Geometrically Equivariant Graph Neural Networks: A Survey

Jiaqi Han[1,2] , **Yu Rong**[3] , **Tingyang Xu**[3] and **Wenbing Huang**[1✉]

[1]Institute for AI Industry Research (AIR), Tsinghua University
[2]Department of Computer Science and Technology, Tsinghua University
[3]Tencent AI Lab
✉ Corresponding email: hwenbing@126.com

Abstract

Many scientific problems require to process data in the form of geometric graphs. Unlike generic graph data, geometric graphs exhibit symmetries of translations, rotations, and/or reflections. Researchers have leveraged such inductive bias and developed geometrically equivariant Graph Neural Networks (GNNs) to better characterize the geometry and topology of geometric graphs. Despite fruitful achievements, it still lacks a survey to depict how equivariant GNNs are progressed, which in turn hinders the further development of equivariant GNNs. To this end, based on the necessary but concise mathematical preliminaries, we analyze and classify existing methods into three groups regarding how the message passing and aggregation in GNNs are represented. We also summarize the benchmarks as well as the related datasets to facilitate later researches for methodology development and experimental evaluation. The prospect for future potential directions is also provided.

1 Introduction

Many problems particularly in physics and chemistry require to process data in the form of *geometric graphs* [Bronstein *et al.*, 2021]. Distinct from generic graph data, geometric graphs assign each node not only a feature but also a geometric vector. For example, a molecule/protein can be regarded as a geometric graph, where the 3D position coordinates of atoms are the geometric vectors; or in a general multi-body physical system, the 3D states (positions, velocities or spins) are the geometric vectors of the particles. Notably, geometric graphs exhibit symmetries of translations, rotations and/or reflections. This is because the physical law controlling the dynamics of the atoms (or particles) is the same no matter how we translate or rotate the molecular (or a general physical system) from one place to another. When tackling this type of data, it is essential to incorporate the inductive bias of symmetry into the design of the model, which motivates the study of geometrically equivariant Graph Neural Networks (GNNs).

GNNs, originally proposed by [Sperduti and Starita, 1997], have demonstrated their prominence in modeling graph structures under the umbrella of recent advancements of deep learning [Hamilton *et al.*, 2017; Huang *et al.*, 2018; Rong *et al.*, 2020; Sanchez-Gonzalez *et al.*, 2019]. While abundant architectures have been developed, most previous GNNs are not geometrically equivariant[1], making them not suitable for geometric graphs. To achieve geometric equivariance, plenty of works have been proposed to refine the message passing and aggregation mechanism in GNNs. These works include TFN [Thomas *et al.*, 2018] equivariant on the group SE(3)—a set of 3D translation and rotation transformations, LieConv [Finzi *et al.*, 2020] on Lie Group—a set of differential transformations beyond 3D translations and rotations, and EGNN [Satorras *et al.*, 2021b] on all n-dimensional Euclidean transformations including translations, rotations and reflections.

Given the fruitful achievements, however, there is still not a survey paper to depict the whole picture of how equivariant GNNs are progressed. It not only prevents the external researchers from entering this domain rapidly, but could also hinder the extraction of lessons, new ideas and visions from existing papers for the researchers who want to push the boundary further. Hence, we establish this survey to enable a complete introduction of geometrically equivariant GNNs in a systematic way, explaining the challenges they have addressed and prospecting the potential directions for future exploration. We summarize our contributions as follows.

- **Easy reference**. We provide necessary mathematical preliminaries including the definitions of equivariance, group and group representation. We try to make the math part complete but concise to avoid any unnecessary notation that could confuse readers. More importantly, all typical models are introduced by a general framework in consistent notations, such that readers are able to distinguish the difference between different methods more easily.

- **Novel taxonomy**. We put forward a new taxonomy to track the development path of various equivariant GNNs. By focusing on how the message passing and aggregation are represented, we categorize equivariant

---

[1]GNNs are always permutation equivariant but not inherently geometrically equivariant. This paper mainly discusses the latter unless otherwise specified.

GNNs into three styles: irreducible representation, regular representation, and scalarization.

- **Rich resources**. Besides the methodology, we investigate the application scope of existing researches. We depict an entire list of benchmarks according to the data types they use and the tasks they target on. This can be exploited as a hand-on guidance for model development, experimental evaluation and comparison.

- **Future prospects**. Provided the analyses on current works and frontier progress, we discuss the potential future directions in both theoretical and practical perspectives. In particular, we prospect the four points: theoretical completeness, scalability, hierarchy, and more real-world applications and datasets.

2 Backgrounds

In this section, we introduce two key factors, namely Graph Neural Networks (GNNs) and equivariance, as the preliminaries of discussing geometrically equivariant GNNs.

2.1 Message-passing GNNs

GNNs have been widely adopted for handling relational data. Consider a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where $\mathcal{V}$ and $\mathcal{E}$ are the set of nodes and edges, respectively. Each node is assigned a node feature, denoted as h_i for node i. We can also optionally have an edge feature e_{ij} for the edge connecting node i and j. In [Gilmer *et al.*, 2017], a seminal message-passing scheme has been refined to unify the dominated GNNs into a general architecture. It iteratively conducts message computation and neighborhood aggregation for each node (or edge). In general, we have

$$m_{ij} = \psi_m\left(h_i, h_j, e_{ij}\right), \tag{1}$$

$$h_i' = \psi_h\left(\{m_{ij}\}_{j \in \mathcal{N}(i)}, h_i\right), \tag{2}$$

where $\mathcal{N}(i)$ is the set of neighbors around node i (without self-loop by default), and ψ_m, ψ_h are parametric functions.

An intriguing property of the update depicted in Eq. (1-2) is that it satisfies permutation equivariance, as long as ψ_h is permutation equivariant. Modern GNNs are indeed designed to meet this constraint [Keriven and Peyré, 2019; Azizian and marc lelarge, 2021]. Several works employ GNNs on complex systems typically with geometric inputs, *e.g.*, the 3D coordinates of particles. DPI-Net [Li *et al.*, 2018] constructs a dynamic interaction graph based on which a physics simulator is learned for particle dynamics. HRN [Mrowca *et al.*, 2018] similarly builds up an interaction graph for complex objects yet in a hierarchical manner for accurate dynamics prediction. Instead of manually specifying the graph for modeling interactions, NRI [Kipf *et al.*, 2018] automatically infers the latent interaction graph and achieves promising performance in modeling the dynamics of multi-body particle systems as well as human motion capture. These works have extensively demonstrated the advantages of GNNs in modeling the dynamics of real-world geometric systems, owing to their capability of reducing the combinatorial complexity of input orders due to permutation equivariance.

