Supplementary Material: Benchmarking Environments for MSACL

Yongwei Zhang, Yuanzhe Xing, Quanyi Liang, Quan Quan, *Senior Member, IEEE*, and Zhikun She

This supplementary material provides a detailed description of the nonlinear dynamical systems selected as benchmarks for the paper *"MSACL: Multi-Step Actor-Critic Learning with Lyapunov Certificates for Exponentially Stabilizing Control"*. The official implementation, encompassing both the MSACL algorithm and these simulation environments, is available on GitHub at https://github.com/YuanZhe-Xing/MSACL.

These canonical benchmarks, drawn from nonlinear control (NLC) [1]–[3], reinforcement learning (RL) [4], [5], and optimal control literature [6], encompass diverse dynamics from underactuated mechanics to rigid-body dynamics on the $\mathrm{SE}(3)$ manifold. We categorize these tasks by their control objectives: *Stabilization* and *Tracking*. Formal definitions and reward structures are detailed below.

I. Unified Control Objectives

Despite the distinct physical dynamics inherent to each system, the core control tasks are formulated using a consistent mathematical structure.

a) Stabilization Tasks [4], [7], [8]: The first four systems (VanderPol, Pendulum, DuctedFan, and Two-link) are configured as stabilization tasks. For these systems, the control objective is to find a policy that generates a sequence of control inputs $\{\mathbf{u}_t\}_{t=0}^{T}$ to drive the system state $\mathbf{x}_t$ from a randomly sampled initial state $\mathbf{x}_0$ toward a fixed equilibrium point $\mathbf{x}_g$ (typically the origin, $\mathbf{x}_g = \mathbf{0}$) and maintain stability at this equilibrium for the duration of the episode.

b) Tracking Tasks [3], [9], [10]: The remaining two systems (SingleCarTracking and QuadrotorTracking) are formulated as trajectory tracking tasks. Here, the objective is defined in terms of the tracking error $\mathbf{e}_t = \mathbf{x}_t - \mathbf{x}_t^{\text{ref}}$. The goal is to determine a control sequence such that the error trajectory $\{\mathbf{e}_t\}_{t=0}^{T}$ converges to the zero-error state $\mathbf{e}_g = \mathbf{0}$, thereby ensuring that the system state $\mathbf{x}_t$ accurately follows a time-varying reference trajectory $\mathbf{x}_t^{\text{ref}}$.

II. Unified Reward Function Design

To align the learning process with the aforementioned control objectives, we employ a consistent reward function structure across all six environments:

$$r_t = \underbrace{-\left(\mathbf{s}_t^\top \mathbf{Q}_r \mathbf{s}_t + \mathbf{u}_t^\top \mathbf{R}_r \mathbf{u}_t\right)}_{\text{Penalty}} + \underbrace{r_{\text{approach}}(\mathbf{s}_t)}_{\text{Encouragement}},$$

where $\mathbf{s}_t$ represents the generalized state vector, defined as $\mathbf{s}_t = \mathbf{x}_t$ for stabilization tasks and $\mathbf{s}_t = \mathbf{e}_t$ for tracking tasks. The matrices $\mathbf{Q}_r$ and $\mathbf{R}_r$ are positive semi-definite weighting matrices that penalize state deviations and control effort, respectively.

The specific parameter configurations for each environment, including the diagonal elements of $\mathbf{Q}_r$ and $\mathbf{R}_r$, are summarized in Table I. The term $\mathbb{I}(\cdot)$ denotes the indicator function, which provides a sparse encouragement reward when the system enters a predefined tolerance region around the target state. For all environments, the state cost matrix $\mathbf{Q}_c$ is simply defined as an identity matrix $\mathbf{I}_n$ of appropriate dimensions, where n denotes the dimension of state space.

III. Details of VanderPol

A. System Overview

The VanderPol oscillator, originally proposed by Balthasar van der Pol in 1920 [11], is a classic model describing self-sustaining oscillatory circuits characterized by nonlinear damping. This system serves as a fundamental benchmark for electronic circuit stabilization (e.g., vacuum tube oscillators), mechanical vibration control, and biological modeling, such as cardiac pacemaker dynamics [12]. In this study, we extend the autonomous system with a control input to evaluate stabilization performance within a nonlinear regime.

B. Dynamics and Physical Parameters

The continuous-time dynamics of the controlled VanderPol are governed by the following second-order nonlinear differential equations:

$$\begin{aligned}\dot{x}_1 &= x_2,\\ \dot{x}_2 &= \mu\left(1 - x_1^2\right) x_2 - x_1 + u,\end{aligned}$$

where x_1 and x_2 represent the oscillator displacement and velocity, respectively. The physical meanings of the variables are summarized in Table II, and the specific parameter values and simulation settings used in our experiments are detailed in Table III.

C. State and Action Spaces

For the reinforcement learning formulation, the system state vector is defined as $\mathbf{x} = [x_1, x_2]^\top$. To reflect the physical limits of the oscillator and ensure numerical stability during training, the state space is constrained to $\mathcal{X} = [-10, 10] \times [-10, 10]$, where both displacement and velocity are bounded within the range of ± 10. The control objective is to drive the system from a random initial configuration to the equilibrium state (goal state) defined as $\mathbf{x}_g = [0, 0]^\top$, corresponding to the stable origin of the phase plane.

