Leveraging Symmetry to Accelerate Learning of Trajectory Tracking Controllers for Free-Flying Robotic Systems

Jake Welde*, Nishanth Rao*, Pratik Kunapuli*, Dinesh Jayaraman, and Vijay Kumar

***Abstract*— Tracking controllers enable robotic systems to accurately follow planned reference trajectories. In particular, reinforcement learning (RL) has shown promise in the synthesis of controllers for systems with complex dynamics and modest online compute budgets. However, the poor sample efficiency of RL and the challenges of reward design make training slow and sometimes unstable, especially for high-dimensional systems. In this work, we leverage the inherent Lie group symmetries of robotic systems with a floating base to mitigate these challenges when learning tracking controllers. We model a general tracking problem as a Markov decision process (MDP) that captures the evolution of both the physical and reference states. Next, we show that symmetry in the underlying dynamics and running costs leads to an MDP homomorphism, a mapping that allows a policy trained on a lower-dimensional "quotient" MDP to be lifted to an optimal tracking controller for the original system. We compare this symmetry-informed approach to an unstructured baseline, using Proximal Policy Optimization (PPO) to learn tracking controllers for three systems: the** `Particle` **(a forced point mass), the** `Astrobee` **(a fully-actuated space robot), and the** `Quadrotor` **(an underactuated system). Results show that a symmetry-aware approach both accelerates training and reduces tracking error after the same number of training steps.**

I. Introduction

To achieve real-time operation, most robotic systems utilize a "tracking controller" to stabilize a pre-planned reference trajectory. However, tracking controllers designed analytically often assume properties not enjoyed by all robotic systems (*e.g.*, "full actuation" [1]–[3] or "differential flatness" [4]), while optimization-based methods frequently rely on linearization or simplified models to meet compute constraints [5]. In contrast, controllers trained via reinforcement learning (RL) have relaxed structural assumptions while enabling real-time operation with moderate resources [6]. In [7], the authors train a single hovering policy for deployment across a range of quadrotors, generalizing satisfactorily to moving references. Meanwhile, massively parallel training of quadrupedal walking policies from high-dimensional observations enabled startling robustness to uneven terrain [8], and learned controllers augmented with adaptive feedforward compensation have been shown to reject large disturbances [9]. Unfortunately, these benefits come at a price: RL tends to scale poorly with the size of the given Markov decision process (MDP), making it challenging to perform the exploration needed to discover high-performance policies.

To mitigate this burden, an RL agent should share experience across all those states that can be considered "equivalent" with respect to the reward and dynamics. Indeed, robotic systems enjoy substantial symmetry [10]–[12], which has been thoroughly exploited in analytical control design [13]–[15] and optimization [16]. In fact, many learned controllers have leveraged symmetry in an ad hoc or approximate manner (*e.g.*, penalizing the *error* between actual and reference states [7] or working in the body frame [9]). More formally, the optimal policy of an MDP with symmetry is equivariant (and its value function is invariant) [17], and neural architectures can be designed accordingly to improve sample efficiency and generalization [18].

Instead of incorporating symmetry into the network architecture, [19] proposed "MDP homomorphisms", which establish a mapping from the given MDP to one of lower dimension. There, a policy may be trained more easily (using standard tools) and then lifted back to the original setting. Such methods were originally restricted to discrete state and action spaces, necessitating coarse discretization of robotic tasks (which are naturally described on smooth manifolds). [20] explored related ideas in continuous state and action spaces, but assumed deterministic dynamics (whereas stochasticity is fundamental to many tasks). However, [21] recently extended the theory of homomorphisms of stochastic MDPs to the continuous setting, recovering analogous value equivalence and policy lifting results. They also learned approximate homomorphisms from data, but do not give a sufficient condition to construct a well-behaved homomorphism (*i.e.*, for which the new state and action spaces are also smooth manifolds) from a continuous symmetry known *a priori* (as is the case for free-flying robotic systems [11]).

In this work, we explore the role of the continuous symmetries of free-flying robotic systems in learned tracking control. After reviewing mathematical preliminaries in Sec. II, in Sec. III we cast a general tracking control problem as a continuous MDP, using a stochastic process to model the (*a priori* unknown) reference trajectory. We show that this MDP inherits the symmetry enjoyed by the underlying dynamics and running costs, and in Sec. IV we show that such symmetries can be used to construct an MDP homomorphism, reducing the dimensionality. Unfortunately, the proofs are relegated to a forthcoming preprint due to space constraints. In Sec. V, we briefly describe three example systems. Finally, in Sec. VI we use these tools to learn tracking controllers for the example systems, accelerating training,

* indicates equal contribution. The authors are with the GRASP Laboratory at the University of Pennsylvania, Philadelphia, PA 19104 USA. emails: {`jwelde`, `nishrao`, `pratikk`, `dineshj`, `kumar`}`@seas.upenn.edu`. We gratefully acknowledge the support of ARL DCIST CRA W911NF-17-2-0181, NSF Grant CCR-2112665, NVIDIA, and the NSF Graduate Research Fellowship Program. Open-source code is available at `https://github.com/PratikKunapuli/EQTrackingControl`.

improving tracking accuracy, and generalizing zero-shot to new trajectories. We discuss our results and contributions in Secs. VII-VIII. Ultimately, these insights will facilitate the efficient development of accurate tracking controllers for various robotic systems.

