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

Fengze Xie*[1], Sizhe Wei*[2], Yue Song[1], Yisong Yue[1], Lu Gan[2]

***Abstract*—We present a morphological-symmetry-equivariant heterogeneous graph neural network (MS-HGNN) for robotic dynamics learning. MS-HGNN unifies robotic kinematic structures and morphological symmetries within a single graph-based neural-symbolic architecture. By embedding these structural priors as symbolic constraints in the network design, MS-HGNN achieves strong generalization, high sample efficiency, and compact model complexity. This neural-symbolic integration enables the model to reason over the physical structure of multi-body dynamic systems while retaining the flexibility of data-driven learning. We formally prove the morphological-symmetry-equivariant property of MS-HGNN and empirically validate its effectiveness on a range of quadruped robot learning tasks using both real-world and simulated datasets. Code is publicly available at https://github.com/lunarlab-gatech/MorphSym-HGNN/.**

***Index Terms*—Neural-symbolic learning, Morphological symmetry, Graph neural network**

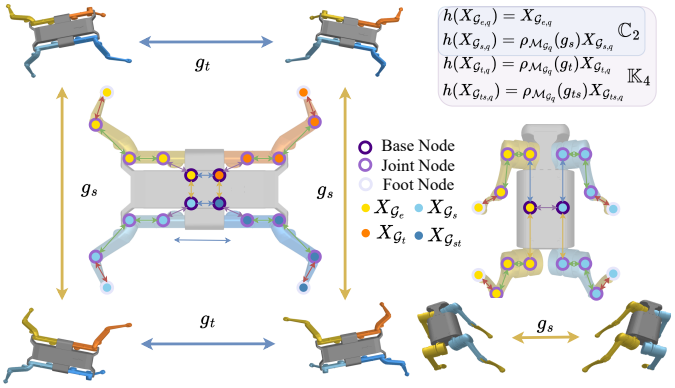

Fig. 1: Morphological symmetry groups in MS-HGNN $\mathbb{G} := \mathbb{K}_4$ (left, Solo robot) and $\mathbb{G} := \mathbb{C}_2$ (right, A1 robot).

I. Introduction

A rigid body system is a collection of interconnected components that do not deform under external forces. Existing approaches to controlling and planning 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 [1, 2], 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 [3, 4] 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. The learned model can implicitly account for the robot's physical configuration by embedding this structural information, enhancing interpretability and data efficiency. The morphology of a rigid body system has two key components: the kinematic chain structure and symmetry. A kinematic chain [5, 6] in a rigid body system consists of interconnected links joined by joints that allow relative motion, such as rotation or translation. Each joint imposes specific movement constraints, enabling the system to perform complex actions through combinations of simpler joint motions. In robotics, kinematic chains are crucial for modeling and controlling the movement of articulated structures like robotic arms, quadrupeds [7, 8, 9], and humanoids. Integrating kinematic chain information 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 [10, 11]. Integrating these geometric priors into the learning model enhances data efficiency by reducing the need for extensive training samples, improves the model's ability to generalize across a diverse range of robot configurations and task scenarios, and promotes greater interpretability through alignment with the underlying physical and structural properties of the system.

In this work, we propose MS-HGNN, a morphological-symmetry-equivariant heterogeneous graph neural network that integrates both kinematic structure and morphological symmetries into a unified learning framework. The key contribution of our work lies in formulating this architecture as a neural-symbolic system: the kinematic structure and symmetry priors are encoded as symbolic constraints within the graph network, while the learning process remains fully data-driven. This neural-symbolic integration allows MS-HGNN to reason over physical structure and relationships in a way that is both interpretable and generalizable. We formally prove the symmetry-equivariant properties of MS-HGNN and empirically validate its effectiveness on a wide range of quadruped robot learning tasks using both real-world and simulated data.

* Equal contribution, [1]California Institute of Technology, [2]Georgia Institute of Technology, lgan@gatech.edu.

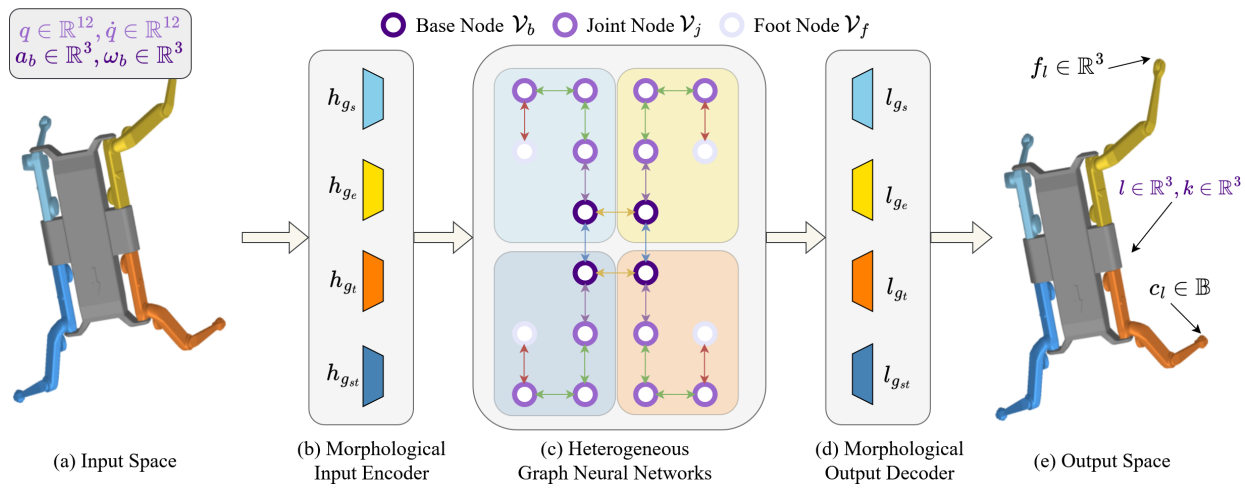

Fig. 2: Overview of the MS-HGNN framework for robots with symmetry type $\mathbb{G} := \mathbb{K}_4$. (a) The input space consists of the robot's current state observations, which are mapped to relevant nodes in the heterogeneous graph neural network (HGNN). (b) and (d) The morphological symmetry encoder-decoder pair ensures that the learned representations adhere to the robot's morphological structure. (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.

II. 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 [2]. On the other hand, data-driven techniques, such as neural networks and reinforcement learning [3, 4], 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 [12, 13, 14, 15, 16]. These approaches typically embed physical laws as constraints or regulators within the learning model.

