FRAME AVERAGING FOR INVARIANT AND EQUIVARIANT NETWORK DESIGN

Anonymous authors
Paper under double-blind review

ABSTRACT

Many machine learning tasks involve learning functions that are known to be invariant or equivariant to certain symmetries of the input data. However, it is often challenging to design neural network architectures that respect these symmetries while being expressive and computationally efficient. For example, Euclidean motion invariant/equivariant graph or point cloud neural networks.
We introduce Frame Averaging (FA), a general purpose and systematic framework for adapting known (backbone) architectures to become invariant or equivariant to new symmetry types. Our framework builds on the well known group averaging operator that guarantees invariance or equivariance but is intractable. In contrast, we observe that for many important classes of symmetries, this operator can be replaced with an averaging operator over a small subset of the group elements, called a frame. We show that averaging over a frame guarantees exact invariance or equivariance while often being much simpler to compute than averaging over the entire group. Furthermore, we prove that FA-based models have maximal expressive power in a broad setting and in general preserve the expressive power of their backbone architectures. Using frame averaging, we propose a new class of universal Graph Neural Networks (GNNs), universal Euclidean motion invariant point cloud networks, and Euclidean motion invariant Message Passing (MP) GNNs. We demonstrate the practical effectiveness of FA on several applications including point cloud normal estimation, beyond 2-WL graph separation, and n-body dynamics prediction, achieving state-of-the-art results in all of these benchmarks.

1 INTRODUCTION

Many tasks in machine learning (ML) require learning functions that are invariant or equivariant with respect to symmetric transformations of the data. For example, graph classification is invariant to a permutation of its nodes, while node prediction tasks are equivariant to node permutations. Consequently, it is important to design expressive neural network architectures that are by construction invariant or equivariant for scalable and efficient learning. This recipe has proven to be successful for many ML tasks including image classification and segmentation (LeCun et al., 1998; Long et al., 2015), set and point-cloud learning (Zaheer et al., 2017; Qi et al., 2017a), and graph learning (Kipf & Welling, 2016; Gilmer et al., 2017; Battaglia et al., 2018).

Nevertheless, for some important instances of symmetries, the design of invariant and/or equivariant networks is either illusive (Thomas et al., 2018; Dym & Maron, 2020), computationally expensive or lacking in expressivity (Xu et al., 2018a; Morris et al., 2019; Maron et al., 2019; Murphy et al., 2019). In this paper, we propose a new general-purpose framework, called Frame Averaging (FA), that can systematically facilitate expressive invariant and equivariant networks with respect to a broad class of groups. At the heart of our framework, we build on a basic fact that arbitrary functions $\phi : V \to \mathbb{R}$, $\Phi : V \to W$, where V, W are some vector spaces, can be made invariant or equivariant by *symmetrization*, that is averaging over the group (Yarotsky, 2021; Murphy et al., 2018), i.e.,

$$\psi(X) = \frac{1}{|G|} \sum_{g \in G} \phi(g^{-1} \cdot X) \quad \text{or} \quad \Psi(X) = \frac{1}{|G|} \sum_{g \in G} g \cdot \Phi(g^{-1} \cdot X). \tag{1}$$

where $G = \{g\}$ denotes the group, $\psi : V \to \mathbb{R}$ is invariant and $\Psi : V \to W$ is equivariant with respect to G. Furthermore, since invariant and equivariant functions are fixed under group averaging,

i.e., $\psi = \phi$ for invariant ϕ and $\Psi = \Phi$ for equivariant Φ, the above scheme often leads to universal (i.e., maximally expressive) models (Yarotsky, 2021). However, the challenge with equation 1 is that when the cardinality of G is large (e.g., combinatorial groups such as permutations) or infinite (e.g., continuous groups such as rotations), then exact averaging is intractable. In such cases, we are forced to approximate the sum via heuristics or Monte Carlo (MC), thereby sacrificing the exact invariance/equivariance property for computational efficiency, e.g., Murphy et al. (2018; 2019) define heuristic averaging strategies for approximate permutation invariance in GNNs; similarly, Hu et al. (2021) and Shuaibi et al. (2021) use MC averaging for approximate rotation equivariance in GNNs.

The key observation of the current paper is that the group average in equation 1 can be replaced with an average over a carefully selected subset $\mathcal{F}(X) \subset G$ while retaining both exact invariance/equivariance and expressive power. Therefore, if $\mathcal{F}$ can be chosen so that the cardinality $|\mathcal{F}(X)|$ is mostly small, averaging over $\mathcal{F}(X)$ results in both expressive and efficient invariant/equivariant model. We call the set-valued function $\mathcal{F} : V \to 2^G$, a *frame*, and show that it can successfully replace full group averaging if it satisfies a *set equivariance property*. We name this framework Frame Averaging (FA) and it serves as the basis for the design of invariant/equivariant networks in this paper.

We instantiate the FA framework by considering different choices of symmetry groups G, their actions on data spaces V, W (manifested by choices of group representations), and the backbone architectures (or part thereof) ϕ, Φ we want to make invariant/equivariant to G. We consider: (i) Multi-Layer Perceptrons (MLP), and Graph Neural Networks (GNNs) with node identification (Murphy et al., 2019; Loukas, 2020) adapted to permutation invariant Graph Neural Networks (GNNs); (ii) Message-Passing GNN (Gilmer et al., 2017) adapted to be invariant/equivariant to Euclidean motions, $E(d)$; (iii) Set network, DeepSets and PointNet (Zaheer et al., 2017; Qi et al., 2017a) adapted to be equivariant or *locally* equivariant to $E(d)$; (iv) Point cloud network, DGCNN (Wang et al., 2018), adapted to be equivariant to $E(d)$.

Theoretically, we prove that the FA framework maintains the expressive power of its original backbone architecture which leads to some interesting corollaries: First, (i) results in invariant universal graph learning models; (ii) is an $E(d)$ invariant/equivariant GNN that maintain the power of message passing (Xu et al., 2018a; Morris et al., 2019); and (iii), (iv) furnish a universal permutation and $E(d)$ invariant/equivariant models. We note that both the construction and the proofs are arguably considerably simpler than the existing alternative constructions and proofs for this type of symmetry (Thomas et al., 2018; Fuchs et al., 2020; Dym & Maron, 2020). We experimented with FA on different tasks involving symmetries including: point-cloud normal estimation, beyond 2-Weisfeiler-Lehman graph separation, and n-body dynamics predictions, reaching state of the art performance in all.

2 Frame Averaging

In this section we introduce the FA approach and theory in generic formulation, while in the next section we show the power of this general approach by instantiating it to (seemingly different) problems of interest.

2.1 Frame averaging for function symmetrization

Let $\phi : V \to \mathbb{R}$ and $\Phi : V \to W$ be some arbitrary functions, where V, W are (finite dimensional) normed linear spaces. For example, ϕ, Φ can be thought of as neural networks. We consider a group $G = \{g\}$ that describes some symmetry we want to incorporate into ϕ, Φ. The way the symmetries $g \in G$ are applied to vectors in V, W is described by the group's *representations* $\rho_1 : G \to \mathrm{GL}(V)$, and $\rho_2 : G \to \mathrm{GL}(W)$, where $\mathrm{GL}(V)$ is the space of invertible linear maps $V \to V$ (automorphisms). A representation ρ_i preserves the group structure by satisfying $\rho_i(gh) = \rho_i(g)\rho_i(h)$ for all $g, h \in G$ (see e.g., Fulton & Harris (2013)). As customary, we will sometimes refer to the linear spaces V, W as representations. Particular examples of G, V, W, ρ_i are presented in the Section 3.

Our goal is to make ϕ into an *invariant* function, namely satisfy $\phi(\rho_1(g)X) = \phi(X)$, for all $g \in G$ and $X \in V$; and Φ into an *equivariant* function, namely $\Phi(\rho_1(g)X) = \rho_2(g)\Phi(X)$, for all $g \in G$ and $X \in V$. We will do that by averaging over group elements, but instead of averaging over the entire group every time (as in equation 1) we will average on a subset of the group elements called a *frame*.

Definition 1. *A* frame *is defined as a set valued function* $\mathcal{F} : V \to 2^G \setminus \emptyset$.

1. *A frame is* G-equivariant *if*
$$\mathcal{F}(\rho_1(g)X) = g\mathcal{F}(X), \quad \forall X \in V,\ g \in G, \tag{2}$$
where as usual, $g\mathcal{F}(X) = \{gh \mid h \in \mathcal{F}(X)\}$*, and the equality in equation 2 should be understood as equality of sets.*

2. *A frame is* bounded *over a domain* $K \subset V$ *if there exists a constant* $c > 0$ *so that* $\|\rho_2(g)\| \leq c$*, for all* $g \in \mathcal{F}(X)$ *and all* $X \in K$*, where* $\|\cdot\|$ *denotes the induced operator norm over* W.

Figure 1 provides an illustration. How are equivariant frames useful? Consider a scenario where an equivariant frame is easy to compute, and furthermore its cardinality, $|\mathcal{F}(X)|$, is not too large. Then averaging over the frame, denoted $\langle\cdot\rangle_{\mathcal{F}}$ and defined by

$$\langle\phi\rangle_{\mathcal{F}}(X) = \frac{1}{|\mathcal{F}(X)|}\sum_{g\in\mathcal{F}(X)} \phi(\rho_1(g)^{-1}X) \tag{3}$$

$$\langle\Phi\rangle_{\mathcal{F}}(X) = \frac{1}{|\mathcal{F}(X)|}\sum_{g\in\mathcal{F}(X)} \rho_2(g)\Phi(\rho_1(g)^{-1}X) \tag{4}$$

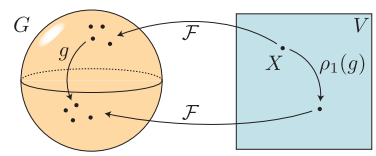

Figure 1: Frame equivariance (sphere shape represents the group G; square represents V).

provides the required function symmetrization. The following theorem formalizes the frame based symmetrization framework:

Theorem 1 (Frame Averaging)**.** *Let* $\mathcal{F}$ *be a* G *equivariant frame, and* $\phi : V \to \mathbb{R}$*,* $\Phi : V \to W$ *some functions. Then,* $\langle\phi\rangle_{\mathcal{F}}$ *is* G *invariant, while* $\langle\Phi\rangle_{\mathcal{F}}$ *is* G *equivariant.*

The proof is provided in Appendix A.1. One trivial choice of a frame is $\mathcal{F}(X) \equiv G$, that is, taking the frame to be the entire group for all $X \in V$. This choice can be readily checked to be equivariant, and turns the FA equations 3, 4 into standard group averaging operators, equation 1. The problem with this choice, however, is that it often results in an intractable or challenging computation, e.g., when the group is large or infinite. In contrast, as we show below, in some useful cases one can compute a manageable size frame and can use it to build invariant or equivariant operators in a principled way. In this paper, we focus on finite frames, we discuss infinite frames shortly in Appendix B.