The control action u is a one-dimensional continuous input restricted to the action space $\mathcal{U} = [-5, 5]$. These bounds are selected to provide sufficient authority for stabilization while preventing excessively large inputs that could violate physical actuation constraints or lead to divergent gradients. At the start of each episode, the initial state $\mathbf{x}_0$ is sampled from a uniform distribution $\mathbf{x}_0 \sim \text{Uniform}([-5, 5] \times [-5, 5])$. This initialization range is intentionally chosen to place the

TABLE I
REWARD FUNCTION PARAMETERS FOR ALL BENCHMARK SYSTEMS

System	**State Penalty $\mathbf{Q}_r$** (diag)	**Control Penalty $\mathbf{R}_r$** (diag)	**Encouragement** (r_{approach})
VanderPol	$[2, 1]$	$[0.1]$	$\mathbb{I}(\|\mathbf{x}_t\|_\infty \leq 10^{-2})$
Pendulum	$[2, 1]$	$[0.1]$	$\mathbb{I}(\|\mathbf{x}_t\|_\infty \leq 10^{-2})$
DuctedFan	$[2, 2, 2, 1, 1, 1]$	$[0.1, 0.1]$	$\mathbb{I}(\|\mathbf{x}_t\|_\infty \leq 10^{-2})$
Two-link	$[2, 2, 1, 1]$	$[0.1, 0.1]$	$\mathbb{I}(\|\mathbf{x}_t\|_\infty \leq 10^{-2})$
SingleCarTracking	$[2, 2, 1, 1, 1, 1, 1]$	$[0.1, 0.1]$	$\mathbb{I}(\|\mathbf{e}_t\|_\infty \leq 10^{-2})$
QuadrotorTracking	$\mathbf{I}_{12}$	$[10^{-4}, 0.01, 0.01, 0.01]$	$10(1 - \frac{\|\mathbf{e}_t\|_\infty}{0.1}) \cdot \mathbb{I}(\|\mathbf{e}_t\|_\infty \leq 0.1)$

TABLE II
VARIABLE DEFINITIONS FOR CONTROLLED VANDERPOL

Variable	Physical Meaning
x_1	Displacement of the oscillator (m)
x_2	Velocity of the oscillator (m/s)
$\dot{x}_2$	Acceleration of the oscillator (m/s^2)
μ	Nonlinear damping strength parameter (dimensionless)
u	Control input force applied to the oscillator (N)

TABLE III
PARAMETER VALUES FOR CONTROLLED VANDERPOL

Parameter	Value
μ (nonlinear strength)	1.0 (dimensionless)
Simulation time step (dt)	0.01 s
Control update interval	5 simulation steps
Maximum episode steps	1000

system within a significant nonlinear regime, ensuring that the controller must actively counteract the self-sustaining oscillations to achieve stabilization.

IV. DETAILS OF PENDULUM

A. System Overview

The Pendulum is a classic rigid-body system with a pivot point constrained to a vertical plane, where the primary control objective is to stabilize the system in an inverted (upright) position against gravity. Beyond its role as a fundamental pedagogical tool for illustrating nonlinear stability and feedback control [13], its underlying principles are essential for robotic locomotion in bipedal systems [14], aerospace attitude control, and industrial crane load stabilization [7].

B. Dynamics and Physical Parameters

The continuous-time dynamics of the Pendulum are derived from Newtonian mechanics and are defined by the following equations of motion:

$$\dot{\theta} = \dot{\theta},$$
$$\ddot{\theta} = \frac{mgL\sin(\theta) - b\dot{\theta} + u}{mL^2},$$

where θ and $\dot{\theta}$ denote the angular displacement and angular velocity, respectively. The physical variables and their definitions are summarized in Table IV, while the specific parameter values and simulation settings used in this study are provided in Table V.

TABLE IV
VARIABLE DEFINITIONS FOR PENDULUM

Variable	Physical Meaning
θ	Angular displacement of the pendulum (rad)
$\dot{\theta}$	Angular velocity of the pendulum (rad/s)
$\ddot{\theta}$	Angular acceleration of the pendulum (rad/s^2)
m	Mass of the pendulum bob (kg)
g	Gravitational acceleration (m/s^2)
L	Length of the pendulum rod (m)
b	Damping coefficient of the pendulum joint (N·m·s/rad)
u	Control input torque applied to the pivot (N·m)

TABLE V
PHYSICAL PARAMETER VALUES FOR PENDULUM

Parameter	Value
g	9.81 m/s^2
L	0.5 m
m	0.15 kg
b	0.1 N·m·s/rad
Simulation time step (dt)	0.01 s
Control update interval	5 simulation steps
Maximum episode steps	1000

C. State and Action Spaces

The system state vector is defined as $\mathbf{x} = [\theta, \dot{\theta}]^\top$, with the state space constrained to $\mathcal{X} = [-\pi, \pi] \times [-10, 10]$. This range covers the full angular displacement and accounts for physical angular velocity limits. The control objective is to drive the system to the unstable equilibrium point (upright position) defined as $\mathbf{x}_g = [0, 0]^\top$.

The control action u is a one-dimensional continuous torque restricted to the action space $\mathcal{U} = [-5, 5]$. These torque bounds are selected to ensure the system has sufficient authority for the swing-up maneuver while remaining within realistic actuation limits. At the start of each episode, the initial state $\mathbf{x}_0$ is uniformly sampled from the entire state space, $\mathbf{x}_0 \sim \text{Uniform}([-\pi, \pi] \times [-10, 10])$, requiring the controller to handle both stabilization from near-upright positions and complex swing-up maneuvers from high-energy initial states.

