Learning Multi-Robot Coordination with Invariant Consensus Stabilization

Hang Yin HAYI@DI.KU.DK
Department of Computer Science, University of Copenhagen, Denmark

Christos K. Verginis CHRISTOS.VERGINIS@ANGSTROM.UU.SE
Department of Electrical Engineering, Uppsala University, Sweden

Danica Kragic DANI@KTH.SE
Robotics, Perception and Learning Group, KTH Royal Institute of Technology, Sweden

Editors: G. Sukhatme, L. Lindemann, S. Tu, A. Wierman, N. Atanasov

Abstract

Coordinating multi-robot systems for highly dexterous tasks is challenging due to the complexity of inducing desired interactions among robots with high-dimensional dynamics. This paper introduces a learning-based multi-robot control algorithm that generates complex trajectories for executing such tasks. In particular, we design a controller that achieves multi-robot consensus; unlike standard consensus protocols, the controller is parametrized by neural networks that are derived from convex potentials and represent diffeomorphic functions of the robots' relative states. The algorithm trains the neural networks to learn consensus policies that enable coordinated, high-precision multi-robot behaviors. A key feature of our approach is translation invariance, which ensures generalization to untrained state spaces. We prove the theoretical correctness of the proposed algorithm for an arbitrary number of robots and validate its effectiveness in two dynamic tasks, namely cooperative object transportation and forceful peg insertion. The results show that the proposed controller and policy learning significantly outperform baseline methods in terms of learning efficiency and generalization under untrained task configurations.

Keywords: Multi-robot coordination; Equivariant representation; Reinforcement learning

1. Introduction

Coordination of multi-robot systems has received significant attention during the last decades due to the advantages over single-robot setups (Verginis, 2020). Critical applications range from swarm navigation (Han et al., 2022), satellite formation (Ahn, 2020), and cooperative object manipulation (Verginis et al., 2019b). Multi-robot coordination can entail tasks, such as cooperative peg insertion in a socket, that require high, often human-like, levels of dexterity for safe interaction, i.e., avoid intra- and inter-robot collisions, and accommodate physical contacts.

Control methods typically employ distributed consensus protocols to achieve multi-robot coordination (Olfati-Saber et al., 2007), which, however, cannot produce complex trajectories required for the safe execution of highly dexterous tasks. Collision-free motion is often addressed by integrating path-tracking methods with planning or potential-field methods (Verginis et al., 2022; Panagou, 2016). Nevertheless, potential fields suffer from local minimas, whereas planning-based methods are limited in the tracking accuracy required in high-precision tasks, such as peg-in-hole insertion. Additionally, planning methods are typically developed in a centralized manner, and hence face scalability issues on multiple articulated robots with many degrees of freedom (LaValle, 2006).

Learning-based synthesis of trajectories constitutes a promising direction toward dexterous multi-robot coordination. However, neural network policies used in deep reinforcement learning are usually agnostic to the structure of the coordination task. Dynamical-system-based approaches (Ijspeert et al., 2013; Khansari-Zadeh and Billard, 2011) represent trajectories or autonomous behaviors as parameterized dynamical systems. Learning these parameters concentrates almost exclusively on the convergence of single-robot systems to a fixed point (Rana et al., 2020; Khader et al., 2020, 2021b). These are not directly applicable in multi-robot tasks that accommodate the inter-robot dynamics and the relative robot pose stability. Existing policy learning with consensus stabilization (Yin et al., 2022) addressed these issues while it does not preserve the invariance of common consensus protocols, limiting the policy generalization in untrained state spaces.

CONTRIBUTIONS: This paper studies the multi-robot coordination toward the safe execution of dexterous tasks. We develop a multi-robot algorithm that integrates reinforcement learning with consensus-based stabilization. In particular, our algorithm achieves consensus for robots with 2nd-order continuous-time dynamics, modeling skillful cooperative trajectories with system passivity. The consensus algorithm operates on a diffeomorphic transformation of the robots' relative states. This transformation is encoded by flow networks, whose parameters are trained through a reinforcement-learning procedure to achieve the safety and dexterity needed. The proposed algorithm extends our previous work (Yin et al., 2022) by guaranteeing *translation invariance* for an arbitrary number of robots. That is, the proposed algorithm ensures the execution of the considered multi-robot task independently from the robots' position in the workspace. We evaluate the effectiveness of the algorithm in a comparative experimental setting involving two simulated multi-robot manipulation tasks. The results demonstrate that the proposed algorithm outperforms the algorithm of (Yin et al., 2022) and the baselines with deep neural networks, further illustrating the efficacy of the proposal algorithm to learn the desired skills efficiently.

2. Preliminaries

We begin the presentation with a background introduction on the problem formulation, used notations and supporting techniques such as neural networks with special structures.

2.1. Multi-Robot System and Consensus

We consider n robotic agents, with $\mathcal{N} = \{1, \dots, n\}$, whose configurations are described by the generalized coordinates $\mathbf{x}_i \in \mathbb{R}^d$ for $i \in \mathcal{N}$. Each agent obeys the second-order dynamics with force input τ_i:

$$\mathbf{M}_i(\mathbf{x}_i)\ddot{\mathbf{x}}_i + \mathbf{C}(\mathbf{x}_i, \dot{\mathbf{x}}_i)\dot{\mathbf{x}}_i + \mathbf{g}(\mathbf{x}_i) = \tau_i + \tau_{\text{ext},i}. \tag{1}$$

where $\mathbf{M}_i : \mathbb{R}^d \to \mathbb{R}^{d\times d}$ are the positive definite inertia matrices, $\mathbf{C}_i : \mathbb{R}^d \to \mathbb{R}^{d\times d}$ are the Coriolis matrices, and $\mathbf{g}_i : \mathbb{R}^d \to \mathbb{R}^d$ are the gravity terms, for $i \in \mathcal{N}$; τ_{ext} denotes forces external to the entire system.

