On discrete symmetries of robotics systems: A group-theoretic and data-driven analysis

Daniel Ordonez-Apraez*§, Mario Martin†‡, Antonio Agudo* and Francesc Moreno-Noguer*

* Institut de Robòtica i Informàtica Industrial, CSIC-UPC. †Barcelona Supercomputing Center BSC
§ Istituto Italiano di Tecnologia IIT, ‡ Departament de Ciències de la Computació, Universitat Politècnica de Catalunya UPC

`daniel.ordonez@iit.it, [aagudo, fmoreno]@iri.upc.edu, mmartin@cs.upc.edu`

Abstract—We present a comprehensive study on discrete morphological symmetries of dynamical systems, which are commonly observed in biological and artificial locomoting systems, such as legged, swimming, and flying animals/robots/virtual characters. These symmetries arise from the presence of one or more planes/axis of symmetry in the system's morphology, resulting in harmonious duplication and distribution of body parts. Significantly, we characterize how morphological symmetries extend to symmetries in the system's dynamics, optimal control policies, and in all proprioceptive and exteroceptive measurements related to the system's dynamics evolution. In the context of data-driven methods, symmetry represents an inductive bias that justifies the use of data augmentation or symmetric function approximators. To tackle this, we present a theoretical and practical framework for identifying the system's morphological symmetry group $\mathcal{G}$ and characterizing the symmetries in proprioceptive and exteroceptive data measurements. We then exploit these symmetries using data augmentation and $\mathcal{G}$-equivariant neural networks. Our experiments on both synthetic and real-world applications provide empirical evidence of the advantageous outcomes resulting from the exploitation of these symmetries, including improved sample efficiency, enhanced generalization, and reduction of trainable parameters.

I. INTRODUCTION

Discrete Morphological Symmetries (DMSs) are ubiquitous in both biological and robotic systems. The vast majority of living and extinct animal species, including humans, exhibit bilateral/sagittal reflection symmetry, where the right side of the body is approximately a reflection of the left side (see fig. 1-left). Similarly, a significant number of species exhibit radial symmetry, characterized by two or more morphological symmetry planes/axis (see fig. 1-center) [11]. These symmetries are a consequence of nature's tendency to symmetric body parts and harmonic duplication and distribution of limbs. A pattern perfected and exploited in the design of robotic systems.

To exploit morphological symmetries for control, learning, and computational design, it is necessary to establish a rigorous definition of morphological symmetry within the framework of dynamical systems theory. In section IV, we define a DMS as an energy-preserving linear transformation of the system state configuration, which allows the system to imitate some reflection, rotation, or translation of space. For instance, see how the bipedal robot Atlas and the quadruped Solo in fig. 1 imitate the reflection of space (g_s) with a discrete change in their body and limbs pose (state configuration). The existence of a DMS is subjected to constraints in the system's morphology, which manifest in identifiable symmetry constraints of the system's generalized mass matrix (section III).

Symmetries of the state-space of a dynamical system translate to symmetries of the system's dynamics and control [29]. Thus, DMSs imply the presence of symmetries in the dynamics and control of body motions, resulting in symmetries in all proprioceptive and exteroceptive measurements related to the evolution of the system's dynamics (e.g., joint position/velocity/torque, depth images, contact forces). This property, in data-driven applications, opens the door for the use of data augmentation to mitigate challenges of data collection in the fields of robotics, computer graphics, and computational biology. Similarly, the use of symmetry constraints in machine learning algorithms is a known technique to enhance generalization and sample efficiency, while reducing the number of trainable parameters [29, 6, 23].

Despite the potential benefits of exploiting symmetry and the ubiquitous presence of morphological symmetries in robotic/biological/virtual systems, this relevant inductive bias is frequently left unexploited in data-driven applications in robotics, computational biology, and computer graphics. We attribute the scarce adoption of these techniques to a missing theoretical framework that consolidates the concept of morphological symmetries, facilitating their study and identification. And, to a missing practical framework enabling the efficient and convenient exploitation of symmetries in real-world data-driven applications.

The identification of morphological symmetries and how these extend to symmetries of proprioceptive and exteroceptive data is currently a laborious and error-prone system-specific process, due to the lack of a clear theoretical framework. As a result, most recent works that exploit some morphological symmetry (e.g., [27, 1, 28] in computer graphics and [23, 16, 9, 5] in robotics/dynamical systems) have only been applied to simple systems and the simplest morphological symmetry: reflection/sagittal symmetry (see fig. 1-left), with the exception of Finzi et al. [5]. However, these works provide little guidance on how to apply these techniques to other systems, particularly those with more than a single morphological symmetry.

Contrary to previous works, this paper focuses on understanding and exploiting morphological symmetries in arbitrary dynamical systems, with any number of symmetries. To
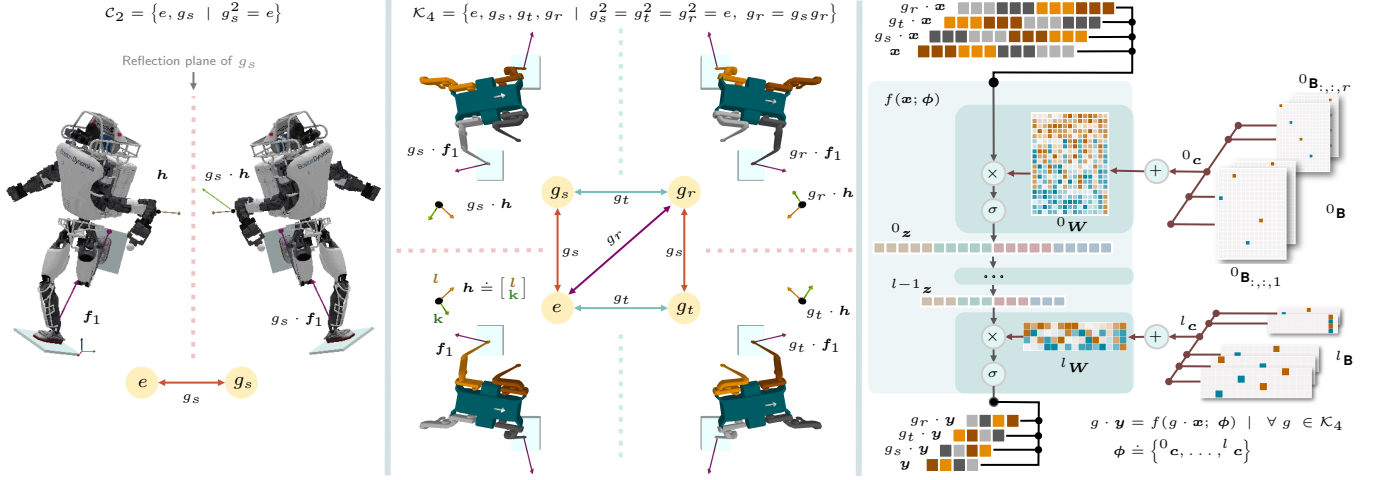
Fig. 1: **Left:** Symmetric configurations of the bipedal robot Atlas (3D animation) illustrating its morphological symmetry described by the reflection group $\mathcal{C}_2$. The robot can imitate reflections g_s (hint: note the non-reflected text on the chest). **Middle:** Top-view of symmetric configurations of the quadruped robot Solo (3D animation) showcasing its morphological symmetries described by the Klein four-group $\mathcal{K}_4$. The robot can imitate two perpendicular reflections (g_s, g_t) and a $180°$ rotation (g_r) of space (hint: observe the unreflected/unrotated robot's heading direction and legs coloring). Symmetry transformations (arrows) affect the robot's configuration, as well as proprioceptive measurements (center of mass linear $\boldsymbol{l}$ and angular $\mathbf{k}$ momentum) and exteroceptive measurements (terrain elevation, external force $\boldsymbol{f_1}$). **Right:** Diagram of a toy $\mathcal{K}_4$-equivariant neural network, processing the symmetric states of robot Solo $\boldsymbol{x}$ and outputting the symmetric binary foot contact states $\boldsymbol{y}$ (see section V).

achieve this, we study morphological symmetries from the lens of dynamical systems and of group theory (the field of mathematics that studies symmetries, broadly used in machine learning and physics) (sections II and III). In summary, our work presents the following theoretical contributions:

✤ Identification of the set of DMSs of a dynamical system as a symmetry group $\mathcal{G}$, that is isomorphic to a group of isometries of the Euclidean space (section IV).
✤ Characterization of DMSs as transformations to which the system's generalized mass matrix is equivariant. Enabling algorithmic identification of $\mathcal{G}$ (section III).
✤ Characterization of how symmetries in proprioceptive and exteroceptive measurements arise from DMSs.

Furthermore, our practical contributions (section V) are:

❖ An open-access repository[1] with example dynamical systems with DMSs, and the tools to prototype large-scale $\mathcal{G}$-equivariant Neural Networks (NN) for arbitrary DMSs.
❖ Proof of an approximate $1/|\mathcal{G}|$ reduction in the trainable parameters of NN. Being $|\mathcal{G}|$ the number of DMSs.
❖ Derivation of an optimal initialization for the trainable parameters of $\mathcal{G}$-equivariant NN layers.

Lastly, we provide optional appendices where we extend our theoretical derivations and provide tutorial-like examples.

II. BACKGROUND ON SYMMETRY GROUPS

Group theory is the default language for studying symmetry transformations. Thus, we provide a shallow introduction to the field[2] and define the notation required for our develop-

---

[1] github.com/Danfoa/RobotEquivariantNN
[2] Being this short section undoubtedly an unsatisfactory introduction to group theory, we refer the uninitiated and interested reader to Carter [4] for intuition and to Bronstein et al. [2] for a machine learning introduction.

ment. In a nutshell, a symmetry group in group theory is an abstraction of the set of symmetries that different geometric objects have. Understanding symmetry as a transformation that conserves a relevant property of the object (e.g. energy).

For instance, in fig. 1-left the reflection group $\mathcal{C}_2$ describes the symmetries that vectors, pseudo-vectors, rigid-bodies, and the robot Atlas have to a reflection of space g_s. Being the symmetries, transformations that preserve vector magnitudes and the robot's energy. Similarly in fig. 1-center, the Klein four-group $\mathcal{K}_4$ describes the symmetries that the quadruped robot Solo has to $180°$ rotations (g_r) and two perpendicular reflections (g_s, g_t). While on fig. 1-right the same group describes the symmetries of the input $\boldsymbol{x}$ and output $\boldsymbol{y}$ vector spaces of a $\mathcal{K}_4$-equivariant NN.

This formalism enables us to study the set of DMSs of a system and the set of symmetries of proprioceptive and exteroceptive data measurements as different representations of the same symmetry group. Formally, a symmetry group is a set of invertible symmetry transformations (or *actions*) $\mathcal{G} = \{e, g_1, g_1^{-1}, g_2, \dots\}$, including the trivial action e, which leaves objects unchanged. A group has an associative composition operator $(\cdot) : \mathcal{G} \times \mathcal{G} \rightarrow \mathcal{G}$ mapping group actions to other group actions. Since two different geometric objects can share the same symmetry group, an action of the group must act differently on the two objects. Here, is where group *representations* allow us to use the familiar language of linear algebra to characterize how an action g transforms a specific geometric object, say $\boldsymbol{x} \in \mathcal{X} \subseteq \mathbb{R}^k$. A representation $\rho_{\mathcal{X}} : \mathcal{G} \rightarrow GL(k)$ is a mapping from group actions to the set of invertible square matrices of k dimensions (the General Linear group GL). Thus, a representation specifies how objects $\boldsymbol{x} \in \mathcal{X}$ are transformed by group actions: $g \cdot \boldsymbol{x} \doteq \rho_{\mathcal{X}}(g)\boldsymbol{x}$.

A fundamental concept for this work is the notion of function $\mathcal{G}$-equivariance and $\mathcal{G}$-invariance. The function $f : \mathbb{R}^n \rightarrow \mathbb{R}^m$, is said to be $\mathcal{G}$-equivariant or $\mathcal{G}$-invariant if:

$$\underbrace{g \cdot \boldsymbol{y} = f(g \cdot \boldsymbol{x}) \,|\, \forall g \in \mathcal{G}}_{Equivariance} \quad \text{or} \quad \underbrace{\boldsymbol{y} = f(g \cdot \boldsymbol{x}) \,|\, \forall g \in \mathcal{G}}_{Invariance}. \quad (1)$$

Roughly speaking, an equivariant function maps symmetries of the input to symmetries of the output, while an invariant function maps symmetries of the input to an invariant output.

III. LAGRANGIAN MECHANICS AND SYMMETRIES OF DYNAMICAL SYSTEMS

Here we provide a group-theoretic perspective of symmetries in an arbitrary dynamical system. The definitions and notations of this section are fundamental for understanding the objective of this work, namely DMSs. To this end, let us consider a dynamical system with generalized coordinates $\boldsymbol{q} \in Q \subseteq \mathbb{R}^n$ and velocities $\dot{\boldsymbol{q}} \in T_{\boldsymbol{q}}Q \subseteq \mathbb{R}^n$. Being Q the constrained configuration space, and $T_{\boldsymbol{q}}Q$ the space of constrained generalized velocities (i.e., the configuration tangent space at $\boldsymbol{q}$). Additionally, consider a Lagrangian function $\mathcal{L} : Q \times T_{\boldsymbol{q}}Q \rightarrow \mathbb{R} = \mathcal{T}(\boldsymbol{q}, \dot{\boldsymbol{q}}) - \mathcal{U}(\boldsymbol{q}, \dot{\boldsymbol{q}})$ specifying the energy state of the system at any state. Where $\mathcal{T}(\boldsymbol{q}, \dot{\boldsymbol{q}}), \mathcal{U}(\boldsymbol{q}, \dot{\boldsymbol{q}})$ describe the state kinetic and potential energies, respectively.

The symmetries of a dynamical system are defined as transformations in the space of generalized coordinates that keep the energy of the system invariant [18, 13]. In this work, we focus on time-invariant linear transformations of generalized coordinates $\rho_Q(g) : \mathbb{R}^n \rightarrow \mathbb{R}^n$, which are the representations of actions of a symmetry group: $g \cdot \boldsymbol{q} \doteq \rho_Q(g)\boldsymbol{q} \,|\, \forall \boldsymbol{q} \in Q, g \in \mathcal{G}$. Note that because of the linearity of the transformation, the velocity and acceleration of the transformed coordinates are given by $g \cdot \dot{\boldsymbol{q}} \doteq \rho_Q(g)\dot{\boldsymbol{q}}$ and $g \cdot \ddot{\boldsymbol{q}} \doteq \rho_Q(g)\ddot{\boldsymbol{q}}$, respectively.

Formally, we say that a dynamical system has a symmetry group $\mathcal{G}$ if its Lagrangian is $\mathcal{G}$-invariant:

$$\mathcal{L}(\boldsymbol{q}, \dot{\boldsymbol{q}}) = \mathcal{L}(g \cdot \boldsymbol{q}, g \cdot \dot{\boldsymbol{q}}) \,|\, \forall g \in \mathcal{G}, \boldsymbol{q} \in Q, \dot{\boldsymbol{q}} \in T_{\boldsymbol{q}}Q. \quad (2)$$

Being g a feasible symmetry if the transformed state is a feasible state, i.e. when $g \cdot \boldsymbol{q} \in Q$ and $g \cdot \dot{\boldsymbol{q}} \in T_{\boldsymbol{q}}Q$ (assuming both Q and $T_{\boldsymbol{q}}Q$ are connected sets).

