Research Square

Preprints are preliminary reports that have not undergone peer review.
They should not be considered conclusive, used to inform clinical practice,
or referenced by the media as validated information.

Characterization of a Fixed Reinforcement Learning Policy for Aerial Robot with Suspended Payload under Variable Flight Conditions

Ali Tahir Karasahin

alitahir.karasahin@erbakan.edu.tr

Necmettin Erbakan University

Ziniu Wu

University of Bristol

Basaran Bahadir Kocer

University of Bristol

Article

Keywords:

Posted Date: March 4th, 2026

DOI: https://doi.org/10.21203/rs.3.rs-8603104/v1

License: This work is licensed under a Creative Commons Attribution 4.0 International License.
Read Full License

Additional Declarations: No competing interests reported.

Version of Record: A version of this preprint was published at Scientific Reports on October 26th, 2026. See the published version at https://doi.org/10.1038/s41598-026-46111-x.

Characterization of a Fixed Reinforcement Learning Policy for Aerial Robot with Suspended Payload under Variable Flight Conditions

Ali Tahir Karasahin[1,2,*], **Ziniu Wu**[1], **and Basaran Bahadir Kocer**[1]

[1]University of Bristol, School of Civil, Aerospace and Design Engineering, Bristol, BS8 1QU, United Kingdom
[2]Necmettin Erbakan University, Faculty of Engineering, Department of Mechatronics Engineering, Konya, 42090, Turkey
[*]alitahir.karasahin@erbakan.edu.tr

ABSTRACT

Flights with suspended payloads are particularly challenging because of their coupled dynamics, which lead to instability and increased sensitivity to disturbances. Although reinforcement learning (RL) has successfully achieved controller performance, the generalization and robustness of a single policy remain significant areas of investigation. In this study, we characterized the performance and robustness of a single RL policy for an aerial robot with different trajectory profiles, including a smooth, feasible lemniscate curve and a sharp-turning, infeasible pentagram, under varying velocity references ($0.5\ \mathrm{m \cdot s^{-1}}$ and $1.0\ \mathrm{m \cdot s^{-1}}$) and crosswind disturbances ($1.0\ \mathrm{m \cdot s^{-1}}$). We trained a single RL policy using Proximal Policy Optimization (PPO) with collective thrust and body-rate (CTBR) control using a high-fidelity physics simulator based on the SimpleFlight framework. Real-world experimental results on the Crazyflie 2.1 platform show that the single RL policy successfully generalizes to different trajectory profiles and velocity references and maintains stability under a crosswind disturbance of up to $1.0\ \mathrm{m \cdot s^{-1}}$ which is a substantial challenge for this small class platform and even smaller payload underneath, where such aerodynamic forces are significant compared to the available control authority and system mass. Furthermore, the single RL policy was systematically evaluated using the Mean Euclidean distance (MED) error, cable length transitions, and swing angle distributions. Although the single RL policy maintained a robust control performance, the experimental results indicated performance degradation at higher velocities owing to increased dynamic challenges such as nonlinear aerodynamic drag and actuator saturation. This study provides a detailed performance characterization that highlights the generalization capability of a single-payload-aware RL policy in real-world applications and the limitations arising from the hybrid dynamics of the system.

Introduction

Aerial robots are increasingly used in complex applications including payload transportation, environmental sensing and aerial manipulation[1–6]. However, precisely tracking a reference trajectory with a suspended payload is complicated by coupled and underactuated dynamics[7–10]. Furthermore, the adaptation capability of these systems is more challenging under external wind disturbances with unmodeled dynamic conditions.

Reinforcement learning (RL) has become a powerful alternative method that enables agents to meet different expectations based on the design of their observation space, action space, and reward function[11]. An RL has been successfully demonstrated in agile flight with a payload[12] and payload transportation with multiple agents[13]. However, whether a single trained policy can cover different expectations, such as trajectory profiles, velocity references, and wind disturbance rejection, remains unclear.

Systematically evaluating RL policies is not only essential for diverse trajectory profiles and disturbance conditions but also to ensure their effectiveness in real-world environments. This limitation exists in payload-aware applications because the coupled dynamics between the aerial robot and suspended payload make trajectory tracking more challenging in nonstationary environments. Therefore, the performance characterization of a single RL policy skill can be evaluated for aerial robots with suspended payload applications.

Recent studies have focused on improving learning capabilities by varying the observation space and shaping the reward function[14,15]. These studies primarily focused on the effects of the observation space and reward function design on the learning capacity of RL policies. Although RL observation spaces and reward functions typically rely on standard inputs, such as agent states, velocities, and goal-reaching bonuses, these studies demonstrate that using different configurations for each component, such as quaternions instead of rotation matrices, is crucial for learning the reward curves. However, these assessments primarily focus on aerial robots in static environments and do not adequately characterize the performance of a single policy. Without this deeper characterization, questions regarding how a policy responds to different configurations and whether switching