The agents communicate with each other over a network represented by an undirected graph $\mathcal{G} = (\mathcal{N}, \mathcal{E})$, where $\mathcal{E} \subseteq \mathcal{N} \times \mathcal{N}$ is the respective edge set. The adjacency matrix associated with the graph $\mathcal{G}$ is denoted by $\mathbf{A} = [a_{ij}] \in \mathbb{R}^{n\times n}$, with $a_{ij} \in \{0, 1\}$, $i, j \in \{1, \dots, n\}$. If $a_{ij} = 1$, then agent i obtains information regarding the state x_j of agent j (i.e., $(i, j) \in \mathcal{E}$), whereas if $a_{ij} = 0$ then there is no state-information flow from agent j to agent i (i.e., $(i, j) \notin \mathcal{E}$). Furthermore, the set

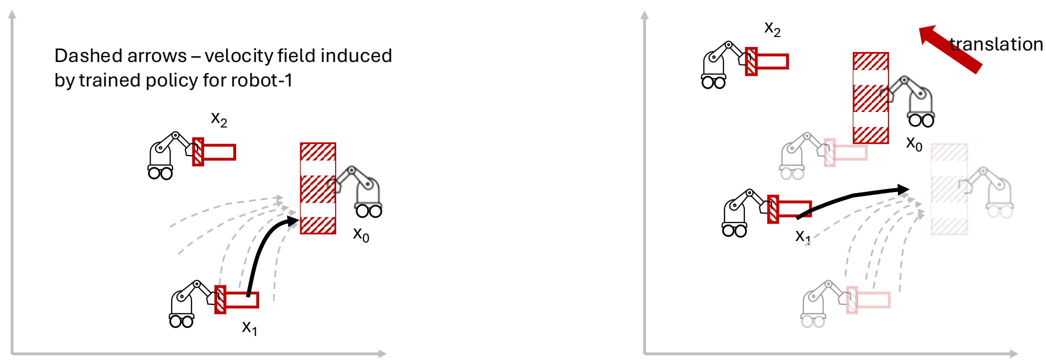

Figure 1: The policy for robot-1 will yield different velocity command at the new location after the whole robot team is translated, or equivalently when the origin of reference frame is shifted. Translation-invariant control is needed for preserving desired interactions.

of neighbors of agent i is denoted by $\mathcal{N}_i = \{j \in \mathcal{N} : (i,j) \in \mathcal{E}\}$, and the degree matrix is defined as $\mathbf{D} = \text{diag}\{|\mathcal{N}_1|, \ldots, |\mathcal{N}_n|\}$. Since the graph is undirected, the adjacency is a mutual relation, i.e., $a_{ij} = a_{ji}$, rendering $\mathbf{A}$ symmetric. We denote the graph Laplacian of the network $\mathbf{L} \in \mathbb{R}^{n\times n}$ by $\mathbf{L} = \mathbf{D} - \mathbf{A}$, which is symmetric and positive semi-definite. For a connected graph[1], the kernel of L is $\mathsf{span}\{\mathbf{1}\}$, where $\mathbf{1}$ is the vector of ones of appropriate dimension, implying that $\mathbf{L1} = 0$. The graph Laplacian has been widely applied to design consensus closed-loop systems

$$\dot{\mathbf{x}} = -\tilde{\mathbf{L}}\mathbf{x} \tag{2}$$

where $\mathbf{x} = [\mathbf{x}_1^\top, ..., \mathbf{x}_n^\top]^\top$ is the aggregated state and $\tilde{\mathbf{L}} = \mathbf{L} \otimes I_d$ denotes the Kronecker product that accounts for multidimensional systems; (2) serves as a linear rule for realizing asymptotic consensus among agents, i.e., $\lim_{t\to\infty} \tilde{\mathbf{L}}\mathbf{x}(t) = 0$. In order to account for safe dexterous tasks, our previous work (Yin et al., 2022) uses a diffeomorphic transformation $\Phi = [\phi^\top, ..., \phi^\top]^\top, \phi : \mathbb{R}^d \longrightarrow \mathbb{R}^d$ in the consensus-stabilizing controller achieving $\tilde{\mathbf{L}}\Phi(\mathbf{x}) = 0$; the proposed controller, however, is not translation-invariant.

2.2. Translation-Invariant Control

A map $\Phi : \mathbb{R}^{n\times d} \to \mathbb{R}^{n\times d}$, is said to be *equivariant* to transformations $T_1, T_2 : \mathbb{R}^{n\times d} \to \mathbb{R}^{n\times d}$ if $\Phi(T_1(\mathbf{x})) = T_2(\Phi(\mathbf{x}))$. A special case of equivarience studied here is when $T_2 = I$, i.e., $\Phi(T_1(\mathbf{x})) = \Phi(\mathbf{x})$, where Φ is said to be *invariant* to T_1. Endowing a control law $\mathbf{u}(\mathbf{x})$ with invariance properties enhances robustness and generalizability by ensuring consistent behavior under transformations such as coordinate changes or scaling. This enables broader applicability across varying operational contexts without redesigning the controller. For instance, for a team of aerial robots performing formation control, a translation-invariant control law ensures that the formation behavior remains consistent regardless of the group's absolute position.

In the context of multiple robots, we illustrate an example of the cooperative behavior in Figure 1, and expect the interactive behaviors to be preserved while approaching each other, e.g.

1. A graph is connected if there exists a path between any two agents.

for avoiding collisions. Clearly, the linear consensus law (2) is invariant to a translation shifts $\mathbf{c} \in \mathbb{R}^d$ due to the nullspace of $\tilde{\mathbf{L}}$. To the contrary, nonlinear laws developed in (Yin et al., 2022) using a transformation $\Phi(\mathbf{x})$ do not exhibit this property since Φ is not necessarily invariant to $T_1(\mathbf{x}) = \mathbf{x} - \mathbf{c}$. Changing the state variable from $\mathbf{x}$ to $\Phi(\mathbf{x})$ therefore breaks the symmetry and makes the control policy more likely to fail at untrained $\mathbf{x}$ and unapplicable when the sensor origin is shifted. Our work aims to address this to allow for globally invariant relative motion between agents as in the linear law (2), while retaining the rich behavior from nonlinearity for an arbitrary number of agents.

2.3. Input Convex Neural Networks

Fully connected Input Convex Neural Networks (Amos et al., 2017), is designed to be a scalar-valued function that is convex in its input, hence the name ICNN. The convexity is ensured by stacking the following layers:

$$\mathbf{z}_{k+1} = g_k(W_k^{(\mathbf{z})}\mathbf{z}_k + W_k^{(\mathbf{x})}\mathbf{x} + \mathbf{b}_k) \tag{3}$$

where $\mathbf{z}_0 = 0$ and $W_0^{(\mathbf{z})} = 0$. $W_k^{(\mathbf{z})}$ are non-negative matricies that can be attained through $\exp(*)$ or ReLU. The nonlinear activation functions g_k are convex and non-decreasing, such as ReLU. The remaining parameters $W_k^{(\mathbf{x})}$ and $\mathbf{b}_k$ have no constraints. It is also common to augment ICNN with a quadratic term to make it strongly convex in its input, as shown in (Khader et al., 2021a).

