RL-Enhanced Disturbance-Aware MPC for Fast and Robust UAV Trajectory Tracking

Haoxun Shen[1], Junfei Zhan[1], Tengjiao He[2]

***Abstract*—This paper presents a robust Model Predictive Control (MPC) framework for trajectory tracking of unmanned aerial vehicles (UAVs), enhanced by a reinforcement learning (RL) policy for warm-start initialization and a sliding mode observer (SMO) for disturbance estimation. The proposed architecture addresses multifaceted performance degradation, including residual trajectory tracking errors, reduced control responsiveness, and slow early-stage convergence, primarily caused by external disturbances, model uncertainties, and time-varying environmental conditions. An enhanced adaptive super-twisting SMO is developed to estimate lumped disturbances in real time. These estimates are integrated into the MPC prediction model to compensate for mismatches between nominal dynamics and true system behavior. To accelerate control convergence, particularly in improving early stage performance, an offline RL warm-start policy is used to generate an initial control sequence for MPC. The overall framework preserves the constraint handling and predictive capabilities of conventional MPC, while significantly improving robustness, stability, and tracking accuracy. The simulation results validate the effectiveness of the proposed method in achieving reliable and precise trajectory tracking under challenging and uncertain operating scenarios.**

***Index Terms*—Model Predictive Control (MPC), Unmanned Aerial Vehicles (UAVs), Reinforcement Learning (RL), Sliding Mode Observer (SMO).**

I. Introduction

UAVs are increasingly integral in applications such as surveillance, transportation, and environmental monitoring. These tasks require UAVs to follow precise trajectories [1]–[3]. Given these stringent operational demands, the development of robust control strategies for accurate trajectory tracking has emerged as a critical research challenge in UAV control [4], [5].

Precise UAV trajectory tracking faces significant challenges primarily due to external disturbances, such as wind gusts and variable weather, which induce deviations from intended flight paths. In addition, modeling uncertainties, resulting from simplified or inaccurate representations of UAV dynamics, complicate the design of precise control strategies [6]. Furthermore, physical limitations inherent in actuators and sensors impose practical restrictions that hinder accurate trajectory tracking [2].

Among existing trajectory tracking control approaches, MPC has been widely adopted for its ability to optimize control inputs over a prediction horizon while explicitly handling system constraints [7]. However, conventional MPC designs often exhibit degraded performance in the presence of substantial disturbances and unmodeled dynamics [8], leading to residual tracking errors and reduced control responsiveness. These limitations have motivated the development of intelligent and robust control frameworks to enhance performance.

Recent advances in learning-augmented MPC have shown promise in addressing the challenge of precise UAV trajectory tracking under dynamic and uncertain environments [9], [10]. To address the computational burden and disturbance sensitivity of traditional MPC, various learning-based strategies have been proposed. Approaches such as memory-based learning with stability guarantees [11], Gaussian Mixture Model (GMM)-based warm-start strategies [12], and self-supervised learning-based acceleration methods [13] have demonstrated resilience to uncertainty and convergence speed enhancement.

Beyond learning enhancements, real-time disturbance estimation provides a complementary path to improving MPC robustness. Sliding mode observers (SMOs) offer robust and finite-time estimation of disturbances and unmeasured states under model uncertainties [8], [14].

Recent work has extended SMOs to industrial Cyber-Physical Systems (CPS) scenarios [15], while in UAV applications they have been used successfully to reject external disturbances, such as wind gusts [4], [16].

To address the challenges of external disturbances, modeling uncertainties, and physical limitations, we propose *RL-Observer-Augmented MPC (ROAM)*, a hybrid architecture that enhances UAV trajectory tracking by integrating an RL policy with an adaptive sliding mode observer (AST-SMO). The RL warm-start policy, pre-trained on simplified UAV dynamics, generates initial control sequences to warm-start the MPC solver, thus accelerating convergence and improving early-stage responsiveness [13]. Concurrently, the AST-SMO estimates disturbances in real time, enabling MPC to compensate for uncertainties. In ROAM, the RL policy expedites MPC convergence, MPC stabilizes the observer, and the observer refines MPC prediction, forming a feedback-driven synergy.

The contributions of this paper are as follows.

- We introduce *ROAM*, a hybrid MPC framework for robust UAV trajectory tracking under uncertainty, which integrates RL and AST-SMO.
- An RL-based policy efficiently warm-starts MPC, accelerating convergence and improving early-stage control.
- A novel AST-SMO enables an accurate real-time disturbance estimation.

Corresponding author: Tengjiao He (email: htj2018@jnu.edu.cn).

[1]Department of Electrical and Systems Engineering, University of Pennsylvania, Philadelphia, PA 19104, USA.

[2]College of Information Science and Technology, Jinan University, Guangzhou, China.

- Simulations demonstrate that *ROAM* outperforms baseline MPC in tracking accuracy, robustness, and efficiency under significant disturbances.

