The Aeronautical Journal (2026), 1–31
doi:10.1017/aer.2025.10114

RESEARCH ARTICLE

New fuzzy-RNN autopilot system for the Cessna Citation X aircraft with a fuzzy transition algorithm

Seyed Mohammad Hosseini[1], Georges Ghazi[1] and Ruxandra Mihaela Botez[2]

[1]Laboratory of Applied Research in Active Control, Avionics and AeroServoElasticity (LARCASE), École de technologie supérieure, AIAA Member, Montréal, QC, Canada
[2]Canada Research Chair Holder Level 1 in Aircraft Modeling and Simulation, Head of the Laboratory of Applied Research in Active Controls, Avionics and AeroServoElasticity (LARCASE), École de technologie supérieure, AIAA Fellow, Montréal, QC, Canada
Corresponding author: Ruxandra Mihaela Botez; Email: ruxandra@gpa.etsmtl.ca

Received: 6 October 2024; **Revised:** 24 November 2025; **Accepted:** 24 November 2025

Keywords: artificial intelligence; autopilot control system; fuzzy logic; recurrent neural networks; sliding mode controller

Abstract

This paper presents a novel artificial intelligence-based autopilot control system designed for the Cessna Citation X (CCX) aircraft longitudinal motion during cruise. In this control methodology, the unknown aircraft dynamics in the state-space representations of each vertical speed (VS) mode and altitude hold (AH) mode were approximated by two multiplayer fuzzy recurrent neural networks (MFRNNs) trained online using a novel approach based on particle swarm optimisation and backpropagation algorithms. These MFRNNs were used with two sliding mode controllers to guarantee the robustness of both VS and AH modes. In addition, a novel fuzzy logic-based transition algorithm was proposed to efficiently switch the controller between these autopilot modes. The performance of the controllers was evaluated with a nonlinear simulation platform developed for the CCX based on data from a Level D research aircraft flight simulator certified by the FAA. The system stability and robustness were proved by the Lyapunov theorem. The simulation, tested under 925 flight conditions, demonstrated the controllers exceptional tracking capability in a variety of uncertainties, including turbulent and non-turbulent flight scenarios. In addition, the design ensured that the smoothness of the control input signals was maintained in order to preserve the mechanical integrity of the elevator actuation system.

Nomenclature

α_{damp}	inertia damping parameter in the PSO
CAS	calibrated airspeed (kts)
D	turbulence
$e_{(vs,h)}$	tracking errors (ft/s for vertical speed and ft for altitude)
$\hat{F}_{(vs,h)}$	functions approximated by the multiplayer fuzzy recurrent neural network (MFRNN)
$\tilde{F}_{(vs,h)}$	approximation error
$F_{(vs,h)}, G_{(vs,h)}$	unknown nonlinear functions
$g_{(h,vs)}$	positive constant
h	altitude (ft)
h_{ref}	desired altitude (ft)
h_{trans}	transition altitude (ft)
i_{max}	max number of iterations in the PSO
$k_{(h,vs)}$	gains of the switching control laws
$L_{(h,vs)}$	postive design parameters
$m_{(h,vs)}$	mode transition coefficients
p_{max}	swarm size in the PSO

https://doi.org/10.1017/aer.2025.10114 Published online by Cambridge University Press

$S_{(h,vs)}$	sliding surfaces
$u_{eq_{(h,vs)}}$	equivalent control laws
$u_{sw_{(h,vs)}}$	switching control laws
$u_{(h,vs)}$	control laws for the altitude hold (h) and vertical speed vs) modes (deg/sec)
u_{AT}	ultimate autopilot control law (deg/sec)
VS	vertical speed (ft/s)
VS_{ref}	desired vertical speed (ft/s)
W	weights of the MFRNN
x	aircraft state variable
$x_{(vs,h)}$	inputs of the MFRNN
X_{cg}	centre of gravity (%)
X_{lo}, X_{up}	lower/ upper bounds of the searching range
$\mu_{1m}(x),\ \mu_{2n}(x)$	Gaussian membership functions
μ_{NHE}	membership function of the negative high error (NHE) linguistic term
μ_{ME}	membership function of the medium error (ME) linguistic term
μ_{NHE}	membership function of the positive high error (PHE) linguistic term
Ω_1, Ω_2	acceleration parameters in the PSO

1.0 Introduction

Improving aircraft safety, efficiency and reliability have been the main topics of many articles in the aerospace and aeronautics fields. Although recent technological advances have helped reduce system failures, human errors remain the cause of approximately 80% of aircraft accidents and incidents [1]. While conventional control systems continue to perform adequately in flight control applications, artificial intelligence- (AI) based methodologies offer promising enhancements, such as adaptability and improved real-time control performance to autopilot systems. These approaches have the potential to handle uncertainties, unknown dynamics, external disturbances and varying flight conditions without frequent parameter tuning, which is essential for the conventional systems. In addition, they can also help to reduce the pilot's workload, particularly during high-demand tasks or complex manoeuvres. Therefore, this article introduces a new generation of AI-based autopilot systems for commercial aircraft by using a novel control strategy based on a multilayer fuzzy recurrent neural network (MFRNN) with a two-mode sliding mode control (SMC).

Previously, various conventional methodologies were applied to control unmanned aerial vehicles (UAVs) and quadcopters.

In previous studies, various conventional methodologies were applied to control UAVs and quadcopters. Ahsan and Hanif [2] presented two methods based on the proportional integral derivative (PID) method to control the altitude of an Aerosonde UAV; one with a single elevator command and the other using both throttle and elevator commands, showing that the second approach could enhance transient response. For the same application, Ahsan et al. [3] developed two altitude controllers: one based on PID control and the other using a lead compensator, and found that the lead compensator operated better than the PID.

To improve the altitude tracking performance of UAVs, Liu et al. [4] developed linear proportional (P), derivative (D), and two derivatives (DD) controllers integrated with a Kalman filter to estimate both the climb and descent rates. Cárdenas et al. [5] applied a pole-placement methodology to control the altitude and heading of an UAV by using linearised longitudinal and lateral-directional models. A back-tracking search optimisation method combined with PID control was used to enhance the autopilot performance of an unmanned helicopter by improving settling time, rise time and maximum overshoot specifications [6].

Liu et al. [7] proposed a fine-tuned PID-based airspeed, heading and altitude controllers for an UAV. Win et al. [8] developed a PID-based pitch attitude hold autopilot for an YTUEC001 UAV.

Qian and Liu [9] used an uncertainty and disturbance estimator- (UDE) based translational controller for near-hovering phases and an inner-loop attitude controller, ensuring global asymptotic stability under disturbance and payload.

Qi et al. [10] showed that using a modified UDE within a cascade PID architecture can ensure accurate attitude tracking of a quadrotor while effectively rejecting disturbances. Similarly, Feng and Liu [11] proposed a nonlinear model predictive control (MPC) strategy designed to prevent collisions between two UAVs around obstacles using a new flight scenario. Zhao et al. [12] applied global fast terminal SMC for controlling the attitude of an UAV to approach a virtual tanker aircraft for an aerial refueling. As another approach for collision avoidance, Zhang et al. [13] developed an autopilot system based on a sliding mode observer (SMO) combined with an energy-efficient trajectory planning strategy for quadrotors, aimed at ensuring the safety of the actuation system.

Zareb et al. [14] suggested that genetic algorithms (GAs) can be used to optimise the performance of two fuzzy controllers and four PIDs for controlling the attitude and altitude of an AR Drone V2 quadrotor. Wilburn et al. [15] also developed a new GA to optimise controller gains to guarantee and ensure effective trajectory tracking performance for UAVs. Zhou et al. [16] proposed a control strategy combining a finite-time SMC system with disturbance observers and an event-triggered communication mechanism. This approach aims to improve the trajectory tracking performance of UAVs while reducing communications load. In another study, the adaptive neuro-fuzzy inference system (ANFIS) was integrated with the brain storm optimisation (BSO) algorithm to improve UAV thrust performance. This hybrid approach effectively optimised parameters related to the propeller, battery and engine [17].

Besides the design of autopilot controllers for UAVs and quadrotors, different methodologies were developed for hypersonic and supersonic aircraft, as discussed below.

Sahbon et al. [18] proposed a Thrust Vector Control (TVC)-based path-following autopilot system for the X-15 aircraft using three PID controllers for roll, altitude and yaw control to compensate for Coriolis effects. Dong et al. [19] implemented a GA-based fractional-order PID controller to achieve active disturbance rejection performance in aircraft pitch control. To achieve rapid altitude tracking performance, Gao et al. [20] studied an state observer based cascaded PID altitude controller for hypersonic aircraft to ensure its attitude stability. Fenfen et al. [21] used an auto-disturbance rejection system (ADRS) combined with a PID controller to mitigate coupling and external disturbances, thereby enhancing tracking performance in hypersonic vehicles.

Taha et al. [22] developed a model reference adaptive control (MRAC) effective for designing authentication header (AH) mode, with improved results using online parameter estimation for the speed autopilot.

Perez et al. [23] developed an adaptive PID controller using a GA-based immunity algorithm to enhance supersonic fighter control and ensure model robustness under failures. Perez et al. [24] used an immunity-based algorithm with dynamics inversion and MRAC to maintain fighter performance under failures with a minimal pilot input. Park et al. [25] designed a gain-scheduling autopilot for hypersonic vehicles to control the pitch rate, acceleration and flight path angle amid model uncertainties.

Hiliuta et al. [26,27] evaluated ANFIS and fuzzy clustering to approximate aerodynamic forces in a F/A-18 aeroservoelastic model. They found that the least square method was ineffeive for unsteady aerodynamic forces analyses in the frequency domain and proposed combining fuzzy clustering with shape-preserving methods to improve model accuracy for intermediate frequencies using STructural Analysis RoutineS (STARS) software.

Furthermore, several studies have focused on developing controllers for hypersonic, fighter, supersonic, commercial and business aircraft, as outlined below.

PID-based loop shaping and an H_∞ controllers were employed for the autolanding system of a twin-engine civil aircraft to deal with a 25-knots crosswind [28]. Meanwhile, Qiu et al. [29] implemented a non-minimum phase dynamic inversion controller using a U-model root solver for the altitude-hold autopilot of a B747 aircraft to address right-half-plane zeros. Similarly, Santos and Oliveira [30] demonstrated the accurate performance of a PID control methodology using an X-plane flight simulator.

https://doi.org/10.1017/aer.2025.10114 Published online by Cambridge University Press

This method was also applied for a general aviation civil aircraft, Islam et al. [31] and for a Piper PA-28-236-DAKOTA aircraft [32]. This controller was developed by using root-locus design technique for parameter adjustment, aiming to achieve an overshoot of less than 15% in turbulent condition.

Different AI-based autopilot systems were also designed for civil aircraft. For instance, Nivison and Khargonekar [33] proposed a deep gated recurrent neural network (DGRNN) controller for the aircraft longitudinal motion to achieve robustness and trajectory-tracking performance. Using a combination of RNN and GA systems, Juang and Chiou [34] developed an enhanced autolanding system under severe wind shear conditions. Vandana et al. [35] designed a two-loops autopilot system that integrated a robust control system combined with an Neural Network (NN) and a UDE for a rigid-body aircraft.

In this article, the proposed longitudinal autopilot control system was tested and validated on a nonlinear simulation platform designed for Cessna Citation X (CCX) business aircraft. Previously, a linear quadratic regulator (LQR) was combined with the guardian maps theory to achieve robustness against uncertainties for the CCX lateral motion [36]. Boughari et al. [37] combined an optimal controller with a meta-heuristic optimisation technique to ensure stability of linear and nonlinear models of the CCX aircraft. A combination of PI controller and guardian maps theory was used into the stability and control augmentation systems to control the CCX pitch rate [38].

In addition to conventional controllers, several AI-based methodologies have also been proposed for the CCX aircraft Quintin et al. [39] integrated a dynamic inversion system into adaptive neural network (ANN), and a PID controller to control the Cessna Citation X (CCX) roll rate control. Furthermore, combining MPC system with recursive least square (RLS) algorithm and adaptive control system can be efficient to control the CCX pitch rate [40].

Previously, the authors developed different AI-based controllers for the CCX aircraft, including type-two adaptive fuzzy SMC (T2AFSMC) [41] and a model-referenced adaptive RNN controller [42] for its longitudinal motion and a combination of T2AFLS, PSO-based and adaptive super-twisting SMC systems for its lateral motion [43] across different flight conditions.

This article focuses on developing an MFRNN-based autopilot system for the CCX longitudinal motion during cruise, under both ideal and turbulent conditions using Dryden turbulence model with moderate-intensity value (10^{-3})). The novel methodology introduces new features to flight control systems designed using the AI. This autopilot system does not rely on aircraft dynamics models, and it can handle the uncertainties, highly nonlinear and unmodeled dynamics by its adaptive and real-time learning characteristics. These properties can guarantee the aircraft robustness and performance across a wide range of flight conditions without the need for constant parameters adjustment.

Therefore, the main contributions of this research are listed below.

1. A novel MFRNN-based sliding mode control system for this application

In previous studies, researchers mostly focused on conventional control systems, such as PID, LQR and conventional neural networks (e.g., ANN, GRU). In contrast, in this research, we assumed that the aircraft nonlinear dynamics model is unknown, and we have no knowledge about aircraft systems. To address this challenge, we used an MFRNN as a dynamic approximator within our control system. This approach enhances both robustness and adaptability, enabling the system to effectively manage highly nonlinear dynamics; capabilities that cannot be reliably achieved using simpler architectures or linear controllers.

2. Novel hybrid online training process using PSO-based backpropagation

Although the MFRNN structure was adapted as proposed in Fei et al. [44], this paper introduces a new training algorithm that combines particle swarm optimisation (PSO) with a backpropagation algorithm. The proposed methodology includes:

https://doi.org/10.1017/aer.2025.10114 Published online by Cambridge University Press

- **Off-line initialisation** of MFRNN weights using PSO, which facilitates faster convergence.
- **On-line weight adjustments** using backpropagation algorithm to ensure real-time adaptability during flight.

This combination of an off-line initialisation and an on-line training is both unique and novel for the following reasons. This model does not require any dataset or any prior knowledge about the unknown dynamics of aircraft. In addition, the handling of timeseries inputs is a big challenge to approximate spontaneously aircraft dynamics even under turbulence; this capability cannot be found in other research articles. Moreover, the simulation results showed that this methodology could guarantee the robustness and adaptability of the autopilot system by approximating the aircraft dynamics appropriately and integrating it into the SMC system.

3. Application to a high-fidelity nonlinear simulation platform

This autopilot system is validated using a nonlinear simulation platform for CCX aircraft, developed based on a Level D Research aircraft flight simulator (RAFS) at the Laboratory of Applied Research in Active Controls, Avionics and Aeroservoelasticity (LARCASE). This RAFS provides flight dynamics data certified to the highest level of accuracy by the FAA, ensuring excellent validation results.

4. A novel fuzzy-based mode transition algorithm

A new fuzzy logic-based mode transition algorithm was developed to manage switching between the altitude hold (AH) and vertical speed (VS) modes. This algorithm was inspired by the approach developed by us in-house at LARCASE [45]. Unlike conventional autopilot systems, which include three modes such as AH, VS and altitude capture, our system proposes a novel fuzzy logic-based mode transition algorithm with two modes improving performance and efficiency of the autopilot system by reducing the complexity of the transition algorithm across the entire flight envelope of the CCX aircraft during cruise (all 925 flight conditions). The proposed method ensures universal applicability and adaptability, even under both ideal and turbulent conditions.