D. Comparison with OpenAI Gym Pendulum

Our Pendulum implementation is significantly more rigorous than the standard OpenAI Gym version [15]. Specifically, we expand the initial state space to the full range $[-\pi, \pi] \times [-10, 10]$ and increase the actuation limit to $[-5, 5]$, whereas the Gym environment utilizes a narrower velocity

initialization ($[-1, 1]$) and limited torque range ($[-2, 2]$). Furthermore, we impose strict boundary-based termination and extend the episode horizon to $T = 1000$ steps. These more stringent constraints are intentionally designed to expose the limitations of standard RL baselines; for instance, while PPO converges readily in the Gym setting, it often fails to achieve robust stabilization in our high-energy regime.

V. Details of DuctedFan

A. System Overview

The DuctedFan system models the coupled translational and rotational dynamics of a ducted-fan aircraft under the influence of gravitational and control forces [16]. As a canonical benchmark for Vertical Takeoff and Landing (VTOL) research, it provides a rigorous platform for validating hover stabilization and attitude control algorithms [10], [17]. The system is underactuated and exhibits strong nonlinear coupling between its lateral displacement and angular orientation.

B. Dynamics and Physical Parameters

The continuous-time dynamics of the DuctedFan are derived from Newtonian mechanics and rigid-body dynamics, defined by the following state-space equations:

$$\begin{aligned}
\dot{x} &= \dot{x}, \\
\dot{y} &= \dot{y}, \\
\dot{\theta} &= \dot{\theta}, \\
\ddot{x} &= \frac{-mg\sin(\theta) - d\dot{x} + u_1\cos(\theta) - u_2\sin(\theta)}{m}, \\
\ddot{y} &= \frac{mg(\cos(\theta) - 1) - d\dot{y} + u_1\sin(\theta) + u_2\cos(\theta)}{m}, \\
\ddot{\theta} &= \frac{ru_1}{J},
\end{aligned}$$

where $u_1 = f_1$ and $u_2 = f_2 - mg$, with f_1 and f_2 representing the control forces applied perpendicular and parallel to the fan axis, respectively. The variable definitions and physical parameters used for the simulation are summarized in Table VI and Table VII.

TABLE VI
Variable Definitions for DuctedFan

Variable	Physical Meaning
x, y	Horizontal and vertical displacement (m)
θ	Angular orientation (rad)
$\dot{x}, \dot{y}$	Horizontal and vertical velocity (m/s)
$\dot{\theta}$	Angular velocity (rad/s)
m	System mass (kg)
g	Gravitational acceleration (m/s^2)
r	Moment arm of the control force (m)
d	Translational damping coefficient (N·s/m)
J	Moment of inertia (kg·m^2)
u_1, u_2	Control force inputs (N)

C. State and Action Spaces

The system state vector is defined as $\mathbf{x} = [x, y, \theta, \dot{x}, \dot{y}, \dot{\theta}]^\top$. The admissible state space is $\mathcal{X} = [-5, 5]^2 \times [-\pi/2, \pi/2] \times [-5, 5]^3$, which covers the translational workspace, the pitch angle range, and the respective velocity limits. The control objective is to stabilize the vehicle at the equilibrium state $\mathbf{x}_g = [0, 0, 0, 0, 0, 0]^\top$, corresponding to a steady hover at the origin.

TABLE VII
Physical Parameter Values for DuctedFan

Parameter	Value
g	9.81 m/s^2
m	8.5 kg
r	0.26 m
d	0.95 N·s/m
J	0.048 kg·m^2
Simulation time step (dt)	0.01 s
Control update interval	5 simulation steps
Maximum episode steps	1000

The control vector $\mathbf{u} = [u_1, u_2]^\top$ consists of two continuous force inputs, with the action space defined as $\mathcal{U} = [-5, 5] \times [-5, 5]$. These bounds reflect physical actuation limits while ensuring sufficient control authority to counteract gravity and damping. At the start of each episode, the initial state $\mathbf{x}_0$ is sampled from a uniform distribution $\mathbf{x}_0 \sim \text{Uniform}([-0.5, 0.5]^6)$. This initialization provides a localized yet challenging set of starting conditions that require the controller to coordinate multi-axis forces to achieve precise stabilization.

VI. Details of Two-link

A. System Overview

The Two-link system captures the rigid-body rotational dynamics of a serial two-link planar structure subject to gravitational effects and joint torque inputs. This system serves as a representative benchmark for underactuated or fully actuated mechanical systems, where the nonlinear coupling between links presents a significant control challenge [8]. Key applications of this model include robotic manipulator trajectory tracking, humanoid robot joint coordination, and the study of multi-degree-of-freedom Lagrangian systems [18].

B. Dynamics and Physical Parameters

The continuous-time dynamics of the Two-link Planar Robot is derived from Lagrangian mechanics and are expressed by the standard Euler-Lagrange equation:

$$\mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{G}(\mathbf{q}) = \mathbf{u},$$

where $\mathbf{q} = [\theta_1, \theta_2]^\top$ denotes the vector of joint angles, $\mathbf{M}(\mathbf{q})$ is the 2×2 symmetric positive-definite mass matrix, $\mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})$ is the Coriolis and centrifugal matrix, $\mathbf{G}(\mathbf{q})$ is the gravity vector, and $\mathbf{u} = [u_1, u_2]^\top$ is the control vector of joint torques. The explicit forms of these matrices are given by:

$$\mathbf{M}(\mathbf{q}) = \begin{bmatrix} M_{11} & M_{12} \\ M_{21} & M_{22} \end{bmatrix},$$