The remainder of the paper is organized as follows. Section II describes the dynamic model of the UAV. Section III presents the proposed RL-enhanced disturbance-aware MPC framework. Section IV provides theoretical analysis including the MDP formulation and stability proof. Section V reports the simulation results and comparative evaluations. Finally, Section VI concludes the paper.

II. QUADROTOR DYNAMIC MODELING

The 12-DOF UAV state vector is defined as:

$$\begin{aligned} \mathbf{x} &= \begin{bmatrix} x^{pos} & x^{att} \end{bmatrix}^T, \\ x^{pos} &= \begin{bmatrix} x & y & z & u & v & w \end{bmatrix}, \\ x^{att} &= \begin{bmatrix} p & q & r & \phi & \theta & \psi \end{bmatrix}, \end{aligned} \tag{1}$$

where x, y, z are the position components, u, v, w are the linear velocity components, p, q, r are the angular velocity components, and ϕ, θ, ψ are the Euler angles representing roll, pitch, and yaw, respectively.
The input vector is:

$$\mathbf{u} = \begin{bmatrix} T & \tau_\phi & \tau_\theta & \tau_\psi \end{bmatrix}^T, \tag{2}$$

where T is the total thrust including f_1, f_2, f_3, f_4 as seen in Fig. 1, while $\tau_\phi, \tau_\theta, \tau_\psi$ are the roll, pitch and yaw torques. Then the nonlinear system is given by:

$$\begin{aligned} \dot{x} &= u, \quad \dot{y} = v, \quad \dot{z} = w, \\ \dot{u} &= \frac{T}{m}(C_\psi S_\theta C_\phi + S_\psi S_\phi) - \frac{k_x}{m}u, \\ \dot{v} &= \frac{T}{m}(S_\psi S_\theta C_\phi - C_\psi S_\phi) - \frac{k_y}{m}v, \\ \dot{w} &= \frac{T}{m}(C_\theta C_\phi) - g - \frac{k_z}{m}w, \\ \dot{p} &= \frac{(I_{yy} - I_{zz})}{I_{xx}}qr + \frac{\tau_\phi}{I_{xx}}, \\ \dot{q} &= \frac{(I_{zz} - I_{xx})}{I_{yy}}pr + \frac{\tau_\theta}{I_{yy}}, \\ \dot{r} &= \frac{(I_{xx} - I_{yy})}{I_{zz}}pq + \frac{\tau_\psi}{I_{zz}}, \\ \dot{\phi} &= p + S_\phi T_\theta q + C_\phi T_\theta r, \\ \dot{\theta} &= C_\phi q - S_\phi r, \\ \dot{\psi} &= \frac{S_\phi}{C_\theta}q + \frac{C_\phi}{C_\theta}r, \end{aligned} \tag{3}$$

where $C_x = \cos(x)$, $S_x = \sin(x)$, and $T_x = \tan(x)$. The constants k_x, k_y, k_z are linear drag coefficients, and g denotes the gravitational constant.

Under typical UAV operating conditions near hover, the angular deviations remain small. Therefore, standard small-angle approximations are used to linearize trigonometric expressions:

$$S_x \approx x, \quad C_x \approx 1, \quad \dot{\phi} = p, \quad \dot{\theta} = q, \quad \dot{\psi} = r. \tag{4}$$

This yields a simplified linear state-space representation suitable for MPC design. By introducing unknown external disturbances, the UAV system with additive disturbance inputs can be rewritten as follows:

$$\begin{aligned} \dot{u} &= -g\theta - \frac{k_x}{m}u + d_u, \\ \dot{v} &= -g\phi - \frac{k_y}{m}v + d_v, \\ \dot{w} &= \frac{T}{m} - \frac{k_z}{m}w + d_w, \\ \dot{p} &= \frac{\tau_\phi}{I_{xx}} + d_p, \\ \dot{q} &= \frac{\tau_\theta}{I_{yy}} + d_q, \\ \dot{r} &= \frac{\tau_\psi}{I_{zz}} + d_r. \end{aligned} \tag{5}$$

The linearized 12-DOF UAV dynamics under a small-angle approximation and external disturbances can be written in the standard state-space form:

$$\dot{\mathbf{x}} = A\mathbf{x} + B\mathbf{u} + Ed, \tag{6}$$

where the state vector $\mathbf{x} \in \mathbb{R}^{12}$, control input $\mathbf{u} \in \mathbb{R}^4$, and disturbance vector $d \in \mathbb{R}^6$.