5. Rigorous stability proof using the Lyapunov theorem

This article presents a Lyapunov-based stability analysis for the entire autopilot system, demonstrating that the proposed autopilot system is asymptotically stable. Previous works almost neglected the stability analysis or have not applied Lyapunov theory rigorously across the entire system.

6. Pitch rate control system design in the inner loop of the autopilot system

The inner loop of our proposed autopilot to control the CCX pitch rate was developed based on our previous work on the application of a Type-one adaptive fuzzy sliding mode control (T1AFSMC) system in Hosseini et al. [46].

Table 1 illustrates a brief explanation of the methodologies proposed in previous articles to clarify the contributions of this research article to the development of an aircraft autopilot system.

This research is novel for its application to a commercial aircraft such as the CCX and in proposing a new fuzzy-based transition algorithm and MFRNN approximation method, trained with a new approach. A summary of the contributions is shown in Fig. 1.

The rest of this paper is organised as follows: Section 2 introduces the aircraft nonlinear model, turbulence model, PSO-based MFRNN design, VS and AH mode controllers and the fuzzy-based transition algorithm. Section 3 presents the results and performance analysis, and Section 4 concludes with a summary of the findings.

2.0 Background

In this section, the three methods used in this study are discussed for designing an autopilot for the aircraft longitudinal motion during cruise.

https://doi.org/10.1017/aer.2025.10114 Published online by Cambridge University Press

***Table 1.** Previous methodologies developed for the aircraft autopilot systems*

Article	Methodology	Data preprocessing or dynamics approximation	Training process	Stability proof	Aircraft type	Flight conditions	Simulation platform
Ahsan et al. [3]	PID	–	–	No	Generic UAV	Ideal	Linearised and nonlinear
Ahsan and Hanif [2]	PID	–	–	No	Aerosonde UAV	Ideal	Linearised
Cárdenas et al. [5]	Pole-placement, LQG and state feedback	–	–	No	Unmanned airplane	Severe atmospheric disturbance	Linearised
Win et al. [8]	PID	–	–	No	YTU-EC 001 UAV	Small disturbance	Linearised
Quin and Liu [9]	UDE-based controller	–	–	Lyapunov & reduction theorem	Quadrotor	Ideal	Nonlinear
Baomar and Bentley [47]	Modular ANN systems	ANN	Offline	–	B787 (X-plane)	Extreme weather condition	Nonlinear
Liu et al. [48]	Deep deterministic policy gradient (DDPG)	NN	Offline	–	B737 model (X-plane)	Crosswind conditions	Nonlinear
Nivison and Khargonekar [33]	GRU-based RNN	Polynomial approximations	Offline	Lyapunov funnels	Flight vehicle model	Noise and input disturbance	Nonlinear
Juang and Chiou [34]	Hybrid RNN-GA	–	Gradient descent algorithms	–	Commercial aircraft	Under various wind shear	Simplified
Vandana et al. [35]	UDE-based MLP-NN	–	Backpropagation algorithm	–	Rigid model	External disturbances	Nonlinear mathematical model

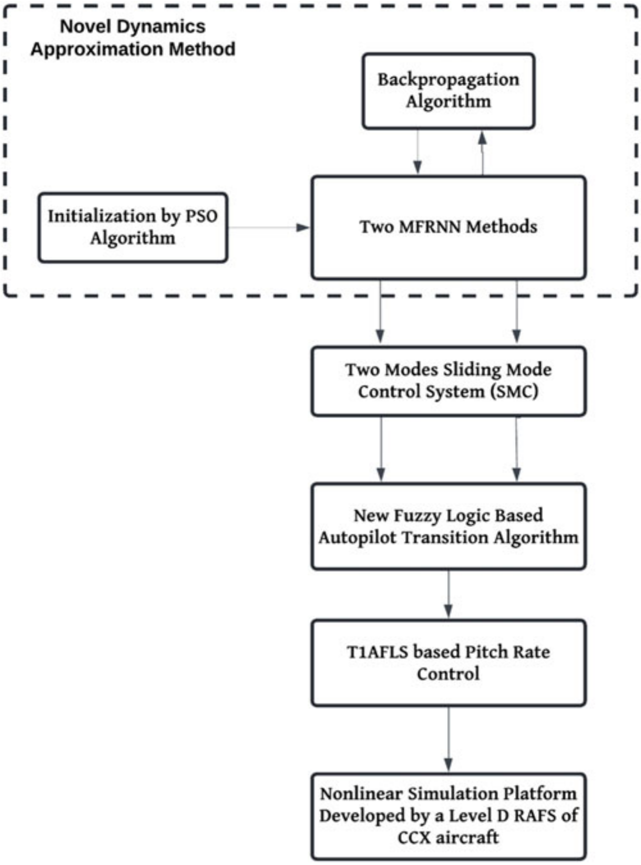

Novel Dynamics Approximation Method
Backpropagation Algorithm
Initialization by PSO Algorithm
Two MFRNN Methods
Two Modes Sliding Mode Control System (SMC)
New Fuzzy Logic Based Autopilot Transition Algorithm
T1AFLS based Pitch Rate Control
Nonlinear Simulation Platform Developed by a Level D RAFS of CCX aircraft

Figure 1. *Simple scheme of the research contributions.*

2.1 Nonlinear state-space aircraft representation

The state-space model is a well-known representation of system dynamics. This representation is typically characterised by two functions, $F_{(vs,h)}$ and $G_{(vs,h)}$, which reflect the aircraft dynamics variations, as the functions existing in the vertical speed (vs) and altitude (h) control systems [49]. Generally, $G_{(vs,h)}$ in aircraft control systems shows the effectiveness of the control inputs which varies in small ranges during the cruise. Moreover, it was observed that $G_{(vs,h)}$ varied with values around 1. Therefore, to reduce the complexity of the control system design, we assumed that $G_{(vs,h)}$ is approximately equal to 1 $G_{(vs,h)} \approx 1$. Consequently, the aircraft state-space representation for the vertical speed and altitude modes can be described with a vector of state variables and the control input (i.e., command) denoted by $x = [VS, h]$ and $u_{(vs,h)}$, respectively:

$$x^{(n)} = F_{(vs,h)} + G_{(vs,h)} u_{(vs,h)} + D = F_{(vs,h)} + u_{(vs,h)} + D \tag{1}$$

where n is the order of derivative, and D is the atmospheric turbulence.

Furthermore, we used a simulation platform developed at the LARCASE for the CCX business jet aircraft [45]. This platform was validated using the flight data from a Level-D RAFS available at LARCASE for this business aircraft, as illustrated in Fig. 2. Level D represents the highest certification be issued by the Federal Aviation Authority (FAA) for aircraft flight simulation in terms of flight data accuracy [45].

https://doi.org/10.1017/aer.2025.10114 Published online by Cambridge University Press

Figure 2. *Level D RAFS For the CCX aircraft at LARCASE.*

2.1.1 Turbulence model

The Dryden turbulence model is commonly used in aerospace engineering to simulate the atmospheric gusts and turbulence that an aircraft may encounter during flight. This realistic representation of atmospheric turbulence helps to evaluate the safety and performance of the proposed control system. We used the continuous Dryden turbulence model available in MATLAB/Simulink (R2023b) Aerospace Environment blockset configured according to MIL-F-8785C. In this configuration, the turbulence probability of the exceedance of high-altitude intensity was selected as moderate (10^{-3}) with wind speed of 5 meters per second and scale length at medium/high altitudes of 533.4m. In addition, the wingspan of the CCX aircraft was chosen to configure the turbulence model. The turbulence intensity varies with altitude, and it is influenced by the probability of exceeding [50]. In Equation (1), D denotes the atmospheric turbulence in the state-space representation.

2.2 PSO-based MFRNN approximators

As discussed earlier, the nonlinear state-space representations of the CCX aircraft contain the unknown functions $F_{(vs,h)}$, which were approximated by the following MFRNN, as proposed in Fei et al. [44], for the design of autopilot system.

The main architecture of the MFRNN is presented in Fig. 3, which includes four main layers: (1) the input layer, (2) the membership layer, (3) the rules layer and (4) the output layer. There are some feedback loops in the MFRNN, as shown in Fig. 3, which can provide the MFRNN with a self-learning ability to approximate the aircraft dynamics.

In this autopilot system, two MFRNNs were developed, one for VS mode and the other architecture for AH mode. The main structure of the layers in these MFRNNs is explained as follows, which is inspired from the well-known structure of the fuzzy logic system, including a fuzzifier (membership layer), an inference engine (rule layer), a defuzzifier (output layer):

1. Input layer (2 neurons):

Initially, two signals were measured from the CCX simulation platform: the vertical speed *VS* and the aircraft altitude h. These signals were used in the definition of the tracking errors e_{vs}, and e_h as calculated

https://doi.org/10.1017/aer.2025.10114 Published online by Cambridge University Press

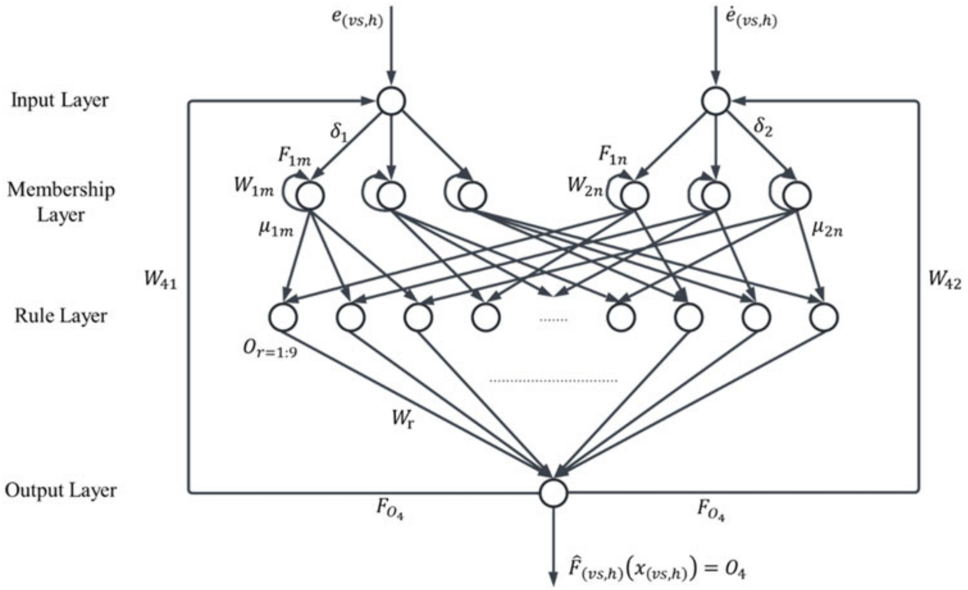


Figure 3. Multilayer fuzzy recurrent neural network (MFRNN) scheme.

in Equations (2) and (3):

$$e_{vs} = VS - VS_{ref} \tag{2}$$

$$e_h = h - h_{ref} \tag{3}$$

Two inputs vectors represented by I_{vs} (for the VS mode) and I_h (for the AH mode) were used in the MFRNNs, each composed of the tracking error $e_{(vs,h)}$ and its first-order derivative $\dot{e}_{(vs,h)}$, as shown in Equations (4.a) and (4.b), for the VS and AH modes, respectively.

$$I_{vs} = [e_{vs}; \dot{e}_{vs}] \tag{4.a}$$

$$I_h = [e_h; \dot{e}_h] \tag{4.b}$$

The inputs defined in Equations (4.a) and (4.b) reflect the control objectives by capturing essential information about deviations in the aircraft dynamics without requiring explicit dependence on the full set of dynamics variables. In addition, the choice of these inputs avoids redundancy, since pitch-rate variations were already represented in the inner loop, and it enhanced stable learning by reducing the effects of undesired variations such as turbulence in the learning process.

In these MFRNNs, shown in Fig. 3, there is a feedback loop from the output layer ($\hat{F}_{(vs,h)} = O_4$) to the input layer expressed by F_{O_4}. Using the elements in the inputs vectors given in Equation (4.a) for the VS mode denoted by I_{vs}, and in Equation (4.b) for the AH mode represented by I_h, we can write:

$$\delta_1 = e_{(vs,h)}.W_{41}.F_{O_4} \tag{5.a}$$

$$\delta_2 = \dot{e}_{(vs,h)}.W_{42}.F_{O_4} \tag{5.b}$$

where δ_1 is the first neuron output, and δ_2 is the second neuron output in the input layer. Moreover, W_{41} and W_{42} denote the recurrent weights acting on the connections (internal feedback loop) between the output and the input layers.

2. Membership layer (6 neurons):

In this layer, each neuron is a Gaussian membership function denoted by μ_{1m} and μ_{2n} that processes the output of the previous layer (the input layer), denoted by δ_1 and δ_2 in Equations (5.a) and (5.b), and

https://doi.org/10.1017/aer.2025.10114 Published online by Cambridge University Press

Table 2. *Configurations of the membership functions in the VS mode*

m, n indices	Parameters	Value	Parameters	Value
1	$a_{11}= a_{21}$	400	$b_{11} = b_{21}$	0
2	$a_{12} = a_{22}$	400	$b_{12} = b_{22}$	1825
3	$a_{13} = a_{23}$	400	$b_{13} = b_{23}$	3650

Table 3. *Configurations of the membership functions in the AH mode*

m, n indices	Parameters	Value	Parameters	Value
1	$a_{11}= a_{21}$	4000	$b_{11} = b_{21}$	0
2	$a_{12} = a_{22}$	4000	$b_{12} = b_{22}$	3810
3	$a_{13} = a_{23}$	4000	$b_{13} = b_{23}$	7620

the feedback connections shown by F_{1m} and F_{2n}, multiplied by their weights W_{1m} and W_{2n}, respectively. F_{1m} and F_{2n} contains the previous output values (feedback) of the membership functions calculated by Equations (8.a) and (8.b) [44].

$$\delta_{1m} = \delta_1 + W_{1m}.F_{1m} \tag{7.a}$$

$$\delta_{2n} = \delta_2 + W_{2n}.F_{2n} \tag{7.b}$$

$$\mu_{1m}(x) = e^{-0.5\left(\frac{\delta_{1m}-b_{1m}}{a_{1m}}\right)^2} \tag{8.a}$$

$$\mu_{2n}(x) = e^{-0.5\left(\frac{\delta_{2n}-b_{2n}}{a_{2n}}\right)^2} \tag{8.b}$$

In Equations (7.a)–(8.b), $m = \{1, 2, 3\}$ and $n = \{1, 2, 3\}$ are the neurons indices. Equations (7.a) and (7.b) show that when calculating the input layers δ_{1m} and δ_{2n}, the weights W_{1m} and W_{2n} act on the internal feedback loops (F_{1m} and F_{2n}) in the membership layers. The values of the parameters in the Gaussian memberships are provided in Tables 2 and 3 for each VS and AH mode approximator, as follows.

3. Rules layer (9 neurons)

Each neuron in this layer was used to calculate the product of the input signals serving as the product inference engine in the fuzzy logic system (FLS) architecture. The output of this layer O_r is given in Equation (9):

$$O_r = \prod_{m.n=1}^{3} \mu_{1m}(x).\mu_{2n}(x) \tag{9}$$

where $r = \{1, \ldots, 9\}$ is the total number of the rules.

