Scalars are universal: Gauge-equivariant machine learning, structured like classical physics

Anonymous Author(s)
Affiliation
Address
email

Abstract

There has been enormous progress in the last few years in designing conceivable (though not always practical) neural networks that respect the gauge symmetries—or coordinate freedom—of physical law. Some of these frameworks make use of irreducible representations, some make use of higher order tensor objects, and some apply symmetry-enforcing constraints. Different physical laws obey different combinations of fundamental symmetries, but a large fraction (possibly all) of classical physics is equivariant to translation, rotation, reflection (parity), boost (relativity), and permutations. Here we show that it is simple to parameterize universally approximating polynomial functions that are equivariant under these symmetries, or under the Euclidean, Lorentz, and Poincaré groups, at any dimensionality d. The key observation is that nonlinear O(d)-equivariant (and related-group-equivariant) functions can be expressed in terms of a lightweight collection of scalars—scalar products and scalar contractions of the scalar, vector, and tensor inputs. These results demonstrate theoretically that gauge-invariant deep learning models for classical physics with good scaling for large problems are feasible right now.

1 Introduction

There is a great deal of current interest in building machine-learning methods that respect exact or approximate symmetries, such as translation, rotation, and physical gauge symmetries [16, 40]. Some of this interest is inspired by the great success of convolutional neural networks (CNNs) [43], which are naturally translation equivariant. The implementation of convolutional layers in CNNs has been given significant credit for the success of deep learning, in a domain (natural images) in which the convolutional symmetry is only approximately valid. In many data-analysis problems in astronomy, physics, and chemistry there are *exact* symmetries that must be obeyed by any generalizable law or rule. If the convolutional symmetry helped in the natural-image domain, then we have high hopes for the value of encoding exact symmetries for problems where these symmetries are known to hold exactly.

In detail, the symmetries of physics are legion. Translation symmetry (including conservation of linear momentum, rotation symmetry (including conservation of angular momentum), and time-translation symmetry (including conservation of energy) are the famous symmetries [57]. But there are many more: there is a form of reflection symmetry (charge-parity-time or CPT); there is a symplectic symmetry that permits reinterpretations of positions and momenta; there are Lorentz and Poincaré symmetries (the fundamental symmetries of Galilean relativity and special relativity) that include velocity boosts; there is the generalization of these (general covariance) that applies in curved spacetime; there are symmetries associated with baryon number, lepton number, flavor, and color; and there are dimensional and units symmetries (not usually listed as symmetries, but they

Submitted to 35th Conference on Neural Information Processing Systems (NeurIPS 2021). Do not distribute.

Orthogonal	$\mathrm{O}(d) = \{Q \in \mathbb{R}^{d\times d} : Q^\top Q = Q\,Q^\top = I_d\}$,
Rotation	$\mathrm{SO}(d) = \{Q \in \mathbb{R}^{d\times d} : Q^\top Q = Q\,Q^\top = I_d,\ \det(Q) = 1\}$
Translation	$\mathrm{T}(d) = \{w \in \mathbb{R}^d\}$
Euclidean	$\mathrm{E}(d) = \mathrm{T}(d) \rtimes \mathrm{O}(d)$
Lorentz	$\mathrm{O}(1,d) = \{Q \in \mathbb{R}^{(d+1)\times(d+1)} : Q^\top \Lambda\, Q = \Lambda,\ \Lambda = \mathrm{diag}([1,-1,..,-1])\}$
Poincaré	$\mathrm{IO}(1,d) = \mathrm{T}(d+1) \rtimes \mathrm{O}(1,d)$
Permutation	$\mathrm{S}_n = \{\sigma : [n] \to [n]$ bijective function$\}$

Table 1: **The groups considered in this work.**

Orthogonal; Lorentz	$Q \star (v_1,..,v_n) = (Q\,v_1,..,Q\,v_n)$
Translation	$w \star (v_1,..,v_n) = (v_1 + w,..,v_k + w, v_{k+1},..,v_n)$ (where the first k vectors are position vectors)
Euclidean; Poincaré	$(w,Q) \star (v_1,..,v_n) = (Q\,v_1 + w,..,Q\,v_k + w, Q\,v_{k+1},..,Q\,v_n)$
Permutation	$\sigma \star (v_1,..,v_n) = (v_{\sigma(1)},..,v_{\sigma(n)})$

Table 2: **The actions of the groups on vectors.** For the Euclidean group, the position vectors are positions of points; for the Poincaré group, the position vectors are positions of *events*.

are) that restrict what kinds of quantities can be multiplied or added. If it were possible to parameterize a universal or universally approximating function space that is explicitly equivariant under a large set of non-trivial symmetries, that function space would—in some sense—contain within it *every possible law of physics*. It would also provide a basis for good new machine learning methods.

Most approaches to gauge-equivariant machine learning make use of irreducible representations of the relevant symmetry groups. Implementing these approaches requires a way to explicitly decompose tensor products of known representations into their irreducible components. This is called the *Clebsch-Gordan problem*. The solution for SO(3) has been implemented, and there is recent exciting progress for other Lie groups [1, 34]. This is an area of current, active research.

Here we give an approach that bypasses the need for a solution to this problem. We find that, for a large class of problems relevant to classical physics, the space of equivariant functions can be constructed from functions only of a complete subset of the scalar products and scalar contractions of the input vectors and tensors. That is, invariant scalars are powerful objects, and point the way to building universally approximating functions that are constrained by exact, non-trivial symmetries.

Equivariance is the form of symmetry core to physical law: For an equivariant function, transforming the input results in an output representation transformed in the same way. An *invariant* function, on the other hand, produces the same output for transformed and non-transformed inputs.

Definition 1*:* Given a function $f : \mathcal{X} \to \mathcal{Y}$ and a group G acting on $\mathcal{X}$ and in $\mathcal{Y}$ as $\star$ (possibly the action is defined differently in $\mathcal{X}$ and $\mathcal{Y}$). We say that f is:

$$G\text{-invariant:} \quad f(g \star x) = f(x) \quad \text{for all } x \in \mathcal{X},\ g \in G\,; \tag{1}$$

$$G\text{-equivariant:} \quad f(g \star x) = g \star f(x) \quad \text{for all } x \in \mathcal{X},\ g \in G\,. \tag{2}$$

Our contributions: In this work we propose a complete and computationally tractable characterization of all scalar functions $f : (\mathbb{R}^d)^n \to \mathbb{R}$, and of all vector functions $h : (\mathbb{R}^d)^n \to \mathbb{R}^d$ that satisfy all of the symmetries of classical physics. The groups corresponding to these symmetries are given in Table 1; they act according to the rules in Table 2. The characterization we find is physically principled: It is based on invariant scalars. It is also connected to the symmetries encoded in the Einstein summation rules, a common notation in physics to write expressions compactly but that also allows only equivariant objects to be produced (see Appendix F).

