Proceedings of Machine Learning Research vol 283:1–14, 2025 7th Annual Conference on Learning for Dynamics and Control

Morphological-Symmetry-Equivariant Heterogeneous Graph Neural Network for Robotic Dynamics Learning

Fengze Xie[*1] FXXIE@CALTECH.EDU
Sizhe Wei[*2] SWEI@GATECH.EDU
Yue Song[1] YUESONG@CALTECH.EDU
Yisong Yue[1] YYUE@CALTECH.EDU
Lu Gan[2] LGAN@GATECH.EDU
∗ *Indicates equal contribution*
[1] *California Institute of Technology*
[2] *Georgia Institute of Technology*

Editors: N. Ozay, L. Balzano, D. Panagou, A. Abate

Abstract

We propose MS-HGNN, a Morphological-Symmetry-Equivariant Heterogeneous Graph Neural Network for robotic dynamics learning, which integrates robotic kinematic structures and morphological symmetries into a unified graph network. By embedding these structural priors as inductive biases, MS-HGNN ensures high generalizability, sample and model efficiency. This architecture is versatile and broadly applicable to various multi-body dynamic systems and dynamics learning tasks. We prove the morphological-symmetry-equivariant property of MS-HGNN and demonstrate its effectiveness across multiple quadruped robot dynamics learning problems using real-world and simulated data. Our code is available at https://github.com/lunarlab-gatech/MorphSym-HGNN/.[†]

Keywords: Morphological symmetry, geometric deep learning, graph neural network, robotic dynamics learning, quadruped robots

1. Introduction

A rigid body system is a collection of interconnected components that do not deform under external forces. Existing approaches to planning and controlling for rigid body systems fall into two categories: safe but inflexible methods and adaptive yet risky methods. Traditional methods provide safety and stability by relying on well-understood dynamics models (Murray et al., 1994; Spong et al., 2005), but they struggle in complex, unpredictable environments where modeling becomes difficult. Conversely, machine learning-based approaches offer greater adaptability by learning dynamic interactions and planning strategies across diverse environments (Miki et al., 2022; Li et al., 2021) but suffer from unseen and highly dynamic environments.

To bridge traditional and learning-based methods, it is essential to incorporate morphological information from the robot's structure into our learning architecture. By embedding this structural information, the model can implicitly account for the robot's physical configuration, enhancing interpretability and data efficiency. The morphology of a rigid body system has two key components: the kinematic structure and morphological symmetry. A kinematic structure (Mruthyunjaya, 2003; Taheri and Mozayani, 2023) consists of interconnected links connected by joints that allow

[†] Project website: https://lunarlab-gatech.github.io/MorphSym-HGNN. The appendix can be found on the project website.

© 2025 F. Xie, S. Wei, Y. Song, Y. Yue & L. Gan.

relative motion, such as rotation or translation. Each joint imposes specific movement constraints, enabling the system to perform complex actions through the combination of simpler joint motions. In robotics, kinematic structures are fundamental for modeling and controlling articulated structures like robotic arms, quadrupeds (Camurri et al., 2017; Kang et al., 2023; Yang et al., 2023), and humanoids. Integrating kinematic structure into the learning model can help establish the relative relationships between each component, aligning the model closely with the robot's physical design. On the other hand, morphological symmetries are structural symmetries in a robot's body that allow it to mimic certain spatial transformations, such as rotations, reflections, or translations (Smith et al., 2024; Ordoñez-Apraez et al., 2025). Integrating these geometric priors into the learning model can improve data efficiency, enabling the model to generalize better across configurations and tasks.

In this work, we propose MS-HGNN, a morphological-symmetry-equivariant heterogeneous graph neural network that integrates both the robotic kinematic structures and morphological symmetries into the learning process. We further validate the symmetry properties of the neural network through theoretical analysis, and demonstrate how these embedded features enhance the model's interpretability and efficiency in extensive robotic dynamics learning experiments.

2. Related Work

Rigid Body Systems. In robotics, rigid body systems are essential for representing complex articulated structures like robotic arms, quadrupeds, and humanoids. Traditional rigid body modeling relies on established mathematical frameworks to describe motion and calculate the forces and torques necessary for desired movements (Spong et al., 2005). On the other hand, data-driven techniques, such as neural networks and reinforcement learning (Li et al., 2021; Miki et al., 2022), have been introduced to model and control rigid body systems, bringing adaptability and flexibility to these systems in diverse or unstructured settings. Recently, several approaches have emerged that bridge classic and data-driven methods, leveraging the strengths of both (O'Connell et al., 2022; Lutter and Peters, 2023; Neary and Topcu, 2023; Xie et al., 2024; Lupu et al., 2025). These approaches typically embed physical laws as constraints or regulators within the learning model.

Geometric Deep Learning. Traditional deep learning methods are effective for grid-like data structures, such as images, but often struggle with irregular, non-Euclidean domains. Geometric deep learning overcomes this limitation by developing architectures that preserve the inherent geometric properties of the data (Wang et al., 2021; Du et al., 2022; Rivière et al., 2020; Ordoñez-Apraez et al., 2025). These properties can be captured through various representations (Bronstein et al., 2021), motivating the use of specialized architectures (Satorras et al., 2021; Cohen and Welling, 2016; Wang et al., 2021; Zaheer et al., 2017). In robotics, geometric deep learning offers a physics-informed approach that enhances model interpretability and improves sample efficiency.

Physics-Informed Learning for Robotics. Recent advancements in physics-informed learning have shown significant promise in enhancing learning performance by embedding physical laws and dynamics directly into learning models (Nghiem et al., 2023; Djeumou et al., 2022). Unlike traditional data-driven approaches, physics-informed methods leverage underlying principles (Raissi et al., 2019; Greydanus et al., 2019; Cranmer et al., 2020; Duong et al., 2024) to improve model interpretability, robustness, and data efficiency (Sanchez-Gonzalez et al., 2018; Kim et al., 2021; Butterfield et al., 2024). Our neural network architecture employs geometric deep learning to embed kinematic structure and symmetry information from rigid-body systems, forming a physics-informed model. Compared to prior work (Ordoñez-Apraez et al., 2025; Su et al., 2024; Mittal

et al., 2024), our approach not only achieves the same symmetry guarantees but also captures detailed structural information from the kinematic tree through a graph neural network.