Neural-Symbolic Learning for Robotics. Neural-symbolic learning combines structured domain knowledge with data-driven models, aiming to improve interpretability, sample efficiency, and generalization. In robotics, this involves embedding physical principles, such as kinematic constraints, dynamics models, and system symmetries, into neural network architectures [17, 18, 19]. Graph neural networks (GNNs) are particularly well-suited to this paradigm, as they naturally encode the relational structure of rigid-body systems. Equivariant GNNs further enhance this by incorporating symmetry priors [11], enabling efficient learning across morphologically similar components. Our proposed MS-HGNN builds on this neural-symbolic foundation by unifying kinematic structure and morphological symmetry within a single heterogeneous GNN architecture. Compared to prior work, our model enforces symmetry-equivariance while explicitly modeling the system's physical topology, leading to a scalable and interpretable framework for robotic dynamics learning.

III. Methodology

This work employs a heterogeneous graph neural network (HGNN) to model the morphological symmetry (MS) and kinematic structure of rigid body systems.

Heterogeneous graph neural networks [20], denoted as $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, are a type of graph neural network 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. In this work, we construct a HGNN whose node and edge types are directly derived from the robot's kinematic structure. This construction closely follows the framework introduced in [21], with detailed implementation specifics provided in the Appendix.A.

A rigid body system is a collection of solid bodies that maintain a fixed shape and size while moving under the influence of forces and torques. When these bodies are connected through joints allowing relative motion, they form a kinematic chain, which describes the movement of interconnected rigid bodies in a structured manner. The morphological symmetry is from morphological or structural similarity resulting from replicated kinematic chains and body parts with symmetric mass distributions, and the details of morphological symmetry are provided in the Appendix.B. The kinematic structure of the robot can be represented as an adjacency matrix within a graph neural network, allowing the GNN architecture to process geometric proprioceptive sensor readings associated with specific links and joints.

Our approach comprises two key components (Fig.2): (1) an automatically constructed HGNN that preserves the system's 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 chain 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 $\mathbb{G}_{\mathbb{E}}$ denotes 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 the element in group $\mathbb{G}_m$, and subgraphs with same q lies in the same orbit.
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, completes each group orbit by replicating elements along missing transformations and label them as $\mathcal{G}_{p,q}$.
5) Connect the base nodes $\{\mathcal{V}_{b,p}\}$ using Cayley Graph [22]. Connect each subgraph $\mathcal{G}_{p,q}$ to its corresponding base node $\mathcal{V}_{b,p}$ with edge type $\mathcal{E}_q$, formalizing the full graph $\mathcal{G}$.
6) Add input 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.

After completing the first 5 steps of our construction process, we obtain a graph G that preserves the system's inherent geometric symmetry. To ensure that the learned representations from heterogeneous graph neural networks respect the morphological symmetry group $\mathbb{G}$, we integrate an additional encoder-decoder pair in step 6, as shown in Fig. 2 (b), (d). This enables the HGNN to capture structural equivalences, preserving the morphological symmetry of the robot within the overall learning framework. We provide a mathematical proof demonstrating that our constructed graph is equivariant under morphological symmetry transformations in Appendix.C.

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.

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. 1.

IV. Experiments

We present MS-HGNN as a general model for tasks involving rigid body systems, focusing on quadruped robots. Leveraging a specialized GNN structure , our model effectively captures morphological information and is validated on multiple tasks: contact state detection with real-world data (classification), ground reaction force (GRF) estimation and centroidal momentum estimation with simulated data (regression) from various quadruped platforms. These components are essential for understanding quadruped dynamics and enabling effective control. We compare our results with CNN [23], state-of-the-art $\mathbb{G}$-equivariant neural networks CNN-Aug and ECNN with $\mathbb{C}_2$ symmetry [24], and the morphology-aware model, MI-HGNN [21]. GRF results are provided in Appendix F; detailed dynamics are in Appendix D.

A. Contact State Detection for Mini-Cheetah Robot

We address contact prediction for the Mini-Cheetah robot [25] using real-world data [23], covering diverse gaits and terrains such as sidewalk, asphalt, concrete, pebbles, forest, and grass. Each sample contains measured joint angles $\mathbf{q} \in \mathbb{R}^{12}$, joint velocities $\dot{\mathbf{q}} \in \mathbb{R}^{12}$, base linear acceleration $\mathbf{a}_b \in \mathbb{R}^3$, and base angular velocity $\boldsymbol{\omega}_b \in \mathbb{R}^3$ from the IMU. It also includes estimated foot position $\mathbf{p}_l \in \mathbb{R}^3$ and velocity $\mathbf{v}_l \in \mathbb{R}^3$ via forward kinematics, where $l = \{LF, LH, RF, RH\}$ indexes the legs. The ground-truth binary contact state $\mathbf{c}_l \in \mathbb{B}, \mathbb{B} = \{0, 1\}$, is generated offline using a non-causal algorithm [23]. The dataset contains approximately 1M synchronized samples at 1000 Hz. Following [24], we use the same test sequences and split the rest into 85% train and 15% val. Since the test set contains unseen gait-terrain combinations, this partition supports evaluating generalization to out-of-distribution scenarios.

Mini-Cheetah exhibits $\mathbb{G} = \mathbb{K}_4$ symmetry, motivating use of both $\mathbb{K}_4$ group and its subgroup $\mathbb{C}_2$ to our model design. All models take a 150-sample histories $[\mathbf{q}, \dot{\mathbf{q}}, \mathbf{a}_b, \omega_b, \mathbf{p}, \mathbf{v}] \in \mathbb{R}^{54}$, to predict a 4-leg contact state $\hat{\mathbf{c}} \in \mathbb{B}^4$ at t. For MI-HGNN and MS-HGNN, the inputs are structured as a graph where the base $(\mathbf{a}_b, \boldsymbol{\omega}_b)$, joint $(q_j, \dot{q}_j)$, and foot $(\mathbf{p}_l, \mathbf{v}_l)$ measurements are assigned to base $(\mathcal{V}_b)$, joint $(\mathcal{V}_t)$ and foot $(\mathcal{V}_f)$ nodes, respectively. Contact state predictions are produced at each foot node $(\mathcal{V}_f)$. In MS-HGNN-$\mathbb{C}_2$ and MS-HGNN-$\mathbb{K}_4$ models, the 2 and 4 base nodes receive identical input features. MS-HGNN uses 8 message-passing layers, hidden size 128, and is trained for 49 epochs with a learning rate of 10^{-4}. We use metrics form [21, 24]: foot-wise binary F1-score, averaged F1-score (mean across legs), and 16-state contact accuracy, where a prediction is correct only if all legs are classified correctly.