Our characterization is based on simple mathematical observations. The first is the First Fundamental Theorem of Invariant Theory for O(d): *a function of vector inputs returns an invariant scalar if and only if it can be written as a function only of the invariant scalar products of the input vectors* [76, Section II.A.9]. There is a similar statement for the Lorentz group O$(1,d)$. The second observation

is that it follows from this that *a function of vector inputs returns an equivariant vector if and only if it can be written as a linear combination of invariant scalar functions times the input vectors*. In particular, if $h : (\mathbb{R}^d)^n \to \mathbb{R}^d$ of inputs $v_1, .., v_n$ is $\mathrm{O}(d)$ or $\mathrm{O}(1, d)$-equivariant, then it can be expressed as:

$$h(v_1, v_2, .., v_n) = \sum_{t=1}^{n} f_t\Big(\langle v_i, v_j\rangle_{i,j=1}^{n}\Big)\, v_t \ , \tag{3}$$

where f_t can be arbitrary functions, but if h is a polynomial function the f_t can be chosen to be polynomials. In other words, the $\mathrm{O}(d)$ and $\mathrm{O}(1, d)$-equivariant vector functions are generated as a module over the ring of invariant scalar functions by the projections to each input vector. In this expression, $\langle \cdot, \cdot \rangle$ denotes the invariant scalar product, which can be the usual Euclidean inner product, or the Minkowski inner product defined in terms of a metric Λ (see Table 1):

$$\text{Euclidean: } \langle v_i, v_j\rangle = v_i^\top v_j \ , \qquad \text{Minkowski: } \langle v_i, v_j\rangle = v_i^\top \Lambda\, v_j \ . \tag{4}$$

Our mathematical observations lead to a simple characterization for a very general class of equivariant functions, simpler than any based on irreducible representations or the imposition of symmetries through constraints (these methods are currently state of the art; see Section 2). This implies that very simple neural networks based on message passing—that enormously generalize those in [66]—can universally approximate invariant and equivariant functions. We note that the formulation in (3) might superficially resemble an attention mechanism [73], but it actually comes from the characterization of invariant functions of the group, in particular the picture for $\mathrm{SO}(d)$-invariant functions is a little different (see Proposition 5).

In Section 2 (and Appendix A) we describe the state of the art for encoding symmetries, the expressive power of graph neural networks, and universal approximation of invariant and equivariant functions. In Section 3 we mathematically characterize the invariant and equivariant functions with respect to the groups in Table 1. In Section 4 we present some examples of physically meaningful equivariant functions, and show how to express them in the parameterization developed in Section 3. In Section 5 we discuss which (of all possible) pairwise inner products one ought to provide. In Section 6 we discuss some limitations of our approach, and finally in Section 7 we describe how to use our framework to provide a universally approximating parameterization of equivariant functions.

2 Related work

Group invariant and equivariant neural networks: Symmetries have been used successfully for learning functions on images, sets, point clouds, and graphs. Neural networks can be designed to parameterize classes of functions satisfying different forms of symmetries, from the classical (approximately) translation-invariant convolutional neural networks [44], to more recent architectures that define permutation invariant and equivariant functions on point clouds pioneered by deep sets and pointnets [80, 60, 61], to permutation-equivariant functions on graphs expressed as graph neural networks [30, 67, 39, 21, 27, 13].

For instance, deep nets and pointnets parameterize functions on $(\mathbb{R}^d)^n$ that are invariant or equivariant with respect to the group of permutations S_n acting as in Table 2. Invariant theory shows that all invariant and equivariant functions with respect to such actions can be approximated by easily characterized invariant polynomials [76]. However, the permutation group can act in significantly more complicated ways. For instance, graph neural networks are equivariant with respect to a different action by permutations (conjugation) that is much harder to characterize (see Appendix A).

Since being introduced by [60, 80], deep learning on point clouds has been extremely fruitful, especially in computer vision [32]. Recently, new symmetries and invariances have been incorporated into the design of neural networks on point clouds; especially invariances and equivariances with respect to rigid motions such as translations and rotations. Many architectures have been proposed to satisfy those symmetries such as [71] based on irreducible representations (irreps), [41, 81] based on convolutions, [26] employing spherical harmonics and irreps, [82] using quaternions, and [25] applying a set of constraints to satisfy the symmetries. Most of the implementations of the approaches mentioned above (except for [25]) are limited to 2D or 3D point clouds. We provide an overview of the main approaches below and in Appendix A.

Universal approximation via linear invariant layers and irreducible representations: Universally approximating invariant functions can be obtained by taking universal non-invariant functions and averaging them over the group orbits [79, 56]. However, this approach is not practical for large groups like S_n or infinite groups like $\mathrm{O}(d)$. A classical result in neural networks shows that feed-forward networks with non-polynomial activations can universally approximate continuous functions [46]. This arguably inspired the use of neural networks that are the composition of linear invariant or equivariant layers with compatible non-linear activation functions to create expressive equivariant models [40, 53]. Linear G-equivariant functions can be written in terms of the irreducible representations of G (we explain this and refer the reader to related literature in Appendix A). However, the explicit parameterization of the linear maps is only known for a few groups (for instance, SO(3)-equivariant linear maps are parameterized using the Clebsh-Gordan coefficients [26, 71, 7]). Moreover, very recent work [22] shows that the classes of functions defined in terms of neural networks over irreducible representations in [26, 71] are universal. In particular, every continous SO(3)-equivariant function can be approximated uniformly in compacts sets by those neural networks. Despite universality, there is a limitation to this approach: Although decompositions into irreps are broadly studied in mathematics (also as plethysms), the explicit transformation to the irreps is not known or possible for general groups.

Expressive power of (non-universal) neural networks on graphs: Graph neural networks express functions on graphs that are equivariant with respect to a certain action by permutations. However, the architectures that are used in practice, typically based on message passing [27] or graph convolutions [21, 18], are not universal in general. Implementating a universally approximating graph neural network using the formulation from the previous section would be prohibitively expensive. There is work characterizing the expressive power of message passing neural networks, mainly in terms of the graph isomorphism problem [77, 55, 14, 49, 70, 12], and there is research on the design of graph networks that are expressive enough to perform specific tasks, like solving specific combinatorial optimization problems [6, 11, 58, 37, 78, 36, 8].

Machine learning for physics with symmetry preservation: Machine learning has been applied extensively to problems in physics. While many of these problems require certain symmetries—in many cases, exact symmetries—most applied projects to date rely on the data to encode the symmetry and hope that the model learns it. For instance, CNNs are commonly used to classify galaxy images; data augmentation is used to teach the model rotational symmetry [33, 3, 20, 29]. One well-known example is the Kaggle Galaxy Challenge, a classification competition based on the Galaxy Zoo project [48]; the winning model improved performance by concatenating features from transformed images of each galaxy before further training [19].

There have been recent successes in enforcing physical symmetries in the architecture of the models themselves. In quantum many-body physics, recent work has shown that the symmetries of quantum energy levels on lattices can be enforced with gauge equivariant and invariant neural networks [15, 9, 74, 50, 64, 51]. There is significant work on imposing permutation symmetry in jet assignment for high-energy particle collider experiments with self-attention networks [24, 45]. In molecular dynamics, rotationally invariant neural networks have been shown to better learn molecular properties [2, 75, 68], and Hamiltonian neural networks have been constructed to better preserve molecular conformations [47]. More broadly, Hamiltonian networks have been shown to improve physical characteristics, such as better conservation of energy, and to better generalize [31, 65, 83], and Lagrangian neural networks can also enforce conservation laws [52, 17].

