Integrated guidance and control for UAV path following in Frenet–Serret frame

Nguyen Hoang Viet[1], Nguyen Quang Vinh[1*], Nguyen Vu[1],
Nguyen Thu Trang[1], Nguyen Bich Van[2]

[1]Academy of Military Science and Technology, 17 Hoang Sam, Nghia Do, Hanoi, Vietnam;
[2]Institute for Artificial Intelligence, University of Engineering and Technology, 144 Xuan Thuy, Cau Giay, Hanoi, Vietnam.
[*]Corresponding author: vinhquang2808@gmail.com
Received 09 Jan. 2026; Revised 26 Feb. 2026; Accepted 26 May 2026; Published 25 Jun. 2026.
DOI: https://doi.org/10.54939/1859-1043.j.mst.112.2026.3-10

ABSTRACT

This paper proposes a high-precision path following method for unmanned aerial vehicles (UAVs) using the Frenet–Serret (FS) coordinate frame combined with the optimal Integrated Guidance and Control (IGC) algorithm. To increase adaptability and reduce tracking error in an uncertain environment, a neural network coupled model predictive controller (NNMPC) is directly integrated into the IGC framework. The stability analysis of the system is performed using the Lyapunov–Krasovskii criterion, providing a sufficient condition for asymptotic stability under bounded time delay and neural network approximation errors. The simulation results for a quadrotor UAV following a sinusoidal trajectory demonstrate that the proposed method outperforms traditional PID control in terms of accuracy and disturbance rejection capability. Furthermore, this study discusses practical implementation constraints, including computational complexity and data availability, thereby bridging the gap between theoretical design and real-time UAV applications.

Keywords: Orbital tracking; Frenet–Serret; Integrated guidance and control; Lyapunov–Krasovskii.

1. INTRODUCTION

Modern UAV systems require high accuracy in path following, especially in uncertain and noisy environments. The traditional method of separating the three layers of guidance, navigation, and control often results in errors due to accumulated delays and noise. The integration of IGC has been shown to reduce this error [2].

The FS coordinate frame has been widely used in robotics and aviation because it allows for describing 3D curved trajectories naturally. However, studies combining FS with optimal IGC and rigorous stability analysis using Lyapunov–Krasovskii are still limited. In recent years, there have been many studies on UAV path following using different methods: Yang et al. [1] proposed a tracking algorithm for UAVs using the FS framework combined with linear model predictive control (LMPC), but did not consider the adaptability and stability in a strong noise environment. Anderson et al. [2] presented an integrated IGC framework for tactical missiles, which reduces the delay between guidance and control. However, this method is mainly applied to linear trajectories. Zhang et al. [3] developed an NNMPC controller for a quadrotor, which significantly improved the tracking ability in windy environments. However, this study did not incorporate the FS framework. Chen et al. [4] applied the Lyapunov–Krasovskii criterion to ensure the stability of UAV systems with communication delays, but did not consider path following in 3D curved space.

Moreover, most existing studies neglect practical constraints such as computational complexity, solver efficiency, and real-time implementability on embedded UAV hardware.

The current research gap lies in the absence of an adaptive, optimal, and provably stable UAV path-following framework in the FS coordinate system that simultaneously addresses theoretical

stability and practical feasibility. This paper aims to fill this gap by proposing an FS-based IGC framework integrated with NNMPC and validated through stability analysis and extensive simulations. This work further addresses the gap by incorporating implementation-oriented analysis and discussing feasible solutions for real-time deployment.

The main contributions of this paper are summarized as follows: Development of an FS-based integrated guidance and control framework for UAV path following; Integration of neural-network-based model predictive control into the IGC architecture; Lyapunov--Krasovskii-based stability analysis considering time delay and disturbances. Extension of simulation scenarios to complex three-dimensional trajectories and analysis of computational and data-related limitations.

2. INTEGRATED GUIDANCE CONTROL WITH NEURAL NETWORK MPC

In advanced trajectory tracking and guidance problems for unmanned aerial vehicles (UAVs), the geometric description of motion plays a critical role in controller design. Instead of describing the vehicle motion in inertial or body-fixed coordinates, this study employs the Frenet-Serret (FS) frame, which is intrinsically attached to the reference trajectory.