$$\mathbf{C}(\mathbf{q}, \dot{\mathbf{q}}) = \begin{bmatrix} h\dot{\theta}_2 & h(\dot{\theta}_1 + \dot{\theta}_2) \\ -h\dot{\theta}_1 & 0 \end{bmatrix},$$

$$\mathbf{G}(\mathbf{q}) = \begin{bmatrix} -(m_1 l_{c1} + m_2 l_1)g\sin\theta_1 - m_2 l_{c2} g\sin(\theta_1 + \theta_2) \\ -m_2 l_{c2} g\sin(\theta_1 + \theta_2) \end{bmatrix},$$

where $h = -m_2 l_1 l_{c2} \sin\theta_2$. The individual elements of the mass matrix are defined as:

$$M_{11} = I_1 + I_2 + m_1 l_{c1}^2 + m_2(l_1^2 + l_{c2}^2 + 2l_1 l_{c2}\cos\theta_2),$$
$$M_{12} = M_{21} = I_2 + m_2(l_{c2}^2 + l_1 l_{c2}\cos\theta_2),$$
$$M_{22} = I_2 + m_2 l_{c2}^2.$$

The variable definitions and the specific physical parameters used in our simulation are summarized in Table VIII and Table IX, respectively.

TABLE VIII
VARIABLE DEFINITIONS FOR TWO-LINK PLANAR ROBOT

Variable	Physical Meaning
θ_1, θ_2	Angular orientation of link 1 and link 2 (rad)
$\dot{\theta}_1, \dot{\theta}_2$	Angular velocity of link 1 and link 2 (rad/s)
$\ddot{\theta}_1, \ddot{\theta}_2$	Angular acceleration of link 1 and link 2 (rad/s^2)
l_1, l_2	Length of link 1 and link 2 (m)
m_1, m_2	Mass of link 1 and link 2 (kg)
l_{c1}, l_{c2}	Distance from joint to center of mass (COM) (m)
I_1, I_2	Moment of inertia about the COM (kg·m^2)
u_1, u_2	Torque input at joint 1 and joint 2 (N·m)

TABLE IX
PHYSICAL PARAMETER VALUES FOR TWO-LINK PLANAR ROBOT

Parameter	Value
g	9.81 m/s^2
l_1, l_2	1.0 m, 1.0 m
m_1, m_2	1.0 kg, 1.0 kg
l_{c1}, l_{c2}	0.5 m, 0.5 m
I_1, I_2	0.0833 kg·m^2, 0.0833 kg·m^2
Simulation time step (dt)	0.01 s
Control update interval	5 simulation steps
Maximum episode steps	1000

C. State and Action Spaces

The system state vector is defined as $\mathbf{x} = [\theta_1, \theta_2, \dot{\theta}_1, \dot{\theta}_2]^\top$. To incorporate physical constraints, the state space is bounded by $\mathcal{X} = [-\pi/2, \pi/2]^2 \times [-20, 20]^2$, where the joint angles are limited to a $\pm\pi/2$ range and the angular velocities are capped at ± 20 rad/s. The control objective is to achieve stabilization at the equilibrium state $\mathbf{x}_g = [0, 0, 0, 0]^\top$, representing the system at rest in the downward vertical position.

The control action vector $\mathbf{u} = [u_1, u_2]^\top$ consists of continuous joint torques within the action space $\mathcal{U} = [-20, 20] \times [-20, 20]$. These torque limits are established to prevent excessively high control efforts while providing sufficient authority to manage the gravitational and inertial coupling effects. For each training episode, the initial state $\mathbf{x}_0$ is sampled from a uniform distribution $\mathbf{x}_0 \sim \text{Uniform}([-0.5, 0.5]^4)$, ensuring that the controller begins each run in a neighborhood of the target equilibrium while requiring active correction to maintain stability.

VII. DETAILS OF SINGLECARTRACKING

A. System Overview

The SingleCarTracking environment is designed for high-fidelity ground vehicle trajectory tracking. It utilizes a single-track (bicycle) model that incorporates both lateral and longitudinal dynamics of a rear-wheel-drive vehicle, accounting for nonlinear tire-road friction [19] and steering constraints. This benchmark is crucial for validating control strategies in autonomous driving [9], [20], such as path following and high-precision maneuvers in mobile robotics [21].

B. Reference Trajectory and Tracking Error Formulation

In this study, the reference trajectory focuses on the spatial path in the 2D plane. Let $\boldsymbol{p}_t^{\text{ref}} = [x_t^{\text{ref}}, y_t^{\text{ref}}]^\top \in \mathbb{R}^2$ denote the target position on the horizontal plane. While the full reference state $\mathbf{x}_t^{\text{ref}}$ includes heading and velocity, the core objective is defined by the straight-line trajectory $\boldsymbol{p}_t^{\text{ref}} = [t, 0]^\top$ for $t \in [0, 50]$. The reference parameters are set to a longitudinal velocity $v_{\text{ref}} = 1.0$ m/s, zero acceleration $a_{\text{ref}} = 0.0$ m/s^2, and a zero heading angle $\psi_{\text{ref}} = 0.0$ with zero angular velocity $\omega_{\text{ref}} = 0.0$ rad/s. A friction scaling factor $\mu_{\text{scale}} = 0.1$ is applied to adjust tire nonlinearity.