The work by [Townshend *et al.*, 2021b] applies 3D GNN to predict molecular property.

Nevertheless, in this paper we focus on the geometric perspective, and the equivariance we discuss is restrained to the Euclidean space instead of the permutation group. For the above methods, it is still left to discover how the geometric symmetries in 3D space could play a role. We will explain what geometric equivariance is and why we need it in the following subsection.

2.2 Equivariance

Let $\mathcal{X}$ and $\mathcal{Y}$ be the input and output vector spaces, respectively, both of which are endowed with a set of transformations G: $G \times \mathcal{X} \rightarrow \mathcal{X}$ and $G \times \mathcal{Y} \rightarrow \mathcal{Y}$. The function $\phi : \mathcal{X} \rightarrow \mathcal{Y}$ is called equivariant with respect to G if when we apply any transformation to the input, the output also changes via the same transformation or under a certain predictable behavior. In form, we have:

Definition 1 (Equivariance). *The function $\phi : \mathcal{X} \mapsto \mathcal{Y}$ is G-equivariant if it commutes with any transformation in G,*

$$\phi(\rho_{\mathcal{X}}(g)x) = \rho_{\mathcal{Y}}(g)\phi(x), \forall g \in G, \tag{3}$$

where $\rho_{\mathcal{X}}$ and $\rho_{\mathcal{Y}}$ are the group representations in the input and output space, respectively. Specifically, ϕ is called invariant if $\rho_{\mathcal{Y}}$ is the identity.

Definition 2 (Group). *A group G is a set of transformations with a binary operation "·" satisfying these properties: "·" is closed under associative composition, there exists an identity element, and each element G must have an inverse.*

Given the definition of groups, we provide some examples here (more details are referred to [Esteves, 2020]):

- O(n) is an n-dimensional orthogonal group that consists of rotations and reflections.

- SO(n) is a special orthogonal group that only consists of rotations.

- E(n) is an n-dimensional Euclidean group that consists of rotations, reflections, and translations.

- SE(n) is a special Euclidean group that consists of rotations and translations.

- Lie Group is a group whose elements form a differentiable manifold. Actually, all the groups above are specific examples of Lie Group.

Group representation A representation of a group is an invertible linear map $\rho(g) : G \mapsto \mathcal{V}$ that takes as input the group element $g \in G$ and acts on a vector space $\mathcal{V}$, while at the same time it is linear: $\rho(g)\rho(h) = \rho(g \cdot h), \forall g, h \in G$. For instance, a matrix representation for O(n) is the orthogonal matrix $\boldsymbol{O} \in \mathbb{R}^n$ subject to $\boldsymbol{O}^\top \boldsymbol{O} = \boldsymbol{I}$. The instantiation of Eq. (3) on O(3) becomes $\phi(\boldsymbol{O}x) = \boldsymbol{O}\phi(x)$ if the input and output spaces share the same representation. For translation equivariance, we have $\phi(x - \boldsymbol{t}) = \phi(x) - \boldsymbol{t}$ with $t \in \mathbb{R}^n$.

Equivariance (*a.k.a.* covariance) is initially applied to preserve symmetry in physics [Anderson *et al.*, 2019]. The endeavor of injecting Euclidean equivariance into modern deep learning frameworks originates from [Cohen and Welling, 2016a], where the convolution operation in CNNs has been
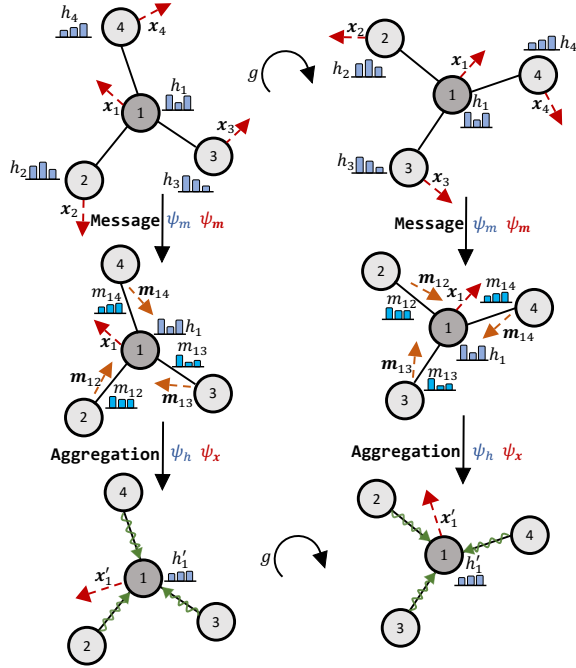
Figure 1: An illustration of the geometrically equivariant message-passing in the case of rotation. Both scalar and directional messages are produced and then aggregated, resulting in an equivariant update.

generalized to discrete subgroups of rotations and reflections. The work by [Cohen and Welling, 2016b] further introduces steerability into the convolutional kernel, generalizing the feature maps from regular representations in [Cohen and Welling, 2016a] to irreducible and quotient representations. In what follows, we will survey how to integrate equivariance into the message passing in current GNNs.

3 Geometrically Equivariant GNNs

This section first describes our formulation of geometric graphs and subsequently summarizes the geometrically equivariant GNNs that promisingly tackle this type of data.

3.1 ==Geometric Graphs==

In many applications, the graphs we tackle contain not only the topological connections and node features (as illustrated in § 2.1) but also certain geometric information. Taking a molecule as an example, each atom is assigned a scalar feature $h_i \in \mathbb{R}^H$ (*e.g.*, charge, mass, etc) as well as a geometric vector $\boldsymbol{x}_i \in \mathbb{R}^3$ (*e.g.*, position or velocity coordinates). When processing such kind of data via GNNs, we can inject the inductive bias of equivariance into the graph model. For example, when predicting the energy of a molecule, we need the output to be invariant to any rotation of the geometrical vectors; when predicting the molecular dynamics, however, we require the output to be equivariant to the input of each atom's position. For better discrimination, we denote the geometric vector in bold and the non-geometric value via a plain symbol throughout this paper.