3. Preliminaries

3.1. Morphology-Informed Heterogeneous Graph Neural Network

Heterogeneous Graph Neural Networks (HGNNs) (Shi, 2022), denoted as $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, are a type of Graph Neural Network (GNN) designed to handle graphs with multiple types of nodes $\mathcal{V}$ and edges $\mathcal{E}$, capturing complex relationships and rich semantic information. Unlike traditional GNNs, which assume a uniform graph structure, HGNNs apply specialized aggregation and transformation functions tailored to different node and edge types. This makes them particularly effective in applications such as recommendation systems, knowledge graphs, and robotics, where diverse interactions between entities must be accurately modeled. A Morphology-Informed Heterogeneous Graph Neural Network (MI-HGNN) (Butterfield et al., 2024) is an HGNN with node and edge types directly derived from the system's kinematic structure. Based on the functional roles of nodes within the kinematic structure, we assign them to distinct node classes $\mathcal{V} = \{\mathcal{V}_1, \dots, \mathcal{V}_n\}$, where each class $\mathcal{V}_i = \{v_i^1, \dots, v_i^m\}$ contains individual nodes v_i^j. Links in the kinematic structure are represented as edges in the graph, with the edge type $e(v_i, v_j) \in \mathcal{E}_{ij}$ depending on the node types at both ends, where $\mathcal{E}_{ij} \in \mathcal{E}$. For example, in a floating-base system, components like the base, joints, and feet can be represented by distinct types of nodes $\mathcal{V}_b$, $\mathcal{V}_t$, and $\mathcal{V}_f$, shown in Fig.1(c), while the links connecting these components are modeled as edges.

3.2. Morphological Symmetries in Rigid Body Systems

A rigid body system is a collection of solid bodies that maintain a fixed shape and size while moving under external forces and torques. When connected through joints allowing relative motion, these bodies form a kinematic tree that defines the structured movement of interconnected rigid bodies. Morphological symmetry arises from replicated kinematic chains and body parts with symmetric mass distributions. The morphological symmetry group $\mathbb{G}$ represents feasible state transformation including reflection and rotation that map a robot state $(\mathbf{q}, \dot{\mathbf{q}})$ to an equivalent reachable state $(g \,\hat{\triangleright}\, \mathbf{q}, g \,\hat{\triangleright}\, \dot{\mathbf{q}})$, where $\mathbf{q} \in \mathbb{R}^{n_q}$ is the generalized position coordinates with n_q as the number of states. We distinguish between standard group actions $(g \triangleright x)$, conjugate actions on linear maps $(g \diamond A)$, and morphological symmetry-induced state mappings $(g \,\hat{\triangleright}\, x)$, following the conventions in Ordoñez-Apraez et al. (2025). The formal definition of morphological symmetry action is given by:

$$(g \,\hat{\triangleright}\, \mathbf{q}, g \,\hat{\triangleright}\, \dot{\mathbf{q}}) := \left(\begin{bmatrix} \mathbf{X}_g \mathbf{X}_B \mathbf{X}_g^{-1} \\ \rho_{\mathcal{M}}(g) \mathbf{q}_{js} \end{bmatrix}, \begin{bmatrix} \mathbf{X}_g \dot{\mathbf{X}}_B \mathbf{X}_g^{-1} \\ \rho_{\mathcal{T}_{\mathbf{q}}\mathcal{M}}(g) \dot{\mathbf{q}}_{js} \end{bmatrix} \right). \tag{1}$$

Here, the base transformation $g \,\hat{\triangleright}\, \mathbf{X}_B = \mathbf{X}_g \mathbf{X}_B \mathbf{X}_g^{-1} \in \mathbb{SE}_d$, where $\mathbf{X}_B := \begin{bmatrix} \mathbf{R}_B & \mathbf{r}_B \\ \mathbf{0} & 1 \end{bmatrix} \in \mathbb{SE}_d$ represents the base configuration, with $\mathbf{R}_B \in \mathbb{SO}_d$ as the rotation matrix representing the base's orientation. The transformation matrix $\mathbf{X}_g := \begin{bmatrix} \mathbf{R}_g & \mathbf{r}_g \\ \mathbf{0} & 1 \end{bmatrix} \in \mathbb{E}_d$ includes $\mathbf{R}_g \in \mathbb{O}_d$, an orthogonal matrix representing reflection or rotation. The joint space transformation is defined as $g \,\hat{\triangleright}\, \mathbf{q}_{js} := \rho_{\mathcal{M}}(g) \mathbf{q}_{js}$, where $\rho_{\mathcal{M}}(g)$ is the representation of the group element g. Once the base is identified, the robot's unique kinematic branches are represented by $\mathbb{S} = \{\mathbb{S}_1, \dots, \mathbb{S}_k\}$. Each branch $\mathbb{S}_i$ has $n_{\text{dof}}(\mathbb{S}_i) \in \mathbb{N}$

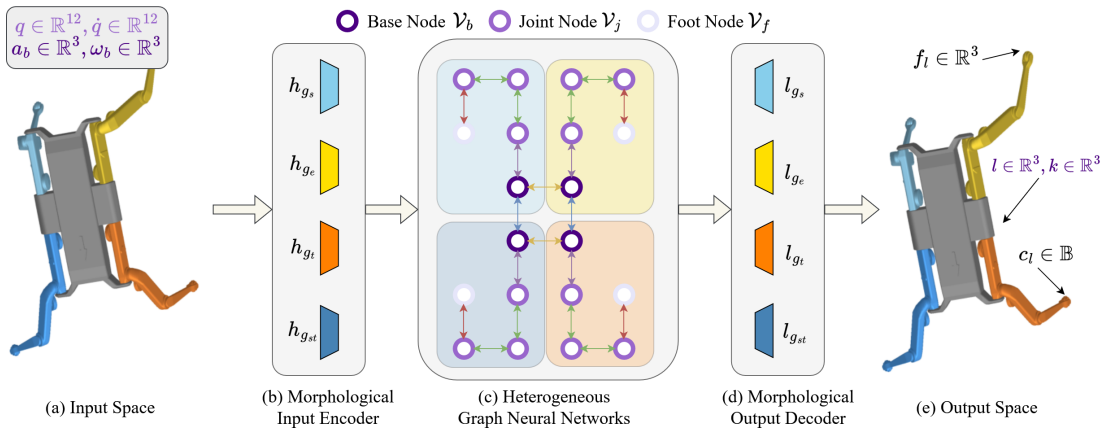

Figure 1: Overview of the MS-HGNN framework for robots with symmetry type $\mathbb{G} := \mathbb{K}_4$. (a) The input space consists of the robot's current state observations, which are mapped to corresponding nodes in the HGNN. (b) and (d) The morphological symmetry encoder-decoder pair ensures that the learned representations respect the robot's morphology. (c) The HGNN is automatically constructed to preserve geometric symmetry. (e) The output space consists of dynamics-relevant variables, obtained from their corresponding nodes in the HGNN.

degrees of freedom and is replicated $n_{\text{rep}}(\mathbb{S}_i) \in \mathbb{N}$ times throughout the kinematic structure. The instance labels for branch $\mathbb{S}_i$ are $\mathbb{S}_i = \{\mathbb{S}_{i,1}, \ldots, \mathbb{S}_{i,n_{\text{rep}}(\mathbb{S}_i)}\}$. The details of morphological symmetry in kinematic structures are provided in the Appendix. This morphological symmetry can be naturally modeled using a GNN, where the kinematic structure is encoded by an adjacency matrix and components like the base, joints, and feet are distinguished by node types. The symmetry is embedded in the GNN architecture itself, enabling weight sharing across repeated structures and improving generalization from proprioceptive sensor inputs.