Incorporating G as a second symmetry. An important use case of frame averaging is with the backbones ϕ, Φ already invariant/equivariant w.r.t. some symmetry group H and our goal is to make it invariant/equivariant to $H \times G$. For example, say we want to add $G = E(3)$ equivariance to permutation invariant set or graph functions, i.e., $H = S_n$. We will provide sufficient conditions for the FA to provide this desired invariance/equivariance. First, let us assume H is acting on V and W by the representations $\tau_1 : H \to \mathrm{GL}(V)$ and $\tau_2 : H \to \mathrm{GL}(W)$, respectively. Assume ϕ is H invariant and Φ is H equivariant. We say that representations ρ_1 and τ_1 *commute* if $\rho_1(g)\tau_1(h)X = \tau_1(h)\rho_1(g)X$ for all $g \in G$, $h \in H$, and $X \in V$. If ρ_1 and τ_1 commute then the map $\gamma_1 : H \times G \to \mathrm{GL}(V)$ defined by $\gamma_1(h, g) = \tau_1(h)\rho_1(g)$ is a representation of the group $H \times G$. Second, we would need that the frame $\mathcal{F}(X)$ is *invariant* to H, that is $\mathcal{F}(\tau_1(h)X) = \mathcal{F}(X)$. We show a generalization of Theorem 1:

Theorem 2 (Frame Average second symmetry)**.** *Assume* $\mathcal{F}$ *is* H*-invariant and* G*-equivariant. Then,*

1. *If* $\phi : V \to \mathbb{R}$ *is* H *invariant and* ρ_1, τ_1 *commute then* $\langle\phi\rangle_{\mathcal{F}}$ *is* $G \times H$ *invariant.*

2. *If* $\Phi : V \to W$ *is* H *equivariant and* ρ_i, τ_i*,* $i = 1,2$*, commmute then* $\langle\Phi\rangle_{\mathcal{F}}$ *is* $G\times H$ *equivariant.*

Right actions. Above we used left actions for the definition of equivariance. There are other flavors of equivariance, e.g., if one of the actions is right. For example, if g multiplies $\mathcal{F}(X)$ from the right, then equivariance will take the form:

$$\mathcal{F}(\rho_1(g)X) = \mathcal{F}(X)g^{-1}, \quad \forall X \in V,\ g \in G \tag{5}$$

and accordingly

$$\langle\phi\rangle_{\mathcal{F}}(X) = \frac{1}{|\mathcal{F}(X)|}\sum_{g\in\mathcal{F}(X)} \phi(\rho_1(g)X), \quad \langle\Phi\rangle_{\mathcal{F}}(X) = \frac{1}{|\mathcal{F}(X)|}\sum_{g\in\mathcal{F}(X)} \rho_2(g)^{-1}\Phi(\rho_1(g)X) \tag{6}$$

are G invariant and equivariant, respectively.

Efficient calculation of invariant frame averaging. There could be instances of the FA framework (indeed we discuss such a case later) where $|\mathcal{F}(X)|$ is still too large to evaluate equations 3,4. In the invariant case, there is a more efficient form of FA, that can potentially be applied. To show it, let us start by defining the subgroup of symmetries of X, i.e., its stabilizer. The stabilizer of an element $X \in V$ is a subgroup of G defined by $G_X = \{g \in G \mid \rho_1(g)X = X\}$. G_X naturally induces an equivalence relation $\sim$ on $\mathcal{F}(X)$, with $g \sim h \iff hg^{-1} \in G_X$. The equivalence classes (orbits) are $[g] = \{h \in \mathcal{F}(X) | g \sim h\} = G_X g \subset \mathcal{F}(X)$, for $g \in \mathcal{F}(X)$, and the quotient set is denoted $\mathcal{F}(X)/G_X$.

Theorem 3. *Equivariant frame $\mathcal{F}(X)$ is a disjoint union of equal size orbits, $[g] \in \mathcal{F}(X)/G_X$.*

The proof is in A.3. The first immediate consequence of Theorem 3 is that the cardinality of $\mathcal{F}(X)$ is at-least that of the stabilizer (intuitively, the inner-symmetries) of X, namely $|\mathcal{F}(X)| \geq |G_X|$. Therefore, there could be cases, such as when X describes a symmetric graph, where $|\mathcal{F}(X)|$ could be too large to average over. A remedy comes from the following observation: for every $h \in [g]$, we have that $h = rg$, $r \in G_X$, and $\phi(\rho_1(h)^{-1}X) = \phi(\rho_1(g)^{-1}\rho_1(r)^{-1}X) = \phi(\rho_1(g)^{-1}X)$, since also $r^{-1} \in G_X$. Therefore the summands in equations 3 are constant over orbits, and we get

$$\langle \phi \rangle_{\mathcal{F}}(X) = \frac{1}{m_{\mathcal{F}}} \sum_{[g] \in \mathcal{F}(X)/G_X} \phi(\rho_1(g)^{-1}X), \tag{7}$$

where $m_{\mathcal{F}} = |\mathcal{F}(X)/G_X| = |\mathcal{F}(X)|/|G_X|$. This representation of invariant FA requires only $m_{\mathcal{F}} = |\mathcal{F}(X)|/|G_X|$ evaluations, compared to $|\mathcal{F}(X)|$ in the original FA in equation 3.

Approximation of invariant frame averaging. Unfortunately, enumerating $\mathcal{F}(X)/G_X$ could yet be challenging. Nevertheless, equation 7 is still very useful: it turns out we can easily *draw a random element* from $\mathcal{F}(X)/G_X$ with uniform probability. This is an immediate application of the equal orbit size in Theorem 3:

Corollary 1. *Let $\mathcal{F}(X)$ be an equivariant frame, and $g \in \mathcal{F}(X)$ be a uniform random sample. Then $[g] \in \mathcal{F}(X)/G_X$ is also uniform.*

Therefore, an efficient approximation strategy is averaging over uniform samples, $g_i \in \mathcal{F}(X)$, $i \in [k]$,

$$\langle\!\langle \phi \rangle\!\rangle_{\mathcal{F}}(X) = \frac{1}{k} \sum_{i=1}^{k} \phi(\rho_1(g_i)^{-1}X). \tag{8}$$

This approximation is especially useful, compared to the full-blown FA, when $m_{\mathcal{F}} = |\mathcal{F}(X)|/|G_X|$ is small, i.e., when $|G_X|$ is large, or X has many symmetries. To better understand the role of $m_{\mathcal{F}}$ in approximating equation 7 we first note that equation 8 can be written as $\langle\!\langle \phi \rangle\!\rangle_{\mathcal{F}}(X) = \sum_{[g] \in \mathcal{F}(X)/G_X} \hat{\mu}_{[g]} \phi(\rho_1(g_i)^{-1}X)$, where $\hat{\mu}$ is the empiricial distribution over $\mathcal{F}(X)/G_X$, assigning to each element $[g] \in \mathcal{F}(X)/G_X$ the fraction of samples landed in $[g]$, i.e., $\hat{\mu}_{[g]} = |\{i \in [k] | g_i \in [g]\}| k^{-1}$. We next present a lower bound on the probability of any particular $\hat{\mu}$ that provides a good approximation $\langle\!\langle \phi \rangle\!\rangle_{\mathcal{F}}(X) \approx \langle \phi \rangle_{\mathcal{F}}(X)$.

Theorem 4. *Let $\mathcal{F}$ be an equivariant frame. The probability of an arbitrary $\hat{\mu} \in \left\{\hat{\mu} \,\middle|\, \sup_{\phi \in \mathcal{Q}} |\langle \phi \rangle_{\mathcal{F}}(X) - \langle\!\langle \phi \rangle\!\rangle_{\mathcal{F}}(X)| \leq \epsilon\right\}$ is bounded from below as follows,*

$$P(\hat{\mu}) \geq (1+k)^{-m_{\mathcal{F}}} \exp(-2m_{\mathcal{F}} k \epsilon^2),$$

where $\mathcal{Q} = \{\phi \in \mathcal{C}(V, \mathbb{R}) \mid |\phi(X)| \leq 1, \forall X \in V\}$ the set of bounded, continuous functions $V \to \mathbb{R}$.

The main takeoff from Theorem 4 is that smaller $m_{\mathcal{F}}$ increases the guaranteed probability for a good approximation $\langle\!\langle \phi \rangle\!\rangle_{\mathcal{F}}(X) \approx \langle \phi \rangle_{\mathcal{F}}(X)$. We demonstrate this numerically in Section 5.2.

2.2 Expressive power

Another benefit in frame averaging as presented in equations 3 and 4 is that it preserves the expressive power of the base models, ϕ, Φ. Consider a function space $\mathcal{H} = \{\Phi\} \subset \mathcal{C}(V, W)$ consisting of continuous functions $V \to W$. The case of scalar functions ϕ is a special case where $W = \mathbb{R}$, and $\rho_2(g) = 1$ for all $g \in G$. $\mathcal{H}$ can be seen as the collection of all functions represented by a certain class of neural networks, e.g., Multilayer Perceptron (MLP), or DeepSets (Zaheer et al., 2017), or message passing (Gilmer et al., 2017). Let $\bar{\mathcal{H}}$ denote the closure of this function space under the maximum norm evaluated on a compact set $K \subset V$, that is

$$\|\Phi\|_{\infty,K} = \max_{X \in K} \|\Phi(X)\|,$$

where the latter norm is the norm in W. We further assume K is invariant, that is $\rho_1(g)K = K$, for all $g \in G$. Intuitively $\bar{\mathcal{H}}$ consists of all functions that can be uniformly approximated with functions from $\mathcal{H}$. We denote by $\langle\mathcal{H}\rangle$ the collection of functions $\Phi \in \mathcal{H}$ after applying the frame averaging in equation 4,

$$\langle\mathcal{H}\rangle = \{\langle\Phi\rangle_{\mathcal{F}} \mid \Phi \in \mathcal{H}\}.$$

Theorem 5 (proof in Appendix A.5) shows that $\langle\mathcal{H}\rangle$ is at-least *as expressive as* its backbone $\mathcal{H}$. What does it mean? We will show that $\langle\mathcal{H}\rangle$, which is equivariant by construction, can approximate any G-equivariant function that is approximable with our backbone model $\mathcal{H}$. That is,

$$\overline{\langle\mathcal{H}\rangle} \supset \bar{\mathcal{H}} \cap \mathcal{E},$$

where $\mathcal{E}$ is the collection of G-equivariant functions in $\mathcal{C}(V, W)$.

Theorem 5 (Expressive power of FA)**.** *If $\mathcal{F}$ is a bounded G equivariant frame over an invariant domain K then $\langle\mathcal{H}\rangle$ is at-least as expressive as $\mathcal{H}$. That is, $\overline{\langle\mathcal{H}\rangle} \supset \bar{\mathcal{H}} \cap \mathcal{E}$.*

This theorem can be used to prove universality results if the backbone model is universal. Below we will use it to prove universality of frame averaged architectures for graphs, and point clouds with Euclidean motion symmetry.

3 Model instances

In this section we instantiate the frame averaging framework for the design of several invariant/equivariant architectures. We will consider the following variants of the framework: i) The symmetry group G, representations ρ_1, ρ_2 and the underlying frame $\mathcal{F}$; ii) The backbone network architecture for ϕ (invariant case) or Φ (equivariant case), based on components from existing models.