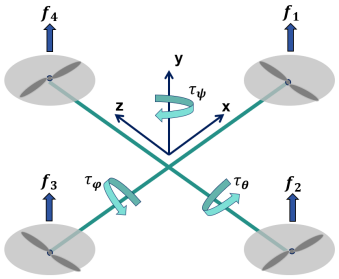

Fig. 1. UAV Body Frame

III. CONTROLLER DESIGN

The overall control framework is illustrated in Fig. 2. The RL-based warm-start policy accelerates the convergence of disturbance-aware MPC by providing trajectory-consistent initial guesses that enhance early-stage responsiveness and solution robustness. The AST-SMO estimates external disturbances $\hat{\boldsymbol{d}}$, which are fed into an augmented MPC prediction model. As the system approaches steady-state, the proposed MPC progressively dominates the control process, leveraging accurate predictions to maintain tracking performance. This integrated loop ensures fast and robust trajectory tracking.

A. RL-Enhanced Warm-Start Policy

The discrete-time state-space model for the UAV is expressed as:

$$\mathbf{x}(k+1) = A\mathbf{x}(k) + B\mathbf{u}(k) = f(x_k, u_k), \tag{7}$$

where $\mathbf{x}(k) \in \mathbb{R}^n$ is the state vector, $u(k) \in \mathbb{R}^m$ is the control input, $y(k) \in \mathbb{R}^p$ is the output, and A, B are the system matrices derived from the UAV dynamics.

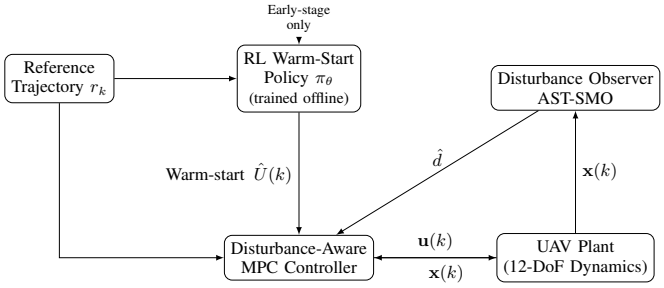

Fig. 2. Overall ROAM architecture UAV trajectory tracking

To compute future states, the model predicts $\mathbf{x}(k+1), \mathbf{x}(k+2), \ldots, \mathbf{x}(k+P)$ over a prediction horizon P using the dynamics as follows:

$$\mathbf{x}(k+P) = A^P \mathbf{x}(k) + \sum_{i=0}^{P-1} A^{P-1-i} B \mathbf{u}(k+i). \quad (8)$$

The state and control input predictions can be reformulated in compact matrix form as:

$$X(k) = F\mathbf{x}(k) + GU(k), \quad (9)$$

where $X(k)$ contains all predicted states while $U(k)$ contains all predicted control inputs over a horizon M, and F and G are prediction matrices derived from A and B as:

$$X(k) = \begin{bmatrix} \mathbf{x}(k+1) \\ \mathbf{x}(k+2) \\ \vdots \\ \mathbf{x}(k+P) \end{bmatrix}, \quad U(k) = \begin{bmatrix} \mathbf{u}(k) \\ \mathbf{u}(k+1) \\ \vdots \\ \mathbf{u}(k+M-1) \end{bmatrix}, \quad (10)$$

$$F = \begin{bmatrix} A \\ A^2 \\ \vdots \\ A^P \end{bmatrix}, \quad G = \begin{bmatrix} B & 0 & \cdots & 0 \\ AB & B & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ A^{P-1}B & A^{P-2}B & \cdots & A^{P-M}B \end{bmatrix}. \quad (11)$$

The objective of the MPC is to minimize the tracking error and control effort over the prediction horizon, formulated as:

$$J_1(k) = \sum_{i=0}^{P-1} [(\mathbf{x}(k+i) - r(k+i))^T Q_1 (\mathbf{x}(k+i) - r(k+i)) + \mathbf{u}(k+i)^T R_1 \mathbf{u}(k+i)], \quad (12)$$

where $r(k)$ is the reference trajectory, Q_1 is the state weight matrix, and R_1 is the control weight matrix. The optimal sequence is derived by solving the quadratic programming problem:

$$U(k) = \left(G^T Q_x G + R_x\right)^{-1} G^T Q_x \left(W_r(k) - F\mathbf{x}(k)\right), \quad (13)$$

where $W_r(k)$ contains stacked reference states over a prediction horizon P. The matrix Q_x is the block diagonal state weighting matrix built from Q_1, and R_x is the corresponding block-diagonal control weight matrix derived from R_1.