4. Methodology

This work leverages an HGNN to model the morphological symmetry and kinematic structure of rigid body systems. Our approach consists of two key components as shown in Fig. 1: (1) an automatically constructed HGNN that preserves the system's inherent geometric symmetry, and (2) an encoder-decoder module that transforms geometric symmetry into morphological symmetry, ensuring consistency with the system's dynamic properties. In the following sections, we provide a step-by-step framework for constructing an MS-HGNN, guided by the system's kinematic structure and morphological symmetry principles.

1. Determine the morphological symmetry group $\mathbb{G}_m < \mathbb{G}_{\mathbb{E}}$ and the unique kinematic branches $\mathbb{S}$ of the system, where $\mathbb{G}_{\mathbb{E}}$ is the generalized euclidean group.
2. Create subgraphs for all kinematic branches as $\mathcal{G}_i = \{\mathcal{G}_{i,1}(\mathbb{S}_{i,1}), \ldots, \mathcal{G}_{i,n_{\text{rep}}(\mathbb{S})}(\mathbb{S}_{i,n_{\text{rep}}(\mathbb{S}_i)})\}$, where $\mathcal{G}_{i,j_1} \cong \mathcal{G}_{i,j_2}, \forall j_1, j_2 \in \mathbb{N} \leq n_{\text{rep}}(\mathbb{S}_i)$.
3. Label each subgraph $\mathcal{G}_{i,j}$ as $\mathcal{G}_{p,q}$, where $p \leq |\mathbb{G}_m|$ corresponds to an element in group $\mathbb{G}_m$, and subgraphs with same q lies in the same orbit.

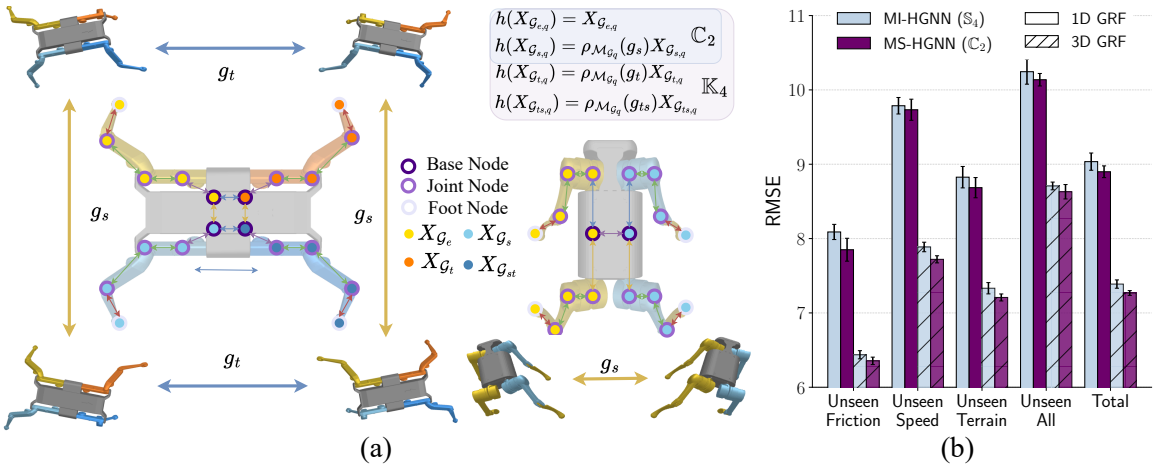

Figure 2: (a) The visualization of the MS-HGNN architecture is shown for the morphological symmetry groups $\mathbb{G} := \mathbb{K}_4$ (left, Solo) and $\mathbb{G} := \mathbb{C}_2$ (right, A1). Inputs and outputs of the MS-HGNN are distributed over graph nodes mapped to corresponding robot components (base, joint, foot). Variables representing the entire robot are placed on base nodes. Different node types represent distinct components of the robot's kinematic structure, with contour colors indicating node types and filling colors denoting the group elements. The encoder and decoder types depend on the group elements, while edges between nodes are determined by node types and the robot's symmetry. (b) Ground reaction force estimation test RMSE on simulated A1 dataset (Butterfield et al., 2024).

4. For any subgraph class $\{\mathcal{G}_q\}$, including the base node $\{\mathcal{V}_b\}$ that lacks the full set of $|\mathbb{G}_m|$ graphs, complete each group orbit by replicating elements along missing transformations and label them as $\mathcal{G}_{p,q}$.
5. Connect $\{\mathcal{V}_{b,p}\}$ with Cayley Graph (Cayley, 1878). Connect each subgraph $\mathcal{G}_{p,q}$ to $\mathcal{V}_{b,p}$ with edge type $\mathcal{E}_q$, formalizing a graph $\mathcal{G}$.
6. Add input encoders and output decoders for each node based on the subgraph class p it belongs to, ensuring morphological symmetry equivariance $\mathbb{G}_m$ in our GNN.

Next, we provide a mathematical proof demonstrating that our constructed graph is equivariant under morphological symmetry transformations. Details of the proof are provided in the appendix. After completing the first five steps of our construction process, we obtain a graph $\mathcal{G}$ that preserves the system's inherent geometric symmetry and is composed of subgraphs $\{\mathcal{G}_1, \ldots, \mathcal{G}_q\}$. Each subgraph $\mathcal{G}_i$ is further subdivided into instances $\{\mathcal{G}_{i,1}, \ldots, \mathcal{G}_{i,p}\}$, where $p \in \mathbb{N}$ denotes the number of instances and $n_{\text{ftr}}(\mathcal{G}_i) \in \mathbb{N}$ represents the number of node features per instance. The parameter q corresponds to the types of kinematic chains (e.g., legs, arms), while p identifies the type of element within a group. To ensure that the learned representations respect the morphological symmetry group $\mathbb{G}$, we integrate an additional encoder-decoder pair, as shown in Fig. 1(b), (d). This enables the HGNN to capture structural equivalences, preserving the morphological symmetry of the robot within the overall learning framework.

We define two types of group actions: the Euclidean reflection and rotation group action, denoted as $g_m \triangleright (\cdot)$, and the morphological reflection and rotation group action, denoted as $g_m \,\hat{\triangleright}\, (\cdot)$. For each subgraph instance $\mathcal{G}_{p,q}$, the Euclidean group action on our graph satisfies the property

$g_m \triangleright \mathcal{G}_{p,q} = \mathcal{G}_{g_m(p),q}$, where g_m is an element of the morphological transformation group $\mathbb{G}_m$. We further define $\rho_{\mathcal{G}_q}(g_m) \in \mathbb{R}^{p\times p}$ as the permutation matrix associated with the group action g_m.