4. Output layer (1 neuron)

Using the 9 outputs O_r obtained at each neuron in the rules layer and their corresponding weights, the final output of the MFRNN-based approximators can be expressed in Equation (10) for both the VS and AH controllers:

$$\hat{F}_{(vs,h)}\left(x_{(vs,h)}\right) = \sum_{r=1}^{9} W_r O_r \tag{10}$$

https://doi.org/10.1017/aer.2025.10114 Published online by Cambridge University Press

Table 4. *PSO algorithm parameters*

Constants	Values
Acceleration parameter (Ω_1)	2
Acceleration parameter (Ω_2)	2
Inertia damping parameter (α_{damp})	0.99
Swarm size (total population) (p_{max})	500
Maximum number of the iterations (i_{max})	10
Lowest bound of searching range (X_{lo})	-1
Highest bound of searching range (X_{up})	1

2.2.1 The MFRNN training algorithm

Each of the MFRNN-based approximators, explained earlier for both autopilot modes, was trained using a new approach, as follows:

i. Initially the MFRNNs receive the vector x_{vs} in the VS controller and the vector x_h in AH controller as input layers.

ii. All weights $\{W_{41}, W_{42}, W_{1m}, W_{2n}$ and $W_r\}$ were initialised using the PSO algorithm designed by the expressions in Equations (11) and (12), Gad [51] and the cost function given in Equation (13), while $W_{41} = W_{42}$ and $W_{1m} = W_{2n}$ on the feedback loops. Thus, the PSO was used to find the values of three weights, called *decision variables* $=\{W_{4(1,2)}, W_{(1m,2n)}, W_r\}$. Each particle X_p^i in the PSO searches in a space between its bounds $[X_{lo}, X_{up}]$, which means that all decision variables could only take a value within the defined range. In addition, in Equation (11), P_{best} and G_{best} are the personal and global best results, respectively, and β_1 and β_2 denote two random values within [0, 1] [51].

$$\sum_{i=1}^{i_{max}} \sum_{p=1}^{p_{max}} Y_p^{i+1} = \alpha_{damp} \times Y_p^i + \Omega_1 \times \beta_1 * \left(P_{best} - X_p^i\right) + \Omega_2 \times \beta_2 \left(G_{best} - X_p^i\right) \tag{11}$$

$$\sum_{i=1}^{i_{max}} \sum_{p=1}^{p_{max}} X_p^{i+1} = X_p^i + Y_p^{i+1} \tag{12}$$

$$Cost\left(Y_p^i\right) = \frac{1}{2} \times \sum_{0}^{\tau} \left(e_{(vs,h)}\right)^2 \tag{13}$$

Equation (13) was selected to directly minimise the tracking error, which constitutes the main control objective of this study. Although the literature shows that model-free approximation methods may incorporate the equations of motion (EOMs) as constraints within the cost function, such an approach contradicts our objective of prioritising tracking-error minimisation. In practice, it was observed that introducing the EOMs as additional penalty terms tended to bias the optimisation toward reproducing intermediate dynamics variables rather than minimising the tracking error. Therefore, the defined cost function in Equation (13) solely based on the tracking errors ensures consistent agreement between the learning and control objectives without degrading the performance of the autopilot system.

In Equations (11)–(13), Ω_1 and Ω_2 are the personal and social acceleration parameters, respectively, α_{damp} is the damping inertia, p_{max} is the total swarm size (population), i_{max} is the maximum number of iterations, Y_p^i is the particle velocity and X_p^i is the particle position. The values for each parameter in Equations (11) and (12) in Table 4.

As shown in Table 4, the maximum number of iterations i_{max} was set to 10 to achieve a good balance between computational efficiency and accuracy. It was observed that increasing the number of iterations primarily led to longer search times for the PSO, without significantly improving control system performance. Similarly, the swarm size p_{max} was set to 500, which was empirically

https://doi.org/10.1017/aer.2025.10114 Published online by Cambridge University Press

determined through extensive trial and error testing to ensure a good exploration of the search space. Increasing the swarm size beyond this value did not meaningfully enhance performance but substantially increased computational time.

iii. Finally, the weights connecting the rules layer to the output layer, denoted by $W_{r(1\times 9)}$, were updated with the initial values founded by the PSO, using a backpropagation (BP) algorithm, as shown below.

$$W_r\,(t+1) = W_r\,(t) - \eta_{w_r}\Delta W_r \tag{14}$$

$$\tilde{F}_{(vs,h)} = F_{(vs,h)} - \hat{F}_{(vs,h)} \tag{15}$$

$$\Delta W_r = \tilde{F}_{(vs,h)} O_r \tag{16}$$

where $\eta_{w_r} = 0.7$ is the learning rate which was found after trying different values, $W_r(t)$ represents the current weights and $W_r\,(t+1)$ denotes the future value of the weights, and$\tilde{F}_{(vs,h)}$ is the error between the predicted signal$\hat{F}_{(vs,h)}$ and the reference signal $F_{(vs,h)}$. This training algorithm helps to update the approximated function with respect to the given reference online at each iteration. The reference signal in the VS controller was $F_{vs} = \dot{VS}$, and in the AH controller $F_h = \dot{h}$ measured from the nonlinear simulation platform for the CCX aircraft.

2.3 Design of the VS and AH mode control systems

This section describes the control methodologies designed for the VS and AH modes. These controllers were developed based on an SMC system combined with PSO-based MFRNN approximators, as explained in the previous section. Furthermore, a novel transition algorithm was implemented using the FLS to switch between autopilot modes during the flight, thus efficiently to capture the altitude.

2.3.1 VS mode control system design

In the VS mode control system, a MFRNN-based sliding mode controller was proposed to force the aircraft to track the desired vertical speed. Accordingly, the tracking error $e_{vs} = VS - VS_{ref}$ given in Equation (2) and its first-order derivative $\dot{e}_{vs}$ were used to define the sliding surface, given in Equation (17) [52]:

$$S_{vs} = \dot{e}_{vs} + L_{vs}e_{vs} \tag{17}$$

where $L_{vs} > 0$.

Taking the first-order derivative of the sliding surface defined in Equation (17), and knowing that $\ddot{e}_{vs} = \ddot{VS} - \ddot{VS}_{ref} = F_{vs} + u_{vs} + D - \ddot{VS}_{ref}$ from the state-space representation in Equation (1), it gives:

$$\dot{S}_{vs} = \ddot{e}_{vs} + L_{vs}\dot{e}_{vs} = F_{vs} + u_{vs} + D - \ddot{VS}_{ref} + L_{vs}\dot{e}_s \tag{18}$$

Therefore, the equivalent control law $u_{eq_{vs}}$ in the SMC system can be obtained by considering $\dot{S}_{vs} = 0$ [52]. In addition, the turbulence D is unknown, so $u_{eq_{vs}}$ becomes:

$$u_{eq_{vs}} = -F_{vs} + \ddot{VS}_{ref} - L_{vs}\dot{e}_{vs} \tag{19}$$

The switching control law $u_{sw_{vs}}$ was designed in its conventional form [52], where *sat* stands for the saturation function, and $k_{vs} > 0$:

$$u_{sw_{vs}} = -k_{vs}sat\,(S_{vs}) \tag{20}$$

In Equation (19), we assumed that F_{vs} is an unknown nonlinear function; therefore, it should be replaced with its approximated function$\hat{F}_{vs}$, as calculated with Equation (10). Thus, the VS mode control law u_{vs}, using both Equations (19) and (20), becomes:

$$u_{vs} = u_{eq_{vs}} + u_{sw_{vs}} = -\hat{F}_{vs} + \ddot{VS}_{ref} - L_{vs}\dot{e}_{vs} - k_{vs}sat\,(S_{vs}) \tag{21}$$

To prove the stability of the system, the following Lyapunov candidate V_{vs} is defined [52]:

$$V_{vs} = \frac{1}{2}(S_{vs})^2 \tag{22}$$

Then, the derivative of the V_{vs} is calculated as given in Equation (23), by using the expression of $\dot{S}_{vs}$ in Equation (18) and by replacing u_{vs} with its expression in Equation (21), as follows:

$$\dot{V}_{vs} = S_{vs}\dot{S}_{vs} = S_{vs}\left[F_{vs} + u_{vs} + D - \ddot{V}S_{ref} + L_{vs}\dot{e}_{vs}\right] = S_{vs}\left[F_{vs} - \hat{F}_{vs} + D - k_{vs}sat\left(S_{vs}\right)\right] \tag{23}$$

In this equation, we assumed that F_{vs} and D are two unknown bounded functions such that $\left|F_{vs}\right| \leq F_{vs_{MAX}}$ and $\left|D\right| \leq D_{MAX}$ [53–55], where $F_{vs_{MAX}}$ and D_{MAX} are two positive values. Thus, due to the following aspects,$\hat{F}_{vs}\left(x_{vs}\right)$ can be considered as a bounded function:

i. As presented in the membership layer of the MFRNN, we used two Gaussian memberhsip functions denoted by $\mu_{1m}\left(x\right)$ and $\mu_{2n}\left(x\right)$ in Equations (8.a) and (8.b), respectively. These membership functions are smooth and naturally bounded, with outputs that change over a finite interval, meaning that they are bounded. These outputs are propagated to the next layer (the rules layer) which calculates the product of the received inputs. Thus, mathematically, the product of some bounded inputs is also bounded [56].

ii. As explained in Section 2.2.1, the weights between the rules layer and the output layer are updated by $W_r\left(t+1\right) = W_r\left(t\right) - \eta_{w_r}\Delta W_r$ (see Equation (14)). In this equation, η_{w_r} should be selected to be small enough to ensure that weights W_r are adjusted such that the error between the approximated function and its given reference is reduced over time.

iii. According to the explanations above and by knowing the boundedness of the output of the rule layer, we can conclude that$\hat{F}_{vs}$ also varies in a bounded interval, so that $\left|\hat{F}_{vs}\right| \leq Q_{vs_{MAX}}$, where $Q_{vs_{MAX}} > 0$.

Therefore, using the triangle inequality [57], we can write:

$$\left|F_{vs} - \hat{F}_{vs}\right| \leq \left|F_{vs}\right| + \left|-\hat{F}_{vs}\right| \tag{24}$$

where $\left|F_{vs}\right| \leq F_{vs_{MAX}}$ and $\left|\hat{F}_{vs}\right| \leq Q_{vs_{MAX}}$. Thus, $\left|F_{vs} - \hat{F}_{vs}\right| \leq F_{vs_{MAX}} + Q_{MAX} = \varepsilon_{vs}$ and ε_{vs} is a too small positive constant.

According to the proof of boundedness in Equation (23), we can rewrite $\dot{V}_{vs}$ as follows:

$$\dot{V}_{vs} \leq \left|S_{vs}\right|\left|F_{vs} - \hat{F}_{vs}\right| + \left|S_{vs}\right|\left|D\right| - k_{vs}\left|S_{vs}\right| \leq \left|S_{vs}\right|\left(\varepsilon_{vs} + D_{MAX} - k_{vs}\right) \tag{25}$$

So that:

$$\dot{V}_{vs} \leq \left|S_{vs}\right|\left(\varepsilon_{vs} + D_{MAX} - k_{vs}\right) \tag{26}$$

The expression in Equation (26) indicates that for $k_{vs} > \varepsilon_{vs} + D_{MAX}$, there will be always $\dot{V}_{vs} < 0$, and therefore the proposed control system will be asymptotically stable.

2.3.2 AH mode control system design

Similarly, the AH mode controller can be designed using the sliding surface defined in terms of the tracking error e_h in Equation (3) and its first-order derivative $\dot{e}_h$, as follows [52]:

$$S_h = \dot{e}_h + L_h e_h \tag{27}$$

where L_h is a positive constant.

Taking the derivative of the sliding surface S_h, and using the state-space representation $\ddot{h}\left(x\right) = F_h + u_h + D$ according to Equation (1), and the second order time derivative of the tracking error $\ddot{e}_h = \ddot{h} - \ddot{h}_{ref}$ from Equation (3), the sliding surface derivative becomes:

$$\dot{S}_h = \ddot{e}_h + L_h\dot{e}_h = F_h + u_h + D - \ddot{h}_{ref} + L_h\dot{e}_h \tag{28}$$

https://doi.org/10.1017/aer.2025.10114 Published online by Cambridge University Press

As presented for the VS mode in Section 2.3.1, for the AH mode, the equivalent control term u_{eq_h} was designed using $\dot{S}_h = 0$ [52], which lead to the final form of the control law u_h:

$$u_h = u_{eq_h} + u_{sw_h} = \underbrace{-\hat{F}_h + \ddot{h}_{ref} - L_h \dot{e}_h}_{u_{eq_h}} \underbrace{-k_h sat\left(S_h\right)}_{u_{sw_h}} \tag{29}$$

To prove the stability of the control law in Equation (29), the Lyapunov candidate in Equation (30) and its derivative in Equations (31) and (32) become [52].

$$V_h = \frac{1}{2}\left(S_h\right)^2 \tag{30}$$

$$\dot{V}_h = S_h \dot{S}_h = S_h \left[F_h + u_h + D - \ddot{h}_{ref} + L_h \dot{e}_h\right] \tag{31}$$

$$\dot{V}_h = S_h \left[F_h - \hat{F}_h + D - k_h sat\left(S_h\right)\right] \tag{32}$$

The same calculations were applied for the AH mode controller as for the VS mode controller. Replacing u_h in Equation (31) with its expression given in Equation (29) and knowing that $\left|F_h - \hat{F}_h\right| \leq F_{h_{MAX}} + Q_{h_{MAX}} = \varepsilon_h$, which was calculated from the triangle inequality [57] (with $\left|F_h\right| \leq F_{h_{MAX}}$ and $\left|\hat{F}_h\right| \leq Q_{h_{MAX}}$, where $F_{h_{MAX}}$ and $Q_{h_{MAX}}$ are positive values and $\left|sat\left(S_h\right)\right| \leq \left|S_h\right|$), the final expression of $\dot{V}_h$ is obtained as shown in Equation (33):

$$\dot{V}_h \leq \left|S_h\right| \left(\varepsilon_h + D_{MAX} - k_h\right) \tag{33}$$

Similarly, with respect to the expression in Equation (33) and with $k_h > \varepsilon_h + D_{MAX}$, it can be concluded that $\dot{V}_h < 0$, representing the asymptotical stability of the proposed control methodology for the AH mode system.

To ensure the appropriate transition between the VS and AH modes, a new algorithm was developed to switch between these modes and is described next in Section 2.4.

2.4 FLS-based autopilot mode transition algorithm

This section presents a novel mode transition algorithm based on the FLS to make smooth transition between modes. Using this transition algorithm, when the aircraft climbs or descends to the pilot's commanded altitude, as soon as it reaches a specific distance (Δh)) from this commanded altitude, the transition algorithm switches from the VS mode to the AH mode. Once the aircraft reaches this distance, the AH control system tries to maintain the commanded altitude, preventing the aircraft from climbing further or descending to lower altitudes.