3 Equivariant maps

In this Section we provide a simple characterization of all invariant and equivariant functions with respect to the actions in Table 2 by the groups in Table 1. In what follows, $v_1, v_2, \ldots, v_n$ will be vectors in $\mathbb{R}^d$, G will be a group acting in $\mathbb{R}^d$ and $(\mathbb{R}^d)^n$ as in Table 2, $f : (\mathbb{R}^d)^n \to \mathbb{R}$ will be an invariant function with respect to the action, and $h : (\mathbb{R}^d)^n \to \mathbb{R}^d$ will be an equivariant function with respect to the same action. V will denote a $d \times n$ matrix whose columns are the v_i vectors.

O(d) and SO(d) invariance and equivariance: The following classical result in invariant theory (e.g., [76, Section II.A.9]) shows that $\mathrm{O}(d)$ invariant functions are functions of the scalars $v_i^\top v_j$.

Lemma 1 (First Fundamental Theorem for $\mathrm{O}(d)$)*:* If f is an $\mathrm{O}(d)$-invariant scalar function of vector inputs $v_1, \ldots, v_n \in \mathbb{R}^d$, then $f(v_1, v_2, \ldots, v_n)$ can be written as a function of only the scalar products

of the v_i. That is, there is a function $g(\cdot)$ such that

$$f(v_1, v_2, .., v_n) = g(V^\top V) = g\big((v_i^\top v_j)_{i,j=1}^n\big) \ . \tag{5}$$

Proof. Given $M = V^\top V \in \mathbb{R}^{n\times n}$, we can reconstruct $v_1, .., v_n$ modulo the orthogonal group O(d) by computing the Cholesky decomposition of M (see for instance [72] p. 174). Therefore, the function f is uniquely determined by the inner product matrix $V^\top V$. □

The classical theorem also includes the fact that if f is polynomial (in the entries of the v_j's), then g can be taken to be polynomial. In Section 5 we observe that the function g can be determined by a small subset of the scalars $v_i^\top v_j$. There is an analogous statement for SO(d) (again see [76, Section II.A.9]):

Lemma 2 (First Fundamental Theorem for SO(d))*:* If f is an SO(d)-invariant scalar function of vector inputs $v_1, v_2, .., v_n \in \mathbb{R}^d$, then $f(v_1, v_2, .., v_n)$ can be written as a function of the scalar products of the v_i and the $d \times d$ subdeterminants of the $d \times n$ matrix V.

Lemma 3*:* If h is an O(d)-equivariant vector function of n vector inputs $v_1, v_2, .., v_n$, then $h(v_1, v_2, .., v_n)$ must lie in the subspace spanned by the input vectors $v_1, v_2, .., v_n$.

Proof. Let $\{w_1, .., w_r\} \subset \mathbb{R}^d$ be an orthonormal basis of the orthogonal complement to span$(v_1, .., v_n)$. Then we can write $h(v_1, .., v_n) = \sum_{t=1}^n \alpha_t v_t + \sum_{t=1}^r \beta_t w_t$ for some choice of α_t, β_t. We claim that the equivariance of h implies $\beta_t = 0$ for $t = 1, .., r$: Consider $\hat{Q} \in O(d)$ such that $\hat{Q}(v) = v$ for all span$(v_1, .., v_n)$, and $\hat{Q}(w) = -w$ for all w in the orthogonal complement.

Since $(\hat{Q}v_1, .., \hat{Q}v_n) = (v_1, .., v_n)$ we have $h(\hat{Q}v_1, .., \hat{Q}v_n) = \sum_{t=1}^d \alpha_t v_t + \sum_{t=1}^r \beta_t w_t$ while $\hat{Q}(h(v_1, .., v_n)) = \sum_{t=1}^d \alpha_t v_t - \sum_{t=1}^r \beta_t w_t$. Therefore equivariance implies that all $\beta_t = 0$. □

Note that Lemma 3 doesn't hold for SO(d): although when the codimension of span$(v_1, \ldots, v_n)$ is 0 or ≥ 2 a similar argument works, the situation $\dim_{\mathbb{R}} \operatorname{span}(v_1, \ldots, v_n) = d - 1$ breaks the proof, and there do exist equivariant vector functions that don't lie in the span when it has codimension 1. For instance, the cross product of two vectors in $\mathbb{R}^3$ is an SO(3)-equivariant function that is not in the span of its inputs. Proposition 5 shows that generalized cross products are the *only* way that SO(d)-equivariant vector functions can escape the span of the inputs. We further discuss this in Appendix F. Proposition 4 below gives a characterization of all O(d)-equivariant functions in terms of the scalars. We prove it in Appendix B.

Proposition 4*:* If h is an O(d)-equivariant vector function of n vector inputs $v_1, v_2, .., v_n$, then there are n O(d)-invariant scalar functions $f_t(\cdot)$ such that

$$h(v_1, v_2, .., v_n) = \textstyle\sum_{t=1}^n f_t(v_1, v_2, .., v_n)\, v_t \ . \tag{6}$$

Moreover, if h is a polynomial function, the f_t can be chosen to be polynomial.

The equivariant scalar functions form a module over the ring of invariant scalar functions. Proposition 4 states that this module is generated by the projections $(v_1, \ldots, v_n) \mapsto v_j$. Proposition 5 extends our results to SO(d) in terms of the generalized cross product. The definition of the generalized cross product and the proof of Proposition 5 are in Appendix B.

Proposition 5*:* If h is an SO(d)-equivariant vector function of n vector inputs $v_1, v_2, .., v_n$ then the characterization of Proposition 4 holds, except when $v_1, v_2, .., v_n$ span a $(d-1)$-dimensional space. In that case, there exist O(d)-invariant scalar functions $f_t(\cdot)$ such that

$$h(v_1, v_2, .., v_n) = \textstyle\sum_{t=1}^n f_t(v_1, v_2, .., v_n)\, v_t + \sum_{S\in\binom{[n]}{d-1}} f_S(v_1, v_2, .., v_n)\, v_S \ , \tag{7}$$

where $[n] := \{1, \ldots, n\}$, $\binom{[n]}{d-1}$ is the set of all $(d-1)$-subsets of $[n]$, and v_S represents the generalized cross product of vectors v_j with $j \in S$ (taken in ascending order). Moreover, if h is polynomial, the f_t and f_S can be taken to be polynomial.

E(d) invariance and equivariance: When modelling point clouds, we may want to express functions that are translation invariant or equivariant with respect to a subset of the input vectors (for instance, position vectors are translation equivariant). To this end, we consider the group of translations parametrized by $w \in \mathbb{R}^d$ acting on the vectors in $(\mathbb{R}^d)^n$ by translating the position vectors, and leaving everything else unchanged: $w\star(v_1, .., v_k, v_{k+1}, .., v_n) = (v_1+w, .., v_k+w, v_{k+1}, .., v_n)$.

In this Section we characterize all functions that are translation and rotation invariant/equivariant. In the exposition below we assume for simplicity that all vectors are position vectors, but the results generalize trivially to a mix of vectors.