II. Background and Preliminaries

We now introduce some mathematical concepts. $\mathsf{B}(\mathcal{X})$ denotes the Borel σ-algebra of $\mathcal{X}$, and $\Delta(\mathcal{X})$ denotes the set of Borel probability measures on $\mathcal{X}$ (see [21, Appx. B]). Throughout the paper, we largely follow the treatment of [21], which (along with their prior work [22]) extends [19] to study homomorphisms of Markov decision processes with continuous (*i.e.*, not discrete) state and action spaces.

Definition 1 *(see [21]).* A *continuous Markov decision process*[1] (*i.e.*, an MDP) is a tuple $\mathcal{M} = (\mathcal{S}, \mathcal{A}, R, \tau, \gamma)$, where:

- the *state space* $\mathcal{S}$ is a smooth manifold,
- the *action space* $\mathcal{A}$ is a smooth manifold,
- the *instantaneous reward* is $R : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$,
- the *transition dynamics* are $\tau : \mathcal{S} \times \mathcal{A} \to \Delta(\mathcal{S})$, and
- the *discount factor* γ is a value in the interval $[0, 1)$.

After taking action a_t from state s_t, the probability that s_{t+1} is contained in a set $B \in \mathsf{B}(\mathcal{S})$ is given by $\tau(B \,|\, s_t, a_t)$. A *policy* for $\mathcal{M}$ is a map $\pi : \mathcal{S} \to \Delta(\mathcal{A})$. The *action-value function* $Q^\pi : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$ of a given policy π is defined by

$$Q^\pi(s,a) := \mathop{\mathbb{E}}_{\tau\sim\pi}\left[\sum_{t=0}^{\infty}\gamma^t R(s_t,a_t)\,\middle|\, s_0 = s, a_0 = a\right], \quad (1)$$

where $\tau \sim \pi$ denotes the expectation over both the transitions and the policy (*i.e.*, $s_{t+1} \sim \tau(\,\cdot\,|\, s_t, a_t)$ and $a_t \sim \pi(\,\cdot\,|\, s_t)$ for all $t \in \mathbb{N}$). A policy π^* is *optimal* if, for all $s \in \mathcal{S}$,

$$\pi^* = \arg\max_{\pi} \mathop{\mathbb{E}}_{\tau\sim\pi}\left[\sum_{t=0}^{\infty}\gamma^t R(s_t,a_t)\,\middle|\, s_0 = s\right]. \quad (2)$$

A. Homomorphisms of Markov Decision Processes

The following notion describes a powerful link between two continuous MDPs of (perhaps) different dimensions.

Definition 2 *(see [21, Defs. 11 and 14]).* A pair of maps $p : \mathcal{S} \to \widetilde{\mathcal{S}}$ and $h : \mathcal{S} \times \mathcal{A} \to \widetilde{\mathcal{A}}$ is called a *continuous MDP homomorphism* from $\mathcal{M} = (\mathcal{S}, \mathcal{A}, R, \tau, \gamma)$ to $\widetilde{\mathcal{M}} = (\widetilde{\mathcal{S}}, \widetilde{\mathcal{A}}, \widetilde{R}, \widetilde{\tau}, \gamma)$ if p and, for each $s \in \mathcal{S}$, the map $h_s : a \mapsto h(s, a)$ are measurable, surjective maps, such that

$$R(s,a) = \widetilde{R}\big(p(s), h(s,a)\big), \quad (3a)$$

$$\tau\big(p^{-1}(\widetilde{B}) \,|\, s, a\big) = \widetilde{\tau}\big(\widetilde{B} \,|\, p(s), h(s,a)\big) \quad (3b)$$

for all $s \in \mathcal{S}$, $a \in \mathcal{A}$, and $\widetilde{B} \in \mathsf{B}(\widetilde{S})$. Given a continuous MDP homomorphism (p, h), a policy π for $\widetilde{\mathcal{M}}$, and a policy $\widetilde{\pi}$ for $\mathcal{M}$, π is called a *lift of* $\widetilde{\pi}$ if for all $s \in \mathcal{S}$ and $A \in \mathsf{B}(\mathcal{A})$,

$$\pi\big(h_s^{-1}(\widetilde{A}) \,|\, s\big) = \widetilde{\pi}\big(\widetilde{A} \,|\, p(s)\big). \quad (4)$$

Subsequently, we often omit the word "continuous" for brevity. MDP homomorphisms facilitate the synthesis of an optimal policy for the original MDP $\mathcal{M}$ from an optimal policy for the "quotient" MDP $\widetilde{\mathcal{M}}$, via the following theorem.