As explained in Ghazi [45], the distance at which the AH mode becomes engaged can be calculated with in Equation (34):

$$\Delta h = h_{ref} - h_{trans} = \left(\frac{VS_{ref}{}^2}{0.05 \times g}\right)\left(\frac{e-1}{e}\right) \tag{34}$$

where $g \approx 9.81$ is the gravitational acceleration.

This fuzzy logic transition algorithm is the core decision-making component in this autopilot system, which switches between its modes. As shown in Fig. 4, an FLS is composed of four main components: (1) fuzzifier, (2) fuzzy rule base, (3) inference engine and (4) defuzzifier. These components are explained in details in Wang [55].

In this algorithm, we used the altitude tracking error e_h in Equation (3) as the input that was given to the FLS. This variable was fuzzified using three different membership functions defined by some linguistic variables such as Negative High Error (NHE), Medium Error (ME) and Positive High Error (PHE). Regarding these linguistic variables, we defined a minimum number of fuzzy rules to simplify the design of the FLS. In addition, the outputs were selected by two singleton values between 0 and 1

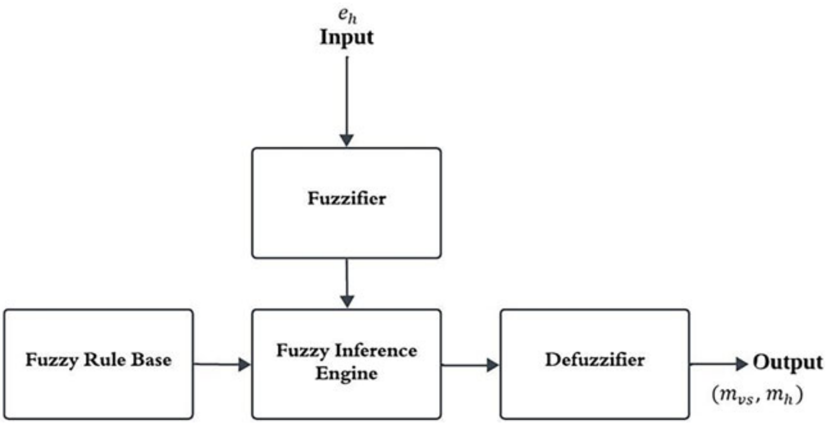


Figure 4. *Main components of the FLS.*

for each m_{vs} and m_h coefficient. The equations of each membership function are presented in Equations (35.a)–(35.c):

$$\mu_{\mathrm{NHE}}\left(e_h, a_1, a_2, a_3, a_4\right) = \begin{cases} 0 & \mathit{if}\ e_h < a_1 \\ \dfrac{e_h - a_1}{a_2 - a_1} & \mathit{if}\ a_1 \le e_h \le a_2 \\ 1 & \mathit{if}\ a_2 \le e_h \le a_3 \\ \dfrac{a_4 - e_h}{a_4 - a_3} & \mathit{if}\ a_3 \le e_h \le a_4 \\ 0 & \mathit{if}\ e_h < a_4 \end{cases} \tag{35.a}$$

$$\mu_{\mathrm{ME}}\left(e_h, b_1, b_2, b_3\right) = \begin{cases} 0 & \mathit{if}\ e_h < b_1 \\ \dfrac{e_h - b_1}{b_2 - b_1} & \mathit{if}\ b_1 \le e_h \le b_2 \\ \dfrac{b_3 - e_h}{b_3 - b_2} & \mathit{if}\ b_2 \le e_h \le b_3 \\ 0 & \mathit{if}\ e_h > b_3 \end{cases} \tag{35.b}$$

$$\mu_{\mathrm{PHE}}\left(e_h, c_1, c_2, c_3, c_4\right) = \begin{cases} 0 & \mathit{if}\ e_h < c_1 \\ \dfrac{e_h - c_1}{c_2 - c_1} & \mathit{if}\ c_1 \le e_h \le c_2 \\ 1 & \mathit{if}\ c_2 \le e_h \le c_3 \\ \dfrac{c_4 - e_h}{c_4 - c_3} & \mathit{if}\ c_3 \le e_h \le c_4 \\ 0 & \mathit{if}\ e_h > c_4 \end{cases} \tag{35.c}$$

The parameters values selected for each membership function are presented in Table 5.

The values of h_{trans} were calculated in accordance with the variations of VS_{ref}, as shown in Equation (34). After fuzzifying the altitude tracking error e_h using the membership functions in Equations (35.a) to (35.c), the fuzzy rules should be defined to specify the relationship between the input and the outputs in this FLS. These fuzzy rules were proposed using a single input e_h and multiple outputs such as m_{vs} and m_h. Applying each linguistic variables shown by *NHE*, *ME* and *PHE*, we defined two combined *IF-THEN* fuzzy rules, as follows:

Table 5. Parameter values of the membership functions in the transition algorithm

μ_{NHE}		μ_{ME}		μ_{PHE}	
Constant	Value	Constant	Value	Constant	Value
a_1	-51000	b_1	$-h_{trans}$	c_1	0
a_2	-50000	b_2	0	c_2	h_{trans}
a_3	$-h_{tans}$	b_3	h_{trans}	c_3	50000
a_4	0			c_4	51000

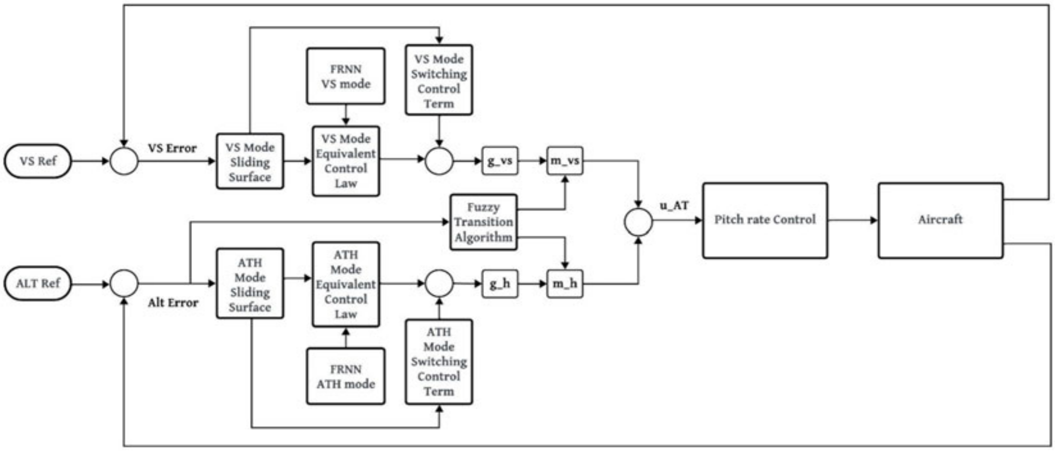

VS Ref
VS Error
VS Mode Sliding Surface
FRNN VS mode
VS Mode Equivalent Control Law
VS Mode Switching Control Term
g_vs
m_vs
Fuzzy Transition Algorithm
u_AT
Pitch rate Control
Aircraft
ALT Ref
Alt Error
ATH Mode Sliding Surface
ATH Mode Equivalent Control Law
FRNN ATH mode
ATH Mode Switching Control Term
g_h
m_h

Figure 5. *Block diagram of the developed autopilot system.*

- Rule 1. *IF* $\boldsymbol{e_h}$ is **NHE** *OR* $\boldsymbol{e_h}$ is **PHE** *THEN* $\boldsymbol{m_{vs}}$ is **Activated**
- Rule 2. *IF* $\boldsymbol{e_h}$ is **ME** *AND NOT* **NHE** *OR IF* $\boldsymbol{e_h}$ is ***ME*** *AND NOT* **PHE** *THEN* $\boldsymbol{m_h}$ is **Activated**

Typically, in an FLS, the operator *AND* is implemented by the *min* function, and *NOT* is a complement operator that can be expressed by $1 - \mu$. The relationship between the antecedent (*IF*-part) and the consequent (*THEN*-part) parts was aggregated using the *OR* operator implemented by the *max* function [55]. Therefore, the rules 1 and 2, given above, were defined as shown in Equations (36) and (37):

$$m_{vs} = \max\left[\min\left(\mu_{NHE}, 1\right), \min\left(\mu_{PHE}, 1\right)\right] \tag{36}$$

$$m_h = max\left[\min\left(\mu_{\text{ME}}, 1 - \mu_{NHE}\right), min\left(\mu_{\text{ME}}, 1 - \mu_{PHE}\right)\right] \tag{37}$$

By using this algorithm, at each time, one of the coefficients m_{vs} and m_h will be active with values between 0 and 1. Hence, the final control law u_{AT} was developed for the autopilot system with Equation (38), using the values obtained from Equations (36) and (37) for each m_{vs} and m_h, respectively:

$$u_{AT} = m_{vs} g_{vs} u_{vs} + m_h g_h u_h \tag{38}$$

where $g_{vs}, g_h > 0$.

Figure 5 illustrates the main architecture of the developed autopilot system in this article, where $_{AT}$ acts as the reference pitch rate signal given to the pitch rate controller in the inner loop. The pitch rate controller was designed by using a new T1AFSMC methodology, developed by researchers at LARCASE and published in Hosseini et al. [46].

https://doi.org/10.1017/aer.2025.10114 Published online by Cambridge University Press

Table 6. *Parameters for selecting 925 flight conditions*

Altitude (ft)	CAS (kts)	Xcg (% MAC)	Weight (lbs)
8000	180	24	26000
10000	200	26	27000
15000	230	28	28000
20000	250	30	29000
25000	300	32	30000
30000	330		
35000			
38000			
42000			

Table 7. *Parameters of the pitch rate control system (T1AFSMC)*

Parameters	Values
Sliding surface coefficient (C)	1.3
Adaptation law parameter (σ_f)	10^{-4}
Adaptation law parameter $\left(\sigma_g\right)$	10^{-4}
Adaptation law parameter (γ_f)	100
Adaptation law parameter (γ_g)	100
Switching control gain (L)	550
Integral control gain (k)	5500

Table 8. *Design parameters of the autopilot control laws*

Mode	Parameter	Value	Mode	Parameter	Value
VS mode	L_{vs}	5.3	AH Mode	L_h	5.5
	k_{vs}	1.3		k_h	0.8
	g_{vs}	0.085		g_h	0.085

3.0 Results

This section presents the experimental results obtained from a high-fidelity nonlinear simulation platform developed for the CCX aircraft, as well as comprehensive analysis using the methodologies proposed in this article for the design of the autopilot system. This platform was initialised using the parameter ranges given in Table 6 to generate a total of 925 different flight conditions, covering the entire flight envelope of the CCX. It is important to note that the values shown in each row of Table 6 do not represent any combinations of altitude, CAS, Weight and Xcg. Rather, they define the overall range of each parameter. The 925 flight conditions were generated by iteratively combining parameter values in (Altitude-CAS) and (Weight-Xcg) pairs, without implying that each altitude corresponds to a single or fixed CAS, Weight or Xcg value.

The pitch rate control system in the inner loop of this autopilot system was designed using the T1AFSMC presented in Hosseini et al. [46], whose parameters are given in Table 7.

In addition, the other parameters values in the proposed autopilot control laws, given in Equations (21), (29) and (38) are shown in Table 8.

It should be stated that all hyperparameters, gains and design parameters were carefully and experimentally selected to ensure the performance of the autopilot system across all flight conditions with and without turbulence.

https://doi.org/10.1017/aer.2025.10114 Published online by Cambridge University Press

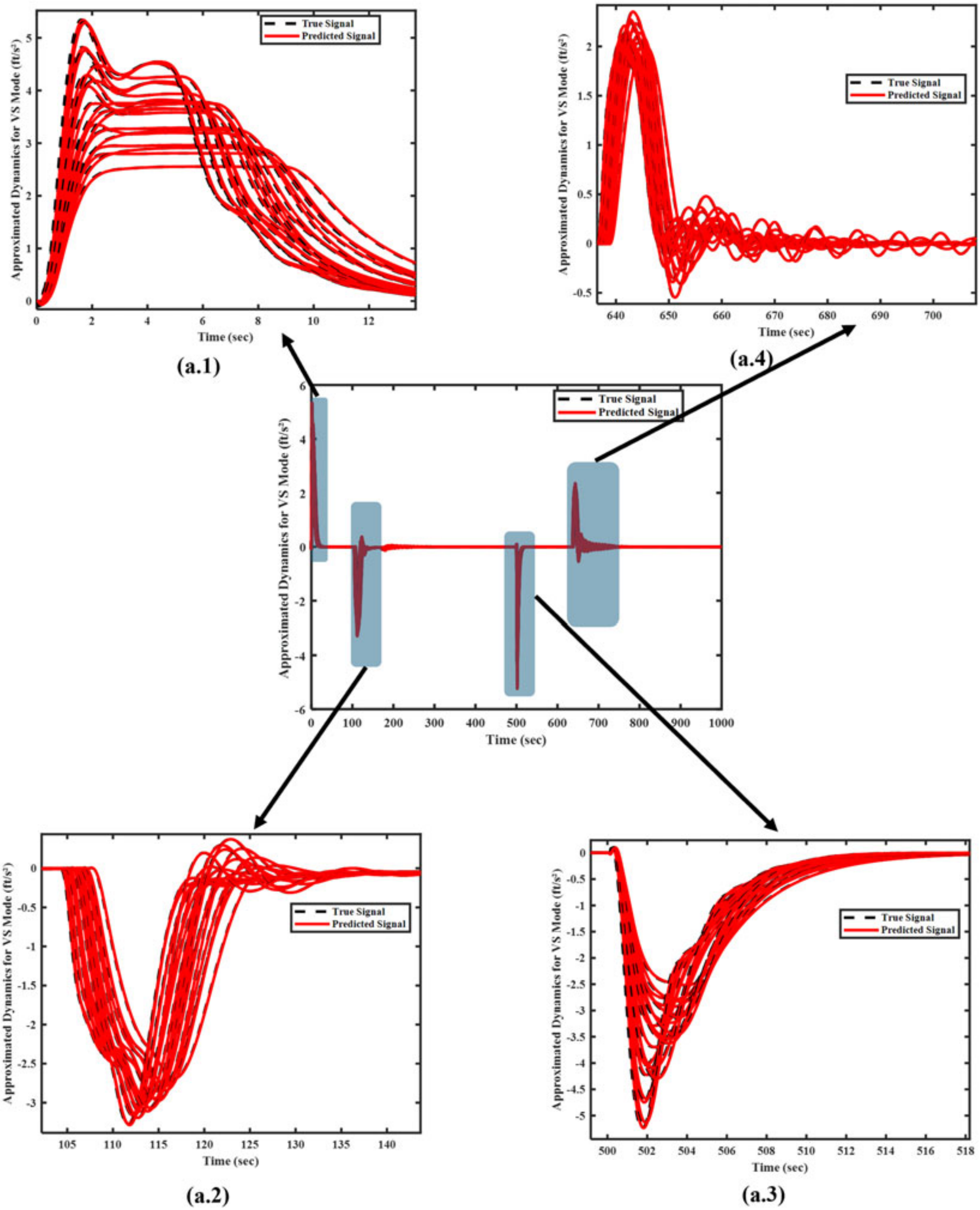
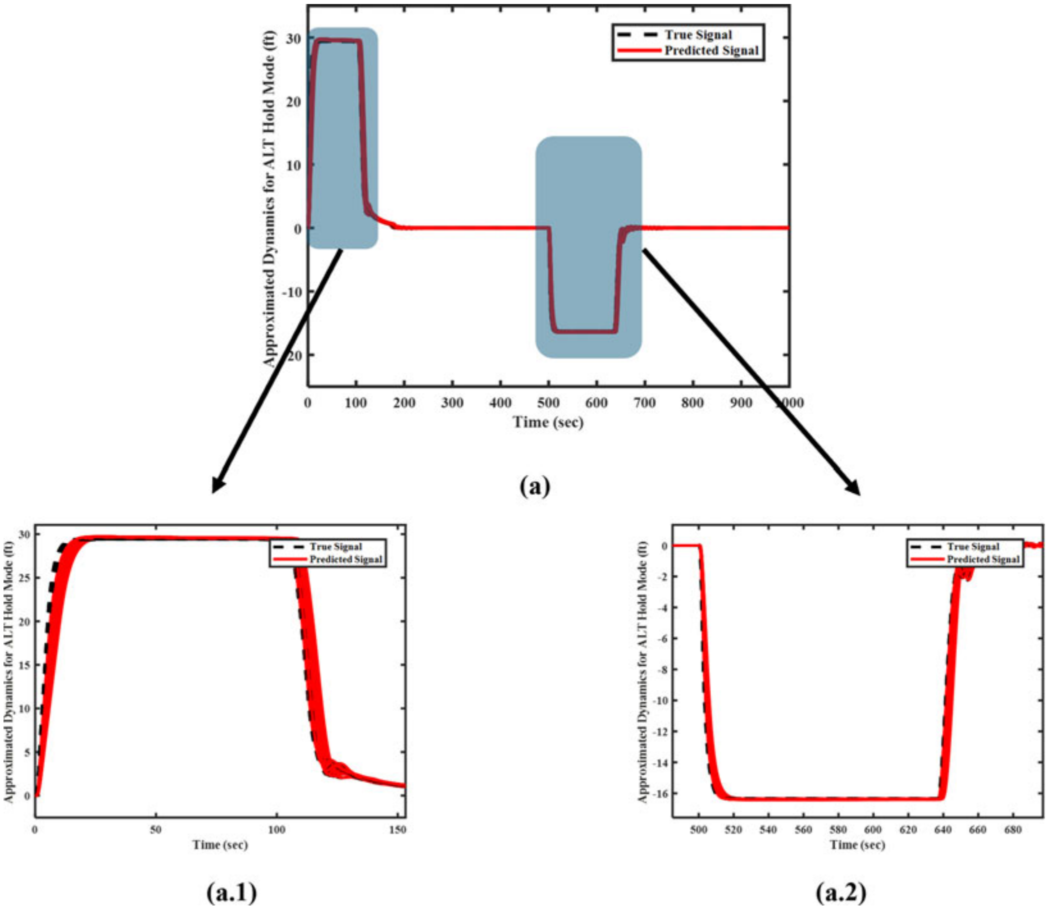


