). *Control PID avanzado*. Pearson Educación.

Åström, K. J., & Wittenmark, B. (2011). *Computer-controlled systems: Theory and design* (3rd ed.). Dover Publications. ISBN: 978-0486486130

Beard, R. W., & McLain, T. W. (2012). *Small unmanned aircraft: Theory and practice* (2nd ed.). Princeton, NJ: Princeton University Press.

Edwards, N. W. (2014). *System identification of an unmanned tailsitter aircraft* (Master's thesis, Brigham Young University). Brigham Young University ScholarArchive.

Lipera, L. (2000). Micro Craft ducted air vehicle.

Lipera, L., Colbourne, J. D., Tischler, M. B., Mansur, M. H., Field, M., & Rotkowitz, M. C. (2001). The Micro Craft iSTAR micro air vehicle: Control system design and testing. Annual Forum Proceedings of the American Helicopter Society, 57(2), 1998–2008. https://mcrotk.github.io/publications/ahs01.pdf

Islas Gómez, O., Ramos Velasco, L. E., García Lamont, J., Ramos Fernández, J. C., & Espejel Rivera, M. A. (2011). Implementation of different wavelets in an auto-tuning Wavenet PID controller and its application to a DC motor. In *Proceedings of the 2011 IEEE Electronics, Robotics and Automotive Mechanics Conference (CERMA 2011)* (pp. 301–306). IEEE. https://doi.org/10.1109/CERMA.2011.55

Navarrete, M. A., Ramos-Velasco, L. E., Domínguez Mayorga, C. R., Carreón, P. A., Hernández, J. V., Delgado, V. M., Parra Vega, V., & Márquez Vera, M. A. (2018). Output feedback self-tuning Wavenet control for underactuated Euler–Lagrange systems. *IFAC-PapersOnLine, 51*(13), 633–638. https://doi.org/10.1016/j.ifacol.2018.07.351

Ramos Velasco, L. E., Domínguez Ramírez, O. A., & Parra Vega, V. (2016). Wavenet fuzzy PID controller for nonlinear MIMO systems: Experimental validation on a high-end haptic robotic interface. *Applied Soft Computing, 40,* 199–205. https://doi.org/10.1016/j.asoc.2015.11.014

Saeed, A. S., Younes, A. B., Cai, C., & Cai, G. (2018). A survey of hybrid unmanned aerial vehicles. *Progress in Aerospace Sciences, 98,* 91–105. https://doi.org/10.1016/j.paerosci.2018.03.007

Spong, M. W., & Vidyasagar, M. (2008). *Robot dynamics and control* (2nd ed.). Wiley-Interscience. ISBN: 978-0471612438

Wang, X., Chen, Z., & Yuan, Z. (2015). Modeling and control of an agile tail-sitter aircraft. *Journal of the Franklin Institute, 352*(12), 5437–5472. https://doi.org/10.1016/j.jfranklin.2015.09.012