In general, Eq. (1-2) are renewed as:

$$m_{ij} = \psi_m \left(\boldsymbol{x}_i, \boldsymbol{x}_j, h_i, h_j, e_{ij} \right), \quad (4)$$

$$\boldsymbol{m}_{ij} = \psi_{\boldsymbol{m}} \left(\boldsymbol{x}_i, \boldsymbol{x}_j, h_i, h_j, e_{ij} \right), \quad (5)$$

$$h_i' = \psi_h \left(\{ m_{ij} \}_{j \in \mathcal{N}(i)}, h_i \right), \quad (6)$$

$$\boldsymbol{x}_i' = \psi_{\boldsymbol{x}} \left(\{ \boldsymbol{m}_{ij} \}_{j \in \mathcal{N}(i)}, \boldsymbol{x}_i \right), \quad (7)$$

where, m_{ij} and $\boldsymbol{m}_{ij}$ are separately the scalar and directional outputs by the message passing over the edge (i, j), ψ_h and $\psi_{\boldsymbol{x}}$ are the message aggregation functions for the scalar feature and geometric vector, respectively. Besides, ψ_m is G-invariant *w.r.t.* the input $(\boldsymbol{x}_i, \boldsymbol{x}_j)$, $\psi_{\boldsymbol{m}}$ is G-equivariant *w.r.t.* $(\boldsymbol{x}_i, \boldsymbol{x}_j)$, and $\psi_{\boldsymbol{x}}$ is G-equivariant *w.r.t.* $\{ \boldsymbol{m}_{ij} \}_{j \in \mathcal{N}(i)}$ and $\boldsymbol{x}_i$. We elaborate an illustration of the above equivariant message-passing protocol in Figure 1.

Various equivariant GNNs have been proposed, and they are generally different specifications of Eq.(4-7) on different groups. We summarize them in Table 1. In terms of how the message is represented, we divide current methods into three classes: irreducible representation, regular representation, and scalarization. In most cases, translation equivariance is trivially satisfied since the relative position $\boldsymbol{x}_i - \boldsymbol{x}_j$ is translation invariant, and the additive residual update in $\boldsymbol{x}_i'$ permits translation equivariance. For this reason, our following discussions are mainly concerned with rotations and other transformations beyond translations.

Table 1: A summary of the equivariant GNNs. "$*$" denotes methods augmented by the attention mechanism.

	Method	Group	Property
TFN [Thomas *et al.*, 2018]	Irrep.	SE(3)	Equiv.
Cormorant [Anderson *et al.*, 2019]	Irrep.	SO(3)	Equiv.
SE(3)-Tr. [Fuchs *et al.*, 2020]*	Irrep.	SE(3)	Equiv.
SEGNN [Brandstetter *et al.*, 2022]	Irrep.	E(3)	Equiv.
LieConv [Finzi *et al.*, 2020]	Regul.	Lie	Invar.
LieTr. [Hutchinson *et al.*, 2021]*	Regul.	Lie	Invar.
SchNet [Schütt *et al.*, 2018]	Scala.	E(3)	Invar.
DimeNet [Klicpera *et al.*, 2020]	Scala.	E(3)	Invar.
Radial Field [Köhler *et al.*, 2020]	Scala.	E(n)	Equiv.
GVP-GNN [Jing *et al.*, 2021]	Scala.	E(3)	Equiv.
EGNN [Satorras *et al.*, 2021b]	Scala.	E(n)	Equiv.
GMN [Huang *et al.*, 2022]	Scala.	E(n)	Equiv.
PaiNN [Schütt *et al.*, 2021]	Scala.	SO(3)	Equiv.
ET [Thölke and Fabritiis, 2022]*	Scala.	O(3)	Equiv.
GemNet [Klicpera *et al.*, 2021]	Scala.	SE(3)	Equiv.

3.2 Irreducible Representation

==According to representation theory [Esteves, 2020], the linear representations of a compact group can be expressed by the direct sum of *irreducible representations* (or *irreps* for short) up to a similarity transformation.== Specifically for the group SO(3), the irreps are $(2l + 1) \times (2l + 1)$ Wigner-D matrices $\boldsymbol{D}^l$ with positive integer $l = 0, 1, \cdots$. For every SO(3) representation, we have

$$\rho(g) = \boldsymbol{Q}^\top \left(\bigoplus_l \boldsymbol{D}^l(g) \right) \boldsymbol{Q}, \quad (8)$$

where $\boldsymbol{D}^l$ is the Wigner-D matrix, $\boldsymbol{Q}$ is an orthogonal matrix accounting for the change of basis, and $\bigoplus$ is the direct sum or concatenation of matrices along the diagonal. By this means, the vector space is partitioned into l subspaces, each transformed by $\boldsymbol{D}^l$, and the vector lies in the l-th subspace is dubbed type-l vector. For instance, in our case the scalar h_i is a type-0 vector with H channels, and $\boldsymbol{x}_i$ is a type-1 vector. These vectors interact via the tensor product "$\otimes$", and the tensor product of Wigner-D matrices yields the Clebsch-Gordan (CG) coefficients $\boldsymbol{C}^{lk} \in \mathbb{R}^{(2l+1)(2k+1)\times(2l+1)(2k+1)}$ via the CG decomposition:

$$\boldsymbol{D}^k(g) \otimes \boldsymbol{D}^l(g) = (\boldsymbol{C}^{lk})^\top \left(\bigoplus_{J=|k-l|}^{k+l} \boldsymbol{D}^J(g) \right) \boldsymbol{C}^{lk}. \quad (9)$$