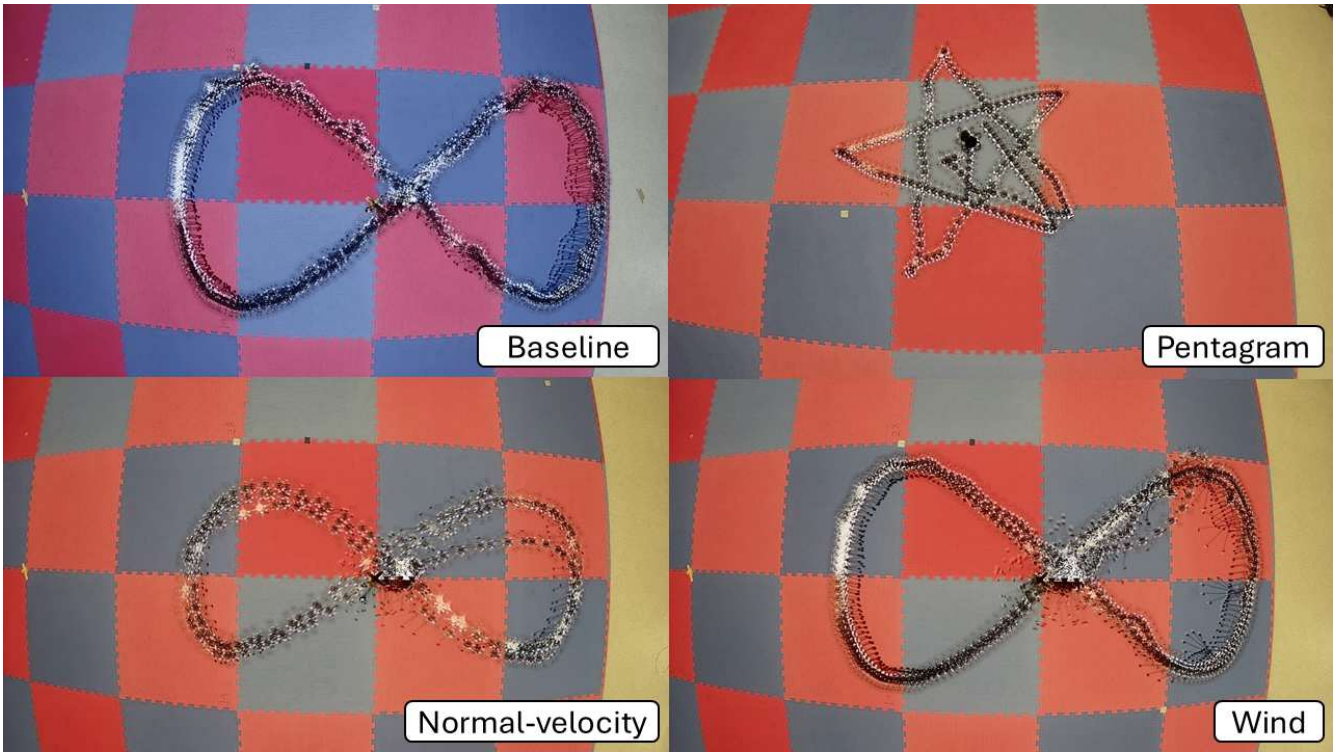


Figure 1. Demonstration of a payload-aware RL policy for different configurations, such as the baseline (low-velocity lemniscate), a pentagram trajectory, increased velocity references, and the introduction of wind disturbances.

to strategies such as online adaptation or continual learning is necessary to maintain stability in the face of environmental variability remain unclear.

In this study, we systematically explored a single payload-aware RL policy to explicitly characterize its generalization capability across diverse trajectory profiles, velocity references, and wind disturbances. Using a high-fidelity simulator built on the SimpleFlight framework, we investigated the performance of the single policy for different trajectory profiles, velocity references, and wind disturbance rejections. We analyze critical research questions, such as the cross-scenario reliability of a single policy and the identification of its operational limits. Specifically, we characterized the performance of a policy trained on a 0.3 m cable with a 5 g payload against diverse test cases, including infeasible pentagram trajectories, velocity increases from 0.5 to 1.0 $\mathrm{m \cdot s^{-1}}$, and crosswind disturbances of 1.0 $\mathrm{m \cdot s^{-1}}$. The contributions of this study are as follows:

(1) We introduce a payload-aware RL policy that explicitly considers suspended payload dynamics.

(2) We characterize the generalization and adaptation capabilities of a single policy across different trajectory profiles, velocity references, and wind disturbance rejection criteria.

(3) We demonstrate the applicability of policy to different trajectory profiles and velocity references without requiring any tuning.

The remainder of this paper is organized as follows. Section II explains the related studies. Section III introduces the proposed payload-aware RL policy method, including the problem formulation and policy components. Section IV presents the experimental results obtained from the simulation and real-world experiments, and discussion. Finally, Section V presents the conclusions and discusses potential future work.

Related Work

Reinforcement Learning for Aerial Robots

RL has been widely used in aerial robots to learn policies through environmental interactions. In addition, this approach has been remarkably successful in acrobatic flight[16], vision-based flight[17] and drone racing[18]. However, aerial robots with suspended payloads differ from these applications. The complex dynamics introduced by the suspended payload present a

challenge for RL policies, particularly in designing the observation space and reward function. Furthermore, representing the cable model in the simulation environment is another issue in policy development[19].

Payload-Aware Control in Aerial Robotics

Aerial robots with suspended payloads have significantly improved their capabilities, including agile flight[12] and decentralized multi-agent tasks[13]. Additionally, some studies have demonstrated agile flight with a suspended payload using vision-based control[20]. Despite these successes, the evaluation of payload-aware RL policies based on diverse trajectory-tracking profiles and velocity references is notably absent. Our study addresses this critical gap by investigating payload-aware RL policies for robust trajectory-tracking. We examined this using both simulations and real-world experiments, which included trajectory types and wind disturbance.

Methodology

System dynamics of aerial robot with suspended payload

Aerial robots with suspended payload systems can be formulated in various ways. We selected an approach that included slack and taut modes for modelling. This approach defines two modes to represent the cable states: slack and taut. The aerial robot and payload are dynamically coupled when the remains taut. The system dynamics of an aerial robot with a suspended payload are as follows:

$$
\begin{aligned}
&(M_q + M_l)(\ddot{\boldsymbol{p}}_{\boldsymbol{q}} + L\ddot{\boldsymbol{u}} - \boldsymbol{g}) = \left(\boldsymbol{u}^T \mathbf{R_b} \boldsymbol{F_T} - M_q L(\dot{\boldsymbol{u}^T} \cdot \dot{\boldsymbol{u}})\right) \boldsymbol{u} \\
&\boldsymbol{p_l} = \boldsymbol{p_q} + L\boldsymbol{u}, \;\; \dot{\mathbf{R}_\mathbf{b}} = \mathbf{R}\hat{\boldsymbol{\Omega}} \\
&M_q L\ddot{\boldsymbol{u}} = -M_q L(\dot{\boldsymbol{u}^T} \cdot \dot{\boldsymbol{u}})\boldsymbol{u} + \boldsymbol{u} \times (\boldsymbol{u} \times \mathbf{R_b} \boldsymbol{F_T}) \\
&J_q \dot{\boldsymbol{\Omega}} = \boldsymbol{U_M} - \boldsymbol{\Omega} \times J_q \boldsymbol{\Omega}
\end{aligned}
\tag{1}
$$

where $\mathbf{F_T} = [0,0,F_T]^T$ denotes the thrust force vector produced by the aerial robot, $\mathbf{R_b} \in SO(3)$ is the rotation matrix transforming vectors from the body frame to the inertial frame of the aerial robot, and $\mathbf{g} = [0,0,g]^T$ denotes the gravitational vector. The parameter $\mathbf{L}$ represents the cable length of the suspended payload, whereas $\mathbf{u} \in \mathbb{R}^3$ is a unit vector along the cable, pointing from the aerial robot to the suspended payload. The positions vectors of the aerial robot and suspended payload are represented by $\mathbf{p_q} \in \mathbb{R}^3$ and $\mathbf{p_l} \in \mathbb{R}^3$, respectively, and $\blacksquare \in \mathbb{R}^3$ indicates the aerial robot's body angular velocity values. The mass parameters of the aerial robot and suspended payload are represented by $\mathbf{M_q}$ and $\mathbf{M_l}$. The inertia matrix of the aerial robot is used as $\mathbf{J_q} \in \mathbb{R}^{3x3}$. $\mathbf{U_M} \in \mathbb{R}^3$ denotes the control moment vector applied to the aerial robot.

In slack mode, the aerial robot follows its nominal, unconstrained dynamics, whereas the payload experiences free fall under the force of gravity.

This transition behavior produces hybrid dynamics in aerial robots, which challenges traditional optimization-based control methods owing to their increased computational load. To address these challenges, we selected a representative RL approach that can handle nonlinear transitions, learn from them, and adapt without requiring a model.