Theorem 1 (see [21, Thms. 12 and 16])**.** *Suppose (p, h) is an MDP homomorphism from $\mathcal{M}$ to $\widetilde{\mathcal{M}}$ and π is a lift of* any *policy $\widetilde{\pi}$ for $\widetilde{\mathcal{M}}$. Then, $Q^\pi(s,a) = \widetilde{Q}^{\widetilde{\pi}}\big(p(s), h(s,a)\big)$. Moreover, if $\widetilde{\pi}$ is optimal for $\widetilde{\mathcal{M}}$, then π is optimal for $\mathcal{M}$.*

B. Lie Group Symmetries of Markov Decision Processes

A *(left) group action* of a Lie group $\mathcal{G}$ on a smooth manifold $\mathcal{X}$ is a smooth map $\Phi : \mathcal{G} \times \mathcal{X} \to \mathcal{X}$ (often written $\Phi_g(x) := \Phi(g, x)$ for brevity) such that for all $x \in \mathcal{X}$ and $g, h \in \mathcal{G}$, $\Phi(1_\mathcal{G}, x) = x$ (where $1_\mathcal{G} \in \mathcal{G}$ is the identity) and $\Phi\big(g, \Phi(h, x)\big) = \Phi\big(gh, x\big)$. The Φ-*orbit* of x is the set $\Phi_G(x) := \{\Phi_g(x) : g \in \mathcal{G}\}$, while $\mathcal{X}/\mathcal{G}$ is a set whose elements are all the orbits of Φ. An action Φ is *proper* if the map $(g, x) \mapsto \big(\Phi_g(x), x\big)$ is proper (*i.e.*, the preimage of any compact set is compact), and *free* if $\Phi_g(x) = x$ implies $g = 1_\mathcal{G}$. A group $\mathcal{G}$ acts on itself via $L : (h, g) \mapsto hg$.

A group action can describe a symmetry of some object defined on the manifold. We now formulate the following definition of a Lie group symmetry of a continuous MDP.

Definition 3. Given an MDP $\mathcal{M} = (\mathcal{S}, \mathcal{A}, R, \tau, \gamma)$, a pair of Lie group actions (Φ, Ψ) of $\mathcal{G}$ on $\mathcal{S}$ and $\mathcal{A}$ respectively is called a *Lie group symmetry* of $\mathcal{M}$ if, for all Φ-invariant sets $B \in \mathsf{B}(\mathcal{S})$ and all $s \in \mathcal{S}$, $a \in \mathcal{A}$, and $g \in \mathcal{G}$, we have

$$R\big(s,a\big) = R\big(\Phi_g(s), \Psi_g(a)\big), \quad (5a)$$

$$\tau\big(B \,|\, s, a\big) = \tau\big(B \,|\, \Phi_g(s), \Psi_g(a)\big). \quad (5b)$$

Remark 1. The qualifier "Φ-invariant" on B broadens the class of symmetries considered (and is more general than [17] and [18], as noted in [23, Def. 35]). The deterministic case (*i.e.*, when $\tau(\,\cdot\,|\, s_t, a_t)$ is the Dirac measure corresponding to $\{s_{t+1}\} \subseteq \mathcal{S}$) gives the intuition, since then (5b) requires the image of any orbit in $\mathcal{S} \times \mathcal{A}$ to lie within some orbit in $\mathcal{S}$, *without* enforcing equivariance *within* each orbit.

III. Tracking Control Problems With Lie Group Symmetries

In this section, we formulate a general trajectory tracking problem as an MDP that models the evolution of both the physical and reference systems. We give a sufficient condition for this MDP to have a Lie group symmetry that will be used (in Sec. IV) to reduce the problem size.

Definition 4. A *tracking control problem* is a tuple $\mathcal{T} = (\mathcal{X}, \mathcal{U}, f, J_\mathcal{X}, J_\mathcal{U}, \rho, \gamma)$, where:

- $\mathcal{X}$ is the *physical state space* (a smooth manifold),
- $\mathcal{U}$ is the *physical action space* (a smooth manifold),
- $f : \mathcal{X} \times \mathcal{U} \to \Delta(\mathcal{X})$ is the the *physical dynamics* (*i.e.*, $x_{t+1} \sim f(\,\cdot\,|\, x_t, u_t)$ describes the system's evolution),
- $J_\mathcal{X} : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ is the *tracking cost*,
- $J_\mathcal{U} : \mathcal{U} \times \mathcal{U} \to \mathbb{R}$ is the *effort cost*,
- $\rho \in \Delta(\mathcal{U})$ is the *reference action distribution*, and
- $\gamma \in [0, 1)$ is the *discount factor*.

The distribution ρ is not usually included in the definition of a tracking problem but will play an essential role in our approach (see Remark 2).

[1] The more general definition in [21] does *not* assume $\mathcal{S}$ and $\mathcal{A}$ are smooth manifolds, nor that $\tau(\,\cdot\,|\, s, a)$ is a Borel measure, but this is all we need.

