DIFFERENTIAL-POLICY DOUBLE DEEP Q-NETWORK WITH DECOUPLED ARCHITECTURE FOR QUADROTOR ATTITUDE TRACKING

Anonymous authors
Paper under double-blind review

ABSTRACT

This paper presents a differential-policy double deep Q-network (DP-DDQN) for quadrotor attitude tracking. The controller accumulates discrete torque increments into bounded commands and includes the previous command in its observation. Three axis-specific value networks share the full rotational observation, reducing joint-action enumeration to three scalar decisions while retaining information about cross-axis motion. For bounded rewards and exact value iteration, the clipped channel-wise Bellman operators are contractions with unique fixed points. The incremental update also bounds the change in the applied command between control samples. Numerical simulations and hardware-in-the-loop (HITL) experiments evaluate regulation, tilted-gate tracking, and recovery from random initial attitudes. DP-DDQN succeeds in all 200 regulation runs and all 40 gate-crossing runs. In HITL recovery, it reduces the root mean square error from 2.942° to 2.406° and the mean response time from 0.736 s to 0.304 s relative to the PX4 proportional–integral–derivative controller, with every inference completed within the 10-ms control period.

1 INTRODUCTION

Reinforcement learning (RL) offers a data-driven approach to nonlinear control (Ma et al., 2024; Wang et al., 2025a). Alongside proportional–integral–derivative (PID) control and model predictive control (MPC) (Sun et al., 2024; Najafqolian et al., 2024), learned policies provide a way to handle nonlinear dynamics and uncertain operating conditions (Song et al., 2023; Xue et al., 2023). Offline policy optimization and online network inference are particularly relevant to attitude control, where commands must be updated within a short sampling period.

Policy-gradient and actor–critic methods are widely used for continuous control (Wang et al., 2025b; Fujimoto et al., 2018; Haarnoja et al., 2018; Chen et al., 2025). Applications include trajectory planning based on deep deterministic policy gradient and fusion control based on proximal policy optimization (PPO) (Wang et al., 2025b; Chen et al., 2025). Studies on quadrotor unmanned aerial vehicles (UAVs) and variable-pitch micro aerial vehicles (MAVs) combine RL with system identification, domain randomization, curriculum learning, and cascaded control (Hua & Fang, 2023; Wang & Zhao, 2025).

Related aerial-control research addresses disturbance rejection, payload uncertainty, and robust tracking through fixed-time observation, adaptive control, and sliding-mode control (Ma et al., 2021; Muthusamy et al., 2022; Rios et al., 2019). Guided learning and neuroadaptive control support tracking and fault accommodation (Hua et al., 2025; Weng et al., 2026). Quantized control, reconfiguration, and observer-based designs further address actuator and sensor faults (Zhao et al., 2023; Derrouaoui et al., 2023; Salmi et al., 2024). These studies motivate control architectures that combine nonlinear decision making with bounded actuator commands.

Optimization behavior remains an active topic in actor–critic control (Fan et al., 2025; Abdul Hameed et al., 2023). Deep Q-network (DQN) and double deep Q-network (DDQN) methods estimate values for a finite action set (Li et al., 2024; Xiao et al., 2023). Actuator control requires compact action sets with fine effective control resolution. Coarse absolute-command quantization can produce tracking error and chattering, while finer quantization increases the number of

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

actions (Zhu et al., 2025; Łyskawa & Wawrzyński, 2024; Wang et al., 2024). Branching dueling Q-networks (BDQ) assign one action branch to each dimension (Tavakoli et al., 2018), and decoupled Q-networks (DecQN) learn discrete action values by dimension (Seyde et al., 2023).

We propose a differential-policy double deep Q-network (DP-DDQN) that combines incremental torque actions, previous-command augmentation, and axis-specific value networks. Each network observes the complete rotational state and stored command, then selects one torque increment. Accumulation converts these increments into bounded actuator commands.

The contributions cover the control formulation, architecture, and validation. First, the augmented-state formulation yields clipped channel-wise Bellman operators with unique fixed points under bounded rewards and exact value iteration, together with a bound on sampled command variation. Second, three independent networks retain the coupled nine-dimensional observation while reducing the number of action values from M^3 to $3M$ for M levels per axis. Third, matched numerical comparisons and hardware-in-the-loop (HITL) experiments demonstrate attitude regulation, body-clearance-aware gate tracking, and execution on a flight-control processor.

2 Differential-Policy Formulation and Analysis

The differential policy is formulated as an augmented Markov decision process (MDP). Its analysis connects clipped value iteration with bounded changes in the applied control command.

2.1 Augmented State and Bellman Operator

Consider a control-affine nonlinear system with state $\boldsymbol{x} \in \mathcal{X} \subset \mathbb{R}^n$, input $\boldsymbol{u} \in \mathcal{U} = [\boldsymbol{u}_{\min}, \boldsymbol{u}_{\max}] \subset \mathbb{R}^m$, and disturbance $\boldsymbol{d}$. The admissible input set is compact. Sampling at period T_s gives the transition

$$\boldsymbol{x}_{k+1} = \Phi_{T_\mathrm{s}}(\boldsymbol{x}_k, \boldsymbol{u}_k, \boldsymbol{d}_k) \triangleq \mathcal{F}_{T_\mathrm{s}}(\boldsymbol{x}_k, \boldsymbol{u}_k) + \boldsymbol{w}_k, \tag{1}$$

where Φ_{T_s} is the disturbed transition over one control period, $\mathcal{F}_{T_\mathrm{s}}$ is its nominal counterpart, and $\boldsymbol{w}_k$ is a bounded discrete perturbation. The command is held constant between samples.

The differential policy selects an increment $\boldsymbol{a}_k$ from a compact action set $\mathcal{A} \subset \mathbb{R}^m$. The actuator command is obtained by accumulation and projection:

$$\boldsymbol{u}_k = \mathrm{sat}_{\mathcal{U}}(\boldsymbol{u}_{k-1} + \boldsymbol{a}_k), \tag{2}$$

where $\mathrm{sat}_{\mathcal{U}}$ denotes componentwise projection onto $\mathcal{U}$.

For regulation, take $\boldsymbol{s}_k = \boldsymbol{x}_k \in \mathcal{S}$. Equation (2) makes the next command depend on the stored command, so the Markov state is $\tilde{\boldsymbol{s}}_k = (\boldsymbol{s}_k, \boldsymbol{u}_{k-1}) \in \tilde{\mathcal{S}} = \mathcal{S} \times \mathcal{U}$. For a time-varying reference, $\boldsymbol{s}_k$ also contains the reference-generator variables that determine its evolution.

For the roll, pitch, and yaw channels $j \in \{\phi, \theta, \psi\}$, let $\mathcal{A}_j \subset \mathbb{R}$ be a nonempty finite set of scalar increments. The transition kernel $\mathcal{P}_j$ advances the coupled plant after changing channel j and holding the other stored torques. Let $\chi_j(\tilde{\boldsymbol{s}}, a_j, \tilde{\boldsymbol{s}}') \in \{0, 1\}$ denote termination and $\mathcal{R}_j$ the immediate channel reward, with

$$|\mathcal{R}_j(\tilde{\boldsymbol{s}}, a_j, \tilde{\boldsymbol{s}}')| \le R_{\max} < \infty \tag{3}$$

on the operating domain. The target interval is $\mathcal{C} = [y_{\min}, y_{\max}]$, with scalar projection $\Pi_{\mathcal{C}}$. The discount $\gamma \in (0, 1)$ and bootstrap coefficient $\beta > 0$ give the effective discount $\bar{\gamma} = \beta\gamma \in (0, 1)$. For a bounded action-value function $Q_j : \tilde{\mathcal{S}} \times \mathcal{A}_j \to \mathbb{R}$, define

$$\begin{aligned}
&(\mathcal{T}^{\mathcal{C}}_{\Delta,j} Q_j)(\tilde{\boldsymbol{s}}_k, a_{k,j}) \\
&= \mathbb{E}_{\tilde{\boldsymbol{s}}_{k+1} \sim \mathcal{P}_j}\Big[\Pi_{\mathcal{C}}\Big(\mathcal{R}_j(\tilde{\boldsymbol{s}}_k, a_{k,j}, \tilde{\boldsymbol{s}}_{k+1}) \\
&\qquad + \bar{\gamma}\big(1 - \chi_j(\tilde{\boldsymbol{s}}_k, a_{k,j}, \tilde{\boldsymbol{s}}_{k+1})\big) \max_{a' \in \mathcal{A}_j} Q_j(\tilde{\boldsymbol{s}}_{k+1}, a')\Big)\Big].
\end{aligned} \tag{4}$$

2.2 Value Convergence and Command Variation

The reward bound gives an unclipped return bound of $R_{\max}/(1 - \bar{\gamma})$. Projection onto $\mathcal{C}$ is nonexpansive and keeps the target within a fixed interval. These properties yield the following result for exact value iteration.