Since the Lagrangian structure differs between the original $(\boldsymbol{q}, \dot{\boldsymbol{q}})$ and transformed coordinates $(g \cdot \boldsymbol{q}, g \cdot \dot{\boldsymbol{q}}) \,|\, \forall g \in \mathcal{G}$, when we derive the Equations of Motion (EoM) of the system in the set of transformed coordinates, we obtain a set of EoMs describing the system dynamics in different coordinate systems. Formally, if we derive the EoM through the Euler-Lagrange equation of the second order $\left(\frac{d}{dt} \frac{\partial \mathcal{L}(\boldsymbol{q}, \dot{\boldsymbol{q}})}{\partial \dot{\boldsymbol{q}}} - \frac{\partial \mathcal{L}(\boldsymbol{q}, \dot{\boldsymbol{q}})}{\partial \boldsymbol{q}} \equiv \mathbf{M}(\boldsymbol{q})\ddot{\boldsymbol{q}} - \boldsymbol{\tau}(\boldsymbol{q}, \dot{\boldsymbol{q}}) = \mathbf{0} \right)$, the distinct EoM are equivariant to each other [13], a property we will refer to as dynamics $\mathcal{G}$-equivariance:

$$g \cdot \underbrace{[\mathbf{M}(\boldsymbol{q})\ddot{\boldsymbol{q}}}_{Inertial} - \underbrace{\boldsymbol{\tau}(\boldsymbol{q}, \dot{\boldsymbol{q}})]}_{Moving} = \underbrace{\mathbf{M}(g \cdot \boldsymbol{q})g \cdot \ddot{\boldsymbol{q}}}_{Inertial} - \underbrace{\boldsymbol{\tau}(g \cdot \boldsymbol{q}, g \cdot \dot{\boldsymbol{q}})}_{Moving} = \mathbf{0}$$

$$| \,\forall g \in \mathcal{G}, \boldsymbol{q} \in Q, \dot{\boldsymbol{q}} \in T_{\boldsymbol{q}}Q. \quad (3)$$

Denoting $\mathbf{M}(\boldsymbol{q}) : Q \rightarrow \mathbb{R}^{n \times n}$ as the generalized mass matrix function and $\boldsymbol{\tau}(\boldsymbol{q}, \dot{\boldsymbol{q}}) : Q \times T_{\boldsymbol{q}}Q \rightarrow \mathbb{R}^n$ as the generalized moving forces at a given state $(\boldsymbol{q}, \dot{\boldsymbol{q}})$. Note that, in eq. (3) the original and transformed dynamics are related linearly by the Jacobian of the coordinate transformation [25]. Which in this case is $\rho_Q(g)$ (reduced to g to preserve notation).

Note that to ensure dynamics $\mathcal{G}$-equivariance (eq. (3)), both the generalized inertial and moving forces need to be independently equivariant, implying:

$$\mathbf{M}(g \cdot \boldsymbol{q}) = g\mathbf{M}(\boldsymbol{q})g^{-1} \quad \wedge \quad g \cdot \boldsymbol{\tau}(\boldsymbol{q}, \dot{\boldsymbol{q}}) = \boldsymbol{\tau}(g \cdot \boldsymbol{q}, g \cdot \dot{\boldsymbol{q}})$$

$$| \,\forall g \in \mathcal{G}, \boldsymbol{q} \in Q, \dot{\boldsymbol{q}} \in T_{\boldsymbol{q}}Q. \quad (4)$$

The resultant $\mathcal{G}$-equivariance of the generalized mass matrix becomes an identifying property of symmetrical systems, providing a pathway for the identification of action representations of the symmetry group $\rho_Q(g) \,|\, g \in \mathcal{G}$ (section IV-B). While the equivariance of the generalized moving forces (which in practice usually incorporates control, constraint, and external forces) implies that dynamics $\mathcal{G}$-equivariance (eq. (3)) is upheld until a symmetry breaking force violates the equivariance of $\boldsymbol{\tau}$.

To gain some intuition, consider as an example the bipedal robot Atlas, with symmetry group $\mathcal{G} = \mathcal{C}_2 = \{e, g_s\}$. According to eq. (2) both robot states in fig. 1-left are symmetric states (related by the action g_s). Then, eq. (3) suggests that any trajectory of motion, starting from the left robot state, will be equivalent (up to transformation by g_s) to a motion trajectory starting from the right robot state, if and only if, the moving forces driving both trajectories are equivalent (up to transformation by g_s). That is if the control and external forces are $\mathcal{C}_2$-equivariant (eq. (4)). Note, we can perform a similar analysis for each symmetric state and action of systems with larger symmetry groups (e.g. Solo in fig. 1-center).

Floating-base dynamical systems: All robotic, biological, and virtual systems that move in a Euclidean space of d dimensions, can be modeled as floating-base dynamical systems. Hence, without loss in generality, we assume the system's configuration space can be decoupled into $Q \doteq \mathbb{E}_d \times Q_J$, being $\mathbb{E}_d$ the space of all possible base configurations (all rotations/reflections and translations), and Q_J the joint-space (or internal configuration space). Resulting in the decoupling $\boldsymbol{q} = \begin{bmatrix} \boldsymbol{X}_B \\ \hat{\boldsymbol{q}} \end{bmatrix} \begin{smallmatrix} \in \mathbb{E}_d \\ \in Q_J \end{smallmatrix}$. Where $\boldsymbol{X}_B$ is a homogenous matrix describing the base position and orientation in d dimensions [3], and $\hat{\boldsymbol{q}} \subseteq \mathbb{R}^{n_J}$ represents the internal Degrees of Freedom (DoF) configuration. This separation of the configuration space becomes useful to study the effect of a symmetry transformation, since, we decouple the effect of the symmetry actions $g \cdot \boldsymbol{q} = \rho_Q(g) \, \boldsymbol{q} = \begin{bmatrix} \rho_{\mathbb{E}_d}(g) & \mathbf{0} \\ \mathbf{0} & \rho_{Q_J}(g) \end{bmatrix} \begin{bmatrix} \boldsymbol{X}_B \\ \hat{\boldsymbol{q}} \end{bmatrix} \,|\, \forall g \in \mathcal{G}$. With $\rho_{\mathbb{E}_d}(g) \in \mathbb{E}_d$ and $\rho_{Q_J}(g) \in \mathbb{R}^{n_J \times n_J}$ being representations of how action g transforms the base and joint-space configuration.

Symmetries due to Euclidean isometries: floating-base systems are known for having symmetries to (some) translations,

---

[3] We use the homogeneous matrix representation of $\boldsymbol{X}_B$ instead of a vector-quaternion representation, with some abuse of notation.

rotations, and reflections of space (i.e. *Euclidean isometries*). Giving origin to the conservation of linear/angular momentum, in conservative systems[15]. We can understand these as:

Definition 1 (Symmetry due to Euclidean isometries). A floating-base system with generalized coordinates $\boldsymbol{q} \in \mathrm{Q} \doteq \mathbb{E}_d \times \mathrm{Q}_J$, is said to be symmetric w.r.t a set of Euclidean isometries $\overline{g} \in \overline{\mathcal{G}} \subseteq \mathbb{E}_d$ (involving a true rotation, reflection or translation in space), if eq. (2) holds for $\overline{\mathcal{G}}$.

Because rotations, reflections, and translations of space preserve the mass and inertia of bodies, as well as distances between them, these symmetries leave both the joint-space configuration and the generalized mass matrix invariant: $\rho_{\mathrm{Q}_J}(\overline{g}) = \boldsymbol{I}_{n_J}$ and $\mathbf{M}(\overline{g} \cdot \boldsymbol{q}) = \mathbf{M}(\boldsymbol{q}) \mid \forall \, \overline{g} \in \overline{\mathcal{G}} \subseteq \mathbb{E}_d$.

IV. DISCRETE MORPHOLOGICAL SYMMETRIES (DMSS)

A dynamical system is said to possess a DMS if it can imitate the effects of a rotation, reflection, or translation in space through a feasible discrete change in its configuration. To gain some intuition, before introducing a formal definition, we can analyze the simplest and most common DMS.

Reflection DMS: Although most floating-base dynamical systems are symmetric with respect to reflections of space (definition 1), these symmetries are infeasible due to the impossibility to execute reflections in the real-world [21]. However, systems with sagittal symmetry (e.g., Atlas in fig. 1-left, or humans) can imitate the effect of a reflection with a feasible discrete change in their configuration, by rotating their body and modifying their limbs' pose. These systems share the same symmetry group, the reflection group $\mathcal{G} \equiv \mathcal{C}_2$.

Multiple DMSs: This property can be extended to the case of a floating-base system having multiple DMSs, allowing it to imitate multiple distinct Euclidean isometries. Most frequently systems can imitate a set of rotations and reflections, making $\mathcal{G}$ a Cyclic $\mathcal{C}_k$ or Dihedral $\mathcal{D}_{2k}$ group. See examples for $\mathcal{C}_3$ in fig. 6, and for $\mathcal{D}_4 \equiv \mathcal{K}_4$ in fig. 1-center.

We can formalize this property with as:

Definition 2 (Discrete morphological symmetry). A floating-base dynamical system with generalized coordinates $\boldsymbol{q} \in \mathrm{Q} \doteq \mathbb{E}_d \times \mathrm{Q}_J$, is said to have a DMS if, for a given Euclidean isometry $\overline{g} \in \mathbb{E}_d$, there exists a feasible action $g \in \mathcal{G}$ with a non-trivial representation in joint-space ($\rho_{\mathrm{Q}_J}(g) \neq \boldsymbol{I}_{n_J}$), such that both g and $\overline{g}$ are equivalent symmetries of the system:

$$\mathcal{L}(\boldsymbol{q}, \dot{\boldsymbol{q}}) = \mathcal{L}(\overline{g} \cdot \boldsymbol{q}, \overline{g} \cdot \dot{\boldsymbol{q}}) = \mathcal{L}(g \cdot \boldsymbol{q}, g \cdot \dot{\boldsymbol{q}})$$
$$\mid g \in \mathcal{G}, \; \overline{g} \in \mathbb{E}_d, \; \forall \, g \cdot \boldsymbol{q}, \boldsymbol{q} \in \mathrm{Q}, \; g \cdot \dot{\boldsymbol{q}}, \dot{\boldsymbol{q}} \in \mathrm{T}_{\boldsymbol{q}}\mathrm{Q}. \quad (5)$$

The set of DMSs of the system forms its symmetry group $\mathcal{G}$. Because each DMS is related with a system's symmetry $\overline{g}$ due to a Euclidean isometry (definition 1), the group $\mathcal{G}$ is isomorphic to a subset of the Euclidean isometries of the system.

Hence, after identifying a potential Euclidean isometry to imitate $\overline{g} \in \mathbb{E}_d$, we can determine the DMS representation,

considering that in any system state:

$$\mathcal{L}\left(\begin{bmatrix} \rho_{\mathbb{E}_d}(\overline{g})\boldsymbol{X}_B \\ \hat{\boldsymbol{q}} \end{bmatrix}, \begin{bmatrix} \rho_{\mathbb{E}_d}(\overline{g})\dot{\boldsymbol{X}}_B \\ \dot{\hat{\boldsymbol{q}}} \end{bmatrix}\right) = \mathcal{L}\left(\begin{bmatrix} \rho_{\mathbb{E}_d}(g)\boldsymbol{X}_B \\ \rho_{\mathrm{Q}_J}(g) \; \hat{\boldsymbol{q}} \end{bmatrix}, \begin{bmatrix} \rho_{\mathbb{E}_d}(g)\dot{\boldsymbol{X}}_B \\ \rho_{\mathrm{Q}_J}(g) \; \dot{\hat{\boldsymbol{q}}} \end{bmatrix}\right)$$

$$\left| \quad |\rho_{\mathbb{E}_d}(\overline{g})| = \pm 1, \; |\rho_{\mathbb{E}_d}(g)| = 1 \right.$$
$$\rho_{\mathbb{E}_d}(g)\boldsymbol{X} = \rho_{\mathbb{E}_d}(\overline{g})\boldsymbol{X}\rho_{\mathbb{E}_d}(\overline{g})^{\text{-}1} \quad (6)$$

Where the existence of the DMSs is subjected to the system's generalized mass matrix being $\mathcal{G}$-equivariant (eq. (4)), and to the transformation $\rho_{\mathbb{E}_d}(g)$ (defined through group conjugation) being proper/feasible. In practice, these restrictions represent a pathway for the identification of $\mathcal{G}$ for any floating-base system (section IV-C).

A. Data augmentation in systems with DMS

Recall from section III that due to the linearity, the action representation $\rho_{\mathrm{Q}}(g)$ acts on elements of configuration space Q, configuration tangent space $\mathrm{T}_{\boldsymbol{q}}\mathrm{Q}$ and any higher order tangent spaces, including the spaces of generalized accelerations and forces $g \cdot \boldsymbol{\tau} = \rho_{\mathrm{Q}}(g)\boldsymbol{\tau}$ (eq. (3)). Since for floating-base systems $\mathrm{Q} \doteq \mathbb{E}_d \times \mathrm{Q}_J$, this property translates to the action representations on $\mathbb{E}_d$ and Q_J. This effectively implies that $\rho_{\mathbb{E}_d}(g)$ can be used to augment any point, vector, and orientation in $\mathbb{E}_d$ and in $\mathbb{E}_d$ higher order tangent spaces (e.g. locations of tactile sensing, linear & angular velocities/accelerations, depth maps, external forces, terrain height-maps). Likewise the representation $\rho_{\mathrm{Q}_J}(g)$ can be used to augment members of Q_J and its higher order tangent spaces (e.g. joints positions/velocities/accelerations/torques).

In practice, this means that any proprioceptive and extero-ceptive measurements relevant to the evolution of the system's dynamics can be augmented solely with combinations of $\rho_{\mathbb{E}_d}(\overline{g})$, $\rho_{\mathbb{E}_d}(g)$ and $\rho_{\mathrm{Q}_J}(g)$. Since these measurements consist of elements in Q_J, $\mathbb{E}_d$, and their higher order tangent spaces (see examples in sections C-C1 and C-D1). Furthermore, for any data point, there exist $|\mathcal{G}|$ symmetric data points (being $|\mathcal{G}|$ the order of the symmetry group). Therefore, for a system with a symmetry group of order, say, $|\mathcal{G}| = 4$ (as in fig. 1-center), we can obtain an additional 3 minutes of recordings for every minute of recorded data simply by considering the symmetric states of the data.

To exploit the symmetries in the measurements, we first need to identify the joint-space representations $\rho_{\mathrm{Q}_J}(g)$, which requires additional assumptions about the system's dynamics. In this work, we focus on the case of rigid-body dynamics, although a similar analysis can be extended to other types of systems, such as soft robots.

B. DMS in the case of rigid-body dynamics

Consider dynamical systems composed of n_B intercon-nected rigid bodies evolving in $\mathbb{E}_d$. This is the usual scenario in robotics, computer graphics, and computational biology.