Fig. 3-left reports classification performance and parameter size. Compared to non-graph baselines (CNN, CNN-Aug, ECNN), graph-based models perform better with fewer parameters. Specifically, MS-HGNN-$\mathbb{K}_4$ improves accuracy by 11% over ECNN, the best-performing non-graph-based model, while using only 38% of its parameters. This demonstrates the effectiveness and efficiency of graph-based morphology-aware architectures. Morphology-informed graph network constrains

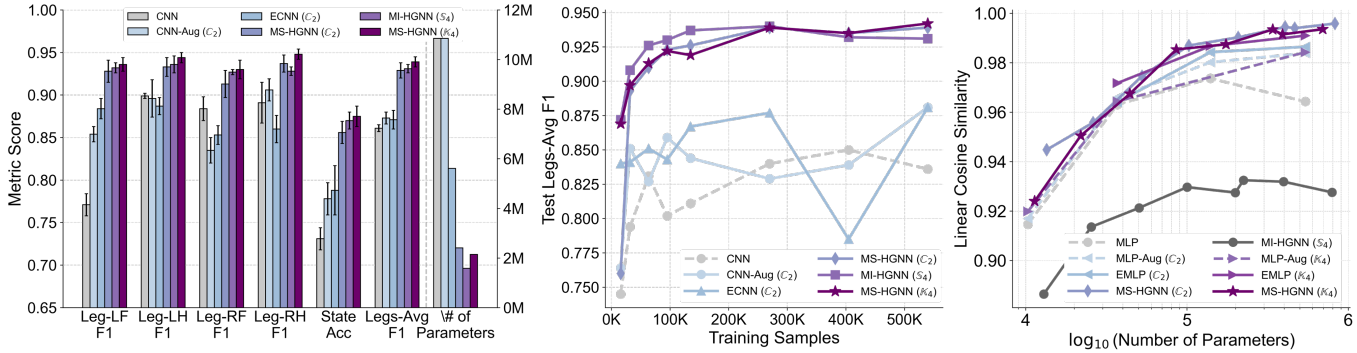

Fig. 3: **Left: Contact state detection on the real-world Mini-Cheetah dataset** [23]. F1-score for each leg, averaged F1-score, and 16-state contact accuracy, averaged over 4 random runs. Parameter counts for each model are also reported. **Center: Sample efficiency on the Mini-Cheetah dataset** [23]. Averaged F1-score of models trained with varying portions of the training set. Our MS-HGNN ($\mathbb{C}_2$ and $\mathbb{K}_4$) achieves an F1-score of approximately 0.9 using only 5% of the data. **Right: Model efficiency on the synthetic Solo dataset** [24]. Linear cosine similarity for models of varying sizes. MS-HGNN ($\mathbb{C}_2$ and $\mathbb{K}_4$) demonstrates superior parameter efficiency and avoids overfitting.

Model	**Lin. Cos. Sim** ↑	**Ang. Cos. Sim** ↑	**Test MSE** ↓
MLP	0.9617	0.9523	0.0499
MLP-Aug ($\mathbb{C}_2$)	0.9639	0.9535	0.0478
EMLP ($\mathbb{C}_2$)	0.9610	0.9528	0.0503
MS-HGNN ($\mathbb{C}_2$)	**0.9903**	**0.9804**	**0.0161**
MLP-Aug ($\mathbb{K}_4$)	0.9647	0.9549	0.0472
EMLP ($\mathbb{K}_4$)	0.9673	0.9580	0.0435
MS-HGNN ($\mathbb{K}_4$)	<u>0.9877</u>	<u>0.9799</u>	<u>0.0189</u>

TABLE I: **Centroidal momentum estimation results** on the synthetic Solo dataset [24]. Each model is evaluated on test sequences using linear cosine similarity, angular cosine similarity, and mean squared error (MSE), averaged over 4 random runs. Standard deviations are reported in Appendix G. **Bold** indicates the best performance, while <u>underlined</u> values denote suboptimal results.

the information flow via message passing along the robot's kinematic chain, therefore, incorporating physical knowledge as a prior and enhancing the model's causality. A direct consequence is the ability to capture the complex correlation with far fewer parameters. MS-HGNN-$\mathbb{K}_4$ also outperforms MI-HGNN (F1: 0.939 vs. 0.931, accuracy: 0.875 vs. 0.870), showing the benefits of preserving morphological symmetry. MI-HGNN uses $\mathbb{S}_4$ geometric symmetry, which enforces permutation-equivariant across all legs and over-constrains the learning problem, leading to suboptimal performance.

Among models following $\mathbb{C}_2$ (excluding MI-HGNN), MS-HGNN-$\mathbb{C}_2$ achieves the best classification results. In addition, the performance gain introduced by following $\mathbb{K}_4$ over $\mathbb{C}_2$ further demonstrated the benefits of exploiting the morphological symmetries. From a data-augmentation perspective, using $\mathbb{K}_4$ doubles the training set compared to $\mathbb{C}_2$. Further analysis on model and sample efficiency can be found at Appendix E.

B. Centroidal Momentum Estimation for Solo Robot

This task estimates linear $\boldsymbol{l} \in \mathbb{R}^3$ and angular $\boldsymbol{k} \in \mathbb{R}^3$ momentum from joint-space position and velocity $\mathbf{q} \in \mathbb{R}^{12}, \dot{\mathbf{q}} \in \mathbb{R}^{12}$. Data is simulated in PINOCCHIO [26] for Solo robot [11] with $\mathbb{G} = \mathbb{K}_4$ symmetry. Unlike contact estimation tasks, this setting challenges the model to infer angular momentum from multiple base nodes. We construct MS-HGNNs with $\mathbb{C}_2$ and $\mathbb{K}_4$, attaching morphology encoders to joint nodes and decoders to base nodes. MSE losses from all base nodes are averaged during training.