Theorem 1. *For each channel j, let $\mathscr{Q}_j$ be the space of bounded real-valued functions on $\tilde{\mathcal{S}} \times \mathcal{A}_j$, equipped with the supremum norm $\|\cdot\|_\infty$. The operator $\mathcal{T}^{\mathcal{C}}_{\Delta,j}$ is a $\bar{\gamma}$-contraction on this Banach space. Consequently, $Q_j^{(b+1)} = \mathcal{T}^{\mathcal{C}}_{\Delta,j} Q_j^{(b)}$ converges to a unique fixed point $Q^*_{\Delta,j}$ as the iteration index b increases.*

The complete proof is given in Appendix A.1.

The neural controller approximates these channel-wise values with online and target networks. Each candidate transition retains the coupled plant dynamics; the three selected increments are then applied together. Section 4 specifies the fitted update.

The projection $\mathrm{sat}_{\mathcal{U}}$ in Eq. (2) is nonexpansive in the infinity norm. This property also bounds the rate of change of the physically applied command.

Proposition 1. *For an initial command $\boldsymbol{u}_{-1} \in \mathcal{U}$, the sequence $\{\boldsymbol{u}_k\}_{k=0}^{\infty}$ generated by Eq. (2) has uniformly bounded sample-to-sample variation and is Lipschitz continuous on the sampling grid.*

The complete proof is given in Appendix A.2.

3 Quadrotor Attitude-Tracking Problem

The quadrotor attitude controller uses the sampled rotational state and bounded torque increments.

Appendix B gives the rigid-body model, physical constants, and rotor-inertia approximation used in the numerical and HITL evaluations.

3.1 State, Observation, and Incremental Action

This subsection instantiates the differential policy for UAV inner-loop attitude tracking. The outer loop supplies a bounded attitude reference, while the controller considered here acts on the rotational dynamics.

The rotational state is $\boldsymbol{x}_k = [\boldsymbol{\Theta}_k^{\mathrm{T}}, \boldsymbol{\omega}_k^{\mathrm{T}}]^{\mathrm{T}} \in \mathbb{R}^6$, where $\boldsymbol{\Theta}_k = [\phi_k, \theta_k, \psi_k]^{\mathrm{T}}$ contains the Euler angles and $\boldsymbol{\omega}_k$ contains the body angular rates. The input $\boldsymbol{u}_k = [U_{2,k}, U_{3,k}, U_{4,k}]^{\mathrm{T}}$ contains the roll, pitch, and yaw torques. Channel indices ϕ, θ, ψ correspond to body axes x, y, z, respectively. The sampled attitude dynamics are

$$\boldsymbol{x}_{k+1} = \Phi_{T_{\mathrm{s}}}(\boldsymbol{x}_k, \boldsymbol{u}_k, \boldsymbol{d}_k), \tag{5}$$

where $T_{\mathrm{s}} = 0.01$ s is the 100-Hz observation and control-update period.

For constant-reference regulation, the augmented state is $[\boldsymbol{\Theta}_k^{\mathrm{T}}, \boldsymbol{\omega}_k^{\mathrm{T}}, \boldsymbol{u}_{k-1}^{\mathrm{T}}]^{\mathrm{T}} \in \mathbb{R}^9$. The action is $\boldsymbol{a}_k = [a_{k,\phi}, a_{k,\theta}, a_{k,\psi}]^{\mathrm{T}} \in \mathcal{A}_\phi \times \mathcal{A}_\theta \times \mathcal{A}_\psi$. Each channel has a torque limit $u_j^{\max} > 0$ and admissible interval $\mathcal{U}_j = [-u_j^{\max}, u_j^{\max}]$. Accumulation followed by projection enforces these limits (Wu et al., 2026):

$$\boldsymbol{u}_k = \mathrm{sat}_{\mathcal{U}}\left(\boldsymbol{u}_{k-1} + \boldsymbol{a}_k\right). \tag{6}$$

Here $\mathcal{U} = \mathcal{U}_\phi \times \mathcal{U}_\theta \times \mathcal{U}_\psi$.

For the bounded outer-loop reference $\boldsymbol{\Theta}_k^{\mathrm{ref}}$, define the tracking error $\boldsymbol{e}_k = \boldsymbol{\Theta}_k - \boldsymbol{\Theta}_k^{\mathrm{ref}}$ and network observation $\boldsymbol{o}_k = [\boldsymbol{e}_k^{\mathrm{T}}, \boldsymbol{\omega}_k^{\mathrm{T}}, \boldsymbol{u}_{k-1}^{\mathrm{T}}]^{\mathrm{T}} \in \mathbb{R}^9$. With a fixed reference, this observation is equivalent to the augmented regulation state. Section 2.1 specifies the reference augmentation for a time-varying MDP.

4 Decoupled Control Architecture

Figure 1 connects value learning, channel-wise action selection, and the quadrotor control loop.

4.1 Decoupled Multi-Axis Control Architecture

With M torque levels on each of three axes, joint-action enumeration requires M^3 values. DP-DDQN assigns one network to each axis, giving $3M$ outputs in total. For the seven-level action sets used here, this reduces 343 joint values to 21 channel values.

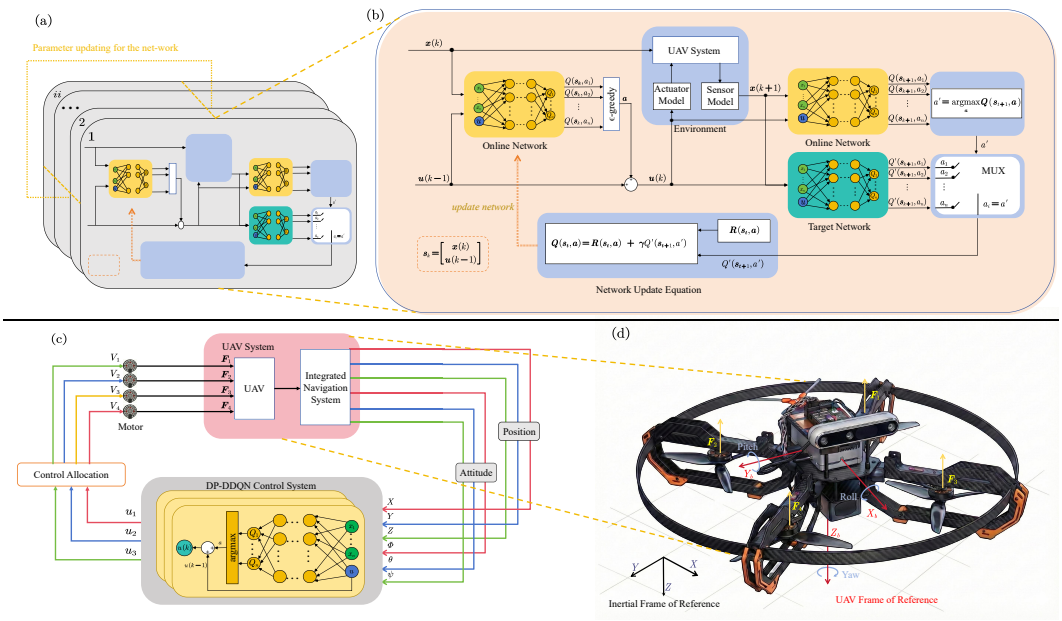

Figure 1: DP-DDQN architecture: (a) training; (b) differential-policy decisions; (c) deployment; (d) UAV system.

All three networks receive the full observation $\boldsymbol{o}_k$. Their independent parameters support axis-specific action selection while the shared angular-rate and error inputs retain information about cross-axis motion.

The channel value functions are

$$Q_j(\boldsymbol{o}_k, a_{k,j}), \qquad j \in \{\phi, \theta, \psi\}. \tag{7}$$

Each network selects one scalar increment; the three components form the joint action $\boldsymbol{a}_k$.

During training, each candidate increment is evaluated by advancing the coupled rotational model with the other torques held at their stored values. This transition retains the gyroscopic term $\boldsymbol{\omega} \times (\boldsymbol{J}\boldsymbol{\omega})$. During control, all three selected increments are applied simultaneously.

4.2 Network Topology and Value Learning

Each network has nine inputs, three $\tanh$ hidden layers with 128, 128, and 64 neurons, and seven linear action-value outputs. The three networks contain 79,509 trainable parameters in total. Denote the online and target parameter vectors by $\boldsymbol{\eta}_j$ and $\boldsymbol{\eta}_j^-$, respectively.