The kinetic energy for these systems is determined by $\mathcal{T}(\boldsymbol{q}, \dot{\boldsymbol{q}}) = \frac{1}{2} \sum_k^{n_B} m_k \dot{\mathbf{r}}_k^2 + \boldsymbol{w}_k^{\mathsf{T}} \mathbf{I}_k \boldsymbol{w}_k = \frac{1}{2} \dot{\boldsymbol{q}}^{\mathsf{T}} \mathbf{M}(\boldsymbol{q}) \dot{\boldsymbol{q}}$. Being m_k, $\mathbf{I}_k$, $\dot{\mathbf{r}}_k$ and $\boldsymbol{w}_k$ the mass, inertia, linear velocity, and angular velocity of body k. Considering that the energy-preservation property of symmetries (*eq.* (2)) is dependent

solely on the $\mathcal{G}$-equivariance of $\mathbf{M}(\boldsymbol{q})$ (eq. (4)), we can assert the existence of DMSs by analyzing $\mathbf{M}(\boldsymbol{q})$. The generalized mass matrix is given by $\mathbf{M}(\boldsymbol{q}) = \sum_k^{n_B} \mathbf{J}_{T_k}(\boldsymbol{q})^\mathsf{T} m_k \mathbf{J}_{T_k}(\boldsymbol{q}) + \mathbf{J}_{R_k}(\boldsymbol{q})^\mathsf{T} \mathbf{I}_k \mathbf{J}_{R_k}(\boldsymbol{q})$, being $\mathbf{J}_{T_k}(\boldsymbol{q}) : \mathbf{Q} \to \mathbb{R}^{d \times n}$ and $\mathbf{J}_{R_k}(\boldsymbol{q}) : \mathbf{Q} \to \mathbb{R}^{d \times n}$ the position and orientation Jacobians, used to map generalized velocities to the linear ($\dot{\mathbf{r}}_k = \mathbf{J}_{T_k}(\boldsymbol{q})\dot{\boldsymbol{q}}$) and angular ($\boldsymbol{w}_k = \mathbf{J}_{R_k}(\boldsymbol{q})\dot{\boldsymbol{q}}$) velocities of the body k [26].

These Jacobians are functions of the kinematic parameters of the system[4]. While the mass and inertia of all bodies are the system's dynamic parameters. A DMS implies symmetries over both kinematic and dynamic parameters.

Symmetries of kinematic parameters (Kinematic Tree): The symmetry in kinematic parameters can be thought of as a kinematic tree symmetry. I.e., the DMS g must transform the system state in a way that produces a kinematic tree indistinguishable from the one obtained by applying the Euclidean isometry $\overline{g}$. Ignoring the dynamic parameters makes it easier to see this.

Consider that applying the Euclidean isometry $\overline{g}$ conserves kinetic energy as velocity vectors are only rotated or reflected. Hence, for g to imitate the effect of $\overline{g}$, the velocity of the kth body after applying $\overline{g}$ must be equal to the velocity of the ith body after applying g. In other words, $\overline{g} \cdot \dot{\mathbf{r}}_k = g \cdot \dot{\mathbf{r}}_i \doteq \mathbf{J}_{T_i}(g \cdot \boldsymbol{q}) g \cdot \dot{\boldsymbol{q}}$ and $\overline{g} \cdot \boldsymbol{w}_k = g \cdot \boldsymbol{w}_i \doteq \mathbf{J}_{R_i}(g \cdot \boldsymbol{q}) g \cdot \dot{\boldsymbol{q}} \mid \forall \, \{(k,i)|k, i \in [n_B]\}$. This results in the following position Jacobian constraints:

$$\mathbf{J}_{T_i}(g \cdot \boldsymbol{q})g = \overline{g} \cdot \mathbf{J}_{T_k}(\boldsymbol{q}) \mid \forall \boldsymbol{q} \in \mathbf{Q}, \{(k,i)|k, i \in [n_B]\}$$
$$\mathbf{J}_{T_i}(\rho_\mathbf{Q}(g)\boldsymbol{q})\rho_\mathbf{Q}(g) = \rho_{\mathbb{E}_d}(\overline{g}) \cdot \mathbf{J}_{T_k}(q). \tag{7}$$

Analog constraints apply to the rotational Jacobian.

Equation 7 specifies the kinematic parameter constraints required for g to be a DMS, ensuring the g-equivariance of $\mathbf{M}(\boldsymbol{q})$. Note that If $k \neq i$, the representation $\rho_{\mathbf{Q}_J}(g)$ entails a permutation of the i and k joint-space configurations. For example, g_s swaps the left and right leg configurations of Atlas and Solo in Figure 1.

Symmetries of dynamic parameters (Mass & Inertia): In addition to the kinematic tree symmetry of eq. (7), the mass, CoM, and inertia of the bodies i and k must be equivalent for g to preserve kinetic energy. To understand this morphological constraint, consider how, in eq. (6), the base body configuration $\boldsymbol{X}_B \in \mathbb{SE}_d$ is transformed by the DMS $\rho_{\mathbb{E}_d}(g) \in \mathbb{SE}_d$, and by the Euclidean isometry $\rho_{\mathbb{E}_d}(\overline{g}) \in \mathbb{E}_d$. For both base configurations to have the same dynamics, their CoM must coincide, and crucially, the reflected Inertia matrices $\mathbf{I}_B$ need to be identical. This constraint is satisfied if the reflected Inertia is invariant to the transformation $\boldsymbol{X}_B = \boldsymbol{X}_B \rho_{\mathbb{E}_d}(\overline{g})^{-1}$. In practice, this invariance implies a symmetric mass distribution of the rigid body (see geometric proof in section A-B).

Let's consider the robot Solo in fig. 1-center as an example. It can imitate two reflections of space ($\overline{g}_t$, $\overline{g}_s$) and a 180° rotation $\overline{g}_r$. The existence of the robot's DMSs can be attributed to two factors. First, the base body of Solo

[4]The Denavit–Hartenberg parameters are a common convention of kinematic parameters adopted in robotics and computer graphics.
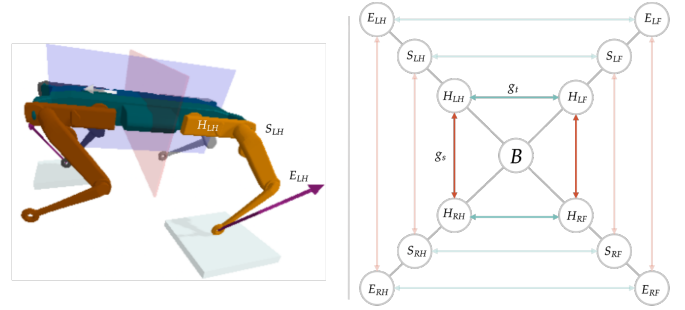

Fig. 2: **Left:** Solo sagittal (blue) and transversal (red) symmetry planes of the base body. **Right:** Solo's kinematic tree, and permutation symmetries of the legs/tree-branches.

possesses two symmetry planes (fig. 2), leading to symmetric mass distributions and the invariance of the body's reflected inertia under the transformation $\boldsymbol{X}_B \rho_{\mathbb{E}_d}(\overline{g})^{-1} \mid \overline{g} \in \mathcal{K}_4$. Second, the modularity of the kinematic tree. This arises from all four legs (tree branches) being structurally identical, resulting in the tree branches having equivalent inertial and kinematic parameters (see fig. 2). Consequently, we can define each $\rho_{\mathbf{Q}_J}(g)$ as a permutation of the leg configurations. For instance, for g_s, the configurations of the left and right legs are interchanged, while for g_t the front and back legs' configuration is interchanged. It is important to note that this interchange of leg configurations would violate eq. (2) if the legs were not composed of identical/reflected rigid bodies (i.e, having different mass distributions) and would violate eq. (3) if the dynamics and constraints of the leg joints differed (e.g., different position/velocity/torque limits).

C. Identification of DMS group $\mathcal{G}$ in rigid-body dynamics

The identification of the DMS group $\mathcal{G}$ in a floating-base dynamical system composed of rigid bodies can be achieved through the following four steps (see tutorial examples in fig. 6):

1) Identify all the unique bodies in the kinematic tree, including the base of the system, ensuring there are no duplicated or reflected versions of the same body.
2) Determine the set of Euclidean isometries $\overline{g} \in \mathbb{E}_d$ for which the inertia of each unique body remains invariant. These candidate Euclidean isometries represent potential DMSs for the system.
3) Identify any modularity in the kinematic tree, such as sets of duplicated or reflected symmetric kinematic subchains.
4) Utilize eq. (7) to assess the feasibility and existence of $\rho_{\mathbf{Q}_J}(g) \in \mathcal{G}$, progressing from the base to the end-effectors.

While the presented analysis may seem extensive for simple systems and DMS groups, these abstractions open avenues for studying more complex systems and symmetry groups, enabling computational design of symmetric robotic systems, as well as algorithmic identification of DMSs and data augmentation.

V. $\mathcal{G}$-EQUIVARIANT FUNCTION APPROXIMATORS

After identifying the DMS group $\mathcal{G}$ of our system (section IV-C) and understanding how these symmetries manifest in proprioceptive and exteroceptive measurements (section IV-A), we can now exploit these symmetries in our data. Consider the symmetric input $\mathcal{X}$ and output $\mathcal{Y}$ vector spaces, of any $\mathcal{G}$-equivariant/invariant (eq. (1)) function $f : \mathcal{X} \to \mathcal{Y}$, that we desire to approximate with a model $\hat{f}$. We assume that $\mathcal{X}$ and $\mathcal{Y}$ are symmetric spaces as they are composed of (potentially several) proprioceptive/exteroceptive measurements. For instance, both spaces could contain measurements of the system state, terrain elevation, external forces, depth images, contact states, etc. See section VI for examples.

To enhance the generalization and sample efficiency of our approximation, we can enforce the $\mathcal{G}$-equivariant/invariant constraints of the original function on our model $\hat{f}$. In this section, we outline the process of incorporating these constraints when $\hat{f}$ is a neural network parameterized by ϕ. By imposing symmetry constraints, we can reduce the number of trainable parameters in the architecture. Our approach builds upon the theory and implementation of Finzi et al. [6]'s framework for $\mathcal{G}$-equivariant NNs. Our main motivation is to overcome the limitations that hinder the construction of large-scale $\mathcal{G}$-equivariant NNs, which are commonly encountered in real-life applications (see details in Section B).

Consider c to be composed of multiple perceptrons (or convolutional) layers of the form ${}^l\boldsymbol{y} := \sigma({}^l\boldsymbol{W}^l\boldsymbol{x} + {}^l\boldsymbol{b})$, where ${}^l\boldsymbol{x} \in \mathbb{R}^n$, ${}^l\boldsymbol{y} \in \mathbb{R}^m$ are the lth layer input-output. ${}^l\boldsymbol{W} \in \mathbb{R}^{m \times n}$ and ${}^l\boldsymbol{b}$ are the layer's linear map and bias; and $\sigma : \mathbb{R} \to \mathbb{R}$ is a strictly monotonic nonlinearity [20]. With this parametrization, the equivariance constraints of eq. (1) can be reduced to constraints on the linear map $\boldsymbol{W}$ (dropping the layer index l for notation clarity):[5]

$$\rho_{out}(g)\boldsymbol{W} = \boldsymbol{W}\rho_{in}(g) \iff (\rho_{\boldsymbol{W}}(g) - \boldsymbol{I})\boldsymbol{w} = \boldsymbol{0} \quad | \,\forall\, g \in \mathcal{G}. \quad (8)$$

The right-side of eq. (8) is a reformulation of the linear map equivariance constraints (left-side) as a standard set of linear equations. Denoting $\boldsymbol{w} = vec(\boldsymbol{W}) \in \mathbb{R}^{mn}$ as a vectorized version of $\boldsymbol{W}$ and $\rho_{\boldsymbol{W}}(g) = \rho_{out}(g) \otimes \rho_{in}(g^{-1})^\mathsf{T} \in \mathbb{R}^{mn \times mn}$ as the action representation acting on the parameter space of the linear map ($\otimes$ stands for the Kronecker product). Here, we consider the group acting on $\boldsymbol{W}$ a semi-direct product[6] of the input and output groups (refer to Finzi et al. [6] for details). Since the constraint imposed by each g is linear in $\boldsymbol{W}$, we can stack them into a single large system of linear equations $\boldsymbol{Cw} = \boldsymbol{0}$. The nullspace of this system of equations $\boldsymbol{B} \in \mathbb{R}^{mn \times r}$ describes the r basis vectors spawning the entire space of equivariant linear maps. Allowing to parameterize all

---

[5]A similar analysis can be made for the bias vector $\boldsymbol{b}$.

[6] Since with DMSs the input and output symmetry groups are isomorphic, using a direct product in eq. (8) implies an over-constraining of the linear map. Resulting in an excessive reduction in the number of trainable parameters.

$\mathcal{G}$-equivariant $\boldsymbol{W}$ as:

$$\boldsymbol{w} = \sum_k^r c_k \boldsymbol{B}_{:,k} \iff \begin{aligned} \boldsymbol{W} &= \sum_k^r c_k \, \text{unvec}(\boldsymbol{B}_{:,k}) \\ &\doteq \sum_k^r c_k \mathsf{B}_{:,:,k}. \end{aligned} \quad (9)$$

Where the basis coefficients $\boldsymbol{c} \in \mathbb{R}^r$ represent the free variables of the system of equations and the trainable parameters of the equivariant layer (see fig. 1 right).

A. Dealing with memory complexity of equivariant layers:

An equivariant layer needs to store the matrices $\rho_{\boldsymbol{W}}(g) \in \mathbb{R}^{mn \times mn}$ and $\boldsymbol{B} \in \mathbb{R}^{mn \times r}$, in addition to the typical memory complexity of a perceptron or convolutional layer. These matrices' memory complexity quickly becomes intractable for moderate input-output dimensions (see section C). Fortunately, the symmetry groups of DMSs (finite groups) have sparse action matrix representations, resulting in both of the aforementioned matrices being highly sparse. We extend the API from Finzi et al. [6] to handle sparse matrix representations, limiting the additional memory footprint to a minimum.

B. Dealing with the computational complexity of determining the equivariant basis $\boldsymbol{B}$

Computing $\boldsymbol{B}$ amounts to finding the nullspace of a large linear system of equations. Finzi et al. [6] proposes a Krylov gradient-based method able to handle both finite and Lie groups' arbitrary representations. While Van der Pol et al. [23] approximates $\boldsymbol{B}$ through SVD of a matrix $\bar{\boldsymbol{W}} \in \mathbb{R}^{z \times mn}$ ($z \geq mn$). Both approaches run in polynomial time $\mathcal{O}(r^2(mn)^2)$ and approximate the space rank r numerically. These approaches become intractable for large vector spaces.

To handle the computational cost we exploit the fact that for DMSs all $\rho_{\mathbb{Q}_J}$ and $\rho_{\mathcal{Y}_l}$ (representations of the output space of internal layers of the NN), can be expressed as permutation representations. Reducing the computation of the nullspace $\boldsymbol{B}$ to a search of the permutations (or orbits) of each dimension of $\boldsymbol{w}$. A problem that can be solved in linear time.

For permutation representations, the constraints imposed by each $\rho_{\boldsymbol{W}}(g)$ on $\boldsymbol{w}$ can be interpreted as parameter sharing constraints. Becoming every unique orbit of the dimensions of the linear map $\mathcal{G} \cdot w_k = \{g \cdot w_k : \forall\, g \in \mathcal{G}\} = \{w_k, -w_i, \dots, w_j\}$, a vector of the null-space of $\boldsymbol{C}$ (i.e., $\boldsymbol{B}_i$). Each unique orbit describes the sharing scheme of a free variable of the system of equations, that is, the sharing of the trainable parameter c_i over multiple positions in $\boldsymbol{W}$ (see the parameter orbits of length 4 in fig. 1-right, for $\mathcal{K}_4$). The orbits of all $w \in \boldsymbol{w}$ are trivially computed with $[\boldsymbol{w}, \rho_{\boldsymbol{W}}(g_1)\boldsymbol{w}, \dots, \rho_{\boldsymbol{W}}(g_{|\mathcal{G}|})\boldsymbol{w}]$, while the unique r orbits can be identified in $\mathcal{O}(mn)$ time. Our proposed solution can be thought of as a linear-time version of Ravanbakhsh et al. [20].