The physical state of the vehicle is described by the vector $\mathbf{x} = [x, y, \delta, v, \psi, \dot{\psi}, \beta]^\top$, representing position, steering angle, longitudinal speed, heading angle, yaw rate, and side-slip angle. For control purposes, we define the tracking error vector as $\mathbf{e} = [s_{xe}, s_{ye}, \delta_e, v_e, \psi_e, \dot{\psi}_e, \beta_e]^\top$, where $s_{xe} = x - x_{\text{ref}}$ and $s_{ye} = y - y_{\text{ref}}$ are the longitudinal and lateral position errors (also denoted as e_x and e_y respectively in the main text), and $\delta_e = \delta$, $v_e = v - v_{\text{ref}}$, $\psi_e = \psi - \psi_{\text{ref}}$, $\dot{\psi}_e = \dot{\psi} - \omega_{\text{ref}}$, $\beta_e = \beta$. The admissible error space $\mathcal{E}$ is constrained to $[-1, 1]^2 \times [-1.066, 1.066] \times [-1, 1] \times [-\pi/2, \pi/2]^2 \times [-\pi/3, \pi/3]$, reflecting realistic tracking tolerances and physical actuation limits. The control objective is to maintain the system at the equilibrium error state $\mathbf{e}_g = \mathbf{0}$. At the start of each episode, the error is initialized via $\mathbf{e}_0 \sim \text{Uniform}([-0.5, 0.5]^7)$.

C. Dynamics and Physical Parameters

The system dynamics are modeled as an affine nonlinear system $\dot{\mathbf{e}} = \mathbf{f}(\mathbf{e}) + \mathbf{g}(\mathbf{e})\mathbf{u}$, where the control input $\mathbf{u} = [\dot{\delta}, a_{\text{long}}]^\top$ consists of the steering angular velocity and longitudinal acceleration, bounded by $\mathcal{U} = [-5, 5]^2$. To ensure numerical stability and physical accuracy across different speed regimes, the model transitions between a dynamic tire model and a kinematic bicycle model.

a) Dynamic Model ($|v| \geq 0.1$m/s): In the high-speed regime, the drift vector field is defined as $\mathbf{f}(\mathbf{e}) = [v\cos(\psi_e + \beta_e) - v_{\text{ref}} + \omega_{\text{ref}} s_{ye}, v\sin(\psi_e + \beta_e) - \omega_{\text{ref}} s_{xe}, 0, -a_{\text{ref}}, \dot{\psi}_e, f_6(\mathbf{e}), f_7(\mathbf{e})]^\top$. The nonlinear yaw and side-slip terms are:

$$f_6(\mathbf{e}) = \frac{\mu m}{I_z(l_f + l_r)}\Bigg[-\frac{l_f^2 C_{Sf} g l_r + l_r^2 C_{Sr} g l_f}{v}\dot{\psi} + (l_r C_{Sr} g l_f - l_f C_{Sf} g l_r)\beta_e + (l_f C_{Sf} g l_r)\delta_e \Bigg],$$

$$f_7(\mathbf{e}) = \left[\frac{\mu(C_{Sr} g l_f l_r - C_{Sf} g l_r l_f)}{v^2(l_f + l_r)} - 1\right]\dot{\psi} - \frac{\mu(C_{Sr} g l_f + C_{Sf} g l_r)}{v(l_f + l_r)}\beta_e + \frac{\mu C_{Sf} g l_r}{v(l_f + l_r)}\delta_e.$$

where C_{Sf} and C_{Sr} are the cornering stiffness parameters for the front and rear wheels

$$C_{Sf} = C_{Sr} = -\frac{K_{\text{tire}}}{D_{\text{tire}}},$$

and

$$\mu = \mu_{\text{scale}} \cdot D_{\text{tire}}$$

The control gain matrix is $\mathbf{g}(\mathbf{e}) = [\mathbf{0}_{2\times2}, [1,0], [0,1], \mathbf{0}_{1\times2}, [0, \mathbf{g}_{6,2}(\mathbf{e})], [0, \mathbf{g}_{7,2}(\mathbf{e})]]^\top$. The terms $g_{6,2}$ and $g_{7,2}$ account for CG height h_s and vertical load shifts:

$$\begin{aligned} g_{6,2}(\mathbf{e}) =& \frac{\mu m}{I_z(l_f+l_r)}\Bigg[-\frac{-l_f^2 C_{Sf}h_s + l_r^2 C_{Sr}h_s}{v}\dot{\psi} \\ &+ (l_r C_{Sr}h_s + l_f C_{Sf}h_s)\beta_e - (l_f C_{Sf}h_s)\delta_e\Bigg], \end{aligned}$$

$$\begin{aligned} g_{7,2}(\mathbf{e}) =& \frac{\mu(C_{Sr}h_s l_r + C_{Sf}h_s l_f)}{v^2(l_f+l_r)}\dot{\psi} \\ &- \frac{\mu(C_{Sr}h_s - C_{Sf}h_s)}{v(l_f+l_r)}\beta_e - \frac{\mu C_{Sf}h_s}{v(l_f+l_r)}\delta_e, \end{aligned}$$

b) Kinematic Model ($|v| < 0.1\text{m/s}$): To avoid singularities at low speeds, the model simplifies to a kinematic representation where side-slip is negligible. Let $L = l_f + l_r$ denote the wheelbase. The drift field $\mathbf{f}(\mathbf{e})$ and input matrix $\mathbf{g}(\mathbf{e})$ are expressed as:

$$\begin{aligned} \mathbf{f}(\mathbf{e}) =& \Big[v\cos(\psi_e+\beta_e) - v_{\text{ref}} + \omega_{\text{ref}} s_{ye}, \\ & v\sin(\psi_e+\beta_e) - \omega_{\text{ref}} s_{xe}, \\ & 0, \quad -a_{\text{ref}}, \quad \frac{v\cos(\beta_e)}{L}\tan(\delta_e) - \omega_{\text{ref}}, \quad 0, \quad 0\Big]^\top, \\ \mathbf{g}(\mathbf{e}) =& \Big[\mathbf{0}_{2\times2}, \quad [1,0], \quad [0,1], \quad \mathbf{0}_{1\times2}, \\ & [\frac{\cos(\beta_e)\tan(\delta_e)}{L}, G_{6,2}] \quad [\beta_e', 0]\Big]^\top, \end{aligned}$$

where the side-slip rate auxiliary term is

$$G_{6,2} = \frac{1}{L}\left(-v\sin(\beta)\tan(\delta)\beta' + \frac{v\cos(\beta)}{\cos^2(\delta)}\right)$$

$$\beta_e' = \frac{l_r/(L\cos^2(\delta_e))}{1+\left(\tan(\delta_e)\frac{l_r}{L}\right)^2}.$$

The system parameters are detailed in Tables X and XI.

TABLE X
VARIABLE DEFINITIONS FOR SINGLECARTRACKING

Category	Variable	Physical Meaning
True State	$\mathbf{x} \in \mathbb{R}^7$	$\mathbf{x} = [x, y, \delta, v, \psi, \dot{\psi}, \beta]^\top$: Position (x, y), steering angle, speed, heading, yaw rate, side-slip.
Tracking Error	$\mathbf{e} \in \mathbb{R}^7$	$\mathbf{e} = [s_{xe}, s_{ye}, \delta_e, v_e, \psi_e, \dot{\psi}_e, \beta_e]^\top$: Tracking errors relative to reference trajectory.
Control	$\mathbf{u} \in \mathbb{R}^2$	$\mathbf{u} = [\dot{\delta}, a_{\text{long}}]^\top$: Steering rate (rad/s) and longitudinal acceleration (m/s^2).
Parameters	m, I_z, l_f, l_r, h_s	Mass, yaw inertia, distance to front/rear axles, and CG height.
	$\mu, K_{\text{tire}}, D_{\text{tire}}$	Tire-road friction coefficient, tire stiffness parameters and tire friction coefficient.

TABLE XI
PHYSICAL PARAMETER VALUES FOR SINGLECARTRACKING

Parameter	Value	Parameter	Value
m	1093 kg	I_z	1792 kg·m^2
l_f, l_r	1.155 m, 1.423 m	h_s	0.614 m
$\delta_{\max}$	±1.066 rad	$a_{\text{long, max}}$	±5 m/s^2
D_{tire}	1.0489	K_{tire}	-21.92
dt	0.01 s	Horizon T	1000
μ_{scale}	0.1		

VIII. DETAILS OF QUADROTORTRACKING

A. System Overview

The QuadrotorTracking environment evaluates the 6-degree-of-freedom (6-DoF) rigid-body dynamics of a quadrotor UAV [22]. This benchmark incorporates complex rotor thrust-torque coupling and attitude kinematics defined on the special Euclidean group SE(3) [3]. It serves as a standard testbed for geometric control and reinforcement learning algorithms in high-agility scenarios [23], such as autonomous aerial inspection and precise path following [24], [25].

B. Reference Trajectory and Tracking Error Formulation

The control objective is to track a time-varying reference trajectory that specifies position, velocity, and heading. Let $\boldsymbol{p}_t^{\text{ref}} = [x_t^{\text{ref}}, y_t^{\text{ref}}, z_t^{\text{ref}}]^\top$ denote the desired 3D position at time step t. The benchmark utilizes a dynamic helical trajectory defined by:

$$\begin{aligned} \boldsymbol{p}_t^{\text{ref}} &= [0.4t, 0.4\sin(t), 0.6\cos(t)]^\top, \\ \boldsymbol{v}_t^{\text{ref}} &= [0.4, 0.4\cos(t), -0.6\sin(t)]^\top, \\ \boldsymbol{a}_t^{\text{ref}} &= [0, -0.4\sin(t), -0.6\cos(t)]^\top, \end{aligned}$$

where $\boldsymbol{v}_t^{\text{ref}}$ is the reference velocity. The desired heading is determined by a rotating unit vector $\boldsymbol{b}_{1,t}^{\text{ref}} = [\cos(t), \sin(t), 0]^\top$.

The system state is $\mathbf{x} = [\mathbf{x}_p^\top, \mathbf{v}^\top, \mathbf{R}, \boldsymbol{\Omega}^\top]^\top$, where $\mathbf{x}_p \in \mathbb{R}^3$ is the position, $\mathbf{v} \in \mathbb{R}^3$ is the translational velocity, $\mathbf{R} \in \text{SO}(3)$ is the rotation matrix, and $\boldsymbol{\Omega} \in \mathbb{R}^3$ is the body-frame angular velocity. Although $\mathbf{R}$ is a 3×3 matrix, it fundamentally represents three degrees of freedom (3-DoF). Thus for the sake of notational consistency, with a slight abuse of notation, we treat $\mathbf{R}$ as a three-dimensional vector, thereby concatenating it with $\mathbf{x}_p^\top$, $\mathbf{v}^\top$, and $\boldsymbol{\Omega}^\top$ to constitute the system state $\mathbf{x}$.