The last recipe for building an equivariant message passing layer is the spherical harmonics Y_{Jm} that serves as an equivariant basis for SO(3). With the above building blocks, [Thomas *et al.*, 2018] proposes the TFN layer satisfying SE(3)-equivariance,

$$\boldsymbol{m}_{ij}^l = \sum_{k\geq 0} \boldsymbol{W}^{lk}(\boldsymbol{x}_i - \boldsymbol{x}_j)\boldsymbol{x}_j^k, \quad (10)$$

$$\boldsymbol{x}_i'^l = w_{ll}\boldsymbol{x}_i^l + \sum_{j\in\mathcal{N}(i)} \boldsymbol{m}_{ij}^l, \quad (11)$$

where $\boldsymbol{x}_i^l \in \mathbb{R}^{2l+1}$ denotes the geometric vector of degree l for node i, $\boldsymbol{x}_i \in \mathbb{R}^3$ is the node coordinate, w_{ll} is the self-interaction weight, and the filter $\boldsymbol{W}^{lk}(\boldsymbol{x}) \in \mathbb{R}^{(2l+1)\times(2k+1)}$ is rotation-steerable, implying $\boldsymbol{W}^{lk}(\boldsymbol{D}^1(r)\boldsymbol{x}) = \boldsymbol{D}^l(r)\boldsymbol{W}^{lk}(\boldsymbol{x})(\boldsymbol{D}^k(r))^{-1}$ for an arbitrary rotation $r \in$ SO(3). To be specific, $\boldsymbol{W}^{lk}(\boldsymbol{x}) = \sum_{J=|l-k|}^{l+k} \varphi_J^{lk}(\|\boldsymbol{x}\|) \sum_{m=-J}^{J} Y_{Jm}(\boldsymbol{x}/\|\boldsymbol{x}\|)\boldsymbol{C}_{Jm}^{lk}$, a combination of learnable radius functions $\varphi_J^{lk} \in \mathbb{R}$, spherical harmonics $Y_{Jm} \in \mathbb{R}$, and CG coefficients $\boldsymbol{C}_{Jm}^{lk} \in \mathbb{R}^{(2l+1)\times(2k+1)}$.

TFN regards $\boldsymbol{x}_i^l$ as a signal function of $\boldsymbol{x}_i$, and the computations above only update $\boldsymbol{x}_i^l$ while keeping $\boldsymbol{x}_i$ fixed over all layers. It is easy to check the equivariance of $\boldsymbol{x}_i^l$ with the help of the steerability of $\boldsymbol{W}^{lk}(\boldsymbol{x})$. TFN respectively realizes the general form in Eq. (4)&(6) by setting $l = 0$, and Eq. (5)&(7) by setting $l = 1$ if $\boldsymbol{x}_i$ is considered as a function of itself.

Several extensions have been made since TFN. [Fuchs *et al.*, 2020] further incorporate the attention mechanism to Eq. 10 by multiplying m_{ij}^l with an SE(3)-invariant attentive term. [Dym and Maron, 2021] theoretically reveals that both TFN [Thomas *et al.*, 2018] and SE(3)-Transformer [Fuchs *et al.*, 2020] are universal approximators for SE(3)-equivariant functions. Cormorant [Anderson *et al.*, 2019] similarly leverages the irreps but with Clebsch-Gordan non-linearities. Recent work by [Brandstetter *et al.*, 2022] generalizes to steerable vectors and builds a competitive E(3)-equivariant GNN. However, the computational overhead of these methods is generally expensive, restricting the usage of vectors from higher degrees. Moreover, the irreps are only widely known for groups like SO(3), while not implementation-friendly to many other groups, *e.g.*, E(n).

3.3 Regular Representation

Another line of work directly seeks the solution to obtaining equivariance from group convolution [Cohen and Welling, 2016a] using regular representation, which defines convolution filters as functions on groups. Nevertheless, the integral in group convolution becomes intractable when dealing with continuous and smooth groups, and one feasible tool to leverage in this case is Lie algebra. To this end, [Finzi *et al.*, 2020] proposes LieConv that figures out group convolution via lifting (mapping the input in $\mathcal{X}$ to a group element $g \in G$) and discretization of the convolution integral via the PointConv trick. Particularly, with our consistent notations, LieConv is formulated as follows.

$$m_{ij} = \varphi\left(\log(u_i^{-1}u_j)\right) h_j, \quad (12)$$

$$h_i' = \frac{1}{|\mathcal{N}(i)| + 1} \left(h_i + \sum_{j\in\mathcal{N}(i)} m_{ij} \right), \quad (13)$$

where $u_i \in G$ is a lift of $\boldsymbol{x}_i$, the logarithm $\log$ maps each group member onto the Lie Algebra $\mathfrak{g}$ that is a vector space, and φ is a parametric MLP. Besides, Eq. (13) conducts normalization by the division of the number of all nodes, *i.e.* $\mathcal{N}(i) + 1$. It is clear that LieConv only specifies the update of node features h_i while keeping the geometric vectors $\boldsymbol{x}_i$ unchanged. That means LieConv is invariant[2].

Following a similar idea, LieTransformer [Hutchinson *et al.*, 2021] employs the self-attention strategy to dynamically re-weight the convolutional kernel for an increase in model capacity and performance. The regular-representation-based methods enjoy higher flexibility since the equivariance can be obtained on arbitrary Lie groups as well as their discrete subgroups. On the other hand, due to discretization and sampling, they also encounter a trade-off between computational complexity and performance. Another weakness is that it is non-trivial to extend Eq.(12-13) for the propagation of geometric vectors, unless we introduce external Hamiltonian dynamics to renew geometric vectors akin to [Finzi *et al.*, 2020].

3.4 Scalarization

Aside from group representation theory, a number of works adopt a generic way of modeling equivariance through scalarization. Typically, the geometric vectors are firstly transformed into invariant scalars, followed by several MLPs to control the magnitude, finally added up in the original directions to obtain equivariance. This idea has originally been implied in SchNet [Schütt *et al.*, 2018] and DimeNet [Klicpera *et al.*, 2020], but only in the invariant flavor. Radial Field [Köhler *et al.*, 2020] implements the equivariant version, but it only operates on geometric vectors without the consideration of node features. EGNN [Satorras *et*

---

[2]LieConv is originally claimed as equivariant in [Finzi *et al.*, 2020], which is explained in the sense that h_i' is a signal function of $\boldsymbol{x}_i'$ (not a function of $\boldsymbol{x}_i$ in our case).