C. Optimal parameter initialization for equivariant layers

Proper initialization of the equivariant layer's trainable parameters $\boldsymbol{c}_l$ (eq. (9)) is required to avoid activations/gradients from vanishing or exploding [12]. Following the same derivation of the Kaiming initialization [10] (see section B-B), we can conclude that the parameters should be initially sampled
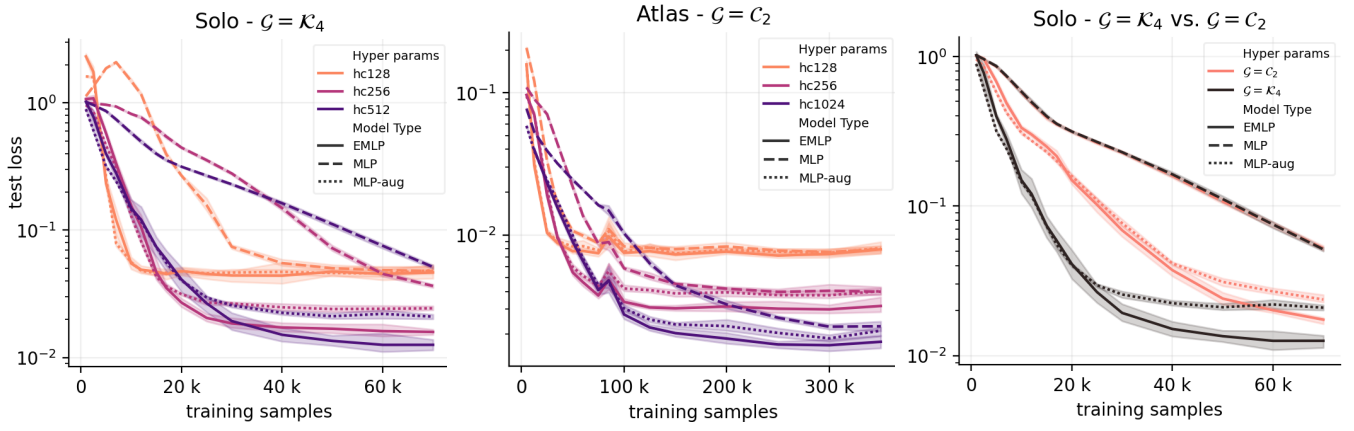
Fig. 3: **CoM-estimation results comparing MLP, MLP-aug, and EMLP models**. **Left and Middle:** Test set sample efficiency of model variants with different capacities (number of neurons hc in hidden layers) for robot Solo and Atlas. **Right:** Sample efficiency for robot Solo with models having $hc = 512$, when exploiting $\mathcal{G} = \mathcal{K}_4$ (sagittal and traversal symmetries) and $\mathcal{G} = \mathcal{C}_2 = e, g_s \subset \mathcal{K}_4$ (only sagittal symmetry). The plots depict the average and standard deviation across 10 seeds.

from a distribution with $\text{Var}(\boldsymbol{c}_l) = {}^m/\lambda_{\mathbf{B}}\gamma_\sigma$, to ensure constant variance of activations throughout the network layers (see fig. 7). Where $\lambda_{\mathbf{B}} = \sum_i^m \sum_j^n \sum_k^r \mathbf{B}_{i:j:k}^2$ and γ_σ is a nonlinearity dependant scalar (e.g., $\gamma_{\text{ReLu}} = {}^1/2$, $\gamma_{\text{SeLu}} = 1$ following Klambauer et al. [12]). This initialization depends only on $\mathbf{B}$. Thus, is applicable for any symmetry group.

D. Reduction of trainable parameters in equivariant layers:

Determining analytically the number of trainable parameters (i.e. the rank r) of an $\mathcal{G}$-equivariant layer is, in general, an unresolved problem. However, for DMS groups and permutation representations, it becomes trivial to show that the number of trainable parameters of a $\mathcal{G}$-equivariant layer can range from $|\boldsymbol{w}|/|\mathcal{G}| \leq r \leq |\boldsymbol{w}|$, depending on the number of dimensions of the input-output spaces left invariant by the symmetry actions (see details in section B-A). In practice, this implies that for a $\mathcal{G}$-equivariant layer without any input-output fixed points (e.g., all intermediate layers of a $\mathcal{G}$-equivariant NN), the number of trainable parameters is reduced by ${}^1/|\mathcal{G}|$ being $|\mathcal{G}|$ the group order. Therefore a $\mathcal{G}$-equivariant architecture with $\mathcal{G} = \mathcal{C}_2$ (or $\mathcal{G} = \mathcal{K}_4$) will have approximately ${}^1/2$ (or ${}^1/4$) of the trainable parameters of an unconstrained NN of the same architectural size (this applies to NN processing data from robot Atlas and Solo fig. 1). The reduction of parameters is caused by the parameter sharing constraints (eq. (9)) and is visually depicted in fig. 1-right.

VI. EXPERIMENTS

We demonstrate the effectiveness of DMSs for data augmentation and training equivariant functions through two supervised learning experiments: a regression task using synthetic data and a classification task using real-world data. These experiments showcase the impact of exploiting DMSs on sample efficiency and generalization capacity. While we keep the presentation concise, all the technical aspects are detailed in section C and the code repository [1].

A. CoM momentum estimation (Regression)

In this experiment, we train a NN to approximate a robot's center-of-mass momentum given the joint-space position and velocities: $\boldsymbol{h} = \mathbf{A}_G(\hat{\boldsymbol{q}})\dot{\hat{\boldsymbol{q}}}$, where $\boldsymbol{h} = [\boldsymbol{l}^\intercal \ \mathbf{k}^\intercal]^\intercal$ are the linear $\boldsymbol{l}$ and angular $\mathbf{k}$ momentum components and $\mathbf{A}_G$ is the Centroidal Momentum Matrix (CMM) of Orin et al. [17]. This analytical function is highly non-linear and $\mathcal{G}$-equivariant w.r.t the robot's symmetry group $\mathcal{G}$ (eqs. (18) and (20)).

We test two robots: Atlas, a 32-DoF humanoid robot with $\mathcal{G} = \mathcal{C}_2$ sagittal reflection symmetry (fig. 1-left), and Solo, a 12-DoF quadruped robot with $\mathcal{G} = \mathcal{K}_4$ (fig. 1-center). We compare tree function approximation variants: a standard Multi-Layer Perceptron (MLP), an augmented MLP s(MLP-aug), and a hard-equivariant MLP (E-MLP).

In fig. 3-left-&-middle, we compare the model variants. Across both robots and all model capacities, E-MLP and MLP-Aug outperform MLP in terms of sample efficiency (better generalization with fewer data) and robustness to overfitting when training data is limited. Among the E-MLP and MLP-Aug variants, lower capacity versions exhibit similar behavior, but as capacity increases, E-MLP demonstrates superior sample efficiency and generalization. In addition, fig. 3-right shows a comparison for the Solo robot, evaluating the performance of the model variants when exploiting either the entire symmetry group ($\mathcal{K}_4$) or a subgroup of the true symmetry group ($\mathcal{C}_2 \subset \mathcal{K}_4$). The results indicate that sample efficiency and generalization capacity increase with the number of true symmetries of the data exploited.

B. Static-friction-regime contact detection (Classification)

In this experiment, we utilize the dataset introduced in Lin et al. [14] for estimating static-friction-regime contacts in the foots of the Mini-Cheetah quadruped robot. The dataset consists of real-world proprioceptive data ($\hat{\boldsymbol{q}}$, $\dot{\hat{\boldsymbol{q}}}$, base linear acceleration, base angular velocity, and leg feet positions and velocities) captured over a history of 150 time-frames. These measurements were obtained from inboard sensors during
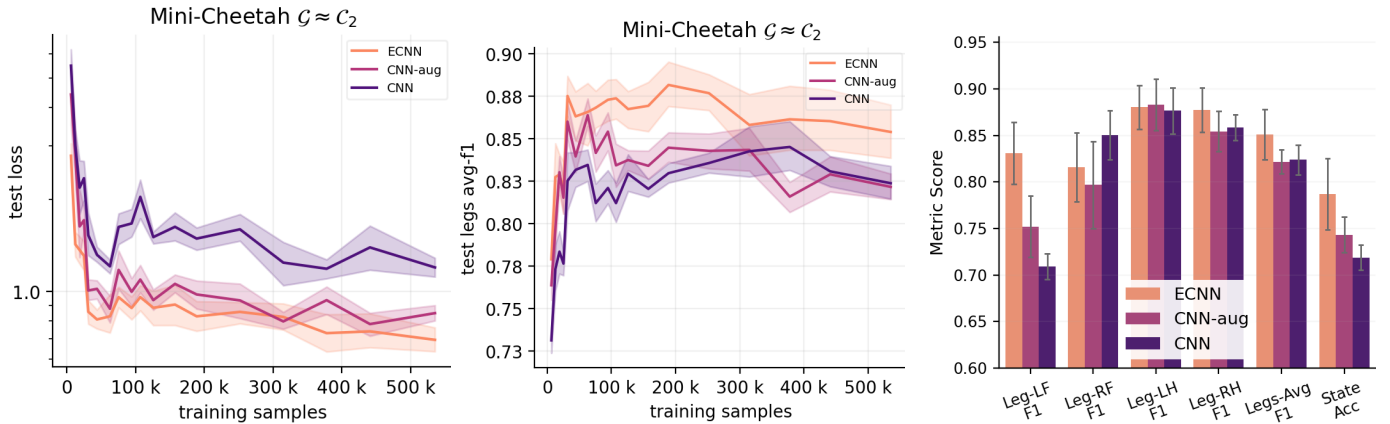
Fig. 4: **Static-Friction-Regime contact detection results comparing CNN, CNN-aug, and ECNN. Left:** Sample efficiency in log-log scale. **Middle:** Average legs F1-score. **Right:** Classification metrics on test set performance of models trained with the entire training set. The selected metrics include contact-state ($y \in \mathbb{R}^{16}$) accuracy (Acc) and f1-score (F1) for each leg binary contact state. Due to the sagittal symmetry of the robot, the left front (LF) and right front (RF) legs are expected to be symmetric, as well as the left hind (LH) and right hind (RH) legs. F1-score is presented considering the dataset class imbalance (see section C-D). The reported values represent the average and standard deviation across 8 different seeds.

locomotion, encompassing various gaits and terrains. The dataset also includes $y \in \mathbb{R}^{16}$, representing the ground truth contact state of the robot, which was estimated offline using a non-causal algorithm. Our goal is to train a causal function approximator $\hat{f}(x; \phi)$ to predict the contact state based on the input proprioceptive data.

The Mini-Cheetah robot in the real-world exhibits an approximate reflection symmetry group, $\mathcal{G} \approx \mathcal{C}_2$. As a result, both the proprioceptive data x and the contact state y share the symmetry group $\mathcal{G}$ (see section C-D). In this experiment, we compare three variants of function approximators: the original Convolutional Neural Network architecture proposed by Lin et al. [14] (CNN), a version of CNN trained with data augmentation (CNN-aug), and a version of CNN that incorporates hard-equivariance constraints (E-CNN).

The sampling efficiency and average leg contact state classification results are depicted in fig. 4-left-&-middle. The equivariant model, E-CNN, demonstrates superior generalization performance and robustness to dataset biases compared to the unconstrained models (refer to section C-D2). Following E-CNN, CNN-aug exhibits better performance than the original CNN. In fig. 4-right, we evaluate the classification metrics of the test set when using the entire training data. The E-CNN model outperforms both CNN-aug and CNN in contact state classification and average leg contact detection. Notably, exploiting symmetries helps mitigate suboptimal asymmetries in the models, preventing them from favoring the classification of one leg over others (observe legs LF and RF in fig. 4-right). Further details can be found in section C-E.

VII. Conclusions & Discussion

In this work, we present the definition of Discrete Morphological Symmetry (DMS). A capability of some dynamical systems to imitate the effect of rotations, translations, and infeasible reflections of space with a feasible discrete change in the system configuration. Using the language of group theory we study the set of DMSs of a dynamical system as a symmetry group $\mathcal{G}$ and conclude that: (1) A system with a symmetry group $\mathcal{G}$ exhibits $\mathcal{G}$-equivariant generalized mass matrix and dynamics. (2) That the symmetries of the dynamics extend to optimal control policies as well as to any proprioceptive and exteroceptive measurements, related to the evolution of the system's dynamics.

We establish the necessary theoretical abstractions to investigate and identify DMSs in any dynamical system, irrespective of the number of symmetries present. This new formalism allows us to identify the reflection/sagittal symmetry, prevalent in humans, animals, and most robots, as the simplest morphological symmetry group $\mathcal{G} = \mathcal{C}_2$. Crucially, we use the same formalism to identify and exploit DMSs in real-world robotic systems with a greater number of symmetries.

In addition, we provide an open-access repository that facilitates the efficient prototyping of $\mathcal{G}$-equivariant neural networks for exploiting DMS in various applications involving rigid-body dynamics, such as robotics, computer graphics, and computational biology. This repository includes a growing collection of symmetric dynamical systems, with their corresponding symmetry groups already identified. Furthermore, we present compelling empirical and theoretical evidence supporting the utilization of DMSs in data-driven applications through data augmentation and the adoption of $\mathcal{G}$-equivariant neural networks. Both symmetry exploitation techniques result in improved sample efficiency and generalization.

Limitations: Our work makes two assumptions: (1) that the system symmetry group $\mathcal{G}$ is finite, and (2) that the symmetries of the system are exact. For details see section D.

Further work: For data-driven applications the benefits of DMSs suggest the computational design of symmetrical dynamical systems. While for control applications, the $\mathcal{G}$-equivariance nature of the generalized mass matrix suggests research on the numerical implications of this previously unexploited constraint in optimal control.

REFERENCES

[1] Farzad Abdolhosseini, Hung Yu Ling, Zhaoming Xie, Xue Bin Peng, and Michiel Van de Panne. On learning symmetric locomotion. In *Motion, Interaction and Games*, pages 1–10. 2019.

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

[3] 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 *2019 IEEE/SICE International Symposium on System Integration (SII)*, pages 614–619. IEEE, 2019.

[4] Nathan Carter. *Visual group theory*, volume 32. American Mathematical Soc., 2021.

[5] Marc Finzi, Gregory Benton, and Andrew G Wilson. Residual pathway priors for soft equivariance constraints. *Advances in Neural Information Processing Systems*, 34: 30037–30049, 2021.

[6] Marc Finzi, Max Welling, and Andrew Gordon Wilson. A practical method for constructing equivariant multilayer perceptrons for arbitrary matrix groups. In *International Conference on Machine Learning*, pages 3318–3328. PMLR, 2021.

[7] Niklas Funk, Charles Schaff, Rishabh Madan, Takuma Yoneda, Julen Urain De Jesus, Joe Watson, Ethan K Gordon, Felix Widmaier, Stefan Bauer, Siddhartha S Srinivasa, et al. Benchmarking structured policies and policy optimization for real-world dexterous object manipulation. *IEEE Robotics and Automation Letters*, 7(1): 478–485, 2021.