3. Methods

This section presents our proposed methodology. The methodology consists of two main steps, namely a translation-invariant distributed consensus law under a flow-based transformation, and a Centralized Training with Decentralized Execution (CTDE) policy learning framework that ensures such a transformation generates dexterous trajectories. Note the consensus law is agnostic about how the parameters are optimized. Other multi-agent reinforcement learning approaches can also be used. Section 3.1 provides the distributed control law that robots can apply, along with a proof of correctness, while Section 3.2 provides the details of the policy-learning algorithm.

3.1. Main Results

Our main result is a distributed control law, yielding a control force for robot agent with index i:

$$\tau_i = \mathbf{g}_i(\mathbf{x}_i) - \mathbf{D}_i\dot{\mathbf{x}}_i - \sum_{j\in\mathcal{N}_i}[\nabla P^\top(\sum_{k\in\mathcal{N}_i}(\mathbf{x}_i - \mathbf{x}_k)) - \nabla P^\top(\sum_{l\in\mathcal{N}_j}(\mathbf{x}_j - \mathbf{x}_l))] \tag{4}$$

with $D_i \in \mathbb{R}^{d\times d}$ as positive definite matrices, and P as a convex potential function. Our proof leverages diffeomorphism generated from the gradient of the convex function and the property of the graph Laplacian. Formally, we have the following theoretical results:

Theorem 1 *Let a system of n robotic agents evolving according to (1) under a static and undirected communication graph $\mathcal{G}$, with $\mathbf{L}$ as the corresponding Laplacian. Further, let a strictly convex and continuously differentiable function $P : \mathbb{R}^d \to \mathbb{R}$. If the graph $\mathcal{G}$ is connected, the distributed control design (4) guarantees that (i) $\lim_{t\to\infty}\big(\mathbf{x}_i(t) - \mathbf{x}_j(t)\big) = 0, \forall i, j \in \mathcal{N}$, if $\tau_{\text{ext}} = 0$, and (ii) the closed-loop multi-agent system is passive, if $\tau_{\text{ext}} \neq 0$.*

Remark 1 *Although we consider the gravity terms $g_i(x_i)$ known, the proposed control scheme can be extended to accommodate parametric uncertainty via standard adaptive-control schemes Verginis et al. (2019a).*

Before we present the proof of Theorem 1, we introduce some technical lemmas.

Lemma 1 *Let a continuously differentiable map $P : \mathbb{R}^d \to \mathbb{R}$ that is strictly convex. The gradient $\nabla P : \mathbb{R}^d \to \mathbb{R}^d$ is bijective.*

Proof Surjective is obvious given the codomain is defined as the image of ∇P. Assuming there exist $\mathbf{x} \neq \mathbf{y} \in \mathbb{R}^d$ satisfying $\nabla P(\mathbf{x}) = \nabla P(\mathbf{y})$. We have $\nabla P(\mathbf{x})(\mathbf{y}-\mathbf{x}) > P(\mathbf{y}) - P(\mathbf{x})$ and similarly $\nabla P(\mathbf{y})(\mathbf{x}-\mathbf{y}) > P(\mathbf{x}) - P(\mathbf{y})$ given P is strictly convex. The latter can be rewritten as $\nabla P(\mathbf{x})(\mathbf{y}-\mathbf{x}) < P(\mathbf{y}) - P(\mathbf{x})$ by multiplying both sides with -1 and noticing $\nabla P(\mathbf{x}) = \nabla P(\mathbf{y})$. This contradicts the former inequation. We reach $\nabla P(\mathbf{x}) = \nabla P(\mathbf{y}) \Rightarrow \mathbf{x} = \mathbf{y}$ and hence the conclusion. ■

Lemma 2 *It holds that $\mathbf{A}\mathbf{x} = 0 \iff \mathbf{A}^\top \mathbf{A}\mathbf{x} = 0$ for any $A \in \mathbb{R}^{d\times d}$ and $\mathbf{x} \in \mathbb{R}^d$.*

Proof From left to right is obvious. For the other direction, we have $\mathbf{x}^\top \mathbf{A}^\top \mathbf{A}\mathbf{x} = 0 \to \|\mathbf{A}\mathbf{x}\| = 0$ and hence $\mathbf{A}\mathbf{x} = 0$ given $\|\cdot\|$ is a norm. ■

We are now ready to present the proof of Theorem 1.
Proof of Theorem 1. We first write the control design (4) in a stacked vector form

$$\tau = \mathbf{g}(\mathbf{x}) - \mathbf{D}\dot{\mathbf{x}} - \tilde{\mathbf{L}}\nabla\tilde{P}^\top(\tilde{\mathbf{L}}\mathbf{x}) \tag{5}$$

where $\tilde{\mathbf{L}} = \mathbf{L} \otimes I_d$ and $\nabla\tilde{P}(\tilde{\mathbf{L}}\mathbf{x}) = [\nabla P(\tilde{\mathbf{L}}_1\mathbf{x}), \nabla P(\tilde{\mathbf{L}}_2\mathbf{x}), ..., \nabla P(\tilde{\mathbf{L}}_n\mathbf{x})]$, with $\tilde{\mathbf{L}}_i$ indicating the i-th row block of $\tilde{\mathbf{L}}$. The gravity and damping terms are formed by concatenating each component, e.g. as a block diagonal matrix $\mathbf{D} = \text{diag}\{\mathbf{D}_1, \mathbf{D}_2, ..., \mathbf{D}_n\}$, Consider now the continuously differentiable function

$$V = \sum_{i=1}^{n} P(\tilde{\mathbf{L}}_i\mathbf{x}) + \frac{1}{2}\dot{\mathbf{x}}^\top \mathbf{M}(\mathbf{x})\dot{\mathbf{x}}$$

where V is not a Lyapunov function. We tackle first the case where $\tau_{\text{ext}} = 0$. By differentiating V and using (1) and $\tilde{\mathbf{L}}^\top = \tilde{\mathbf{L}}$ and the skew-symmetry of $\dot{\mathbf{M}} - 2\mathbf{C}$, we obtain

$$\dot{V} = \dot{\mathbf{x}}^\top \tilde{\mathbf{L}}\nabla\tilde{P}^\top(\tilde{\mathbf{L}}\mathbf{x}) + \dot{\mathbf{x}}^\top(\tau - \mathbf{g}(\mathbf{x}))$$

which, by substituting (5), yields $\dot{V} = -\dot{\mathbf{x}}^\top \mathbf{D}\dot{\mathbf{x}}$. Hence, $V(t)$ remains bounded, for all $t \geq 0$. Since P and $\mathbf{M}$ are continuous, we also conclude the boundedness of the solution $\mathbf{x}(t)$, $\dot{\mathbf{x}}(t)$, for all $t \geq 0$.