Figure 6. The variations of approximated dynamics vs the true signals in VS hold mode for 19 random flight conditions at different flight levels.

The validation of proposed methodologies in this article starts with the simulation results obtained for approximators proposed in Section 2.2. As previously described, two MFRNNs were used to approximate the aircraft dynamics in both VS and AH mode, without any prior knowledge of the aircraft model or its subsystems.

Figures 6 and 7 illustrate the variations of the approximated dynamics functions with respect to the true signals, which were defined as $F_{vs} = \dot{V}S$ for the VS mode, and $F_h = \dot{h}$ for the AH mode.

https://doi.org/10.1017/aer.2025.10114 Published online by Cambridge University Press



(a)

(a.1)

(a.2)

Figure 7. *The variations of approximated dynamics vs the true signals in altitude hold mode for 19 random flight conditions at different flight levels.*

These results were obtained for 19 random flight conditions at different altitudes, showing very good dynamics approximations with a minimum number of measured inputs to be integrated into the design of autopilot systems.

To achieve our first control objective, which is to satisfy the tracking performance of both autopilot modes in ideal flight conditions (without turbulence), the following flight scenario was considered:

i. VS mode control: In this mode, the desired vertical speed was set to 9 m/s (≈1800 ft/min) to specify the rate of climb (ROC).

ii. We equalised the desired altitude h_{ref} to $h_{trim} + 1000$ meters (≈3280 ft), where h_{trim} is the aircraft initial altitude at the beginning of simulation which varies at each flight condition. In this scenario, the aircraft started climbing at the commanded vertical speed. As soon as the aircraft arrived at a specified altitude ($h_{ref} \pm h_{trans}$), the transition algorithm automatically engaged the AH mode to capture and maintain the desired altitude h_{ref}.

iii. After 500 secs, we set the desired rate of descent was set to 5 m/s (≈1000 ft/min) and then the desired altitude h_{ref} was changed to $h_{trim} + 300$ meters (≈984 ft), meaning that the aircraft descended by 700 meters (≈ 2296 ft) with respect to the previous reference altitude h_{ref} in the climb manoeuvre.

https://doi.org/10.1017/aer.2025.10114 Published online by Cambridge University Press

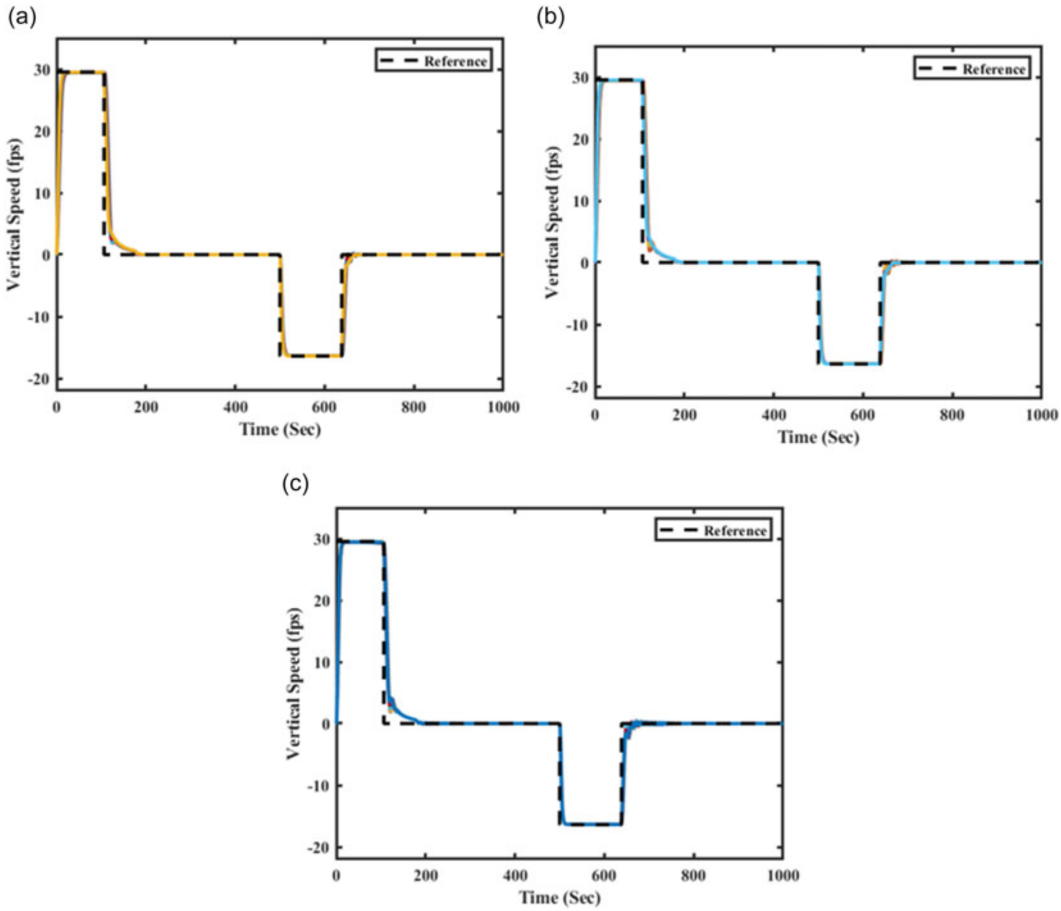


Figure 8. *VS time variations at the altitudes (a) 8000 to 15000 ft, (b) 20000 to 30000 ft and (c) 35000 to 42000 ft in ideal flight conditions (no turbulence) across 925 flight conditions (coloured lines).*

The results obtained for the scenario described above are shown in Figs. 8 to 13. For the VS mode, we have chosen the climbing rate at 9 m/s (≈1800 ft/min). This rate was immediately increased to reach the desired vertical speed. Once the aircraft reached the desired climbing rate, it maintained the desired vertical speed until it reached a distance h_{trans} from the desired altitude that was calculated using Equation (34). At this point, the VS mode control system was deactivated (the VS signal returned to zero). During the climb, the VS mode remained engaged until $t = 106$ sec, and then it was reactivated between $t = 500$ sec and $t = 638$ sec for the descent manoeuvre. The AH mode was engaged outside of these periods during the simulation. As shown in Figs. 8 and 9, at $t = 500$ sec, a descent command was issued, making the AH mode inactive, and the VS started to decrease to 5 m/s (≈1000 ft/min). As the aircraft achieved the desired descent rate, it remained within an interval of $h_{ref} \overset{+}{-} h_{trans}$, where the AH mode was activated. Figures 8 and 9 represent the results for the time variations of the aircraft vertical speed, and the aircraft altitude for flight conditions with altitudes (a) 8000, 10000 and 15000 ft, (b) 20000, 25000 and 30000 ft and (c) 35000, 38000 and 42000 ft.

The initial altitude values at each flight condition were different; therefore, for the sake of clarity in the presentation of our results, we superimposed all altitude signals to start from zero in Fig. 9. In addition, Fig. 9 shows the variations of aircraft altitude according to the performance of the VS mode and the AH mode controllers. The dashed blue lines in Fig. 9 depict the altitude range ($h_{ref} \overset{+}{-} h_{trans}$) where

https://doi.org/10.1017/aer.2025.10114 Published online by Cambridge University Press

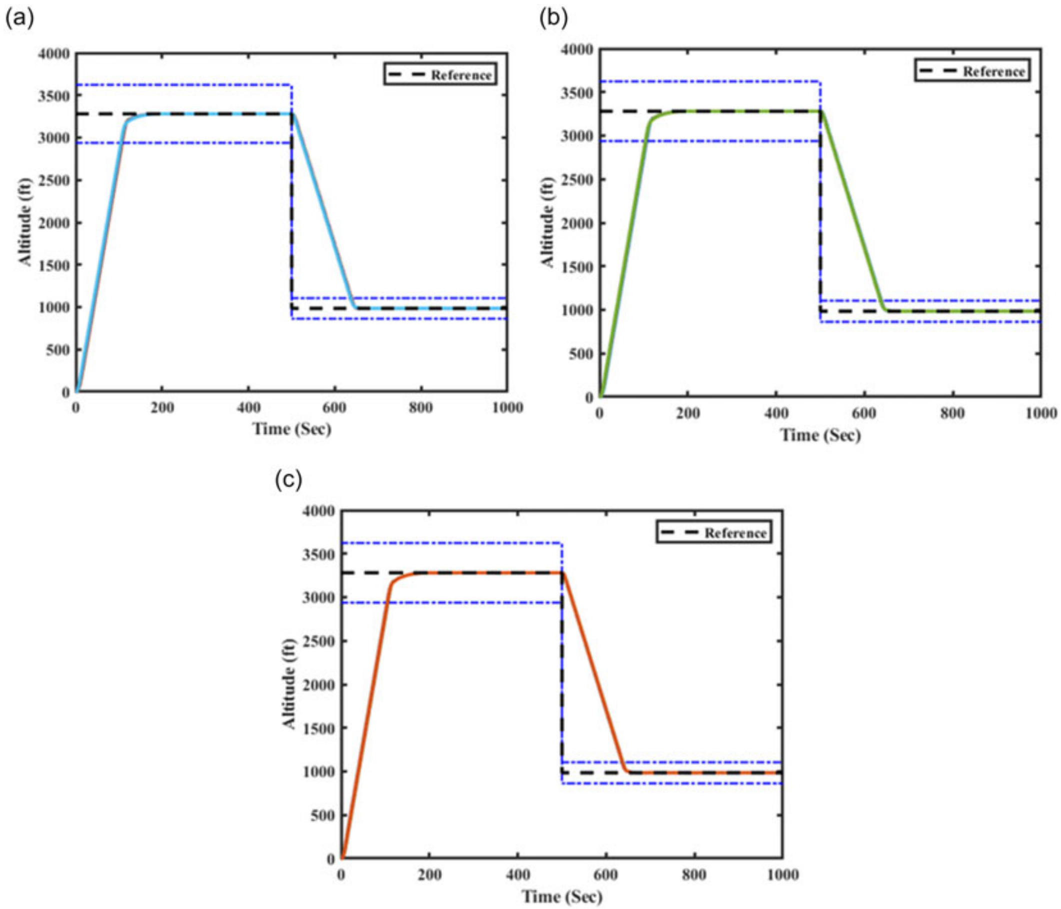


Figure 9. *Altitude time variations at altitudes (a) 8000 to 15000 ft, (b) 20000 to 30000 ft and (c) 35000-to 42000 ft in ideal flight conditions (no turbulence) across 925 flight conditions (coloured lines).*

the AH mode was engaged (between the blue lines); outside of this range, the VS mode control system became engaged. Figure 9 clearly shows how the width of the region varied with respect to the variations of the VS_{ref}.

The results in Figs. 8 and 9 demonstrate the excellent performance of the proposed autopilot system under ideal flight conditions, with no oscillations or overshoots. The variations of the altitude signal also showed that the aircraft successfully captured the commanded altitude h_{ref} using the proposed controllers, and that it was adequately stabilised at the altitude reference signal over the time. A detailed analysis of these results is presented in Fig. 10 to illustrate the performance of the fuzzy logic-based transition algorithm described in Section 2.4.