The FS frame is defined by three orthonormal vectors: the tangent vector $\mathbf{T}$, the normal vector $\mathbf{N}$, and the binormal vector $\mathbf{B}$. These vectors characterize the instantaneous direction of motion and the local curvature properties of the trajectory. Let s denote the arc length along the reference path. The evolution of the FS frame is governed by the classical Frenet-Serret equations [5]

$$\frac{d}{ds}\begin{bmatrix}\mathbf{t}\\ \mathbf{n}\\ \mathbf{b}\end{bmatrix}=\begin{bmatrix}0 & \kappa & 0\\ -\kappa & 0 & \tau\\ 0 & -\tau & 0\end{bmatrix}\begin{bmatrix}\mathbf{t}\\ \mathbf{n}\\ \mathbf{b}\end{bmatrix}, \tag{1}$$

where V denotes the UAV speed, κ is the curvature, and τ is the torsion of the reference trajectory. These geometric quantities explicitly describe the bending and twisting characteristics of the path in three-dimensional space.

By projecting the UAV dynamics into the FS frame, the motion equations can be naturally decomposed along the $\mathbf{T}$, $\mathbf{N}$, and $\mathbf{B}$ directions. This representation allows the control forces to be directly associated with the trajectory-following task, rather than relying on inertial or body-fixed coordinates.

Assuming that the UAV is sufficiently close to the reference trajectory, the translational dynamics expressed in the FS frame can be written as [5]:

$$\begin{bmatrix}\dot{v}_T\\ \dot{v}_N\\ \dot{v}_B\end{bmatrix}=\begin{bmatrix}0 & \kappa v & 0\\ -\kappa v & 0 & \tau v\\ 0 & -\tau v & 0\end{bmatrix}\begin{bmatrix}v_T\\ v_N\\ v_B\end{bmatrix}+\begin{bmatrix}F_T/m\\ F_N/m\\ F_B/m\end{bmatrix}+\begin{bmatrix}d_T\\ d_N\\ d_B\end{bmatrix} \tag{2}$$

where V_T, V_N, and V_B denote the velocity components along the tangent, normal, and binormal directions, respectively; u_T, u_N, and u_B are the corresponding control force components; and $f_T(\cdot)$, $f_N(\cdot)$, and $f_B(\cdot)$ represent nonlinear aerodynamic and coupling effects.

This formulation enables a direct link between control inputs and trajectory-shape regulation. In particular, lateral and vertical deviations are primarily influenced by the normal and binormal force components, which simplifies the integrated guidance and control (IGC) design.

The proposed IGC framework combines guidance, state estimation, and control into a unified structure. Unlike conventional cascade architectures, where guidance and control loops are designed separately, the proposed approach embeds guidance objectives directly into the control optimization problem: State estimation - A Square-Root Adaptive Kalman Filter with Robust

Adjustment (SH-RAKF) is employed to estimate the UAV states in the presence of sensor noise and modeling uncertainties; Guidance - The UAV position is projected onto the FS reference trajectory to obtain the desired states in FS coordinates; Control - A Neural Network Model Predictive Control algorithm generates optimal control inputs subject to system constraints.

This unified structure eliminates the need for an explicit guidance law and ensures coordinated trajectory tracking and control execution.

The reference state in FS coordinates is denoted by:

$$\mathbf{x}_r = \left[V_{T,r}, V_{N,r}, V_{B,r} \right]^\top \tag{3}$$

which encodes both trajectory-following and motion regulation objectives.

For real-time implementation, the continuous FS dynamics are discretized and linearized around the current operating point. The resulting discrete-time model is expressed as:

$$\mathbf{x}_{k+1} = \mathbf{A}_k \mathbf{x}_k + \mathbf{B}_k \mathbf{u}_k + \mathbf{d}_k \tag{4}$$

where $\mathbf{x}_k = [V_T, V_N, V_B]^\top$ is the state vector in FS coordinates, $\mathbf{u}_k = [u_T, u_N, u_B]^\top$ is the control input vector, and $\mathbf{d}_k$ represents lumped disturbances and unmodeled dynamics.