For a channel-j candidate transition from $\boldsymbol{o}_k$ to $\boldsymbol{o}_{k+1}$, the online network selects the next action:

$$a^*_{k+1,j} = \underset{a' \in \mathcal{A}_j}{\operatorname{argmax}}\, Q_j(\boldsymbol{o}_{k+1}, a'; \boldsymbol{\eta}_j). \tag{8}$$

The target network evaluates this action. Let $\chi_{k,j}$ equal one when the sampled state belongs to the channel's near-equilibrium set $\Omega_{0,j}$ or the candidate next state leaves the admissible envelope $\mathcal{D} \subseteq \tilde{\mathcal{S}}$, and zero otherwise. With immediate reward $r_{k,j} = \mathcal{R}_j(\tilde{\boldsymbol{s}}_k, a_{k,j}, \tilde{\boldsymbol{s}}_{k+1})$, the fitted target is

$$Y_{k,j} = \Pi_{\mathcal{C}}\Big[r_{k,j} + (1 - \chi_{k,j})\bar{\gamma} Q_j(\boldsymbol{o}_{k+1}, a^*_{k+1,j}; \boldsymbol{\eta}_j^-)\Big]. \tag{9}$$

Each online network minimizes the mean squared error over the seven targets using adaptive moment estimation (Adam). The separation of action selection and evaluation follows the DDQN update, with projection onto $\mathcal{C}$.

The Bellman training procedure is summarized in Algorithm 1.

Algorithm 1 Decoupled DP-DDQN Training Algorithm

Require: Coupled dynamics, sampled-state distributions, action sets $\mathcal{A}_j$, γ, β, and iteration count B

Initialization:

Initialize $Q_j(\boldsymbol{\eta}_j)$ and target networks $\boldsymbol{\eta}_j^- \leftarrow \boldsymbol{\eta}_j$ for $j \in \{\phi, \theta, \psi\}$.
for $b = 1$ to B **do**
  for each channel $j \in \{\phi, \theta, \psi\}$ **do**
    Draw random one-step states from the coarse, fine, and near-equilibrium strata and add paired observation noise.
    for each $a \in \mathcal{A}_j$ **do**
      Set $u_{k,j} = \mathrm{sat}_{\mathcal{U}_j}(u_{k-1,j} + a)$ and advance the coupled plant for one control interval.
      Compute $r_{k,j}$ and the clipped Double-Q target $Y_{k,j}$.
    end for
    Fit $Q_j(\boldsymbol{\eta}_j)$ to the seven action targets using Adam.
  end for
  Every two iterations, set $\boldsymbol{\eta}_j^- \leftarrow \boldsymbol{\eta}_j$ for all j.
end for

The channel reward combines potential decrease with state, torque, increment, torque-reference, saturation, and terminal penalties. Its complete definition is given in Appendix C.1.

Table 4 specifies the three positive increment magnitudes for each axis. Adding their negatives and zero gives the seven-level action sets.

5 Experimental Evaluation

5.1 Evaluation Setup and Baselines

The evaluation comprises two numerical experiments and one HITL experiment. The nonlinear plant in Section 3, including $\boldsymbol{\omega} \times \boldsymbol{J}\boldsymbol{\omega}$, is integrated at 1 kHz; all inner-loop controllers update at 100 Hz and hold their commands between updates. Matched numerical cases share the initial state, reference, disturbance, actuator noise, and torque limits $[0.12, 0.12, 0.04]^{\mathrm{T}}$ N·m.

The numerical evaluation examines training convergence, parameter sensitivity, and tilted-gate tracking. Comparisons include PID, MPC, PPO, and twin delayed deep deterministic policy gradient (TD3), with configurations in Table 6. PPO and TD3 map the same nine-dimensional observation to three bounded torques and are trained from random initialization.

The learned policies are fixed before gate evaluation. Appendix D gives their training and selection procedures.

The HITL platform uses an STM32F765 flight-control processor. A host computer integrates the nonlinear plant at 1 kHz and supplies inertial, magnetic, barometric, and positioning measurements. State estimation and either the PX4 cascaded attitude/rate PID controller or DP-DDQN execute on the physical processor. The allocated actuator commands return to the host plant, completing the feedback loop.

5.2 Training Convergence and Sensitivity

Figure 2 shows how the three-axis responses and validation residual evolve during training. Responses at iterations 1, 5, 10, and 20 share the initial condition and noise realization. Shading indicates 95% confidence intervals (CIs) across ten independent training runs.

The complete reward constants, sampled-state distributions, and optimization settings are listed in Appendix C.

Ten independently trained policies each undergo 20 matched regulation cases, giving 200 runs. Performance is measured by root mean square error (RMSE) on each axis and the aggregate $E_3 =$

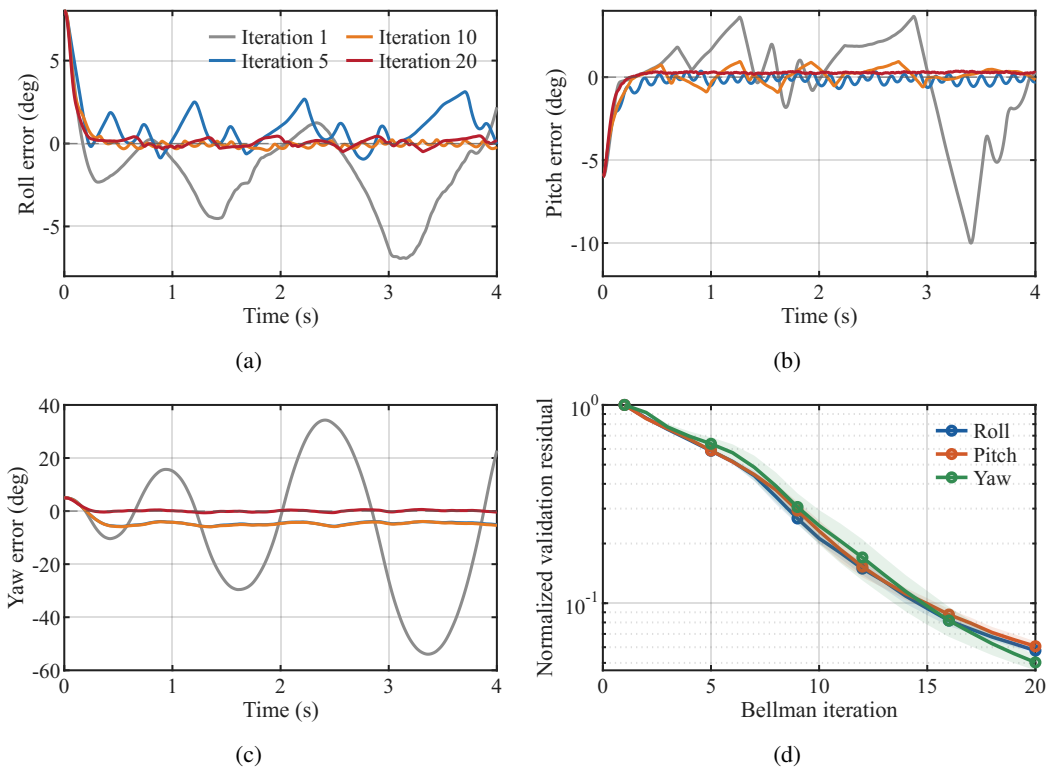

Figure 2: Training convergence: (a) roll error; (b) pitch error; (c) yaw error; (d) normalized validation Bellman residual.

$[(\mathrm{RMSE}_\phi^2 + \mathrm{RMSE}_\theta^2 + \mathrm{RMSE}_\psi^2)/3]^{1/2}$. Appendix C.3 specifies the initial conditions, success criterion, and independent validation metric.

The residual decreases on all axes, and the responses settle near zero by iteration 20. All 200 matched regulation cases meet the success criterion, with roll, pitch, and yaw RMSEs of $1.460 \pm 0.015°$, $1.072 \pm 0.019°$, and $1.164 \pm 0.009°$, and $E_3 = 1.243 \pm 0.008°$.

Each sensitivity setting uses three independently trained policies and the same 20 cases. Three action levels retain zero and the largest symmetric increments; five also retain the smallest. With the reference learning rate 2×10^{-4}, varying the discount factor, progress weight, and action-set size gives $E_3 = 1.230°$–$1.330°$.

Table 5 summarizes the sensitivity results.

5.3 Attitude Tracking Through Tilted Gates

Tilted-gate tracking tests attitude control during translational motion. A common planner supplies a dynamically consistent position–attitude reference. Within ± 1.15 m of the gate, the roll reference follows a cosine profile from zero to the gate tilt and back. The outer position PID uses proportional and derivative gain vectors $\boldsymbol{k}_p^{\mathrm{pos}} = [0.8, 20, 12]^{\mathrm{T}}$ and $\boldsymbol{k}_d^{\mathrm{pos}} = [1, 8, 4]^{\mathrm{T}}$, while each compared controller closes the attitude loop. The 0.80×0.45 m gates have tilts of $10°$, $30°$, $50°$, and $70°$, with ten matched trials per tilt. The cases use still-air conditions and common actuator noise.

A crossing qualifies when the complete rotor envelope fits inside the opening with a 0.03-m edge margin and the three-axis attitude-error norm is at most $5°$. The crossing state is interpolated at the gate plane. The center-to-rotor distance is 0.10 m and the rotor radius is 0.02 m. Gate-local attitude RMSE is evaluated over $|x - x_g| \leq 0.50$ m, where x_g is the gate's longitudinal position.