[8] Xavier Glorot and Yoshua Bengio. Understanding the difficulty of training deep feedforward neural networks. In *Proceedings of the thirteenth international conference on artificial intelligence and statistics*, pages 249–256. JMLR Workshop and Conference Proceedings, 2010.

[9] Kaveh Akbari Hamed and Jessy W Grizzle. Event-based stabilization of periodic orbits for underactuated 3-d bipedal robots with left-right symmetry. *IEEE Transactions on Robotics*, 30(2):365–381, 2013.

[10] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In *Proceedings of the IEEE international conference on computer vision*, pages 1026–1034, 2015.

[11] Gábor Holló. Demystification of animal symmetry: Symmetry is a response to mechanical forces. *Biology Direct*, 12(1):1–18, 2017.

[12] Günter Klambauer, Thomas Unterthiner, Andreas Mayr, and Sepp Hochreiter. Self-normalizing neural networks. *Advances in neural information processing systems*, 30, 2017.

[13] Cornelius Lanczos. *The variational principles of mechanics*. University of Toronto press, 2020.

[14] Tzu-Yuan Lin, Ray Zhang, Justin Yu, and Maani Ghaffari. Legged robot state estimation using invariant kalman filtering and learned contact events. In *5th Annual Conference on Robot Learning*, 2021.

[15] Emmy Noether. Invariante variationsprobleme, mathphys. *Klasse, pp235-257*, 1918.

[16] Daniel Ordonez-Apraez, Antonio Agudo, Francesc Moreno-Noguer, and Mario Martin. An adaptable approach to learn realistic legged locomotion without examples. In *2022 International Conference on Robotics and Automation (ICRA)*, pages 4671–4678. IEEE, 2022.

[17] David E Orin, Ambarish Goswami, and Sung-Hee Lee. Centroidal dynamics of a humanoid robot. *Autonomous robots*, 35(2):161–176, 2013.

[18] Jim Ostrowski and Joel Burdick. Geometric perspectives on the mechanics and control of robotic locomotion. In *Robotics Research*, pages 536–547. Springer, 1996.

[19] Robert J Quigley. Pseudovectors and reflections. *American Journal of Physics*, 41(3):428–430, 1973.

[20] Siamak Ravanbakhsh, Jeff Schneider, and Barnabas Poczos. Equivariance through parameter-sharing. In *International conference on machine learning*, pages 2892–2901. PMLR, 2017.

[21] Jon M Selig. *Geometric fundamentals of robotics*, volume 128. Springer, 2005.

[22] Alex Simpkins. System identification: Theory for the user, (ljung, l.; 1999)[on the shelf]. *IEEE Robotics & Automation Magazine*, 19(2):95–96, 2012.

[23] Elise Van der Pol, Daniel Worrall, Herke van Hoof, Frans Oliehoek, and Max Welling. Mdp homomorphic networks: Group symmetries in reinforcement learning. *Advances in Neural Information Processing Systems*, 33: 4199–4210, 2020.

[24] Rui Wang, Robin Walters, and Rose Yu. Approximately equivariant networks for imperfectly symmetric dynamics. *arXiv preprint arXiv:2201.11969*, 2022.

[25] James T. Wheeler. General coordinate covariance of the euler lagrange equations. Classical Mechanics class notes, 2014. URL http://www.physics.usu.edu/Wheeler/ClassicalMechanics/CMCoordinateinvarianceofEulerLagrange.pdf.

[26] P-B Wieber. Holonomy and nonholonomy in the dynamics of articulated motion. In *Fast motions in biomechanics and robotics*, pages 411–425. Springer, 2006.

[27] Raymond Yeh, Yuan-Ting Hu, and Alexander Schwing. Chirality nets for human pose regression. *Advances in Neural Information Processing Systems*, 32, 2019.

[28] Wenhao Yu, Greg Turk, and C Karen Liu. Learning symmetric and low-energy locomotion. *ACM Transactions on Graphics (TOG)*, 37(4):1–12, 2018.

[29] Martin Zinkevich and Tucker Balch. Symmetry in markov decision processes and its implications for single agent and multi agent learning. In *In Proceedings of the 18th International Conference on Machine Learning*. Citeseer, 2001.

APPENDIX A
PROPERTIES OF ROBOTIC SYSTEMS WITH DMSS

Here, we present a geometric (instead of albegraic) development analog to section IV-B. For clarity of the explanation, let us imagine two different Euclidean spaces and two versions of the robot: the original space (with reference frame o) and robot with coordinates $\boldsymbol{q}$ and $\dot{\boldsymbol{q}}$, and the virtual rotated/reflected space (with a reference frame $\overline{o}$, with configuration ${}^{o}\boldsymbol{X}_{\overline{o}} = \begin{bmatrix} \boldsymbol{R}_{\overline{g}} & \mathbf{r}_{\overline{o}} \\ \boldsymbol{0} & 1 \end{bmatrix}$) and virtual robot with coordinates $\overline{g} \cdot \boldsymbol{q}$ and $\overline{g} \cdot \dot{\boldsymbol{q}}$ referenced to $\overline{o}$. Noting that in the case of a reflection, the virtual robot has reflected versions of each rigid body.

For eqs. (3) and (5) to hold, there must exist an action $g \in \mathcal{G}$ transforming the real robot configuration $g \cdot \boldsymbol{q}, g \cdot \dot{\boldsymbol{q}}$ resulting in the same kinetic energy as the virtual robot's kinetic energy:

$$
\begin{aligned}
\mathcal{T}(g \cdot \boldsymbol{q}, g \cdot \dot{\boldsymbol{q}}) &= \tfrac{1}{2} \sum_{i=1}^{n_B} m_i \dot{\mathbf{r}}_{g,i}^{\mathsf{T}} \dot{\mathbf{r}}_{g,i} + \boldsymbol{w}_{g,i}^{\mathsf{T}} \mathbf{I}_i \boldsymbol{w}_{g,i} \\
&\doteq \tfrac{1}{2} \sum_{k=1}^{n_B} \overline{m}_k \dot{\overline{\mathbf{r}}}_k^{\mathsf{T}} \dot{\overline{\mathbf{r}}}_k + \overline{\boldsymbol{w}}_k^{\mathsf{T}} \overline{\mathbf{I}}_k \overline{\boldsymbol{w}}_k = \mathcal{T}(\overline{g} \cdot \boldsymbol{q}, \overline{g} \cdot \dot{\boldsymbol{q}}),
\end{aligned}
\tag{10}
$$

where $\dot{\mathbf{r}}_{g,i}$, $\boldsymbol{w}_{g,i}$, m_i and $\mathbf{I}_i$ are the linear and angular velocity, mass, and inertia matrix of the transformed body i (referenced to o). Likewise, $\dot{\overline{\mathbf{r}}}_i$, $\overline{\boldsymbol{w}}_i$, $\overline{m}_i$ and $\overline{\mathbf{I}}_i$ are the equivalent quantities for the virtual robot and body i (referenced to $\overline{o}$).

A. Symmetries of kinematic parameters:

Ignore momentarily the influence of the mass and inertia in terms of the real and virtual bodies. We can assert that for eq. (10) to hold, the transformed configuration should result in a kinematic tree indistinguishable from the virtual robot's. Thus, for everybody i in the real robot kinematic tree, there should exist an equivalent virtual body k (as seen in fig. 6, not always $k = i$). By equating the linear and angular velocities of the real and virtual bodies, referenced to o, and expressing the velocities as functions of the generalized coordinates we obtain:

$$
\dot{\mathbf{r}}_{g,i} = \dot{\overline{\mathbf{r}}}_k \doteq \boldsymbol{R}_{\overline{g}} \cdot \dot{\mathbf{r}}_k
$$
$$
\mathbf{J}_{T_i}(g \cdot \boldsymbol{q}) g \cdot \dot{\boldsymbol{q}} = \boldsymbol{R}_{\overline{g}} \cdot \mathbf{J}_{T_k}(\boldsymbol{q}) \dot{\boldsymbol{q}}
$$
$$
\mathbf{J}_{T_i}(g \cdot \boldsymbol{q}) g = \boldsymbol{R}_{\overline{g}} \cdot \mathbf{J}_{T_k}(\boldsymbol{q}) \tag{11}
$$
$$
\boldsymbol{w}_{g,i} = \overline{\boldsymbol{w}}_k \doteq |\boldsymbol{R}_{\overline{g}}| \boldsymbol{R}_{\overline{g}} \cdot \boldsymbol{w}_k
$$
$$
\mathbf{J}_{R_i}(g \cdot \boldsymbol{q}) g \cdot \dot{\boldsymbol{q}} = |\boldsymbol{R}_{\overline{g}}| \boldsymbol{R}_{\overline{g}} \cdot \mathbf{J}_{R_k}(\boldsymbol{q}) \dot{\boldsymbol{q}},
$$
$$
\mathbf{J}_{R_i}(g \cdot \boldsymbol{q}) g = |\boldsymbol{R}_{\overline{g}}| \boldsymbol{R}_{\overline{g}} \cdot \mathbf{J}_{R_k}(\boldsymbol{q}), \tag{12}
$$

where $\mathbf{J}_{T_i}(\boldsymbol{q})$, $\mathbf{J}_{R_i}(\boldsymbol{q}) \in \mathbb{R}^{3 \times n}$ are the position and orientation analytical Jacobians (describing the instantaneous velocity vectors contributed by each DoF to body i) of the real robot at a configuration $\boldsymbol{q}$ [26]. Formulating eq. (12) for each of the n_B bodies of the robot we obtain at best $n_B \times 3 \times n$ non-linear equations that can be used to assert if g exists. In practice, the action representation $\rho_{\mathrm{Q}}(g)$ and especially its component acting on the joint space $\rho_{\mathrm{Q}_J}(g)$ can be trivially determined by solving eq. (12) (or equivalently eq. (7)) for each body from top to bottom of the kinematic tree (i.e., base first, end-effectors last), if g exists.
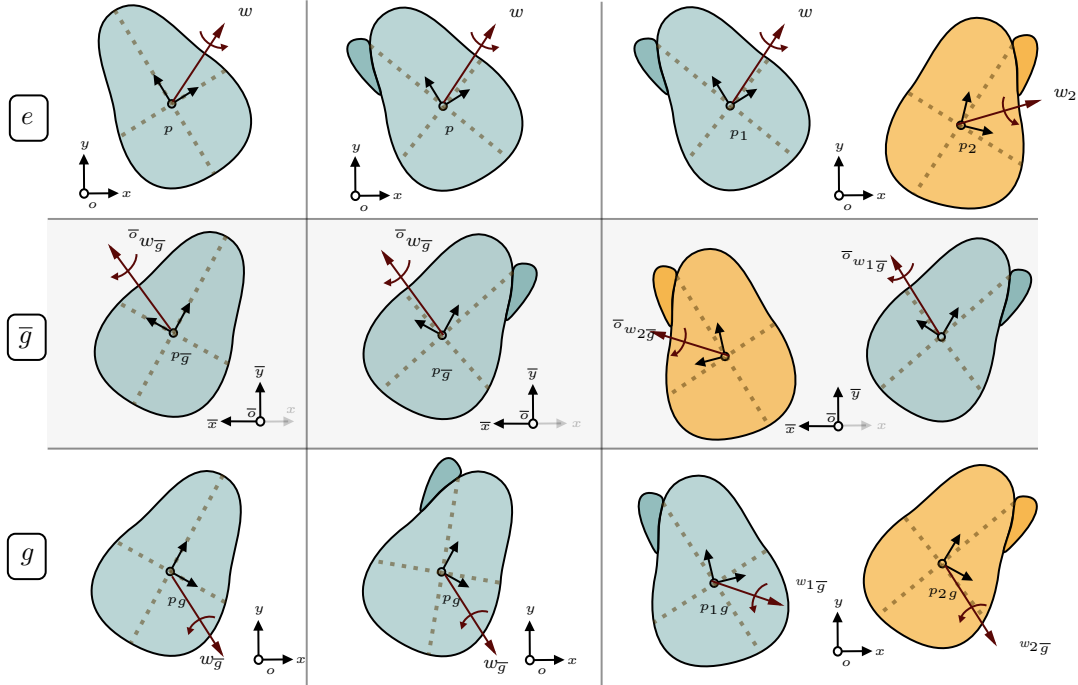
Fig. 5: Properties of bodies capable of imitating a true reflection $\overline{g}$ of space (w.r.t the yz-plane in this case), with a proper transformation g involving only a rotation and translation. The first row shows the original bodies with their respective angular velocities $\boldsymbol{w}$, subjected to trivial symmetry transformation e (dashed lines represent the principle axes of inertia of the bodies), and the second and third rows display the effect of $\overline{g}$ and g on the bodies and angular velocities, respectively. The first column displays a rigid body with symmetric mass distribution, for which g exists, as the reflected and rotated bodies share an equivalent angular kinetic energy. The second column shows a rigid body with asymmetrical mass distribution, for which the rotation g, that produces a kinematic symmetry, results in the reflected and rotated bodies having different angular kinetic energies (eq. (2)). The third column shows two bodies with asymmetrical mass distributions, each a reflected version of the other, in this case, the action g swaps bodies configurations to imitate the configuration and energy state of the reflected bodies transformed with $\overline{g}$. Angular velocity is a pseudo-vector (or axial-vector), for which a reflection transformation is computed as $\boldsymbol{w}_{\overline{g}} = |\boldsymbol{R}_{\overline{g}}|\boldsymbol{R}_{\overline{g}}\boldsymbol{w}$ (see Quigley [19]).

B. Symmetries of dynamic prameters

Let us assume kinematic symmetry and direct our attention now to the influence of the mass and inertia terms on the kinetic energy of a single rigid body when it is transformed with the action g, which imitates a true reflection of space $\overline{g}$. Focus on the first two columns of fig. 5. Because of the kinematic symmetry the CoM of the reflected and transformed bodies coincide, both bodies have equivalent linear components of kinetic energy. However, for arbitrary rigid bodies, the reflected and transform bodies will have different angular components of kinetic energy. Note that in the general case, the transformed and reflected bodies' inertia will differ, thus even if both bodies have the same angular velocities, their kinetic energy will differ.