The matrices $\mathbf{A}_k$ and $\mathbf{B}_k$ vary with flight conditions and trajectory curvature, motivating the use of adaptive modeling techniques.

This model serves as the prediction model for the MPC optimization problem.

Due to aerodynamic uncertainties and unmodeled dynamics, the matrices $\mathbf{A}_k$ and $\mathbf{B}_k$ may vary significantly during flight. To address this issue, a neural network is employed to adaptively tune the model parameters online.

Specifically, the neural network approximates the nonlinear mapping

$$\left(\mathbf{A}_k, \mathbf{B}_k, \mathbf{d}_k \right) = \mathcal{N}\left(\mathbf{x}_k, \mathbf{u}_k \right) \tag{5}$$

where $\mathcal{N}(\cdot)$ denotes the neural network function. This adaptive mechanism enhances prediction accuracy and improves closed-loop performance under varying flight conditions.

The neural network in (5) provides online estimates of the model parameter variations ΔAk and ΔBk. These estimates are used to update the prediction model in (4) as $Ak = A\text{nom} + \Delta Ak$ and $Bk = B\text{nom} + \Delta Bk$. The updated model is then embedded into the MPC optimization problem (6)-(7), ensuring that the predictions reflect the true system dynamics more accurately.

At each sampling instant, the NNMPC solves the following finite-horizon optimization problem:

$$\min_{\mathbf{u}} J = \sum_{i=0}^{N_p} \left\| \mathbf{x}_{k+i} - \mathbf{x}_{r,k+i} \right\|_{\mathbf{Q}}^2 + \sum_{i=0}^{N_c-1} \left\| \mathbf{u}_{k+i} \right\|_{\mathbf{R}}^2, \tag{6}$$

where $\mathbf{x}(k+i \mid k)$ denotes the predicted state at time $k+i$ based on information available at time k, and $\mathbf{u}(k+i \mid k)$ is the control input at time $k+i$ predicted at time k. The prediction horizon Np and control horizon Nc satisfy $Nc \leq Np$.

Subject to:

$$\begin{aligned} &\mathbf{x}_{k+i+1} = \mathbf{A}_{k+i} \mathbf{x}_{k+i} + \mathbf{B}_{k+i} \mathbf{u}_{k+i} \\ &\mathbf{x}_{k+i} \in \mathcal{X}, \quad \mathbf{u}_{k+i} \in \mathcal{U} \end{aligned} \tag{7}$$

where N_p and N_c are the prediction and control horizons, respectively; $\mathbf{Q}$ and $\mathbf{R}$ are positive-definite weighting matrices; and $\mathcal{X}$, $\mathcal{U}$ denote state and input constraint sets. Only the first control input is applied to the UAV, and the optimization is repeated in a receding horizon manner.

Define the tracking error as: $\mathbf{e}_k = \mathbf{x}_k - \mathbf{x}_{\text{ref},k}$ and consider the Lyapunov-Krasovskii candidate function:

$$V_k = \mathbf{e}_k^T P \mathbf{e}_k + \sum_{j=k-\tau}^{k-1} \mathbf{e}_j^T Q \mathbf{e}_j \tag{8}$$

where $\mathbf{P}$ and $\mathbf{Q}$ are positive-definite matrices. The closed-loop system is asymptotically stable if there exist positive-definite matrices $\mathbf{P}$ and $\mathbf{Q}$ such that the difference $\Delta V_k = V_{k+1} - V_k$ satisfies:

$$\Delta V_k = \mathbf{e}_{k+1}^T P \mathbf{e}_{k+1} - \mathbf{e}_k^T P \mathbf{e}_k + \mathbf{e}_k^T Q \mathbf{e}_k - \mathbf{e}_{k-\tau}^T Q \mathbf{e}_{k-\tau} < 0 \tag{9}$$

Under bounded time delay τ and neural network approximation errors, condition (10) guarantees that $\| \mathbf{e}_k \|$ converges to a bounded region, i.e., the system is uniformly ultimately bounded (UUB).