Tab. I reports results (mean over 4 runs) evaluated with cosine similarity and MSE. MI-HGNN is excluded due to its incomparable performance (e.g., linear cosine similarity 0.9301±0.0017, angular cosine similarity 0.5173 ± 0.0016, and test MSE 0.3421 ± 0.0009). Our MS-HGNN outperforms all baselines. MI-HGNN's underperformance stems from its use of $\mathbb{S}_4$, which misaligns with Solo's morphology, rendering it fails to capture angular dynamics effectively. In contrast, our MS-HGNN accurately embeds true symmetries using $\mathbb{C}_2$ and $\mathbb{K}_4$, enabling improved representation and learning. Fig. 3-right shows model efficiency: MS-HGNN-$\mathbb{C}_2$ reaches 0.9448 cosine similarity with only 13,478 parameters. Both MS-HGNN variants scall well with model size, while MI-HGNN and MLP tend to overfit as parameter count increases.

V. CONCLUSIONS

We present MS-HGNN, a general and versatile network architecture for robotic dynamics learning by integrating symbolic inductive biases such as robotic kinematic structures and morphological symmetries into a neural graph-based framework. By embedding these structured priors, MS-HGNN bridges the strengths of symbolic reasoning with the flexibility of neural networks, aligning with the broader goals of neural-symbolic learning. Our theoretical analysis and empirical results demonstrate that exploiting robot morphology not only improves generalization and sample efficiency but also leads to more compact and interpretable models. Future work will explore incorporating additional physical priors, extending to meta- and reinforcement learning settings, and unifying perception and control within a neural-symbolic paradigm.

References

[1] R. M. Murray, Z. Li, and S. S. Sastry, A mathematical introduction to robotics manipulation. CRC Press, 1994.

[2] M. W. Spong, S. Hutchinson, and M. Vidyasagar, Robot Modeling and Control. Wiley, 2005.

[3] T. Miki, J. Lee, J. Hwangbo, L. Wellhausen, V. Koltun, and M. Hutter, "Learning robust perceptive locomotion for quadrupedal robots in the wild," Science Robotics, vol. 7, no. 62, p. eabk2822, 2022.

[4] Z. Li, X. Cheng, X. B. Peng, P. Abbeel, S. Levine, G. Berseth, and K. Sreenath, "Reinforcement learning for robust parameterized locomotion control of bipedal robots," in IEEE International Conference on Robotics and Automation, pp. 2811–2817, 2021.

[5] T. Mruthyunjaya, "Kinematic structure of mechanisms revisited," Mechanism and Machine Theory, vol. 38, no. 4, pp. 279–320, 2003.

[6] H. Taheri and N. Mozayani, "A study on quadruped mobile robots," Mechanism and Machine Theory, vol. 190, p. 105448, 2023.

[7] M. Camurri, M. F. Fallon, S. Bazeille, A. Radulescu, V. Barasuol, D. G. Caldwell, and C. Semini, "Probabilistic contact estimation and impact detection for state estimation of quadruped robots," IEEE Robotics and Automation Letters, vol. 2, no. 2, pp. 1023–1030, 2017.

[8] J. Kang, H. Kim, K. H. Choi, and K. Kim, "External force estimation of legged robots via a factor graph framework with a disturbance observer," in IEEE International Conference on Robotics and Automation, pp. 12120–12126, 2023.

[9] Y. Yang, G. Shi, X. Meng, W. Yu, T. Zhang, J. Tan, and B. Boots, "Cajun: Continuous adaptive jumping using a learned centroidal controller," in Conference on Robot Learning, pp. 2791–2806, 2023.

[10] H. Smith, A. Shankar, J. Gielis, J. Blumenkamp, and A. Prorok, "So(2)-equivariant downwash models for close proximity flight," IEEE Robotics and Automation Letters, vol. 9, no. 2, pp. 1174–1181, 2024.

[11] D. Ordoñez-Apraez, G. Turrisi, V. Kostic, M. Martin, A. Agudo, F. Moreno-Noguer, M. Pontil, C. Semini, and C. Mastalli, "Morphological symmetries in robotics," The International Journal of Robotics Research, vol. 0, no. 0, 2025.

[12] M. O'Connell, G. Shi, X. Shi, K. Azizzadenesheli, A. Anandkumar, Y. Yue, and S. Chung, "Neural-fly enables rapid learning for agile flight in strong winds," Science Robotics, vol. 7, no. 66, 2022.

[13] M. Lutter and J. Peters, "Combining physics and deep learning to learn continuous-time dynamics models," The International Journal of Robotics Research, vol. 42, no. 3, pp. 83–107, 2023.

[14] C. Neary and U. Topcu, "Compositional learning of dynamical system models using port-hamiltonian neural networks," in Learning for Dynamics and Control Conference, pp. 679–691, 2023.

[15] F. Xie, G. Shi, M. O'Connell, Y. Yue, and S. Chung, "Hierarchical meta-learning-based adaptive controller," in IEEE International Conference on Robotics and Automation, pp. 18309–10315, 2024.

[16] E. S. Lupu, F. Xie, J. A. Preiss, J. Alindogan, M. Anderson, and S. Chung, "Magic^{vfm}-meta-learning adaptation for ground interaction control with visual foundation models," IEEE Transactions on Robotics, vol. 41, pp. 180–199, 2025.

[17] M. Raissi, P. Perdikaris, and G. 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, vol. 378, pp. 686–707, 2019.

[18] S. Greydanus, M. Dzamba, and J. Yosinski, "Hamiltonian neural networks," in Advances in Neural Information Processing Systems, pp. 15353–15363, 2019.

[19] A. Sanchez-Gonzalez, N. Heess, J. T. Springenberg, J. Merel, M. A. Riedmiller, R. Hadsell, and P. W. Battaglia, "Graph networks as learnable physics engines for inference and control," in International Conference on Machine Learning, pp. 4467–4476, 2018.

[20] C. Shi, "Heterogeneous graph neural networks," in Graph Neural Networks: Foundations, Frontiers, and Applications, Springer Nature Singapore, 2022.

[21] D. Butterfield, S. S. Garimella, N.-J. Cheng, and L. Gan, "MI-HGNN: Morphology-informed heterogeneous graph neural network for legged robot contact perception," arXiv preprint arXiv: 2409.11146, 2024.

[22] A. Cayley, "Desiderata and suggestions: No. 2. the theory of groups: graphical representation," American Journal of Mathematics, vol. 1, no. 2, pp. 174–176, 1878.

[23] T. Lin, R. Zhang, J. Yu, and M. Ghaffari, "Legged robot state estimation using invariant Kalman filtering and learned contact events," in Conference on Robot Learning, pp. 1057–1066, 2021.

