Universally Invariant Learning in Equivariant GNNs

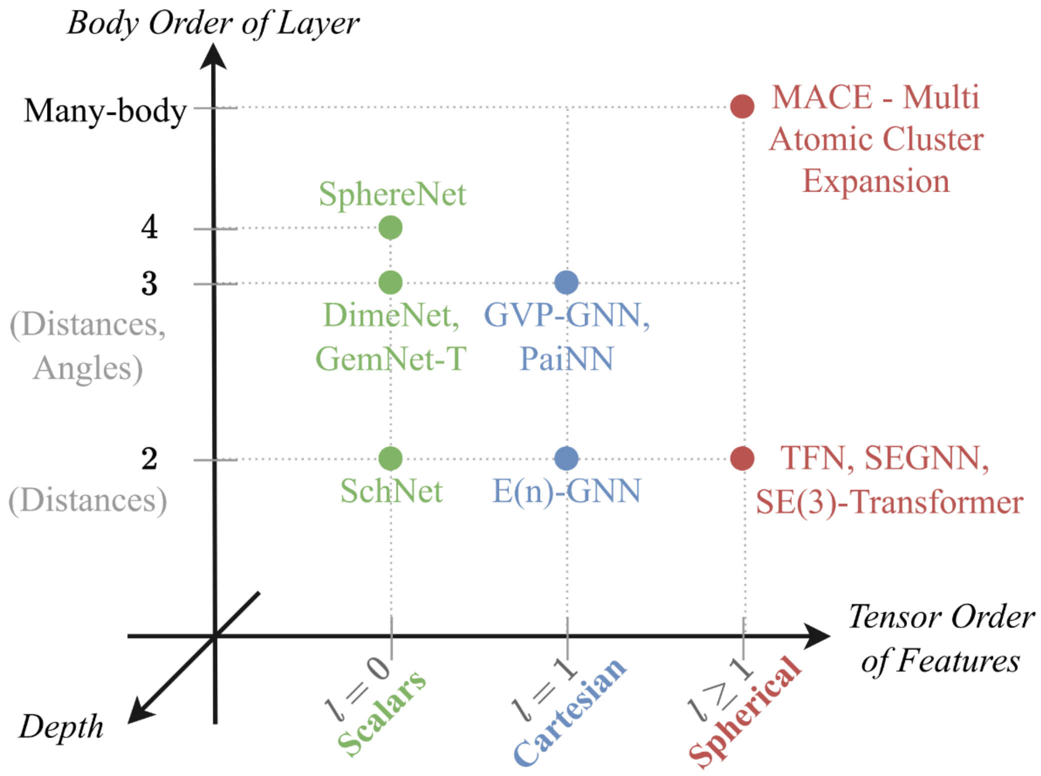
Jiacheng Cen, Anyi Li, Ning Lin, Tingyang Xu, Yu Rong
Deli Zhao, Zihe Wang, Wenbing Huang