A. Modeling a Tracking Control Problem as an MDP

We model the tracking task for reference trajectories that are unknown *a priori* in the following manner.

Definition 5. A given tracking control problem $\mathcal{T} = \big(\mathcal{X}, \mathcal{U}, f, J_{\mathcal{X}}, J_{\mathcal{U}}, \rho, \gamma\big)$ induces a *tracking control MDP* given by $\mathcal{M}_{\mathcal{T}} = (\mathcal{S} = \mathcal{X} \times \mathcal{X} \times \mathcal{U}, \mathcal{A} = \mathcal{U}, R, \tau, \gamma)$, where:

- the state is $(x, x^{\mathrm{d}}, u^{\mathrm{d}})$, where x, $x^{\mathrm{d}} \in \mathcal{X}$ are the *actual* and *reference states* and $u^{\mathrm{d}} \in \mathcal{U}$ is the *reference action*,
- the *actions* are $a = u \in \mathcal{U}$ (*i.e.*, the *actual action*),
- the instantaneous reward $R : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$ is given by

$$R\big((x, x^{\mathrm{d}}, u^{\mathrm{d}}), u\big) := -J_{\mathcal{X}}(x, x^{\mathrm{d}}) - J_{\mathcal{U}}(u, u^{\mathrm{d}}), \quad (6)$$

- and the transitions $\tau : \mathcal{S} \times \mathcal{A} \to \Delta(\mathcal{S})$ are defined by

$$\begin{aligned} x_{t+1} &\sim f(\,\cdot\,|\, x_t, u_t), \\ x^{\mathrm{d}}_{t+1} &\sim f(\,\cdot\,|\, x^{\mathrm{d}}_t, u^{\mathrm{d}}_t), \quad u^{\mathrm{d}}_{t+1} \sim \rho. \end{aligned} \quad (7)$$

Remark 2. This formulation allows us to model a tracking control problem over a broad class of reference trajectories (*i.e.*, those generated by a certain stochastic process) as a single *stationary* MDP (*i.e.*, with time-invariant transitions and reward). While we could also formulate a (*non-stationary*) MDP corresponding to a *particular* reference trajectory by making the tracking cost a function of time t and the actual state x, an optimal policy for that MDP would be useless for tracking *other* references. In Sec. VI, we will show empirically that policies trained in the proposed manner also effectively track pre-planned reference trajectories, for which the sequence of reference actions $\{u^{\mathrm{d}}_0, u^{\mathrm{d}}_1, u^{\mathrm{d}}_2, \cdots\}$ is chosen to induce a pre-selected state trajectory $\{x^{\mathrm{d}}_0, x^{\mathrm{d}}_1, x^{\mathrm{d}}_2, \cdots\}$.

B. Symmetries of Tracking Control MDPs

We now show that the MDP induced by a tracking control problem with certain symmetries will inherit a related symmetry with certain convenient properties.

Theorem 2. *Consider a tracking control problem $\mathcal{T} = (\mathcal{X}, \mathcal{U}, f, J_{\mathcal{X}}, J_{\mathcal{U}}, \rho, \gamma)$ as well as Lie group actions $\Upsilon : \mathcal{K} \times \mathcal{X} \to \mathcal{X}$ and $\Theta : \mathcal{H} \times \mathcal{U} \to \mathcal{U}$. Suppose that:*

- *$J_{\mathcal{X}}$ is Υ-invariant and $J_{\mathcal{U}}$ is Θ-invariant,* i.e.*, for all $x, x^{\mathrm{d}} \in \mathcal{X}$, $u, u^{\mathrm{d}} \in \mathcal{U}$, $k \in \mathcal{K}$, and $h \in \mathcal{H}$, we have*

$$J_{\mathcal{X}}(x, x^{\mathrm{d}}) = J_{\mathcal{X}}\big(\Upsilon_k(x), \Upsilon_k(x^{\mathrm{d}})\big), \quad (8a)$$

$$J_{\mathcal{U}}(u, u^{\mathrm{d}}) = J_{\mathcal{U}}\big(\Theta_h(u), \Theta_h(u^{\mathrm{d}})\big). \quad (8b)$$

- *For each $(k, h) \in \mathcal{K} \times \mathcal{H}$, there exists $k' \in \mathcal{K}$ such that for all $(x, u) \in \mathcal{X} \times \mathcal{U}$ and $B \in \mathsf{B}(\mathcal{X})$, we have*

$$f\big(\Upsilon_{k'}(B) \,|\, x, u\big) = f\big(B \,|\, \Upsilon_k(x), \Psi_h(u)\big). \quad (9)$$

Define actions of the direct product group $\mathcal{G} = \mathcal{K} \times \mathcal{H}$ on $\mathcal{S} = \mathcal{X} \times \mathcal{X} \times \mathcal{U}$ and $\mathcal{A} = \mathcal{U}$, given respectively by

$$\Phi_{(k,h)}(x, x^{\mathrm{d}}, u^{\mathrm{d}}) := \big(\Upsilon_k(x), \Upsilon_k(x^{\mathrm{d}}), \Theta_h(u^{\mathrm{d}})\big), \quad (10a)$$

$$\Psi_{(k,h)}(u) := \Theta_h(u). \quad (10b)$$

Then, (Φ, Ψ) is a Lie group symmetry of $\mathcal{M}_{\mathcal{T}}$. Moreover, if Υ and Θ are free and proper, then Φ is also free and proper.