The control input for the current step is extracted as:

$$\mathbf{u}(k) = d_1^T U(k), \quad (14)$$

where $d_1^T = [1, 0, \ldots, 0]$ selects the first control input from the sequence. The optimization problem is subject to the following constraints:

$$\begin{aligned} &\text{System Dynamics in Eq. (7)}, \\ &u_{\min} \le \mathbf{u}(k) \le u_{\max}, \\ &\Delta u_{\min} \le \Delta \mathbf{u}(k) \le \Delta u_{\max}. \end{aligned} \quad (15)$$

To accelerate early-stage UAV trajectory tracking, we propose a direction-conditioned RL warm-start policy. At runtime, the UAV's current displacement vector from the initial state is normalized to compute a direction vector $\mathbf{v} = \frac{\mathbf{x}(k) - \mathbf{x}_0}{\|\mathbf{x}(k) - \mathbf{x}_0\|_2}$, which is used as a conditioning input to the RL policy:

$$\hat{\mathbf{u}}(k) = \pi_\theta(\mathbf{x}(k), \mathbf{v}), \quad (16)$$

which produces a feasible control input aligned with the desired motion direction. This initial action $\hat{\mathbf{u}}(k)$ can be directly used to warm-start the MPC solver with a single-step initialization. Alternatively, a full warm-start sequence $\hat{U}(k)$ can be constructed by recursively rolling out the RL policy under nominal system dynamics:

$$\hat{U}(k) = \begin{bmatrix} \pi_\theta(\mathbf{x}(k), \mathbf{v}) \\ \pi_\theta(\hat{\mathbf{x}}(k+1), \mathbf{v}) \\ \vdots \\ \pi_\theta(\hat{\mathbf{x}}(k+M-1), \mathbf{v}) \end{bmatrix}, \quad (17)$$

providing a trajectory-consistent warm-start that improves QP solver performance and robustness.

Algorithm 1: RL Warm-Start via Direction-Conditioned Goal Sampling and MPC Imitation

Input: Nominal dynamics $\mathbf{x}(k+1) = f(\mathbf{x}(k), \mathbf{u}(k))$, directions $\{\mathbf{v}_i\}$, reward r, MPC expert
Output: Trained policy $\pi_\theta(\mathbf{x}, \mathbf{v})$
for *each direction* $\mathbf{v}_i$ **do**
  Set goal $\mathbf{x}_g = \mathbf{x}_0 + \alpha \mathbf{v}_i$;
  for *episode* $= 1$ *to* M **do**
    Initialize $\mathbf{x}_0$;
    for $k = 0$ *to* $T-1$ **do**
      Sample $\mathbf{u}(k) \sim \pi_\theta(\mathbf{x}(k), \mathbf{v}_i)$;
      Apply $\mathbf{u}(k)$, observe $\mathbf{x}(k+1)$, compute r;
      Compute expert action $\mathbf{u}(k)^{\text{exp}}$ via MPC;
      Store $(\mathbf{x}(k), \mathbf{v}_i, \mathbf{u}(k), r, \mathbf{x}(k+1), \mathbf{u}(k)^{\text{exp}})$;
    Update π_θ and Q_ϕ using hybrid loss:
    $\mathcal{L} = -\mathbb{E}_{(\mathbf{x},\mathbf{v})} [Q_\phi(\mathbf{x}, \mathbf{v}, \pi_\theta(\mathbf{x}, \mathbf{v}))] +$
    $\lambda \|\pi_\theta(\mathbf{x}, \mathbf{v}) - \mathbf{u}^{\text{exp}}(\mathbf{x}, \mathbf{v})\|^2$;
return $\pi_\theta(\mathbf{x}, \mathbf{v})$

The RL policy π_θ is trained offline over direction-conditioned tasks defined by virtual goals $\mathbf{x}_g = \mathbf{x}_0 + \alpha \mathbf{v}_i$, where α is a scaling factor controlling task length, using a hybrid actor-critic objective which is primarily guided by MPC

imitation, with Q-based reinforcement serving as auxiliary refinement. This RL warm-start policy substantially reduces computational cost during the early control stage. For the formal MDP formulation that underpins this training process, see Subsection IV-A.

B. Disturbance observer

In practical scenarios, external disturbances are often unmodeled or uncertain. Consequently, the utilization of SMO is typically capable of robustly estimating external disturbances, represented by the variable $\hat{d}(k)$ realization of equivalent input compensation:

$$\tilde{\mathbf{x}}(k+1) = A\tilde{\mathbf{x}}(k) + B\mathbf{u}(k) + Ed(k), \tag{18}$$

$$\hat{\mathbf{x}}(k+1) = A\hat{\mathbf{x}}(k) + B\mathbf{u}(k) + E\hat{d}(k), \tag{19}$$

where $\tilde{\mathbf{x}}(k)$ denotes the actual system state with external disturbances, and $\hat{\mathbf{x}}(k)$ is the estimated system state.

The sliding mode surface is constructed using the acceleration-level states, which are inferred from the system model or smoothed numerical differentiation:

$$s(k) = \Lambda\left(\mathbf{x}_d(k) - \hat{\mathbf{x}}_d(k)\right), \tag{20}$$

where $\mathbf{x}_d(k) = \begin{bmatrix}\dot{u}(k) & \dot{v}(k) & \dot{w}(k) & \dot{p}(k) & \dot{q}(k) & \dot{r}(k)\end{bmatrix}^T \in \mathbb{R}^6$, and $\Lambda \in \mathbb{R}^{6\times 6}$ is a diagonal gain matrix that defines the convergence rate.