3.1 Point clouds, Euclidean motions.

Symmetry and frame choice. We would like to incorporate Euclidean symmetry to existing permutation invariant/equivaraint point cloud networks. The symmetry of interest is $G = E(d) = O(d) \rtimes T(d)$, namely the group of Euclidean motions in $\mathbb{R}^d$ defined by rotations and reflections $O(d)$, and translations $T(d)$. We also discuss $G = SE(d) = SO(d) \rtimes T(d)$, where $SO(d)$ is the group of rotations in $\mathbb{R}^d$. We define $V = \mathbb{R}^{n\times d}$, and the group representation is $\rho_1(g)\boldsymbol{X} = \boldsymbol{X}(\boldsymbol{R}^T + \mathbf{1}\boldsymbol{t}^T)$, where $\boldsymbol{R} \in O(d)$ or $\boldsymbol{R} \in SO(d)$, and $\boldsymbol{t} \in \mathbb{R}^d$ denotes the translation. W, ρ_2 are defined similarly, unless translation invariance is desired in which case we use the representation $\rho_2(g)\boldsymbol{X} = \boldsymbol{X}\boldsymbol{R}^T$. For $G = E(d)$, we define the frame $\mathcal{F}(\boldsymbol{X})$ based on Principle Component Analysis (PCA), i.e., the collection of $E(3)$ Euclidean transformations defined by pairs $(\boldsymbol{R}, \boldsymbol{t})$, where $\boldsymbol{t} = \frac{1}{n}\boldsymbol{X}^T\mathbf{1}$ is the centroid of $\boldsymbol{X}$, and $\boldsymbol{R} \in O(3)$ are all the orthogonal 3×3 matrices consisting of eigenvectors (ordered by decreasing order of eigenvalues) of the covariance matrix $\boldsymbol{C} = \boldsymbol{X}^T(\boldsymbol{I} - \frac{1}{n}\mathbf{1}\mathbf{1}^T)\boldsymbol{X}$. Note that generically the frame would consist of 2^d elements which for typical dimensions $d = 2, 3$ amount to frames of size $4, 8$, respectively, while for rare inputs $\boldsymbol{X}$ the frame can be an infinite set (e.g., isomorphic to $O(d-1)$). For $G = SE(d)$, $\boldsymbol{R}$ can be taken as orthogonal, positive orientation matrices; generically there are 2^{d-1} such elements, which amounts to $2, 4$ elements for $d = 2, 3$, respectively.

Proposition 1. *The frames $\mathcal{F}(\boldsymbol{X})$ based on the covariance and centroid are $E(d)$ or $SE(d)$ equivariant and bounded.*

Since we would like to incorporate $E(d)$ symmetries to an already S_n invariant/equivariant architectures, per Theorem 2, we will also need to show that the ρ_1 (and similarly ρ_2) commute with $\tau : S_n \to \mathrm{GL}(V)$ defined by $\tau(h)\boldsymbol{X} = \boldsymbol{P}\boldsymbol{X}$, where $\boldsymbol{P} = \boldsymbol{P}_h$, $h \in S_n$, is the permutation representation. That is $\boldsymbol{P}_h \in \mathbb{R}^{n\times n}$ is the permutation matrix representing $h \in S_n$, that is, $\boldsymbol{P}_{ij} = 1$ if $i = h(j)$ and 0 otherwise. Indeed $\tau(h)\rho_1(g)\boldsymbol{X} = \boldsymbol{P}\boldsymbol{X}(\boldsymbol{R}^T + \mathbf{1}\boldsymbol{t}^T) = \rho_1(g)\tau(h)\boldsymbol{X}$. Furthermore, note that $\mathcal{F}(\tau(h)\boldsymbol{X}) = \mathcal{F}(\boldsymbol{X})$, therefore $\mathcal{F}$ is also S_n invariant.

Backbone architectures. We incorporate the FA framework with two existing popular point cloud network layers: i) PointNet (Qi et al., 2017a); and ii) DGCNN (Wang et al., 2018). We denote both architectures by $\Phi_{d,d'} : \mathbb{R}^{n\times d} \to \mathbb{R}^{n\times d'}$ for the S_n equivariant version of these models. To simplify the discussion, we omit particular choices of layers and feature dimensions, Appendix C.1 provides the full details. We experimented with two design choices: First, consider frame averaged $\Phi_{3,3}$, i.e.,

$\langle \Phi_{3,3} \rangle_{\mathcal{F}}$, yielding a universal, $E(3)$ equivariant versions for PointNet and DGCNN, dubbed FA-PointNet and FA-DGCNN. A more complex design choice, taking inspiration from deep architectures with multiple modules, is to compose blocks of FA networks. For example, to build a *local* $E(3)$ equivariant version of PointNet, denote also $\Upsilon_{d,d'} : \mathbb{R}^d \to \mathbb{R}^{d'}$ an MLP. We decompose the input point cloud to k-nn patches $\boldsymbol{X}_i \in \mathbb{R}^{k\times 3}$, $i \in [n]$, where each patch is sorted by distance. Next, feed each patch into an equivariant FA $\Upsilon_{3k,3d}$, each with its own frame $\mathcal{F}_i$, resulting in $E(3)$ equivariant features in $\mathbb{R}^{3d}$ for every point, $\boldsymbol{Y} \in \mathbb{R}^{n\times 3d}$; then applying equivariant FA $\Phi_{3d,3}$ providing output in $\mathbb{R}^{n\times 3}$. That is, $\Psi(\boldsymbol{X}) = \langle \Phi_{3d,3} \rangle_{\mathcal{F}} \left(\left[\langle \Upsilon_{3k,3d} \rangle_{\mathcal{F}_1} (\boldsymbol{X}_1), \ldots, \langle \Upsilon_{3k,3d} \rangle_{\mathcal{F}_n} (\boldsymbol{X}_n) \right] \right)$, where brackets denote concat in the first dimension. We name this construction FA-Local-PointNet.

Universality. We use Theorem 5 to prove that using a universal set invariant/equivariant backbone ϕ, Φ, such as DeepSets or PointNet (see e.g., Zaheer et al. (2017); Qi et al. (2017a); Segol & Lipman (2019)) leads to a universal model. Let $\mathcal{H}$ be any such universal set-equivariant function collection. Hence $\bar{\mathcal{H}}$ is the collection of all continuous set functions (i.e., S_n invariant or equivariant) over K. Therefore, $\bar{\mathcal{H}} \cap \mathcal{E}$ is the set of *all* continuous set functions over K that are also $E(d)$ equivariant. Consequently, Theorem 5 implies:

Corollary 2 (FA-DeepSets/PointNet are universal)**.** *Given a bounded $E(d)$ (or $SE(d)$) equivariant frame $\mathcal{F}$ with the commuting property, and ϕ, Φ taken as DeepSets/PointNet, then FA-DeepSets/PointNet is $E(d) \times S_n$ invariant/equivariant and can approximate arbitrary continuous function that is $G \times S_n$ invariant/equivariant.*

For $SE(d)$, this result provides a similar expressiveness guarantee as the one from (Dym & Maron, 2020) analyzing Tensor Field Networks (Thomas et al., 2018; Fuchs et al., 2020).

3.2 Graphs, permutations

Symmetry and frame choice. Let $G = S_n$, and $V = \mathbb{R}^{n\times d} \times \mathbb{R}^{n\times n}$, where $\mathbf{X} = (\boldsymbol{Y}, \boldsymbol{A}) \in V$ represents a set of node features $\boldsymbol{Y} \in \mathbb{R}^{n\times d}$, and an adjacency matrix (or some edge attributes) $\boldsymbol{A} \in \mathbb{R}^{n\times n}$; we assume undirected graphs, meaning $\boldsymbol{A} = \boldsymbol{A}^T$. Let $\mathbf{X} \in V$, the representation ρ_1 is defined by $\rho_1(g)\mathbf{X} = (\boldsymbol{P}\boldsymbol{Y}, \boldsymbol{P}\boldsymbol{A}\boldsymbol{P}^T)$, where $\boldsymbol{P} = \boldsymbol{P}_g$ is the permutation matrix representing $g \in S_n$. We define $\mathcal{F}(\mathbf{X})$ to contain all $g \in S_n$ that sort the rows of the matrix $\boldsymbol{S}(\mathbf{X})$ in column lexicographic manner, i.e., $\boldsymbol{P}\boldsymbol{S}(\mathbf{X})$ is lexicographically sorted; the matrix $\boldsymbol{S}$ is defined as follows. Let $\boldsymbol{L} = \mathrm{diag}(\boldsymbol{A}\mathbf{1}) - \boldsymbol{A}$ be the graph's Laplacian. Inspired by (Fürer, 2010), for every eigenspace (traversed in increasing eigenvalue order), spanned by the orthogonal basis $\boldsymbol{u}_1, \ldots, \boldsymbol{u}_k$, we add the (equivariant) column $\mathrm{diag}(\sum_{i=1}^k \boldsymbol{u}_i \boldsymbol{u}_i^T)$ to $\boldsymbol{S}$. Hence, the number of columns of $\boldsymbol{S}$ equals to the number of unique eigenvalues of $\boldsymbol{L}$.

Proposition 2. *The frame $\mathcal{F}(\mathbf{X})$ defined by sorting of $\boldsymbol{S}(\mathbf{X})$ is S_n-equivariant and bounded.*

Backbone architectures. In this case we perform only invariant tasks, for which we chose two universal backbone architectures for ϕ: (i) MLP applied to $\mathrm{vec}(\boldsymbol{Y}, \boldsymbol{A})$ and (ii) GNN+ID (Murphy et al., 2019; Loukas, 2020), denoting a GNN backbone equipped with node identifiers as node features. We perform FA with ϕ according to the frame constructed in Proposition 2. Note that Theorem 3 implies, $|\mathcal{F}(X)| \geq |\mathrm{Aut}(G)|$, since the stabilizer $G_{\mathbf{X}}$ is the automorphism group of the graph, i.e., $g \in G_{\mathbf{X}} = \mathrm{Aut}(\mathbf{X})$ iff $\rho_1(g)\mathbf{X} = \mathbf{X}$. This means that for symmetric graphs equation 3 can prove too costly. In this case, we use the approximate invariant FA, equation 8.

Universality. Let us use Theorem 5 to prove FA provides a universal GNN if we consider either of the above backbones for ϕ, Φ. Let us consider the equivariant case (that contains the invariant case, as discussed in Section 2.2). Let $K \subset V$ be an invariant compact domain, then $\bar{\mathcal{H}} = \mathcal{C}(K, W)$ (see e.g., (Pinkus, 1999; Loukas, 2020; Puny et al., 2020)). Now, since any continuous S_n-equivariant graph function $\Psi : K \to W$ is in $\mathcal{C}(K, W) \cap \mathcal{E}$, Theorem 5 implies Ψ can be approximated in the uniform norm over K with a function in $\langle \mathcal{H} \rangle_{\mathcal{F}}$.

Corollary 3 (FA-MLP and FA-GNN+ID is graph universal)**.** *Given an equivariant frame $\mathcal{F}$, and ϕ, Φ taken as MLPs or GNN+ID, then FA-MLP and FA-GNN+ID are S_n invariant/equivariant and can approximate arbitrary continuous graph functions over compact invariant domain $K \subset V$.*

3.3 GRAPHS, PERMUTATIONS AND EUCLIDEAN MOTIONS.

Symmetry and frame choice. We consider the group $G = E(d)$ acting on graphs, i.e., $V = \mathbb{R}^{n\times d} \times \mathbb{R}^{n\times n}$, where $\mathbf{X} = (\boldsymbol{Y}, \boldsymbol{A}) \in V$ represents a set of node and edge attributes, as described above. The group representation is $\rho_1(g)\mathbf{X} = \rho_1(g)(\boldsymbol{Y}, \boldsymbol{A}) = (\boldsymbol{Y}(\boldsymbol{R}^T + \mathbf{1}\boldsymbol{t}^T), \boldsymbol{A})$. We define the frame $\mathcal{F}(\mathbf{X}) = \mathcal{F}(\boldsymbol{Y})$ using the node features as in the point cloud case, Section 3.1. Therefore Proposition 1 implies $\mathcal{F}$ is equivariant and bounded. Next, also in this case we would like to incorporate $E(d)$ symmetries to an already S_n invariant/equivariant graph neural network architectures; again per Theorem 2, we will also need to show that the ρ_1 (and similarly ρ_2) commutes with $\tau : S_n \to \mathrm{GL}(V)$ defined by $\tau(h)\mathbf{X} = (\boldsymbol{P}\boldsymbol{Y}, \boldsymbol{P}\boldsymbol{A}\boldsymbol{P}^T)$, where $\boldsymbol{P} = \boldsymbol{P}_h$ is the permutation matrix of $h \in S_n$. Indeed $\tau(h)\rho_1(g)\mathbf{X} = (\boldsymbol{P}\boldsymbol{Y}(\boldsymbol{R}^T + \mathbf{1}\boldsymbol{t}^T), \boldsymbol{P}\boldsymbol{A}\boldsymbol{P}^T) = \rho_1(g)\tau(h)\mathbf{X}$. Furthermore, note that as in the point cloud case $\mathcal{F}(\tau(h)\mathbf{X}) = \mathcal{F}(\mathbf{X})$, therefore $\mathcal{F}$ is also S_n invariant.