We can hence invoke LaSalle's invariance principle (Khalil, 2002, Chapter 4) to state that $\mathbf{x}(t)$ converges to the largest invariant set S in $E = \{(\mathbf{x}, \dot{\mathbf{x}}) \in \mathbb{R}^{2\times n\times d} : \dot{V} = 0\}$; E consists of all the points satisfying $\dot{\mathbf{x}} = 0$, due to the positive definiteness of $\mathbf{D}$. In view of the closed-loop system, consisting of (1) and (5), the largest invariant set in E is the set $S = \{(\mathbf{x}, \dot{\mathbf{x}}) \in \mathbb{R}^{2\times n\times d} : \dot{\mathbf{x}} =$

$0, \ddot{\mathbf{x}} = 0\}$. Therefore, by substituting (5) in (1), we conclude that S consists of all the points that satisfy $\tilde{\mathbf{L}}\nabla\tilde{P}^\top(\tilde{\mathbf{L}}\mathbf{x}) = 0$, which reads

$$\nabla P^\top(\tilde{\mathbf{L}}_1\mathbf{x}) = \nabla P^\top(\tilde{\mathbf{L}}_2\mathbf{x}) = ... = \nabla P^\top(\tilde{\mathbf{L}}_n\mathbf{x}) \tag{6}$$

Since ∇P is bijective (Lemma 1), this further implies

$$\tilde{\mathbf{L}}_1\mathbf{x} = \tilde{\mathbf{L}}_2\mathbf{x} = ... = \tilde{\mathbf{L}}_n\mathbf{x} \tag{7}$$

Let $\mathbf{y}_i = \tilde{\mathbf{L}}_i\mathbf{x}$ and $\mathbf{y} = [\mathbf{y}_1^\top, ..., \mathbf{y}_n^\top]^\top$. We can have $\tilde{\mathbf{L}}\mathbf{y} = 0$ from $\mathbf{y}_1 = \mathbf{y}_2 = ... = \mathbf{y}_n$, and hence $\tilde{\mathbf{L}}\tilde{\mathbf{L}}\mathbf{x} = \tilde{\mathbf{L}}\mathbf{y} = 0$. By invoking Lemma 2 and and the transpose symmetry of $\tilde{\mathbf{L}}$, we reach $\tilde{\mathbf{L}}\mathbf{x} = 0$ and eventually $\mathbf{x}_1 = \mathbf{x}_2 = ... = \mathbf{x}_n$, which proves part (i).

Next, we consider the case where $\tau_{\text{ext}} \neq 0$. By following similar steps, the derivative of V becomes $\dot{V} = \dot{\mathbf{x}}^\top\tau_{\text{ext}} - \dot{\mathbf{x}}^\top\mathbf{D}\dot{\mathbf{x}} \leq \dot{\mathbf{x}}^\top\tau_{\text{ext}}$, which implies that the multi-robot system is passive under τ_{ext}, proving part (ii). ■

The aforementioned control law is applicable for 2nd-order systems of the form (1); For velocity-controlled first-order systems of the form $\dot{\mathbf{x}}_i = \mathbf{u}_i$, one can obtain a similar consensus protocol via:

$$\mathbf{u}_i = \dot{\mathbf{x}}_i = -\nabla P^\top(\sum_{j\in\mathcal{N}_i}(\mathbf{x}_i - \mathbf{x}_j)) \tag{8}$$

The proof is similar to Theorem 1 with V consisting only of the first part of the summation.

3.2. Policy Learning with Invariant Consensus Stabilization

We use (4) to construct policies in a learning loop, e.g. by augmenting additive noise for exploration in reinforcement learning. The control is unconditionally parameterized for consensus stabilization, so the policies are a drop-in replacement in the implementation of reinforcement learning. We propose using ICNN (Amos et al., 2017) as in (3) to instantiate the convex potential P. Note that ICNN can parameterize a strongly convex function, which implies the strict convexity required in (1). The ReLU non-linearity can break the function differentiability at finite points, while it is still differentiable almost everywhere. We found that this instantiation and the PyTorch subgradient work well in practice. We stick to a centralized training for the ease of integration to popular reinforcement learning implementations. However, distributed execution is possible due to the structure of (4).

The equilibrium configuration $\mathbf{x}_1 = \mathbf{x}_2 = ... = \mathbf{x}_n$ can be generalized to a formation with each $\mathbf{x}_i$ stopping at $\mathbf{c}_i$ relative to the rendezvous location. Namely, we can change each policy observation with an offset $\mathbf{c}_i$ for $\hat{\mathbf{x}}_i = \mathbf{x}_i - \mathbf{c}_i$ such that $\mathbf{x}_1 - \mathbf{c}_1 = \mathbf{x}_2 - \mathbf{c}_2 = ... = \mathbf{x}_n - \mathbf{c}_n$. We exploit these to design cooperative policies that are used in Section 4.

4. Experimental Results

In this section, we show the empirical benefits of learning policies with invariant consensus stabilization in multi-robot coordination tasks. Specifically, we explore answers to two research questions: 1) *can policies with an invariant consensus structure learn the tasks with improved data-efficiency?* 2) *can policies with an invariant consensus structure generalize better under untrained task configurations?* We design multi-robot reinforcement learning tasks and validate the performance of the proposed policy and baselines. The experiment setup, results on learning efficiency and generalization are presented below.

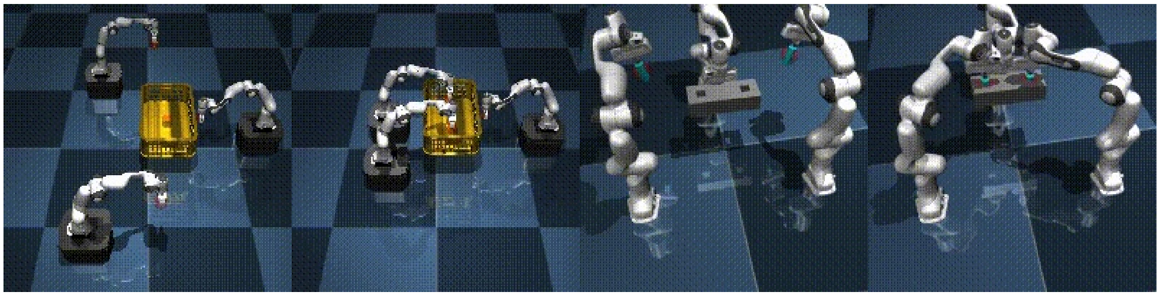

Figure 2: MuJoCo simulation environments for reinforcement learning of multi-robot coordination tasks: 1) Placing - mobile arms with one holding a basket and two others placing held objects to designated positions relative to the basket; 2) Insertion - one robot holding a block with two sockets for two other robots to plug pegs into desginated sockets.