Problem Formulation

To model the aerial robot with a suspended payload task, we used Eq. (1) to represent its kinematics and dynamics, respectively. We defined a Markov Decision Process (MDP) as $M = (S, A, P, R, \gamma)$, where S introduces the state space of the aerial robot with a suspended payload. The action space is defined by A, where each action output $a \in A$ corresponds to the control input applied to the aerial robot. The transition dynamics P are modeled as the conditional probability $P(s_{t+1}|s_t, a_t)$, which defines the next state s_{t+1} from the current state s_t and the applied action a_t. The reward function $r_t = R(s_t, a_t)$ at each time step produces a scalar value based on the state s_t and action a_t. Finally, $\gamma \in [0,1)$ is the discount factor that tunes the relative importance of future rewards. Using the MDP approach, our goal is to obtain the optimal policy, π^*, that maximizes the expected cumulative discounted return, J_π.

$$
\pi^* = \underset{\pi}{\operatorname{argmax}} J_\pi = \sum_{t=0}^{\infty} \gamma^t \cdot \mathbb{E}\left[R(s_t, a_t)|\pi\right] \tag{2}
$$

This objective function was created to develop a policy for an aerial robot with a suspended payload to learn the different capabilities, including both diverse trajectory tracking and velocity references and wind disturbance in the high-fidelity SimpleFlight framework[14] using the PPO algorithm[21].

Observation and Reward Function

For all experiments, we used the same observations as follows:

$$o = \left[\mathbf{p_r}, \mathbf{p_t}, \mathbf{v_q}, \mathbf{v_l}, \mathbf{R_q}\right] \tag{3}$$

where $\mathbf{p_r} = \mathbf{p_q} - \mathbf{p_l}$ represents the relative position of the aerial robot with respect to the suspended payload. The remaining components include$\mathbf{p_t}$ as the next k reference waypoints to the suspended payload, $\mathbf{v_q}$ as the linear velocity of the aerial robot, $\mathbf{v_l}$ as the linear velocity of the suspended payload, and $\mathbf{R_q}$ as the orientation of the aerial robot based on the heading, lateral, and up vectors.

To teach the aerial robot certain skills, we designed a reward function that covers conditions such as different trajectory profiles, velocity references, and wind disturbances, with and without them. The reward function is designed as follows:

$$r = r_{\text{pos}} + r_{\text{pos}}(r_{\text{up}} + r_{\text{spin}}) + r_{\text{act_smooth}} \tag{4}$$

We designed a payload-aware reward function that did not restrict the positions of aerial robots during training. r_{pos} represents the difference between the desired payload trajectory and the actual position of the payload. This component was designed for the payload-aware feature and was used with other components, such as r_{up} and r_{spin}, to influence the total reward. r_{up} encourages the aerial robot to maintain an upright position, which supports the performance of payload trajectory tracking. r_{spin} penalizes the yaw rotation of the aerial robot and improves the trajectory tracking capabilities of the payload. $r_{\text{act_smooth}}$ maintains the smoothness of the action output and supports the sim2real gap for applicable action outputs under real-world conditions. The components and hyperparameters of the total reward function are listed in Table 1.

Table 1. The reward components and hyperparameters used in the suspended payload UAV task.

Reward Term	Function	Weight	Category
r_{pos}	$r_{pos} = k_d \cdot e^{-\|p_r\|}$	$k_d = 5.0$	Guidance
r_{up}	$r_{up} = k_u \cdot \frac{0.5}{1+\|1-u_z\|^2}$	$k_u = 1.0$	Stability
r_{spin}	$spin = (w_z^b)^2$ $r_{spin} = k_s \cdot \frac{0.5}{1+spin^2}$	$k_s = 1.0$	Stability
r_{act_smooth}	$r_{act_smooth} = k_{sm} \cdot e^{-\|\Delta a\|}$	$k_{sm} = 2.0$	Smoothness

Training and Implementation Details

We used the SimpleFlight framework[14] for both its zero-shot capability and its use of the PPO algorithm[21]. A policy network was created using a three-layer multilayer perceptron with 256 neurons. The activation function was an exponential linear unit (ELU). The last layer of the actor network outputs a four-dimensional vector representing the collective thrust and body rate (CTBR) with a Gaussian distribution. SimpleFlight uses an asymmetric actor-critic architecture. SimpleFlight is built on the high-fidelity physics simulator Nvidia Isaac Sim[22,23], which is GPU accelerated. The payload-aware policy was trained for 7,500 epochs. It uses a single policy for various trajectory profiles and velocity references, with or without wind disturbances. In this study, a payload-aware policy was developed using SimpleFlight. The physical parameters of the aerial robot are listed in Table 2.

Table 2. The physical parameters of the aerial robot used in the training, simulation and real-world environments.

Parameter (unit)	Crazyflie Platform
Quadrotor mass, m (kg)	0.0321
Inertia, J ($\text{kg} \cdot \text{m}^2$)	[1.4e-5, 1.4e-5, 2.17e-5]
Arm length (m)	0.043
Thrust-to-Weight-Ratio	1.95
Maximum thrust (N)	0.575
Motor time constant (s)	0.025
Payload mass (kg)	0.005
Cable length (m)	0.3

Simulation Environment Setup

We used the Isaac Sim simulation environment, which provides high-fidelity physics and a highly parallel environment. This setup can efficiently simulate an aerial robot with adjustable dynamics and control the algorithms. This is particularly useful for developing and testing various RL algorithms. We performed all simulations and RL training on a laptop equipped with an Intel®Alder Lake Core™i7-12700H processor, an NVIDIA™GeForce RTX 4060 GPU with 8 GB GDDR6 memory, and 32 GB DDR4 RAM (2×16 GB, 3200 MHz).

Experiments And Results

We evaluated the learned single RL policy based on different trajectory tracking profiles and velocity references, with and without wind disturbances. To meet the requirements of the RL policy, we trained an RL agent powered by a curriculum learning mechanism that covered the complex trajectory profiles of the target. During the 7,500-epoch training period, the RL agent attempted to maximize the reward based on the designed functions.

Real-world Deployments

To evaluate the real-world performance of a single RL policy, we conducted four sets of experimental scenarios in a flight arena, with a total of 20 real-world flight trials. To ensure the consistency and statistical reliability of the performance characterization, each set consisted of multiple trials under baseline, pentagram trajectory, normal velocity reference, and wind disturbance conditions. We used a Crazyflie 2.1 aerial robot with a suspended payload (5 g). The aerial robot position and orientation values were collected using the VICON motion capture system at 100 Hz. The single RL policy calculated the CTBR control commands based on these observations. This output was sent via the crazyswarm2 library at 100 Hz by Crazyradio 2.0. In addition, we used the same hardware implementation as the SimpleFlight framework in the real-world experiments[14]. We used an industrial fan to simulate wind disturbance during real-world experiments. We measured the wind force at the fixed point with an anemometer of approximately 1.0 $\mathrm{m}\cdot\mathrm{s}^{-1}$. During the wind experiments, the industrial fan did not follow the aerial robot and was applied as a crosswind.