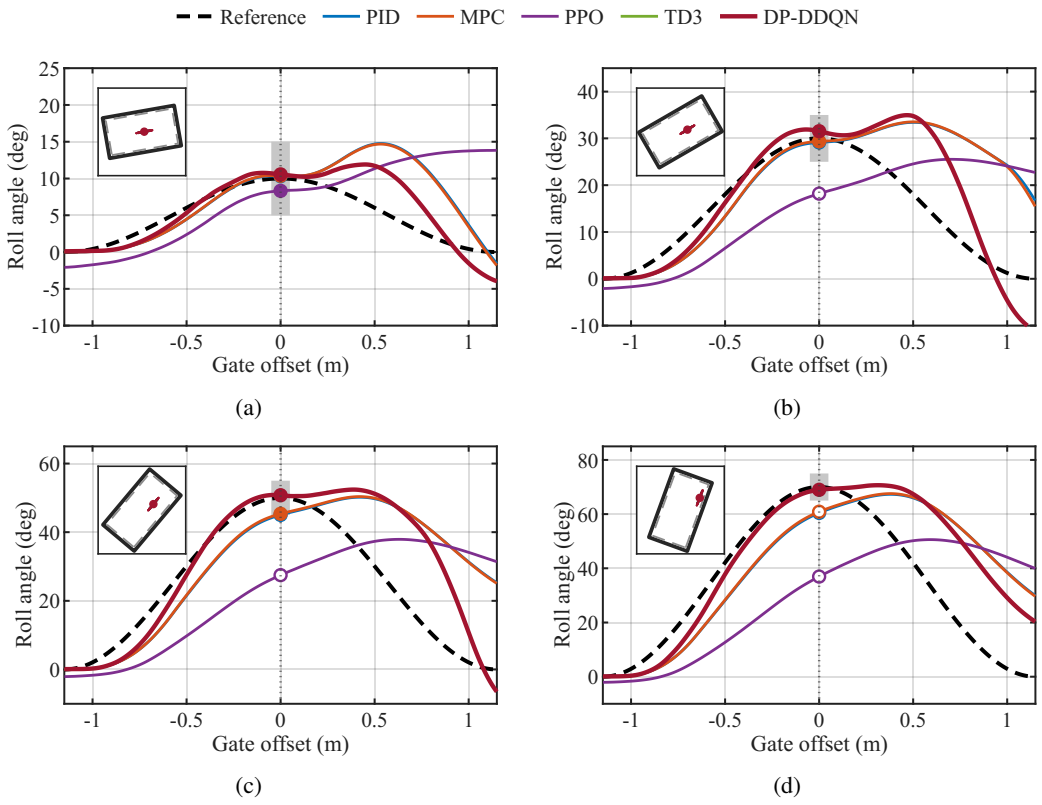

Figure 3: Roll tracking at gate tilts of (a) 10°; (b) 30°; (c) 50°; (d) 70°.

Figure 3 shows one matched case at each tilt. The dashed curve denotes the reference, the vertical dotted line the gate plane, and the gray band the $\pm 5°$ tolerance. Filled and open markers distinguish qualified and unqualified crossings. Insets show the opening, clearance boundary, and DP-DDQN rotor envelope. Table 1 reports all matched cases; dashes indicate metrics unavailable when a trajectory does not reach the gate region.

DP-DDQN qualifies in all 40 cases and attains the lowest gate-local attitude RMSE at every tilt. PID satisfies the crossing criteria through 30°, MPC through 50°, and PPO in eight of ten cases at 10°. Exact 95% Clopper–Pearson intervals are $[0.692, 1]$, $[0.444, 0.975]$, and $[0, 0.308]$ for success counts of 10/10, 8/10, and 0/10. Holm-corrected paired McNemar tests give $p = 0.0078$ against TD3 at all tilts, PPO at 30°–70°, PID at 50°–70°, and MPC at 70°.

The crossing errors and gate-local trajectories in Table 1 show that DP-DDQN combines attitude alignment with complete-body clearance over the four tested tilts.

5.4 HITL Recovery from Random Initial Attitudes

The HITL experiment evaluates zero-attitude recovery over 0°–20°, 20°–40°, and 40°–60°. Each range contains ten matched cases with randomized signs and axes and Euler angles within $\pm 90°$. Each PX4 PID/DP-DDQN pair shares the initial quaternion, plant, estimator, duration, and zero reference; execution order alternates.

A run succeeds if all errors remain within $\pm 2°$ during the final second and no axis exceeds $\pm 10°$ after first entering $\pm 5°$. Response time is the first simultaneous entry into $\pm 5°$; settling time begins when all axes remain within $\pm 2°$ for at least 1 s. Reverse overshoot is measured in degrees as

$$\max\left\{0, -\min_t \frac{\boldsymbol{e}^{\mathsf{T}}(0)\boldsymbol{e}(t)}{\|\boldsymbol{e}(0)\|_2}\right\}.$$

Table 1: Gate-crossing results over ten matched trials per tilt (mean ± standard deviation).

Metric	Controller	10°	30°	50°	70°
Qualified runs	PID	10/10	10/10	0/10	0/10
	MPC	10/10	10/10	10/10	0/10
	PPO	8/10	0/10	0/10	0/10
	TD3	0/10	0/10	0/10	0/10
	DP-DDQN	10/10	10/10	10/10	10/10
Crossing roll error (°)	PID	$\mathbf{0.310 \pm 0.012}$	0.857 ± 0.011	4.966 ± 0.011	9.555 ± 0.012
	MPC	0.376 ± 0.014	$\mathbf{0.635 \pm 0.013}$	4.637 ± 0.013	9.116 ± 0.014
	PPO	1.676 ± 0.126	11.824 ± 0.088	22.519 ± 0.101	33.043 ± 0.109
	TD3	–	–	–	–
	DP-DDQN	0.572 ± 0.055	1.498 ± 0.206	$\mathbf{0.667 \pm 0.087}$	$\mathbf{1.067 \pm 0.086}$
Gate-local attitude RMSE (°)	PID	4.254 ± 0.010	8.804 ± 0.012	12.150 ± 0.012	16.486 ± 0.013
	MPC	4.149 ± 0.012	8.676 ± 0.008	11.942 ± 0.014	16.211 ± 0.016
	PPO	11.192 ± 0.083	18.096 ± 0.080	26.527 ± 0.092	35.266 ± 0.105
	TD3	–	–	–	–
	DP-DDQN	$\mathbf{2.084 \pm 0.151}$	$\mathbf{5.353 \pm 0.044}$	$\mathbf{7.001 \pm 0.085}$	$\mathbf{8.892 \pm 0.136}$

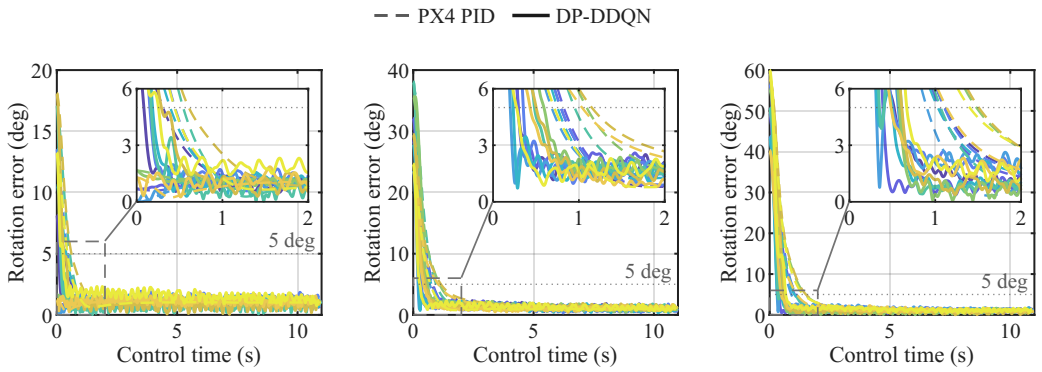

Figure 4: HITL recovery: (a) small (0°–20°); (b) medium (20°–40°); (c) large (40°–60°) initial attitudes; insets magnify 0–2 s and 0°–6°.

Both controllers recover in all 30 cases. DP-DDQN reduces mean RMSE by 0.536° (95% CI $[0.410, 0.663]$°), response time by 0.432 s (95% CI $[0.339, 0.525]$ s), and settling time by 0.392 s (95% CI $[0.001, 0.783]$ s). Table 2 reports the full set of recovery metrics, including reverse overshoot.

The controller operates at a mean update rate of 100.427 Hz. Mean inference time is 5.603 ms, the largest per-run 99th percentile is 5.754 ms, and the maximum is 7.543 ms. Every inference therefore completes within the 10-ms control period.