Consequently, the group action on a stack of subgraph instances can be expressed as:

$$\rho_{\mathcal{G}_q}(g_m)\begin{bmatrix}\mathcal{G}_{p_1,q} & \mathcal{G}_{p_2,q} & \cdots\end{bmatrix}^T = \begin{bmatrix}\mathcal{G}_{g_m(p_1),q} & \mathcal{G}_{g_m(p_2),q} & \cdots\end{bmatrix}^T. \tag{2}$$

We denote node space representation as an identity matrix as $\rho_{b\mathcal{M}_{\mathcal{G}_q}}(g_m) := I_{n_{\text{ftr}}(\mathbb{G}_m)}$. The graph space permutation matrix $g_m \triangleright X_{\mathcal{G}} = \rho_b X_{\mathcal{G}}$ is defined as

$$\rho_b := \text{diag}\left(\rho_{b\mathcal{M}_{[\mathcal{G}_1]}}(g_m), \ldots, \rho_{b\mathcal{M}_{[\mathcal{G}_k]}}(g_m)\right), \quad \text{with} \quad \rho_{b\mathcal{M}_{[\mathcal{G}_i]}}(g_m) := \rho_{\mathcal{G}_i}(g_m) \otimes \rho_{b\mathcal{M}_{\mathcal{G}_i}}(g_m). \tag{3}$$

Theorem 1 (Permutation Automorphism) *Assume our $\mathcal{G}$ with adjacency matrix $A_{\mathcal{G}}$ and node features $X_{\mathcal{G}}$, where different types of edges and nodes are represented by different integers. The mapping $\phi_{\rho_b} : \mathcal{G} \to \mathcal{G}$ is an automorphism if the edge and node features are preserved as:*

$$\forall \rho_b \in \mathbb{G}_m, \quad \phi_{\rho_b}(A_{\mathcal{G}}) = \rho_b A_{\mathcal{G}} \rho_b^T = A_{\mathcal{G}} \quad \textit{and} \quad \phi_{\rho_b}(X_{\mathcal{G}}) = \rho_b X_{\mathcal{G}} = X_{\mathcal{G}} \tag{4}$$

With the above automorphism, the equivariance to Euclidean symmetry immediately follows:

Lemma 2 (Euclidean Group Equivariance) *If $\phi_{\rho_b} : \mathcal{G} \to \mathcal{G}$ is an automorphism of graph $\mathcal{G}$ and $z_{\mathcal{G}}$ is the representation of the GNN based on $\mathcal{G}$, the GNN is equivariant to Euclidean group actions (Hamilton, 2020):*

$$\forall g_m \in \mathbb{G}_m, \quad g_m \triangleright z_{\mathcal{G}}(X_{\mathcal{G}}) = z_{\mathcal{G}}\big(\phi_{\rho_b}(X_{\mathcal{G}})\big) = z_{\mathcal{G}}(\rho_b X_{\mathcal{G}}) = z_{\mathcal{G}}(g_m \triangleright X_{\mathcal{G}}). \tag{5}$$

However, we would like our neural network to achieve equivariance on morphological reflection and rotation transformation groups, which requires $\forall g_m \in \mathbb{G}_m,\ g_m \mathbin{\overset{\circ}{\triangleright}} z_{\mathcal{G}}(X_{\mathcal{G}}) = z_{\mathcal{G}}(g_m \mathbin{\overset{\circ}{\triangleright}} X_{\mathcal{G}})$, rather than Euclidean reflection and rotation group actions.

Theorem 3 (Morphological-Symmetry-Equivariant HGNN) *With the input encoder h and the output decoder l that satisfies the following condition:*

$$\forall g_{m,p} \in \mathbb{G}_m, \quad h(X_{\mathcal{G}_{p,q}}) = \rho_{\mathcal{M}_{\mathcal{G}_q}}(g_{m,p}) X_{\mathcal{G}_{p,q}} \quad \textit{and} \quad l(X_{\mathcal{G}_{p,q}}) = \rho_{\mathcal{M}_{\mathcal{G}_q}}(g_{m,p})^{-1} X_{\mathcal{G}_{p,q}}, \tag{6}$$

where $\rho_{\mathcal{M}_{\mathcal{G}_q}}$ denotes the transformation of the coordinate frames attached to each joint belonging to the subgraph class $\mathcal{G}_q$. h and l transform Euclidean and Morphological symmetries as follows:

$$\forall g_m \in \mathbb{G}_m, \quad g_m \mathbin{\overset{\circ}{\triangleright}} l(x) = l(g_m \triangleright x) \quad \textit{and} \quad g_m \triangleright h(x) = h(g_m \mathbin{\overset{\circ}{\triangleright}} x) \tag{7}$$

Our GNN is equivariant to morphological group actions:

$$\forall g_m \in \mathbb{G}_m, \quad g_m \mathbin{\overset{\circ}{\triangleright}} f_{\mathcal{G}}(X_{\mathcal{G}}) = f_{\mathcal{G}}(g_m \mathbin{\overset{\circ}{\triangleright}} X_{\mathcal{G}}). \tag{8}$$

where $f_{\mathcal{G}}$ denotes the graph representation $f_{\mathcal{G}}(X_{\mathcal{G}}) = l(z_{\mathcal{G}}(h(X_{\mathcal{G}})))$.

Our proposed MS-HGNN architecture is designed to be equivariant to morphological symmetry and is generalizable to various robotic systems and different kinds of tasks. To demonstrate its effectiveness, we specifically implement the architecture for the Mini-Cheetah and Solo robots, which exhibit the $\mathbb{K}_4$ symmetry group, and the A1 robot, which exhibits the $\mathbb{C}_2$ symmetry group. These cases were chosen due to the availability of experimental data and their suitability for visualization, as illustrated in Fig. 2.

It is important to note that since both $\mathbb{K}_4$ and $\mathbb{C}_2$ have elements that are involutions, the encoder and decoder operations are structurally identical. However, this equivalence does not hold for higher-order cyclic symmetry groups such as $\mathbb{C}_n$ with $n > 2$, which are commonly found in other symmetric rigid-body robotic systems, such as multi-arm robots.

5. Experiments

We evaluate MS-HGNN as a generalizable model across various rigid body tasks, with a focusing on quadruped robots. Our experiments empirically demonstrate that the specialized graph structure of MS-HGNN effectively captures morphological information, validated on multiple tasks: contact state detection using real-world data (classification), Ground Reaction Force (GRF) estimation and centroidal momentum estimation using simulated data (regression) from various quadruped platforms. These components are critical for modeling quadruped dynamics and enabling effective control. Given the generalized velocity, acceleration, and torques of the system as $\dot{\mathbf{q}}$, $\ddot{\mathbf{q}}$, and τ, respectively, the dynamics of quadrupeds are governed by:

$$\mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} + g(\mathbf{q}) = \mathbf{S}^T\tau + \mathbf{J}_{\text{ext}}(\mathbf{q})^T\mathbf{f}_{\text{ext}}, \tag{9}$$