[24] D. F. O. Apraez, M. Martín, A. Agudo, and F. Moreno, "On discrete symmetries of robotics systems: A group-theoretic and data-driven analysis," in Robotics: Science and Systems, 2023.

[25] B. Katz, J. D. Carlo, and S. Kim, "Mini cheetah: A platform for pushing the limits of dynamic quadruped control," in International Conference on Robotics and Automation, pp. 6295–6301, 2019.

[26] J. Carpentier, G. Saurel, G. Buondonno, J. Mirabel, F. Lamiraux, O. Stasse, and N. 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, pp. 614–619, 2019.

[27] W. L. Hamilton, Graph Representation Learning. Morgan & Claypool Publishers, 2020.

[28] J. An and I. Lee, "Artificial neural network-based ground reaction force estimation and learning for dynamic-legged robot systems," PeerJ Computer Science, vol. 9,

p. e1720, 2023.
[29] P. Arena, M. F. P. Cusimano, L. Patanè, and P. Manoonpong, "Ground reaction force estimation in a quadruped robot via liquid state networks," in International Joint Conference on Neural Networks, pp. 1–8, 2022.
[30] J. Norby, Y. Yang, A. Tajbakhsh, J. Ren, J. K. Yim, A. Stutt, Q. Yu, N. Flowers, and A. M. Johnson, "Quad-SDK: Full stack software framework for agile quadrupedal locomotion," in ICRA Workshop on Legged Robots, 2022.

Appendix A
Morphology-Informed Heterogeneous Graph Neural Network

A Morphology-Informed Heterogeneous Graph Neural Network (MI-HGNN) [21] is an HGNN with node and edge types directly inferred 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.2(c), while the links connecting these components are modeled as edges.

Appendix B
Morpholofical Symmetry

The morphological symmetry is from morphological or structural similarity resulting from replicated kinematic chains and body parts with symmetric mass distributions. Morphological symmetry group $\mathbb{G}$ represents feasible state transformation including reflection and rotation that adjust the robot's state $(\mathbf{q}, \dot{\mathbf{q}})$ to a reachable state $(g \mathbin{\hat{\triangleright}} \mathbf{q}, g \mathbin{\hat{\triangleright}} \dot{\mathbf{q}})$, where g is the group action, $\mathbf{q} \in \mathbb{R}^{n_q}$ is the generalized position coordinates, with n_q as the number of states [11]. The formal definition of morphological symmetry action is presented in Eq. 1.

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

This transformation includes a reorientation of the base's body $g \mathbin{\hat{\triangleright}} \mathbf{X}_B = \mathbf{X}_g \mathbf{X}_B \mathbf{X}_g^{-1} \in \mathbb{SE}_d$, where $\mathbf{X}_B := \begin{bmatrix} \mathbf{R}_B & \mathbf{r}_B \\ \mathbf{0} & 1 \end{bmatrix} \in \mathbb{SE}_d$ is the base's state configuration, with $\mathbf{R}_B \in \mathbb{SO}_d$ as the rotation matrix representing the base's orientation, and transformation of the joint space configuration $g \mathbin{\hat{\triangleright}} \mathbf{q}_{js} := \rho_{\mathcal{M}}(g) \mathbf{q}_{js}$. Once the base of the kinematic structure is identified, the number of unique kinematic branches in the robot can be determined with representation $\mathbb{S} = \{\mathbb{S}_1, \dots, \mathbb{S}_k\}$. Each branch $\mathbb{S}_i$ contains $n_{\text{dof}}(\mathbb{S}_i) \in \mathbb{N}$ degrees of freedom and is replicated $n_{\text{rep}}(\mathbb{S}_i) \in \mathbb{N}$ times throughout the robot's kinematic structure. The set of labels for the instance of branch $\mathbb{S}_i$ is given by $\mathbb{S}_i = \{\mathbb{S}_{i,1}, \dots, \mathbb{S}_{i,n_{\text{rep}}(\mathbb{S}_i)}\}$.

A typical quadruped consists of a single kinematic chain $\mathbb{S}_s = \{\mathbb{S}_{leg}\}$, which is replicated $n_{\text{rep}}(\mathbb{S}_{leg}) = 4$ times. The action of a morphological symmetry in the joint space results in a permutation of the roles of branches with the same type denoted as $g \triangleright s_{i,j} := s_{i,g(j)} \in \mathbb{S}_i$ is the label that j is mapped to under the permutation induced by g. This leads to the decomposition of the joint space configuration:

$$\begin{aligned} & g \triangleright s_{i,j} := s_{i,g(j)} \in \mathbb{S}_i, \\ & \rho_{\mathbb{S}_i}(g) \begin{bmatrix} s_{i,1} \\ s_{i,2} \\ \vdots \end{bmatrix} = \begin{bmatrix} s_{i,g(1)} \\ s_{i,g(2)} \\ \vdots \end{bmatrix}, \quad \forall i \in [1,k],\ j \in [1, n_{\text{rep}}(s_i)] \end{aligned} \tag{2}$$

where $\rho_{\mathbb{S}_i}(g)$ is the permutation representation acting on the labels of the instances of branch type s_i. Following our example with the quadruped robot, the action of g induces a permutation of the left and right configurations $g \triangleright s_{leg,lf} = s_{leg,rf}$, $g \triangleright s_{leg,rf} = s_{leg,lf}$, $g \triangleright s_{leg,lh} = s_{leg,rh}$, and $g \triangleright s_{leg,rh} = s_{leg,lh}$. Given that these permutations do not mix the distinct branch types, we can adopt a basis for the joint space configuration space, leading to the decomposition of its associated group representation.

$$\begin{aligned} \mathcal{M} &:= \mathcal{M}_{[\mathbb{S}_1]} \times \cdots \times \mathcal{M}_{[\mathbb{S}_k]} \subseteq \mathbb{R}^{n_j}, \\ \mathcal{M}_{[\mathbb{S}_i]} &:= \bigotimes_{j=1}^{n_{\text{rep}}(\mathbb{S}_i)} \mathcal{M}_{\mathbb{S}_i}, \\ \rho_{\mathcal{M}} &:= \rho_{\mathcal{M}_{[\mathbb{S}_1]}} \oplus \cdots \oplus \rho_{\mathcal{M}_{[\mathbb{S}_k]}}, \\ \rho_{\mathcal{M}_{[\mathbb{S}_i]}} &:= \rho_{\mathbb{S}_i} \otimes \rho_{\mathcal{M}_{\mathbb{S}_i}}. \end{aligned} \tag{3}$$

where $\mathcal{M}_{\mathbb{S}_i} \subseteq \mathbb{R}^{n_{\text{dof}}(\mathbb{S}_i)}$ represents the configuration space of a single instance of type $\mathbb{S}_i$. For further details, we refer the reader to [11].