To enable continuous but robust correction action, the hyperbolic tangent function is used as a smooth approximation of the sign function:

$$H(s(k)) = \tanh(as(k)) = \frac{e^{as(k)} - e^{-as(k)}}{e^{as(k)} + e^{-as(k)}}, \quad a > 0, \tag{21}$$

where a controls the slope steepness.

The disturbance observer is implemented in discrete time using a two-stage update:

$$\delta\hat{d}(k) = -k_1|s(k)|^{1/2} \cdot H(s(k)) - k_2 \cdot \xi(k), \tag{22}$$

$$\hat{d}(k+1) = \hat{d}(k) + \delta\hat{d}(k) \cdot \Delta t, \tag{23}$$

where $k_1, k_2 > 0$ are observer positive gains, and $\xi(k)$ is an integral auxiliary variable recursively updated as $\xi(k+1) = \xi(k) + \Delta t \cdot H(s(k))$.

To improve transient convergence and steady-state smoothness, we introduce an adaptive mechanism for the first-stage gain:

$$k_1(k) = k_{1,\text{base}}\left(1 + \gamma \cdot |s(k)|\right), \tag{24}$$

where $k_{1,\text{base}} > 0$ is the nominal gain, and $\gamma > 0$ is a tunable gain adaptation coefficient. This gain grows during large transients and shrinks when the error is small.

This AST-SMO formulation ensures fast disturbance estimation with robustness against measurement noise and bounded perturbations, making it suitable for real-time integration with robust model predictive control frameworks.

C. Disturbance-aware MPC

To improve robustness against external disturbances and model mismatch, the nominal MPC is enhanced by incorporating disturbance estimates $\hat{d}(k)$ obtained from the SMO. These estimates are used as real-time additive compensation in the prediction model.

The MPC minimizes a quadratic cost over a prediction horizon P:

$$J_2(k) = \sum_{i=0}^{P-1}\left[e_i^\top Q_2 e_i + \Delta u_i^\top R_2 \Delta u_i\right], \tag{25}$$

where $e_i = \hat{\mathbf{x}}(k+i) - r(k+i)$ is the predicted tracking error, $\Delta\mathbf{u}_i = \mathbf{u}(k+i) - \mathbf{u}(k+i-1)$ is the control increment, and $Q_2 \succeq 0$, $R_2 \succ 0$ are weighting matrices.

The prediction model is augmented by disturbance compensation using the disturbance observer estimation:

$$\mathbf{x}(k+1) = A\mathbf{x}(k) + B\mathbf{u}(k) - E\hat{d}(k). \tag{26}$$

The constrained optimization problem is formulated as:

$$\begin{aligned}\min_{\Delta u(k)} \quad & \sum_{i=0}^{M-1}\left[e_{k+i}^T Q_2 e_{k+i} + u_{k+i}^T R_2 u_{k+i}\right] \\ \text{s.t. } & \text{System Dynamics in Eq. (26)}, \\ & \Delta u_{\min} \le \Delta\mathbf{u}(k) \le \Delta u_{\max}, \\ & u_{\min} \le \mathbf{u}(k) \le u_{\max}.\end{aligned} \tag{27}$$

IV. Analysis

A. MDP formulation of warm-start policy

The warm-start policy can be formally described using a contextual Markov Decision Process (MDP), which provides the theoretical foundation for the learning framework introduced earlier. We define the contextual MDP as the tuple $(\mathcal{S}, \mathcal{V}, \mathcal{A}, P, r)$, where:

- $\mathcal{S} \subset \mathbb{R}^{12}$: the UAV state space;
- $\mathcal{V} \subset \mathbb{R}^3$: the context space, representing normalized direction vectors toward virtual goals;
- $\mathcal{A} \subset \mathbb{R}^4$: the continuous control action space;
- $P(s' \mid s, a)$: the nominal UAV transition dynamics;
- $r(s, v, a) = -\|s - \mathbf{x}_g\|^2 - \lambda\|a - a^{\text{exp}}(s, v)\|^2$: a hybrid reward that encourages the agent to reach a virtual goal $\mathbf{x}_g = \mathbf{x}_0 + \alpha v$ and imitate the MPC expert action $a^{\text{exp}}(s, v)$.

The implementation details are provided in Subsection III-A.