4.1. Reinforcement Learning Environments and Baselines

We use the MuJoCo dynamic simulator (Todorov et al., 2012) to create two Gymnasium environments (Towers et al., 2024) as illustrated in Figure 2:

- **RidgebackFrankaPlacingEnv** - three Franka arms are mounted on Ridgeback mobile bases, with two robots placing grasped objects to a rectangular-shaped container held by the third robot, while avoiding collision between them. This is a velocity-based control task to investigate policies built upon the first-order consensus rule (8).

- **FrankaInsertionEnv** - three Franka arms with fixed bases and coordinate inserting two pegs to the sockets, with a clearance of 1mm. This is a force-based control task to highlight the necessity of the 2nd-order consensus rule (5) in contact-rich manipulation.

In both tasks, $\mathbf{x}_i$ represent the 3D positions of end-effectors. The robots are actuated by additional low-level controllers based on inverse kinematics and operational impedance control, respectively, to keep the initial orientation and motion compliance in FrankaInsertionEnv. The robots holding the container and the socket are indexed by 0. The rewards are simply defined as:

$$R(\mathbf{x}, \mathbf{u}) = \exp(-\|\mathbf{x}_1 - \mathbf{c}_1 - \mathbf{x}_0\| - \|\mathbf{x}_2 - \mathbf{c}_2 - \mathbf{x}_0\|) - \alpha\|\mathbf{u}\| \tag{9}$$

where $\mathbf{u} = \dot{\mathbf{x}}$ for the velocity control task and $\mathbf{u} = \tau$ for the force control task. We use α as the coefficient of penalization of the control magnitude and found that this was necessary for baseline policies to learn smooth motions. $\mathbf{c}_1$ and $\mathbf{c}_2$ are the offsets between the designated goal of $\mathbf{x}_{1/2}$ and $\mathbf{x}_0$, i.e. the location of sockets relative to robot-0 in FrankaInsertionEnv.

We consider two baselines to compare against the proposed policy with diffeomorphic transformations generated from ICNN (5):

- **Standard FCNN** - policy parameterized as fully-connected neural networks.

- **Consensus Normflow** - policy parameterized by Normalizing Flow networks (Rezende and Mohamed, 2015) for consensus stabilization as in (Yin et al., 2022).

All networks have two hidden layers with ReLU non-linearity. Policies are trained with PPO implemented in Stable-baseline3 (Raffin et al., 2021), with identical value function networks[2].

4.2. Data Efficiency

We train each task with all competing methods under the same PPO parameters and 1M environment steps, with the initial exploration standard deviation tuned for the output range of each policy network. The learning curves are depicted as in Figure 3. The results reveal that Consensus ICNN consistently outperforms baselines. Consensus Normflow policies show a closer performance, especially in the RidgebackFrankaPlacingEnv task, thanks to the inductive bias from consensus dynamics. The gap between Consensus ICNN and Consensus Normflow is more visible in FrankaInsertionEnv, possibly due to the task difficulty resembled by much narrower passages for navigating towards the rewarding states. The task difficulty is also manifested by larger variance of returns due to failing random seeds. Overall, we conclude that the proposed controller allows for faster reinforcement learning compared to neural networks with a standard architecture and a non-invariant consensus inductive bias. We hence get an affirmative answer to the first research question.

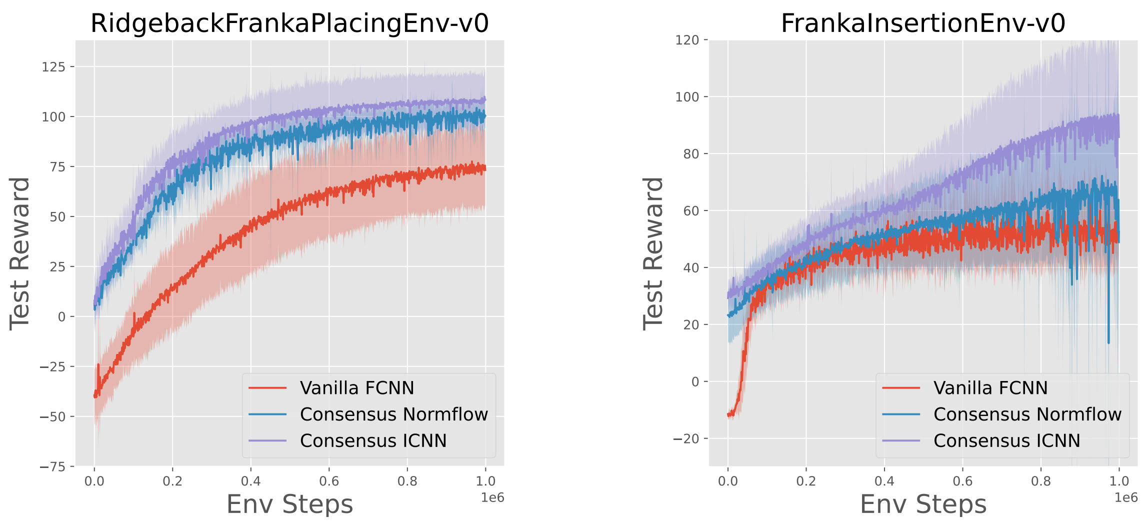

Figure 3: Return curves of reinforcement learning on multi-robot coordination tasks. The statistics are obtained from 10 random seeds, with the shades representing the standard deviation.

4.3. Policy Generalization

We further evaluate whether the learned policies can be generalized to untrained configurations. To that end, before deploying the trained policies, we displace the robot team by a certain distance (significantly larger than the dimension of the region where the robots spawn). The policies are run for 10 trials, and the success rates of task completion are recorded for comparison as in Figure 4.