The procedure implemented by the proposed fuzzy logic-based transition algorithm is illustrated in Fig. 10(a.1), showing the variations of the m_{vs}, while Fig. 10(a.2) illustrates the controller output ($m_{vs}u_{vs}$) for the VS mode control system. The variations for the AH control system are shown in Figs. 10(b.1) and 10(b.2). In these figures, when the VS mode controller or the AH mode controller is engaged, m_{vs} and m_h, respectively, are equal to 1, and when they are not engaged, they decrease to 0. Due to the abrupt variations of m_{vs} and m_h, there are some spikes on the signals in Fig. 8(a.2), e.g., at $t = 500$ sec, and in Fig. 10(b.2) at $t = 638$ sec. To smooth out these signals, we used a filter to produce an input that could be easily handled by the pitch rate control system in the inner loop of the proposed autopilot system

https://doi.org/10.1017/aer.2025.10114 Published online by Cambridge University Press

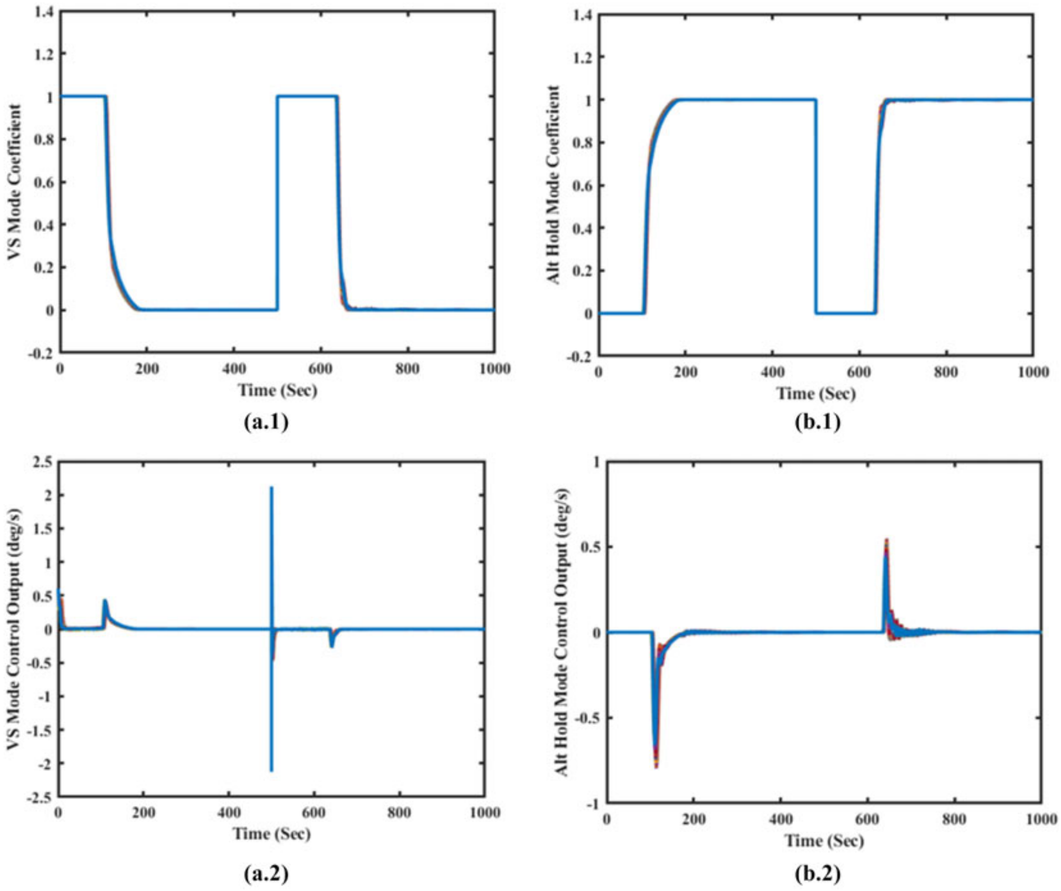


Figure 10. *T variations of the coefficients* $\mathrm{m_{vs}}$ *and* $\mathrm{m_h}$ *((a.1) and (b.2), respectively), and of the signals; (a.2)* $\mathrm{m_{vs}u_{vs}}$ *and (b.2)* $\mathrm{m_h u_h}$*, without turbulence across 925 flight conditions (coloured lines).*

(see Equation (39)). It should be mentioned that the sum of the control signals u_{vs} and u_h generated by $u_{AT} = q_{ref}$, which was calculated in Equation (38), is reference pitch rate signal (deg/s) controlled by the T1AFSMC.

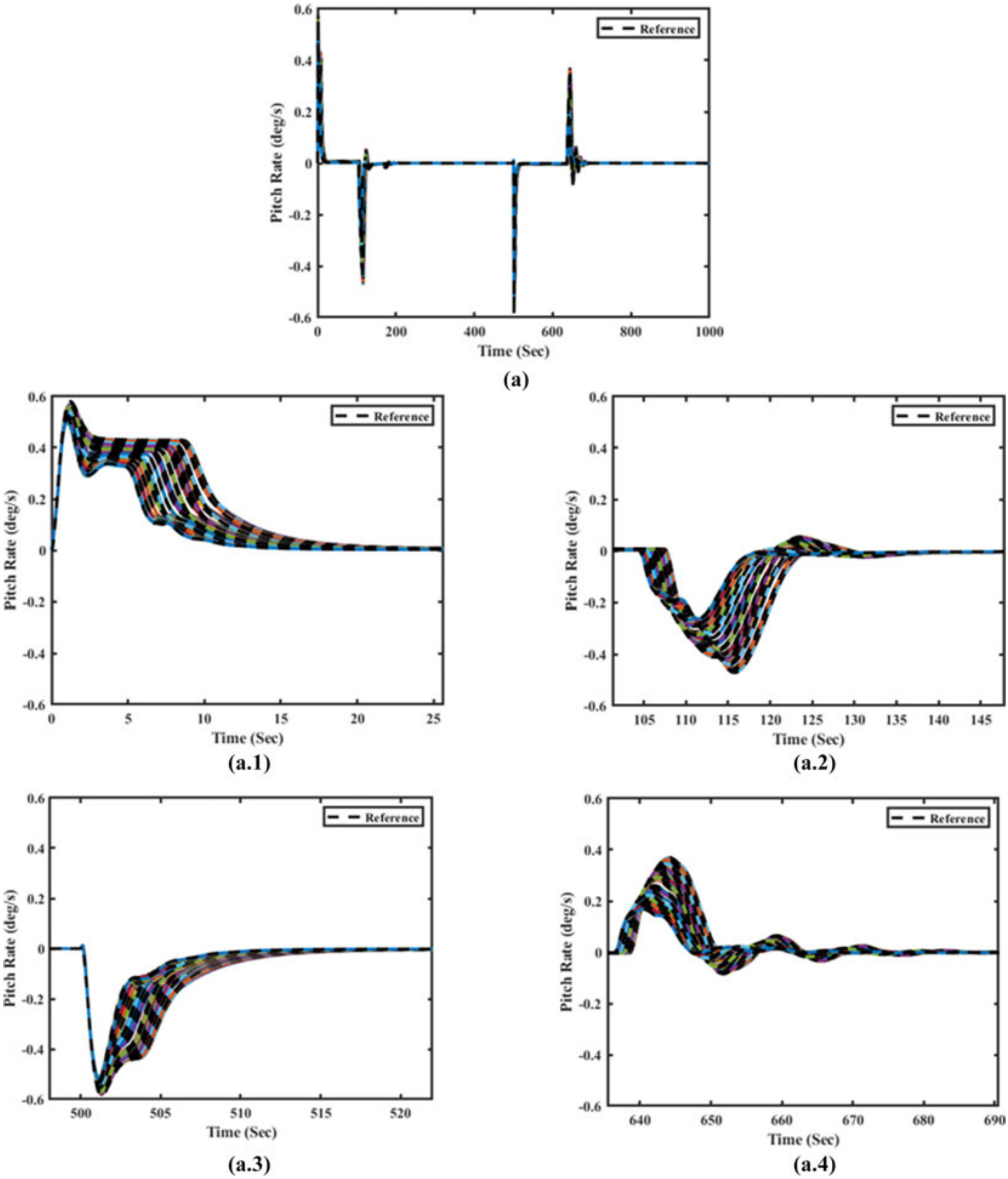
$$\frac{q_{ref}}{r} = \frac{\omega_n^2}{s^2 + 2\xi\omega_n s + \omega_n^2} \tag{39}$$

In Equation (39), $\omega_n = 3$ rad/s is the natural frequency and $\xi = 0.7$ is the damping ratio.

Thus, the obtained pitch rate signal used in the inner loop of the autopilot system is presented in Fig. 11, for altitudes 8000, 10000, 15000 and 20000 ft, and for altitudes 25000, 30000, 35000, 38000 and 42000 ft in Fig. 12.

The pitch rate reference signals shown in Figs. 11 and 12 for all 925 flight conditions were zoomed out in Figs. 11(a.1) to 11(a.4) and Figs. 12(b.1) to 12(b.4) to represent the smoothness of the pitch rate reference signal produced by the filter in Equation (39).

Another goal considered in the design of the autopilot control system was to generate elevator control signals without high-frequency oscillations to avoid mechanical damage to the elevator actuators. This characteristic can be clearly seen in Fig. 13 for the elevator deflections across all 925 flight conditions.



(a)

(a.1)

(a.2)

(a.3)

(a.4)

Figure 11. *Time variations of the pitch rate reference at the altitudes of 8000, 10000, 15000 and 20000 ft without turbulence across 925 flight conditions (coloured lines).*

To evaluate the performance of our autopilot system in critical flight conditions and to provide evidence of the validity of our proposed methodologies, further simulations were performed in the presence of the moderate-intensity turbulence (10^{-3}) during flight.

For this purpose, another flight scenario was defined in which the aircraft was climbing to a specific altitude and then maintained that altitude. The results shown in Figs. 14 and 15 indicated that the VS

https://doi.org/10.1017/aer.2025.10114 Published online by Cambridge University Press

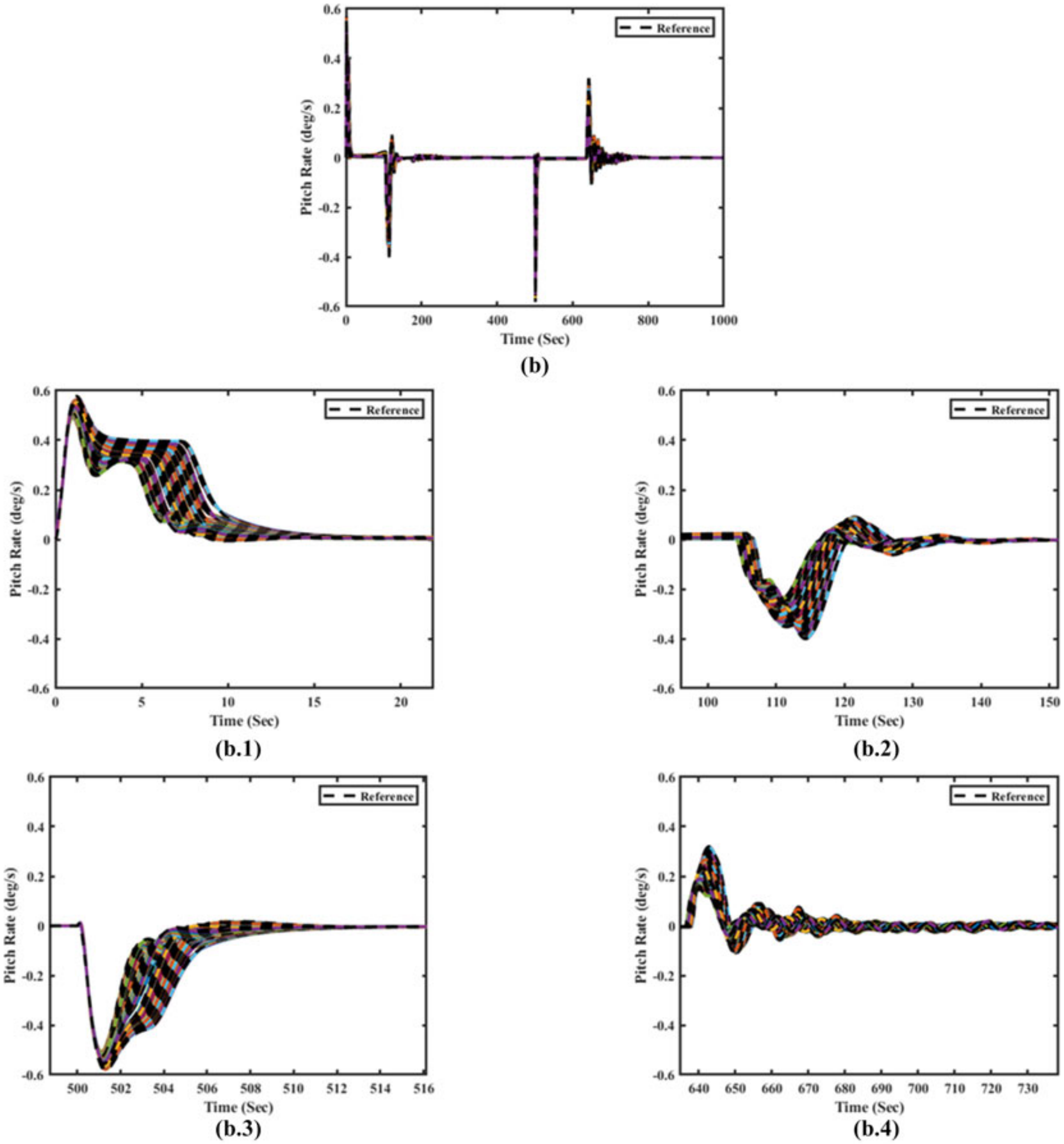
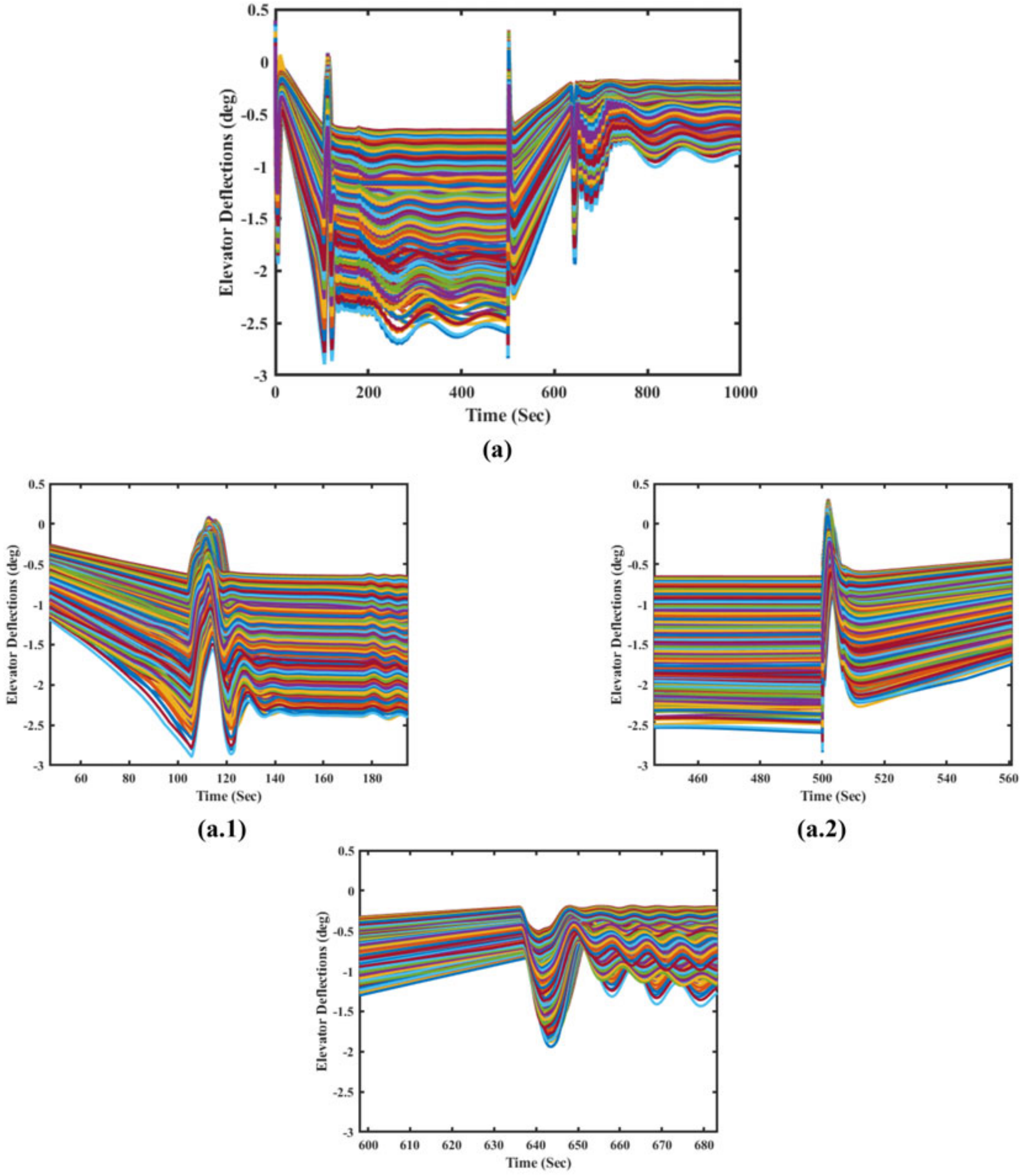


Figure 12. *Time variations of the pitch rate reference at 25000, 30000, 35000, 38000 and 42000 ft without turbulence across 925 flight conditions (coloured lines).*

mode controller remained activated until the aircraft reached the transition altitude h_{trans} represented by the blue line (used to calculate the distance Δh in Equation (34)). At this point, the transition algorithm automatically engaged the AH mode control system to maintain the aircraft at the specified reference altitude, even in the presence of the turbulence effects.