B. Practical finite-time convergence analysis of the observer

Theorem 1. *Let $k_1 > 0$, $k_2 > 0$, and $H(s) = \tanh(as)$ with $a > 0$ (smooth, odd, bounded, and strictly increasing). Define $\phi := \sigma/a$ for some fixed $\sigma > 0$. Then the system*

$$\dot{s} = \eta, \qquad \dot{\eta} = -k_1|s|^{1/2}H(s) - k_2 H(s)$$

achieves practical finite-time convergence*: there exists $t_f < \infty$ such that $|s(t)| \le \phi$ for all $t \ge t_f$, and $|s(t)|$ converges exponentially to an $O(\phi)$ neighborhood of the origin. As $a \to \infty$ (i.e., $\phi \to 0$), the residual can be made arbitrarily small.*

Proof. Define the standard Lyapunov function:

$$V(s,\eta)=\tfrac{1}{2}s^2+\tfrac{1}{2}\eta^2, \qquad \dot{V}=s\eta-k_1\eta|s|^{1/2}H(s)-k_2\eta H(s).$$

Outer region ($|s| \geq \phi = \sigma/a$): In this region, we have $|as| \geq \sigma$ and hence $H(s)\,\mathrm{sign}(s) = \tanh(as)\,\mathrm{sign}(s) \geq \underline{h} := \tanh(\sigma) > 0$. Then:

$$\dot{V} \leq s\eta - k_1\underline{h}|\eta||s|^{1/2} - k_2\underline{h}|\eta|.$$

Using Young's inequality $s\eta \leq \frac{\varepsilon}{2}s^2 + \frac{1}{2\varepsilon}\eta^2$ and norm equivalence $V \geq c_1|s| + c_2|\eta|^2$, we obtain:

$$\dot{V} \leq -\gamma_1|\eta||s|^{1/2} - \gamma_2|\eta| + \bar{c}V \leq -\gamma V^{\delta},$$

for some constants $\gamma > 0$, $\delta \in (0,1)$. By the comparison lemma, $V(t)$ reaches the level $|s| = \phi$ in finite time:

$$t_f \leq \frac{V(0)^{1-\delta}}{\gamma(1-\delta)}.$$

Inner region ($|s| \leq \phi$): Here, $|H(s)| \leq a|s|$. Then:

$$\dot{V} = s\eta - k_2 a\, s\eta = -(k_2 a - 1)\, s\eta.$$

If $k_2 a > 1$, we have $|s\eta| \leq \frac{1}{2}(s^2+\eta^2) = V$, and hence:

$$\dot{V} \leq -\lambda V, \quad \text{where} \quad \lambda := k_2 a - 1 > 0.$$

This yields exponential convergence for $t \geq t_f$:

$$V(t) \leq e^{-\lambda(t-t_f)}V(t_f),$$

which implies exponential convergence of $|s(t)|$ to an $O(\phi)$ neighborhood of the origin. As $a \to \infty$, $\phi \to 0$, and the residual vanishes. □

Remark 1. *The observer is implemented in discrete time via Euler approximation. For sufficiently small sampling time T_s, such that discretization error is dominated by the convergence rate, the practical convergence remains valid.*

V. Simulation Results

The MPC optimization problem is restructured in a lifted matrix form. The simulation parameters for ROAM are set as follows. The UAV has a mass of $0.65\,\mathrm{kg}$ and is subject to standard gravity $g = 9.81\,\mathrm{m/s^2}$. Body inertia along the principal axes is given by $I_x = I_y = 8 \times 10^{-3}\,\mathrm{kg{\cdot}m^2}$ and $I_z = 1.5 \times 10^{-2}\,\mathrm{kg{\cdot}m^2}$. The control increment is constrained within $[-30, 30]$, and the control input bounds are set to $[-400, 400]$. The system runs at a sampling frequency of $50\,\mathrm{Hz}$ ($T_s = 0.02\,\mathrm{s}$), with the MPC configured using a prediction horizon $P = 30$, control horizon $M = 50$, penalty weights $Q_1 = Q_2 = I$ and $R_1 = R_2 = 0.001 \cdot I$. An RL-based warm-start policy initializes the control sequence $\boldsymbol{u}_0$, loaded from a pretrained checkpoint. External disturbances are modeled as additive sinusoidal signals with Gaussian noise: $d_i(t) = A_i \sin(2\pi f_i t) + \mathcal{N}(0, \sigma_i^2)$, where $f = [0.3, 0.3, 0.3, 0.5, 0.5, 0.5]\,\mathrm{Hz}$, $A_i = [0.75, 0.75, 0.75, 0.5, 0.5, 0.5]$, and $\sigma = [0.02, 0.02, 0.02, 0.015, 0.015, 0.015]$. The enhanced AST-SMO uses the gains $a = [5.0, 5.0, 5.0, 5.0, 5.0, 5.0]^\top$, $\Lambda = 2.75 \cdot I$, $k_{1,\mathrm{base}} = [0.52, 0.52, 0.52, 0.47, 0.47, 0.47]^\top$, the integral gains $k_2 = [0.1, 0.1, 0.1, 0.1, 0.1, 0.1]^\top$, and the smoothing factor $\gamma = 0.01$. The reference trajectory is defined as $x_{\mathrm{ref}}(t) = \sin(0.035t)$, $y_{\mathrm{ref}}(t) = \cos(0.045t)$, $z_{\mathrm{ref}}(t) = 0.2t$, and $\psi_{\mathrm{ref}}(t) = 45°$, representing a smooth 3D flight path with fixed yaw.