Backbone architecture. The backbone architecture we chose for this instantiation is the Message Passing GNN in Gilmer et al. (2017), an S_n equivariant model denoted $\Phi_{d,d'} : \mathbb{R}^{n\times d} \times \mathbb{R}^{n\times n} \to \mathbb{R}^{n\times d'} \times \mathbb{R}^{n\times n}$. In this case we constructed a model, as suggested above, by composing two equivariant blocks, $\Psi(\mathbf{X}) = \langle \Phi^{(2)}_{3d',3} \rangle_{\mathcal{F}} \circ \langle \Phi^{(1)}_{6,3d'} \rangle_{\mathcal{F}}(\mathbf{X})$. The input feature size is 6 since we use this network to learn point clouds with velocities (n-body problem). We name this model FA-GNN.

4 PREVIOUS WORKS

Rotation invariant and equivariant point networks. State of the art S_n invariant networks, e.g., (Qi et al., 2017a;b; Atzmon et al., 2018; Li et al., 2018; Xu et al., 2018b; Wang et al., 2019) are not invariant/equivariant to rotations/reflections by construction (Chen et al., 2019). Invariance to global or local rotations can be achieved by modifying the 3D convolution operator or modifying the input representation. Relative angles and distances across points (Deng et al., 2018; Zhang et al., 2019) or angles and distances w.r.t. normals (Gojcic et al., 2019) can be used for rotation invariance. Other works use some local or global frames to achieve invariance to rotations and translations. Xiao et al. (2020); Yu et al. (2020); Deng et al. (2018) also use PCA to define rotation invariance, and can be seen as instances of the FA framework. We augment this line of work by introducing a more general framework that includes equivariance to rotation/reflection and translation, more general architectures and symmetries as well as theoretical analysis of the expressive power of such models.
Equivariance is a desirable property for 3D recognition, registration (Ao et al., 2021), and other domains such as complex physical systems (Kondor, 2018). A popular line of work utilizes the theory of spherical harmonics to achieve equivariance (Worrall et al., 2017; Esteves et al., 2018; Liu et al., 2018; Weiler et al., 2018; Cohen et al., 2018). Notably, Tensor Field Networks (TFN), $SE(3)$ transformers, and Group-Equivariant Attention (Thomas et al., 2018; Fuchs et al., 2020; Romero & Cordonnier, 2021) achieve equivariance to both translation and rotation, i.e., $SE(3)$, and are maximally expressive (i.e., universal) as shown in Dym & Maron (2020). These methods, however, are specifically adapted to $SE(3)$ and require high order $SO(3)$ representations as feature vectors. Recently Deng et al. (2021) propose a rotation equivariant network by introducing tensor features, linear layers that act equivariantly on them and equivariant non-linearities etc. However, their architecture is not proved to be universal. Discrete convolutions (Cohen et al., 2019; Li et al., 2019; Worrall & Brostow, 2018) have also been used for achieving equivariance. In particular, Chen et al. (2021) propose point networks that are equivariant to transforms in $SE(3)$ and use separable discrete convolutions.

Graph neural networks. Message passing (MP) GNNs (Gilmer et al., 2017) are designed to be S_n equivariant. Kondor et al. (2018) introduces a broader set of equivariant operators in MP-GNNs, while Maron et al. (2018) provides a full characterization of linear permutation invariant/equivariant GNN layers. In a parallel approach, trying to avoid harming expressively due to restricted architectures (Xu et al., 2018a; Morris et al., 2019), other works suggested symmetrization of non invariant/equivariant backbones. Ranging from eliminating all symmetries by a canonical ordering (Niepert et al., 2016) to averaging over the entire symmetry group (Murphy et al., 2019; 2018), which amount to the trivial frame $\mathcal{F} \equiv \rho(G)$, with the symmetry group $G = S_n$. As we have also shown, this approach comes at a cost of high variance approximations hindering the learning process. Our FA framework reduces the variance both by choosing a canonical ordering and addressing the fact that it may not be unique.

More recently, a body of work studies GNNs with invariance/equivariance to $E(3)$ (or a similar group) to deal with symmetries in molecular data or dynamical systems. Many $SE(3)$ equivariant constructions (Anderson et al., 2019; Fuchs et al., 2020; Batzner et al., 2021; Klicpera et al., 2021)

extend TFN (Thomas et al., 2018) and inherit its expensive higher order feature representations in the intermediate layers. Finally, a recent work by Satorras et al. (2021) provides an efficient message passing construction that achieves $E(d)$ equivariance but is not shown to be universal, thus far.

5 EXPERIMENTS

We evaluate our FA framework on a few invariant/equivaraint point cloud and graph learning tasks: point cloud normal estimation ($O(3)$ equivariant and translation invariant); graph separation tasks (S_n invariant); and particles position estimation, i.e., the n-body problem ($E(3)$ equivariant).

5.1 POINT CLOUDS: NORMAL ESTIMATION

Normal estimation is a core *geometry processing* task, where the goal is to estimate normal data from an input point cloud, $\boldsymbol{X} \in \mathbb{R}^{n\times 3}$. This task is $O(3)$ equivariant and translation invariant. To test the effect of rotated data on the different models we experimented with three different settings: i) I/I - training and testing on the original data; ii) $I/SO(3)$ - training on the original data and testing on randomly rotated data; and iii) $SO(3)/SO(3)$ - training and testing with randomly rotated data. We used the ABC dataset (Koch et al., 2019) that contains 3 collections (10k, 50k, and 100k models each) of *Computer-Aided Design* (CAD) models. We follow the protocol of the benchmark suggested in Koch et al. (2019), and quantitatively measure normal estimation quality via $1 - (\boldsymbol{n}^T\hat{\boldsymbol{n}})^2$, with $\boldsymbol{n}$ being the ground truth normal and $\hat{\boldsymbol{n}}$ the normal prediction. We used the same random train/test splits from Koch et al. (2019). For baselines, we chose the PointNet (Qi et al., 2017a) and DGCNN (Wang et al., 2019) architectures, which are popular permutation equivariant 3D point cloud architectures. In addition, we also compare to VN-PointNet and VN-DGCNN (Deng et al., 2021), a recent state of the art approach for $SO(3)$ equivariant network design. We also tested our FA models, FA-PointNet and FA-DGCNN as described in Section 3.1. In addition, we tested our local $O(3)$ equivariant model, FA-Local-PointNet. See Appendix C.1 for the further implementation details. The results in Table 1 showcase the advantages of our FA framework: Incorporating FA to existing architectures is beneficial in scenarios (ii-iii), outperforming augmentation by a large margin. In contrast to VN models, FA models maintain (and in some cases even improve) the baseline estimation quality (i), attributed to the expressive power of the FA models.

Method	10k			50k			100k		
	I/I	$I/SO(3)$	$SO(3)/SO(3)$	I/I	$I/SO(3)$	$SO(3)/SO(3)$	I/I	$I/SO(3)$	$SO(3)/SO(3)$
PointNet	0.185	0.442	0.228	0.170	0.405	0.195	0.168	0.413	0.215
VN-PointNet	0.281	0.279	0.271	0.153	0.200	0.207	0.144	0.205	0.159
FA-PointNet	0.158	0.156	0.157	0.151	0.154	0.152	0.149	0.148	0.150
FA-Local-PointNet	0.098	0.098	0.099	0.092	0.090	0.090	0.091	0.091	0.091
DGCNN	0.083	0.183	0.122	**0.054**	0.170	0.111	**0.058**	0.167	0.113
VN-DGCNN	0.105	0.117	0.105	0.107	0.091	0.091	0.089	0.092	0.091
FA-DGCNN	**0.068**	**0.069**	**0.070**	0.057	**0.060**	**0.061**	0.066	**0.061**	**0.061**

Table 1: Normal estimation, ABC dataset (Koch et al., 2019) benchmark.

5.2 GRAPHS: EXPRESSIVE POWER

Producing GNNs that are both expressive and computationally tractable is a long standing goal of the graph learning community. In this experiment we test graph separation (S_n invariant task): the ability of models to separate and classify graphs, a basic trait for graph learning. We use two datasets: GRAPH8c (Balcilar et al., 2021) that consists of all non-isomorphic, connected 8 node graphs; and EXP (Abboud et al., 2021) that consists of 3-WL distinguishable graphs that are not 2-WL distinguishable. There are two tasks: (i) count pairs of graphs not separated by a randomly initialized model in GRAPH8c and EXP; and (ii) learn to classify EXP to two classes. We follow Balcilar et al. (2021) experimental setup. As baselines we use GCN Kipf & Welling (2016), GAT Veličković et al. (2018), GIN Xu et al. (2018a), CHEBNET Tang et al. (2019), PPGN Maron et al. (2019), and GNNML3 Balcilar et al.

Model	GRAPH8c	EXP	EXP-classify
GCN	4755	600	50%
GAT	1828	600	50%
GIN	386	600	50%
CHEBNET	44	71	82%
PPGN	0	0	**100%**
GNNML3	0	0	**100%**
GA-MLP	0	0	50%
FA-MLP	0	0	**100%**
GA-GIN+ID	0	0	50%
FA-GIN+ID	0	0	**100%**

Table 2: Graph separation (Balcilar et al., 2021; Abboud et al., 2021).

(2021), all of which are equivariant by construction. We compare to our FA-MLP, and FA-GIN+ID, as described in Section 3.2 that provide tractable option for universal GNNs. We also compare to the trivial (i.e., entire group) frame averaging $\mathcal{F} \equiv G$, denoted GA-MLP and GA-GIN+ID as advocated in (Murphy et al., 2019). Appendix C.2 provides full implementation details.

The two first columns in Table 2 show the results of the separation task (i). As expected both FA and GA provide perfect separation, however, are not perfectly invariant unless one computes the full averages on $\mathcal{F}$ and G, respectively. Since for some graphs (with many symmetries) full averages are not feasible, we compare the *invariance error* (see appendix for the exact definition) of GA with that of the approximate FA, equation 8, for 50 randomly permuted inputs. Figure 2-left shows the mean, std, and 90% percentile of *invariance error* for increasing sample size k. Note that approximate FA is more invariant even with as little as $k = 1$ samples, which explain why in the classification task (ii), in Table 2, right column, FA is able to learn while GA fails, when both are trained with sample size $k = 1$. Figure 2-right compares $m_{\mathcal{F}}$ and m_G (the maximal number of unique elements in the FA, equation 7) for GRAPH8c dataset. The color and size of the points in the plot represents how many graphs in the dataset have the corresponding $(m_G, m_{\mathcal{F}})$ values. This demonstrates the benefit (in one example) of FA over GA in view of Theorem 4 and approximation in equation 8. Of course, other, more powerful frames can be chosen, e.g., using higher order WL (Morris et al., 2019) or substructure counting (Bouritsas et al., 2020) that will further improve the approximation of equation 8.