al., 2021b] further refines the idea via a flexible paradigm:

$$m_{ij} = \varphi_m\left(h_i, h_j, \|\boldsymbol{x}_i - \boldsymbol{x}_j\|^2, e_{ij}\right), \quad (14)$$

$$\boldsymbol{x}_i' = \boldsymbol{x}_i + \sum_{j \neq i} (\boldsymbol{x}_i - \boldsymbol{x}_j)\,\varphi_x(m_{ij}), \quad (15)$$

$$h_i' = \varphi_h(h_i, \sum_{j \in \mathcal{N}(i)} m_{ij}), \quad (16)$$

where, $\|\boldsymbol{x}_i - \boldsymbol{x}_j\|^2$ is the scalarization of the geometric vectors $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$; φ_m, φ_x, and φ_h are arbitrary MLPs. By setting $\boldsymbol{m}_{ij} = (\boldsymbol{x}_i - \boldsymbol{x}_j)\,\varphi_x(m_{ij})$, EGNN implement Eq. (4-7) by jointly propagating the node features h_i and geometric vectors $\boldsymbol{x}_i$ in an equivariant yet straightforward way. The essence lies in constructing the invariant message m_{ij}, and then transforming back to the equivariant output along the radial directions, analogous to the way we compute the resultant Coulomb (or gravitational) force exerted on pairs of charged particles. Notice that Eq. (15), different from Eq. (16), aggregates messages from all nodes in addition to the neighbors around i, which reflects the law that the dynamics of each node is influenced by all others.

Beyond EGNN, GMN [Huang *et al.*, 2022] extends the aggregation to deal with multiple geometric vectors (*e.g.* positions and velocities plus forces), denoted by $\boldsymbol{Z} \in \mathbb{R}^{3 \times m}$, showing that $\boldsymbol{Z}' = \boldsymbol{Z}\varphi(\boldsymbol{Z}^\top \boldsymbol{Z})$ is indeed a universal form in this case. By this means, the interaction in Eq. (14-15) could span across the space with $\boldsymbol{Z}$ as the basis instead of the radial direction alone. This is particularly desirable in constrained systems, where non-radial message vectors (*e.g.*, angular momentum, torque) might be produced by the interaction between and within objects. Despite the simplicity, the theoretical efficacy of these methods is endorsed by [Villar *et al.*, 2021], stating that the scalarization (inner product) techniques are universal to achieve equivariance. GemNet [Klicpera *et al.*, 2021] leverages this universality to inject rich geometric information, for example, the dihedral angle, in the message passing, putting a step forward from the framework of DimeNet [Klicpera *et al.*, 2020].

One interpretation of Eq. (15) is that the multiplication of invariant scalar and equivariant vector still produces an equivariant vector. In this principle, another line of work conducts equivariant message passing in various forms. PaiNN [Schütt *et al.*, 2021] and the attentive Equivariant Transformer [Thölke and Fabritiis, 2022] augment the invariant SchNet into equivariant flavor by projecting the interatomic distances via radial basis functions and iteratively updating the vectors along with the scalar features. GVP-GNN [Jing *et al.*, 2021] leverages a similar idea but comes with a stronger theoretical guarantee of universality.

4 Application

Equivariant GNNs have wide applications on various types of real-world geometric data, ranging from physical systems to chemical substances. In this section, we introduce the application in scenarios involving physical systems, molecular data, and point clouds, respectively. An overview of the resources of datasets is displayed in Table 2.

4.1 Physical Dynamics Simulation

Modeling the dynamics of complex physical systems has long been a challenging topic, and neural networks have been applied to infer the interaction and dynamics. Within the physical system are objects like charged particles that interact through forces abiding by physical laws. [Kipf *et al.*, 2018] introduces the N-body simulation, in which multiple charged particles are driven by Coulomb force. The task here is to predict the dynamics of the particles given the initial conditions, including positions, velocities, and charges. Such a task is E(3)-equivariant, since the dynamics of particles translate, rotate, and reflect together with the entire system. Both [Fuchs *et al.*, 2020] and [Satorras *et al.*, 2021b] showcase the strong performance of equivariant GNNs on this task. [Huang *et al.*, 2022] equips the system with connected rigid bodies like sticks and hinges, contributing to a more challenging scenario dubbed Constrained N-body. [Brandstetter *et al.*, 2022] creates a counterpart of the system by instead using gravitational force and increasing the number of particles significantly. [Kipf *et al.*, 2018] and [Huang *et al.*, 2022] also adopt human motion capture data [CMU, 2003] which involves human motion trajectories towards different directions.

4.2 Molecules

Another essential application scenario lies in molecular data, where atoms in molecules are affected by complicated chemical interactions. For molecular data, the scalar node feature h_i is typically the atom number, and the connectivity between nodes is either provided by chemical bonds or taken by a cut-off based on a distance threshold. We review Equivariant GNNs on molecules, including prediction and generation.

Prediction QM9 [Ramakrishnan *et al.*, 2014] is a widely adopted dataset comprising 12 quantum properties as invariant tasks. MD17 [Chmiela *et al.*, 2017] includes molecular dynamics trajectory for 8 small molecules alongside the energy and interaction force as labels. Similar to MD17, ISO17 [Schütt *et al.*, 2018] contains short MD trajectories of 129 isomers. The Open Catalyst 2020 (OC20) dataset considers the binding process of catalyst and absorbate, and the tasks here include predicting the relaxed energy or structure given the initial structures. A large number of works discussed in § 3 have demonstrated the efficacy on these datasets by predicting the objectives accurately while requiring less computational cost than the traditional simulation method via Density Functional Theory (DFT).

Moving forward, AlphaFold2 [Jumper *et al.*, 2021] achieves striking performance in predicting the folding structure of proteins, which initiates endeavors towards large chemical compounds with the help of geometric equivariance. Several works extend the basic frameworks described in § 3 on large-scale molecular systems like proteins. [Eismann *et al.*, 2020] employs TFN [Thomas *et al.*, 2018] on protein data in a hierarchical manner by sampling the key substructures. [Ganea *et al.*, 2022] targets the problem of rigid body protein-protein docking by extending [Satorras *et al.*, 2021b] to a matching network that copes with two geometric graphs simultaneously. ARES [Townshend *et al.*, 2021a] applies a rotationally equivariant network on RNA data, and inter-

Table 2: Datasets for the evaluation of equivariant GNNs.