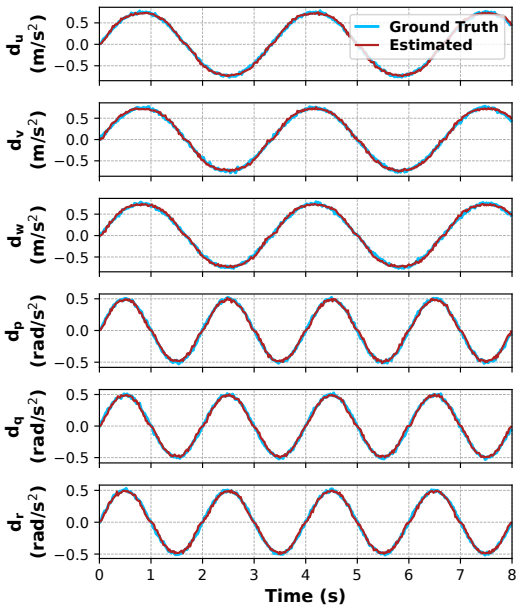

Fig. 3. Comparison between AST-SMO estimations and ground-truth disturbances

Firstly, to evaluate the effectiveness of the proposed disturbance observer, a set of known time-varying external disturbances was designed and applied to the system. From Fig. 3, it is evident that the estimated disturbance tracks the ground-truth disturbance with minimal amplitude error, even in the presence of stochastic noise. In particular, the estimation closely follows different frequency conditions with high accuracy, indicating strong robustness. Overall, the results validate that the observer possesses excellent noise resilience, dynamic tracking capability, and consistency across six different channels.

TABLE I
Multi-Metric Evaluation of Baseline MPC vs. ROAM

Metric	Baseline MPC	ROAM	Improvement
Early stage solving time	$\approx \mathcal{O}(n^2 \sim n^3)$	$\approx \mathcal{O}(n)$	↓ Complexity
Nearest point error (Terminal)	0.0950 m	**0.0295 m**	↓ 69.0%
Nearest point error (Avg.)	0.0751 m	**0.0532 m**	↓ 29.1%

Secondly, Table I quantitatively compares the baseline MPC with the proposed ROAM in terms of computational complexity and trajectory tracking precision. At the early stage of control, ROAM reduces the solving complexity from $\mathcal{O}(n^2 \sim n^3)$ to $\mathcal{O}(n)$ by leveraging an efficient warm-start strategy, enabling faster closed-loop response. For tracking accuracy, ROAM achieves an average 29.1% reduction in the overall nearest point error and a significant improvement of

69.0% in the terminal point. These confirm ROAM enhances both computational efficiency and tracking performance under disturbances.

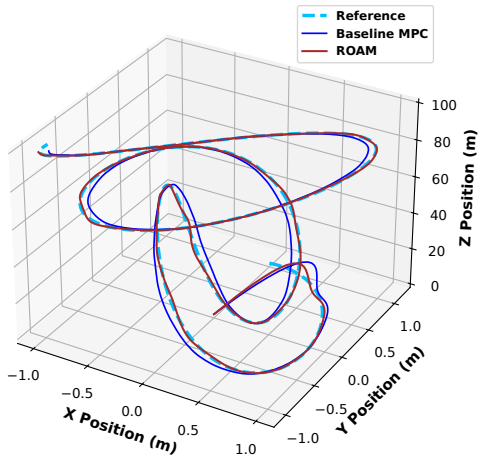

Fig. 4. Trajectory tracking 3D plot comparison under disturbances

Finally, Fig. 4 presents the 3D trajectory tracking results under different control strategies in the presence of external disturbances. The baseline MPC demonstrates slower convergence and noticeable early-stage deviation, requiring over a longer period of time to fully align with the reference path. In contrast, the proposed ROAM achieves substantially faster convergence within less time, while maintaining smoother control behavior. This improvement results from the warm-start policy, which offers an informed initial control action, thereby accelerating early trajectory alignment, and also the disturbance observer, which facilitates real-time rejection of model mismatches and external perturbations. Moreover, the RL-augmented design does not introduce instability or overshoot, confirming the compatibility between learning-based initialization and baseline MPC. While both approaches aim to follow the desired trajectory, the proposed method achieves markedly lower transient errors and higher control efficiency, especially under time-critical or disturbance conditions where baseline MPC struggles to maintain accuracy.

VI. Conclusion

This paper proposes a unified and novel control architecture that tightly couples RL, SMO, and MPC for robust and fast UAV trajectory tracking in disturbance-prone environments. Departing from conventional modular approaches, our design emphasizes the deep interdependence among the learning, observation, and control components.