The tracking error $\mathbf{e} \in \mathbb{R}^{12}$ is defined by four components: position error $\mathbf{e}_p = \mathbf{x}_p - \boldsymbol{p}_t^{\text{ref}}$, velocity error $\mathbf{e}_v = \mathbf{v} - \boldsymbol{v}_t^{\text{ref}}$, attitude error $\mathbf{e}_R$, and angular velocity error $\mathbf{e}_\Omega$. Following [3], the attitude error is computed via the skew-symmetric matrix $\mathbf{S} = \mathbf{R}_t^{\text{ref}\top}\mathbf{R} - \mathbf{R}^\top\mathbf{R}_t^{\text{ref}}$, where $\mathbf{e}_R = 0.5[\mathbf{S}_{2,1}, \mathbf{S}_{0,2}, \mathbf{S}_{1,0}]^\top$, and $\mathbf{S}_{i,j}$ denotes the (i,j)-th element of the $\mathbf{S}$ matrix. The angular velocity error is $\mathbf{e}_\Omega = \boldsymbol{\Omega} - \mathbf{R}^\top\mathbf{R}_t^{\text{ref}}\boldsymbol{\Omega}_t^{\text{ref}}$. The composite error vector $\mathbf{e} = [\mathbf{e}_p^\top, \mathbf{e}_v^\top, \mathbf{e}_R^\top, \mathbf{e}_\Omega^\top]^\top$ is used as the input for the controller. At the start of each episode, the initial state $\mathbf{x}_0$ is uniformly randomly sampled to calculate $\mathbf{e}_0$, specifically

$$\begin{cases} \mathbf{x}_p^\top \sim \text{Uniform}\left([-0.01, 0.01]^3\right), \\ \mathbf{v}^\top \sim \text{Uniform}\left([-0.01, 0.01]^3\right), \\ \text{vec}(\mathbf{R} - \mathbf{I}_3) \sim \text{Uniform}\left([-0.01, 0.01]^9\right), \mathbf{R} \text{ is orthogonal}, \\ \boldsymbol{\Omega}^\top \sim \text{Uniform}\left([-0.01, 0.01]^3\right). \end{cases}$$

C. Dynamics and Physical Parameters

The continuous-time dynamics follow the Newton-Euler equations for rigid bodies:

$$\begin{cases} \dot{\mathbf{x}}_p = \mathbf{v}, & \dot{\mathbf{R}} = \mathbf{R}[\boldsymbol{\Omega}]_\times, \\ \dot{\mathbf{v}} = -\mathbf{g} + \frac{F}{m}\mathbf{b}_3, & \dot{\boldsymbol{\Omega}} = \mathbf{J}^{-1}(\mathbf{M} - \boldsymbol{\Omega} \times \mathbf{J}\boldsymbol{\Omega}), \end{cases}$$

where F is the total thrust and $\mathbf{M} = [M_x, M_y, M_z]^\top$ are the control moments. $\mathbf{b}_1$,$\mathbf{b}_2$ and $\mathbf{b}_3$ denote the desired x-axis, y-axis, and z-axis directions, respectively. $[\boldsymbol{\Omega}]_\times$denotes the skew-symmetric matrix corresponding to the angular velocity vector $\boldsymbol{\Omega} = [\Omega_x, \Omega_y, \Omega_z]$:

$$[\boldsymbol{\Omega}]_\times = \begin{bmatrix} 0 & -\Omega_z & \Omega_y \\ \Omega_z & 0 & -\Omega_x \\ -\Omega_y & \Omega_x & 0 \end{bmatrix}.$$

The control action $\mathbf{u} = [F, M_x, M_y, M_z]^\top$ is derived from the individual rotor thrusts f_i via the configuration matrix:

$$\begin{bmatrix} F \\ M_x \\ M_y \\ M_z \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 0 & -d & 0 & d \\ d & 0 & -d & 0 \\ -c_{tf} & c_{tf} & -c_{tf} & c_{tf} \end{bmatrix} \begin{bmatrix} f_1 \\ f_2 \\ f_3 \\ f_4 \end{bmatrix}$$

where d is the moment arm and c_{tf} is the thrust-torque coefficient.

Detailed parameter values are provided in Tables XII and XIII.

TABLE XII
VARIABLE DEFINITIONS FOR QUADROTORTRACKING

Category	Variable	Physical Meaning
True State	$\mathbf{x} = [\mathbf{x}_p^\top, \mathbf{v}^\top, \mathbf{R}, \boldsymbol{\Omega}^\top]^\top$	3D position, velocity, rotation matrix elements, and angular velocity.
Tracking Error	$\mathbf{e} = [\mathbf{e}_p^\top, \mathbf{e}_v^\top, \mathbf{e}_R^\top, \mathbf{e}_\Omega^\top]^\top$	Geometric tracking errors in position, velocity, attitude, and angular rates.
Control	$\mathbf{u} = [F, M_x, M_y, M_z]^\top$	Total thrust (N) and three-axis body moments (N·m).
Parameters	$m, \mathbf{J}, d, c_{tf}$	Mass, inertia matrix, rotor distance, and torque coefficient.

TABLE XIII
PHYSICAL PARAMETER VALUES FOR QUADROTORTRACKING

Parameter	Value	Parameter	Value
m	4.34 kg	$\mathbf{g}$	$[0, 0, 9.8]^\top$
$\mathbf{J}$	$\mathrm{diag}(0.08, 0.09, 0.14)$	d	0.315 m
$F_{\max}$	85.06 N	$M_{\max}$	± 10 N·m
dt	0.01 s	T	1000

REFERENCES

[1] H. K. Khalil, *Nonlinear Systems*, 3rd ed. Upper Saddle River, NJ, USA: Prentice Hall, 2002.

[2] M. W. Spong, "The swing up control problem for the Acrobot," *IEEE Control Syst. Mag.*, vol. 15, no. 1, pp. 49–55, Feb. 1995.