Evaluation Metrics

To analyze the performance characterization of a payload-aware reinforcement learning policy, we evaluated both the trajectory-tracking performance and payload swing dynamics. The performance of the task and dynamics of the cable-suspended payload system are reflected in the following metrics:

Mean Euclidean Distance (MED): The primary task objective is to track the desired payload position, and we are not interested in the aerial robot position. The Mean Euclidean Distance represents the average tracking error during the experiment.

$$\mathrm{MED} = \frac{1}{N}\sum_{t=1}^{N} \left\|\mathbf{p}_p(t) - \mathbf{p}_d(t)\right\|_2 \tag{5}$$

where $\mathbf{p}_p(t)$ and $\mathbf{p}_d(t)$ denote the payload and desired positions at time t, respectively. A lower MED indicates better trajectory-tracking performance under the given conditions.

Swing Angle θ: The movement of the payload can be represented by the movement of a pendulum. This movement provides the characteristic output required to evaluate the control performance of the aerial robot. The swing angle is defined as the angle between the cable direction and the vertical axis, as follows:

$$\theta(t) = \arccos\left(\frac{\mathbf{r}(t)}{\|\mathbf{r}(t)\|_2}\cdot\mathbf{e}_z\right) \tag{6}$$

where $\mathbf{r}(t) = \mathbf{p}_p(t) - \mathbf{p}_q(t)$ is the drone-to-payload vector, and $\mathbf{e}_z$ is the unit vertical axis. We calculated the mean ($\bar{\theta}$), standard deviation (σ_θ), maximum peak ($\theta_{\max}$), and 95th percentile (θ_{95}) to characterize the general oscillation magnitudes.

Swing Rate $\dot{\theta}$: While the magnitude of oscillations is represented by the parameter θ, its derivative indicates the control authority:

$$\dot{\theta}(t) = \frac{\theta(t) - \theta(t-\Delta t)}{\Delta t} \tag{7}$$

We summarized the swing rate using the root mean square:

$$\dot{\theta}_{\mathrm{RMS}} = \sqrt{\frac{1}{N-1}\sum_{t=2}^{N}\dot{\theta}(t)^2} \tag{8}$$

A lower $\dot{\theta}_{\mathrm{RMS}}$ indicates an improved controller-damping capability and reduced high-frequency oscillations.

Cable Length L: the instantaneous cable length is

$$L(t) = \|\mathbf{r}(t)\|_2 \tag{9}$$

When the cable is taut, the payload behaves as a single-link pendulum with well-defined dynamics. Therefore, the cable length was used to examine both the stability and control authority. To systematically evaluate the performance characteristics of the learned policy, we examined the following core questions:

Q1 Trajectory Generalization: How do MED and θ_{95} change between different reference trajectories (e.g., lemniscate vs. pentagram), and what does this indicate about the policy's ability to generalize across motion patterns?

Q2 Speed Robustness: How does increasing the trajectory speed affect $\dot{\theta}_{\mathrm{RMS}}$? Does the policy retain the damping authority under higher velocity excitation?

Q3 Disturbance Resilience: How does the behavior of cable length, $\theta_{\max}$ and swing angle transition under wind perturbation?

To characterize the performance of a single RL policy, we systematically investigated it based on the following research questions: We evaluated the comprehensive results obtained from the flight arena and provided an assessment of these results.

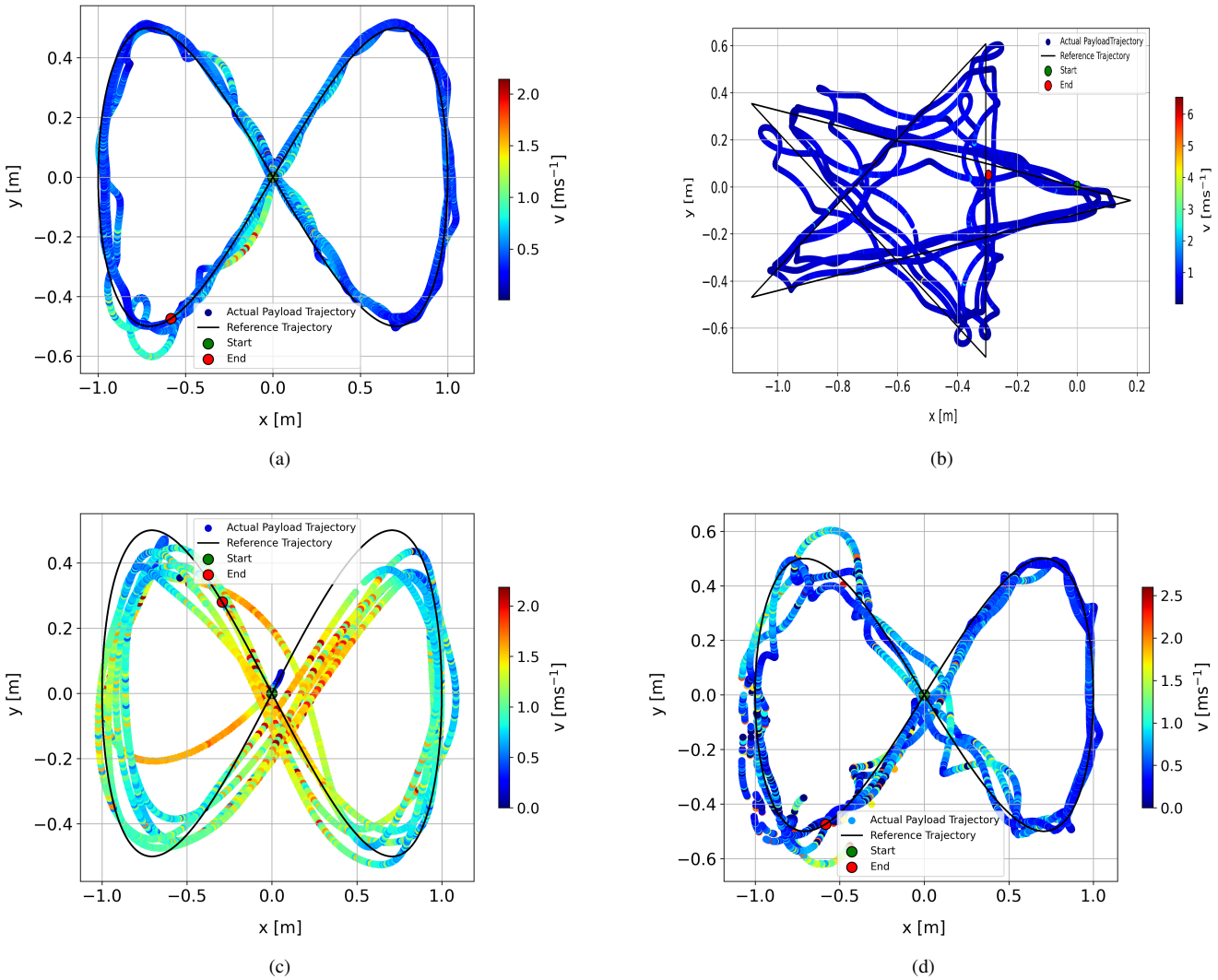


Figure 2. Real-world flight arena experiments evaluating the single-RL policy with a suspended payload: (a) baseline, (b) pentagram trajectory, (c) normal-velocity reference and (d) wind disturbances.