While model-based components dominate performance, the RL warm-start policy serves as a flexible warm-start to aid early-stage optimization. Additionally, the MPC embeds real-time disturbance estimates from the SMO directly into its predictive model, internalizing robustness within the optimization horizon. Furthermore, consistent control induced by MPC and RL coordination enhances the stability of SMO estimation. This mutually strengthening integration results in a system that excels in tracking accuracy, convergence speed, and resilience to disturbances. Simulations confirm substantial improvements over baseline MPC, both in transient and steady-state regimes. Future efforts will focus on theoretical analysis of the coupling mechanism and real-world hardware experiments.

References

[1] D. Huo, L. Dai, R. Chai, R. Xue, and Y. Xia, "Collision-free model predictive trajectory tracking control for uavs in obstacle environment," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 59, no. 3, pp. 2920–2932, 2023.

[2] S. Zhao, J. Zheng, F. Yi, X. Wang, and Z. Zuo, "Exponential predefined-time trajectory tracking control for fixed-wing uav with input saturation," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 60, no. 5, pp. 6406–6419, 2024.

[3] B. Ma, Z. Liu, W. Zhao, J. Yuan, H. Long, X. Wang, and Z. Yuan, "Target tracking control of UAV through deep reinforcement learning," *IEEE Transactions on Intelligent Transportation Systems*, vol. 24, no. 6, pp. 5983–6000, 2023.

[4] F. Wang, H. Gao, K. Wang, C. Zhou, Q. Zong, and C. Hua, "Disturbance observer-based finite-time control design for a quadrotor uav with external disturbance," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 57, no. 2, pp. 834–847, Apr. 2021.

[5] E. Tal and S. Karaman, "Accurate tracking of aggressive quadrotor trajectories using incremental nonlinear dynamic inversion and differential flatness," *IEEE Transactions on Control Systems Technology*, vol. 29, no. 3, pp. 1203–1218, May 2021.

[6] L. Kong, J. Reis, W. He, and C. Silvestre, "Comprehensive nonlinear control strategy for vtol-uavs with windowed output constraints," *IEEE Transactions on Control Systems Technology*, vol. 31, no. 6, pp. 2673–2684, 2023.

[7] M. A. Najafqolian, K. Alipour, R. Mousavifard, and B. Tarvirdizadeh, "Control of aerial robots using convex qp lmpc and learning-based explicit-mpc," *IEEE Transactions on Industrial Informatics*, vol. 20, no. 9, pp. 10 883–10 891, Sept. 2024.

[8] M. Rubagotti, D. M. Raimondo, A. Ferrara, and L. Magni, "Robust model predictive control with integral sliding mode in continuous-time sampled-data nonlinear systems," *IEEE Transactions on Automatic Control*, vol. 56, no. 3, pp. 556–570, Mar. 2011.

[9] R. Sambharya, G. Hall, B. Amos, and B. Stellato, "Learning to warm-start fixed-point optimization algorithms," *arXiv preprint arXiv:2309.07835*, 2023. [Online]. Available: https://arxiv.org/abs/2309.07835

[10] S. Han, S. Park, and S. Lee, "Sampled-data-based iterative cost-learning model predictive control for t–s fuzzy systems," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 54, no. 8, pp. 4701–4713, Aug. 2024.

[11] L. Schwenkel, M. Gharbi, S. Trimpe, and C. Ebenbauer, "Online learning with stability guarantees: A memory-based warm starting for real-time mpc," *Automatica*, vol. 122, p. 109247, 2020.

[12] M.-K. Bouzidi, Y. Yao, D. Goehring, and J. Reichardt, "Learning-aided warmstart of model predictive control in uncertain fast-changing traffic," in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, 2024.

[13] Z. Li, X. Wang, L. Chen, R. Paleja, S. Nageshrao, and M. Gombolay, "Faster model predictive control via self-supervised initialization learning," *arXiv preprint arXiv:2408.03394*, 2025. [Online]. Available: https://arxiv.org/abs/2408.03394

[14] G. P. Incremona, M. Rubagotti, and A. Ferrara, "Sliding mode control of constrained nonlinear systems," *IEEE Transactions on Automatic Control*, vol. 62, no. 6, pp. 2965–2979, Jun. 2017.

[15] H. Shen, R. Guo, D. Liu, and S. K. Kommuri, "Advanced observer design for sensorless control in industrial physical systems," in *2024 IEEE 7th International Conference on Industrial Cyber-Physical Systems (ICPS)*, May 2024, pp. 1–6.

[16] Y. Hou, D. Chen, and S. Yang, "Adaptive robust trajectory tracking controller for a quadrotor uav with uncertain environment parameters based on backstepping sliding mode method," *IEEE Transactions on Automation Science and Engineering*, vol. 22, pp. 4446–4456, 2025.