Remark 3. Because we do *not* assume that $k' = k$, (9) is more general than equivariance of the transitions. However, k' must depend only on k and h, and not on x and u.

IV. Continuous MDP Homomorphisms Induced by Lie Group Symmetries

We will use the following theorem to show that symmetries of a tracking control MDP can be used to reduce its dimension via a homomorphism and also give an explicit formula for policy lifting. Although related results are known in the discrete [19] and deterministic [20] settings, we require a more general result due to our continuous state and action spaces and the random sampling of the reference actions (even when the underlying dynamics f are deterministic).

Theorem 3. *Consider an MDP $\mathcal{M} = (\mathcal{S}, \mathcal{A}, R, \tau, \gamma)$ with a Lie group symmetry (Φ, Ψ). Suppose that Φ is free and proper and $\lambda : \mathcal{S} \to \mathcal{G}$ is any equivariant map. Define*

$$p : S \to \mathcal{S}/\mathcal{G}, \;\; s \mapsto \Phi_G(s), \quad (11a)$$

$$h : \mathcal{S} \times \mathcal{A} \to \mathcal{A}, \;\; (s, a) \mapsto \Psi_{\lambda(s)^{-1}}(a). \quad (11b)$$

Then, (p, h) is an MDP homomorphism from $\mathcal{M}$ to $\widetilde{\mathcal{M}} = \big(\widetilde{\mathcal{S}} = \mathcal{S}/\mathcal{G}, \widetilde{\mathcal{A}} = \mathcal{A}, \widetilde{R}, \widetilde{\tau}, \gamma\big)$, where we define

$$\widetilde{R}(\tilde{s}, \tilde{a}) := R\big(s, \Psi_{\lambda(s)}(\tilde{a})\big)\,\big|_{\,s\,\in\,p^{-1}(\tilde{s})}, \quad (12a)$$

$$\widetilde{\tau}(\widetilde{B} \,|\, \tilde{s}, \tilde{a}) := \tau\big(p^{-1}(\widetilde{B}) \,|\, s, \Psi_{\lambda(s)}(\tilde{a})\big)\,\big|_{\,s\,\in\,p^{-1}(\tilde{s})} \quad (12b)$$

independent of the particular choice of s. Also, for any policy $\widetilde{\pi}$ for $\widetilde{\mathcal{M}}$, a policy for $\mathcal{M}$ that is a lift of $\widetilde{\pi}$ is given by

$$(\widetilde{\pi})^{\uparrow}(A \,|\, s) := \widetilde{\pi}\big(\Psi_{\lambda(s)^{-1}}(A) \,|\, p(s)\big). \quad (13)$$

V. Quotient MDPs for Tracking Control in Free-Flying Robotic Systems

Clearly, Theorems 1, 2, and 3 can be applied together to reduce the MDP induced by a tracking control problem with symmetry in its dynamics and running costs. We refer to our open-source code for details (due to space constraints).

Example 1 *(*`Particle`*)***.** Consider a point mass m with physical state $(r, v) \in T\mathbb{R}^3$ controlled by a force in $\mathbb{R}^3$. Running costs penalize error in position, velocity, and actions. Using the left actions of $T\mathbb{R}^3$ and $\mathbb{R}^3$ on themselves, we derive an MDP homomorphism where $h(s, a) := u - u^{\mathrm{d}}$ and

$$p\big((r, v), (r^{\mathrm{d}}, v^{\mathrm{d}}), u^{\mathrm{d}}\big) := (r - r^{\mathrm{d}}, v - v^{\mathrm{d}}), \quad (14)$$

reducing the tracking MDP from $T\mathbb{R}^3 \times T\mathbb{R}^3 \times \mathbb{R}^3$ to $T\mathbb{R}^3$.

Example 2 *(*`Astrobee` *[24])***.** This rigid body's state is its pose and twist $x = (q, \xi)$ in $\mathcal{X} = SE(3) \times \mathbb{R}^6$, and the action is a wrench $u \in \mathbb{R}^6$. Running costs penalize error in pose, twist, and wrench. Using $SE(3)$ symmetry [11], we derive an MDP homomorphism with $h(s, a) := a$ and

$$p(q, \xi, q^{\mathrm{d}}, \xi^{\mathrm{d}}, u^{\mathrm{d}}) := \big(q^{-1}q^{\mathrm{d}}, \xi, \xi^{\mathrm{d}}, u^{\mathrm{d}}\big), \quad (15)$$

reducing the tracking MDP's state space by 6 dimensions.