Results

We investigated the performance of the single-RL policy in a real-world flight experiment using Crazyflie 2.1 with a suspended payload. The performance of the single-RL policy was evaluated using different metrics, such as the MED, changes in the swing angle, swing rate, and cable length. These metrics were provided in terms of trajectory generalization, speed robustness, and disturbance resilience. To evaluate the performance metrics of a single RL policy, we conducted four real-world experiments, each containing an average of five trials. The performance of the single RL policy is shown in Fig. 2.

The baseline configuration was defined as a lemniscate trajectory with a low-velocity reference and no wind disturbances. The name of the experiment indicates the feature being tested, which differs from that of baseline experiment.

Addressing **Q1**: We evaluated a single RL policy based on three performance criteria: trajectory, velocity, and wind disturbance, in terms of generalization capability. The results are shown in Fig. 3. The minimum MED error was obtained from the baseline configuration (lemniscate trajectory at a slow velocity with no wind) at 0.107 m. This baseline performance highlights the high fidelity of the policy to the training environment. When tested with the pentagram trajectory, the MED error increased slightly to 0.136 m. This result demonstrates that the single RL policy generalizes effectively across different trajectories without requiring manual tuning. The pentagram trajectory can be described as a situation in which an increase in the error is expected because of its sharp transition points.

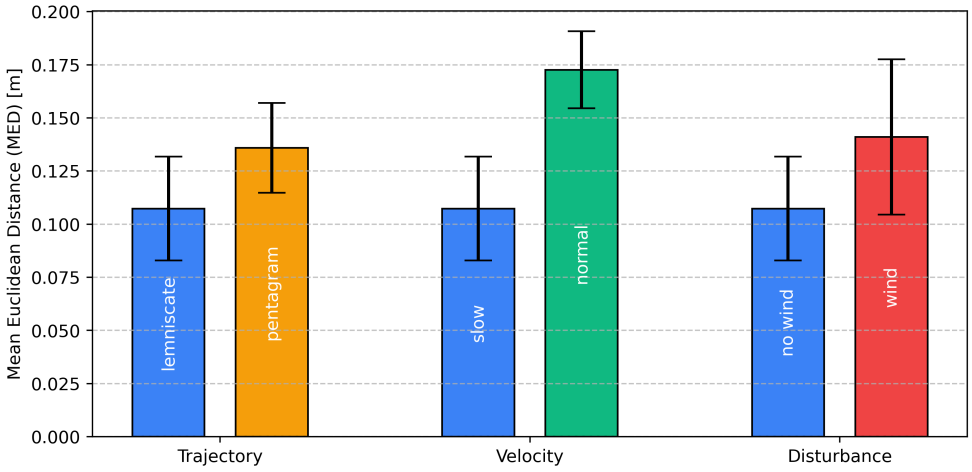


Figure 3. The MED result of the payload-aware single-RL policy obtained from the real-world experiment.

Table 3. The experimental results obtained from performance metrics.

Method	**MED (m)** $\downarrow$	$\bar{\theta}$**(deg)** $\uparrow$	θ_{95}**(deg)** $\uparrow$	$\dot{\theta}_{\mathbf{RMS}}$**(deg/s)** $\downarrow$	$\mathbf{L_{mean}}$**(m)**
baseline	$\mathbf{0.107 \pm 0.02}$	$\mathbf{173.75 \pm 0.9}$	$\mathbf{178.9 \pm 0.26}$	$\mathbf{25.22 \pm 2.5}$	0.296
pentagram	0.136 ± 0.02	165.16 ± 2.86	176.98 ± 0.61	59.24 ± 11.61	0.296
normal-velocity	0.173 ± 0.02	165.9 ± 1.86	176.63 ± 0.19	56.43 ± 9.28	0.296
wind	0.141 ± 0.04	168.92 ± 2.48	178.85 ± 0.14	182.38 ± 88.8	0.289

Addressing **Q2**: The impact of velocity on trajectory tracking performance was investigated with different slow and normal velocity references, such as 0.5 $\mathrm{m \cdot s^{-1}}$ and 1.0 $\mathrm{m \cdot s^{-1}}$. The single RL policy performance produced more MED error as a 0.173 m. This increase indicates that a higher velocity expectation can challenge a single RL policy. Despite the higher mean error, the RL agent remained stable, demonstrating a robust control authority, even under increased dynamic stress. During the normal-velocity experiments, the RL agent did not cause catastrophic failures. Furthermore, as shown in Table 3, $\dot{\theta}_{\mathrm{RMS}}$ value increased when the velocity reference was raised from 0.5 $\mathrm{m \cdot s^{-1}}$ to 1.0 $\mathrm{m \cdot s^{-1}}$. However, there was a slight change in performance relative to θ_{95}, which can be considered within the acceptable performance limits in the context of control authority. Thus, it can be concluded that the policy maintains its damping authority under higher speed excitations.

Addressing **Q3**: Under wind perturbations, the policy demonstrated significant resilience, maintaining a MED of 0.141 m. The results show that although wind disturbances cause fluctuations in the cable length (L) and swing angle (θ) shown in

Table 3, the policy effectively manages these non-stationary environmental factors. The transition between the model-valid (taut) and model-invalid (slack) regimes was observed during the wind trials; however, the policy successfully prevented the regime breakdown by suppressing high-frequency oscillations. As in experiments with different trajectory and velocity references, no catastrophic failures were observed under wind disturbance perturbations.