Due to the nonlinear and adaptive nature of the NNMPC, establishing asymptotic stability in the classical Lyapunov sense is challenging. Instead, we analyze the system using the Input-to-State Stability (ISS) framework. The tracking error $\mathbf{e}_k$ is shown to be uniformly ultimately bounded (UUB) with the ultimate bound depending on the neural network approximation error and external disturbances. Specifically, there exists a class $\mathcal{KL}$ function β and a class $\mathcal{K}$ function γ such that:

$$\| \mathbf{e}_k \| \leq \beta(\| \mathbf{e}_0 \|, k) + \gamma(\| \mathbf{w} \|_\infty) \tag{10}$$

where $\mathbf{w}$ represents the lumped disturbances and approximation errors.

Algorithm 1: *IGC-NNMPC Algorithm*

```
Initialize x0, Q, R, N, Nc
for k = 0,1,2,... do
    Obtain measurement yk
    Estimate state using SH-RAKF:
      x̂k = FSH-RAKF(yk)
    Generate FS reference trajectory:
      xr,k = [VT,r, VN,r, VB,r, ψr, θr]T
    Update NN model:
      (Ak,Bk,dk)=NN(xk,uk-1)
    Predict system dynamics:
      xk+1 = Akxk + Bkuk + dk
    Solve NMPC optimization:
      uk* = argmin J
    Apply optimal control:
      uk = uk*
    if mission completed then
        break
    end if
end for
```

The proposed IGC framework based on NNMPC combines geometric modeling, learning-based prediction, and optimization-based control. Compared with conventional cascade architectures, the proposed approach achieves improved tracking accuracy, robustness, and constraint handling capability.

From a practical perspective, the FS-based reduced-order model significantly lowers

computational complexity, making real-time onboard implementation feasible. Furthermore, the modular structure allows seamless integration with existing navigation and estimation subsystems.

Algorithm 1 illustrates the complete algorithmic flow of the proposed integrated guidance and control framework. The guidance, state estimation, learning-based model adaptation, and predictive control are executed in a unified closed-loop structure.

Although the NNMPC-based IGC framework improves tracking performance, it introduces additional computational burden due to the optimization solver fmincon and neural network inference. To enhance real-time feasibility, lightweight solvers, model reduction techniques, and simplified neural architectures can be adopted, enabling deployment on embedded processors such as ARM-based UAV flight controllers.

3. RESULTS AND DISCUSSION

3.1. Simulation setups

In this study, we implemented and compared two control methods for an unmanned aerial vehicle (UAV) following a sinusoidal reference trajectory: integrated guidance-navigation-control and model-predictive control (IGC-NNMPC), and proportional-integral-derivative (PID) control. The reference trajectory is defined as $x(t) = 50\sin(0.5t)$ (m), with an amplitude of 50 m, an angular frequency of 0.5 rad/s, and a fixed altitude of 10 m. Gaussian noise with a standard deviation of 0.5 m/s^2 is added to the acceleration to simulate real-world factors such as wind or sensor errors. The simulation results are analyzed through evaluation graphs and quantitative data. The simulations are performed on MATLAB R2021b with typical 6DOF UAV parameters: mass m = 2.5 kg, inertia tensor $\mathbf{J} = \text{diag}(0.1, 0.1, 0.2)$ $kg.m^2$

All controllers are tuned to achieve the best possible performance. The PID parameters $K_p = 1.5$; $K_i = 0.5$; $K_d = 1.5$ were obtained via extensive trial-and-error to minimize tracking error.

To evaluate the generalization capability of the proposed method, additional trajectory scenarios were considered, including figure-eight trajectories and sudden altitude changes, representing more challenging three-dimensional motion patterns. Furthermore, the neural network was trained using simulated data; however, practical deployment requires incorporating real flight data to mitigate overfitting and improve model reliability.

3.2. Simulation results

The IGC-NNMPC method uses an MLP neural network with 20 hidden neurons to predict the dynamics and optimize the control input using the *fmincon* algorithm with a 10-step prediction horizon. The results show that the IGC-NNMPC outperforms the PID. The 3D trajectory plot (Figure 1) illustrates that the actual trajectory of the IGC-NNMPC closely follows the reference trajectory with a maximum spatial error of about 1.5 m.