Appendix C
Proof Details

We provide a mathematical proof demonstrating that our constructed graph is equivariant under morphological symmetry transformations. 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 $\{\mathcal{G}_1, \dots, \mathcal{G}_q\}$. Each subgraph $\mathcal{G}_i$ is further subdivided into instances $\{\mathcal{G}_{i,1}, \dots, \mathcal{G}_{i,p}\}$, 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.

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 $g_m \mathbin{\hat{\triangleright}} (\cdot)$. 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

Model	Lin. Cos. Sim ↑	Ang. Cos. Sim ↑	Test MSE ↓
MLP	0.9617 ± 0.0036	0.9523 ± 0.0032	0.0499 ± 0.0037
MLP-Aug ($\mathbb{C}_2$)	0.9639 ± 0.0026	0.9535 ± 0.0029	0.0478 ± 0.0020
EMLP ($\mathbb{C}_2$)	0.9610 ± 0.0039	0.9528 ± 0.0051	0.0503 ± 0.0053
MS-HGNN ($\mathbb{C}_2$)	0.9903 ± 0.0001	0.9804 ± 0.0015	0.0161 ± 0.0006
MLP-Aug ($\mathbb{K}_4$)	0.9647 ± 0.0023	0.9549 ± 0.0023	0.0472 ± 0.0014
EMLP ($\mathbb{K}_4$)	0.9673 ± 0.0045	0.9580 ± 0.0032	0.0435 ± 0.0048
MS-HGNN ($\mathbb{K}_4$)	0.9877 ± 0.0007	0.9799 ± 0.0010	0.0189 ± 0.0007

TABLE II: **Centroidal momentum estimation results** on the synthetic Solo dataset [24]. Each model is evaluated on test sequences using linear cosine similarity, angular cosine similarity, and mean squared error (MSE), averaged over 4 random runs. Results are reported as mean $\pm$ standard deviation. Higher cosine similarity and lower MSE indicate better performance.

$\rho_{\mathcal{G}_q}(g_m) \in \mathbb{R}^{p\times p}$ 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:

$$\rho_{\mathcal{G}_q}(g_m)\begin{bmatrix}\mathcal{G}_{p_1,q}\\ \mathcal{G}_{p_2,q}\\ \vdots\end{bmatrix} = \begin{bmatrix}\mathcal{G}_{g_m(p_1),q}\\ \mathcal{G}_{g_m(p_2),q}\\ \vdots\end{bmatrix}. \tag{4}$$

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

$$\begin{aligned}\rho_b &:= \operatorname{diag}\left(\rho_{b\mathcal{M}_{[\mathcal{G}_1]}}(g_m), \ldots, \rho_{b\mathcal{M}_{[\mathcal{G}_k]}}(g_m)\right),\\ \rho_{b\mathcal{M}_{[\mathcal{G}_i]}}(g_m) &:= \rho_{\mathcal{G}_i}(g_m) \otimes \rho_{b\mathcal{M}_{\mathcal{G}_i}}(g_m).\end{aligned} \tag{5}$$

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:

$$\begin{aligned}\forall \rho_b \in \mathbb{G}_m, \quad \phi_{\rho_b}(A_{\mathcal{G}}) &= \rho_b A_{\mathcal{G}} \rho_b^T = A_{\mathcal{G}},\\ \phi_{\rho_b}(X_{\mathcal{G}}) &= \rho_b X_{\mathcal{G}} = X_{\mathcal{G}}.\end{aligned} \tag{6}$$

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

Lemma 1 (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 [27]:

$$\begin{aligned}\forall g_m \in \mathbb{G}_m, \quad g_m \triangleright z_{\mathcal{G}}(X_{\mathcal{G}}) &= z_{\mathcal{G}}\big(\phi_{\rho_b}(X_{\mathcal{G}})\big)\\ &= z_{\mathcal{G}}(\rho_b X_{\mathcal{G}}) = z_{\mathcal{G}}(g_m \triangleright X_{\mathcal{G}}).\end{aligned} \tag{7}$$

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

Theorem 2 (Morphological-Symmetry-Equivariant HGNN): With the input encoder h and the output decoder l that satisfies the following condition:

$$\begin{aligned}\forall g_{m,p} \in \mathbb{G}_m, \quad h(X_{\mathcal{G}_{p,q}}) &= \rho_{\mathcal{M}_{\mathcal{G}_q}}(g_{m,p}) X_{\mathcal{G}_{p,q}}\\ l(X_{\mathcal{G}_{p,q}}) &= \rho_{\mathcal{M}_{\mathcal{G}_q}}(g_{m,p})^{-1} X_{\mathcal{G}_{p,q}},\end{aligned} \tag{8}$$

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:

$$\forall g_m \in \mathbb{G}_m, \quad g_m \overset{\circ}{\triangleright} l(x) = l(g_m \triangleright x) \tag{9}$$

$$g_m \triangleright h(x) = h(g_m \overset{\circ}{\triangleright} x) \tag{10}$$

Our GNN is equivariant to morphological group actions:

$$\begin{aligned}\forall g_m \in \mathbb{G}_m, \quad g_m \overset{\circ}{\triangleright} l(x) &= l(g_m \triangleright x)\\ g_m \triangleright h(x) &= h(g_m \overset{\circ}{\triangleright} x)\end{aligned} \tag{11}$$

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

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:

$$\begin{aligned}\forall \rho_b \in \mathbb{G}_m, \quad \phi_{\rho_b}(A_{\mathcal{G}}) &= \rho_b A_{\mathcal{G}} \rho_b^T = A_{\mathcal{G}}\\ \phi_{\rho_b}(X_{\mathcal{G}}) &= \rho_b X_{\mathcal{G}} = X_{\mathcal{G}}\end{aligned} \tag{12}$$