where $\mathbf{M}(\mathbf{q})$ is the inertia matrix, $\mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})$ is the Coriolis matrix, $g(\mathbf{q})$ is the gravity vector, $\mathbf{S}^T$ is the selection matrix, $\mathbf{f}_{\text{ext}}$ represents external forces, and $\mathbf{J}_{\text{ext}}(\mathbf{q})$ is the external force contact Jacobian. In quadrupeds, external forces are dominated by GRFs, leading to the approximation $\mathbf{J}_{\text{ext}}(\mathbf{q})^T\mathbf{f}_{\text{ext}} \approx \sum_{l=1}^{4} \mathbf{J}_l(\mathbf{q})^T\mathbf{f}_l$, where $\mathbf{f}_l$ and $\mathbf{J}_l(\mathbf{q})$ are the GRF and its Jacobian for leg l. Accurate GRF estimation and contact state detection are vital for capturing the system's dynamics and are foundational for downstream control and planning An and Lee (2023); Arena et al. (2022). In addition, centroidal momenta-comprising the linear and angular momentum of the robot's center of mass (CoM) relative to an inertial frame-summarize whole-body motion (de Lasa et al., 2010) and are particularly useful involving external disturbances. Accurately estimating centroidal momenta enables adaptive control under dynamic conditions, such as strong wind or human interactions. We compare our method against CNN (Lin et al., 2021), state-of-the-art $\mathbb{G}$-equivariant models CNN-Aug and ECNN with $\mathbb{C}_2$ symmetry (Apraez et al., 2023), and the morphology-aware model, MI-HGNN (Butterfield et al., 2024). To comprehensively evaluate model performance and generalizability, we select three datasets spanning different tasks and platforms: contact states on Mini-Cheetah (Lin et al., 2021), GRF estimation on A1 (Butterfield et al., 2024), and centroid momenta estimation on Solo (Ordoñez-Apraez et al., 2025). This setup allows us to assess MS-HGNN across diverse robotic systems and dynamic learning challenges.

5.1. Contact State Detection for Mini-Cheetah Robot (Classification)

This task involves predicting the 4-leg contact state of a quadruped robot from its proprioceptive sensor data. We adapt the real-world dataset from Lin et al. (2021), collected on a Mini-Cheetah robot (Katz et al., 2019) using various gaits across diverse terrains (e.g., sidewalk, asphalt, concrete, pebbles, forest, grass, etc.). The dataset consists of *measured* joint angles $\mathbf{q} \in \mathbb{R}^{12}$, joint angular velocities $\dot{\mathbf{q}} \in \mathbb{R}^{12}$, base linear acceleration $\mathbf{a}_b \in \mathbb{R}^3$, base angular velocity $\boldsymbol{\omega}_b \in \mathbb{R}^3$ from an Inertial Measurement Unit (IMU), and *estimated* foot positions $\mathbf{p}_l \in \mathbb{R}^3$ and velocities $\mathbf{v}_l \in \mathbb{R}^3$ via forward kinematics, where $l = \{LF, LH, RF, RH\}$ is the index of each leg. The *ground-truth* binary contact state $\mathbf{c}_l \in \mathbb{B}$, $\mathbb{B} = \{0, 1\}$ is obtained offline using a non-causal algorithm (Lin et al., 2021). The dataset has around 1 million samples synchronized at 1000 Hz. Following the data split from Apraez et al. (2023), we use the same unseen sequences for testing and allocate 85% and 15% of the remaining data for training and validation. As test sequences contain unseen gait/terrain combinations, this setup is also helpful for evaluating model's generalization ability on out-of-distribution data.

Given that Mini-Cheetah exhibits $\mathbb{G} = \mathbb{K}_4$ symmetry, we evaluate models using both $\mathbb{K}_4$ group and its subgroup $\mathbb{C}_2$. The input comprises 150-step histories of $[\mathbf{q}, \dot{\mathbf{q}}, \mathbf{a}_b, \omega_b, \mathbf{p}, \mathbf{v}] \in \mathbb{R}^{54}$ up to a time

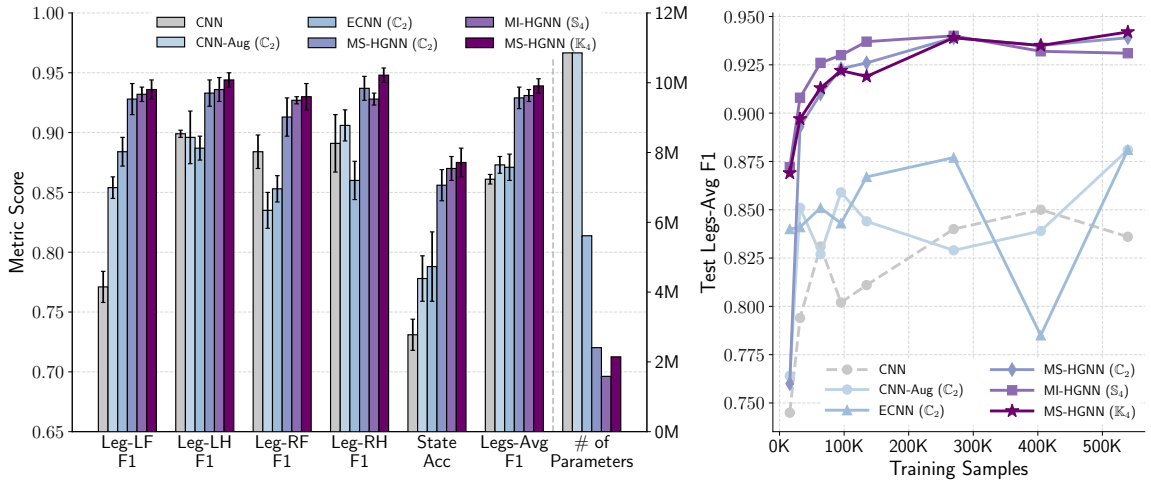

Figure 3: Contact state detection results on the real-world Mini-Cheetah dataset (Lin et al., 2021). Left: F1 scores per leg, averaged F1 score, and 16-state accuracy (4-run average). Parameter counts are shown for each method. Right: Averaged F1 scores versus training set size. MS-HGNN ($\mathbb{C}_2$ & $\mathbb{K}_4$) achieve around 0.9 averaged F1-score with just 5% of training data.

step t, and the prediction is a 16-state contact state for all legs $\hat{\mathbf{c}} \in \mathbb{B}^4$ at t. For graph-based models (MI-HGNN and our MS-HGNN), the input data is grouped into a graph structure and assigned to the corresponding node, i.e., base $(\mathbf{a}_b, \boldsymbol{\omega}_b)$, joint $(q_j, \dot{q}_j)$ being j the joint index, and foot $(\mathbf{p}_l, \mathbf{v}_l)$ measurements are fed into the corresponding base $(\mathcal{V}_b)$, joint $(\mathcal{V}_t)$ and foot $(\mathcal{V}_f)$ nodes, respectively, according to their indices. The foot-wise contact state prediction is output from the corresponding foot node $(\mathcal{V}_f)$. For our MS-HGNN-$\mathbb{C}_2$ and MS-HGNN-$\mathbb{K}_4$ models, all base nodes receive identical inputs. We employ 8 message-passing layers with a hidden size of 128 and train the network for up to 49 epochs using a learning rate of 10^{-4}. We evaluate all models using the metrics form Apraez et al. (2023); Butterfield et al. (2024), including foot-wise binary F1-score, averaged F1-score across legs, and 16-state contact state accuracy (correct only if all four legs are classified correctly).