Lemma 6*:* Any translation-invariant function $f : (\mathbb{R}^d)^n \to \mathbb{R}^\ell$ with inputs $v_1, .., v_n$ can be written uniquely as $f(v_1, v_2, .., v_n) = \tilde{f}(v_2 - v_1, .., v_n - v_1)$, where $\tilde{f} : (\mathbb{R}^d)^{n-1} \to \mathbb{R}^\ell$ is an arbitrary function. If f is polynomial, $\tilde{f}$ is polynomial, and vice versa. If f is equivariant for the action of any given subgroup $G \subset GL(n, \mathbb{R})$, then so is $\tilde{f}$, and vice versa.

The proof is given in Appendix C. There is nothing special about subtracting v_1; there exist more natural choices to express translation invariant functions. For instance, one classical way in physics to express translation invariance is to take class representatives of the form $(v_1, .., v_n)$ where $\sum v_i = 0$ (for example, subtracting the center of mass). We will continue to use the representative per class from Lemma 6 for simplicity.

Lemma 7*:* An O(d)-equivariant function $h : (\mathbb{R}^d)^n \to \mathbb{R}^d$ that is translation-invariant can be written as (6) where the f_t are O(d) and translation invariant and $\sum_{t=1}^n f_t(v_1, v_2, .., v_n) = 0$. Similarly, if h is translation-equivariant then we can choose $\sum_{t=1}^n f_t(v_1, v_2, .., v_n) = 1$.

Proof. Let $h : (\mathbb{R}^d)^n \to \mathbb{R}^d$ be translation-invariant and O(d)-equivariant. Using Lemma 6 we can consider an O(d)-equivariant function $\tilde{h} : (\mathbb{R}^d)^{n-1} \to \mathbb{R}^d$ so that $h(v_1, v_2, .., v_n) = \tilde{h}(v_2 - v_1, .., v_n - v_1)$. Therefore there exists O(d) invariant functions $\tilde{f}_t$

$$\tilde{h}(v_2 - v_1, .., v_n - v_1) = \textstyle\sum_{t=2}^n \tilde{f}_t(v_2 - v_1, .., v_n - v_1)(v_t - v_1). \tag{8}$$

This implies h can be written as

$$h(v_1, v_2, .., v_n) = \textstyle\sum_{t=2}^n f_t(v_1, .., v_n)\, v_t - (\sum_{t=2}^n f_t(v_1, .., v_n))\, v_1 \tag{9}$$

where the functions f_t are O(d)- and translation-invariant, which has the claimed form.

For the case where h is O(d)- and translation-equivariant we first observe that it suffices to take one representative per orbit, define the function on that representative, and extend it everywhere else by translations:

$$h(v_1, .., v_n) = \tilde{h}(v_2 - v_1, .., v_n - v_1) + v_1, \tag{10}$$

where $\tilde{h}$ is O(d)-equivariant. Therefore any function h that is O(d)- and translation-equivariant can be written as

$$h(v_1, .., v_n) = \textstyle\sum_{t=2}^n \tilde{f}_t(v_2 - v_1, .., v_n - v_1)(v_t - v_1) + v_1 \tag{11}$$

$$= \textstyle\sum_{t=2}^n f_t(v_1, v_2.., v_n)\, v_t + (1 - \sum_{t=2}^n f_t(v_1, v_2.., v_n))\; v_1. \tag{12}$$

□

Lorentz symmetry: The Lorentz group acts on Minkowski spacetime as Lorentz transformations, which keep the metric tensor invariant. Lorentz transformations relate space and time between inertial reference frames, which move at a constant relative velocity; spacetime intervals are invariant across frames. The group is made up of spatial rotations in the three space dimensions and linear velocity "boosts" along each dimension. This set of symmetries—required for special relativity—is not O(4), but rather the non-compact group O(1,3) defined in Table 1.

The characterization of invariant and equivariant functions we obtained for the orthogonal group can be extended to the Lorentz group, obtaining a very similar result, summarized in Proposition 8 and proven in Appendix D.

Proposition 8*:* A function $h : (\mathbb{R}^d)^n \to \mathbb{R}^d$ is Lorentz-equivariant (with respect to the action in Table 2) if and only if

$$h(v_1, .., v_n) = \textstyle\sum_{t=1}^n f_t(v_1, .., v_n)\, v_t \tag{13}$$

where $f_t : (\mathbb{R}^d)^n \to \mathbb{R}$ are Lorentz-invariant scalar functions. Moreover, the functions f_t are uniquely determined by the pairwise Minkowski inner product $\langle \cdot, \cdot \rangle$ (4):

$$f_t(v_1, .., v_n) = g_t(\langle v_i, v_j \rangle_{i,j=1}^n)\,. \tag{14}$$

If h is polynomial, the f_t and corresponding g_t can be taken to be polynomial.

Poincaré symmetry: The Poincaré group combines translation symmetry with Lorentz symmetry. Together these complete the symmetries of special relativity, forming the full group of spacetime transformations that preserve the Minkowski metric. The generalization from a Lorentz-equivariant formulation to a Poincaré-equivariant formulation is similar to the generalization from O(d) to E(d): The *position* vectors take on a special role, in which only *differences* of position can appear as vector inputs to the functions; all other vectors can act unchanged.

Proposition 9 generalizes the results above to the Poincaré group action. As before, we assume that all vectors are position vectors for simplicity.

Proposition 9*:* A function $h : (\mathbb{R}^d)^n \to \mathbb{R}^d$ is Poincaré-equivariant (with respect to the action in Table 2) if and only if

$$h(v_1, .., v_n) = \sum_{t=1}^{n} f_t(v_1, .., v_n)\, v_t \tag{15}$$

where the $f_t : (\mathbb{R}^d)^n \to \mathbb{R}$ are translation and Lorentz-invariant scalar functions, determined as in (14) by the pairwise Minkowski inner products (4), but also satisfying $\sum_{t=1}^{n} f_t(v_1, v_2, .., v_n) = 1$. Furthermore, if h is polynomial, the f_t and corresponding g_t (as in (14)) can be taken to be polynomial.

Permutation invariance: Most physics problems are permutation-invariant, in that once you know the masses, sizes, shapes, and so on of the objects in the problem, the physical predictions are invariant to labeling.In particle physics, this invariance is raised to a fundamental symmetry; fundamental particles (like electrons) are identical and exchangeable.

The characterization of permutation-invariant functions with respect to the action in Table 2 is classical [76, pp. 36–39]. Here we prove an extension that describes permutation-invariant vector functions that are also O(d)-equivariant. The proof is in Appendix E.

Proposition 10*:* Let $h : (\mathbb{R}^d)^n \to \mathbb{R}^d$ be an O(d)-equivariant (or O($1, d-1$)-equivariant) function that is also permutation-invariant with respect to the action in Table 2. Then h can be written as

$$h(v_1, .., v_n) = \sum_{t=1}^{n} f(v_t, v_1, .., v_{t-1}, v_{t+1}, .., v_n)\, v_t, \tag{16}$$

where $f(\mathbb{R}^d)^n \to \mathbb{R}$ is O(d)-invariant (or O($1, d-1$)-invariant) and permutation-invariant with respect to the last $n-1$ inputs.

4 Examples

Here we briefly state two classical physics expressions that obey all the symmetries, and show how to formulate them in terms of invariant scalars.