Dataset	Application	Task	Property
N-body [Kipf *et al.*, 2018]	Physical Simulation	Position & Velocity Prediction	Equivariant
Constrained N-body [Huang *et al.*, 2022]	Physical Simulation	Position Prediction	Equivariant
Motion Capture [CMU, 2003]	Physical Simulation	Position Prediction	Equivariant
QM9 [Ramakrishnan *et al.*, 2014]	Small Molecule	Chemical Property Prediction	Invariant
MD17 [Chmiela *et al.*, 2017]	Small Molecule	Energy & Force Prediction	Invariant
ISO17 [Schütt *et al.*, 2018]	Small Molecule	Energy & Force Prediction	Invariant
OC20 [Zitnick *et al.*, 2020]	Molecule (Catalyst)	Relaxed Energy, *etc.* Prediction	Invariant
GEOM [Axelrod and Gomez-Bombarelli, 2022]	Small Molecule	Generation	Equivariant
Atom3D [Townshend *et al.*, 2021b]	Molecule & RNA & Protein	Binding Affinity, *etc.* Prediction	Invariant
MDAnalysis [Richard J. Gowers *et al.*, 2016]	Protein	Position Prediction	Equivariant
ModelNet40 [Wu *et al.*, 2015]	Point Cloud	Classification	Invariant
ScanObjectNN [Uy *et al.*, 2019]	Point Cloud	Classification	Invariant

estingly, the model exhibits strong generalization capability even trained with only 18 RNA structures. On the data side, MDAnalysis [Richard J. Gowers *et al.*, 2016] serves as a solid dataset for molecular dynamics on proteins. Atom3D [Townshend *et al.*, 2021b] is a comprehensive dataset containing 8 prediction tasks on molecules with geometric information, ranging from small molecules to RNA and proteins.

Generation [Axelrod and Gomez-Bombarelli, 2022] introduce a large-scale unlabeled data split into two subsets GEOM-QM9 and GEOM-Drugs. These datasets contain diverse samples of conformations for a large number of small molecules and have served as the unsupervised training set in the task of molecular conformation generation. ConfGF [Shi *et al.*, 2021] and DGSM [Luo *et al.*, 2021] parameterize the score function in the score-based generative model [Song and Ermon, 2019] with a roto-translation equivariant GNN, leading to a generative model that learns the conditional distribution of the conformations. GeoDiff [Xu *et al.*, 2022] further promotes the generative model to denoising diffusion model [Ho *et al.*, 2020], and similarly adapt the diffusion kernel by a GNN with equivariance guarantee. Besides, Equivariant Flow [Köhler *et al.*, 2020] verifies the feasibility of constructing normalizing flow with an equivariant kernel in the change of density. However, only the coordinates are taken into account without the node feature. [Satorras *et al.*, 2021a] makes use of EGNN (Eq. 14-16) as the kernel and jointly models the evolution of both vector and scalar inputs in the dynamics of continuous normalizing flow [Chen *et al.*, 2018]. These methods allow for unconditional generation, where the conformations are generated from scratch without providing the molecular graph.

4.3 Point Clouds

Point cloud is a representation format of objects that describe the shape by a set of points assigned with coordinates. ModelNet40 [Wu *et al.*, 2015] and ScanObjectNN [Uy *et al.*, 2019] are two well-received datasets for object classification based on the point cloud. Since the connectivity of points is absent in point cloud data, the neighborhood used by equivariant GNNs is commonly defined by $\mathcal{N}(i) = \{j \big| \|\boldsymbol{x}_i - \boldsymbol{x}_j\|_2 < d\}$ where d is the maximum distance. Both TFN and SE(3)-Transformer have exhibited competitive performance on point cloud data. Recently, [Chen *et al.*, 2021]

proposes an SE(3)-equivariant convolution scheme for point cloud by iteratively conducting group convolution and point convolution. These approaches demonstrate an advantage of equivariant GNNs compared to traditional 3D CNNs, since they do not require the voxelization of input Euclidean space while still maintaining the desirable equivariance.

5 Future Research Directions

Theoretical completeness Unlike the well established theoretical frameworks for depicting the expressivity and generalization of GNNs, little has been revealed about equivariant GNNs. Although several works [Dym and Maron, 2021; Villar *et al.*, 2021; Jing *et al.*, 2021] have analyzed the universality of some methods, their discussions are merely based on the message-passing function, whereas the property of the entire graph model remains unknown. It will be interesting to see how the existing theoretical framework in building powerful equivariant GNNs converge to this respect.

Scalability As discussed in § 3, the methods leveraging group representation theory consume high computational cost, limiting their scalability towards large and complex systems like proteins. The issue becomes more significant when they are equipped with attention. As a consequence, it is still in demand of speeding up the computation in the equivariant message passing. Possible solutions to improving scalability involve approximation and efficient sampling.

Hierarchical structure Many real-world systems exhibit hierarchical structures. For example, organic molecules are comprised of multiple functional groups, and proteins consist of amino acids. By making use of these hierarchical structures, it is possible for equivariant GNNs to model the systems in multiple granularities. Compared with the flat message passing scheme adopted by existing works, we expect the hierarchical mechanism to enhance the efficiency as well as generalization of the model.

More real-world applications and datasets Many equivariant GNNs have only been evaluated on systems with limited scale and complexity, *e.g.*, N-body system and the small molecular datasets. More challenging tasks involve strengthening the systems with more objects, more diverse constraints, and more complicated interactions. Works on proteins are promising attempts, but they are mostly limited in

diversity due to difficulty in the collection, and a comprehensive evaluation of existing methods has not been established.

6 Conclusion

In this paper, we conduct a survey of geometrically equivariant GNNs. We show that existing works lie in our refined geometric message-passing paradigm as specifications via irreducible representation, regular representation, or scalarization. We discuss their broad application prospect on various tasks, including simulating physical dynamics, modeling molecules or proteins, and handling point clouds. With the promising future directions, we expect to see equivariant GNNs as powerful tools for coping with the tasks in scientific domains.

References

[Anderson *et al.*, 2019] Brandon Anderson, Truong Son Hy, and Risi Kondor. Cormorant: Covariant molecular neural networks. In *NeurIPS*, volume 32, 2019.

[Axelrod and Gomez-Bombarelli, 2022] Simon Axelrod and Rafael Gomez-Bombarelli. Geom: Energy-annotated molecular conformations for property prediction and molecular generation, 2022.