Figure 4 plots the rotation error $\varepsilon_{\mathrm{att}}(t) = 2 \arccos |q_{e,0}(t)|$ in degrees, where $q_{e,0}$ is the scalar part of the unit quaternion relative to the zero-attitude reference. The tabulated recovery metrics use the Euler-angle error $\boldsymbol{e}$. Within each panel, the same color identifies a common initial quaternion; dashed and solid traces denote PX4 PID and DP-DDQN, respectively.

Table 2: HITL recovery results (mean ± standard deviation; ten matched pairs per attitude range).

Range	Controller	Success	RMSE (°)	Response (s)	Reverse overshoot (°)	Settling (s)
0°–20°	PX4 PID	10/10	1.076 ± 0.441	0.245 ± 0.225	$\mathbf{0.607 \pm 0.450}$	0.467 ± 0.420
	DP-DDQN	10/10	$\mathbf{0.885 \pm 0.270}$	$\mathbf{0.101 \pm 0.100}$	1.107 ± 0.713	$\mathbf{0.233 \pm 0.271}$
20°–40°	PX4 PID	10/10	2.838 ± 0.510	0.840 ± 0.194	$\mathbf{0.308 \pm 0.307}$	1.426 ± 0.569
	DP-DDQN	10/10	$\mathbf{2.258 \pm 0.558}$	$\mathbf{0.350 \pm 0.122}$	2.186 ± 1.488	$\mathbf{1.097 \pm 1.017}$
40°–60°	PX4 PID	10/10	4.912 ± 0.754	1.123 ± 0.231	$\mathbf{0.238 \pm 0.376}$	1.779 ± 0.519
	DP-DDQN	10/10	$\mathbf{4.075 \pm 0.783}$	$\mathbf{0.462 \pm 0.126}$	3.234 ± 3.113	$\mathbf{1.166 \pm 1.097}$
All 30 pairs	PX4 PID	30/30	2.942 ± 1.691	0.736 ± 0.427	$\mathbf{0.384 \pm 0.403}$	1.224 ± 0.746
	DP-DDQN	30/30	$\mathbf{2.406 \pm 1.441}$	$\mathbf{0.304 \pm 0.190}$	2.176 ± 2.152	$\mathbf{0.832 \pm 0.951}$

6 Conclusion

DP-DDQN combines discrete torque increments with previous-command augmentation and three axis-specific value networks. The resulting controller retains the coupled rotational observation while making three scalar action decisions. The analysis establishes unique fixed points for the clipped channel-wise value operators and a bound on sampled command variation.

Matched evaluations demonstrate repeatable three-axis regulation, successful gate tracking across four tilts, and random-attitude recovery on a flight-control processor. DP-DDQN succeeds in all 200 regulation cases and all 40 gate crossings, with the lowest gate-local attitude RMSE among the compared controllers. In HITL recovery, it reduces RMSE from $2.942°$ to $2.406°$ and mean response time from 0.736 s to 0.304 s. All inferences complete within the 10-ms sampling period, supporting its use for real-time attitude control.

Artificial intelligence use statement

Generative artificial intelligence (AI) assisted manuscript editing, checks of mathematical notation, and submission formatting. The method, theoretical results, implementation, and experiments were developed before this editorial assistance and without generative AI.

Reproducibility statement

The channel-wise operator and network architecture are specified in Sections 2 and 4; complete proofs are provided in Appendix A. The plant equations and physical constants are given in Appendix B, and the training settings and sensitivity results in Appendix C. Section 5 specifies the comparison conditions, matched trials, success criteria, and numerical and hardware-in-the-loop results.

References

Mohammed Sharafath Abdul Hameed, Gavneet Singh Chadha, Andreas Schwung, and Steven X. Ding. Gradient Monitored Reinforcement Learning. *IEEE Trans. Neural Netw. Learn. Syst.*, 34 (8):4106–4119, 2023. ISSN 2162-237X, 2162-2388. doi: 10.1109/TNNLS.2021.3119853.

Zongliang Chen, Shuguo Pan, Kegen Yu, Yuting Wu, Wang Gao, Zhuoxuan Wang, and Xiaolin Meng. Fusion Control Tracking Strategy for Autonomous Vehicles: A Fast PPO Reinforcement Learning Based on Attention Mechanism and Physical Information. *IEEE Trans. Intell. Transp. Syst.*, 26(11):18906–18920, 2025. ISSN 1524-9050, 1558-0016. doi: 10.1109/TITS.2025.3609483.

Saddam Hocine Derrouaoui, Yasser Bouzid, Amina Belmouhoub, and Mohamed Guiatni. Enhanced nonlinear adaptive control of a novel over-actuated reconfigurable quadcopter. In *2023 International Conference on Unmanned Aircraft Systems (ICUAS)*, pp. 229–234. IEEE, 2023. doi: 10.1109/ICUAS57906.2023.10155908.

Quan-Yong Fan, Meiying Cai, and Bin Xu. An Improved Prioritized DDPG Based on Fractional-Order Learning Scheme. *IEEE Trans. Neural Netw. Learn. Syst.*, 36(4):6873–6882, 2025. ISSN 2162-237X, 2162-2388. doi: 10.1109/TNNLS.2024.3395508.

Scott Fujimoto, Herke van Hoof, and David Meger. Addressing function approximation error in actor-critic methods. In *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pp. 1587–1596. PMLR, 2018.

Tuomas Haarnoja, Aurick Zhou, Pieter Abbeel, and Sergey Levine. Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor. In *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pp. 1861–1870. PMLR, 2018.

Hean Hua and Yongchun Fang. A novel reinforcement learning-based robust control strategy for a quadrotor. *IEEE Trans. Ind. Electron.*, 70(3):2812–2821, Mar. 2023. doi: 10.1109/TIE.2022.3165288.

Hean Hua, Yaonan Wang, Hang Zhong, Hui Zhang, and Yongchun Fang. A novel guided deep reinforcement learning tracking control strategy for multirotors. *IEEE Trans. Autom. Sci. Eng.*, 22:2062–2074, 2025. doi: 10.1109/TASE.2024.3374752.

Taeyoung Lee. Robust adaptive attitude tracking on $\mathrm{so}(3)$ with an application to a quadrotor UAV. *IEEE Trans. Control Syst. Technol.*, 21(5):1924–1930, 2013. doi: 10.1109/TCST.2012.2209887.

Dongfang Li, Jiechao Zhou, Jie Huang, Dali Zhang, Ping Li, Rob Law, and Edmond Q. Wu. State prediction and Anti-Interference-Based flight path-following for UAVs. *IEEE Trans. Intell. Transp. Syst.*, 24(12):15236–15247, 2023. doi: 10.1109/TITS.2023.3308932.

Rui Li, Wenyin Gong, Ling Wang, Chao Lu, Zixiao Pan, and Xinying Zhuang. Double DQN-Based Coevolution for Green Distributed Heterogeneous Hybrid Flowshop Scheduling With Multiple Priorities of Jobs. *IEEE Trans. Autom. Sci. Eng.*, 21(4):6550–6562, 2024. ISSN 1545-5955, 1558-3783. doi: 10.1109/TASE.2023.3327792.

Yanhong Luo, Hang Yu, Huaguang Zhang, and Yu Zhou. A novel Newton–Euler method-based nonlinear anti-swing control for a quadrotor UAV carrying a slung load. *IEEE Trans. Syst. Man Cybern. Syst.*, 54(4):2266–2275, 2024. doi: 10.1109/TSMC.2023.3342027.

Dailiang Ma, Yuanqing Xia, Ganghui Shen, Hongru Jiang, and Chenxi Hao. Practical fixed-time disturbance rejection control for quadrotor attitude tracking. *IEEE Trans. Ind. Electron.*, 68(8): 7274–7283, Aug. 2021. doi: 10.1109/TIE.2020.3001800.

Yong-Sheng Ma, Yong Xu, Jian Sun, and Li-Hua Dou. Data-driven optimal cooperative tracking control for heterogeneous multi-agent systems. *ISA Trans.*, 154:23–31, November 2024. ISSN 00190578. doi: 10.1016/j.isatra.2024.08.026.

Praveen Kumar Muthusamy, Matthew Garratt, Hemanshu Pota, and Rajkumar Muthusamy. Real-time adaptive intelligent control system for quadcopter unmanned aerial vehicles with payload uncertainties. *IEEE Trans. Ind. Electron.*, 69(2):1641–1653, Feb. 2022. doi: 10.1109/TIE.2021.3055170.

Mohamad Amin Najafqolian, Khalil Alipour, Roujin Mousavifard, and Bahram Tarvirdizadeh. Control of aerial robots using convex QP LMPC and learning-based Explicit-MPC. *IEEE Trans. Ind. Informat.*, 20(9):10883–10891, 2024. doi: 10.1109/TII.2024.3397353.