Example 3 *(*`Quadrotor` *[25])***.** This underactuated aerial robot has the same state space as the `Astrobee`, but the actions are the "single-rotor thrusts" $u \in \mathcal{U} = \mathbb{R}^4$ and gravity reduces the symmetry group to $SE(2) \times \mathbb{R}$ (corresponding to horizontal plane displacements and vertical translations). We derive an MDP homomorphism with $h(s, a) := a$ and

$$p(q, \xi, q^{\mathrm{d}}, \xi^{\mathrm{d}}, u^{\mathrm{d}}) := (q^{-1}q^{\mathrm{d}}, R^{\mathrm{T}}\mathrm{e}_3, \xi, \xi^{\mathrm{d}}, u^{\mathrm{d}}), \quad (16)$$

TABLE I

Comparison of RMS Tracking Error on Planned Trajectories

Environment	$\mathcal{G}$	r [m]	v [m/s]	R [rad]	ω [rad/s]
`Particle`	Baseline	2.50±0.25	0.93±0.11	-	-
	$\mathbb{R}^3$	0.12±0.4	0.06±0.23	-	-
	$T\mathbb{R}^3$	0.11±0.41	**0.04±0.19**	-	-
	$T\mathbb{R}^3\times\mathbb{R}^3$	**0.09±0.42**	**0.04±0.19**	-	-
`Astrobee`	Baseline	0.23±0.01	0.17±0.01	0.70±0.03	1.91±0.08
	$SE(3)$	**0.10±0.01**	**0.03±0.04**	**0.41±0.04**	**1.51±0.07**
`Quadrotor`	Baseline	0.91±0.62	0.97±0.78	0.51±0.19	0.27±0.21
	$SE(2)\times\mathbb{R}$	**0.25±0.30**	**0.14±0.26**	**0.29±0.02**	**0.04±0.07**

We report the mean and standard deviation (over $n = 20$ training seeds) of the policy's RMS tracking error (on a dataset of $m = 20$ trajectories).

noting that $R^\mathrm{T}\mathrm{e}_3$ is the gravity direction in body coordinates.

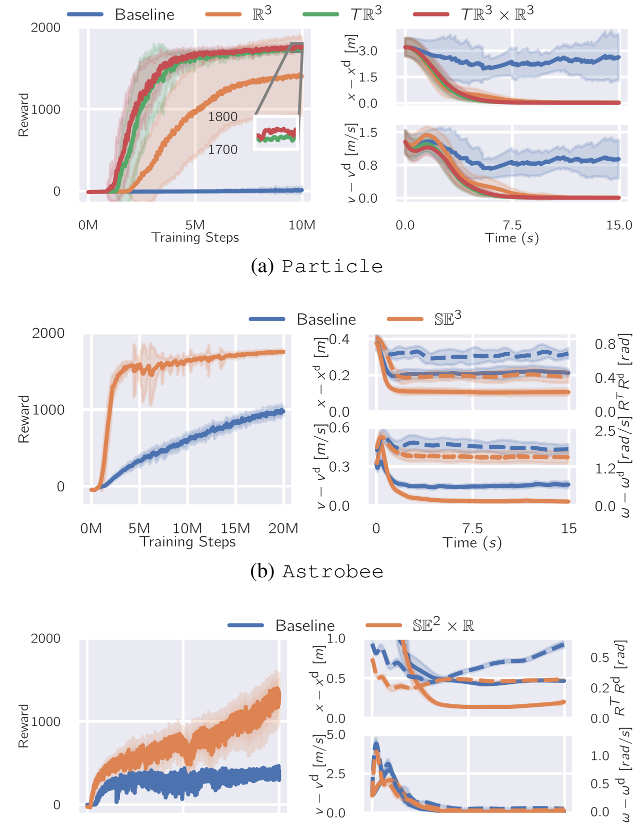
VI. Experiments

We now explore the effects of our symmetry-informed approach on sample efficiency and performance of model-free reinforcement learning for tracking control. RL environments were implemented for each of the tracking control MDPs in Examples 1-3, written in `jax` [26] for performance. To implement environments for the quotient MDP arising from reduction by a symmetry group, we modify each environment's observation to the reduced state given in (14), (15), and (16) (whereas the baseline sees the full-state observation $(x, x^\mathrm{d}, u^\mathrm{d})$). We also modify the actions according to the definition of h. For the `Particle` environment, we isolate the effects of reduction by different subgroups of the symmetry by also implementing environments reduced by translational symmetry alone (*i.e.*, $p(s) := (r - r^\mathrm{d}, v, v^\mathrm{d}, u^\mathrm{d})$) and by translational and velocity symmetry alone (*i.e.*, $p(s) := (r - r^\mathrm{d}, v - v^\mathrm{d}, u^\mathrm{d})$).