[Azizian and marc lelarge, 2021] Waiss Azizian and marc lelarge. Expressive power of invariant and equivariant graph neural networks. In *ICLR*, 2021.

[Brandstetter *et al.*, 2022] Johannes Brandstetter, Rob Hesselink, Elise van der Pol, Erik J Bekkers, and Max Welling. Geometric and physical quantities improve e(3) equivariant message passing. In *ICLR*, 2022.

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

[Chen *et al.*, 2018] Changyou Chen, Chunyuan Li, Liqun Chen, Wenlin Wang, Yunchen Pu, and Lawrence Carin Duke. Continuous-time flows for efficient inference and density estimation. In *ICML*, 2018.

[Chen *et al.*, 2021] Haiwei Chen, Shichen Liu, Weikai Chen, Hao Li, and Randall Hill. Equivariant point network for 3d point cloud analysis. In *CVPR*, pages 14514–14523, 2021.

[Chmiela *et al.*, 2017] Stefan Chmiela, Alexandre Tkatchenko, Huziel E Sauceda, Igor Poltavsky, Kristof T Schütt, and Klaus-Robert Müller. Machine learning of accurate energy-conserving molecular force fields. *Science advances*, 3(5):e1603015, 2017.

[CMU, 2003] CMU. Carnegie-mellon motion capture database. 2003.

[Cohen and Welling, 2016a] Taco S. Cohen and Max Welling. Group equivariant convolutional networks. In *ICML*, 2016.

[Cohen and Welling, 2016b] Taco S. Cohen and Max Welling. Steerable cnns, 2016.

[Dym and Maron, 2021] Nadav Dym and Haggai Maron. On the universality of rotation equivariant point cloud networks. In *ICLR*, 2021.

[Eismann *et al.*, 2020] Stephan Eismann, Raphael J.L. Townshend, Nathaniel Thomas, Milind Jagota, Bowen Jing, and Ron O. Dror. Hierarchical, rotation-equivariant neural networks to select structural models of protein complexes. *Proteins: Structure, Function, and Bioinformatics*, 89(5):493–501, Dec 2020.

[Esteves, 2020] Carlos Esteves. Theoretical aspects of group equivariant neural networks, 2020.

[Finzi *et al.*, 2020] Marc Finzi, Samuel Stanton, Pavel Izmailov, and Andrew Gordon Wilson. Generalizing convolutional neural networks for equivariance to lie groups on arbitrary continuous data. In *ICML*, 2020.

[Fuchs *et al.*, 2020] Fabian Fuchs, Daniel Worrall, Volker Fischer, and Max Welling. Se(3)-transformers: 3d roto-translation equivariant attention networks. In *NeurIPS*, volume 33, 2020.

[Ganea *et al.*, 2022] Octavian-Eugen Ganea, Xinyuan Huang, Charlotte Bunne, Yatao Bian, Regina Barzilay, Tommi S. Jaakkola, and Andreas Krause. Independent SE(3)-equivariant models for end-to-end rigid protein docking. In *ICLR*, 2022.

[Gilmer *et al.*, 2017] Justin Gilmer, Samuel S. Schoenholz, Patrick F. Riley, Oriol Vinyals, and George E. Dahl. Neural message passing for quantum chemistry. In *ICML*, 2017.

[Hamilton *et al.*, 2017] Will Hamilton, Zhitao Ying, and Jure Leskovec. Inductive representation learning on large graphs. In *NeurIPS*, volume 30, 2017.

[Ho *et al.*, 2020] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In *NeurIPS*, volume 33, 2020.

[Huang *et al.*, 2018] Wenbing Huang, Tong Zhang, Yu Rong, and Junzhou Huang. Adaptive sampling towards fast graph representation learning. In *NeurIPS*, volume 31, 2018.

[Huang *et al.*, 2022] Wenbing Huang, Jiaqi Han, Yu Rong, Tingyang Xu, Fuchun Sun, and Junzhou Huang. Constrained graph mechanics networks. In *ICLR*, 2022.

[Hutchinson *et al.*, 2021] Michael J Hutchinson, Charline Le Lan, Sheheryar Zaidi, Emilien Dupont, Yee Whye Teh, and Hyunjik Kim. Lietransformer: equivariant self-attention for lie groups. In *ICML*, 2021.

[Hyd, 2009] The Schrödinger Wave Equation for the Hydrogen Atom. https://faculty.washington.edu/seattle/physics227/reading/reading-26-27.pdf, 2009.

[Jing *et al.*, 2021] Bowen Jing, Stephan Eismann, Patricia Suriana, Raphael John Lamarre Townshend, and Ron Dror.

Learning from protein structure with geometric vector perceptrons. In *ICLR*, 2021.

[Jumper *et al.*, 2021] John Jumper, Richard Evans, Alexander Pritzel, Tim Green, Michael Figurnov, Olaf Ronneberger, Kathryn Tunyasuvunakool, Russ Bates, Augustin Žídek, Anna Potapenko, Alex Bridgland, Clemens Meyer, Simon A A Kohl, Andrew J Ballard, Andrew Cowie, Bernardino Romera-Paredes, Stanislav Nikolov, Rishub Jain, Jonas Adler, Trevor Back, Stig Petersen, David Reiman, Ellen Clancy, Michal Zielinski, Martin Steinegger, Michalina Pacholska, Tamas Berghammer, Sebastian Bodenstein, David Silver, Oriol Vinyals, Andrew W Senior, Koray Kavukcuoglu, Pushmeet Kohli, and Demis Hassabis. Highly accurate protein structure prediction with AlphaFold. *Nature*, 596(7873):583–589, 2021.

[Keriven and Peyré, 2019] Nicolas Keriven and Gabriel Peyré. Universal invariant and equivariant graph neural networks. In *NeurIPS*, volume 32, 2019.

[Kipf *et al.*, 2018] Thomas Kipf, Ethan Fetaya, Kuan-Chieh Wang, Max Welling, and Richard Zemel. Neural relational inference for interacting systems. In *ICML*, 2018.

[Klicpera *et al.*, 2020] Johannes Klicpera, Janek Groß, and Stephan Günnemann. Directional message passing for molecular graphs. In *ICLR*, 2020.