Total mechanical energy: In Newtonian gravity, the total mechanical energy T of n particles with scalar masses m_i, vector positions r_i, and vector velocities v_i is a scalar[1] function:

$$T = \sum_{i=1}^{n} \tfrac{1}{2}\, m_i\, |v_i|^2 - \sum_{i=1}^{n} \sum_{\substack{j=1 \\ j\neq i}}^{n} \frac{G\, m_i\, m_j}{|r_i - r_j|}\ , \tag{17}$$

where G is Newton's constant (a fundamental constant, and hence scalar). Since $|a| \equiv (a^\top a)^{1/2}$, this expression (17) is manifestly constructed from functions only of scalars m_i and scalar products of vectors. It is also worthy of note that the positions r_i only appear in differences of position.

Electromagnetic force law: The total electromagnetic force F acting on a test particle of charge, 3-vector position, and 3-vector velocity (q, r, v) given a set of n other charges (q_i, r_i, v_i) is a vector function [35]:

$$F = \underbrace{\sum_{i=1}^{n} k\, q\, q_i\, \frac{(r - r_i)}{|r - r_i|^3}}_{\text{electrostatic force}} + \underbrace{\sum_{i=1}^{n} k\, q\, q_i\, \frac{v \times (v_i \times (r - r_i))}{c^2\, |r - r_i|^3}}_{\text{magnetic force}}\ , \tag{18}$$

where k is an electromagnetic constant, c is the speed of light, and $a \times b$ represents the cross product that produces a pseudo-vector perpendicular to vectors a and b according to the right-hand rule. This doesn't obviously obey our equivariance requirements, because cross products deliver parity-violating pseudo-vectors; these can't be written in O(3)-equivariant form. However, a cross of a

[1] The energy is *not* a scalar in special relativity or general relativity, but it is a scalar in Newtonian physics.

cross of vectors is a vector, so this expression is in fact O(3)-equivariant, as are all forces (because forces must be O(3)-equivariant vectors in order for the theory to be self-consistent).

We can expand the vector triple product using the identity $a \times (b \times c) = (a^\top c)\, b - (a^\top b)\, c$:

$$\begin{aligned} F &= \sum_{i=1}^{n} k\, q\, q_i \, \frac{(r - r_i)}{|r - r_i|^3} + \sum_{i=1}^{n} k\, q\, q_i \, \frac{(v^\top (r - r_i))\, v_i - (v^\top v_i)\, (r - r_i)}{c^2\, |r - r_i|^3} \\ &= \sum_{i=1}^{n} k\, q\, q_i \left(1 - \frac{v^\top v_i}{c^2}\right) \frac{(r - r_i)}{|r - r_i|^3} + \sum_{i=1}^{n} k\, q\, q_i \, \frac{(v^\top (r - r_i))\, v_i}{c^2\, |r - r_i|^3} \,, \end{aligned} \tag{19}$$

where the quantity $v^\top v_i$ is the scalar product of the velocities. All of the quantities are now straightforwardly functions of invariant scalar products times the input vectors.

5 How many scalars are needed?

Our analysis in Section 3 shows that the invariant and equivariant functions of interest (under actions in Table 2 from groups in Table 1) with input vectors $v_1, .., v_n$, can be expressed in terms of the scalars $\langle v_i, v_j \rangle_{i \geq j=1}^{n}$ (4). This greatly simplifies the parameterization of such functions, but it significantly increases the number of features if $n \gg d$. In this Section we remark that the scalars can be uniquely determined by a small subset of size approximately $(d+1)\, n$. This is related to the rigidity theory of Gram matrices [63] that answers when there exists a unique set of vectors that realize a partial set of distances, and it is closely related to the low rank matrix completion problem [69]. Furthermore, there is a vast literature studying high probability robust reconstruction of all scalars from a random subset via convex optimization techniques [10].

Given a rank d, $n \times n$ matrix M, Example 4 in [59] shows that M is almost always uniquely determined by the entries $\Omega(M) := M_{i,i+s}$ $i = 1, .., n$, $s = 0, .., d$, considering the indices "wrap around" (i.e., $M_{i,n+s}$ corresponds to $M_{i,s} = M_{i,n+s(\mathrm{mod}\ n)}$). In particular invariant functions $f : (\mathbb{R}^d)^n \to \mathbb{R}^\ell$, expressed as (5)

$$f(v_1, .., v_n) = g(M) = \tilde{g}(\langle v_i, v_{i+s(\mathrm{mod}\ n)} \rangle_{1 \leq i \leq n,\ 0 \leq s \leq d}) = \tilde{g}(\Omega(M)), \tag{20}$$

where M is either $V^\top V$ or $V^\top \Lambda\, V$. Therefore, any function $\tilde{g} : \Omega(M) \to \mathbb{R}^\ell$ uniquely determines f on almost all possible inputs $v_1, .., v_n$. This observation provides a parameterization for Lorentz and orthogonal-invariant and equivariant functions with a small number of scalars.

We note that the results from [59] don't assume the matrix M is positive semi-definite, only that it is low rank. This is useful since M is not positive semi-definite in the Minkowski case. We also remark that the learning of the functions (3) and (5) can be done from $\Omega(M)$ without ever needing to compute all the scalars.

One disadvantage of the subset of scalars proposed in this section is that the sampling procedure $\Omega(M)$ is not permutation invariant. One of our future goals is to find a set of permutation invariant scalars that are universally expressive.

6 Limitations and caveats

Although the principal results presented here work for many groups, and work naturally at all spatial dimensions d (unlike methods based on irreps, for example), they do not solve all problems for all use cases of equivariant machine learning. For one, there are myriad groups—and especially discrete groups—that apply to physical and chemical systems where invariant and equivariant functions do not have such a nice characterization. Such is the case for the GNNs discussed in Appendix A.

One example of a situation in which our formulation might not be practical is provided by multipole expansions (for example, those used in the fast multipole method [5] and n-body networks [40]). In the fast multipole method, a hierarchical spatial graph is constructed, and high-order tensors are used to aggregate information from lower-level nodes into higher-level nodes. This aggregation is concise and linear when it is performed using high-order tensors; this aggregation is hard (or maybe impossible) when only scalars can be transmitted, without the use of the irreps of the relevant symmetry groups. This is a research direction we are currently exploring.

Another example that suggests that our universal functions might be cumbersome is the current forms in which classical theories—such as electromagnetism and general relativity—are written. For example, in Section 4 we showed that the electromagnetic force law can be written in the form of functions of scalars and scalar products times vectors, but that is *not* how the theory is traditionally written. It is traditionally written in terms of the magnetic field (a pseudo-vector) or the electromagnetic tensor (an order-2 tensor). As another example, general relativity is traditionally written in terms of contractions of an order-4 curvature tensor. That is, although the theories can in principle be written in the forms we suggest, they will in general be much more concise or simple or clear in forms that make use of higher-order or non-equivariant forms. Stated in terms of machine learning, this might mean that although neural networks that make use of our formalisms could be universally approximating, they might need to be wider or deeper than networks that make use of more sophisticated tools, like the irreps of group theory.