Consensus ICNN policies show a clear advantage compared to baselines. FCNN can achieve a moderate success rate at trained locations (Displacement 0m) in the simpler placing task while the

2. https://github.com/navigator8972/consensus_invariant

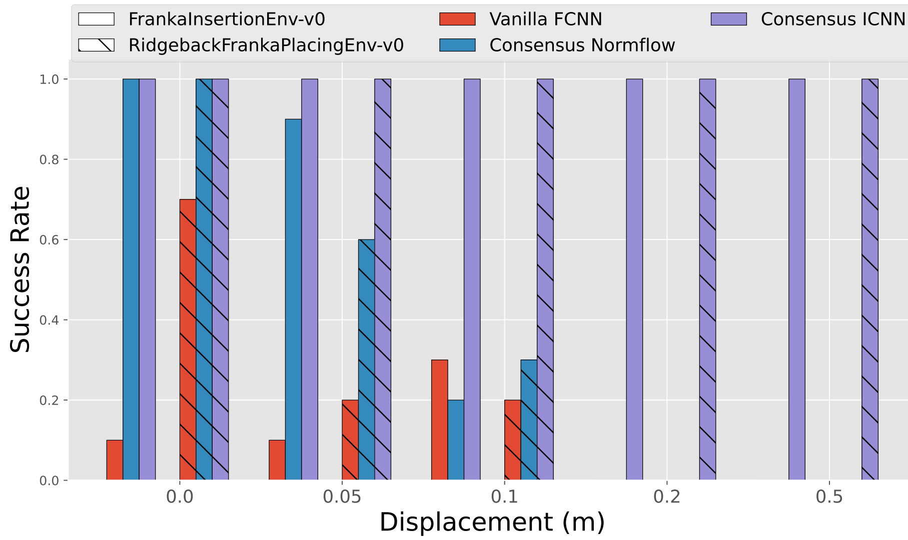

Figure 4: Success rates for robots that are displaced from their initial positions. FCNN and Consensus Normflow completely fail at the last two distances so the bars are not visible.

performance quickly deteriorates as the testing locations are displaced. The Consensus Normflow policies appear slightly more robust, which may again be attributed to the consensus inductive bias. At distant perturbed locations (0.2m, 0.5m), however, neither of the baselines can succeed once, while the Consensus ICNN policy maintains a perfect success rate. This addresses the second research question. Consequently, the results highlight the benefit of translation-invariant consensus stabilization. They also imply that the proposed policy structure would be practical for agents who can only measure their pose relative to neighbors, since only $\mathbf{x}_i - \mathbf{x}_j$ is needed for policy inference. This is necessary for multi-robot systems free of a centralized positioning mechanism.

5. Related Works

Multi-robot consensus works typically design distributed algorithms to achieve synchronization among the robot states (Olfati-Saber et al., 2007), potentially driven by a leader reference (Lin et al., 2024), or formation to a predefined geometric shape (Ahn, 2020; Verginis et al., 2019a). Several variations account for event-triggered computation (Li et al., 2020) or uncertain dynamics (Li et al., 2018; Verginis, 2024). Recent works combine learning with consensus algorithms in order to accelerate convergence (Kishida et al., 2020), account for data privacy (Carrascosa et al., 2022), or optimal coordination problems (Liu et al., 2021). The aforementioned works, however, cannot accommodate dexterous trajectories required by complex tasks while establishing robot safety. Our previous work (Yin et al., 2022) achieved such properties without, however, generalizing to arbitrary initial positions of the robots - a property that is guaranteed by the proposed algorithm.

Diffeomorphism is widely explored in geometric control, e.g. for applications in observer design (Mahony et al., 2022). In robot control, the properties of diffeomorphic transformations are leveraged to preserve stability guarantees as manifested in the early work (Neumann and Steil, 2015), where the transformation was handcrafted. Learning diffeomorphism to model kinematic trajectories was then investigated in imitation learning (Rana et al., 2020). The authors of (Khader et al., 2021b) showed the feasibility of applying diffeomorphism to full robot dynamics and learning forceful control policies. The application of diffeomorphism was also extended to the Lie group for orientation trajectories, with transformations applied in tangential spaces excepting singularity at a polar point (Urain et al., 2022). All these works concern the convergence towards the invariant set of a single point. Recent work (Stölzle et al., 2025) reported the success of generating stable periodic movements, exploiting the diffeomorphism of 2D canonical systems with a limit cycle. Going beyond a single-robot system, (Yin et al., 2022) demonstrated the usage of diffeomorphism in dual-arm coopperative manipulation. However, the control laws are either not preserving the coordination under translation or limited to special Normalizing-Flow networks that are not applicable to more than two robots. The presented work is closer to this piece while differs by allowing for translation invariance and the applicability to an arbitrary number of robot agents. Our results show a significant improvement on generalization performance due to the induced invariance.

Embedding symmetry in neural learning has been actively investigated to efficiently handle data under transformations. Numerous machine learning works focus on graph neural networks (Cohen and Welling, 2016), $E(2)$ convolutions for image data (Weiler and Cesa, 2019) and $E(n)$ graph convolutions for N-body systems and molecular structures (Satorras et al., 2021). Our control law resembles a similar equivariant graph neural network structure. The proposed method focuses on translation symmetry rather than more general Euclidean transformations, but offers stability guarantees that are not inherent in the works reviewed above. Robot learning benefits from learning equivariant observation or action spaces with improved generalization. Successes have been reported in point-cloud-based recognition (Lin et al., 2023) and visuo-manipulation (Gao et al., 2025). In object rearrangement, the spatial relations between scene keypoints were captured by local $SE(3)$ transformations to construct a neural descriptor field that can generalize novel configurations (Simeonov et al., 2023). The authors of (Wang et al., 2022) proposed to learn value-action networks with an inductive bias of $SE(2)$. This improved the efficiency of Q-learning for top-down grasping. Relevant ideas have also been explored in navigation and offline reinforcement learning (Reichlin et al., 2023). The above robotics works tend to enforce equivariance through auxiliary loss terms or optimization. The presented work is distinct with an inherent invariant structure. The policy is hence with provable guarantees which are lacked in peer works.

6. Conclusion

In this paper, we present consensus protocols that can be parameterized with structured neural networks. The networks exploit the diffeomorphism of the gradient of convex potentials and enable consensus stabilization of multi-robot systems with an arbitrary number of robots. In particular, the proposed structure admits learning policies that are invariant to location translation. System passivity is ensured throughout the coordination and hence provides safe guarantees. Our experiments demonstrate its clear superiority in reinforcement learning with less data and stronger generalization performance. Future works outlook improvements with more practical considerations. For instance, the forceful policy (5) implies the necessity of accessing the neighbors' information of an ego-robot's neighbors. This can be implemented by preserving and communicating the information as internal states, while the impact of latency needs to be analyzed and addressed. We also envision applying the idea to other forms of consensus protocols beyond stabilizing relative poses.