Let p, $p_{\overline{g}}$ and p_g be frames located at the CoM of the original, reflected and transformed bodies, aligned with the principal axes of inertia of each of the bodies. Similarly, denote ${}^o\mathbf{I}$ and ${}^o\mathbf{I}_g$ as the original and transformed bodies inertias referenced to o, and ${}^{\overline{o}}\overline{\mathbf{I}}_{\overline{g}}$ as the reflected body inertia referenced to the reflected Euclidean space $\overline{o}$. In order to

comply with eq. (12), we must ensure that:

$$
{}^o\boldsymbol{w}_{\overline{g}}^{\mathsf{T}}\,{}^o\mathbf{I}_g\,{}^o\boldsymbol{w}_{\overline{g}} = {}^{\overline{o}}\boldsymbol{w}_{\overline{g}}^{\mathsf{T}}\,{}^{\overline{o}}\overline{\mathbf{I}}_{\overline{g}}\,{}^{\overline{o}}\boldsymbol{w}_{\overline{g}},,
$$

$$
(\boldsymbol{R}_{\overline{g}}\,{}^o\boldsymbol{w})^{\mathsf{T}}\,{}^o\mathbf{I}_g\,(\boldsymbol{R}_{\overline{g}}\,{}^o\boldsymbol{w}) = {}^{\overline{o}}\boldsymbol{w}_{\overline{g}}^{\mathsf{T}}\,{}^{\overline{o}}\overline{\mathbf{I}}_{\overline{g}}\,{}^{\overline{o}}\boldsymbol{w}_{\overline{g}}
$$

$$
\mid {}^o\boldsymbol{w}_{\overline{g}} = |\boldsymbol{R}_{\overline{g}}|\boldsymbol{R}_{\overline{g}}\,{}^o\boldsymbol{w},
$$

$$
{}^o\mathbf{I}_g = \boldsymbol{R}_{\overline{g}}\,{}^{\overline{o}}\overline{\mathbf{I}}_{\overline{g}}\,\boldsymbol{R}_{\overline{g}}
$$

$$
\mid {}^o\boldsymbol{w} \equiv {}^{\overline{o}}\boldsymbol{w}_{\overline{g}},\quad \boldsymbol{R}_{\overline{g}}\boldsymbol{R}_{\overline{g}} = \mathbf{I},
$$

$$
{}^o\boldsymbol{R}^{p_g}\,{}^{p_g}\mathbf{I}_g\,{}^o\boldsymbol{R}^{p_g\,\mathsf{T}} = \boldsymbol{R}_{\overline{g}}\,{}^{\overline{o}}\boldsymbol{R}^{p_{\overline{g}}}\,{}^{p_{\overline{g}}}\overline{\mathbf{I}}_{\overline{g}}\,{}^{\overline{o}}\boldsymbol{R}^{p_{\overline{g}}\,\mathsf{T}}\boldsymbol{R}_{\overline{g}}
$$

$$
\mid {}^a\mathbf{I} = {}^b\boldsymbol{R}^{a\,b}\mathbf{I}^{\,b}\boldsymbol{R}^{a\,\mathsf{T}},
$$

$$
{}^o\boldsymbol{R}^{p_g}\,{}^{p_g}\mathbf{I}_g\,{}^o\boldsymbol{R}^{p_g\,\mathsf{T}} = \boldsymbol{R}_{\overline{g}}\,{}^o\boldsymbol{R}^p\,{}^p\boldsymbol{R}^{p_{\overline{g}}}\,{}^{p_{\overline{g}}}\overline{\mathbf{I}}_{\overline{g}}\,{}^p\boldsymbol{R}^{p_{\overline{g}}\,\mathsf{T}}\,{}^{\overline{o}}\boldsymbol{R}^{p_{\overline{g}}\,\mathsf{T}}\boldsymbol{R}_{\overline{g}},
$$

$$
{}^o\boldsymbol{R}^{p_g} \doteq \boldsymbol{R}_{\overline{g}}\,{}^o\boldsymbol{R}^p\,{}^p\boldsymbol{R}^{p_{\overline{g}}} \tag{13a}
$$

$$
\mid {}^p\mathbf{I} \equiv {}^{p_g}\mathbf{I}_g \equiv {}^{p_{\overline{g}}}\overline{\mathbf{I}}_{\overline{g}}.
$$

What eq. (13a) states is that in order for the reflected and transformed bodies to have equivalent angular kinetic energy, both bodies should have co-linear (or aligned) principal axes of inertia. This allows us to describe ${}^o\boldsymbol{R}^{p_g}$ as a function of the original body configuration ${}^o\boldsymbol{R}^p$ and two reflection matrices: the true reflection of space $\boldsymbol{R}_{\overline{g}}$ and a body specific diagonal reflection matrix ${}^p\boldsymbol{R}^{p_{\overline{g}}}$, which exists only if the rigid

body has a symmetric mass distribution. A visual example for symmetric and asymmetrical rigid bodies is presented in fig. 5 left and middle columns. A similar analysis follows for DMSs imitating rotations

C. Symmetric position and velocity constraint configuration spaces

Although it is implicitly implied on eq. (2) that the constrained position Q and velocity T_qQ configuration vector spaces should also be symmetric or equivariant, this property might be easily overlooked. As mentioned in section IV the relevance of morphological symmetries relies on the equivariant nature of the system dynamics (eqs. (1) and (3)), which imprints symmetry constraints on optimal control policies and proprioceptive and exteroceptive measurements. However, with non-symmetric constrained configuration spaces, eq. (2) will not hold for every system state $q \in Q$, $\dot{q} \in T_qQ$, and any uncontrolled or controlled trajectory of the system dynamics shall not have a symmetric equivalent trajectory, as this has the potential to violate the constraints of the configuration space.

<center>APPENDIX B</center>
<center>EFFICIENT CONSTRUCTION OF $\mathcal{G}$-EQUIVARIANT NNs FOR DMS GROUPS $\mathcal{G}$</center>

As mentioned in section V our work builds upon the framework for the construction of $\mathcal{G}$-equivariant NN of Finzi et al. [6]. The core limitation of this framework is the inability to handle large dimensional spaces, due to the computational and memory complexities. For instance, for an equivariant layer with input dimension n and output dimension m, the computational complexity of finding the equivariant linear map basis **B** (which is quadratic $\mathcal{O}((mn)^2 r^2)$ through the Krylov subspace method) and the memory complexity of $\mathbf{B} \in \mathbb{R}^{mn \times r} \mid r \leq mn$, become easily intractable for moderate n and m dimensions. This limitation is openly discussed in the EMLP repository *README.md*, but regretfully not in the original paper.

In practice, we found these limitations when trying to construct the equivariant version of the Contact CNN [14] in our second experiment. This architecture in its internal layers has $n, m > 2000$, for which: (i) the Krylov subspace method complexity renders the operation intractable with standard hardware and (ii) the matrices **B** of internal layers required storage of $1[Gb] >$ for moderate input output dimensions ($m, n \approx 250$) and $1[Pb] >$ for $m, n > 2000$). See section C for a comparison between dense and sparse matrix representations.

A. Trainable parameter reduction of $\mathcal{G}$-equivariant layers (for $\mathcal{G}$ a DMS group)

Determining analytically the number of trainable parameters (i.e. the rank r) of an $\mathcal{G}$-equivariant layer is, in general, an unresolved problem. However, for DMS groups, r can be computed once the input-output action representations are known. The requirement to compute r is that actions affecting the linear maps are a semi-direct product[6] of the input-output groups, and the input-output representations are generalized

permutation matrices. These conditions are met for most DMS groups (see section D).

The equivariance constraints of eq. (9) on linear maps of perceptron (or convolutional) layers imply a reduction of trainable parameters from $|\boldsymbol{w}| = mn$ to $|\boldsymbol{c}| = r \leq mn$. For DMS groups, r is associated with the number of unique orbits of the elements of $\boldsymbol{w}$. Thus we can compute this value using the orbit-counting theorem (also known as Burnside's Lemma), which states that the number of orbits is the average number of fix-points of $\mathcal{G}$, that is $r = \frac{1}{|\mathcal{G}|} \sum_{g \in \mathcal{G}} |\boldsymbol{w}^g|$, where $\boldsymbol{w}^g \doteq \{w \in \boldsymbol{w} : g \cdot w = w\}$ represents the set of elements of $\boldsymbol{w}$ that are invariant to g (i.e. fix-points). Those fix-points can be identified by the elements on the diagonal of $\rho_{\boldsymbol{w}}(g)$ that are equal to one. Therefore, for a $\mathcal{G}$-equivariant layer, the number of trainable parameters is determined by:

$$r = \frac{1}{|\mathcal{G}|} \sum_{g \in \mathcal{G}} \chi^{\mathbf{1}}_{\rho_{\boldsymbol{W}}}(g) = \frac{1}{|\mathcal{G}|} \sum_{g \in \mathcal{G}} \chi^{\mathbf{1}}_{\rho_{in}}(g^{\text{-}1}) \cdot \chi^{\mathbf{1}}_{\rho_{out}}(g), \quad (14)$$

denoting $\chi^{\mathbf{1}}_{\rho}(g) : \mathcal{G} \to \mathbb{N}$ as the number of fix-points of the action representation $\rho(g)$. Therefore, the number of trainable parameters can range from $|\boldsymbol{w}|/|\mathcal{G}| \leq r \leq |\boldsymbol{w}|$, depending on the fix-points of the layers' input and output spaces.

B. Parameter initialization of equivariant layers for DMS

Consider a Equivariant Neural Network architecture composed of multiple layers of equivariant linear (or convolutional) layers of the form $^l\boldsymbol{y} := \sigma(^l\boldsymbol{W}\,^l\boldsymbol{x} + \,^l\boldsymbol{b})$, being l the layer index, $^l\boldsymbol{x} \in \mathbb{R}^n$ and $^l\boldsymbol{y} \in \mathbb{R}^m$ the layer's input and output vector spaces, $^l\boldsymbol{W} \doteq \sum_k^r \,^lc_k\,^l\mathbf{B}_{:,:,k} \in \mathbb{R}^{m \times n}$ the layer's linear map, $^l\mathbf{B} \in \mathbb{R}^{m \times n \times r}$ the layer's r basis vectors spawning the space of equivariant linear maps, $^l\boldsymbol{c} \in \mathbb{R}^r$ the layer's trainable parameters, and $^l\boldsymbol{b} \in \mathbb{R}^m$ the layer's bias vector.

For the optimal flow of information throughout the network, it's relevant to initialize the trainable parameters such that the variance of activations (during inference/forward-propagation) and gradients (during back-propagation) is kept constant, avoiding activations/gradients from vanishing or exploiting [8][7].

The derivation is based on the equivalent process for unconstrained layers presented in He et al. [10]. Let the layer's activations before the non-linearity be denoted by $^l\boldsymbol{z} = \,^l\boldsymbol{W}\,^l\boldsymbol{x} + \,^l\boldsymbol{b}$, such that $^l\boldsymbol{y} = \sigma(^l\boldsymbol{z})$, and note that $^l\boldsymbol{x} = \,^{l-1}\boldsymbol{y}$. Furthermore, we will assume the elements of $^l\boldsymbol{c}$ and $^l\boldsymbol{x}$ are mutually independent and sampled from two independent distributions, denoting the random variables of the two distributions as ^lc and $^l\mathrm{x}$.

The core difference in the initialization of unconstrained and equivariant layers lies in the way the linear map is

---

[7]See Pierre Ouannes blog: pouannes.github.io/blog/initialization

parameterized. For equivariant layers we have:

$$\text{Var}(^l\boldsymbol{W}^l\boldsymbol{x} + {}^l\boldsymbol{b}) = \sum_i^m \sum_j^n \text{Var}\left({}^lW_{i,j}\,{}^l\text{x}_j\right) \mid \text{Var}({}^l\boldsymbol{b}) = 0$$

$$= \sum_i^m \sum_j^n \text{Var}\left(\left(\sum_k^r {}^lc_k\,{}^l\mathbf{B}_{m,n,k}\right){}^l\text{x}_j\right)$$

$$, = \text{Var}\left({}^lc\,{}^l\text{x}\right) \underbrace{\sum_i^m \sum_j^n \sum_k^r \mathbf{B}_{m,n,k}^2}_{\lambda_{l\mathbf{B}}} \tag{15}$$

$$\mid \text{Var}\left(\sum_a \underbrace{s_a}_{const}\,\mathbf{p}\right) = \sum_a s_a^2 \text{Var}(\mathbf{p})$$

In the forward-propagation scenario, we are interested in conserving the variance of the activations throughout layers, that is we must ensure $\text{Var}({}^l\mathbf{z}) = \text{Var}({}^{l-1}\mathbf{z})$. Using eq. (15) we obtain:

$$\text{Var}({}^l\boldsymbol{z}) = \text{Var}({}^l\boldsymbol{W}^l\boldsymbol{x} + {}^l\boldsymbol{b})$$

$$m\,\text{Var}({}^l\mathbf{z}) = \lambda_{l\mathbf{B}}\text{Var}\left({}^lc\,{}^l\text{x}\right)$$

$$\text{Var}({}^l\mathbf{z}) = \frac{\lambda_{l\mathbf{B}}}{m}\left(\underbrace{\mathbb{E}({}^lc^2)}_{\text{Var}({}^lc)}\mathbb{E}({}^l\text{x}^2) - \underbrace{\mathbb{E}({}^lc)^2\mathbb{E}({}^l\text{x})^2}_{=0}\right)$$

$$\text{Var}({}^l\mathbf{z}) = \frac{\lambda_{l\mathbf{B}}}{m}\text{Var}({}^lc)\mathbb{E}\left({}^{l-1}\text{y}^2\right)$$

$$\mid {}^l\text{x} = {}^{l-1}\text{y} = \sigma({}^{l-1}\mathbf{z})$$

$$\text{Var}({}^l\mathbf{z}) = \frac{\lambda_{l\mathbf{B}}\lambda_\sigma}{m}\text{Var}({}^lc)\text{Var}({}^{l-1}\mathbf{z}) \tag{16}$$

$$\mid \mathbb{E}\left({}^{l-1}\text{y}^2\right) = \lambda_\sigma\text{Var}\left({}^{l-1}\mathbf{z}\right)$$

$$\text{Var}({}^l\mathbf{z}) \equiv \text{Var}({}^{l-1}\mathbf{z}) \tag{17}$$

$$\mid \text{Var}\left({}^lc\right) \doteq \frac{m}{\lambda_{l\mathbf{B}}\lambda_\sigma}$$

where λ_σ in eq. (16) is a non-linearity dependent scalar computed analytically or empirically (see He et al. [10]). In eq. (17) we conclude that if we sample the equivariant layer trainable parameters ${}^lc$ from a distribution ensuring $\text{Var}({}^lc) \doteq \frac{m}{\lambda_{l\mathbf{B}}\lambda_\sigma}$, the variance of the activations across equivariant layers remain constant. A similar procedure can be applied to the backward propagation case, concluding that in order to maintain a constant variance of the gradients across the network layers we should sample the trainable parameters ensuring $\text{Var}({}^lc) \doteq \frac{n}{\lambda_{l\mathbf{B}}\lambda_\sigma}$. As remarked in He et al. [10] both variance values for the forward and backward propagation cases lead to the proper flow of information in the network. On fig. 7, it can be appreciated that our method achieves equivalent results for equivariant architectures as [10] does for standard linear and convolutional architectures.

APPENDIX C
IMPLEMENTATION DETAILS & CODE

Additional to this section, we provide open-access code with the scripts for reproducing the experiments of this work,

the parameters of the models used for comparison, along with additional interactive examples visualizing morphological symmetries of both robotic systems and data.

A. Efficient data augmentation

Since any input $\boldsymbol{x}$ and output $\boldsymbol{y}$ spaces of equivariant architectures have matrix symmetry action representations, $\rho_{\mathcal{X}}(g)$, $\rho_{\mathcal{Y}}(g)$, it is possible to perform batched data augmentation, reducing the computational complexity of augmenting a batch of N_b samples from N_b matrix-vector multiplications to a single matrix-matrix multiplication, preferably performed after data is loaded to GPU for optimal performance.

B. Hyperparameter tunning

The only hyper-parameter tunned for each model and model variant was the learning rate. For all model variants presented in this work (except the original Contact-CNN model from Lin et al. [14], which we retrained using the same hyperparameters reported by the authors) we ran a grid-search in log-scale among 20 different learning rates. In this scenario, we always used the entire training dataset and optimized w.r.t computed loss in the entire validation partition. The learning rate values used for each model are depicted in section C-D2.