Finally, although our results apply to many physically relevant groups, they do not encode all of the symmetries of classical physics. For example, one critical symmetry is the dimensional or units symmetry: You cannot add or subtract terms that have different units (positions and forces, for example). This symmetry or consideration has implications for the construction of valid polynomials. It also implies that only dimensionless (unitless) scalar quantities can be the arguments of large classes of nonlinear functions, including exponentials or sigmoids. These additional symmetries must be enforced at present with some additional considerations of network architecture or constraints.

7 Model-building considerations

The results in Section 3 provide a simple characterization of all scalar functions and vector functions that satisfy important symmetries (see Table 1) for classical physics and special relativity. In this Section we discuss how we might use this to design a model.

First we need to identify the inputs, what groups act on them, and how the group action affects the inputs and outputs. For example, the translation group acts differently on position vectors than displacement vectors, or any other kind of vectors (velocities, accelerations, fields, etc). Moreover, permutations don't typically act on the entire set of inputs but on tuples of inputs like the individual-particle charge, position, and velocity (q_i, r_i, v_i) in the electromagnetic example in Section 4.

The causal and group-action structure of the model must correspond to the structure of the physics problem. For example, suppose we are interested in solving an n-body problem in which we predict the trajectories of a set of interacting charged particles. The target functions $h_i()$, which predict the position of particle i at time t, can be described as functions of that particle's charge, position, and velocity, and the set of charges, positions, and velocities of all the others. The group actions (rotation, translation, and permutation) can be written as:

$$h_i(q_i, Q\,r_i, Q\,v_i, (q_j, Q\,r_j, Q\,v_j)_{j=1;j\neq i}^n) = Q\,h_i(q_i, r_i, v_i, (q_j, r_j, v_j)_{j=1;j\neq i}^n) \tag{21}$$

$$h_i(q_i, r_i + w, v_i, (q_j, r_j + w, v_j)_{j=1;j\neq i}^n) = h_i(q_i, r_i, v_i, (q_j, r_j, v_j)_{j=1;j\neq i}^n) + w \tag{22}$$

$$h_i(q_i, r_i, v_i, (q_{\sigma_i(j)}, r_{\sigma_i(j)}, v_{\sigma_i(j)})_{j=1;\sigma_i(j)\neq i}^n) = h_i(q_i, r_i, v_i, (q_j, r_j, v_j)_{j=1;j\neq i}^n) \tag{23}$$

$$h_{\sigma(i)}(q_i, r_i, v_i, (q_j, r_j, v_j)_{j=1;j\neq i}^n) = h_i(q_i, r_i, v_i, (q_j, r_j, v_j)_{j=1;j\neq i}^n)\,, \tag{24}$$

where $\sigma_i(j)$ is a permutation that fixes i, i.e., $\sigma_i(i) = i$. The primary results of this paper imply that, in this n-body case, the rotation symmetries can be enforced by constructing the invariant scalar products, and building functions thereof. The translation symmetries can be enforced by taking positional differences with respect to r_i prior to constructing the scalars. The first permutation symmetry (23) can be enforced by working on a set-based neural network (or graph neural network), and the second permutation symmetry (24) can be enforced by taking all the functions $h_i = h$ to be identical. In general, graph-based or set-based methods are probably good frameworks for physics problems [17, 65]. We suggest a construction based on message passing neural networks in Appendix G.

This work provides a method—based on invariant scalars—to parameterize all functions that satisfy all of the symmetries of classical physics. We expect this theoretical tool to create new opportunities for machine learning in the natural sciences.

References

[1] A. Alex, M. Kalus, A. Huckleberry, and J. von Delft. A numerical algorithm for the explicit calculation of SU(N) and SL(N, C) Clebsch–Gordan coefficients. *Journal of Mathematical Physics*, 52(2):023507, 2011.

[2] B. Anderson, T. S. Hy, and R. Kondor. Cormorant: Covariant molecular neural networks. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019.

[3] A. K. Aniyan and K. Thorat. Classifying radio galaxies with the convolutional neural network. *The Astrophysical Journal Supplement Series*, 230(2):20, Jun 2017.

[4] W. Azizian and M. Lelarge. Characterizing the expressive power of invariant and equivariant graph neural networks. *arXiv:2006.15646*, 2020.

[5] R. Beatson and L. Greengard. A short course on fast multipole methods. *Wavelets, Multilevel Methods and Elliptic PDEs*, 1:1–37, 1997.

[6] Y. Bengio, A. Lodi, and A. Prouvost. Machine learning for combinatorial optimization: a methodological tour d'horizon. *European Journal of Operational Research*, 2020.

[7] A. Bogatskiy, B. Anderson, J. Offermann, M. Roussi, D. Miller, and R. Kondor. Lorentz group equivariant neural network for particle physics. In *International Conference on Machine Learning*, pages 992–1002. PMLR, 2020.

[8] G. Bouritsas, F. Frasca, S. Zafeiriou, and M. M. Bronstein. Improving graph neural network expressivity via subgraph isomorphism counting. *arXiv:2006.09252*, 2020.

[9] D. Boyda, G. Kanwar, S. Racanière, D. J. Rezende, M. S. Albergo, K. Cranmer, D. C. Hackett, and P. E. Shanahan. Sampling using $SU(N)$ gauge equivariant flows. *Phys. Rev. D*, 103(7):074504, 2021.

[10] E. J. Candes and B. Recht. Exact low-rank matrix completion via convex optimization. In *2008 46th Annual Allerton Conference on Communication, Control, and Computing*, pages 806–812. IEEE, 2008.

[11] Q. Cappart, D. Chételat, E. Khalil, A. Lodi, C. Morris, and P. Veličković. Combinatorial optimization and reasoning with graph neural networks. *arXiv:2102.09544*, 2021.

[12] Z. Chen, L. Chen, S. Villar, and J. Bruna. Can graph neural networks count substructures? In H. Larochelle, M. Ranzato, R. Hadsell, M. F. Balcan, and H. Lin, editors, *Neural Information Processing Systems*, volume 33, pages 10383–10395. Curran Associates, Inc., 2020.

[13] Z. Chen, L. Li, and J. Bruna. Supervised community detection with line graph neural networks. *International Conference on Learning Representations*, 2019.

[14] Z. Chen, S. Villar, L. Chen, and J. Bruna. On the equivalence between graph isomorphism testing and function approximation with gnns. In *Neural Information Processing Systems*, pages 15868–15876, 2019.

[15] K. Choo, G. Carleo, N. Regnault, and T. Neupert. Symmetries and many-body excitations with neural-network quantum states. *Physical Review Letters*, 121:167204, Oct 2018.

[16] T. Cohen, M. Weiler, B. Kicanaoglu, and M. Welling. Gauge equivariant convolutional networks and the icosahedral cnn. In *International Conference on Machine Learning*, pages 1321–1330. PMLR, 2019.

[17] M. D. Cranmer, S. Greydanus, S. Hoyer, P. W. Battaglia, D. N. Spergel, and S. Ho. Lagrangian neural networks. *arXiv:2003.04630*, 2020.

[18] M. Defferrard, X. Bresson, and P. Vandergheynst. Convolutional neural networks on graphs with fast localized spectral filtering. In *neural information processing systems*, pages 3844–3852, 2016.