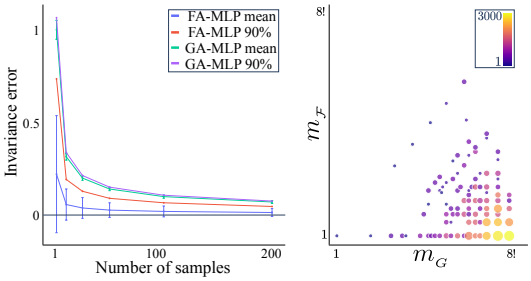

Figure 2: Invariance error (left); $m_{\mathcal{F}}$-m_G (right).

5.3 Graphs: n-Body Problem

In this task we learn to solve the n-body problem ($E(3)$ equivariant). The dataset, created in (Satorras et al., 2021; Fuchs et al., 2020), consists of a collection of $n = 5$ particles systems, where each particle is equipped with its initial position in $\mathbb{R}^3$ and velocity in $\mathbb{R}^3$, and each pair of particles (edge) with a value indicating their charge difference. The task is to predict the particles' locations after a fixed time. We follow the protocol of (Satorras et al., 2021). As baselines we use EGNN (Satorras et al., 2021), GNN (Gilmer et al., 2017), TFN (Thomas et al., 2018), SE(3)-Transformer (Fuchs et al., 2020) and Radial Field (Köhler et al., 2019). We test our FA-GNN, as described in Section 3.3. Table 3 logs the results, where the metric reported is the Mean Squared Error between predicted locations and ground truth. As can be seen, FA-GNN improves over the SOTA by more than 10%. Note that the number of parameters used in FA-GNN and EGNN is roughly the same. More details are provided in Appendix C.3

Method	MSE	Forward time (s)
Linear	0.0819	.0001
SE(3) Transformer	0.0244	.1346
TFN	0.0155	.0343
GNN	0.0107	.0032
Radial Field	0.0104	.0039
EGNN	0.0071	.0062
FA-GNN	**0.0062**	.0041

Table 3: n-body experiment (Satorras et al., 2021).

6 Conclusions

We present Frame Averaging, a generic and principled methodology for adapting existing (backbone) neural architectures to be invariant/equivariant to desired symmetries that appear in the data. We prove the method preserves the expressive power of the backbone model, and is efficient to compute in several cases of interest. We use FA to build universal GNNs, universal point cloud networks that are invariant/equivariant to Euclidean motions $E(3)$, and GNNs invariant/equivariant to Euclidean motions $E(3)$. We empirically validate the effectiveness of these models on several invariant/equivariant learning tasks. We believe the instantiations presented in the paper are only the first step in exploring the full potential of the FA framework. Interesting open questions, not answered by this paper are: What other symmetries and scenarios can benefit from FA? What would be a way to systematically find efficient/small frames? How the frame choice effects the learning process? How to explore useful FA architectures and modules?

REFERENCES

Ralph Abboud, İsmail İlkan Ceylan, Martin Grohe, and Thomas Lukasiewicz. The surprising power of graph neural networks with random node initialization. In *Proceedings of the Thirtieth International Joint Conference on Artifical Intelligence (IJCAI)*, 2021.

Brandon Anderson, Truong-Son Hy, and Risi Kondor. Cormorant: Covariant molecular neural networks. *arXiv preprint arXiv:1906.04015*, 2019.

Sheng Ao, Qingyong Hu, Bo Yang, Andrew Markham, and Yulan Guo. Spinnet: Learning a general surface descriptor for 3d point cloud registration. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 11753–11762, 2021.

Matan Atzmon, Haggai Maron, and Yaron Lipman. Point convolutional neural networks by extension operators. *arXiv preprint arXiv:1803.10091*, 2018.

Muhammet Balcilar, Pierre Héroux, Benoit Gaüzère, Pascal Vasseur, Sébastien Adam, and Paul Honeine. Breaking the limits of message passing graph neural networks. In *Proceedings of the 38th International Conference on Machine Learning (ICML)*, 2021.

Peter W Battaglia, Jessica B Hamrick, Victor Bapst, Alvaro Sanchez-Gonzalez, Vinicius Zambaldi, Mateusz Malinowski, Andrea Tacchetti, David Raposo, Adam Santoro, Ryan Faulkner, et al. Relational inductive biases, deep learning, and graph networks. *arXiv preprint arXiv:1806.01261*, 2018.

Simon Batzner, Tess E Smidt, Lixin Sun, Jonathan P Mailoa, Mordechai Kornbluth, Nicola Molinari, and Boris Kozinsky. Se (3)-equivariant graph neural networks for data-efficient and accurate interatomic potentials. *arXiv preprint arXiv:2101.03164*, 2021.

Giorgos Bouritsas, Fabrizio Frasca, Stefanos Zafeiriou, and Michael M Bronstein. Improving graph neural network expressivity via subgraph isomorphism counting. *arXiv preprint arXiv:2006.09252*, 2020.

Chao Chen, Guanbin Li, Ruijia Xu, Tianshui Chen, Meng Wang, and Liang Lin. Clusternet: Deep hierarchical cluster network with rigorously rotation-invariant representation for point cloud analysis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 4994–5002, 2019.

Haiwei Chen, Shichen Liu, Weikai Chen, Hao Li, and Randall Hill. Equivariant point network for 3d point cloud analysis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 14514–14523, 2021.

Taco Cohen, Maurice Weiler, Berkay Kicanaoglu, and Max Welling. Gauge equivariant convolutional networks and the icosahedral cnn. In *International Conference on Machine Learning*, pp. 1321–1330. PMLR, 2019.

Taco S Cohen, Mario Geiger, Jonas Köhler, and Max Welling. Spherical cnns. In *International Conference on Learning Representations*, 2018.

Amir Dembo and Ofer Zeitouni. *Large Deviations Techniques and Applications*, volume 38 of *Stochastic Modelling and Applied Probability*. Springer Berlin / Heidelberg, Berlin/Heidelberg, 2010. ISBN 9783642033100.

Congyue Deng, Or Litany, Yueqi Duan, Adrien Poulenard, Andrea Tagliasacchi, and Leonidas Guibas. Vector neurons: A general framework for so (3)-equivariant networks. *arXiv preprint arXiv:2104.12229*, 2021.

Haowen Deng, Tolga Birdal, and Slobodan Ilic. Ppf-foldnet: Unsupervised learning of rotation invariant 3d local descriptors. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 602–618, 2018.

Nadav Dym and Haggai Maron. On the universality of rotation equivariant point cloud networks. *arXiv preprint arXiv:2010.02449*, 2020.

Carlos Esteves, Christine Allen-Blanchette, Ameesh Makadia, and Kostas Daniilidis. Learning so (3) equivariant representations with spherical cnns. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 52–68, 2018.

Fabian B Fuchs, Daniel E Worrall, Volker Fischer, and Max Welling. Se (3)-transformers: 3d roto-translation equivariant attention networks. *arXiv preprint arXiv:2006.10503*, 2020.

William Fulton and Joe Harris. *Representation theory: a first course*, volume 129. Springer Science & Business Media, 2013.

Martin Fürer. On the power of combinatorial and spectral invariants. *Linear algebra and its applications*, 432(9):2373–2380, 2010.

Justin Gilmer, Samuel S Schoenholz, Patrick F Riley, Oriol Vinyals, and George E Dahl. Neural message passing for quantum chemistry. In *International conference on machine learning*, pp. 1263–1272. PMLR, 2017.

Zan Gojcic, Caifa Zhou, Jan D Wegner, and Andreas Wieser. The perfect match: 3d point cloud matching with smoothed densities. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 5545–5554, 2019.

Weihua Hu, Muhammed Shuaibi, Abhishek Das, Siddharth Goyal, Anuroop Sriram, Jure Leskovec, Devi Parikh, and C Lawrence Zitnick. Forcenet: A graph neural network for large-scale quantum calculations. *arXiv preprint arXiv:2103.01436*, 2021.

Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

Thomas N Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. *arXiv preprint arXiv:1609.02907*, 2016.

Johannes Klicpera, Florian Becker, and Stephan Günnemann. Gemnet: Universal directional graph neural networks for molecules. *arXiv preprint arXiv:2106.08903*, 2021.

Sebastian Koch, Albert Matveev, Zhongshi Jiang, Francis Williams, Alexey Artemov, Evgeny Burnaev, Marc Alexa, Denis Zorin, and Daniele Panozzo. Abc: A big cad model dataset for geometric deep learning. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019.

Risi Kondor. N-body networks: a covariant hierarchical neural network architecture for learning atomic potentials. *arXiv preprint arXiv:1803.01588*, 2018.

Risi Kondor, Hy Truong Son, Horace Pan, Brandon Anderson, and Shubhendu Trivedi. Covariant compositional networks for learning graphs. *arXiv preprint arXiv:1801.02144*, 2018.

Jonas Köhler, Leon Klein, and Frank Noé. Equivariant flows: sampling configurations for multi-body systems with symmetric energies, 2019.

Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

David A Levin and Yuval Peres. *Markov chains and mixing times*, volume 107. American Mathematical Soc., 2017.

Jiaxin Li, Yingcai Bi, and Gim Hee Lee. Discrete rotation equivariance for point cloud recognition. In *2019 International Conference on Robotics and Automation (ICRA)*, pp. 7269–7275. IEEE, 2019.

Yangyan Li, Rui Bu, Mingchao Sun, Wei Wu, Xinhan Di, and Baoquan Chen. Pointcnn: Convolution on x-transformed points. *Advances in neural information processing systems*, 31:820–830, 2018.

Min Liu, Fupin Yao, Chiho Choi, Ayan Sinha, and Karthik Ramani. Deep learning 3d shapes using alt-az anisotropic 2-sphere convolution. In *International Conference on Learning Representations*, 2018.

Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 3431–3440, 2015.

Andreas Loukas. What graph neural networks cannot learn: depth vs width. In *International Conference on Learning Representations*, 2020. URL https://openreview.net/forum?id=B1l2bp4YwS.

Haggai Maron, Heli Ben-Hamu, Nadav Shamir, and Yaron Lipman. Invariant and equivariant graph networks. *arXiv preprint arXiv:1812.09902*, 2018.

Haggai Maron, Heli Ben-Hamu, Hadar Serviansky, and Yaron Lipman. Provably powerful graph networks. *arXiv preprint arXiv:1905.11136*, 2019.

Christopher Morris, Martin Ritzert, Matthias Fey, William L Hamilton, Jan Eric Lenssen, Gaurav Rattan, and Martin Grohe. Weisfeiler and leman go neural: Higher-order graph neural networks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pp. 4602–4609, 2019.

Ryan Murphy, Balasubramaniam Srinivasan, Vinayak Rao, and Bruno Ribeiro. Relational pooling for graph representations. In *International Conference on Machine Learning*, pp. 4663–4673. PMLR, 2019.

Ryan L Murphy, Balasubramaniam Srinivasan, Vinayak Rao, and Bruno Ribeiro. Janossy pooling: Learning deep permutation-invariant functions for variable-size inputs. *arXiv preprint arXiv:1811.01900*, 2018.

Mathias Niepert, Mohamed Ahmed, and Konstantin Kutzkov. Learning convolutional neural networks for graphs. In Maria Florina Balcan and Kilian Q. Weinberger (eds.), *Proceedings of The 33rd International Conference on Machine Learning*, volume 48 of *Proceedings of Machine Learning Research*, pp. 2014–2023, New York, New York, USA, 20–22 Jun 2016. PMLR. URL https://proceedings.mlr.press/v48/niepert16.html.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems*, 32: 8026–8037, 2019.

Allan Pinkus. Approximation theory of the mlp model in neural networks. *Acta numerica*, 8:143–195, 1999.

Yury Polyanskiy and Yihong Wu. Lecture notes on information theory. *Lecture Notes for ECE563 (UIUC) and*, 6(2012-2016):7, 2014.