Figure 15 also shows that the VS mode control system could track the given VS reference with a small number of oscillations with the least possible amplitudes, which are acceptable in this research (when the VS mode was engaged between $t = 0$ sec and $t = 105$ sec). It should be noted that the oscillations after the activation of the AH mode control system, due to the absence of the aircraft VS control system. The oscillations from $t = 105$ sec to $t = 300$ sec on the vertical speed signal were induced by the turbulence which is natural and expected as the VS mode control system is not active to directly dampen

https://doi.org/10.1017/aer.2025.10114 Published online by Cambridge University Press



(a)

(a.1)

(a.2)

(a.3)

Figure 13. *Time variations of the elevator deflections without turbulence across 925 flight conditions (coloured lines).*

these fluctuations. Instead in this period, the AH mode was active in the autopilot system and the aircraft altitude was stabilised and controlled with too small oscillations, as shown in Fig. 14.

Accordingly, the control signal generated by the autopilot, which is the aircraft pitch rate reference used in the inner loop, is shown in Fig. 16. This signal varied within a bounded interval in response to the turbulence. The observed oscillations are expected to result from the turbulence-induced effects on the aircraft vertical speed and altitude.

https://doi.org/10.1017/aer.2025.10114 Published online by Cambridge University Press

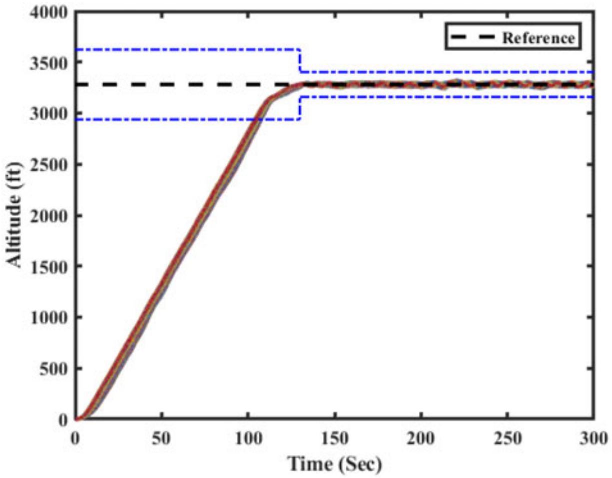


Figure 14. Time variations of the aircraft altitude in turbulent condition across 925 flight conditions (coloured lines).

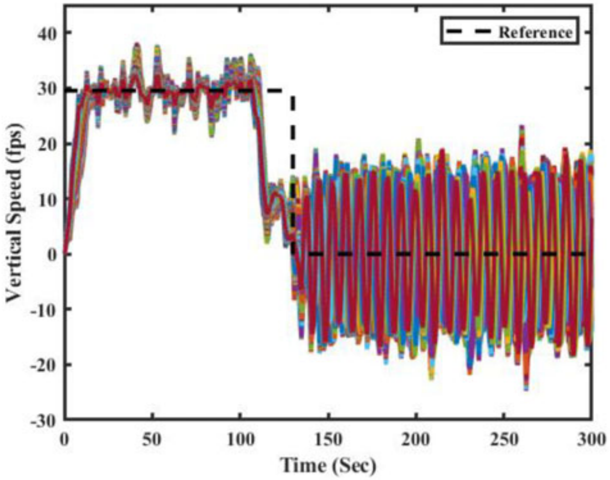


Figure 15. Time variations of the aircraft vertical speed in turbulent condition across 925 flight conditions (coloured lines).

Previously, we analysed the performance of the aircraft autopilot control systems under both turbulent and non-turbulent conditions and assessed the validity of the results. Furthermore, the performances of the VS and AH controllers were also validated in terms of tracking error values, calculated by mean absolute error (MAE) metric with and without turbulence. Table 9 shows the flight condition numbers related to each altitude range at all 925 flight conditions (FC) that were used in Figs. 17 and 18, to examine the precision and the accuracy of the tracking performance of the autopilot system at different altitudes.

Figure 17(a) presents the MAE values for the VS mode calculated with respect to the time variations of the VS signals in Fig. 8. Similarly, Fig. 17(b) presents the MAE values for the AH mode calculated with respect to the time variations of the altitude signals in Fig. 9. In both figures, the black lines depict

https://doi.org/10.1017/aer.2025.10114 Published online by Cambridge University Press

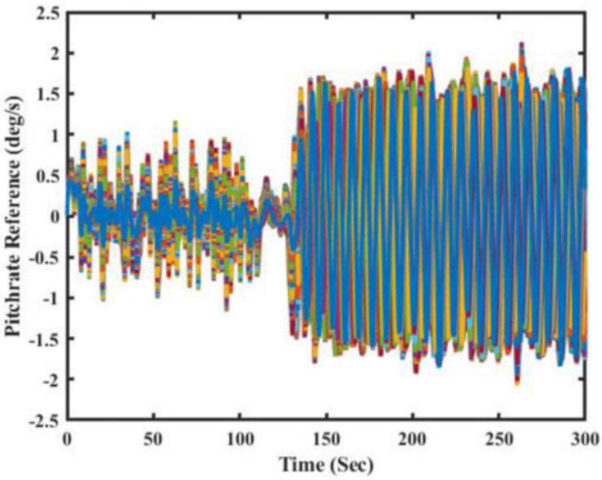


Figure 16. *Time variations of the pitch rate reference across 925 flight conditions (coloured lines) with turbulence.*

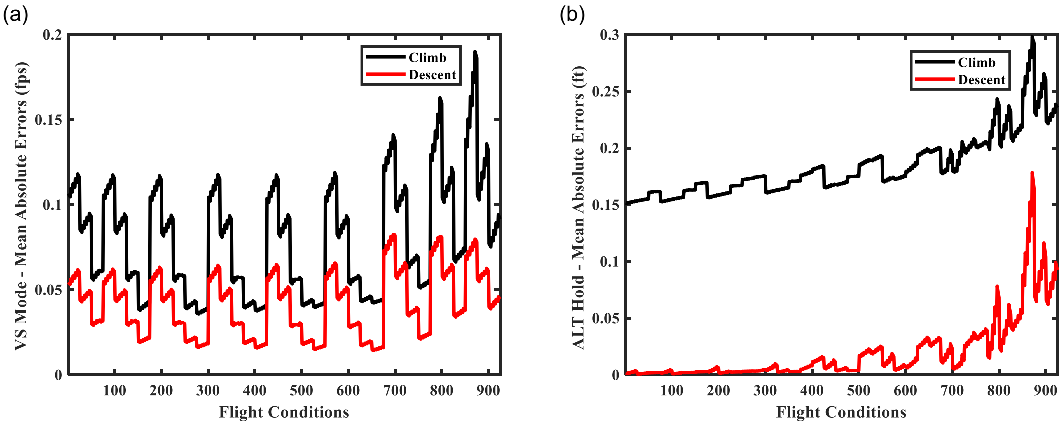


Figure 17. *MAE values at each flight condition for (a) VS mode and (b) AH mode in ideal flight conditions (no turbulence).*

the MAE values during the climb manoeuvre, and the red lines show the MAE values for each flight condition range during the descent.

The MAE values shown in Figs. 17(a) and 17(b) demonstrated that the proposed controllers for both autopilot modes have negligible error values under all flight conditions without turbulence, indicating that the aircraft VS and AH controllers could track very well their commanded reference values.

The MAE values presented in Figs. 18(a) and 18(b) provide more details about the results presented in Figs. 14 and 15. The ranges of the MAE values shown in Fig. 18 for both VS and AH modes are acceptable in the presence of moderate intensity (10^{-3}) turbulence for the scope of this research. In addition, for the VS mode, Fig. 18(a) shows that the developed controller could perform better at higher altitudes than at lower ones, as MAE values decreased as the altitude increased from the first flight condition to the 925th, referring to Table 9.

https://doi.org/10.1017/aer.2025.10114 Published online by Cambridge University Press

Table 9. *Distribution of flight conditions by altitudes authorised in the CCX flight envelope*

F.C No	1–75	76–175	176–300	301–425	426–550	551–675	696–775	776–850	851–925
Altitudes	8000	10000	15000	20000	25000	30000	35000	38000	42000

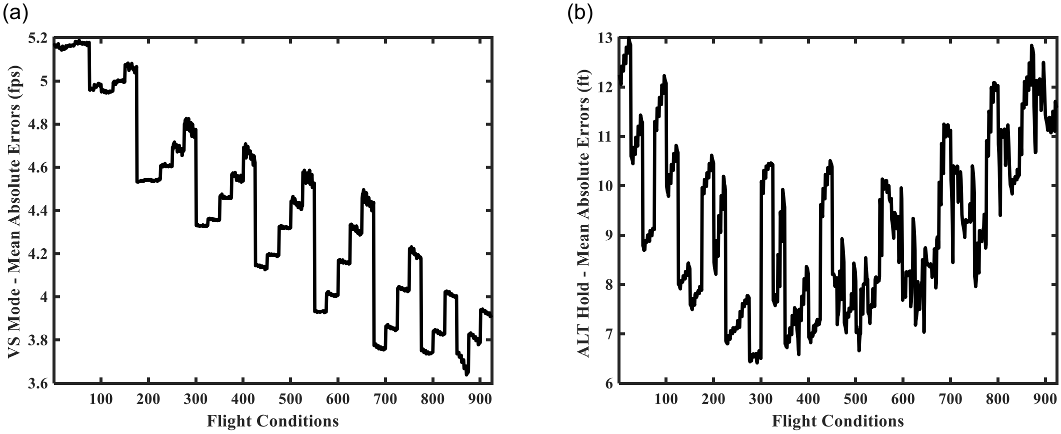


Figure 18. *MAE values at each flight condition for (a) VS mode and (b) AH mode in turbulent condition.*

All the results presented in this section confirm the adaptability and real-time applicability of the proposed autopilot system. The PSO algorithm was executed offline over representative flight conditions spanning the entire CCX flight envelope to initialise all MFRNN weights. During the online learning process, only the last-layer weights were updated using a backpropagation algorithm. This hybrid offline-online learning strategy, combined with the MFRNN internal feedback loops enhanced both the adaptability and corrective responses of the autopilot without any additional computational cost. In addition, the Type-1 adaptive fuzzy sliding-mode control system used in the inner loop further improved the overall robustness and stability of the autopilot system. Furthermore, all hyperparameters and controller gains were kept constants for all 925 flight conditions distributed throughout the CCX flight envelope, both in the presence and absence of turbulence. These results demonstrate that the autopilot system is applicable in real-time across all flight conditions of the CCX aircraft. Hence, this validation process supports the reliability of the proposed methodology.

4.0 Conclusion

This article proposed applying a hybrid control system constructed by use of MFRNNs as approximators for the aircraft dynamics, and a multi-modal sliding mode control system in which each mode was activated using two distinct coefficients. These coefficients values were calculated by a new mode transition algorithm developed based on a fuzzy logic system. This algorithm operated so that when the VS mode was engaged, the AH mode became inactive, and vice versa based on the variations of the altitude tracking error without using a conventional altitude capture mode. As stated earlier, two MFRNNs were developed separately, for approximating the aircraft dynamics in the state-space representations of the VS and AH modes. These approximations were successfully obtained only by using the measured altitude and vertical speed tracking errors and their first-order derivatives even under turbulent flight conditions. These MFRNNs were trained by initialising all weights with the PSO algorithm. Only the weights between the rules layer and the output layer were updated online, using the backpropagation method, while the other weights were fixed at their initialised values. The approximations were obtained without knowledge of the aircraft dynamics mathematics in the state-space representations.

https://doi.org/10.1017/aer.2025.10114 Published online by Cambridge University Press

The Lyapunov theorem was used to prove the stability and boundedness of each controller. In addition, the results showed that the control systems developed for each autopilot mode could meet the desired tracking performance with negligible tracking errors with and without turbulence. In addition, control systems attenuated the drastic effects of moderate-intensity turbulence across the whole flight envelope of the CCX aircraft. This research can be extended in the future by incorporating adaptive fuzzy rules and adaptive membership functions, both of which offer fast convergence with low computational cost while preserving real-time operation. Additional validation scenarios could also be examined such as engine failure, severe turbulence and wind shear, to further evaluate the reliability and robustness of the autopilot system in different critical operating conditions.

Acknowledgements. This research was performed at the Laboratory of Applied Research in Active Controls, Avionics and AeroServoElasticity (LARCASE). This research was realised and funded within the Canada Research Chair in Aircraft Modeling and Simulation New Technologies. For more information related to this research, please visit the LARCASE website at http://larcase.etsmtl.ca. The research aircraft flight simulator (RAFS) for the CCX aircraft was obtained by Ruxandra Botez, full professor, and thanks to research grants from the Canadian Foundation of Innovation and Ministère du Développement Économique, de l'Innovation et de l'Exportation (MDEIE), as well as from CAE, Inc. The authors would like to thank the CAE, Inc., team and Oscar Carranza Moyao for their support in the development of the RAFS at the LARCASE laboratory. The authors also thank Odette Lacasse at the ETS for her support.

References

[1] Rankin, W. MEDA investigation process, Aeromagazine, *ChapterQTR_02,* 2007, (26), pp 15–21. www.boeing.com/commercial/aeromagazine

[2] Ahsan, M. and Hanif, A. A new perspective to altitude acquire-and-hold for fixed wing UAVs, *Student Research Paper Conference 2014*, IST, Pakistan, 2014.