The classification results (mean $\pm$ std across 4 runs) and parameter sizes are summarized in Fig. 3-left. Compared to non-graph-based models (CNN, CNN-Aug, ECNN), graph-based networks achieve substantially better performance with significantly fewer parameters. Specifically, MS-HGNN-$\mathbb{K}_4$ improved 11% contact state accuracy over ECNN, the best-performing non-graph-based model, using only 38% of ECNN's parameters. This demonstrated the effectiveness and efficiency of using morphological structure via graph representations. The morphology-informed graph network enforces information flow consistent with the robot's kinematic structure, embedding physical priors into message-passing. This architecture utilizes physical knowledge as a *prior* and increases the model's *causality*, and also reduces the parameter count needed to capture complex dynamics of the robot. Among graph-based models, our proposed MS-HGNN-$\mathbb{K}_4$ outperforms MI-HGNN in both averaged F1-score (0.939 vs. 0.931) and accuracy (0.875 vs. 0.870), showing the benefit of preserving morphological-symmetry. In contrast, MI-HGNN follows geometric symmetry $\mathbb{S}_4$, permutation-equivariant for any legs, which over-constrains the model and leads to suboptimal results. Among $\mathbb{C}_2$ models (excluding MI-HGNN), MS-HGNN-$\mathbb{C}_2$ achieves the best performance. Furthermore, the performance gap between $\mathbb{K}_4$ and $\mathbb{C}_2$ variants of MS-HGNN empha-

sizes the benefits of exploiting the morphological symmetries in robotic systems, which is offering equivalent to doubling the effective dataset size through $\mathbb{K}_4$ compared to $\mathbb{C}_2$.

Trainable parameters in $\mathbb{G}$-equivariant networks. $\mathbb{G}$-equivariant network reduces the trainable parameters by $1/|\mathbb{G}|$ ($|\mathbb{G}|$ is the group order) compared to unconstrained neural network of the same architectural. Thus, ECNN-$\mathbb{C}_2$ or EMLP-$\mathbb{C}_2$ (Apraez et al., 2023) have approximately twice as many parameters as ECNN-$\mathbb{K}_4$ or EMLP-$\mathbb{K}_4$. Interestingly, our MS-HGNN achieves $\mathbb{G}$-equivariance via structured graphs and minimal edge connections instead of $\mathbb{G}$-equivariant layers, yielding comparable parameters for MS-HGNN-$\mathbb{C}_2$ and MS-HGNN-$\mathbb{K}_4$ (see Fig. 3-left).

Sample efficiency of MS-HGNN. We assess sample efficiency by varying training set size and reporting the test averaged F1-scores (Fig. 3-right). Like MI-HGNN, both MS-HGNN variants ($\mathbb{C}_2$ and $\mathbb{K}_4$) achieve $\sim$0.9 averaged F1-scores with only 5% of the training data. This demonstrates the utility of morphological priors in real-world settings, where data is limited and expensive to collect.

5.2. Ground Reaction Force Estimation for A1 Robot (Regression)

Estimating ground reaction forces is essential for accurate legged robot dynamics modeling and control. Our graph-based network is well-suited to this task, naturally integrating multi-modal sensory inputs from local frames via message passing. We evaluate on the simulated GRF dataset from Butterfield et al. (2024), collected using the QUAD-SDK simulator (Norby et al., 2022) on an A1 quadruped robot exhibiting $\mathbb{G} = \mathbb{C}_2$ symmetry. The data is synchronized at 500 Hz and includes joint states ($\mathbf{q} \in \mathbb{R}^{12}$, $\dot{\mathbf{q}} \in \mathbb{R}^{12}$, $\boldsymbol{\tau} \in \mathbb{R}^{12}$), base linear acceleration ($\boldsymbol{a}_b \in \mathbb{R}^3$), base angular velocity ($\boldsymbol{\omega}_b \in \mathbb{R}^3$), and ground-truth GRFs ($\boldsymbol{f}_l \in \mathbb{R}^3$, l is the leg index). Following Butterfield et al. (2024), we use a 150-step history of input to predict both Z-axis (1D) and full 3D GRFs. The hyperparameters and training settings for MS-HGNN and MI-HGNN remain consistent with the previous task. Fig 2(b) presents test the Root Mean Square Error (RMSE) over 4 runs on test sequences with unseen terrain frictions, robot speeds, and terrain types. MS-HGNN-$\mathbb{C}_2$ consistently achieves lower RMSE than MI-HGNN, with an average improvement of 1.62% in 3D and 1.50% in 1D GRF prediction. The relatively modest gain in 3D is attributed to the low GRF magnitudes in the X/Y directions in this dataset. Theses results highlight the benefit of preserving morphological symmetry in MS-HGNN over the heuristic design of MI-HGNN for force estimation tasks.

5.3. Centroidal Momentum Estimation for Solo Robot (Regression)

In this experiment, we estimate the robot's linear $\boldsymbol{l} \in \mathbb{R}^3$ and angular momentum $\boldsymbol{k} \in \mathbb{R}^3$ from its joint-space position and velocities $\mathbf{q} \in \mathbb{R}^{12}, \dot{\mathbf{q}} \in \mathbb{R}^{12}$. The simulated dataset is generated by Ordoñez-Apraez et al. (2025) using PINOCCHIO (Carpentier et al., 2019) for the Solo quadruped robot exhibiting $\mathbb{G} = \mathbb{K}_4$ symmetry. Different from previous contact estimation problems, this task introduces a new challenge of predicting *angular* momentum from multiple base nodes. To adapt MS-HGNN to this task, we construct models under $\mathbb{C}_2$ and $\mathbb{K}_4$ by attaching the morphology encoder to all joint nodes and morphology decoder to base nodes, and use the MSE losses across base nodes for training. We compare our models with the unconstrained MLP, MLP-Aug, EMLP (Apraez et al., 2023), and MI-HGNN ($\mathbb{C}_2$ and $\mathbb{K}_4$). Quantitative results (mean and standard deviation over 4 runs) evaluated by cosine similarity and MSE are given in Fig. 4-left. MI-HGNN is excluded from the figure due to its incomparable performance (linear cosine similarity 0.9301 ± 0.0017, angular cosine similarity 0.5173 ± 0.0016, test MSE 0.3421 ± 0.0009). Our model outperform all baselines by a significant margin. The degraded performance of MI-HGNN stems from its use of $\mathbb{S}_4$ symmetry,