C. Experiment: CoM Momentum Estimation

The dataset for the CoM estimation experiment was generated using Pinocchio [3], which in turn uses the URDF models of the robots Solo and Atlas, to extract the kinematic and dynamic parameters required to compute the Centroidal Momentum Matrix $\mathbf{A}_G(\boldsymbol{q})$ matrix [17], with which computing the CoM momentum reduces to:

$$g \cdot \boldsymbol{h} = \mathbf{A}_G(g \cdot \hat{\boldsymbol{q}})g \cdot \dot{\hat{\boldsymbol{q}}} \qquad \mid \quad \forall g \in \mathcal{G}. \tag{18}$$

$$g \cdot \boldsymbol{h} \approx \hat{f}\left(g \cdot \hat{\boldsymbol{q}}, g \cdot \dot{\hat{\boldsymbol{q}}}; \phi\right) \qquad \mid \quad \forall g \in \mathcal{G}. \tag{19}$$

Where eq. (18) expresses the analytical $\mathcal{G}$-equivariant function to compute the CoM momentum. While eq. (19) is the approximation of this function by an $\mathcal{G}$-equivariant NN, with parameters ϕ.

1) Determination of the input and output representations $\rho_{\mathcal{X}}(g), \rho_{\mathcal{Y}}(g) \mid g \in \mathcal{G}$: Both robots Solo and Atlas evolve in the Euclidean space of 3-dimensions. Therefore their configuration space can be decoupled into $\text{Q} \doteq \mathbb{E}_3 \times \text{Q}_J$. After identifying their symmetry groups and their corresponding $\mathbb{E}_3$ and Q_J representations $(\rho_{\mathbb{E}_3}(g), \rho_{\text{Q}}(g) \mid \forall g \in \mathcal{G})$, identifying the representations of the input and output spaces of the NN function approximator (eq. (19)) becomes a trivial task considering that:

$$\underbrace{\begin{bmatrix} \rho_{\mathbb{E}_d}(\overline{g}) & \mathbf{0} \\ \mathbf{0} & \rho_{\mathbb{E}_d}(\overline{g}) \end{bmatrix}}_{\rho_{\mathcal{Y}}(g)} \underbrace{\begin{bmatrix} \boldsymbol{l} \\ \mathbf{k} \end{bmatrix}}_{\boldsymbol{y}} \approx f\left(\underbrace{\begin{bmatrix} \rho_{\text{Q}_J}(g) & \mathbf{0} \\ \mathbf{0} & \rho_{\text{Q}_J}(g) \end{bmatrix}}_{\rho_{\mathcal{X}}(g)} \underbrace{\begin{bmatrix} \hat{\boldsymbol{q}} \\ \dot{\hat{\boldsymbol{q}}} \end{bmatrix}}_{\boldsymbol{x}}\right)$$

$$\mid \quad \forall \quad (g, \overline{g}) \mid g \in \mathcal{G}, \overline{g} \in \mathbb{E}_3 \tag{20}$$

Defining $\boldsymbol{x} = \begin{bmatrix} \hat{\boldsymbol{q}} \\ \dot{\hat{\boldsymbol{q}}} \end{bmatrix} \in \mathbb{R}^{2n_J}$ and $\boldsymbol{y} = \boldsymbol{h} \in \mathbb{R}^{2d} \equiv \mathbb{R}^6$. Note that by definition any improper transformation applied to a pseudo-vector (e.g. angular velocity/momentum, torques) is computed as $|\boldsymbol{R}|\boldsymbol{R} \cdot \mathbf{k}$.

Layer Type	n	m	Dense Memory [Bytes] $\rho_{\boldsymbol{w}}(g)$	B	Sparse Memory [Bytes] $\rho_{\boldsymbol{w}}(g)$	B
1D-Conv	54	64	$764.41M$	$191.10M$	$221.18k$	$110.59k$
1D-Conv	64	64	$1.07G$	$268.43M$	$262.14k$	$131.07k$
1D-Conv	64	128	$4.29G$	$1.07G$	$524.28k$	$262.14k$
1D-Conv	128	128	$17.18G$	$4.29G$	$1.04M$	$524.28k$
Percept	4736	2048	$6.02P$	$1.50P$	$620.75M$	$310.37M$
Percept	2048	512	$70.36T$	$17.59T$	$67.10M$	$33.55M$
Percept	512	16	$4.29G$	$1.07G$	$524.28k$	$262.14k$

TABLE I: **Comparison of memory complexity of individual layers of the equivariant version of Contact-CNN [14] (ECNN).** This example compares the sparse and dense representations of matrices $\mathbf{B} \in \mathbb{R}^{mn \times r}$ and the $|\mathcal{G}|$ group action representations $\rho_{\boldsymbol{w}}(g) \in \mathbb{R}^{mn \times mn}$, for the symmetry group $\mathcal{G} = \mathcal{C}_2$ of the Mini-Cheetah robot, with $r = {}^{mn}/_2$ (see eq. (14)). Here, n, m represents the input and output dimensions of each layer. The dense memory complexity of all action representations increases with the group order $|\mathcal{G}|$ while the memory complexity for $\mathbf{B}$ decreases with larger group orders (since $r \leq mn$ becomes smaller). We assume floating point representations with 32 bits.

RF	LF	RH	LH	$\boldsymbol{y}$	$g \cdot \boldsymbol{y}$	LF	RF	LH	RH
0	0	0	0	0	0	0	0	0	0
0	0	0	1	1	2	0	0	1	0
0	0	1	0	2	1	0	0	0	1
0	0	1	1	3	3	0	0	1	1
0	1	0	0	4	8	1	0	0	0
0	1	0	1	5	10	1	0	1	0
0	1	1	0	6	9	1	0	0	1
0	1	1	1	7	11	1	0	1	1
1	0	0	0	8	4	0	1	0	0
1	0	0	1	9	6	0	1	1	0
1	0	1	0	10	5	0	1	0	1
1	0	1	1	11	7	0	1	1	1
1	1	0	0	12	12	1	1	0	0
1	1	0	1	13	14	1	1	1	0
1	1	1	0	14	13	1	1	0	1
1	1	1	1	15	15	1	1	1	1

TABLE II: Symmetric contact state for Mini-Cheetah quadruped robot. Considering its morphological symmetry group $\mathcal{C}_2 = \{e, g\}$. Each leg binary contact state (LF: Left Front, RF: Right Front, LH: Left Hind, RH: Right Hind) is displayed with its corresponding robot contact state $\boldsymbol{y}$.

2) Practical details of the dataset generation: The URDF files of the robots Solo and Atlas are generated using XACRO scripts, which replicate the structure of limbs to their symmetric counterparts, making the dynamics of the robots in simulation exactly $\mathcal{G}$-equivariant. However, the algorithm for computing the CoM momentum from Pinocchio is numerically sensitive, resulting in the orbits of the momentum $\mathcal{G} \cdot \boldsymbol{h}$ deviating slightly from the theoretical orbits. Therefore to reduce numerical errors and ensure the theoretical equivariance of the data, we replace every target variable by the average of its orbit $\boldsymbol{y} = \boldsymbol{h} \doteq \frac{1}{|G|} \sum \mathcal{G} \cdot g^{-1}(\mathbf{A}_G (\rho_Q(g)\hat{\boldsymbol{q}}) \rho_Q(g)\dot{\hat{\boldsymbol{q}}}) \mid \forall g \in \mathcal{G}$.

D. Experiment: Static-Friction-Regime Contact Detection

The dataset presented in [14] is composed of output samples $\boldsymbol{y} \in \mathbb{R}^{16}$, where each dimension of $\boldsymbol{y}$ represents a logit of a specific contact state, among the 16 different combinations of each of the 4 legs possible binary contact states. The input samples $\boldsymbol{y} = \{\boldsymbol{z}_i\}_{i=0}^{150} \in \mathbb{R}^{54 \times 150}$, are a history of 150 samples $\boldsymbol{z} = [\hat{\boldsymbol{q}}, \dot{\hat{\boldsymbol{q}}}, \boldsymbol{a}, \boldsymbol{w}, \boldsymbol{p}, \boldsymbol{v}] \in \mathbb{R}^{54}$. Where $\hat{\boldsymbol{q}} \in \mathbb{R}^{n_J}, \dot{\hat{\boldsymbol{q}}} \in \mathbb{R}^{n_J}, \boldsymbol{a} \in \mathbb{R}^3, \boldsymbol{w} \in \mathbb{R}^3, \boldsymbol{p} \in \mathbb{R}^{12}, \boldsymbol{v} \in \mathbb{R}^{12}$ are the MIT-Mini-Cheetah

robot joint-space positions, velocities, base linear acceleration, base angular velocity, and each of the four legs feet's position and velocities, respectively, referenced to the robots base frame B.

The function approximator to learn is expected to be approximately equivariant to the reflection group $\mathcal{C}_2$, considering the sagittal symmetry of the robot morphology. Therefore:

$$g \cdot \boldsymbol{y} = \hat{f}(g \cdot \boldsymbol{x}; \boldsymbol{\phi}) \quad \mid \quad g \in \mathcal{G} = \mathcal{C}_2 \qquad (21)$$

1) Determination of the input and output representations $\rho_{\mathcal{X}}(g), \rho_{\mathcal{Y}}(g) \mid g \in \mathcal{G}$: The MiniCheetah robot evolves in the Euclidean space of 3-dimensions. Therefore its configuration space can be decoupled into $Q \doteq \mathbb{E}_3 \times Q_J$. After identifying their symmetry groups and their corresponding $\mathbb{E}_3$ and Q_J representations $(\rho_{\mathbb{E}_3}(g), \rho_Q(g) \mid \forall g \in \mathcal{G})$, we can identify the representations of the input and output spaces of the NN function approximator (eq. (21)), considering that:

$$\rho_{\mathcal{Y}}(g)\boldsymbol{y} \approx$$

$$f\left(\underbrace{\begin{bmatrix} \rho_{Q_J}(g) & 0 & 0 & 0 & 0 & 0 \\ 0 & \rho_{Q_J}(g) & 0 & 0 & 0 & 0 \\ 0 & 0 & \rho_{\mathbb{E}_3}(\overline{g}) & 0 & 0 & 0 \\ 0 & 0 & 0 & \rho_{\mathbb{E}_3}(\overline{g}) & 0 & 0 \\ 0 & 0 & 0 & 0 & \rho_{\boldsymbol{p}}(g) & 0 \\ 0 & 0 & 0 & 0 & 0 & \rho_{\boldsymbol{p}}(g) \end{bmatrix}}_{\rho_{\mathcal{X}}(g)} \underbrace{\begin{bmatrix} \hat{\boldsymbol{q}} \\ \dot{\hat{\boldsymbol{q}}} \\ \boldsymbol{a} \\ \boldsymbol{w} \\ \boldsymbol{p} \\ \boldsymbol{v} \end{bmatrix}}_{\boldsymbol{x}}\right)$$

$$\mid \forall (g, \overline{g}) | g \in \mathcal{G}, \overline{g} \in \mathbb{E}_3$$

Where the representation $\rho_{\boldsymbol{p}}(g)$ acting on $\boldsymbol{p} \in \mathbb{R}^{12}$ and $\boldsymbol{v} \in \mathbb{R}^{12}$ is determined understanding that each of the feet positions $(\boldsymbol{p}_{RF}, \boldsymbol{p}_{LF}, \boldsymbol{p}_{RH}, \boldsymbol{p}_{LH})$ and velocities $(\boldsymbol{v}_{RF}, \boldsymbol{v}_{LF}, \boldsymbol{v}_{RH}, \boldsymbol{v}_{LH})$ are simply vectors living in $\mathbb{E}_3$. Thus, we must apply the euclidean action $\rho_{\mathbb{E}_3}(\overline{g})$ while at the same time permuting the feets (similar to the permutation of the

Robot	Model	$\mathcal{G}$	lr	Samples
Solo	MLP/EMLP	$\mathcal{C}_2$ & $\mathcal{K}_4$	2.4×10^{-3}	100k
Atlas	MLP/EMLP	$\mathcal{C}_2$	1.5×10^{-3}	500k
MiniCheetah	CNN	$\mathcal{C}_2$	1.0×10^{-4}	730k
MiniCheetah	E-CNN	$\mathcal{C}_2$	1.0×10^{-5}	

TABLE III: Robot, Models and Dataset parameters.

kinematic subchains described by $\rho_{Q_J}(g)$):

$$g \cdot \boldsymbol{p} = \underbrace{\rho_{\mathbb{R}^4}(g) \otimes \rho_{\mathbb{E}_3}(\overline{g})}_{\rho_{\boldsymbol{p}}(g)}\boldsymbol{p}, \quad | \; \forall (g, \overline{g})|g \in \mathcal{G}, \overline{g} \in \mathbb{E}_3 \quad (22)$$

$$= \begin{bmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix} \otimes \rho_{\mathbb{E}_3}(\overline{g}) \begin{bmatrix} \boldsymbol{p}_{RF} \\ \boldsymbol{p}_{LF} \\ \boldsymbol{p}_{RH} \\ \boldsymbol{p}_{LH} \end{bmatrix} \quad | \; g \neq e$$

$$g \cdot \boldsymbol{v} = \underbrace{\rho_{\mathbb{R}^4}(g) \otimes \rho_{\mathbb{E}_3}(\overline{g})}_{\rho_{\boldsymbol{p}}(g)}\boldsymbol{v} \quad | \; \forall (g, \overline{g})|g \in \mathcal{G}, \overline{g} \in \mathbb{E}_3 \quad (23)$$

$$= \begin{bmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix} \otimes \rho_{\mathbb{E}_3}(\overline{g}) \begin{bmatrix} \boldsymbol{v}_{RF} \\ \boldsymbol{v}_{LF} \\ \boldsymbol{v}_{RH} \\ \boldsymbol{v}_{LH} \end{bmatrix} \quad | \; g \neq e$$

Being $\rho_{\mathbb{R}^4}(g) \mid \forall g \in \mathcal{C}_2$ two regular representations of $\mathcal{C}_2$ stack in block diagonal form to form a permutation representation in 4-dimensional space, representing the right-left symmetries of the 4 kinematic tree's subchains (four legs). The nature of the representation $\rho_{Q_J}(g) \doteq \rho_{\mathbb{R}^4}(g) \otimes \boldsymbol{I}_{n_s}$ might be better understood if you consider that we apply $n_{\mathbb{R}^4}(g)$ to each set of symmetric DoF in the kinematic three, which for the Mini-Cheetah is $n_s = 3$ sets of symmetric DoF (see fig. 2-right). See simpler examples in fig. 6.

Lastly, the representation for the contact state $\rho_{\mathcal{Y}}(g)$ is given by the permutation matrix relating $\boldsymbol{y}$ and $g \cdot \boldsymbol{y}$ described in section C-D.

2) Details on dataset partitioning: We modified the original dataset partitioning to properly evaluate the generalization capacity of the models. The original dataset was composed of 15 different recordings varying ground type and gait type used during data collection (most recordings were performed on a trot gait, which heavily biased the dataset to contact states 0, 6, and 9 of section C-D).

The authors of [14] partitioned all 15 recordings into $(70\%, 15\%, 15\%)$ training, validation and testing. This partition was made such that the first 70% time-samples of each recording were assigned for training, the following 15% to validation, and the rest for testing.