Proof 1: It is easy to find out that the mapping ϕ_{ρ_b} satisfies the following properties:

$$\begin{aligned}&\text{Injective:}\\ &\forall \rho_b \in \mathbb{G}_m,\ A_{\mathcal{G}_1} = A_{\mathcal{G}_2} \Rightarrow \phi_{\rho_b}(A_{\mathcal{G}_1}) = \phi_{\rho_b}(A_{\mathcal{G}_2})\\ &\forall \rho_b \in \mathbb{G}_m,\ X_{\mathcal{G}_1} = X_{\mathcal{G}_2} \Rightarrow \phi_{\rho_b}(X_{\mathcal{G}_1}) = \phi_{\rho_b}(X_{\mathcal{G}_2})\end{aligned} \tag{13}$$

$$\begin{aligned}&\text{Surjective:}\\ &\forall \rho_b \in \mathbb{G}_m,\ \phi_{\rho_b}(\phi_{\rho_b}(A_{\mathcal{G}})) = \phi_{\rho_b}(A_{\mathcal{G}}),\\ &\phi_{\rho_b}(\phi_{\rho_b}(X_{\mathcal{G}})) = \phi_{\rho_b}(X_{\mathcal{G}})\end{aligned} \tag{14}$$

$$\begin{aligned}&\text{Homomorphism:}\\ &\forall \rho_b \in \mathbb{G}_m,\ \phi_{\rho_b}(A_{\mathcal{G}_1} A_{\mathcal{G}_2}) = \rho_b A_{\mathcal{G}_1} (\rho_b^T \rho_b) A_{\mathcal{G}_2} \rho_b^T\\ &= \phi_{\rho_b}(A_{\mathcal{G}_1}) \phi_{\rho_b}(A_{\mathcal{G}_2})\\ &\forall \rho_b \in \mathbb{G}_m,\ \phi_{\rho_b}(X_{\mathcal{G}_1} X_{\mathcal{G}_2}) = \rho_b X_{\mathcal{G}_1} \rho_b X_{\mathcal{G}_2}\\ &= \phi_{\rho_b}(X_{\mathcal{G}_1}) \phi_{\rho_b}(X_{\mathcal{G}_2})\end{aligned} \tag{15}$$

Hence ϕ is an isomorphism from $\mathcal{G}$ to $\mathcal{G}$, which is also known as an automorphism.

Theorem 2 (Morphological-Symmetry-Equivariant HGNN): With the input encoder h and the output decoder l that satisfies the following condition:

$$\forall g_{m,p} \in \mathbb{G}_m, \quad h(X_{\mathcal{G}_{p,q}}) = \rho_{\mathcal{M}_{\mathcal{G}_q}}(g_{m,p}) X_{\mathcal{G}_{p,q}} \quad \text{and} \quad l(X_{\mathcal{G}_{p,q}}) = \rho_{\mathcal{M}_{\mathcal{G}_q}}(g_{m,p})^{-1} X_{\mathcal{G}_{p,q}}, \tag{16}$$

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:

$$\forall g_m \in \mathbb{G}_m, \quad g_m \hat{\triangleright} l(x) = l(g_m \triangleright x) \quad \text{and} \quad g_m \triangleright h(x) = h(g_m \hat{\triangleright} x) \tag{17}$$

Our GNN is equivariant to morphological group actions:

$$\forall g_m \in \mathbb{G}_m, \quad g_m \hat{\triangleright} f_{\mathcal{G}}(X_{\mathcal{G}}) = f_{\mathcal{G}}(g_m \hat{\triangleright} X_{\mathcal{G}}). \tag{18}$$

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

Proof 2: With the pre-defined decoder l, we can show that the Euclidean group actions can be translated into morphological group actions:

$$\begin{aligned} l(g_{m,p_2} \triangleright X_{\mathcal{G}_{p_1,q}}) &= \rho_{\mathcal{M}_{\mathcal{G}_q}}(g_{m,p_1})^{-1} X_{\mathcal{G}_{p_1 p_2,q}} \\ &= \rho_{\mathcal{M}_{\mathcal{G}_q}}(g_{m,p_2}) \rho_{\mathcal{M}_{\mathcal{G}_q}}(g_{m,p_1})^{-1} \rho_{\mathcal{M}_{\mathcal{G}_q}}(g_{m,p_2})^{-1} X_{\mathcal{G}_{p_1 p_2,q}} \\ &= g_{m,p_2} \hat{\triangleright} l(X_{\mathcal{G}_{p_1,q}}) \end{aligned}$$

where $g_{m,p_1} \circ g_{m,p_2} = g_{m,p_2} \circ g_{m,p_1}, \forall g_{m,p_1}, g_{m,p_2} \in \mathbb{G}_m$. Similarly, for the encoder h, the morphological actions can be transformed into Euclidean ones:

$$\begin{aligned} h(g_{m,p_2} \hat{\triangleright} X_{\mathcal{G}_{p_1,q}}) &= \rho_{\mathcal{M}_{\mathcal{G}_q}}(g_{m,p_1}) \rho_{\mathcal{M}_{\mathcal{G}_q}}(g_{m,p_2}) X_{\mathcal{G}_{p_1 p_2,q}} \\ &= \rho_{\mathcal{M}_{\mathcal{G}_q}}(g_{m,p_1 \circ p_2}) X_{\mathcal{G}_{p_1 p_2,q}} = g_{m,p_2} \triangleright h(X_{\mathcal{G}_{p_1,q}}). \end{aligned}$$

Then for the graph representation $f_{\mathcal{G}}(X_{\mathcal{G}}) = l(z_{\mathcal{G}}(h(X_{\mathcal{G}})))$, we have

$$\begin{aligned} f_{\mathcal{G}}(g_m \hat{\triangleright} X_{\mathcal{G}}) &= l(z_{\mathcal{G}}(h(g_m \hat{\triangleright} X_{\mathcal{G}}))) = l(z_{\mathcal{G}}(g_m \triangleright h(X_{\mathcal{G}}))) \\ &= l(g_m \triangleright z_{\mathcal{G}}(h(X_{\mathcal{G}}))) = g_m \hat{\triangleright} l(z_{\mathcal{G}}(h(X_{\mathcal{G}}))) = g_m \hat{\triangleright} f_{\mathcal{G}}(X_{\mathcal{G}}). \end{aligned}$$

which shows the equivariance property of our MS-HGNN to morphological symmetries.