We use a custom implementation of PPO [27] (see code for details), with the same hyperparameters across all variants of each environment. During training, the reference actions are sampled from a stationary distribution (as in Def. 5), but we evaluate zero-shot on pre-planned (dynamically feasible) reference trajectories. Fig. 1 and Table I report total reward (during training) and average tracking error (during evaluation), starting the system from a randomized initial state.

VII. Discussion

Fig. 1 shows a clear trend across the board: greater symmetry exploitation leads to improved sample efficiency. The tracking error evaluation shown in Table I and Fig. 1 follows a similar trend. For the `Particle`, the vast majority of this benefit is achieved by reduction of the translational symmetry, although incorporating the velocity and force symmetries yields modest additional gains. This seems consistent with the large improvement we see for the `Astrobee` and `Quadrotor` after reduction by (a subgroup of) $SE(3)$. Careful reward engineering or hyperparameter tuning might improve performance (especially for the baseline, which currently fails to learn effectively), but we instead focus on analyzing the benefit of exploiting symmetry for a fixed reward. Nonetheless, any reward depending only on the reduced state $\tilde{s} = p(s)$ would preserve the symmetry.



(a) `Particle`

(b) `Astrobee`

(c) `Quadrotor`

Fig. 1. Reward during training and tracking error components during evaluation for the `Particle`, `Astrobee`, and `Quadrotor`, with translational errors as solid lines and rotational errors (when applicable) as dashed lines.

Our approach assumes that at deployment, an upstream planner provides dynamically feasible reference trajectories. For the (underactuated) `Quadrotor`, these trajectories are planned using differential flatness [25] from Lissajous curves in the flat space. However, in theory any other method (*e.g.*, direct collocation [28]) could be used to generate a suitable reference. We expect our policies to generalize well to a wide range of upstream planning methodologies, and future work should explore this hypothesis. Going forward, we also hope to apply these methods to new robot morphologies that are too complex for real-time numerical optimal control or for which no closed-form analytical controllers are known.

VIII. Conclusion

In this work, we exploit the natural Lie group symmetries of free-flying robotic systems to mitigate the challenges of learning trajectory tracking controllers. We formulate the tracking problem as a single stationary MDP, proving that the underlying symmetries of the dynamics and running costs permit the reduction of this MDP to a lower-dimensional problem. When learning tracking controllers for space and aerial robots, training is accelerated and tracking error is reduced after the same number of training steps. We believe our theoretical framework provides insight into the use of RL for systems with symmetry in robotics applications.

REFERENCES

[1] F. Bullo and R. M. Murray, "Tracking for fully actuated mechanical systems: a geometric framework," *Automatica*, vol. 35, no. 1, pp. 17–34, 1999.

[2] D. Maithripala, J. Berg, and W. Dayawansa, "Almost-global tracking of simple mechanical systems on a general class of lie groups," *IEEE Transactions on Automatic Control*, vol. 51, no. 2, pp. 216–225, 2006.

[3] J. Welde and V. Kumar, "Almost Global Asymptotic Trajectory Tracking for Fully-Actuated Mechanical Systems on Homogeneous Riemannian Manifolds," *IEEE Control Systems Letters*, vol. 8, pp. 724–729, 2024.

[4] M. Fliess, J. Levine, P. Martin, F. Ollivier, and P. Rouchon, "Controlling nonlinear systems by flatness," in *Systems and Control in the Twenty-First Century*, C. I. Byrnes, B. N. Datta, C. F. Martin, and D. S. Gilliam, Eds. Boston, MA: Birkhäuser Boston, 1997, pp. 137–154.

[5] K. Nguyen, S. Schoedel, A. Alavilli, B. Plancher, and Z. Manchester, "TinyMPC: Model-Predictive Control on Resource-Constrained Microcontrollers," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2024.

[6] J. Hwangbo, I. Sa, R. Y. Siegwart, and M. Hutter, "Control of a Quadrotor With Reinforcement Learning," *IEEE Robotics and Automation Letters*, vol. 2, pp. 2096–2103, 2017.

[7] A. Molchanov, T. Chen, W. Hönig, J. A. Preiss, N. Ayanian, and G. S. Sukhatme, "Sim-to-(Multi)-Real: Transfer of Low-Level Robust Control Policies to Multiple Quadrotors," in *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2019, pp. 59–66.

[8] N. Rudin, D. Hoeller, P. Reist, and M. Hutter, "Learning to Walk in Minutes Using Massively Parallel Deep Reinforcement Learning," in *Proceedings of the 5th Conference on Robot Learning*, ser. Proceedings of Machine Learning Research, A. Faust, D. Hsu, and G. Neumann, Eds., vol. 164. PMLR, 08–11 Nov 2022, pp. 91–100.

[9] K. Huang, R. Rana, A. Spitzer, G. Shi, and B. Boots, "DATT: Deep Adaptive Trajectory Tracking for Quadrotor Control," in *Proceedings of The 7th Conference on Robot Learning*, ser. Proceedings of Machine Learning Research, J. Tan, M. Toussaint, and K. Darvish, Eds., vol. 229. PMLR, 06–09 Nov 2023, pp. 326–340.