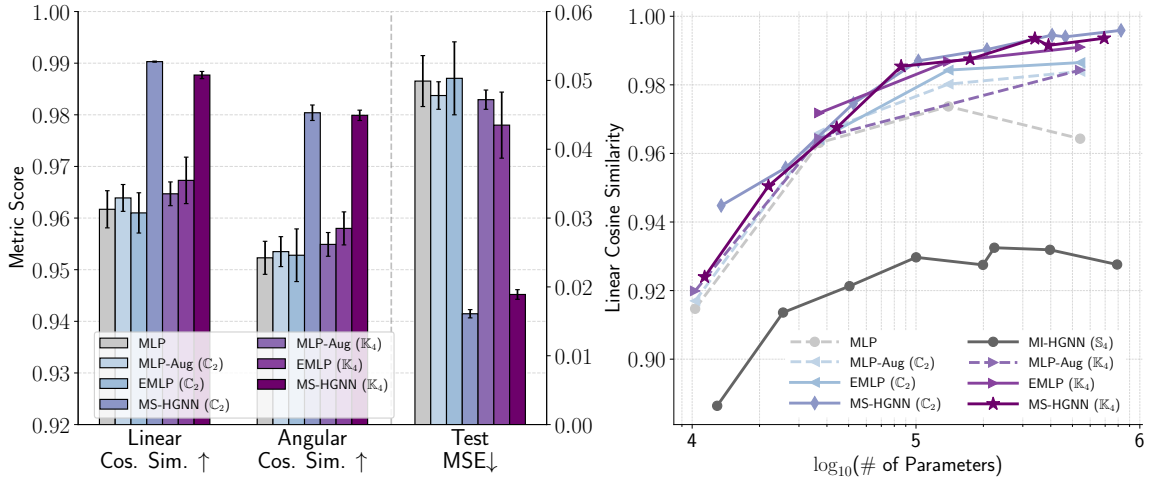

Figure 4: Centroidal momentum estimation results on the synthetic Solo dataset (Apraez et al., 2023). Left: Test linear/angular cosine similarity and MSE of predictions, averaged over 4 runs. Right: Linear cosine similarity for models of varying sizes. Our MS-HGNN ($\mathbb{C}_2$ and $\mathbb{K}_4$) exhibit superior model efficiency without overfitting.

which misaligns with the robot's true morphological structure, making it ineffective in learning angular dynamics. In contrast, MS-HGNN embeds $\mathbb{C}_2$ and $\mathbb{K}_4$ symmetries, enabling effective learning of both linear and angular momentum. We further evaluate the **model efficiency** of MS-HGNN by varying the numbers of parameters and reporting test linear cosine similarity in Fig. 4-right. Notably, MS-HGNN-$\mathbb{C}_2$ achieves 0.9448 cosine similarity with only 13,478 parameters, showing high model efficiency. Furthermore, the performance of both MS-HGNN variants improves steadily as model size increases, whereas MI-HGNN and MLP tend to overfit when scaled up.

6. Conclusions

In this work, we introduce MS-HGNN, a general and versatile network architecture for robotic dynamics learning by integrating both robotic kinematic structures and morphological symmetries. Through rigorous theoretical proof and extensive empirical validation, we demonstrated the advantages of using a morphology-informed graph network structure and morphological-symmetry-equivariant property in robotic dynamics learning. Experiments show that MS-HGNN consistently achieves superior performance, generalizability, sample efficiency and model efficiency across a variety of tasks and platforms, making MS-HGNN particularly suitable for data-scarce robotic applications. Furthermore, the modularity of MS-HGNN allows easy adaptation to diverse robotic systems with varying morphological structures. Future work will extend the framework to embed temporary symmetries in robotic systems and deploy to real robot for more challenging tasks.

Acknowledgments

We would like to thank D. Butterfield, Z. Gan, and X. Wu for their insightful discussions, and L. Zhao for the assistance with the experiments.

References

Jaehoon An and Inho Lee. Artificial neural network-based ground reaction force estimation and learning for dynamic-legged robot systems. *PeerJ Computer Science*, 9:e1720, 2023.

Daniel Felipe Ordoñez Apraez, Mario Martín, Antonio Agudo, and Francesc Moreno. On discrete symmetries of robotics systems: A group-theoretic and data-driven analysis. In *Robotics: Science and Systems*, 2023.

Paolo Arena, Maria Francesca Pia Cusimano, Luca Patanè, and Poramate Manoonpong. Ground reaction force estimation in a quadruped robot via liquid state networks. In *International Joint Conference on Neural Networks*, pages 1–8, 2022.

Michael M. Bronstein, Joan Bruna, Taco Cohen, and Petar Velickovic. Geometric deep learning: Grids, groups, graphs, geodesics, and gauges. *arXiv preprint arXiv: 2104.13478*, 2021.

Daniel Butterfield, Sandilya Sai Garimella, Nai-Jen Cheng, and Lu Gan. MI-HGNN: Morphology-informed heterogeneous graph neural network for legged robot contact perception. *arXiv preprint arXiv: 2409.11146*, 2024.

Marco Camurri, Maurice F. Fallon, Stéphane Bazeille, Andreea Radulescu, Victor Barasuol, Darwin G. Caldwell, and Claudio Semini. Probabilistic contact estimation and impact detection for state estimation of quadruped robots. *IEEE Robotics and Automation Letters*, 2(2):1023–1030, 2017.

Justin Carpentier, Guilhem Saurel, Gabriele Buondonno, Joseph Mirabel, Florent Lamiraux, Olivier Stasse, and Nicolas Mansard. The pinocchio C++ library : A fast and flexible implementation of rigid body dynamics algorithms and their analytical derivatives. In *IEEE/SICE International Symposium on System Integration*, pages 614–619, 2019.

Arthur Cayley. Desiderata and suggestions: No. 2. the theory of groups: graphical representation. *American Journal of Mathematics*, 1(2):174–176, 1878.

Taco Cohen and Max Welling. Group equivariant convolutional networks. In *International Conference on Machine Learning*, pages 2990–2999, 2016.

Miles D. Cranmer, Sam Greydanus, Stephan Hoyer, Peter W. Battaglia, David N. Spergel, and Shirley Ho. Lagrangian neural networks. *arXiv preprint arXiv: 2003.04630*, 2020.

Martin de Lasa, Igor Mordatch, and Aaron Hertzmann. Feature-based locomotion controllers. *ACM Transactions on Graphics*, 29(4):131:1–131:10, 2010.

Franck Djeumou, Cyrus Neary, Eric Goubault, Sylvie Putot, and Ufuk Topcu. Neural networks with physics-informed architectures and constraints for dynamical systems modeling. In *Learning for Dynamics and Control Conference*, pages 263–277, 2022.

Weitao Du, He Zhang, Yuanqi Du, Qi Meng, Wei Chen, Nanning Zheng, Bin Shao, and Tie-Yan Liu. SE(3) equivariant graph neural networks with complete local frames. In *International Conference on Machine Learning*, pages 5583–5608, 2022.

Thai Duong, Abdullah Altawaitan, Jason Stanley, and Nikolay Atanasov. Port-hamiltonian neural ODE networks on lie groups for robot dynamics learning and control. *IEEE Transactions on Robotics*, 40:3695–3715, 2024.

Samuel Greydanus, Misko Dzamba, and Jason Yosinski. Hamiltonian neural networks. In *Advances in Neural Information Processing Systems*, pages 15353–15363, 2019.

William L. Hamilton. *Graph Representation Learning*. Morgan & Claypool Publishers, 2020.