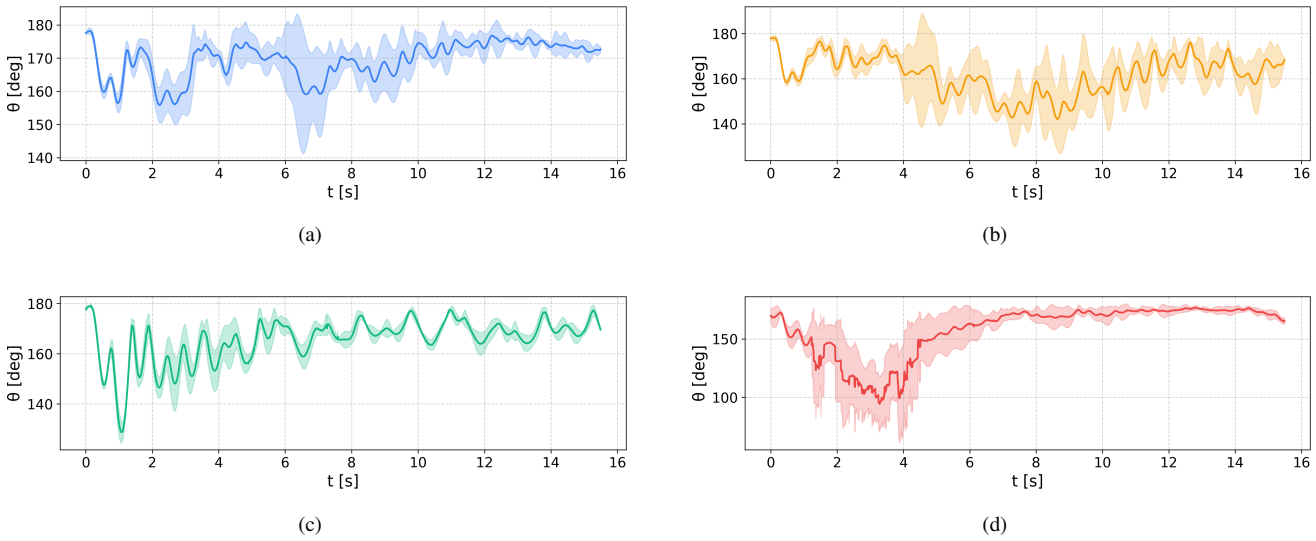


Figure 4. Swing angle changes during the experiments: (a) baseline, (b) pentagram trajectory, (c) normal-velocity reference and (d) wind disturbances.

Fig. 4 shows different information based on damping behavior, crosswind resilience, and dynamic response for the performance of the single RL policy. We observed successful performance owing to a maximum deviation of 6.26° in the baseline configuration. According to this result, we demonstrated that the single RL policy actively worked to dampen and did not generate catastrophic levels in various scenarios. Furthermore, the policy prevents the swing angle from reaching the crash levels despite a non-stationary disturbance of 1.0 $\mathrm{m} \cdot \mathrm{s}^{-1}$ for a crosswind.

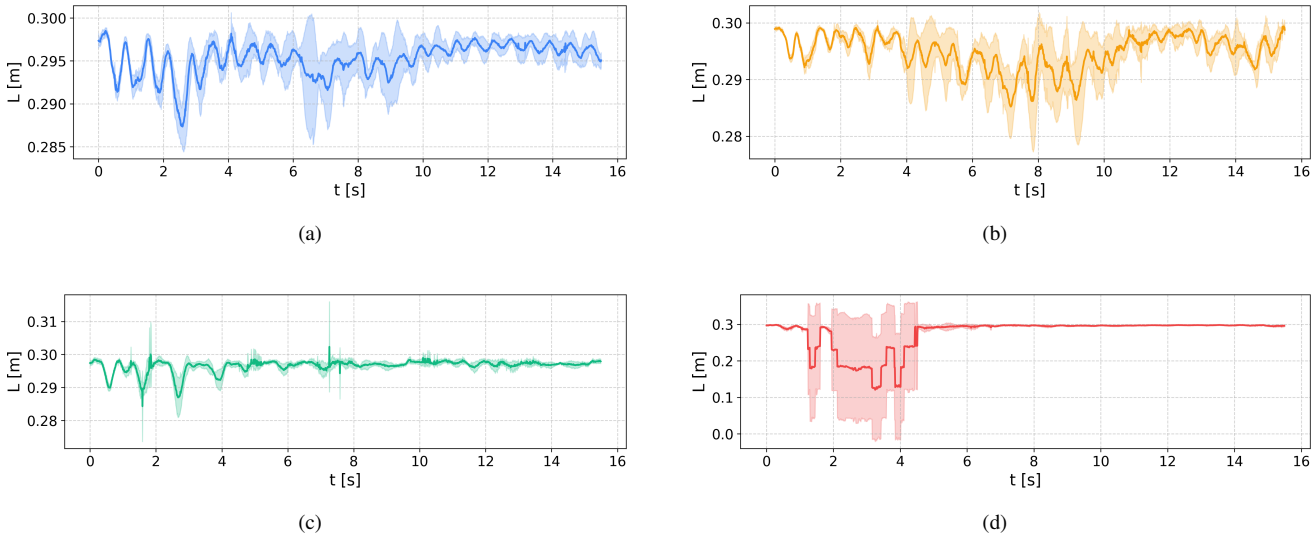


Figure 5. Cable length changes during the experiments: (a) baseline, (b) pentagram trajectory, (c) normal-velocity reference and (d) wind disturbances.

Fig. 5 shows the instantaneous cable length (L), which indicates whether the system is in a taut or slack regime. It is important to evaluate this output to assess the robustness of the single RL policy for mode transitions and for discontinuities. In

the wind disturbance experiments, we observed significant fluctuations in places where it fell well below 0.3 m. This indicates that the cable slackens because of the crosswinds. However, even under these conditions, the policy did not result in catastrophic failures. Furthermore, this output demonstrates that the RL policy learned to manage hybrid dynamics.

Fig. 6 histogram offers a statistical overview of the stability envelope of the single RL policy with respect to operational consistency, generalization stress, and wind perturbation. The distribution in the baseline configuration (a) was tightly clustered around 180°, indicating high precision and low variance, such as 178.9 ± 0.26 obtained from θ_{95}, as listed in Table 3.