Hector Rios, Romeo Falcon, Oscar A. Gonzalez, and Alejandro Dzul. Continuous sliding-mode control strategies for quadrotor robust tracking: Real-time application. *IEEE Trans. Ind. Electron.*, 66(2):1264–1272, Feb. 2019. doi: 10.1109/TIE.2018.2831191.

Abdenour Salmi, Mohamed Guiatni, Yasser Bouzid, Saddam Hocine Derrouaoui, and Far 'es Boudjema. Fault tolerant control based on thau observer of a reconfigurable quadrotor with total loss of actuator. *Unmanned Systems*, 12(4):667–683, 2024. doi: 10.1142/S2301385024500146.

Tim Seyde, Peter Werner, Wilko Schwarting, Igor Gilitschenski, Martin Riedmiller, Daniela Rus, and Markus Wulfmeier. Solving continuous control via Q-learning. In *Proc. Int. Conf. Learn. Representations*, 2023. URL https://openreview.net/forum?id=U5XOGxAgccS.

Xinfeng Shao and Dan Ye. Robust adaptive dynamic memory-event-triggered attitude control for nonlinear multi-UAVs resist actuator hysteresis. *Int. J. Robust Nonlinear Control*, 33:8315–8335, 2023. doi: 10.1002/rnc.6821.

Fulin Song, Zhan Li, Sichen Yang, and Juan J. Rodriguez-Andina. Anti-Disturbance Compensation for Quadrotor Close Crossing Flight Based on Deep Reinforcement Learning. *IEEE Trans. Ind. Electron.*, 70(3):3013–3023, 2023. ISSN 0278-0046, 1557-9948. doi: 10.1109/TIE.2022.3172764.

Haidi Sun, Li Dai, and Peizhan Wang. An efficient moving obstacle avoidance scheme for UAVs via output feedback robust MPC. *IEEE Trans. Aerosp. Electron. Syst.*, 60(5):6199–6212, 2024. doi: 10.1109/TAES.2024.3401094.

Arash Tavakoli, Fabio Pardo, and Petar Kormushev. Action branching architectures for deep reinforcement learning. In *Proc. AAAI Conf. Artif. Intell.*, volume 32, pp. 4131–4138, 2018. doi: 10.1609/aaai.v32i1.11798.

Huanqing Wang, Muxuan Li, Huaguang Zhang, and Siwen Liu. Predefined-time composite adaptive fuzzy nonsingular attitude control for multi-uavs. *IEEE Trans. Fuzzy Syst.*, 32(12):6891–6903, 2024. doi: 10.1109/TFUZZ.2024.3466934.

Xiaolong Wang, Jianfu Cao, Ye Cao, and Feng Zou. Energy-efficient trajectory planning for a class of industrial robots using parallel deep reinforcement learning. *Nonlinear Dyn.*, 113(8):8491–8511, 2025a. ISSN 0924-090X, 1573-269X. doi: 10.1007/s11071-024-10510-4.

Zhikun Wang and Shiyu Zhao. Sim-to-real transfer in reinforcement learning for maneuver control of a variable-pitch MAV. *IEEE Trans. Ind. Electron.*, 72(10):10445–10454, Oct. 2025. doi: 10.1109/TIE.2025.3558030.

Zhikun Wang, Sen Yang, Tian Xie, Jigang Tong, and Wei Hao. Trajectory planning of quadrotor elastic suspension system based on DDPG algorithm. *ISA Trans.*, 167:2063–2077, December 2025b. ISSN 00190578. doi: 10.1016/j.isatra.2025.10.037.

Yan Weng, Zhijia Zhao, Di Zhang, Zhijie Liu, and Keum-Shik Hong. Neuro-learning based fault-tolerant control of a 2-dof helicopter system with sensor fault and unknown dead zone. *IEEE Trans. Ind. Electron.*, 73(6):8946–8957, 2026. doi: 10.1109/TIE.2026.3654623.

Tong Wu, Weifeng Du, Jianan Yang, Ming Liu, Lixian Zhang, and Guangren Duan. Robust and efficient model predictive attitude control for hybrid terrestrial-aerial quadrotors. *IEEE Trans. Ind. Electron.*, 73(2):2588–2599, 2026. doi: 10.1109/TIE.2025.3603069.

Hao Xiao, Xiaowei Pu, Wei Pei, Li Ma, and Tengfei Ma. A Novel Energy Management Method for Networked Multi-Energy Microgrids Based on Improved DQN. *IEEE Trans. Smart Grid*, 14(6): 4912–4926, 2023. ISSN 1949-3053, 1949-3061. doi: 10.1109/TSG.2023.3261979.

Zhewen Xing, Youmin Zhang, and Chun-Yi Su. Active wind rejection control for a quadrotor UAV against unknown winds. *IEEE Trans. Aerosp. Electron. Syst.*, 59(6):8956–8968, 2023. doi: 10.1109/TAES.2023.3315254.

Jun Xue, Ziniu Liu, Guanjun Liu, Ziyuan Zhou, Kaiwen Zhang, Ying Tang, and Jiacun Wang. Robust wind-resistant hovering control of quadrotor UAVs using deep reinforcement learning. *IEEE Trans. Intell. Veh.*, pp. 1–10, 2023. doi: 10.1109/TIV.2023.3324687.

Zhijia Zhao, Jian Zhang, Zhijie Liu, Wei He, and Keum-Shik Hong. Adaptive quantized fault-tolerant control of a 2-dof helicopter system with actuator fault and unknown dead zone. *Automatica*, 148:110792, 2023. doi: 10.1016/j.automatica.2022.110792.

Yuanyang Zhu, Zhi Wang, Yuanheng Zhu, Chunlin Chen, and Dongbin Zhao. Discretizing Continuous Action Space With Unimodal Probability Distributions for On-Policy Reinforcement Learning. *IEEE Trans. Neural Netw. Learn. Syst.*, 36(6):11285–11297, 2025. ISSN 2162-237X, 2162-2388. doi: 10.1109/TNNLS.2024.3446371.

Jakub Łyskawa and Paweł Wawrzyński. ACERAC: Efficient Reinforcement Learning in Fine Time Discretization. *IEEE Trans. Neural Netw. Learn. Syst.*, 35(2):2719–2731, 2024. ISSN 2162-237X, 2162-2388. doi: 10.1109/TNNLS.2022.3190973.

Table 3: Quadrotor parameters and time steps.

Parameter	Symbol	Value	Unit
Quadrotor mass	m_{u}	0.5	kg
Gravitational acceleration	g	9.81	m/s^2
Moment of inertia (x-axis)	I_{xx}	0.0035	kg·m^2
Moment of inertia (y-axis)	I_{yy}	0.003	kg·m^2
Moment of inertia (z-axis)	I_{zz}	0.0058	kg·m^2
Rotor inertia	J_{r}	0.0001	kg·m^2
Residual rotor-speed sum	Ω_{r}	100	rad/s
Integration/control periods	$\delta t/T_{\mathrm{s}}$	0.001/0.010	s

A Complete Proofs

A.1 Contraction of the channel-wise operator

Proof. For any $Q_1, Q_2 \in \mathscr{Q}_j$, nonexpansiveness of the projection and maximum operators, together with $0 \le 1-\chi_j \le 1$, gives

$$\begin{aligned}
&\|\mathcal{T}^{\mathcal{C}}_{\Delta,j}Q_1 - \mathcal{T}^{\mathcal{C}}_{\Delta,j}Q_2\|_\infty \\
&\quad\le \bar\gamma \sup_{\tilde{\boldsymbol{s}},a_j} \mathbb{E}_{\tilde{\boldsymbol{s}}'\sim\mathcal{P}_j}\left[\max_{a'\in\mathcal{A}_j} |Q_1(\tilde{\boldsymbol{s}}',a') - Q_2(\tilde{\boldsymbol{s}}',a')|\right] \\
&\quad\le \bar\gamma \|Q_1 - Q_2\|_\infty.
\end{aligned} \tag{10}$$

Because $\bar\gamma \in (0,1)$, the contraction mapping theorem gives the stated unique fixed point and convergence. □

A.2 Bounded sampled-command variation

Proof. Starting from $\boldsymbol{u}_{-1} \in \mathcal{U}$, projection in Eq. (2) gives $\boldsymbol{u}_k \in \mathcal{U}$ for all $k \ge 0$. Let $t_k = kT_{\mathrm{s}}$ and $C_{\max} = \max_{\boldsymbol{a}\in\mathcal{A}} \|\boldsymbol{a}\|_\infty$, which is finite because $\mathcal{A}$ is compact. Nonexpansiveness of the projection gives

$$\begin{aligned}
\|\boldsymbol{u}_k - \boldsymbol{u}_{k-1}\|_\infty &= \|\,\mathrm{sat}_{\mathcal{U}}(\boldsymbol{u}_{k-1} + \boldsymbol{a}_k) \\
&\quad - \mathrm{sat}_{\mathcal{U}}(\boldsymbol{u}_{k-1})\|_\infty \\
&\le \|\boldsymbol{a}_k\|_\infty \le C_{\max}.
\end{aligned} \tag{11}$$