Omri Puny, Heli Ben-Hamu, and Yaron Lipman. Global attention improves graph networks generalization, 2020.

Charles R Qi, Hao Su, Kaichun Mo, and Leonidas J Guibas. Pointnet: Deep learning on point sets for 3d classification and segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 652–660, 2017a.

Charles R Qi, Li Yi, Hao Su, and Leonidas J Guibas. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. *arXiv preprint arXiv:1706.02413*, 2017b.

David W Romero and Jean-Baptiste Cordonnier. Group equivariant stand-alone self-attention for vision. In *ICLR*, 2021.

Victor Garcia Satorras, Emiel Hoogeboom, and Max Welling. E (n) equivariant graph neural networks. *arXiv preprint arXiv:2102.09844*, 2021.

Nimrod Segol and Yaron Lipman. On universal equivariant set networks. *arXiv preprint arXiv:1910.02421*, 2019.

Muhammed Shuaibi, Adeesh Kolluru, Abhishek Das, Aditya Grover, Anuroop Sriram, Zachary Ulissi, and C Lawrence Zitnick. Rotation invariant graph neural networks using spin convolutions. *arXiv preprint arXiv:2106.09575*, 2021.

Shanshan Tang, Bo Li, and Haijun Yu. Chebnet: Efficient and stable constructions of deep neural networks with rectified power units using chebyshev approximations, 2019.

Nathaniel Thomas, Tess Smidt, Steven Kearnes, Lusann Yang, Li Li, Kai Kohlhoff, and Patrick Riley. Tensor field networks: Rotation-and translation-equivariant neural networks for 3d point clouds. *arXiv preprint arXiv:1802.08219*, 2018.

Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Liò, and Yoshua Bengio. Graph attention networks, 2018.

Yue Wang, Yongbin Sun, Ziwei Liu, Sanjay E Sarma, Michael M Bronstein, and Justin M Solomon. Dynamic graph cnn for learning on point clouds.(2018). *arXiv preprint arXiv:1801.07829*, 2018.

Yue Wang, Yongbin Sun, Ziwei Liu, Sanjay E Sarma, Michael M Bronstein, and Justin M Solomon. Dynamic graph cnn for learning on point clouds. *Acm Transactions On Graphics (tog)*, 38(5):1–12, 2019.

Maurice Weiler, Mario Geiger, Max Welling, Wouter Boomsma, and Taco Cohen. 3d steerable cnns: Learning rotationally equivariant features in volumetric data. *arXiv preprint arXiv:1807.02547*, 2018.

Daniel Worrall and Gabriel Brostow. Cubenet: Equivariance to 3d rotation and translation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 567–584, 2018.

Daniel E Worrall, Stephan J Garbin, Daniyar Turmukhambetov, and Gabriel J Brostow. Harmonic networks: Deep translation and rotation equivariance. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 5028–5037, 2017.

Zelin Xiao, Hongxin Lin, Renjie Li, Lishuai Geng, Hongyang Chao, and Shengyong Ding. Endowing deep 3d models with rotation invariance based on principal component analysis. In *2020 IEEE International Conference on Multimedia and Expo (ICME)*, pp. 1–6. IEEE, 2020.

Keyulu Xu, Weihua Hu, Jure Leskovec, and Stefanie Jegelka. How powerful are graph neural networks? *arXiv preprint arXiv:1810.00826*, 2018a.

Yifan Xu, Tianqi Fan, Mingye Xu, Long Zeng, and Yu Qiao. Spidercnn: Deep learning on point sets with parameterized convolutional filters. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 87–102, 2018b.

Dmitry Yarotsky. Universal approximations of invariant maps by neural networks. *Constructive Approximation*, pp. 1–68, 2021.

Ruixuan Yu, Xin Wei, Federico Tombari, and Jian Sun. Deep positional and relational feature learning for rotation-invariant point cloud analysis. In *European Conference on Computer Vision*, pp. 217–233. Springer, 2020.

Manzil Zaheer, Satwik Kottur, Siamak Ravanbakhsh, Barnabas Poczos, Ruslan Salakhutdinov, and Alexander Smola. Deep sets. *arXiv preprint arXiv:1703.06114*, 2017.

Zhiyuan Zhang, Binh-Son Hua, David W Rosen, and Sai-Kit Yeung. Rotation invariant convolutions for 3d point clouds deep learning. In *2019 International Conference on 3D Vision (3DV)*, pp. 204–213. IEEE, 2019.

A PROOFS

A.1 PROOF OF THEOREM 1

Proof. First note that frame equivariance is defined to be $\mathcal{F}(\rho_1(g)X) = g\mathcal{F}(X)$ which in particular means $|\mathcal{F}(\rho_1(g)X)| = |\mathcal{F}(X)|$.

For invariance, let $g' \in G$

$$\begin{aligned}\langle\phi\rangle_{\mathcal{F}}(\rho_1(g')X) &= \frac{1}{|\mathcal{F}(X)|}\sum_{g\in g'\mathcal{F}(X)}\phi(\rho_1(g)^{-1}\rho_1(g')X) = \frac{1}{|\mathcal{F}(X)|}\sum_{g\in\mathcal{F}(X)}\phi(\rho_1(g'g)^{-1}\rho_1(g')X)\\ &= \frac{1}{|\mathcal{F}(X)|}\sum_{g\in\mathcal{F}(X)}\phi(\rho_1(g)^{-1}X) = \langle\phi\rangle_{\mathcal{F}}(X)\end{aligned}$$

and for equivariance

$$\begin{aligned}\langle\Phi\rangle_{\mathcal{F}}(\rho_1(g')X) &= \frac{1}{|\mathcal{F}(X)|}\sum_{g\in g'\mathcal{F}(X)}\rho_2(g)\Phi(\rho_1(g)^{-1}\rho_1(g')X)\\ &= \frac{1}{|\mathcal{F}(X)|}\sum_{g\in\mathcal{F}(X)}\rho_2(g'g)\Phi(\rho_1(g'g)^{-1}\rho_1(g')X)\\ &= \rho_2(g')\frac{1}{|\mathcal{F}(X)|}\sum_{g\in\mathcal{F}(X)}\rho_2(g)\Phi(\rho_1(g)^{-1}X)\\ &= \rho_2(g')\langle\Phi\rangle_{\mathcal{F}}(X)\end{aligned}$$

□

A.2 PROOF OF THEOREM 2

Proof. The proof above A.1, of Theorem 1, is a special case of the following where the second symmetry is chosen to be trivial. In principle the proofs are quite similar.

First note that frame equivariance and invariance, together with ρ_1, τ_1 commuting mean

$$\mathcal{F}(\gamma_1(h,g)X) = \mathcal{F}(\tau_1(h)\rho_1(g)X) = \mathcal{F}(\rho_1(g)\tau_1(h)X) = g\mathcal{F}(\tau_1(h)X) = g\mathcal{F}(X),$$

which in particular implies that $|\mathcal{F}(\gamma_1(h,g)X)| = |\mathcal{F}(X)|$. Let $(h',g') \in H \times G$ be arbitrary. Then,

$$\begin{aligned}\langle\phi\rangle_{\mathcal{F}}(\gamma_1(h',g')X) &= \frac{1}{|\mathcal{F}(X)|}\sum_{g\in g'\mathcal{F}(X)}\phi(\rho_1(g)^{-1}\tau_1(h')\rho_1(g')X)\\ &= \frac{1}{|\mathcal{F}(X)|}\sum_{g\in\mathcal{F}(X)}\phi(\rho_1(g'g)^{-1}\tau_1(h')\rho_1(g')X)\\ &= \frac{1}{|\mathcal{F}(X)|}\sum_{g\in\mathcal{F}(X)}\phi(\rho_1(g)^{-1}\tau_1(h')X)\\ &= \frac{1}{|\mathcal{F}(X)|}\sum_{g\in\mathcal{F}(X)}\phi(\rho_1(g)^{-1}X)\\ &= \langle\phi\rangle_{\mathcal{F}}(X)\end{aligned}$$

meaning that $\langle\phi\rangle_{\mathcal{F}}$ is $H \times G$ invariant. Next,

$$\begin{aligned}
\langle\Phi\rangle_{\mathcal{F}}(\gamma_1(h',g')X) &= \frac{1}{|\mathcal{F}(X)|}\sum_{g\in g'\mathcal{F}(X)}\rho_2(g)\Phi(\rho_1(g)^{-1}\tau_1(h')\rho_1(g')X) \\
&= \frac{1}{|\mathcal{F}(X)|}\sum_{g\in \mathcal{F}(X)}\rho_2(g'g)\Phi(\rho_1(g'g)^{-1}\tau_1(h')\rho_1(g')X) \\
&= \frac{1}{|\mathcal{F}(X)|}\sum_{g\in \mathcal{F}(X)}\rho_2(g'g)\Phi(\rho_1(g)^{-1}\tau_1(h')X) \\
&= \tau_2(h')\rho_2(g')\frac{1}{|\mathcal{F}(X)|}\sum_{g\in \mathcal{F}(X)}\rho_2(g)\Phi(\rho_1(g)^{-1}X) \\
&= \gamma_2(h',g')\langle\Phi\rangle_{\mathcal{F}}(X)
\end{aligned}$$

showing that $\langle\Phi\rangle_{\mathcal{F}}$ is $H \times G$ equivariant. □

A.3 Proof of Theorem 3 and Corollary 1

Proof. (Theorem 3) First we show G_X acts on $\mathcal{F}(X)$. For an arbitrary $h \in G_X$ and equivariant frame $\mathcal{F}$, we have $\mathcal{F}(X) = \mathcal{F}(\rho_1(h)X) = h\mathcal{F}(X)$, where in the first equality we used the fact that $h \in G_X$ and in the second the equivariance of $\mathcal{F}$. This means that if $g \in \mathcal{F}(X)$ and $h \in G_X$ then also $hg \in \mathcal{F}(X)$, or in other words $\mathcal{F}(X)$ is closed to actions from G_X. Furthermore, since a group acts on itself via bijections we have that the cardinality of all orbits is the same, $|[g]| = |G_X g| = |G_X|$, for all $[g] \in \mathcal{F}(X)/G_X$. Lastly, the equivalence relation $g \sim h \iff hg^{-1} \in G_X$ shows that $\mathcal{F}(X)$ is a union of disjoint orbits of equal cardinality. □

Proof. (Corollary 1) Theorem 3 asserts that all orbits $[g] \in \mathcal{F}(X)/G_X$ have the same number of elements. Therefore, a random choice $g \in \mathcal{F}(X)$ will have equal probability to land in each orbit. □