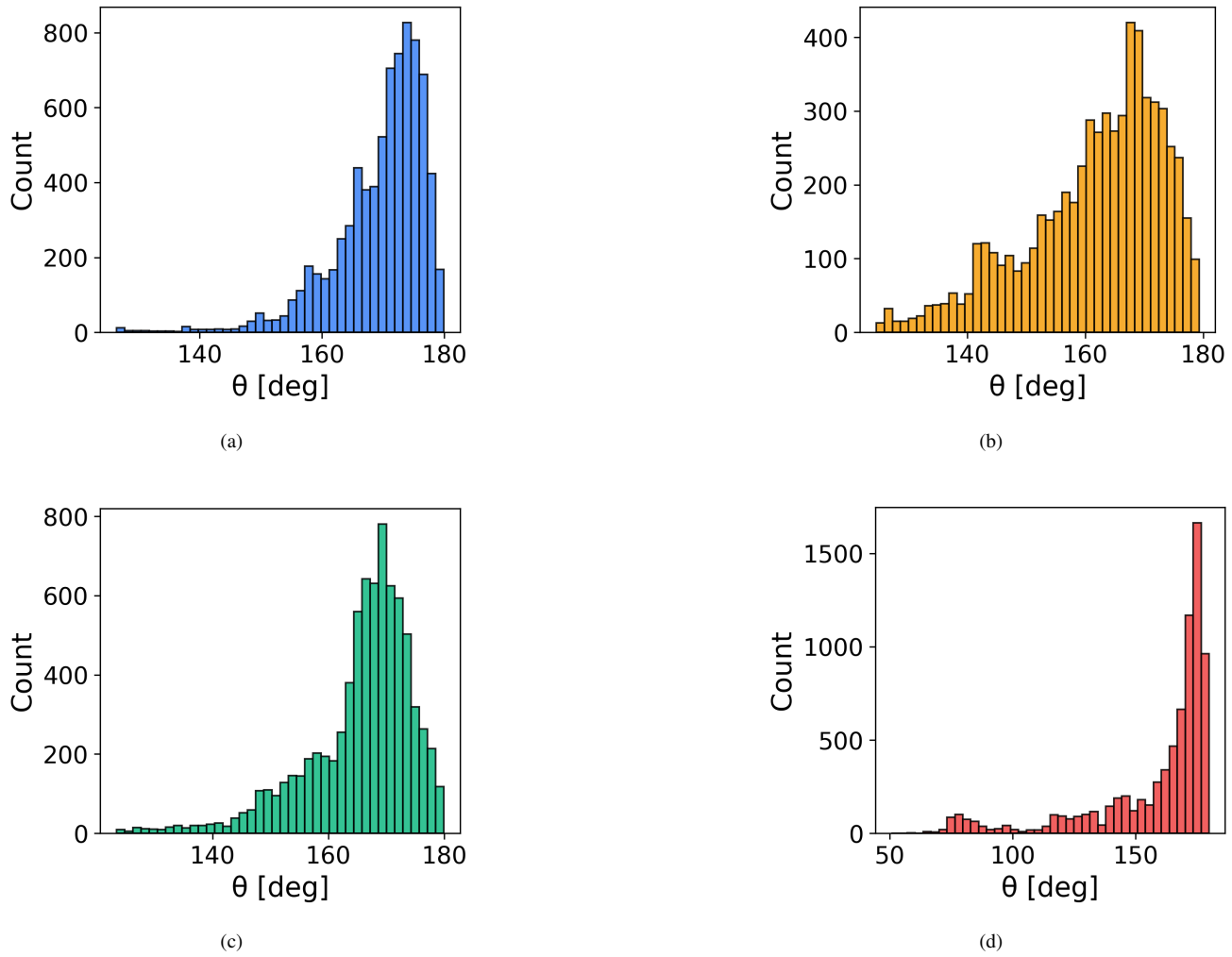


Figure 6. Swing angle histogram during the experiments: (a) baseline, (b) pentagram trajectory, (c) normal-velocity reference and (d) wind disturbances.

As we transitioned to the pentagram (b) and normal speed (c) tasks, the distributions spread further apart from each other. In these experiments, the histogram distribution was considered to represent the limits of the policy for maintaining stability during increased dynamic stress and aggressive maneuvers.

The distribution of wind disturbance (d) showed the most scattered results in real-world experiments with a crosswind configuration. Although the policy is payload-aware, the absence of a predictive wind model in the observation domain prevented the agent from achieving results close to other configurations and maintaining a taut mode of operation. Consequently, the distribution of the swing angles was wider.

Ablation Study

To demonstrate the effectiveness of the payload-aware RL policy, we conducted an ablation study based on hover experiments using two configurations: payload-aware and non-payload-aware RL policies, respectively. For this scenario, we trained two different policies: payload-aware and non-payload-aware hover. We used a similar observation for these policies, for example,

$\{p_r, v_q, R_q\}$. In this observation space, if the selected policy is payload-aware, p_r denotes the relative position of the suspended payload with respect to the target. If the selected policy is non-payload-aware, p_r represents the aerial robot relative to the target.

Table 4. The experimental results of the hover obtained from Mean Euclidean Distance (MED) error

Configuration	MED error	
	Mean [m]	STD [m]
payload-aware	**0.073**	0.036
non-payload-aware	0.158	0.027

Five different experiments were performed for each configuration. As shown in Table 4, we successfully demonstrated the performance of our payload-aware RL policy in this study. The payload-aware RL policy achieved 0.073 m MED error, which is more than twice as accurate as the non-payload-aware MED of 0.158 m. The experimental results are shown in Fig. 7.

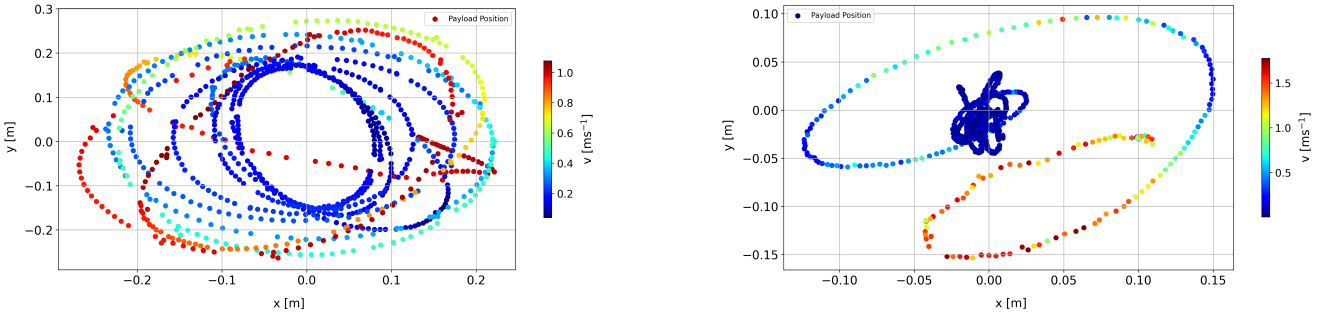


Figure 7. The comparison of the different hover policy: (a) non-payload-aware RL policy, (b) payload-aware RL policy.

According to these results, the performance of the payload-aware RL policy gathered around the origin, indicating a successful experiment. In contrast, it has been observed that the non-payload-aware policy (a) exhibits a much broader and more irregular distribution and fails to compensate for the suspended payload. Additionally, these findings, supported by real-world flight videos (https://tinyurl.com/4uhnbxw2), reveal the performance characteristics of a single RL policy developed for aerial robots.

Discussion

We investigated the fundamental question of "How far can one policy go?". Therefore, we applied different trajectories and velocity references to address this question. We demonstrated the performance of a single RL policy without requiring manual tuning for different problem configurations. The single RL policy maintained a 0.136 m MED error in the pentagram trajectory, although it was not observed during the training. This result originated from focusing on the payload position in the reward function. The policy can perform regardless of the trajectory owing to its highlighted payload position in the reward function.

When the velocity references increased from 0.5 $\mathrm{m \cdot s^{-1}}$ to 1.0 $\mathrm{m \cdot s^{-1}}$, we realized that the single RL policy generated more MED error owing to the increasing centrifugal forces straining the damping capacity. Even in this situation, the policy performed robustly and did not generate crash conditions.

We observed that the cable length fluctuated under wind disturbances and that a slack-mode cable condition occasionally arose. However, the RL policy addressed this problem and maintained a successful performance throughout the experiment without generating crash conditions.