For $k_1 > k_2 \ge 0$, summing the single-step bounds yields

$$\begin{aligned}
\|\boldsymbol{u}_{k_1} - \boldsymbol{u}_{k_2}\|_\infty &\le \sum_{i=k_2+1}^{k_1} \|\boldsymbol{u}_i - \boldsymbol{u}_{i-1}\|_\infty \\
&\le (k_1 - k_2) C_{\max}.
\end{aligned} \tag{12}$$

Using $k_1 - k_2 = |t_{k_1} - t_{k_2}|/T_{\mathrm{s}}$ in the summation bound gives

$$\|\boldsymbol{u}(t_{k_1}) - \boldsymbol{u}(t_{k_2})\|_\infty \le \frac{C_{\max}}{T_{\mathrm{s}}} |t_{k_1} - t_{k_2}|. \tag{13}$$

Thus $L = C_{\max}/T_{\mathrm{s}}$ is a Lipschitz constant on the sampling grid. □

B Quadrotor Model and Physical Constants

B.1 Quadrotor UAV Dynamics Model

The quadrotor is a rigid cruciform body actuated by four rotors. The inertial frame $\mathcal{E} = \{O_{\mathrm{E}}, X_{\mathrm{E}}, Y_{\mathrm{E}}, Z_{\mathrm{E}}\}$ has a ground-fixed origin and a downward vertical axis. The body frame $\mathcal{B} = \{O_{\mathrm{B}}, X_{\mathrm{B}}, Y_{\mathrm{B}}, Z_{\mathrm{B}}\}$ is centered at the vehicle's center of mass. The matrix $\boldsymbol{R}^{\mathrm{E}}_{\mathrm{B}} \in \mathrm{SO}(3)$

maps body-frame vectors to the inertial frame (Lee, 2013), where $\mathrm{SO}(3)$ is the three-dimensional special orthogonal group. Using the Z–Y–X Euler sequence $\boldsymbol{\Theta} = [\phi, \theta, \psi]^{\mathrm{T}}$ and the shorthand $c_\alpha = \cos\alpha$, $s_\alpha = \sin\alpha$, and $t_\alpha = \tan\alpha$, the rotation matrix is

$$\boldsymbol{R}_{\mathrm{B}}^{\mathrm{E}} = \begin{bmatrix} c_\theta c_\psi & s_\phi s_\theta c_\psi - c_\phi s_\psi & c_\phi s_\theta c_\psi + s_\phi s_\psi \\ c_\theta s_\psi & s_\phi s_\theta s_\psi + c_\phi c_\psi & c_\phi s_\theta s_\psi - s_\phi c_\psi \\ -s_\theta & s_\phi c_\theta & c_\phi c_\theta \end{bmatrix}. \tag{14}$$

With gravitational acceleration g and collective thrust U_1, the translational dynamics are

$$\ddot{\boldsymbol{\xi}} = [0, 0, g]^{\mathrm{T}} - \frac{U_1}{m_{\mathrm{u}}} \boldsymbol{R}_{\mathrm{B}}^{\mathrm{E}} \boldsymbol{b}_z, \tag{15}$$

where $\boldsymbol{\xi} = [x, y, z]^{\mathrm{T}}$ is position, m_{u} is mass, and $\boldsymbol{b}_z = [0, 0, 1]^{\mathrm{T}}$ is the third coordinate unit vector. In scalar form,

$$\begin{cases} \ddot{x} = -\dfrac{U_1}{m_{\mathrm{u}}}(c_\phi s_\theta c_\psi + s_\phi s_\psi), \\ \ddot{y} = -\dfrac{U_1}{m_{\mathrm{u}}}(c_\phi s_\theta s_\psi - s_\phi c_\psi), \\ \ddot{z} = g - \dfrac{U_1}{m_{\mathrm{u}}}(c_\phi c_\theta). \end{cases} \tag{16}$$

The complete rotational model includes rigid-body gyroscopic coupling (Li et al., 2023) and rotor inertia J_{r} (Xing et al., 2023), as described by Euler's equations (Shao & Ye, 2023; Luo et al., 2024)

$$\dot{\boldsymbol{\omega}} = \boldsymbol{J}^{-1} \left(\boldsymbol{\tau} - \boldsymbol{\omega} \times (\boldsymbol{J}\boldsymbol{\omega}) - J_{\mathrm{r}} \boldsymbol{\omega} \times \boldsymbol{\Omega}_{\mathrm{sum}} \right), \tag{17}$$

where $\boldsymbol{\omega} = [\omega_x, \omega_y, \omega_z]^{\mathrm{T}}$ contains the body angular rates, $\boldsymbol{\tau} = [U_2, U_3, U_4]^{\mathrm{T}}$ is the torque vector with $\boldsymbol{\tau}(t_k) = \boldsymbol{u}_k$, and $\boldsymbol{\Omega}_{\mathrm{sum}} = [0, 0, \Omega_{\mathrm{r}}]^{\mathrm{T}}$. For the rotor numbering and spin convention used here, $\Omega_{\mathrm{r}} = \Omega_1 - \Omega_2 + \Omega_3 - \Omega_4$, where Ω_i is rotor i's angular speed. The inertia matrix is

$$\boldsymbol{J} = \begin{bmatrix} I_{xx} & 0 & 0 \\ 0 & I_{yy} & 0 \\ 0 & 0 & I_{zz} \end{bmatrix}. \tag{18}$$

Expanding Eq. (17), the coupled rotational dynamics are obtained as

$$\begin{cases} \dot{\omega}_x = \dfrac{1}{I_{xx}} \left[U_2 - \omega_y \omega_z (I_{zz} - I_{yy}) - J_{\mathrm{r}} \omega_y \Omega_{\mathrm{r}} \right], \\ \dot{\omega}_y = \dfrac{1}{I_{yy}} \left[U_3 - \omega_x \omega_z (I_{xx} - I_{zz}) + J_{\mathrm{r}} \omega_x \Omega_{\mathrm{r}} \right], \\ \dot{\omega}_z = \dfrac{1}{I_{zz}} \left[U_4 - \omega_x \omega_y (I_{yy} - I_{xx}) \right]. \end{cases} \tag{19}$$

The numerical and HITL plant uses the rigid-body approximation $J_{\mathrm{r}} \boldsymbol{\omega} \times \boldsymbol{\Omega}_{\mathrm{sum}} = \mathbf{0}$ and retains the coupling term $\boldsymbol{\omega} \times (\boldsymbol{J}\boldsymbol{\omega})$.

Body angular rates and Euler-angle rates satisfy $\dot{\boldsymbol{\Theta}} = \boldsymbol{H}(\boldsymbol{\Theta})\boldsymbol{\omega}$, where

$$\boldsymbol{H}(\boldsymbol{\Theta}) = \begin{bmatrix} 1 & s_\phi t_\theta & c_\phi t_\theta \\ 0 & c_\phi & -s_\phi \\ 0 & s_\phi / c_\theta & c_\phi / c_\theta \end{bmatrix}. \tag{20}$$

The model parameters and time steps are listed in Table 3.

C Training Details and Sensitivity Results

C.1 Three-Axis Reward Function

For channel j, let e_j^{sc} and ω_j^{sc} be the error and angular-rate scales. Define the error surface and normalized potential by

$$\begin{aligned} \sigma_{k,j} &= \omega_{k,j} + \kappa_j e_{k,j}, \\ V_{k,j} &= \left(\frac{e_{k,j}}{e_j^{\mathrm{sc}}} \right)^2 + w_\sigma \left(\frac{\sigma_{k,j}}{\omega_j^{\mathrm{sc}}} \right)^2. \end{aligned} \tag{21}$$

Table 4: DP-DDQN torque limits and positive action increments.

Axis	Torque limit (N·m)	Positive increment magnitudes (N·m)
Roll	0.12	0.100, 0.050, 0.010
Pitch	0.12	0.120, 0.100, 0.050
Yaw	0.04	0.010, 0.005, 0.001

Table 5: One-factor sensitivity of three-axis regulation error E_3.

Factor	Candidate A	Candidate B	Reference setting
Learning rate	$10^{-4}:\ 9.001 \pm 32.718$	$4 \times 10^{-4}:\ 1.250 \pm 0.020$	$2 \times 10^{-4}:\ 1.235 \pm 0.022$
Discount factor γ	$0.88:\ 1.237 \pm 0.039$	$0.96:\ 1.232 \pm 0.016$	$0.92:\ 1.235 \pm 0.022$
Progress weight w_p	$5:\ 1.230 \pm 0.039$	$15:\ 1.254 \pm 0.016$	$10:\ 1.235 \pm 0.022$
Action levels per axis	$3:\ 1.330 \pm 0.065$	$5:\ 1.240 \pm 0.028$	$7:\ 1.235 \pm 0.022$

Here $\kappa_j > 0$ is the error-surface gain and w_σ weights its contribution to the potential. Let $A_j = \max_{a \in \mathcal{A}_j} |a|$. The normalized torque and increment are $\bar{u}_{k,j} = u_{k,j}/u_j^{\max}$ and $\bar{a}_{k,j} = a_{k,j}/A_j$. The near-limit penalty uses $h_{k,j} = \max(|\bar{u}_{k,j}| - \rho, 0)/(1-\rho)$, with activation fraction $\rho \in [0, 1)$.