A.4 Proof of Theorem 4

Proof. For brevity we denote $m = m_{\mathcal{F}}$. Our setting can be formulated as follows. We have the uniform probability distribution, denoted μ, over the discrete space $[m]$ (representing the qoutient set $\mathcal{F}(X)/G_X$); $\mu_j = \frac{1}{m}$, $j \in [m]$, and we have numbers $a_j = \phi(\rho_1(h_j)^{-1}X)$, where $[h_j] \in \mathcal{F}(X)/G_X$, $j \in [m]$ represent exactly one sample per orbit. In this notation $\langle\phi\rangle_{\mathcal{F}}(X) = \sum_{j=1}^m \frac{a_j}{m}$, while the approximation in equation 8 takes the form $\langle\!\langle\phi\rangle\!\rangle_{\mathcal{F}}(X) = \sum_{j=1}^m \hat{\mu}_j a_j$, where $\hat{\mu}_j = k_j/k$, where k_j is the number of samples $g_i \in [h_j]$, $i \in [k]$, i.e., samples that landed in the j-th orbit. Therefore,

$$\sup_{\phi\in\mathcal{Q}}|\langle\phi\rangle_{\mathcal{F}}(X) - \langle\!\langle\phi\rangle\!\rangle_{\mathcal{F}}(X)| = \sup_{|a_j|\le 1}\left|\sum_{j=1}^m a_j\left(\frac{1}{m}-\hat{\mu}_j\right)\right| = \sum_{j=1}^m\left|\frac{1}{m}-\hat{\mu}_j\right| = 2\left\|\mu-\hat{\mu}\right\|_{\mathrm{TV}},$$

where the latter is the total variation norm for discrete measures (Levin & Peres, 2017). We denote $H(\hat{\mu}|\mu) = \sum_{j=1}^m \log(\hat{\mu}_j)\log\left(\frac{\hat{\mu}_j}{\mu_j}\right)$ the KL-divergence. The Pinsker inequality and its inverse for discrete positive measures are (see e.g., (Polyanskiy & Wu, 2014)):

$$\frac{1}{2}\left\|\hat{\mu}-\mu\right\|_{\mathrm{TV}}^2 \le H(\hat{\mu}|\mu) \le \frac{2}{\alpha}\left\|\hat{\mu}-\mu\right\|_{\mathrm{TV}}^2, \tag{9}$$

where $\alpha = \min_{j\in[m]}\mu_j = m^{-1}$. Therefore,

$$\Gamma_\epsilon = \left\{\hat{\mu}\;\middle|\;\sup_{\phi\in\mathcal{Q}}|\langle\phi\rangle_{\mathcal{F}}(X) - \langle\!\langle\phi\rangle\!\rangle_{\mathcal{F}}(X)| \le \epsilon\right\} = \left\{\hat{\mu}\;\middle|\;2\left\|\hat{\mu}-\mu\right\|_{\mathrm{TV}} \le \epsilon\right\} \subset \left\{\hat{\mu}\;\middle|\;H(\bar{\mu}|\mu) \le \frac{\epsilon^2 m}{2}\right\}$$

Now, application of Large Deviation Theory (Lemma 2.1.9 in Dembo & Zeitouni (2010)) provides that for $\hat{\mu}$ so that $H(\hat{\mu}|\mu) \le \frac{\epsilon^2 m}{2}$:

$$P(\hat{\mu}) \ge \frac{1}{(1+k)^m}e^{-\frac{km\epsilon^2}{2}}.$$

□

A.5 Proof of Theorem 5

In the following we will use $\langle\cdot\rangle = \langle\cdot\rangle_{\mathcal{F}}$ for brevity. In fact, we will prove a stronger claim, that is, $\overline{\langle\mathcal{H}\rangle} = \overline{\langle\bar{\mathcal{H}}\rangle}$, where the r.h.s. can be understood as all the functions that can be approximated by frame averaging of functions that can be approximated by $\mathcal{H}$.

Theorem 6 (Expressive power of FA)**.** *If $\mathcal{F}$ is a bounded G-equivariant frame over an invariant domain K then $\langle\mathcal{H}\rangle$ is at-least as expressive as $\mathcal{H}$. That is, $\overline{\langle\mathcal{H}\rangle} = \overline{\langle\bar{\mathcal{H}}\rangle} \supset \bar{\mathcal{H}} \cap \mathcal{E}$.*

Proof. To see why $\overline{\langle\bar{\mathcal{H}}\rangle} \supset \bar{\mathcal{H}} \cap \mathcal{E}$ note that any G-equivariant function $\Psi \in \bar{\mathcal{H}}$ satisfies $\Psi = \langle\Psi\rangle \in \langle\bar{\mathcal{H}}\rangle$.

We will now prove that $\overline{\langle\mathcal{H}\rangle} = \overline{\langle\bar{\mathcal{H}}\rangle}$. First, clearly $\langle\mathcal{H}\rangle \subset \overline{\langle\bar{\mathcal{H}}\rangle}$, and since the latter is closed under the maximum norm we have $\overline{\langle\mathcal{H}\rangle} \subset \overline{\langle\bar{\mathcal{H}}\rangle}$.

The other direction is similar to Yarotsky (2021), but using frame averages rather than full group averages. Let $\epsilon > 0$ be arbitrary, and consider an arbitrary element $\varphi \in \overline{\langle\bar{\mathcal{H}}\rangle}$. We will show $\varphi \in \overline{\langle\mathcal{H}\rangle}$. There exists $\varphi_1 \in \bar{\mathcal{H}}$ so that $\|\varphi - \langle\varphi_1\rangle\|_{\infty,K} < \epsilon$. Next, let $\varphi_2 \in \mathcal{H}$ such that $\|\varphi_1 - \varphi_2\|_{\infty,K} < \epsilon$. Finally, for arbitrary $X \in K$

$$\begin{aligned}
\|\varphi(X) - \langle\varphi_2\rangle(X)\| &\leq \|\varphi(X) - \langle\varphi_1\rangle(X)\| + \|\langle\varphi_1\rangle(X) - \langle\varphi_2\rangle(X)\| \\
&\leq \epsilon + \frac{1}{|\mathcal{F}(X)|}\sum_{g\in\mathcal{F}(X)} \left\|\rho_2(g)\varphi_1(\rho_1(g)^{-1}X) - \rho_2(g)\varphi_2(\rho_1(g)^{-1}X)\right\| \\
&\leq \epsilon + \epsilon \max_{g\in\mathcal{F}(X)} \|\rho_2(g)\| = \epsilon(1 + \max_{g\in\mathcal{F}(X)} \|\rho_2(g)\|)
\end{aligned}$$

This means that for bounded $\rho_2(\mathcal{F}(X))$ we have that $\|\varphi - \langle\varphi_2\rangle\|_{\infty,K}$ can be made arbitrary small with $\varphi_2 \in \mathcal{H}$. Therefore, $\varphi \in \overline{\langle\mathcal{H}\rangle}$, as required. We proved the inclusion $\overline{\langle\mathcal{H}\rangle} \supset \overline{\langle\bar{\mathcal{H}}\rangle}$. □

A.6 Proof of Proposition 1

Proof. Let us prove $\mathcal{F}$ is equivariant (equation 4). Consider a transformation $g = (\boldsymbol{R}, \boldsymbol{t}) \in G$, and let $(\boldsymbol{O}, \boldsymbol{s}) \in \mathcal{F}(\boldsymbol{X})$, then

$$\boldsymbol{C}\boldsymbol{O} = \boldsymbol{O}\Lambda, \quad \boldsymbol{s} = \frac{1}{n}\boldsymbol{X}^T\mathbf{1}.$$

Note that the group product of these transformations is

$$(\boldsymbol{R}, \boldsymbol{t})(\boldsymbol{O}, \boldsymbol{s}) = (\boldsymbol{R}\boldsymbol{O}, \boldsymbol{R}\boldsymbol{s} + \boldsymbol{t}).$$

We need to show $(\boldsymbol{R}\boldsymbol{O}, \boldsymbol{R}\boldsymbol{s} + \boldsymbol{t}) \in \mathcal{F}((\boldsymbol{R}, \boldsymbol{t})\boldsymbol{X})$. Indeed, $\boldsymbol{R}\boldsymbol{O} \in O(3)$ and consists of eigenvectors of $\boldsymbol{C} = \boldsymbol{R}\boldsymbol{X}^T(\boldsymbol{I} - \frac{1}{n}\mathbf{1}\mathbf{1}^T)\boldsymbol{X}\boldsymbol{R}^T$ as can be verified with a direct computation. If $\boldsymbol{O}, \boldsymbol{R} \in SO(d)$ then also $\boldsymbol{R}\boldsymbol{O} \in SO(d)$. Furthermore

$$\frac{1}{n}\left(\boldsymbol{X}\boldsymbol{R}^T + \mathbf{1}\boldsymbol{t}^T\right)^T\mathbf{1} = \frac{1}{n}\left(\boldsymbol{R}\boldsymbol{X}^T + \boldsymbol{t}\mathbf{1}^T\right)\mathbf{1} = \boldsymbol{R}\boldsymbol{s} + \boldsymbol{t}$$

as required. We have shown $(\boldsymbol{R}, \boldsymbol{t})\mathcal{F}(\boldsymbol{X}) \subset \mathcal{F}((\boldsymbol{R}, \boldsymbol{t})\boldsymbol{X})$ for all $\boldsymbol{X}$ and $(\boldsymbol{R}, \boldsymbol{t})$. To show the other inclusion let $\boldsymbol{X} = (\boldsymbol{R}, \boldsymbol{t})^{-1}\boldsymbol{Y}$ and get $\mathcal{F}((\boldsymbol{R}, \boldsymbol{t})^{-1}\boldsymbol{Y}) \subset (\boldsymbol{R}, \boldsymbol{t})^{-1}\mathcal{F}(\boldsymbol{Y})$ that also holds for all $\boldsymbol{Y}$ and $(\boldsymbol{R}, \boldsymbol{t})$. In particular $(\boldsymbol{R}, \boldsymbol{t})\mathcal{F}(\boldsymbol{X}) \supset \mathcal{F}((\boldsymbol{R}, \boldsymbol{t})\boldsymbol{X})$. The frame $\mathcal{F}$ is bounded since for compact $K \subset \mathbb{R}^{n\times d}$, the translations $n^{-1}\boldsymbol{X}^T\mathbf{1}$ are compact and therefore uniformly bounded for $X \in K$, and orthogonal matrices always satisfy $\|\boldsymbol{R}\|_2 = 1$. □

A.7 Proof of Proposition 2

Proof. First note that by definition $\boldsymbol{S}(\mathbf{X})$ is equivariant in rows, namely $\boldsymbol{S}(\rho_1(g)\mathbf{X}) = \rho_1(g)\boldsymbol{S}(\mathbf{X})$. Therefore if $g \in \mathcal{F}(\mathbf{X}) \subset S_n$, then by definition of the frame $\rho_1(g)\boldsymbol{S}$ is sorted. Therefore, $\rho_1(g)\boldsymbol{S} = \rho_1(gh)\rho_1(h)^{-1}\boldsymbol{S}(\mathbf{X}) = \rho_1(gh)\boldsymbol{S}(\rho_1(h)^{-1}\mathbf{X})$ is sorted and we get that $gh \in \mathcal{F}(\rho_1(h)^{-1}\mathbf{X})$. We proved $\mathcal{F}(\mathbf{X})h \subset \mathcal{F}(\rho_1(h)^{-1}\mathbf{X})$ for all $h \in G$ and all $\mathbf{X} \in V$. Taking $\mathbf{X} = \rho_1(h)\mathbf{Y}$ for an arbitrary $\mathbf{Y} \in V$, and $h = g^{-1}$ for arbitrary $g \in G$, we get $\mathcal{F}(\rho_1(g^{-1})\mathbf{Y}) \subset \mathcal{F}(\mathbf{Y})g$, for all $g \in G$ and $\mathbf{Y} \in V$. We proved $\mathcal{F}(\mathbf{X})h = \mathcal{F}(\rho_1(h)^{-1}\mathbf{X})$, which amounts to right action equivariance, see equation 5. The frame is bounded since $\rho_1(G)$ is a finite set. □

B INFINITE FRAMES

In the paper we mostly discuss finite frames. However, we note that (mostly for theoretical completeness, but also for potential future applications) the definition of a frame can work in case $\mathcal{F}(X)$ is infinite and there is a (probability) measure μ over frames (left-)invariant to actions of the group G. For example, one can define, analogously to the finite case, $\langle\phi\rangle_{\mathcal{F}}(X) = \int_{\mathcal{F}(X)} \phi(\rho_1(g)^{-1}X)d\mu(X)$ and the proof of Theorem 1 will work similarly. However, as noted, in this paper we will work with finite frames.