In terms of the operational stability boundary, the swing angle histogram experiments presented an output using only a single RL policy. The swing-angle distribution exhibited an operational stability boundary. In the wind experiments, the policy was more scattered based on the swing angles. If a specific threshold value is present, the operational behavior of the policy under relevant conditions should be examined by considering its distribution.

Conclusion and Future Work

In this study, we explored the comprehensive performance characteristics of a single RL policy for aerial robots with suspended payloads. A single RL policy was evaluated to understand its generalization capability across different trajectory profiles, velocity references, and crosswind disturbances. The test results show that this policy can be generalized to different trajectory profiles that are not observed during the training period, such as pentagram trajectories, and successfully maintain trajectory tracking performance despite the complex transition point of the trajectory. Characterization analysis revealed that the single RL policy maintained stable performance under moderate crosswinds and different reference velocities, even though it generated more MED errors as the dynamic stress increased. Additionally, the policy has prevented failures that could have resulted in catastrophic failures during experiments, even under the cable's transition mode caused by the wind. These findings provide practical insights for developing RL agents in flight for suspended payload applications by mapping the performance characteristics of a single RL policy.

Future work could focus on creating an elastic cable model for suspended loads, updating the reward function design to enhance high-speed performance, and developing a mechanism that allows the trained RL policy to work with different aerial robots. A payload-aware RL approach that deals with uncertainty policies opens up opportunities for obstacle avoidance, agile flight, and complex interaction tasks involving suspended payloads.

Data availability

The datasets used and/or analyzed during the current study are available from the corresponding author upon reasonable request.

References

1. Li, G., Ge, R. & Loianno, G. Cooperative transportation of cable suspended payloads with mavs using monocular vision and inertial sensing. *IEEE Robotics Autom. Lett.* **6**, 5316–5323 (2021).
2. Ho, B., Kocer, B. B. & Kovac, M. Vision based crown loss estimation for individual trees with remote aerial robots. *ISPRS J. Photogramm. Remote. Sens.* **188**, 75–88 (2022).
3. Lan, T., Romanello, L., Kovac, M., Armanini, S. F. & Kocer, B. B. Aerial tensile perching and disentangling mechanism for long-term environmental monitoring. In *2024 IEEE International Conference on Robotics and Automation (ICRA)*, 3827–3833 (IEEE, 2024).
4. Bates, E. *et al.* Leaf level ash dieback disease detection and online severity estimation with uavs. *IEEE Access* (2025).
5. Li, H., Windsor, S. & Kocer, B. B. Treecreeper drone: Adaptive mechanism for passive tree trunk perching. *Adv. Intell. Syst.* 2401101 (2025).
6. Imanberdiyev, N., Sood, S., Kircali, D. & Kayacan, E. Design, development and experimental validation of a lightweight dual-arm aerial manipulator with a cog balancing mechanism. *Mechatronics* **82**, 102719 (2022).
7. Zeng, J., Kotaru, P., Mueller, M. W. & Sreenath, K. Differential flatness based path planning with direct collocation on hybrid modes for a quadrotor with a cable-suspended payload. *IEEE Robotics Autom. Lett.* **5**, 3074–3081 (2020).
8. Li, H. *et al.* Autotrans: A complete planning and control framework for autonomous uav payload transportation. *IEEE Robotics Autom. Lett.* **8**, 6859–6866 (2023).
9. Wang, H., Li, H., Zhou, B., Gao, F. & Shen, S. Impact-aware planning and control for aerial robots with suspended payloads. *IEEE Transactions on Robotics* **40**, 2478–2497 (2024).
10. Recalde, L. F., Sarvaiya, M., Loianno, G. & Li, G. Es-hpc-mpc: Exponentially stable hybrid perception constrained mpc for quadrotor with suspended payloads. *IEEE Robotics Autom. Lett.* **11**, 266–273 (2025).
11. Kober, J., Bagnell, J. A. & Peters, J. Reinforcement learning in robotics: A survey. *The Int. J. Robotics Res.* **32**, 1238–1274 (2013).
12. Cao, D., Zhou, J., Wang, X. & Li, S. Flare: Agile flights for quadrotor cable-suspended payload system via reinforcement learning. *arXiv preprint arXiv:2508.09797* (2025).
13. Zeng, J., Gimenez, A. M., Vinitsky, E., Alonso-Mora, J. & Sun, S. Decentralized aerial manipulation of a cable-suspended load using multi-agent reinforcement learning. *arXiv preprint arXiv:2508.01522* (2025).
14. Chen, J. *et al.* What matters in learning a zero-shot sim-to-real rl policy for quadrotor control? a comprehensive study. *IEEE Robotics Autom. Lett.* (2025).

15. Dionigi, A., Costante, G. & Loianno, G. The power of input: Benchmarking zero-shot sim-to-real transfer of reinforcement learning control policies for quadrotor control. In *2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 11812–11818 (IEEE, 2024).

16. Merk, C. *et al.* Learning real-world acrobatic flight from human preferences. *arXiv preprint arXiv:2508.18817* (2025).

17. Foehn, P. *et al.* Agilicious: Open-source and open-hardware agile quadrotor for vision-based flight. *Sci. robotics* **7**, eabl6259 (2022).

18. Kaufmann, E. *et al.* Champion-level drone racing using deep reinforcement learning. *Nature* **620**, 982–987 (2023).

19. Shen, Y., Franchi, A. & Gabellieri, C. Aerial robots carrying flexible cables: Dynamic shape optimal control via spectral method model. *IEEE Transactions on Robotics* **41**, 3162–3182 (2025).

20. Tang, S., Wüest, V. & Kumar, V. Aggressive flight with suspended payloads using vision-based control. *IEEE Robotics Autom. Lett.* **3**, 1152–1159 (2018).

21. Schulman, J., Wolski, F., Dhariwal, P., Radford, A. & Klimov, O. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347* (2017).

22. Makoviychuk, V. *et al.* Isaac gym: High performance gpu-based physics simulation for robot learning. *arXiv preprint arXiv:2108.10470* (2021).

23. Mittal, M. *et al.* Orbit: A unified simulation framework for interactive robot learning environments. *IEEE Robotics Autom. Lett.* **8**, 3740–3747 (2023).

Acknowledgements

We would like to thank the Bristol Robotics Laboratory for providing access to the flight arena. Ali Tahir Karasahin was also supported by Turkish Scientific and Technological Research Council (Grant Ref. 1059B192302726).

Author contributions

Ali Tahir Karasahin, Ziniu Wu: Conceptualization, Software, Formal analysis, Investigation, Writing - Original Draft, Software, Validation, Data Curation. Basaran Bahadir Kocer: Supervision, Methodology, Resources, Writing - Review & Editing.

Funding

This work is funded and supported by the Turkish Scientific and Technological Research Council (Grant Ref. 1059B192302726).

Declarations

Competing interests

The authors declare no competing interests.