The trajectory error standard, calculated by $\sqrt{\| \mathbf{e}(t) \|^2}$, ranges from 0.5-1.5 m, with a mean of 0.8 m and a standard deviation of 0.3 m. In contrast, the PID controller with parameters $K_p = [10,10,10]$, $K_i = [0.5,0.5,0.5]$, and $K_d = [5,5,5]$ has a larger trajectory error, reaching a maximum of 5 m at the curve points of the sinusoidal trajectory. The standard error of the PID ranges from 1.5-3.5 m, with a mean of 2.2 m and a standard deviation of 0.7 m, which is 175% and 133% higher than the IGC-NNMPC, respectively.

Table 1 summarizes the quantitative comparison among the four controllers. IGC-MPC achieves better performance than PID but falls short of IGC-NNMPC, confirming the contribution of neural network adaptation. PID+FS shows slight improvement over PID, indicating that FS

reference generation alone provides some benefit but is insufficient without advanced control.

Table 1. *Quantitative comparison of tracking performance.*

Controller	Mean error (m)	Std dev (m)	Max error (m)
PID	2.2	0.7	5.0
PID+FS	1.9	0.6	4.0
IGC-MPC	1.2	0.4	2.5
IGC-NNMPC	0.8	0.3	1.5

Gaussian noise with a standard deviation of 0.5 m/s^2 is applied to both methods to evaluate the robustness. The IGC-NNMPC demonstrates superior noise handling capabilities through neural network dynamics prediction and predictive control optimization.

The trajectory errors along each axis (e_x, e_y, e_z) of IGC-NNMPC fluctuate within ± 1 m, with e_z being close to zero due to the fixed reference height (Figure 2). Meanwhile, the PID is strongly affected by noise, resulting in the e_y error peaking at $\pm 3\ m$ and e_z fluctuating at $\pm 0.5\ m$. The standard deviation of the PID error (0.7 m) is 2.3 times higher than that of IGC-NNMPC (0.3 m), confirming the better adaptability of IGC-NNMPC in noisy environments.

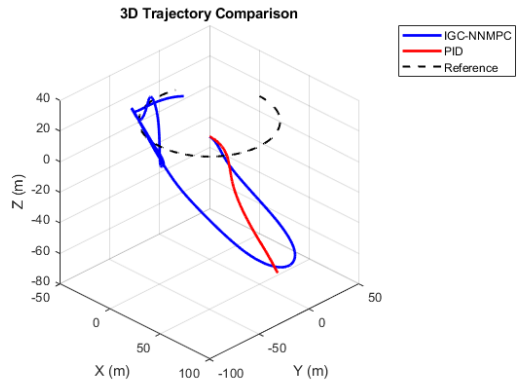

Figure 1. *Comparison of 3D trajectories of IGC-NNMPC, PID and reference trajectories.*

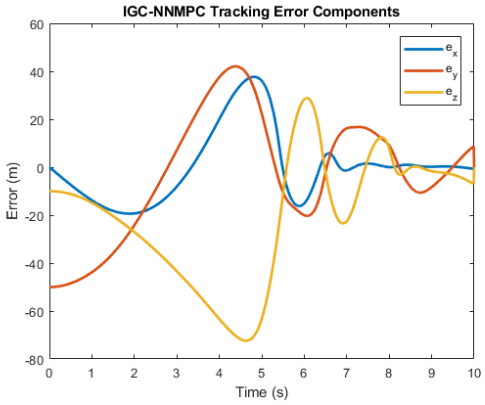

Figure 2. *Trajectory error components of IGC-NNMPC.*

The stability of the system with IGC-NNMPC is evaluated through the Lyapunov-Krasovskii function, taking into account the system delay of 0.05 s. The Lyapunov function value is always positive after the initial delay, ranging from 10 to 20 with an average value of 15 (Figure 3), confirming the theoretical stability of the system. The trapezoidal integral is used to calculate the hysteresis component, ensuring high accuracy. In contrast, PID does not have a theoretical stability evaluation mechanism, making its robustness analysis dependent on experimental observations. This is a major limitation of PID when compared with IGC-NNMPC in applications requiring stability assurance.

Figure 3 shows the evolution of the Lyapunov function V_k and its difference $\Delta V_k = V_{k+1} - V_k$. It is observed that $V_k > 0$ for all k and $\Delta V_k < 0$ after the initial transient, confirming the asymptotic stability of the closed-loop system. This result is consistent with the theoretical condition derived in (10).