Acknowledgments

The authors would like to show their gratitude to the European Research Council, Swedish Research Council, and Knut and Alice Wallenberg Foundation. Hang Yin acknowledges the support from Innovation Foundation Denmark under the Grand Solutions project Fast and Efficient Robotic Automation via Reuse of Data (FERA). Christos K. Verginis further acknowledges the support of the Centre for Interdisciplinary Mathematics (CIM) at Uppsala University and Sweden's Innovation Agency within the research and innovation program Advanced Digitalization via the Competence Center SEDDIT-Sensor Informatics and Decision making for the Digital Transformation.

References

Hyo-Sung Ahn. *Formation control*. Springer, 2020.

Brandon Amos, Lei Xu, and J. Zico Kolter. Input convex neural networks. In Doina Precup and Yee Whye Teh, editors, *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pages 146–155. PMLR, 06–11 Aug 2017. URL https://proceedings.mlr.press/v70/amos17b.html.

Carlos Carrascosa, J Rincón, and Miguel Rebollo. Co-learning: consensus-based learning for multi-agent systems. *International Conference on Practical Applications of Agents and Multi-Agent Systems*, pages 63–75, 2022.

Taco Cohen and Max Welling. Group equivariant convolutional networks. In Maria Florina Balcan and Kilian Q. Weinberger, editors, *Proceedings of the 33rd International Conference on Machine Learning (ICML)*, volume 48 of *Proceedings of Machine Learning Research*, pages 2990–2999, New York, New York, USA, 20–22 Jun 2016. PMLR. URL https://proceedings.mlr.press/v48/cohenc16.html.

Chongkai Gao, Zhengrong Xue, Shuying Deng, Tianhai Liang, Siqi Yang, Lin Shao, and Huazhe Xu. Riemann: Near real-time SE(3)-equivariant robot manipulation without point cloud segmentation. In Pulkit Agrawal, Oliver Kroemer, and Wolfram Burgard, editors, *Proceedings of The 8th Conference on Robot Learning (CoRL)*, volume 270 of *Proceedings of Machine Learning Research*, pages 2164–2182. PMLR, 06–09 Nov 2025. URL https://proceedings.mlr.press/v270/gao25a.html.

Ruihua Han, Shengduo Chen, Shuaijun Wang, Zeqing Zhang, Rui Gao, Qi Hao, and Jia Pan. Reinforcement learned distributed multi-robot navigation with reciprocal velocity obstacle shaped rewards. *IEEE Robotics and Automation Letters*, 7(3):5896–5903, 2022.

Auke Jan Ijspeert, Jun Nakanishi, Heiko Hoffmann, Peter Pastor, and Stefan Schaal. Dynamical movement primitives: Learning attractor models for motor behaviors. *Neural Computation*, 25 (2):328–373, 2013. doi: 10.1162/NECO_a_00393.

Shahbaz Abdul Khader, Hang Yin, Pietro Falco, and Danica Kragic. Stability-guaranteed reinforcement learning for contact-rich manipulation. *IEEE Robotics and Automation Letters*, 6(1):1–8, 2020.

Shahbaz Abdul Khader, Hang Yin, Pietro Falco, and Danica Kragic. Learning deep energy shaping policies for stability-guaranteed manipulation. *IEEE Robotics and Automation Letters*, 6(4), 2021a. doi: 10.1109/LRA.2021.3111962.

Shahbaz Abdul Khader, Hang Yin, Pietro Falco, and Danica Kragic. Learning stable normalizing-flow control for robotic manipulation. In *2021 IEEE International Conference on Robotics and Automation (ICRA)*, pages 1644–1650, 2021b. doi: 10.1109/ICRA48506.2021.9562071.

Hassan K Khalil. *Nonlinear systems; 3rd ed.* Prentice-Hall, Upper Saddle River, NJ, 2002.

S. Mohammad Khansari-Zadeh and Aude Billard. Learning stable nonlinear dynamical systems with gaussian mixture models. *IEEE Transactions on Robotics*, 27(5):943–957, 2011. doi: 10.1109/TRO.2011.2159412.

Masako Kishida, Masaki Ogura, Yuichi Yoshida, and Tadashi Wadayama. Deep learning-based average consensus. *IEEE Access*, 8:142404–142412, 2020.

Steven M LaValle. *Planning algorithms*. Cambridge university press, 2006.

Xianwei Li, Yeng Chai Soh, and Lihua Xie. Robust consensus of uncertain linear multi-agent systems via dynamic output feedback. *Automatica*, 98:114–123, 2018.

Xianwei Li, Zhiyong Sun, Yang Tang, and Hamid Reza Karimi. Adaptive event-triggered consensus of multiagent systems on directed graphs. *IEEE Transactions on Automatic Control*, 66(4):1670–1685, 2020.

Chien Erh Lin, Jingwei Song, Ray Zhang, Minghan Zhu, and Maani Ghaffari. Se(3)-equivariant point cloud-based place recognition. In Karen Liu, Dana Kulic, and Jeff Ichnowski, editors, *Proceedings of The 6th Conference on Robot Learning (CoRL)*, volume 205 of *Proceedings of Machine Learning Research*, pages 1520–1530. PMLR, 14–18 Dec 2023. URL https://proceedings.mlr.press/v205/lin23a.html.

Lin Lin, Jinde Cao, James Lam, Shiyong Zhu, Shun-ichi Azuma, and Leszek Rutkowski. Leader–follower consensus over finite fields. *IEEE Transactions on Automatic Control*, 69(7):4718–4725, 2024.

Chen Liu, Lei Liu, Jinde Cao, and Mahmoud Abdel-Aty. Intermittent event-triggered optimal leader-following consensus for nonlinear multi-agent systems via actor-critic algorithm. *IEEE Transactions on Neural Networks and Learning Systems*, 34(8):3992–4006, 2021.

Robert Mahony, Pieter van Goor, and Tarek Hamel. Observer design for nonlinear systems with equivariance. *Annual Review of Control, Robotics, and Autonomous Systems*, 5(Volume 5, 2022):221–252, 2022. ISSN 2573-5144. doi: https://doi.org/10.1146/annurev-control-061520-010324. URL https://www.annualreviews.org/content/journals/10.1146/annurev-control-061520-010324.