C IMPLEMENTATION DETAILS

C.1 POINT CLOUDS: NORMAL ESTIMATION

Here we provide implementation details for the experiment in section 5.1.

PointNet architecture. Our backbone PointNet is based on the object part segmentation network from Qi et al. (2017a). The network consists of layers of the form

$$\mathrm{FC}(n, d_{\text{in}}, d_{\text{out}}) : \boldsymbol{X} \mapsto \nu\left(\boldsymbol{X}\boldsymbol{W} + \mathbf{1}\boldsymbol{b}^T\right)$$
$$\mathrm{MaxPool}(n, d_{\text{in}}) : \boldsymbol{X} \mapsto \mathbf{1}[\max \boldsymbol{X}\boldsymbol{e}_i]$$

where $\boldsymbol{X} \in \mathbb{R}^{n\times d_{\text{in}}}$, $\boldsymbol{W} \in \mathbb{R}^{d_{\text{in}}\times d_{\text{out}}}$, $\boldsymbol{b} \in \mathbb{R}^{d_{\text{out}}}$ are the learnable parameters, $\mathbf{1} \in \mathbb{R}^n$ is the vector of all ones, $[\cdot]$ is the concatenation operator, $\boldsymbol{e}_i$ is the standard basis in $\mathbb{R}^{d_{\text{in}}}$, and ν is the ReLU activation. In this experiement, for the PointNet baseline and for $\Phi_{3,3}$ (the backbone in FA-PointNet), we used the following architecture:

$$\mathrm{FC}(512,3,64) \xrightarrow{L_1} \mathrm{FC}(512,64,128) \xrightarrow{L_2} \mathrm{FC}(512,128,128) \xrightarrow{L_3} \mathrm{FC}(512,128,512) \xrightarrow{L_4}$$
$$\mathrm{FC}(512,512,2048) \xrightarrow{L_5} \mathrm{MaxPool}(512,2048) \xrightarrow{L_6} [L_1,L_2,L_3,L_4,L_5,L_6] \xrightarrow{L_7}$$
$$\mathrm{FC}(512,4028,256) \xrightarrow{L_8} \mathrm{FC}(512,256,128) \xrightarrow{L_9} \mathrm{FC}(512,128,3).$$

Note that the original PointNet network also contains two T-Net networks, applied to the input and to L_3 (the output of the third layer). Similarly, our baseline implementation made a use of the same T-Net networks. Note that the T-Net networks were *not* part of our FA-PointNet backbone architectures $\Phi_{3,3}$.

For FA-Local-PointNet we made the following design choices. Each point patch is constructed as its k nearest neighbors, with $k = 20$, $\boldsymbol{X}_i \in \mathbb{R}^{20\times 3}$. Then, the backbone $\Upsilon_{3*20,3*42}$ is applied to all patches, with the following PointNet layers:

$$\mathrm{FC}(512, 3*20, 3*21) \to \mathrm{FC}(512, 3*21, 3*42) \to \mathrm{FC}(512, 3*42, 3*42).$$

The second backbone, $\Phi_{3*42,3}$, is built from the following PointNet layers:

$$\mathrm{FC}(512,3*32,128) \xrightarrow{L_1} \mathrm{FC}(512,128,256) \xrightarrow{L_2} \mathrm{FC}(512,256,256) \xrightarrow{L_3} \mathrm{FC}(512,256,512) \xrightarrow{L_4}$$
$$\mathrm{FC}(512,512,2048) \xrightarrow{L_5} \mathrm{MaxPool}(512,2048) \xrightarrow{L_6} [L_1,L_2,L_3,L_4,L_5,L_6] \xrightarrow{L_7}$$
$$\mathrm{FC}(512,5248,256) \xrightarrow{L_8} \mathrm{FC}(512,256,128) \xrightarrow{L_9} \mathrm{FC}(512,128,3).$$

DGCNN architecture. Our backbone DGCNN architecture, $\Phi_{3,3}$, is based on the object part segmentation network from Wang et al. (2018). It consists of $\mathrm{EdgeConv}(n, d_{\text{in}}, d_{\text{out}})$, $\mathrm{FC}(n, d_{\text{in}}, d_{\text{out}})$, and $\mathrm{MaxPool}$ layers.

$$\mathrm{EdgeConv}(512,3,64) \xrightarrow{L_1} \mathrm{EdgeConv}(512,64,64) \xrightarrow{L_2} \mathrm{EdgeConv}(512,64,64) \xrightarrow{L_3}$$
$$\mathrm{FC}(512,64,1024) \xrightarrow{L_4} \mathrm{MaxPool}(512,1024) \xrightarrow{L_5} [L_1,L_2,L_3,L_4,L_5] \xrightarrow{L_6}$$
$$\mathrm{FC}(512,1216,256) \xrightarrow{L_7} \mathrm{FC}(512,256,128) \xrightarrow{L_8} \mathrm{FC}(512,128,3).$$

Note that the DGCNN architecture incorporates T-Net network applied to the input.

Training details. We trained our networks using the ADAM (Kingma & Ba, 2014) optimizer, setting the batch size to 32 and 16 for PointNet and DGCNN respectively. We set a fixed learning rate of 0.001. All models were trained for 250 epochs. Training was done on a single Nvidia V-100 GPU, using PYTORCH deep learning framework (Paszke et al., 2019).

C.2 GRAPHS: EXPRESSIVE POWER

We provide implementation details for the experiments in 5.2. We used two different universal backbones, MLP and GIN equipped with identifiers. The details of those architectures are presented here.

FA/GA GIN+ID architecture. The GIN+ID backbone is based on the GIN (Xu et al., 2018a) network with an addition of identifiers as node features in order to increase expressiveness. For the experiments we used a three-layer GIN with a feature dimension of size 64 and a ReLU activation function. For the added identifiers we defined the input node features as $\boldsymbol{X}^0 = [\boldsymbol{X}^0, \boldsymbol{I}_n]$, where n is the number of nodes in the graph, and to handle graphs of different sizes (EXP dataset) we padded the node features with zeros to fit the size of the maximal graph. Note that we did not apply the permutation generated by the frame on the identifiers.

FA/GA MLP architecture. We used two different MLP networks for the EXP dataset and the GRAPH8c, due to the different size of graphs in the datasets. In the EXP dataset the maximal graph size is 64 and every node in the graph has a one dimensional binary feature, therefore the input for the MLP network is a flatten representation of the graph (with additional padding according to the graph size) $\boldsymbol{x} \in \mathbb{R}^{64^2+64}$. Our architecture consists of layers of the form

$$\mathrm{FC}(d_{\text{in}}, d_{\text{out}}) : \boldsymbol{x} \mapsto \nu\left(\boldsymbol{W}\boldsymbol{x} + \boldsymbol{b}\right)$$

where $\boldsymbol{W} \in \mathbb{R}^{d_{\text{out}} \times d_{\text{in}}}$, $\boldsymbol{b} \in \mathbb{R}^{d_{\text{out}}}$ are the learnable parameters. The final output of the network is denoted by (d_{out}) where d_{out} is the output dimension.

The MLP network structure for the EXP-classify task:

$$\mathrm{FC}(4160, 2048) \overset{L_1}{\to} \mathrm{FC}(2048, 4096) \overset{L_2}{\to} \mathrm{FC}(4096, 2048) \overset{L_3}{\to}$$
$$\mathrm{FC}(2048, 10) \overset{L_4}{\to} \mathrm{FC}(10, 1) \overset{L_5}{\to} (1)$$

with ReLU as the activation function. For the EXP task, which has 10-dimensional output for each graph we omitted the last layer. The GRAPH8c is composed of all the non-isomorphic connected graphs of size 8, hence we did not used any padding of the input here. The nodes have no features and we just used a flatten version of the adjacency matrix. The architecture for the GRAPH8c task:

$$\mathrm{FC}(64, 128) \overset{L_1}{\to} \mathrm{FC}(128, 64) \overset{L_2}{\to} \mathrm{FC}(64, 10) \overset{L_3}{\to} (10)$$

Training details. We followed the protocol from (Balcilar et al., 2021) and trained our model with batch size 100 for 200 epochs. The learning rate was set to 0.001 and did not change during training. For optimization we used the ADAM optimizer. Training was done on a single Nvidia RTX-8000 GPU, using PYTORCH deep learning framework.

Invariance Evaluation. We quantify the permutation invariance of a model ϕ by comparing output of randomly permuted graphs $\rho_1(g_i)\mathbf{X}$, $g_i \in S_n$, $i \in [50]$. The evaluation metric used is the *invariance error*, defined by

$$\frac{1}{m}\sum_{i=1}^{m} \left\|\phi(\rho_1(g_i)\mathbf{X}) - \boldsymbol{v}\right\|_2,$$

where $\boldsymbol{v} = \frac{1}{m}\sum_{i=1}^{m} \phi(\rho_1(g_i)\mathbf{X})$.

The permutation *invariance error* of the models (FA/GA) was measured as a function of the sample size k. We iterated over the entire GRAPH8c dataset and for every graph computed the *invariance error* for the FA-MLP ,GA-MLP and regular MLP models (all with the exact same backbone network). The results presented in Figure 2 (left) are normalized by the error of the regular MLP model. The backbone MLP we used for this experiment is of the form:

$$\mathrm{FC}(64, 128) \overset{L_1}{\to} \mathrm{FC}(128, 128) \overset{L_2}{\to} \mathrm{FC}(128, 128) \overset{L_3}{\to} \mathrm{FC}(128, 10) \overset{L_4}{\to} (10)$$

C.3 GRAPHS: n-BODY PROBLEM

This section describes the implementation details for the n-body experiment from Section 5.3.

FA-GNN architecture. We use the GNN architecture of Gilmer et al. (2017) our base GNN layer, whhere for each node $i \in [n]$ the update rule follows,

$$\boldsymbol{m}_{ij} = \phi_e(\boldsymbol{h}_i^l, \boldsymbol{h}_j^l, a_{ij})$$

$$\boldsymbol{m}_i = \sum_{j \in \mathcal{N}(i)} \boldsymbol{m}_{ij}$$

$$\boldsymbol{h}_i^{l+1} = \phi_h(\boldsymbol{h}_i^l, \boldsymbol{m}_i)$$

where $\mathcal{N}(i)$ are the indices of neighbors of vertex i, $\boldsymbol{h}_i^l$ is the embedding of node i at layer l and $a_{ij} \in \{-1, 1\}$ are edge attributes representing attraction or repelling between pairs of particles. $\boldsymbol{h}^0 \in \mathbb{R}^{n \times 6}$ represents the nodes input features, which in this experiment are a concatenation of the nodes initial 3D position (rotation and translation equivariant) and velocities (rotation-equivariant and translation-invariant). We used node feature dimension of size 60 to maintain a fair comparison with the baselines (Satorras et al., 2021). The functions ϕ_e and ϕ_h are implemented as a two-layer MLP with the SiLU activation function (also chosen for consistency purposes) with an hidden dimension of 121 and 120 respectively.

Our backbone architectures $\Phi^{(2)}_{3d',3}$ and $\Phi^{(1)}_{6,3d'}$, are both composed of two GNN layers; $\Phi^{(2)}_{3d',3}$ has an additional linear embedding of the features as a prefix to the GNN layers while $\Phi^{(2)}_{3d',3}$ is equipped with a two-layer MLP (SiLU activation) as a decoder to extract final positions.

Training details. We followed the protocol from (Satorras et al., 2021) and trained our model with batch size 100 for 10000 epochs. The learning rate was set to 0.001 and did not changed during training. For optimization we used the ADAM optimizer. Training was done on a single Nvidia RTX-6000 GPU, using PYTORCH deep learning framework.