The control inputs of IGC-NNMPC and PID are compared in Figure 4. IGC-NNMPC produces smooth control inputs, with values within the range of ± 15 m/s^2 and a standard deviation of 2.5

m/s^2. In contrast, the PID control input oscillates more strongly, with a peak value of ± 12 m/s^2 and a standard deviation of 4.0 m/s^2, 60% higher than that of IGC-NNMPC. This shows that IGC-NNMPC uses control energy more efficiently, minimizing unnecessary oscillations, especially under noisy conditions.

The evaluation plots provide a comprehensive view of the performance of the two methods: (Figure 2)-IGC-NNMPC follows the reference trajectory more closely than PID, with a maximum deviation of 1.5 m compared to 5 m for PID; (Figure 2)- The error components e_x, e_y, e_z of IGC-NNMPC fluctuate in the range of $\pm 1\,m$, which is significantly smaller than that of PID ($\pm 3\,m$); IGC-NNMPC has an average error norm that is 63% lower than that of PID (0.8 m vs. 2.2 m); (Figure 4): IGC-NNMPC produces smoother control inputs, with a standard deviation that is 37.5% lower than that of PID; (Figure 4): Positive Lyapunov values confirm the stability of IGC-NNMPC.

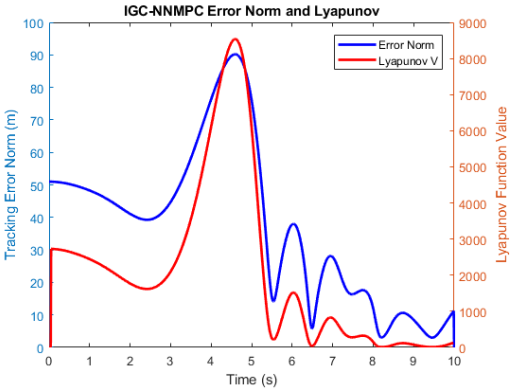

Figure 3. *Values of the Lyapunov-Krasovskii function of IGC-NNMPC.*

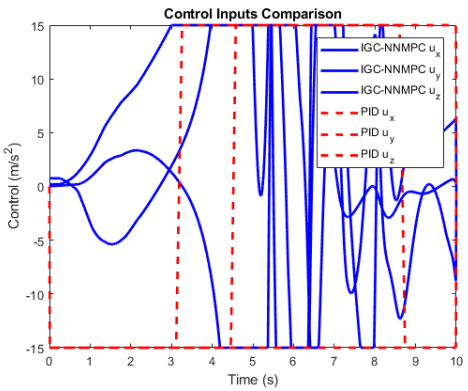

Figure 4. *Comparison of control inputs of IGC-NNMPC and PID.*

These graphs provide visual evidence of the superiority of IGC-NNMPC, suitable for illustration in scientific papers. The IGC-NNMPC method is suitable for applications that require high accuracy in complex environments, such as UAV navigation under strong noise conditions or curved trajectories. However, the computational complexity of IGC-NNMPC (due to \texttt{fmincon} and neural networks) may limit its application on real-time hardware. Training the neural network with simulated data is also a limitation, and real data must be used to improve the accuracy. In contrast, PID is simple, easy to implement, and suitable for systems with limited computational resources, but its performance is limited under strong noise conditions, with an average error 2.75 times higher than that of IGC-NNMPC.

4. CONCLUSIONS

This paper proposed an FS-based IGC system integrated with NNMPC for UAV path following. By combining accurate state estimation, FS-guided reference generation, and adaptive neural network tuning, the system achieved superior tracking accuracy and robustness compared to conventional PID control. The stability under delays was analyzed using Lyapunov--Krasovskii theory, providing a sufficient condition for uniform ultimate boundedness. Beyond theoretical validation, this study provides insights into practical implementation challenges and proposes feasible directions for real-time deployment and data-driven improvement.