Klaus Neumann and Jochen J. Steil. Learning robot motions with stable dynamical systems under diffeomorphic transformations. *Robotics and Autonomous Systems*, 70:1–15, 2015. ISSN 0921-8890. doi: https://doi.org/10.1016/j.robot.2015.04.006.

Reza Olfati-Saber, J Alex Fax, and Richard M Murray. Consensus and cooperation in networked multi-agent systems. *Proceedings of the IEEE*, 95(1):215–233, 2007.

Dimitra Panagou. A distributed feedback motion planning protocol for multiple unicycle agents of different classes. *IEEE Transactions on Automatic Control*, 62(3):1178–1193, 2016.

Antonin Raffin, Ashley Hill, Adam Gleave, Anssi Kanervisto, Maximilian Ernestus, and Noah Dormann. Stable-baselines3: Reliable reinforcement learning implementations. *Journal of Machine Learning Research*, 22(268):1–8, 2021. URL http://jmlr.org/papers/v22/20-1364.html.

Muhammad Asif Rana, Anqi Li, Dieter Fox, Byron Boots, Fabio Ramos, and Nathan Ratliff. Euclideanizing flows: Diffeomorphic reduction for learning stable dynamical systems. In *Proc. Conference on Learning for Dynamics and Control (L4DC)*, 2020.

Alfredo Reichlin, Giovanni Luca Marchetti, Hang Yin, Anastasiia Varava, and Danica Kragic. Learning geometric representations of objects via interaction. In Danai Koutra, Claudia Plant, Manuel Gomez Rodriguez, Elena Baralis, and Francesco Bonchi, editors, *Machine Learning and Knowledge Discovery in Databases: Research Track*, pages 629–644, Cham, 2023. Springer Nature Switzerland. ISBN 978-3-031-43421-1.

Danilo Rezende and Shakir Mohamed. Variational inference with normalizing flows. In *Proceedings of Machine Learning Research*, volume 37, pages 1530–1538, Lille, France, 07–09 Jul 2015. PMLR.

Víctor Garcia Satorras, Emiel Hoogeboom, and Max Welling. E(n) equivariant graph neural networks. In Marina Meila and Tong Zhang, editors, *Proceedings of the 38th International Conference on Machine Learning (ICML)*, volume 139 of *Proceedings of Machine Learning Research*, pages 9323–9332. PMLR, 18–24 Jul 2021. URL https://proceedings.mlr.press/v139/satorras21a.html.

Anthony Simeonov, Yilun Du, Yen-Chen Lin, Alberto Rodriguez Garcia, Leslie Pack Kaelbling, Tomás Lozano-Pérez, and Pulkit Agrawal. Se(3)-equivariant relational rearrangement with neural descriptor fields. In Karen Liu, Dana Kulic, and Jeff Ichnowski, editors, *Proceedings of The 6th Conference on Robot Learning (CoRL)*, volume 205 of *Proceedings of Machine Learning Research*, pages 835–846. PMLR, 14–18 Dec 2023. URL https://proceedings.mlr.press/v205/simeonov23a.html.

Maximilian Stölzle, T. Konstantin Rusch, Zach J. Patterson, Rodrigo Pérez-Dattari, Francesco Stella, Josie Hughes, Cosimo Della Santina, and Daniela Rus. Learning to move in rhythm: Task-conditioned motion policies with orbital stability guarantees, 2025. URL https://arxiv.org/abs/2507.10602.

Emanuel Todorov, Tom Erez, and Yuval Tassa. Mujoco: A physics engine for model-based control. In *IEEE/RSJ Int. Conf. Intelligent Robots and Systems*, pages 5026–5033. IEEE, 2012.

Mark Towers, Ariel Kwiatkowski, Jordan Terry, John U. Balis, Gianluca De Cola, Tristan Deleu, Manuel Goulão, Andreas Kallinteris, Markus Krimmel, Arjun KG, Rodrigo Perez-Vicente, Andrea Pierré, Sander Schulhoff, Jun Jet Tai, Hannah Tan, and Omar G. Younis. Gymnasium: A standard interface for reinforcement learning environments, 2024. URL https://arxiv.org/abs/2407.17032.

Julen Urain, Davide Tateo, and Jan Peters. Learning stable vector fields on lie groups. *IEEE Robotics and Automation Letters*, 7(4):12569–12576, 2022. doi: 10.1109/LRA.2022.3219019.

Christos K Verginis. *Planning and Control of Uncertain Cooperative Mobile Manipulator-Endowed Systems under Temporal Logic Tasks*. PhD thesis, KTH Royal Institute of Technology, 2020.

Christos K Verginis. Asymptotic consensus of multi-agent systems with unknown nonlinear dynamics via smooth barrier integral control. *IEEE Conference on Decision and Control (CDC)*, pages 51–56, 2024.

Christos K Verginis, Alexandros Nikou, and Dimos V Dimarogonas. Robust formation control in se (3) for tree-graph structures with prescribed transient and steady state performance. *Automatica*, 103:538–548, 2019a.

Christos K Verginis, Daniel Zelazo, and Dimos V Dimarogonas. Cooperative manipulation via internal force regulation: A rigidity theory perspective. *arXiv preprint arXiv:1911.01297*, 2019b.

Christos K Verginis, Dimos V Dimarogonas, and Lydia E Kavraki. Kdf: Kinodynamic motion planning via geometric sampling-based algorithms and funnel control. *IEEE Transactions on robotics*, 39(2):978–997, 2022.

Dian Wang, Robin Walters, Xupeng Zhu, and Robert Platt. Equivariant q learning in spatial action spaces. In Aleksandra Faust, David Hsu, and Gerhard Neumann, editors, *Proceedings of the 5th Conference on Robot Learning*, volume 164 of *Proceedings of Machine Learning Research*, pages 1713–1723. PMLR, 08–11 Nov 2022. URL https://proceedings.mlr.press/v164/wang22j.html.

Maurice Weiler and Gabriele Cesa. General E(2)-equivariant steerable cnns. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2019. URL https://papers.neurips.cc/paper/2019/file/8112-General-E2-Equivariant-Steerable-CNNs.pdf.

Hang Yin, Christos K. Verginis, and Danica Kragic. Consensus-based normalizing-flow control: A case study in learning dual-arm coordination. In *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 10417–10424, 2022. doi: 10.1109/IROS47612.2022.9981827.