[10] R. M. Murray, "Nonlinear control of mechanical systems: A Lagrangian perspective," *Annual Reviews in Control*, vol. 21, pp. 31–42, 1997.

[11] J. Ostrowski, "Computing reduced equations for robotic systems with constraints and symmetries," *IEEE Transactions on Robotics and Automation*, vol. 15, no. 1, pp. 111–123, 1999.

[12] D. F. Ordonez-Apraez, M. Martin, A. Agudo, and F. Moreno, "On discrete symmetries of robotics systems: A group-theoretic and data-driven analysis," in *Proceedings of Robotics: Science and Systems*, Daegu, Republic of Korea, July 2023.

[13] R. L. Hatton, Z. Brock, S. Chen, H. Choset, H. Faraji, R. Fu, N. Justus, and S. Ramasamy, "The geometry of optimal gaits for inertia-dominated kinematic systems," *IEEE Transactions on Robotics*, vol. 38, no. 5, pp. 3279–3299, 2022.

[14] J. Welde, M. D. Kvalheim, and V. Kumar, "The Role of Symmetry in Constructing Geometric Flat Outputs for Free-Flying Robotic Systems," in *2023 IEEE International Conference on Robotics and Automation (ICRA)*, 2023, pp. 12 247–12 253.

[15] M. Hampsey, P. van Goor, T. Hamel, and R. Mahony, "Exploiting different symmetries for trajectory tracking control with application to quadrotors," *IFAC-PapersOnLine*, vol. 56, no. 1, pp. 132–137, 2023, 12th IFAC Symposium on Nonlinear Control Systems NOLCOS 2022.

[16] S. Teng, D. Chen, W. Clark, and M. Ghaffari, "An Error-State Model Predictive Control on Connected Matrix Lie Groups for Legged Robot Control," in *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2022, pp. 8850–8857.

[17] D. Wang, R. Walters, X. Zhu, and R. Platt, "Equivariant Q Learning in Spatial Action Spaces," in *Proceedings of the 5th Conference on Robot Learning*, ser. Proceedings of Machine Learning Research, A. Faust, D. Hsu, and G. Neumann, Eds., vol. 164. PMLR, 08–11 Nov 2022, pp. 1713–1723.

[18] E. van der Pol, D. Worrall, H. van Hoof, F. Oliehoek, and M. Welling, "MDP Homomorphic Networks: Group Symmetries in Reinforcement Learning," in *Advances in Neural Information Processing Systems*, H. Larochelle, M. Ranzato, R. Hadsell, M. Balcan, and H. Lin, Eds., vol. 33. Curran Associates, Inc., 2020, pp. 4199–4210.

[19] B. Ravindran, "An algebraic approach to abstraction in reinforcement learning," Ph.D. dissertation, University of Massachusetts Amherst, 2004.

[20] B. Yu and T. Lee, "Equivariant Reinforcement Learning for Quadrotor UAV," in *2023 American Control Conference (ACC)*, 2023, pp. 2842–2847.

[21] P. Panangaden, S. Rezaei-Shoshtari, R. Zhao, D. Meger, and D. Precup, "Policy Gradient Methods in the Presence of Symmetries and State Abstractions," *Journal of Machine Learning Research*, vol. 25, no. 71, pp. 1–57, 2024.

[22] S. Rezaei-Shoshtari, R. Zhao, P. Panangaden, D. Meger, and D. Precup, "Continuous MDP Homomorphisms and Homomorphic Policy Gradient," in *Advances in Neural Information Processing Systems*, vol. 35, 2022, pp. 20 189–20 204.

[23] R. Y. Zhao, "Continuous Homomorphisms and Leveraging Symmetries in Policy Gradient Algorithms for Markov Decision Processes," Master's thesis, McGill University, 2022.

[24] M. Bualat, J. Barlow, T. Fong, C. Provencher, and T. Smith, "Astrobee: Developing a free-flying robot for the international space station," in *AIAA SPACE 2015 conference and exposition*, 2015, p. 4643.

[25] D. Mellinger and V. Kumar, "Minimum snap trajectory generation and control for quadrotors," in *2011 IEEE International Conference on Robotics and Automation*, 2011, pp. 2520–2525.

[26] J. Bradbury, R. Frostig, P. Hawkins, M. J. Johnson, C. Leary, D. Maclaurin, G. Necula, A. Paszke, J. VanderPlas, S. Wanderman-Milne, and Q. Zhang, "JAX: composable transformations of Python+NumPy programs," 2018. [Online]. Available: http://github.com/google/jax

[27] J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov, "Proximal Policy Optimization Algorithms," *arXiv preprint arXiv:1707.06347*, 2017.

[28] M. Kelly, "An introduction to trajectory optimization: How to do your own direct collocation," *SIAM Review*, vol. 59, no. 4, pp. 849–904, 2017.