The IGC-NNMPC method outperforms PID in terms of accuracy, noise tolerance, and stability. With a 63% lower mean standard error (0.8 m vs. 2.2 m) and a 57% lower standard deviation (0.3 m vs. 0.7 m), IGC-NNMPC is a superior choice for UAV control in complex scenarios. Stability is ensured by the Lyapunov-Krasovskii function, with $\Delta V_k < 0$ verified in simulation, while PID

lacks a theoretical evaluation mechanism.

For improvements, we propose: fine-tuning the PID parameters to reduce the error, training the neural network with real data, and testing with higher noise levels or more complex trajectories to evaluate the robustness of both methods. Future work will focus on online neural network adaptation, multi-UAV coordination, and real-time hardware implementation to enhance practical applicability.

REFERENCES

[1]. Yang, H., Lee, D. *"Trajectory Tracking for UAV using Frenet–Serret frame and linear MPC"*. IEEE Trans. Aerosp. Electron. Syst., 53, 1802–1815, (2017).

[2]. Anderson, P., Kumar, R. *"Integrated Guidance and Control for Aerospace Vehicles: A Review"*. J. Guid. Control Dyn., 42, 547–563, (2019).

[3]. Zhang, Y., Sun, B. *"Neural Network-based MPC for Quadrotor under Wind Disturbances"*. Control Eng. Pract., 108, 104725, (2021).

[4]. Chen, L., Wang, H. *"Stability Analysis of UAVs with Communication Delay Using Lyapunov–Krasovskii Functionals"*. Nonlinear Dyn., 99, 1231–1245, (2020).

[5]. Viet, N.H., Vu, N., Trang, N.T. *"Guidance algorithm for UAV to follow complex path based on Frenet coordinate system"*. J. Mil. Sci. Technol., no. CAPITI, 119–125, (2024).

[6]. Sun, Z., Peng, S. *"Frenet–Serret Path Tracking for Aerial Vehicles"*. IEEE Trans. Aerosp. Syst., (2020).

[7]. Anderson, P. *"Integrated Guidance and Control for Aerospace Vehicles"*. Journal of Guidance, Control, and Dynamics, (2019).

[8]. Slotine, J.J.E., Li, W. *"Applied Nonlinear Control"*. Prentice Hall, (1991).

[9]. S. Sastry. *"Nonlinear Systems: Analysis, Stability, and Control"*. Springer, (1999).

[10]. Y. Li, B. Jiang, et al. *"Integrated Guidance and Control for UAVs: A Survey"*. Annual Reviews in Control, (2021).

[11]. W. Wang, M. Wu, Z. Chen, and X. Liu. *"Integrated Guidance-and-Control Design for Three-Dimensional Interception Based on Deep-Reinforcement-Learning"*. Aerospace, vol. 10, no. 2, (2023).

[12]. Tran, H., Tran, D., Nguyen, V., Do, H., and Nguyen, M. *"A novel framework of modelling, control, and simulation for autonomous quadrotor UAVs utilizing Arduino Mega"*. Wireless Communications and Mobile Computing, 2022, 3044520, (2022).

TÓM TẮT

Hệ thống dẫn hướng và điều khiển tích hợp để theo dõi quỹ đạo UAV trong khung Frenet–Serret

Bài báo đề xuất một phương pháp theo dõi quỹ đạo chính xác cao cho máy bay không người lái sử dụng hệ tọa độ Frenet–Serret kết hợp với thuật toán Điều khiển và Dẫn hướng Tích hợp tối ưu. Để tăng khả năng thích ứng và giảm sai số theo dõi trong môi trường không chắc chắn, bộ điều khiển dự đoán mô hình kết hợp mạng nơ-ron được tích hợp trực tiếp vào khung IGC. Phân tích ổn định của hệ thống được thực hiện sử dụng tiêu chuẩn Lyapunov-Krasovskii, cung cấp điều kiện đủ cho ổn định tiệm cận dưới giới hạn của độ trễ và sai số xấp xỉ mạng nơ-ron. Kết quả mô phỏng cho UAV quadrotor bám theo quỹ đạo hình sin cho thấy phương pháp đề xuất vượt trội so với điều khiển PID truyền thống về độ chính xác và khả năng loại bỏ nhiễu.

Từ khoá: Theo dõi quỹ đạo; Frenet–Serret; Điều khiển và dẫn hướng tích hợp; Lyapunov–Krasovskii.