[3] T. Lee, M. Leok, and N. H. McClamroch, "Geometric tracking control of a quadrotor UAV on SE(3)," in *Proc. 49th IEEE Conf. Decis. Control (CDC)*, Atlanta, GA, USA, Dec. 2010, pp. 5420–5425.

[4] A. G. Barto, R. S. Sutton, and C. W. Anderson, "Neuronlike adaptive elements that can solve difficult learning control problems," *IEEE Trans. Syst., Man, Cybern.*, vol. SMC-13, no. 5, pp. 834–846, Sep./Oct. 1983.

[5] M. P. Deisenroth and C. E. Rasmussen, "PILCO: A model-based and data-efficient approach to policy search," in *Proc. 28th Int. Conf. Mach. Learn. (ICML)*, Bellevue, WA, USA, 2011, pp. 465–472.

[6] P. Falcone, F. Borrelli, J. Asgari, H. E. Tseng, and D. Hrovat, "Predictive active steering control for autonomous vehicle systems," *IEEE Trans. Control Syst. Technol.*, vol. 15, no. 3, pp. 566–580, May 2007.

[7] E. M. Abdel-Rahman, A. H. Nayfeh, and Z. N. Masoud, "Dynamics and control of cranes: A review," *J. Vib. Control*, vol. 9, no. 7, pp. 863–908, 2003.

[8] M. W. Spong, S. Hutchinson, and M. Vidyasagar, *Robot Modeling and Control*, 2nd ed. Hoboken, NJ, USA: Wiley, 2020.

[9] J. Kong, M. Pfeiffer, G. Schildbach, and F. Borrelli, "Kinematic and dynamic vehicle models for autonomous driving control design," in *Proc. IEEE Intell. Veh. Symp. (IV)*, Seoul, South Korea, 2015, pp. 1094–1099.

[10] L. Marconi and R. Naldi, "Robust global trajectory tracking for a ducted-fan aerial vehicle," *Automatica*, vol. 43, no. 11, pp. 1909–1920, 2007.

[11] B. van der Pol, "A theory of the amplitude of free and forced triode vibrations," *Radio Rev.*, vol. 1, no. 1, pp. 701–710, 1920.

[12] B. van der Pol and J. van der Mark, "The heartbeat considered as a relaxation oscillation," *Philos. Mag.*, vol. 6, no. 38, pp. 763–775, 1928.

[13] K. Ogata, *Modern Control Engineering*, 5th ed. Upper Saddle River, NJ, USA: Prentice Hall, 2010.

[14] S. Kajita, F. Kanehiro, K. Kaneko, K. Yokoi, and H. Hirukawa, "The 3D linear inverted pendulum mode: A simple modeling for a biped walking pattern generation," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Maui, HI, USA, 2001, pp. 239–246.

[15] G. Brockman, V. Cheung, L. Pettersson, J. Schneider, J. Schulman, J. Tang, and W. Zaremba, "OpenAI Gym," *arXiv preprint arXiv:1606.01540*, 2016.

[16] M. B. Milam, R. Franz, J. E. Hauser, and R. M. Murray, "Receding horizon control of a ducted fan: Experimental results," *IEEE Trans. Control Syst. Technol.*, vol. 13, no. 1, pp. 49–57, Jan. 2005.

[17] J.-M. Pflimlin, P. Binetti, P. Soueres, T. Hamel, and D. Trouchet, "Modeling and attitude control analysis of a ducted-fan micro aerial vehicle," *Control Eng. Pract.*, vol. 15, no. 10, pp. 1231–1245, 2007.

[18] J.-J. E. Slotine and W. Li, "On the adaptive control of robot manipulators," *Int. J. Robot. Res.*, vol. 6, no. 3, pp. 49–59, 1987.

[19] R. Rajamani, *Vehicle Dynamics and Control*, 2nd ed. New York, NY, USA: Springer, 2011.

[20] B. R. Kiran, I. Sobh, V. Talpaert, P. Mannion, A. A. Al Sallab, S. Yogamani, and P. P'erez, "Deep reinforcement learning for autonomous driving: A survey," *IEEE Trans. Intell. Transp. Syst.*, vol. 23, no. 6, pp. 4909–4926, Jun. 2022.

[21] C. Dawson, S. Gao, and C. Fan, "Safe control with learned certificates: A survey of neural Lyapunov, barrier, and contraction methods for robotics and control," *IEEE Trans. Robot.*, vol. 39, no. 3, pp. 1749–1767, Jun. 2023.

[22] R. Mahony, V. Kumar, and P. Corke, "Multirotor aerial vehicles: Modeling, estimation, and control of quadrotor," *IEEE Robot. Autom. Mag.*, vol. 19, no. 3, pp. 20–32, Sep. 2012.

[23] J. Hwangbo, I. Sa, R. Siegwart, and M. Hutter, "Control of a quadrotor with reinforcement learning," *IEEE Robot. Autom. Lett.*, vol. 2, no. 4, pp. 2096–2103, Oct. 2017.

[24] M. Kamel, T. Stastny, K. Alexis, and R. Siegwart, "Model predictive control for trajectory tracking of unmanned aerial vehicles using robot operating system," in *Robot Operating System (ROS): The Complete Reference (Volume 2)*, A. Koubaa, Ed. Cham, Switzerland: Springer, 2017, pp. 3–39.

[25] D. Mellinger and V. Kumar, "Minimum snap trajectory generation and control for quadrotors," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, Shanghai, China, May 2011, pp. 2520–2525.