[19] S. Dieleman, K. W. Willett, and J. Dambre. Rotation-invariant convolutional neural networks for galaxy morphology prediction. *Monthly Notices of the Royal Astronomical Society*, 450(2):1441–1459, Apr 2015.

[20] H. Domínguez Sánchez, M. Huertas-Company, M. Bernardi, D. Tuccillo, and J. L. Fischer. Improving galaxy morphologies for sdss with deep learning. *Monthly Notices of the Royal Astronomical Society*, 476(3):3661–3676, Feb 2018.

[21] D. K. Duvenaud, D. Maclaurin, J. Iparraguirre, R. Bombarell, T. Hirzel, A. Aspuru-Guzik, and R. P. Adams. Convolutional networks on graphs for learning molecular fingerprints. In *Neural Information Processing systems*, pages 2224–2232, 2015.

[22] N. Dym and H. Maron. On the universality of rotation equivariant point cloud networks. *arXiv:2010.02449*, 2020.

[23] A. Einstein. Die Grundlage der allgemeinen Relativitätstheorie. *Annalen der Physik*, 354(7):769–822, Jan. 1916.

[24] M. J. Fenton, A. Shmakov, T.-W. Ho, S.-C. Hsu, D. Whiteson, and P. Baldi. Permutationless Many-Jet Event Reconstruction with Symmetry Preserving Attention Networks. *arXiv:2010.09206*, 2020.

[25] M. Finzi, M. Welling, and A. G. Wilson. A practical method for constructing equivariant multilayer perceptrons for arbitrary matrix groups. *arXiv:2104.09459*, 2021.

[26] F. Fuchs, D. Worrall, V. Fischer, and M. Welling. Se (3)-transformers: 3d roto-translation equivariant attention networks. *Neural Information Processing Systems*, 33, 2020.

[27] J. Gilmer, S. S. Schoenholz, P. F. Riley, O. Vinyals, and G. E. Dahl. Neural message passing for quantum chemistry. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pages 1263–1272. JMLR. org, 2017.

[28] J. Gilmer, S. S. Schoenholz, P. F. Riley, O. Vinyals, and G. E. Dahl. Neural message passing for quantum chemistry. In D. Precup and Y. W. Teh, editors, *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pages 1263–1272. PMLR, 2017.

[29] R. González, R. Muñoz, and C. Hernández. Galaxy detection and identification using deep learning and data augmentation. *Astronomy and Computing*, 25:103–109, 2018.

[30] M. Gori, G. Monfardini, and F. Scarselli. A new model for learning in graph domains. In *Proceedings. 2005 IEEE International Joint Conference on Neural Networks, 2005.*, volume 2, pages 729–734. IEEE, 2005.

[31] S. Greydanus, M. Dzamba, and J. Yosinski. Hamiltonian neural networks. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019.

[32] Y. Guo, H. Wang, Q. Hu, H. Liu, L. Liu, and M. Bennamoun. Deep learning for 3d point clouds: A survey. *IEEE Transactions on Pattern Analysis and machine intelligence*, 2020.

[33] M. Huertas-Company, R. Gravet, G. Cabrera-Vives, P. G. Pérez-González, J. S. Kartaltepe, G. Barro, M. Bernardi, S. Mei, F. Shankar, P. Dimauro, and et al. A catalog of visual-like morphologies in the 5 candels fields using deep learning. *The Astrophysical Journal Supplement Series*, 221(1):8, Oct 2015.

[34] A. Ibort, A. López Yela, and J. Moro. A new algorithm for computing branching rules and clebsch–gordan coefficients of unitary representations of compact groups. *Journal of Mathematical Physics*, 58(10):101702, 2017.

[35] J. D. Jackson. *Classical Electrodynamics; 2nd ed.* Wiley, New York, NY, 1975.

[36] C. K. Joshi, T. Laurent, and X. Bresson. An efficient graph convolutional network technique for the travelling salesman problem. *arXiv:1906.01227*, 2019.

[37] N. Karalias and A. Loukas. Erdos goes neural: an unsupervised learning framework for combinatorial optimization on graphs. *arXiv:2006.10643*, 2020.

[38] N. Keriven and G. Peyré. Universal invariant and equivariant graph neural networks. In *Neural Information Processing Systems*, pages 7092–7101, 2019.

[39] T. N. Kipf and M. Welling. Semi-supervised classification with graph convolutional networks. *arXiv:1609.02907*, 2016.

[40] R. Kondor. N-body networks: a covariant hierarchical neural network architecture for learning atomic potentials. *arXiv:1803.01588*, 2018.

[41] R. Kondor, H. T. Son, H. Pan, B. Anderson, and S. Trivedi. Covariant compositional networks for learning graphs. *arXiv:1801.02144*, 2018.

[42] P.-J. K. Kristof T. Schütt, H. E. Sauceda, S. Chmiela, A. Tkatchenko, and K.-R. Müller. Schnet: a continuous-filter convolutional neural network for modeling quantum interactions. In *Neural Information Processing Systems*, pages 992–1002, 2017.

[43] Y. LeCun, Y. Bengio, and G. Hinton. Deep learning. *Nature*, 521(7553):436–444, 2015.

[44] Y. LeCun, B. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. Hubbard, and L. D. Jackel. Backpropagation applied to handwritten zip code recognition. *Neural Computation*, 1(4):541–551, 1989.

[45] J. S. H. Lee, I. Park, I. J. Watson, and S. Yang. Zero-Permutation Jet-Parton Assignment using a Self-Attention Network. *arXiv:2012.03542*, 2020.

[46] M. Leshno, V. Y. Lin, A. Pinkus, and S. Schocken. Multilayer feedforward networks with a nonpolynomial activation function can approximate any function. *Neural Networks*, 6(6):861–867, 1993.

[47] Z. Li, S. Yang, G. Song, and L. Cai. Hamnet: Conformation-guided molecular representation with hamiltonian neural networks, 2021.

[48] C. J. Lintott, K. Schawinski, A. Slosar, K. Land, S. Bamford, D. Thomas, M. J. Raddick, R. C. Nichol, A. Szalay, D. Andreescu, P. Murray, and J. Vandenberg. Galaxy Zoo: morphologies derived from visual inspection of galaxies from the Sloan Digital Sky Survey*. *Monthly Notices of the Royal Astronomical Society*, 389(3):1179–1189, 2008.

[49] A. Loukas. What graph neural networks cannot learn: depth vs width. In *International Conference on Learning Representations*, 2019.

[50] D. Luo, G. Carleo, B. K. Clark, and J. Stokes. Gauge equivariant neural networks for quantum lattice gauge theories, 2020.

[51] D. Luo, Z. Chen, K. Hu, Z. Zhao, V. M. Hur, and B. K. Clark. Gauge Invariant Autoregressive Neural Networks for Quantum Lattice Models. *arXiv:2101.07243*, 2021.

[52] M. Lutter, C. Ritter, and J. Peters. Deep lagrangian networks: Using physics as model prior for deep learning. In *7th International Conference on Learning Representations (ICLR)*. ICLR, May 2019.

[53] H. Maron, H. Ben-Hamu, N. Shamir, and Y. Lipman. Invariant and equivariant graph networks. In *International Conference on Learning Representations*, 2018.