Appendix D
Dynamics of Quadrupedal Rigid Systems

Given the system's generalized velocity $\dot{\mathbf{q}}$, acceleration $\ddot{\mathbf{q}}$, and torques τ, the dynamics of quadrupeds are governed by:

$$\mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} + g(\mathbf{q}) = \mathbf{S}^T \tau + \mathbf{J}_{\text{ext}}(\mathbf{q})^T \mathbf{f}_{\text{ext}}, \tag{19}$$

where $\mathbf{M}(\mathbf{q})$ is the inertia matrix, $\mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})$ the Coriolis matrix, $g(\mathbf{q})$ the gravitational force vector, $\mathbf{S}^T$ the selection matrix, $\mathbf{f}_{\text{ext}}$ the external forces, and $\mathbf{J}_{\text{ext}}(\mathbf{q})$ the external force contact Jacobian. In quadrupeds, ground reaction forces (GRFs) dominate external forces, 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 $\mathbf{J}_l(\mathbf{q})$ denote the GRF and its Jacobian matrix for leg l. Accurate GRF estimation and contact state detection are critical for predicting robot dynamics and are fundamental to control and planning.

Centroidal momenta, comprising the linear and angular momentum of the center of mass (CoM) relative to an inertial frame, capture the robot's overall motion. Estimating these momenta reliably enables the design of adaptive controllers that handle dynamic environments and external disturbances such as wind or human interaction.

Appendix E
Analysis of Model and Training Efficiency

A. Trainable Parameters in $\mathbb{G}$-Equivariant Networks

In general, $\mathbb{G}$-equivariant networks reduce the number of trainable parameters compared to unconstrained neural networks of the same architectural size. For the ECNN and EMLP implementations in [24], the parameters in each $\mathbb{G}$-equivariant layer are reduced by $1/|\mathbb{G}|$ where $|\mathbb{G}|$ is the group order. Therefore, ECNN-$\mathbb{C}_2$ has approximately twice as many trainable parameters of an ECNN-$\mathbb{K}_4$.

Interestingly, our MS-HGNN achieves $\mathbb{G}$-equivariance through its graph structure and minimal edge connectivity, rather than using explicit $\mathbb{G}$-equivariant layers. As a result, MS-HGNN-$\mathbb{C}_2$ and MS-HGNN-$\mathbb{K}_4$ have nearly the same number of trainable parameters(see Fig. 3-left). We further evaluate the **model efficiency** of MS-HGNN by varying the number of parameters and reporting the test linear cosine similarity, as shown in Fig. 3-right. We adjust the model size by changing the number of layers and the hidden size. Notably, MS-HGNN-$\mathbb{C}_2$ achieves a cosine similarity of 0.9448 with only 13,478 parameters, demonstrating high model efficiency. Moreover, the performance of both MS-HGNN variants improves steadily as the model size increases, whereas MI-HGNN and MLP models tend to overfit when scaled up.

B. Sample Efficiency of MS-HGNN

To evaluate sample efficiency, we vary the number of training samples and report averaged F1-scores on the entire test set in Fig. 3-center. Similar to MI-HGNN, both MS-HGNN-$\mathbb{C}_2$ and MS-HGNN-$\mathbb{K}_4$ achieve high performance with limited data, reaching averaged F1-scores around 0.9, while using only 5% of the training samples. With morphological symmetry constraints, our model maintains MI-HGNN's high sample efficiency, which shows the superiority of our model in robotic dynamic learning problems where real-world data is scarce and expensive to collect.

Appendix F
Ground Reaction Force Estimation for A1 Robot

Estimating ground reaction forces (GRFs) is essential for learning legged robot dynamics and robust locomotion. Due to the challenges posed by the discrepancy in robot dynamics and the complexity of contact modeling, learning-based estimators have been proposed to estimate contact forces from proprioceptive sensor data [28, 29]. Our graph-based network is well-suited for this task, naturally fusing multi-modal sensor measurements acquired at local frames via message passing. In this experiment, we use the simulated GRF dataset from [21]. The dataset consists of 500 Hz synchronized 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 the ground truth GRFs ($\boldsymbol{f}_l \in \mathbb{R}^3$ being l the leg index) collected using a simulated A1 robot with $\mathbb{G} = \mathbb{C}_2$ in QUAD-SDK simulator [30]. The objective of this experiment is to evaluate the advantage of our MS-HGNN with morphological symmetry preserving property over the heuristic MI-HGNN in 3D force estimation. We adopt the same experimental setup as described in [21], using a history of 150 samples as the input to predict the GRFs in Z direction (1D) and the 3D GRFs. The hyperparameters and training procedures of MS-HGNN and MI-HGNN remain the same as in previous classification experiment. The quantitative results (mean and standard deviation over 4 runs) are given in Fig 4, where the Root Mean Square Error (RMSE) of each model's prediction is reported for test sequences generated with unseen terrain friction parameters, unseen robot speeds, and unseen terrain types (mean and std over 4 runs). We refer readers to [21] for detailed dataset information. On all test sequences, our MS-HGNN-$\mathbb{C}_2$ achieved lower RMSE compared to MI-HGNN with an overall 1.62% improvement in 3D and 1.50% improvement in 1D GRF prediction. We attribute this marginal improvement in 3D to the small magnitude of GRFs in X and Y direction in this particular dataset.

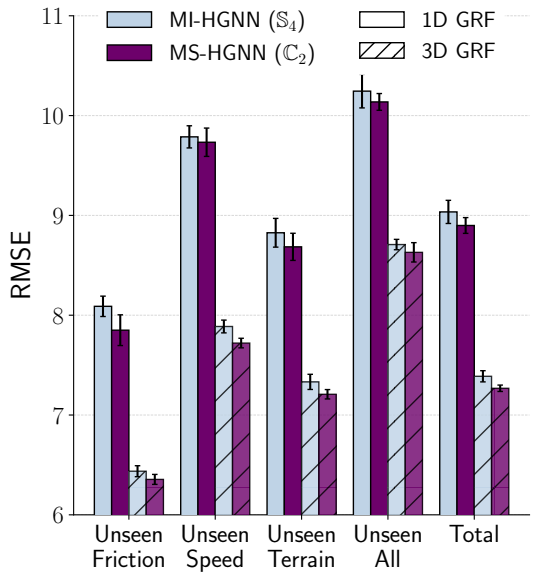

Fig. 4: Ground reaction force estimation test RMSE on simulated A1 dataset [21].

APPENDIX G
FULL TABLE OF RESULTS FOR CENTROIDAL MOMENTUM ESTIMATION ON SOLO ROBOT

The results are reported in Tab. II.