Jeong-Uk Kang, Hyun-Bin Kim, Keun Ha Choi, and Kyung-Soo Kim. External force estimation of legged robots via a factor graph framework with a disturbance observer. In *IEEE International Conference on Robotics and Automation*, pages 12120–12126, 2023.

Benjamin Katz, Jared Di Carlo, and Sangbae Kim. Mini cheetah: A platform for pushing the limits of dynamic quadruped control. In *International Conference on Robotics and Automation*, pages 6295–6301, 2019.

Joanne Taery Kim, Jeongeun Park, Sungjoon Choi, and Sehoon Ha. Learning robot structure and motion embeddings using graph neural networks. *arXiv preprint arXiv: 2109.07543*, 2021.

Zhongyu Li, Xuxin Cheng, Xue Bin Peng, Pieter Abbeel, Sergey Levine, Glen Berseth, and Koushil Sreenath. Reinforcement learning for robust parameterized locomotion control of bipedal robots. In *IEEE International Conference on Robotics and Automation*, pages 2811–2817, 2021.

Tzu-Yuan Lin, Ray Zhang, Justin Yu, and Maani Ghaffari. Legged robot state estimation using invariant Kalman filtering and learned contact events. In *Conference on Robot Learning*, pages 1057–1066, 2021.

Elena Sorina Lupu, Fengze Xie, James A. Preiss, Jedidiah Alindogan, Matthew Anderson, and Soon-Jo Chung. Magic^{vfm}-meta-learning adaptation for ground interaction control with visual foundation models. *IEEE Transactions on Robotics*, 41:180–199, 2025.

Michael Lutter and Jan Peters. Combining physics and deep learning to learn continuous-time dynamics models. *The International Journal of Robotics Research*, 42(3):83–107, 2023.

Takahiro Miki, Joonho Lee, Jemin Hwangbo, Lorenz Wellhausen, Vladlen Koltun, and Marco Hutter. Learning robust perceptive locomotion for quadrupedal robots in the wild. *Science Robotics*, 7(62):eabk2822, 2022.

Mayank Mittal, Nikita Rudin, Victor Klemm, Arthur Allshire, and Marco Hutter. Symmetry considerations for learning task symmetric robot policies. In *IEEE International Conference on Robotics and Automation*, pages 7433–7439, 2024.

T.S Mruthyunjaya. Kinematic structure of mechanisms revisited. *Mechanism and Machine Theory*, 38(4):279–320, 2003.

Richard M. Murray, Zexiang Li, and S. Shankar Sastry. *A mathematical introduction to robotics manipulation*. CRC Press, 1994.

Cyrus Neary and Ufuk Topcu. Compositional learning of dynamical system models using port-hamiltonian neural networks. In *Learning for Dynamics and Control Conference*, pages 679–691, 2023.

Truong X. Nghiem, Ján Drgona, Colin N. Jones, Zoltán Nagy, Roland Schwan, Biswadip Dey, Ankush Chakrabarty, Stefano Di Cairano, Joel A. Paulson, Andrea Carron, Melanie N. Zeilinger, Wenceslao Shaw-Cortez, and Draguna L. Vrabie. Physics-informed machine learning for modeling and control of dynamical systems. In *American Control Conference*, pages 3735–3750, 2023.

Joseph Norby, Yanhao Yang, Ardalan Tajbakhsh, Jiming Ren, Justin K Yim, Alexandra Stutt, Qishun Yu, Nikolai Flowers, and Aaron M Johnson. Quad-SDK: Full stack software framework for agile quadrupedal locomotion. In *ICRA Workshop on Legged Robots*, 2022.

Michael O'Connell, Guanya Shi, Xichen Shi, Kamyar Azizzadenesheli, Anima Anandkumar, Yisong Yue, and Soon-Jo Chung. Neural-fly enables rapid learning for agile flight in strong winds. *Science Robotics*, 7(66), 2022.

Daniel Ordoñez-Apraez, Giulio Turrisi, Vladimir Kostic, Mario Martin, Antonio Agudo, Francesc Moreno-Noguer, Massimiliano Pontil, Claudio Semini, and Carlos Mastalli. Morphological symmetries in robotics. *The International Journal of Robotics Research*, 0(0), 2025.

Maziar Raissi, Paris Perdikaris, and George E. Karniadakis. Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations. *Journal of Computational Physics*, 378:686–707, 2019.

Benjamin Rivière, Wolfgang Hönig, Yisong Yue, and Soon-Jo Chung. GLAS: global-to-local safe autonomy synthesis for multi-robot motion planning with end-to-end learning. *IEEE Robotics and Automation Letters*, 5(3):4249–4256, 2020.

Alvaro Sanchez-Gonzalez, Nicolas Heess, Jost Tobias Springenberg, Josh Merel, Martin A. Riedmiller, Raia Hadsell, and Peter W. Battaglia. Graph networks as learnable physics engines for inference and control. In *International Conference on Machine Learning*, pages 4467–4476, 2018.

Victor Garcia Satorras, Emiel Hoogeboom, and Max Welling. E(n) equivariant graph neural networks. In *International Conference on Machine Learning*, pages 9323–9332, 2021.

Chuan Shi. Heterogeneous graph neural networks. In *Graph Neural Networks: Foundations, Frontiers, and Applications*. Springer Nature Singapore, 2022.

Henry Smith, Ajay Shankar, Jennifer Gielis, Jan Blumenkamp, and Amanda Prorok. So(2)-equivariant downwash models for close proximity flight. *IEEE Robotics and Automation Letters*, 9(2):1174–1181, 2024.

Mark W. Spong, Seth Hutchinson, and M. Vidyasagar. *Robot Modeling and Control*. Wiley, 2005.

Zhi Su, Xiaoyu Huang, Daniel Felipe Ordoñez Apraez, Yunfei Li, Zhongyu Li, Qiayuan Liao, Giulio Turrisi, Massimiliano Pontil, Claudio Semini, Yi Wu, and Koushil Sreenath. Leveraging symmetry in rl-based legged locomotion control. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 6899–6906, 2024.

Hamid Taheri and Nasser Mozayani. A study on quadruped mobile robots. *Mechanism and Machine Theory*, 190:105448, 2023.

Rui Wang, Robin Walters, and Rose Yu. Incorporating symmetry into deep dynamics models for improved generalization. In *International Conference on Learning Representations*, 2021.

Fengze Xie, Guanya Shi, Michael O'Connell, Yisong Yue, and Soon-Jo Chung. Hierarchical meta-learning-based adaptive controller. In *IEEE International Conference on Robotics and Automation*, pages 18309–10315, 2024.

Yuxiang Yang, Guanya Shi, Xiangyun Meng, Wenhao Yu, Tingnan Zhang, Jie Tan, and Byron Boots. Cajun: Continuous adaptive jumping using a learned centroidal controller. In *Conference on Robot Learning*, pages 2791–2806, 2023.

Manzil Zaheer, Satwik Kottur, Siamak Ravanbakhsh, Barnabás Póczos, Ruslan Salakhutdinov, and Alexander J. Smola. Deep sets. In *Advances in Neural Information Processing Systems*, pages 3391–3401, 2017.