[3] Ahsan, M., Shafique, K., Mansoor, A.B. and Mushtaq, M. Performance comparison of two altitude-control algorithms for a fixed-wing UAV, 2013 3rd IEEE International Conference on Computer, Control and Communication (IC4), 2013, pp 1–5. https://doi.org/10.1109/IC4.2013.6653744

[4] Liu, H., Liu, M., Wei, X., Song, Q.J., Ge, Y.J. and Wang, F.L. Auto altitude holding of quadrotor UAVs with Kalman filter based vertical velocity estimation, Proceedings of the World Congress on Intelligent Control and Automation (WCICA), 2015, pp 4765–4770. https://doi.org/10.1109/WCICA.2014.7053520

[5] Cárdenas, E., Boschetti, P. and Celi, M. (2012). Design of control systems to hold altitude and heading in severe atmospheric disturbances for an unmanned airplane, 2012. https://doi.org/10.2514/6.2012-854

[6] Konar, M. and Hatipoğlu, S.A. Maximisation the autonomous flight performance of unmanned helicopter using BSO algorithm, *Aeronaut. J.*, 2024, **128**, pp 2656–2667. https://doi.org/10.1017/aer.2024.44

[7] Liu, M., Egan, G. and Santoso, F. Modeling, autopilot design, and field tuning of a UAV with minimum control surfaces, IEEE Transactions on Control Systems Technology, 2015, 23, p 1. https://doi.org/10.1109/TCST.2015.2398316

[8] Win, T., Nyunt, H.T.C. and Tun, H.M. Pitch attitude hold autopilot for YTU EC-001 fixed-wing unmanned aerial vehicle, 2019 First International Symposium on Instrumentation, Control, Artificial Intelligence, and Robotics (ICA-SYMP), 2019, pp 78–81. https://doi.org/10.1109/ICA-SYMP.2019.8646286

[9] Qian, L. and Liu, H.H.T. Path-following control of a quadrotor UAV with a cable-suspended payload under wind disturbances, *IEEE Trans. Ind. Electron.*, 2020, **67**, pp 2021–2029. https://doi.org/10.1109/TIE.2019.2905811

[10] Qi, Y., Zhu, Y., Wang, J., Shan, J. and Liu, H.H.T. MUDE-based control of quadrotor for accurate attitude tracking, *Control Eng. Pract.*, 2021, **108**, p 104721. https://doi.org/10.1016/j.conengprac.2020.104721

[11] Feng, C. and Liu, H.H.-T. Collision-free path planning and control of coordinated UAVs with tethered payloads, 2023 IEEE Conference on Control Technology and Applications (CCTA), 2023, pp 243–248. https://doi.org/10.1109/CCTA54093.2023.10253112

[12] Zhao, W., Duan, Y., Yu, Z., Qu, Y. and Zhang, Y. Integrated guidance and control for autonomous rendezvous of unmanned aerial vehicle during aerial refueling, 2021 International Conference on Unmanned Aircraft Systems (ICUAS), 2021, pp 814–820. https://doi.org/10.1109/ICUAS51884.2021.9476886

[13] Zhang, W., Jia, J., Zhou, S., Guo, K., Yu, X. and Zhang, Y. A safety planning and control architecture applied to a quadrotor autopilot, *IEEE Rob. Autom. Lett.*, 2023, **8**, pp 680–687. https://doi.org/10.1109/LRA.2022.3230593

[14] Zareb, E.M., Nouibat, W., Bestaoui, Y., Ayad, R. and Bouzid, Y. Evolutionary autopilot design approach for UAV quadrotor by using GA, *Iran. J. Sci. Technol., Trans. Electric. Engin.*, 2019, **44**, pp 347–375. https://doi.org/10.1007/s40998-019-00214-6

[15] Wilburn, B.K., Perhinschi, M.G. and Wilburn, J.N. A modified genetic algorithm for UAV trajectory tracking control laws optimization, *Int. J. Intell. Unmanne. Syst.*, 2014, **2**, pp 58–90. https://doi.org/10.1108/IJIUS-03-2014-0002

[16] Zhou, R., Wang, T., Yu, Z. and Zhang, Y. Event-triggered finite-time sliding-mode fault-tolerant containment control of multi-uavs with actuator faults, 2023 6th International Symposium on Autonomous Systems (ISAS*)*, 2023, pp 1–6. https://doi.org/10.1109/ISAS59543.2023.10164317

https://doi.org/10.1017/aer.2025.10114 Published online by Cambridge University Press

[17] Konar, M., Hatipoğlu, S.A. and Akpınar, M. Improvement of UAV thrust using the BSO algorithm-based ANFIS model, *Aeronaut. J.*, 2024, **128**, pp 2364–2373. https://doi.org/10.1017/aer.2024.30

[18] Sahbon, N., Jacewicz, M., Lichota, P. and Strzelecka, K. Path-following control for thrust-vectored hypersonic aircraft, *Energies*, 2023, **16**, p 2501. https://doi.org/10.3390/en16052501

[19] Dong, T., Qin, C., Song, S., Wang, X. and Lei, Y. (2022). Autopilot design and analysis of hypersonic aircraft based on fractional-order PIAD control. In Y. Jia, W. Zhang, Y. Fu, Z. Yu, & S. Zheng (Eds.), Proceedings of 2021 Chinese Intelligent Systems Conference, Springer, 2022, pp 184–192. https://doi.org/10.1007/978-981-16-6320-8_19

[20] Gao, H., Chen, Z., Sun, M., Huang, J., Wang, Z. and Chen, Z. An efficient fast altitude control for hypersonic vehicle, *Control Eng. Pract.*, 2020, **100**, pp 104426. https://doi.org/10.1016/j.conengprac.2020.104426

[21] Fenfen, W., Xubo, L., Haiming, T. and Tong, Z. An improved auto-disturbance rejection control method for hypersonic vehicle control system, 2020 Chinese Control And Decision Conference (CCDC), 2020, pp 3410–3415. https://doi.org/10.1109/CCDC49329.2020.9164687

[22] Taha, H., Bayuomi, M., El-Bayoumi, G. and Hassan, S. Model reference adaptive flight path stabilization for altitude and velocity hold, International Conference on Aerospace Sciences and Aviation Technology, 2009, 13(AEROSPACE SCIENCES), pp 1–15. https://doi.org/10.21608/asat.2009.23748

[23] Perez, A.E., Moncayo, H., Perhinschi, M., Al Azzawi, D. and Togayev, A. A bio-inspired adaptive control compensation system for an aircraft outside bounds of nominal design, *J Dyn Syst Meas Control*, 2015, **137**. https://doi.org/10.1115/1.4030613

[24] Perez, A.E., Moncayo, H., Moguel, I., Perhinschi, M.G., Al Azzawi, D. and Togayev, A. Development of immunity based adaptive control laws for aircraft fault tolerance, *ASME 2014 Dynamic Systems and Control Conference*, 2014, pp 1–10. https://doi.org/10.1115/DSCC2014-5890

[25] Park, J.-C., Roh, H., Lee, C.-H., Tahk, M.-J., Choi, H.-S. and Kim, S.-Y. Nonlinear three-loop autopilot design for hypersonic vehicles considering aerodynamic-propulsion dynamics couplings, 2023 14th International Conference on Mechanical and Aerospace Engineering (ICMAE), 2023, pp 404–410. https://doi.org/10.1109/ICMAE59650.2023.10424637

[26] Hiliuta, A., Botez, R. and Brenner, M. Approximation of unsteady aerodynamic forcesby use of fuzzy techniques, *IASTED Modeling, Identification and Control, Innsbruck, Austria*, 2005a.

[27] Hiliuta, A., Botez, R.M. and Brenner, M. Approximation of unsteady aerodynamic forces Q(k,M) by use of fuzzy techniques, *AIAA J.*, 2005b, **43**, pp 2093–2099. https://doi.org/10.2514/1.13600

[28] Theis, J., Ossmann, D., Thielecke, F. and Pfifer, H. Robust autopilot design for landing a large civil aircraft in crosswind, *Control Eng. Pract.*, 2018, **76**, pp 54–64. https://doi.org/10.1016/j.conengprac.2018.04.010

[29] Qiu, J., Delshad, I.S., Zhu, Q., Nibouche, M. and Yao, Y. A U-model based controller design for non-minimum phase systems: application to Boeing 747 altitude-hold autopilot, 2017 9th International Conference on Modelling, Identification and Control (ICMIC), 2017, pp 122–127. https://doi.org/10.1109/ICMIC.2017.8321624

[30] Santos, S.R.B. dos and Oliveira, N.M.F.D. Longitudinal autopilot controllers test platform hardware in the loop, *2011 IEEE International Systems Conference*, 2011, pp 379–386. https://doi.org/10.1109/SYSCON.2011.5929071

[31] Islam, M.T., Alam, M.S., Laskar, M.A.R. and Garg, A. Modeling and simulation of longitudinal autopilot for general aviation aircraft, 2016 5th International Conference on Informatics, Electronics and Vision (ICIEV), 2016, pp 490–495. https://doi.org/10.1109/ICIEV.2016.7760051

[32] Raj, K.D.S. and Kumar, P.R. Design and simulation of longitudinal autopilot modes for a conventional aircraft, 2015 2nd International Conference on Electronics and Communication Systems (ICECS), 2015, pp 1040–1046. https://doi.org/10.1109/ECS.2015.7124738

[33] Nivison, S.A. and Khargonekar, P.P. Development of a robust deep recurrent neural network controller for flight applications, 2017 American Control Conference (ACC), 2017, pp 5336–5342. https://doi.org/10.23919/ACC.2017.7963784

[34] Juang, J.-G. and Chiou, H.-K. Hybrid RNN-GA controller for ALS in wind shear condition. *2006 IEEE International Conference on Systems, Man and Cybernetics*, 2006, pp 675–680. https://doi.org/10.1109/ICSMC.2006.384463

[35] Vandana, A., Chaturvedi, I., Jaikumar, S. and Chandar, T.S. Neural network based robust integrated autopilot for an aircraft, 2021 IEEE 6th International Conference on Computing, Communication and Automation (ICCCA), 2021, pp 231–236. https://doi.org/10.1109/ICCCA52192.2021.9666325

[36] Ghazi, G. and Botez, R.M. Lateral controller design for the Cessna citation X with handling qualities and robustness requirements, *62nd CASI Aeronautics Conference and AGM*, Montreal, Quebec, Canada, 2015.

[37] Boughari, Y., Ghazi, G. and Theel, F. New methodology for optimal flight control using differential evolution algorithms applied on the Cessna Citation X business aircraft – Part 2. Validation on aircraft research flight level D simulator, *INCAS BULLETIN*, 2017, **9**, pp 45–59. https://doi.org/10.13111/2066-8201.2017.9.2.4

[38] Ghazi, G., Rezk, K.G. and Botez, R.M. Cessna citation X pitch rate control design using guardian maps. *International Conference on Air Transport INAIR : Proceedings, Amsterdam, Netherlands*, 2016.

[39] Quintin, E., Andrianantara, R.P., Ghazi, G. and Botez, R.M. Neural network adaptive controller with approximate dynamic inversion for the Cessna Citation X lateral control, AIAA SCITECH 2024 Forum. Orlando, FL, 2024. https://doi.org/10.2514/6.2024-0261

[40] Andrianantara, R.P., Ghazi, G. and Botez, R.M. Model predictive controller with adaptive neural networks and online state estimation for pitch rate control of the Cessna Citation X, *AIAA SCITECH 2024 Forum*, Orlando, FL, 2024. https://doi.org/10.2514/6.2024-0118

[41] Hosseini, S.M., Ghazi, G. and Botez, R.M. New type-2-fuzzy-logic-based control system for the Cessna Citation X, *J. Aerosp. Inf. Syst.*, 2024a, **21**, pp 846–864. https://doi.org/10.2514/1.I011398

[42] Hosseini, S., Inga, C., Ghazi, G. and Botez, R.M. Model-referenced adaptive flight controller based on recurrent neural network for the longitudinal motion of Cessna Citation X, AIAA AVIATION 2023 Forum, San Diego, CA and Online, 2023. https://doi.org/10.2514/6.2023-3797

[43] Hosseini, S.M., Bematol, I., Ghazi, G. and Botez, R.M. Enhanced fuzzy-based super-twisting sliding-mode control system for the Cessna Citation X lateral motion, *Aerospace*, 2024, **11**, p 549. https://doi.org/10.3390/aerospace11070549

[44] Fei, J., Wang, Z., Liang, X., Feng, Z. and Xue, Y. Fractional sliding-mode control for microgyroscope based on multilayer recurrent fuzzy neural network, *IEEE Trans. Fuzzy Syst.*, 2022, **30**, pp 1712–1721. https://doi.org/10.1109/TFUZZ.2021.3064704

[45] Ghazi, G. (2014). Développement d'une plateforme de simulation et d'un pilote automatique—Application aux Cessna Citation X et Hawker 800XP [Masters, École Polytechnique de Montréal]. https://publications.polymtl.ca/1535/

[46] Hosseini, S.M., Ghazi, G. and Botez, R.M. Novel controller methodology for the Cessna Citation X under turbulence during cruise, *J. Aerosp. Inf. Syst.*, 2024b, **21**, pp 892–905. https://doi.org/10.2514/1.I011374

[47] Baomar, H. and Bentley, P.J. Autonomous flight cycles and extreme landings of airliners beyond the current limits and capabilities using artificial neural networks, *Appl. Intell.*, 2021, **51**, pp 6349–6375. https://doi.org/10.1007/s10489-021-02202-y

[48] Liu, F., Dai, S. and Zhao, Y. Learning to have a civil aircraft take off under crosswind conditions by reinforcement learning with multimodal data and preprocessing data, *Sensors*, 2021, **21**, pp 1386. https://doi.org/10.3390/s21041386

[49] Yedavalli, R.K. Title: "Flight dynamics and control of aero and space vehicles" by Rama K. YedavalliTextbook Authorship Published By Wiley in 2020 : Aerospace Series; ISBN 9781118934425, 2022.

[50] Moorhouse, D.J. and Woodcock, R.J. Background information and user guide for MIL-F-8785C, military specification—flying qualities of piloted airplanes, 1982. https://apps.dtic.mil/sti/citations/ADA119421

[51] Gad, A.G. Particle swarm optimization algorithm and its applications: a systematic review, *Arch. Comput. Methods Eng.*, 2022, **29**, pp 2531–2561. https://doi.org/10.1007/s11831-021-09694-4

[52] Khalil, H. *Nonlinear Systems* (3rd edition), Prentice Hall, Upper Saddle River, 2002.

[53] Slotine, J.-J.E. and Li, W. *Applied Nonlinear Control*, Prentice Hall International Inc., Upper Saddle River, New Jersey, 1991.

[54] Wang, C.-H., Liu, H.-L. and Lin, T.-C. Direct adaptive fuzzy-neural control with state observer and supervisory controller for unknown nonlinear dynamical systems, *Fuzzy Syst., IEEE Trans. On*, 2002, **10**, pp 39–49. https://doi.org/10.1109/91.983277

[55] Wang, L.-X. A course in fuzzy systems and control (Vol. 1–1 online resource (xviii, 424 pages): illustrations). Prentice Hall PTR, 1997. http://books.google.com/books?id=wbJQAAAAMAAJ

[56] Apostol, T.M. *Mathematical Analysis* (2nd edition), Addison-Wesley, Reading, Mass., 1974.

[57] Mitrinović, D.S., Pečarić, J.E. and Fink, A.M. Triangle inequalities. In *Classical and New Inequalities in Analysis*, vol. **61**, Springer, Dordrecht, 1993, pp. 473–513.

Cite this article: Hosseini S.M., Ghazi G. and Botez R.M. New fuzzy-RNN autopilot system for the Cessna Citation X aircraft with a fuzzy transition algorithm. *The Aeronautical Journal*, https://doi.org/10.1017/aer.2025.10114