For natural frequency $\omega_{n,j}$ and damping ratio ζ_j, define the torque reference

$$u_{k,j}^{\mathrm{d}} = \mathrm{sat}_{\mathcal{U}_j}[-J_j(\omega_{n,j}^2 e_{k,j} + 2\zeta_j \omega_{n,j} \omega_{k,j})],$$

where $(J_\phi, J_\theta, J_\psi) = (I_{xx}, I_{yy}, I_{zz})$ and $\bar{u}_{k,j}^{\mathrm{d}} = u_{k,j}^{\mathrm{d}}/u_j^{\max}$. The channel reward is

$$\begin{aligned} r_{k,j} = {} & w_p(V_{k,j} - V_{k+1,j}) - w_s V_{k+1,j} - w_u \bar{u}_{k,j}^2 - w_a \bar{a}_{k,j}^2 \\ & - w_t(\bar{u}_{k,j} - \bar{u}_{k,j}^{\mathrm{d}})^2 - w_{\mathrm{sat}} h_{k,j}^2 - w_{\mathrm{eq}} \mathbb{I}_{\Omega_{0,j}} \mathbb{I}_{a_{k,j} \neq 0} \\ & - 20 \mathbb{I}_{\{\tilde{\boldsymbol{s}}_{k+1} \notin \mathcal{D}\}}. \end{aligned} \tag{22}$$

The weights $w_p, w_s, w_u, w_a, w_t, w_{\mathrm{sat}}, w_{\mathrm{eq}}$ scale potential decrease, state error, torque magnitude, increments, torque-reference error, near-limit activation, and nonzero actions near equilibrium. The indicator $\mathbb{I}_{\Omega_{0,j}}$ is evaluated at $\tilde{\boldsymbol{s}}_k$. The sets $\Omega_{0,j}$ and $\mathcal{D}$ determine $\chi_{k,j}$, with thresholds specified below. On the compact sampled domain, the action, torque, and disturbance bounds give the bounded reward in Eq. (3).

C.2 Sampled States and Optimization Settings

The reference configuration uses $\mathcal{C} = [-120, 20]$, $\gamma = 0.92$, $\beta = 0.30$, and hence $\bar{\gamma} = 0.276$. The reward weights $(w_s, w_\sigma, w_u, w_a, w_p, w_t, w_{\mathrm{sat}}, w_{\mathrm{eq}})$ are $(0.07, 0.45, 0.015, 0.005, 10, 2, 0.03, 0.18)$, with $\rho = 0.90$. Its channel parameters are $\boldsymbol{e}_{\mathrm{sc}} = [0.35, 0.35, 0.16]$ rad, $\boldsymbol{\omega}_{\mathrm{sc}} = [1.2, 1.2, 0.55]$ rad/s, $\boldsymbol{\kappa} = [5.5, 5.5, 12]$, $\boldsymbol{\omega}_n = [24, 24, 16]$ rad/s, and $\boldsymbol{\zeta} = [1, 1, 1]$. The error, rate, and torque thresholds of $\Omega_{0,j}$ are $[0.0087, 0.0087, 0.0087]$ rad, $[0.0349, 0.0349, 0.0349]$ rad/s, and $[0.003, 0.003, 0.001]$ N·m; the error and rate limits of $\mathcal{D}$ are $[0.90, 0.75, 0.25]$ rad and $[4.0, 4.0, 1.8]$ rad/s.

Each axis uses 20 Bellman iterations, 50,000 random one-step states per iteration, and 6,000 independent validation states. The coarse, fine, and near-equilibrium strata have proportions 0.35, 0.35, and 0.30. Their three-axis error limits are $[1.05, 1.05, 0.35]$, $[0.25, 0.25, 0.10]$, and $[0.035, 0.035, 0.018]$ rad, respectively; the corresponding rate limits are $[3, 3, 1.4]$, $[0.8, 0.8, 0.35]$, and $[0.14, 0.14, 0.07]$ rad/s. Each transition uses ten 1-ms integration steps. Adam optimization uses six epochs, mini-batches of 1024, learning rate 2×10^{-4}, and squared ℓ_2 regularization with coefficient $\lambda_{\mathrm{reg}} = 2 \times 10^{-5}$. Target parameters are replaced every two Bellman iterations.

Table 5 lists each setting followed by mean $E_3 \pm$ the Student-t 95% confidence-interval half-width in degrees over three independently trained policies.

Table 6: Controller configurations for gate-crossing comparisons.

Controller	Command	Observation	Structure or principal setting
PID	$\boldsymbol{u}_k$	$\boldsymbol{e}, \dot{\boldsymbol{e}}$	$\boldsymbol{K}_p^{\mathrm{att}} = \mathrm{diag}(2.5, 2.5, 2.5)$; $\boldsymbol{K}_d^{\mathrm{att}} = \mathrm{diag}(0.42, 0.42, 0.23)$
MPC	$\boldsymbol{u}_k$	$\boldsymbol{e}, \dot{\boldsymbol{e}}$	$N_p = 16$; $\boldsymbol{q}_\Theta = [120, 120, 180]$; $\boldsymbol{q}_\omega = [2, 2, 2.5]$
PPO	$\boldsymbol{u}_k$	$\boldsymbol{e}, \boldsymbol{\omega}, \boldsymbol{u}_{k-1}$	Continuous three-action actor; width 128
TD3	$\boldsymbol{u}_k$	$\boldsymbol{e}, \boldsymbol{\omega}, \boldsymbol{u}_{k-1}$	Continuous actor with twin critics; width 128
DP-DDQN	$\boldsymbol{a}_k$	$\boldsymbol{e}, \boldsymbol{\omega}, \boldsymbol{u}_{k-1}$	Three independent 9-128-128-64-7 networks

C.3 Regulation Evaluation and Validation Metric

Ten independently trained three-axis policies are evaluated under four coupled initial conditions and five unseen actuator-noise realizations, yielding 20 runs per policy and 200 in total.

The initial attitude/rate pairs are

$$([8, -6, 5]^\circ, [0, 0, 0]^\circ/\mathrm{s}), \quad ([-10, 8, -6]^\circ, [3, -2, 2]^\circ/\mathrm{s}),$$
$$([12, 9, -7]^\circ, [-4, 3, -2]^\circ/\mathrm{s}), \quad ([-14, -10, 8]^\circ, [5, -4, 3]^\circ/\mathrm{s}).$$

Success requires all errors to remain within $\pm 2^\circ$ during the final second.

The independent validation residual at iteration b is

$$\mathcal{L}_j^{(b)} = \frac{1}{N_v |\mathcal{A}_j|} \sum_{i=1}^{N_v} \sum_{a \in \mathcal{A}_j} \left(\widehat{Q}_j^{(b)}(\tilde{\boldsymbol{s}}_i, a) - Y_{i,j}^{(b)}(a) \right)^2, \tag{23}$$

where $\widehat{Q}_j^{(b)}$ is the fitted network after iteration b, $|\mathcal{A}_j| = 7$ for the reference policy, and the $N_v = 6000$ states are independent of the fitting set. The aggregate regulation error E_3 is defined in Section 5. The responses in Fig. 2 follow one training run. Validation residuals are normalized to their initial values and summarized across all ten independent training runs using means and Student-t 95% CIs.

D Baseline Controller Configurations

Table 6 lists the attitude-loop gains, MPC prediction horizon N_p, and attitude and rate weight vectors $\boldsymbol{q}_\Theta$ and $\boldsymbol{q}_\omega$. The MPC torque and increment weight vectors are $[0.018, 0.018, 0.012]$ and $[0.20, 0.20, 0.30]$. Both actor–critic baselines use $\gamma = 0.995$ and mini-batches of 128. PPO uses 512-step rollouts, four epochs, clipping coefficient 0.2, generalized advantage estimation coefficient 0.95, entropy coefficient 0.01, and actor/critic learning rates $3 \times 10^{-4}/10^{-3}$. TD3 updates its policy after every two critic updates and uses actor/critic learning rates $10^{-5}/10^{-3}$.

The PPO policy is selected from three independently trained candidates using separate validation cases. TD3 and DP-DDQN each contribute one trained policy, and all policies remain fixed during gate evaluation. PPO and TD3 training comprises 500,000 and 500,052 transitions, respectively. Over the final 500 episodes, 99.0% and 92.4% of their trajectories reach at least 95% of the 300-step horizon; normalized reward drifts are 0.353 and 0.242.