[Klicpera *et al.*, 2021] Johannes Klicpera, Florian Becker, and Stephan Günnemann. Gemnet: Universal directional graph neural networks for molecules. In *NeurIPS*, 2021.

[Köhler *et al.*, 2020] Jonas Köhler, Leon Klein, and Frank Noe. Equivariant flows: Exact likelihood generative learning for symmetric densities. In *ICML*, 2020.

[Li *et al.*, 2018] Yunzhu Li, Jiajun Wu, Russ Tedrake, Joshua B Tenenbaum, and Antonio Torralba. Learning particle dynamics for manipulating rigid bodies, deformable objects, and fluids. *arXiv preprint arXiv:1810.01566*, 2018.

[Luo *et al.*, 2021] Shitong Luo, Chence Shi, Minkai Xu, and Jian Tang. Predicting molecular conformation via dynamic graph score matching. In *NeurIPS*, 2021.

[Mrowca *et al.*, 2018] Damian Mrowca, Chengxu Zhuang, Elias Wang, Nick Haber, Li Fei-Fei, Joshua B. Tenenbaum, and Daniel L. K. Yamins. Flexible neural representation for physics prediction. In *NeurIPS*, 2018.

[Ramakrishnan *et al.*, 2014] Raghunathan Ramakrishnan, Pavlo O Dral, Matthias Rupp, and O Anatole von Lilienfeld. Quantum chemistry structures and properties of 134 kilo molecules. *Scientific Data*, 1, 2014.

[Richard J. Gowers *et al.*, 2016] Richard J. Gowers, Max Linke, Jonathan Barnoud, Tyler J. E. Reddy, Manuel N. Melo, Sean L. Seyler, Jan Domański, David L. Dotson, Sébastien Buchoux, Ian M. Kenney, and Oliver Beckstein. MDAnalysis: A Python Package for the Rapid Analysis of Molecular Dynamics Simulations. In *Proceedings of the 15th Python in Science Conference*, pages 98 – 105, 2016.

[Rong *et al.*, 2020] Yu Rong, Wenbing Huang, Tingyang Xu, and Junzhou Huang. Dropedge: Towards deep graph convolutional networks on node classification. In *ICLR*, 2020.

[Sanchez-Gonzalez *et al.*, 2019] Alvaro Sanchez-Gonzalez, Victor Bapst, Kyle Cranmer, and Peter Battaglia. Hamiltonian graph networks with ode integrators, 2019.

[Satorras *et al.*, 2021a] Victor Garcia Satorras, Emiel Hoogeboom, Fabian Bernd Fuchs, Ingmar Posner, and Max Welling. E(n) equivariant normalizing flows. In *NeurIPS*, 2021.

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

[Schütt *et al.*, 2018] Kristof T Schütt, Huziel E Sauceda, P-J Kindermans, Alexandre Tkatchenko, and K-R Müller. Schnet–a deep learning architecture for molecules and materials. *The Journal of Chemical Physics*, 148(24):241722, 2018.

[Schütt *et al.*, 2021] Kristof Schütt, Oliver Unke, and Michael Gastegger. Equivariant message passing for the prediction of tensorial properties and molecular spectra. In *ICML*, 2021.

[Shi *et al.*, 2021] Chence Shi, Shitong Luo, Minkai Xu, and Jian Tang. Learning gradient fields for molecular conformation generation. In *ICML*, 2021.

[Song and Ermon, 2019] Yang Song and Stefano Ermon. Generative modeling by estimating gradients of the data distribution. In *NeurIPS*, volume 32, 2019.

[Sperduti and Starita, 1997] Alessandro Sperduti and Antonina Starita. Supervised neural networks for the classification of structures. *IEEE Transactions on Neural Networks*, 8(3):714–735, 1997.

[Thölke and Fabritiis, 2022] Philipp Thölke and Gianni De Fabritiis. Equivariant transformers for neural network based molecular potentials. In *ICLR*, 2022.

[Thomas *et al.*, 2018] 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.

[Townshend *et al.*, 2021a] Raphael J. L. Townshend, Stephan Eismann, Andrew M. Watkins, Ramya Rangan, Maria Karelina, Rhiju Das, and Ron O. Dror. Geometric deep learning of rna structure. *Science*, 373(6558):1047–1051, 2021.

[Townshend *et al.*, 2021b] Raphael John Lamarre Townshend, Martin Vögele, Patricia Adriana Suriana, Alexander Derry, Alexander Powers, Yianni Laloudakis, Sidhika Balachandar, Bowen Jing, Brandon M. Anderson, Stephan Eismann, Risi Kondor, Russ Altman, and Ron O. Dror. ATOM3d: Tasks on molecules in three dimensions. In *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track*, 2021.

[Uy *et al.*, 2019] Mikaela Angelina Uy, Quang-Hieu Pham, Binh-Son Hua, Thanh Nguyen, and Sai-Kit Yeung. Revisiting point cloud classification: A new benchmark dataset and classification model on real-world data. In *ICCV*, October 2019.

[Villar *et al.*, 2021] Soledad Villar, David W Hogg, Kate Storey-Fisher, Weichi Yao, and Ben Blum-Smith. Scalars are universal: Equivariant machine learning, structured like classical physics. In *NeurIPS*, 2021.

[Wu *et al.*, 2015] Zhirong Wu, Shuran Song, Aditya Khosla, Fisher Yu, Linguang Zhang, Xiaoou Tang, and Jianxiong Xiao. 3d shapenets: A deep representation for volumetric shapes. In *CVPR*, June 2015.

[Xu *et al.*, 2022] Minkai Xu, Lantao Yu, Yang Song, Chence Shi, Stefano Ermon, and Jian Tang. Geodiff: A geometric diffusion model for molecular conformation generation. In *ICLR*, 2022.

[Zitnick *et al.*, 2020] C. Lawrence Zitnick, Lowik Chanussot, Abhishek Das, Siddharth Goyal, Javier Heras-Domingo, Caleb Ho, Weihua Hu, Thibaut Lavril, Aini Palizhati, Morgane Riviere, Muhammed Shuaibi, Anuroop Sriram, Kevin Tran, Brandon Wood, Junwoong Yoon, Devi Parikh, and Zachary Ulissi. An introduction to electrocatalyst design using machine learning for renewable energy storage, 2020.