Because we are interested in studying the generalization capacity of the models and the out-of-training-distribution performance, we modified this partitioning such that among the 15 different recordings we selected randomly 5 recordings for testing, and the remaining 10 recordings were used for training splitting these recordings into $(85\%, 15\%)$ training and validation splits as in [14], that is, for each recording, the first 85% data-samples go for training and the remaining for validation.

The selected training-validation recordings were: air walking gait, concrete difficult slippery, concrete left circle, middle pebble, rock road, asphalt road, concrete galloping, grass, old asphalt road, sidewalk. While the selected testing recordings were: air jumping gait, concrete pronking, concrete right circle, forest, small pebble.

E. Mitigation of suboptimal asymmetries in model performance

When comparing individual leg classification we see that the equivariant model converges to having a similar performance for each symmetric pair of legs, while the unconstrained models converge to an asymmetrical suboptimal state favoring the contact detection of one leg at the expense of reduced performance for the symmetric leg (see LF and RF f1-scores). This asymmetrical performance is attributed to the CNN and CNN-aug models learning to extract temporal features for both symmetric legs separately, increasing the likelihood of converging to asymmetrical local minima. On the contrary, the equivariant model E-CNN can be thought of as learning to extract a single set of *symmetric* temporal features for each symmetric pair of states (a consequence of the model's equivariance and parameter sharing). This implies that the temporal features used for determining the contact state of, say the left frontal leg, would also be used to determine the contact state of the symmetric leg, the right frontal leg, and vice-versa.

F. Equivariant Conv1D layers

For details on the construction of the Equivariant 1D Convolutional layers reefer to [1]. Note that the symmetry of a single time-sample $\boldsymbol{z}_i$ is shared across all time-samples $\boldsymbol{y} = \{\boldsymbol{z}_i\}_{i=0}^{150}$.

APPENDIX D
LIMITATIONS

Our work makes two main assumptions:

1) **Symmetries are exact**: By assuming that a dynamical system has exact and not approximate symmetries we are departing from the real-world nature of DMSs since for any robotic system in the real-world the manufacturing and assembly process introduces errors/tolerances in the kinematic and dynamic parameters of each of the robot's bodies. Likewise, the dynamics of animals in nature are not perfectly equivariant since morphological symmetries are only approximate symmetries. Although exact symmetries seem to be a strong assumption, in practice, the reality is that it is a common assumption in the fields of robotics and control theory, in which idealized models of the dynamics are often assumed (in simulation and real-world).

   On section VI we show that the exact symmetry bias is justifiable and beneficial for learning function approximators processing the dynamics of approximately symmetrical systems in the real world. However, the authors highlight the necessity to properly address the case of approximate equivariance, which we leave to future work. To address this case, system identification techniques [22] have been wildly used to approximate the deviation of the kinematic and dynamic parameters from the assumed values. While in the case of $\mathcal{G}$-equivariant NN Wang et al. [24], Finzi et al. [5] provide

| **Tri-Finger Robot** $\mathcal{G} = \mathcal{C}_3$ | **Bolt Bipedal Robot** $\mathcal{G} = \mathcal{C}_2$ |
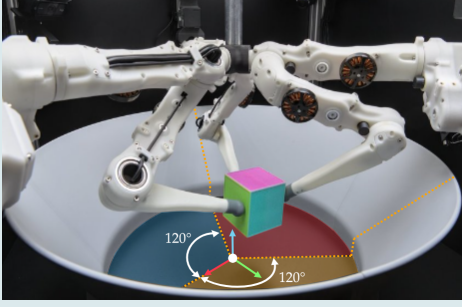
This fixed-based robot is symmetric w.r.t. rotations of space by $\theta = 120°$ in the vertical axis. Therefore, its symmetry group is the cyclic group of order three ($\mathcal{G} = \mathcal{C}_3$). To identify this symmetry group we apply the procedure in section IV-C:

1) **Identify X_B and $\mathbf{I}_B$**: As a fix-based robot, we define X_B to be the mounting structure supporting each finger, and the gray disk delimiting the workspace (see image).
2) **Identify symmetries of $\mathbf{I}_B$**: The inertia of this virtual base $\mathbf{I}_B$ is invariant to rotations by $120°$ in the vertical axis. I.e., $\mathbf{I}_B$ is invariant to $X_B \rho_{\mathbb{E}_3}(\overline{g})^{-1} | \overline{g} \in \{e, \overline{g}_\theta, \overline{g}_\theta^2\} \equiv \mathcal{C}_3$ (eq. (6)).
3) **Identify modularity in the kinematic tree**: There are three symmetric kinematic subchains. Each finger is composed of replicated versions of the same bodies.
4) **Identify the DMS group $\mathcal{G}$**:

Consider that the transformation $\rho_{\mathbb{E}_3}(g)X_B \doteq \rho_{\mathbb{E}_3}(\overline{g})X_B\rho_{\mathbb{E}_3}(\overline{g})^{-1}$ (eq. (6)) can be interpreted as a rotation of the virtual base by $\theta°$ followed by a rotation $-\theta°$ in the z axis. Thus respecting the fix-base constraint of the system. Denote the joint-space $q = \hat{q} = [q_{f1}^\mathsf{T}, q_{f2}^\mathsf{T}, q_{f3}^\mathsf{T}]^\mathsf{T}$ be composed of each finger's DoF ($q_{fi} \in \mathbb{R}^3$).

Then we can define $\rho_{\mathrm{Q}_J}(g) \doteq \rho_{\mathbb{R}^3}(\overline{g}) \otimes \mathbf{I}_3 \mid \overline{g} \in \mathcal{C}_3$. Being $\rho_{\mathbb{R}^3}(\cdot)$ the permutation representation of 3 elements of $\mathcal{C}_3$ (3 kinematic subchains). For the generator action of the group this is $\rho_{\mathbb{R}^3}(\overline{g}) = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{bmatrix}$.

Lastly, we verify if $\mathcal{G} = \mathcal{C}_3$ by testing all tentative group actions for DMSs eq. (5).

Augmentation of data samples: Say we collect a dataset of robot states ($q, \dot{q}$) and cube states X_C at every time step t, to train the manipulation policy [7]. To obtain the symmetric states, at every t, we need to understand that since we are imitating the effect of a true rotation of space $\overline{g}$, the symmetric states are obtained by ($g \cdot q, g \cdot \dot{q}$) and ($\overline{g} \cdot X_C \doteq \rho_{\mathbb{E}_3}(\overline{g})X_C$).
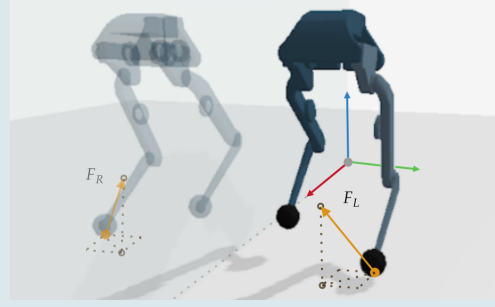
Bolt is a bipedal robot with a sagittal plane reflection symmetry ($\mathcal{G} = \mathcal{C}_2$). This morphological symmetry allows it to imitate the effect of arbitrary reflections of space ($\overline{g} \in \mathbb{E}_3$) by re-configuring its base and legs. To identify this symmetry group we apply the procedure in section IV-C:

1) **Identify X_B and $\mathbf{I}_B$**: X_B is the robot base (hips) body, with its corresponding inertia $\mathbf{I}_B$
2) **Identify symmetries of $\mathbf{I}_B$**: The base body has symmetrical mass distribution w.r.t the sagittal plane. Thus, $\mathbf{I}_B$ is invariant to the transformation $X_B\rho_{\mathbb{E}_3}(\overline{g})^{-1}|\overline{g} \in \{e, \overline{g}_s\} \equiv \mathcal{C}_2$ (eq. (6)).
3) **Identify modularity in the kinematic tree**: There are two symmetric kinematic subchains. The left leg subchain and bodies are reflected versions of the right leg subchain and bodies.
4) **Identify the DMS group $\mathcal{G}$**:

Since a reflection w.r.t to the sagittal plane would imply a true reflection of the rigid bodies of the legs, we need to *permute* each body in the kinematic tree with each reflected version. Denote the joint-space $\hat{q} = [q_L^\mathsf{T}, q_R^\mathsf{T}]^\mathsf{T}$ as composed of the left L and right R legs' DoF ($q_{L/R} \in \mathbb{R}^3$). Denote the sign-relation between the DoF of the Left and right legs' degrees of freedom as $s_{L|R} \in \mathbb{R}^3$.

Then we can define $\rho_{\mathrm{Q}_J}(g) \doteq \rho_{\mathbb{R}^2}(\overline{g}) \otimes (s_{L|R}\mathbf{I}_3) \mid \overline{g} \in \mathcal{C}_2$. Being $\rho_{\mathbb{R}^2}(\cdot)$ the permutation representation of a 2 elements of $\mathcal{C}_2$ (2 kinematic subchains). For the non-trivial action of the group this is $\rho_{\mathbb{R}^2}(\overline{g}_s) = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$.

Lastly, recalling the definition of $\rho_{\mathbb{E}_3}(g)$ in eq. (6), we verify if $\mathcal{G} = \mathcal{C}_2$ by testing all tentative group actions for DMSs eq. (5).

Augmentation of data samples: Say we collect a dataset of robot states ($q, \dot{q}$) and ground reaction forces (f_L, f_R), that we transform to the space of generalized forces as (τ_{f_L}, τ_{f_R}), at every timestep t. This dataset can be used to train a reactive locomotion policy as in Ordonez-Apraez et al. [16]. The symmetric states, at every t, are then defined as: ($g \cdot q, g \cdot \dot{q}$) and ($g \cdot \tau_{f_L}, g \cdot \tau_{f_R}$) $\equiv$ ($\rho_{\mathrm{Q}}(g)\tau_{f_L}, \rho_{\mathrm{Q}}(g)\tau_{f_R}$)

Fig. 6: **Tutorial** example morphological symmetries of the Tri-Finger [7] and Bolt robots.
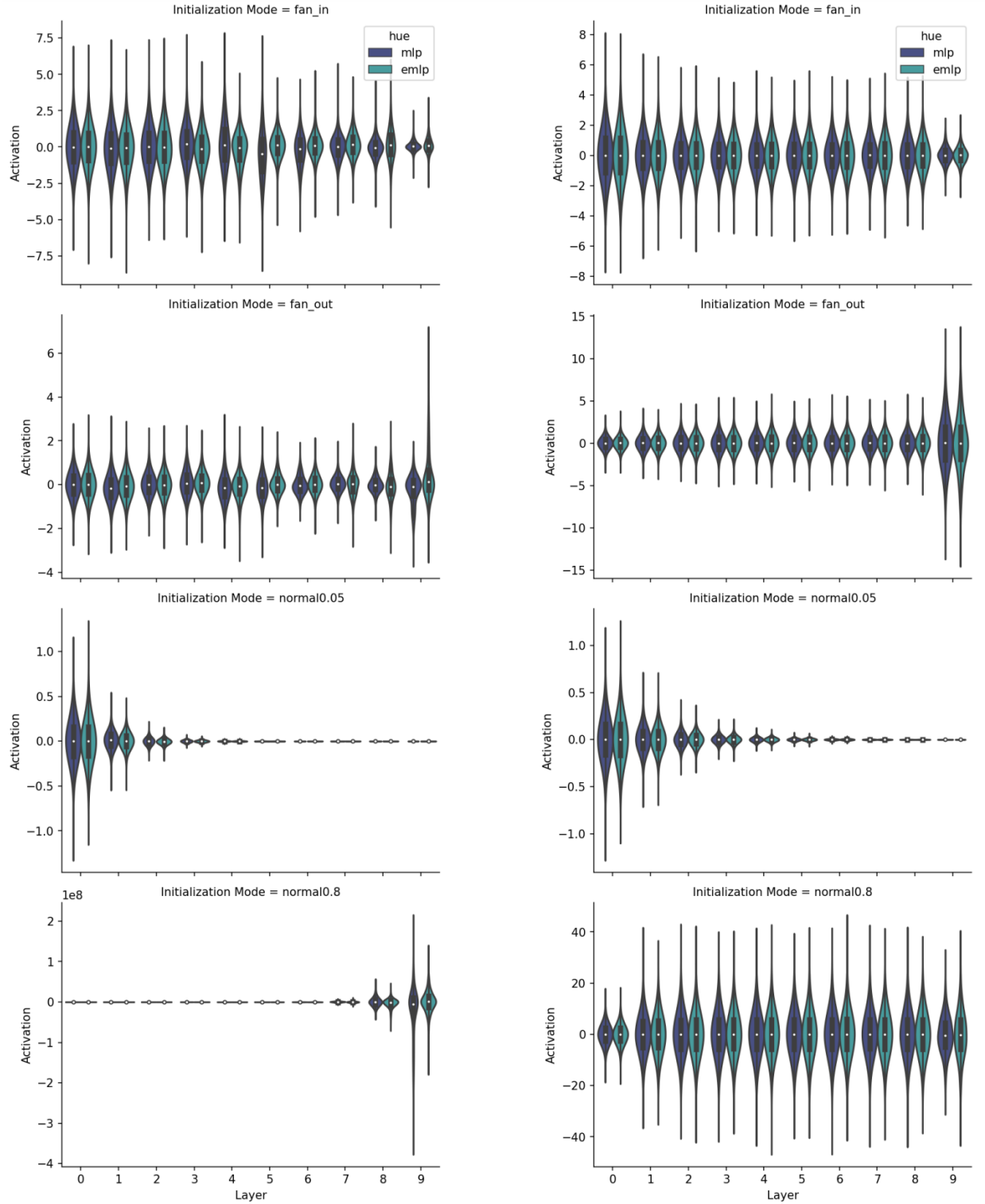
Fig. 7: Comparison of the initialization method of unconstrained layers of He et al. [10] with our initialization method for equivariant layers. Left and right columns correspond to MLP & EMLP architectures with $\sigma = ReLu$ (left) and $\sigma = Tanh$ (right) non-linearities. Each row shows different initialization methods differing in the variance of the initialization distribution of the layer's trainable parameters. First and second rows show the forward and backward propagation cases of [10] for MLP and of section B-B for EMLP, with $\mathrm{Var}(^{l}c) \doteq {m}/{(\lambda_{l_{\mathbf{B}}}\lambda_{\sigma})}$ and $\mathrm{Var}(^{l}c) \doteq {n}/{(\lambda_{l_{\mathbf{B}}}\lambda_{\sigma})}$, respectively. In these cases, the variance of activations through the network depth remains nearly constant, as desired. The last two rows show the initialization of layer parameters with a constant variance of 0.05^2 and 0.8^2, illustrating scenarios of activations vanishings and exploiting. All intermediate layers have 256 neurons. In the equivariant case, the network is $\mathcal{K}_4$-equivariant.

clear and valuable approaches to learn approximate $\mathcal{G}$-equivariant NN.

It is relevant to highlight that, in physics-based simulation, the most common practice is to work with the idealized model of dynamics. Thus, the assumption of exact symmetries is justifiable and encouraged in applications where simulation is a relevant tool.

2) **Symmetry group is finite**: This work narrows its focus to DMSs, even do the definition of DMS eq. (5) can be relaxed to cover continuous morphological symmetries (an example of this continuous symmetry is the capability of a robot arm manipulator to imitate rotations of space by rotating its first degree of freedom). This assumption allows us to compute in linear time the basis **B** of equivariant linear maps (restricted to groups with finite regular matrix representations).