[54] H. Maron, E. Fetaya, N. Segol, and Y. Lipman. On the universality of invariant networks. In *International Conference on Machine Learning*, pages 4363–4371, 2019.

[55] C. Morris, M. Ritzert, M. Fey, W. L. Hamilton, J. E. Lenssen, G. Rattan, and M. Grohe. Weisfeiler and leman go neural: Higher-order graph neural networks. *Association for the Advancement of Artificial Intelligence*, 2019.

[56] R. Murphy, B. Srinivasan, V. Rao, and B. Ribeiro. Relational pooling for graph representations. In *International Conference on Machine Learning*, pages 4663–4673, 2019.

[57] E. Noether. Invariant variation problems. *Transport Theory and Statistical Physics*, 1(3):186–207, 1971.

[58] A. Nowak, S. Villar, A. S. Bandeira, and J. Bruna. A note on learning algorithms for quadratic assignment with graph neural networks. *arXiv:1706.07450*, 2017.

[59] D. L. Pimentel-Alarcón, N. Boston, and R. D. Nowak. A characterization of deterministic sampling patterns for low-rank matrix completion. *IEEE Journal of Selected Topics in Signal Processing*, 10(4):623–636, 2016.

[60] C. R. Qi, H. Su, K. Mo, and L. 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*, pages 652–660, 2017.

[61] C. R. Qi, L. Yi, H. Su, and L. J. Guibas. Pointnet++ deep hierarchical feature learning on point sets in a metric space. In *Neural Information Processing Systems*, pages 5105–5114, 2017.

[62] M. Ricci and T. Levi-Civita. Méthodes de calcul différentiel absolu et leurs applications. *Mathematische Annalen*, 54(1):125–201, 1900.

[63] B. Roth. Rigid and flexible frameworks. *The American Mathematical Monthly*, 88(1):6–21, 1981.

[64] C. Roth and A. H. MacDonald. Group Convolutional Neural Networks Improve Quantum State Accuracy. *arXiv:2104.05085*, 2021.

[65] A. Sanchez-Gonzalez, V. Bapst, K. Cranmer, and P. Battaglia. Hamiltonian Graph Networks with ODE Integrators. *arXiv:1909.12790*, 2019.

[66] V. G. Satorras, E. Hoogeboom, and M. Welling. E(n) equivariant graph neural networks. *arXiv:2102.09844*, 2021.

[67] F. Scarselli, M. Gori, A. C. Tsoi, M. Hagenbuchner, and G. Monfardini. The graph neural network model. *IEEE Transactions on Neural Networks*, 20(1):61–80, 2008.

[68] K. T. Schütt, O. T. Unke, and M. Gastegger. Equivariant message passing for the prediction of tensorial properties and molecular spectra, 2021.

[69] A. Singer and M. Cucuringu. Uniqueness of low-rank matrix completion by rigidity theory. *SIAM Journal on Matrix Analysis and Applications*, 31(4):1621–1641, 2010.

[70] B. Tahmasebi and S. Jegelka. Counting substructures with higher-order graph neural networks: Possibility and impossibility results. *arXiv:2012.03174*, 2020.

[71] N. Thomas, T. Smidt, S. Kearnes, L. Yang, L. Li, K. Kohlhoff, and P. Riley. Tensor field networks: Rotation-and translation-equivariant neural networks for 3d point clouds. *arXiv:1802.08219*, 2018.

[72] L. N. Trefethen and D. Bau III. *Numerical linear algebra*, volume 50. Siam, 1997.

[73] P. Veličković, G. Cucurull, A. Casanova, A. Romero, P. Liò, and Y. Bengio. Graph attention networks. In *International Conference on Learning Representations*, 2018.

[74] T. Vieijra, C. Casert, J. Nys, W. De Neve, J. Haegeman, J. Ryckebusch, and F. Verstraete. Restricted boltzmann machines for quantum states with non-abelian or anyonic symmetries. *Physical Review Letters*, 124(9), Mar 2020.

[75] Z. Wang, C. Wang, S. Zhao, S. Du, Y. Xu, B.-L. Gu, and W. Duan. Symmetry-adapted graph neural networks for constructing molecular dynamics force fields, 2021.

[76] H. Weyl. *The Classical Groups*. Princeton University Press, 1946.

[77] K. Xu, W. Hu, J. Leskovec, and S. Jegelka. How powerful are graph neural networks? In *International Conference on Learning Representations*, 2018.

[78] W. Yao, A. S. Bandeira, and S. Villar. Experimental performance of graph neural networks on random instances of max-cut. In *Wavelets and Sparsity XVIII*, volume 11138, page 111380S. International Society for Optics and Photonics, 2019.

[79] D. Yarotsky. Universal approximations of invariant maps by neural networks. *arXiv:1804.10306*, 2018.

[80] M. Zaheer, S. Kottur, S. Ravanbhakhsh, B. Póczos, R. Salakhutdinov, and A. J. Smola. Deep sets. In *Neural Information Processing Systems*, pages 3394–3404, 2017.

[81] Z. Zhang, B.-S. Hua, D. W. Rosen, and S.-K. Yeung. Rotation invariant convolutions for 3d point clouds deep learning. In *2019 International Conference on 3D Vision (3DV)*, pages 204–213. IEEE, 2019.

[82] Y. Zhao, T. Birdal, J. E. Lenssen, E. Menegatti, L. Guibas, and F. Tombari. Quaternion equivariant capsule networks for 3d point clouds. In *European Conference on Computer Vision*, pages 1–19. Springer, 2020.

[83] Y. D. Zhong, B. Dey, and A. Chakraborty. Symplectic ode-net: Learning hamiltonian dynamics with control. In *7th International Conference on Learning Representations (ICLR)*. ICLR, May 2020.

Checklist

1. For all authors...
   (a) Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope? [Yes]
   (b) Did you describe the limitations of your work? [Yes] See Section 6.
   (c) Did you discuss any potential negative societal impacts of your work? [N/A] Our work is purely theoretical at this stage.
   (d) Have you read the ethics review guidelines and ensured that your paper conforms to them? [Yes]
2. If you are including theoretical results...
   (a) Did you state the full set of assumptions of all theoretical results? [Yes]
   (b) Did you include complete proofs of all theoretical results? [Yes] Some of the full proofs are in the appendix, due to space constraints.
3. If you ran experiments...
   (a) Did you include the code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL)? [N/A] We didn't run experiments.
   (b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? [N/A]
   (c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? [N/A]
   (d) Did you include the total amount of compute and the type of resources used (e.g., type of GPUs, internal cluster, or cloud provider)? [N/A]
4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets...
   (a) If your work uses existing assets, did you cite the creators? [N/A]
   (b) Did you mention the license of the assets? [N/A]
   (c) Did you include any new assets either in the supplemental material or as a URL? [N/A]
   (d) Did you discuss whether and how consent was obtained from people whose data you're using/curating? [N/A]
   (e) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? [N/A]
5. If you used crowdsourcing or conducted research with human subjects...
   (a) Did you include the full text of instructions given to participants and screenshots, if applicable? [N/A]
   (b) Did you describe any potential participant risks, with links to Institutional Review Board (IRB) approvals, if applicable? [N/A]
   (c) Did you include the estimated hourly wage paid to participants and the total amount spent on participant compensation? [N/A]