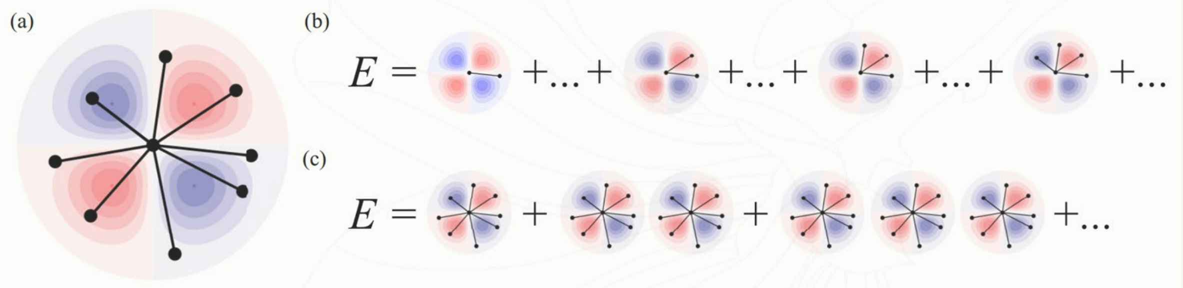
2025.10.01

Major equivariant GNNs Paradigm

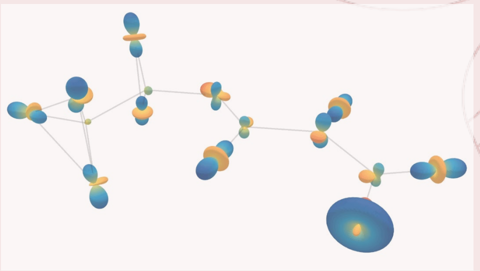
Common ways to improve equivariant GNNs' **expressiveness**:

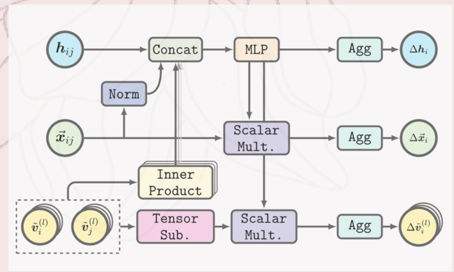
- Increase **Body Order**
- Increase **Represetation Degree**
- Increase **Model Layer**



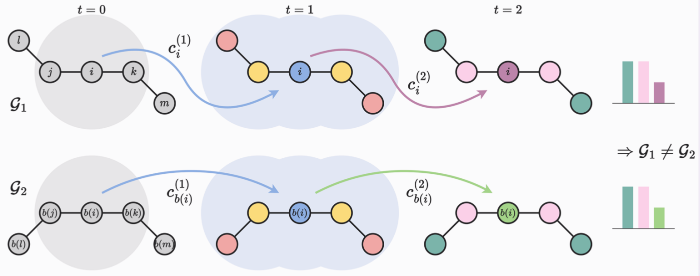


Body Order ++ (e.g. ACE, MACE)





Represetation Degree ++ (e.g.TFN, SEGNN, HEGNN)



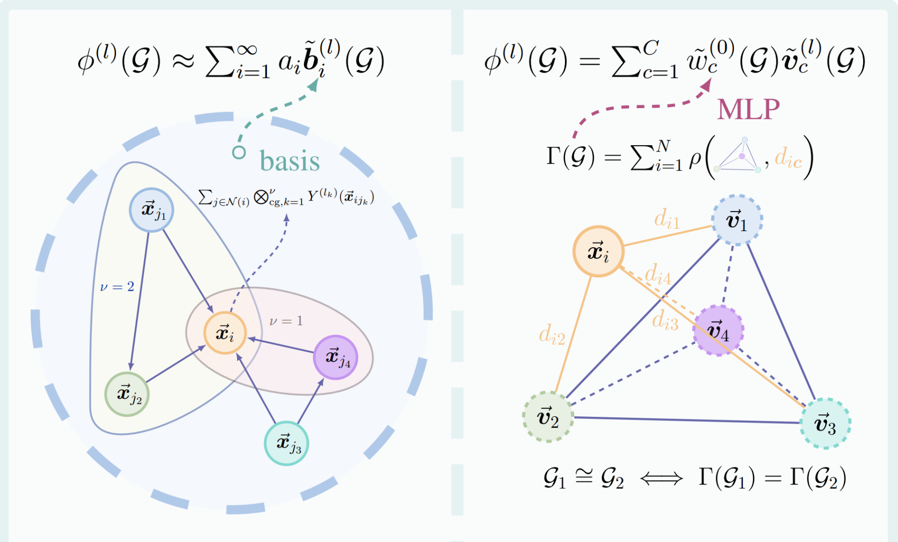
Model Layer ++ (e.g. GWL test)

Rethinking from Basis Construction

Example 3.1 (Basis Set of Common Equivariant GNNs)**.** The basis set for common equivariant GNNs, such as EGNN [38], HEGNN [41], TFN [28], and MACE [30], can be unified into the following form:

$$\mathbb{B}_{\nu}^{(l)} = \left\{\sum_i \sum_{\boldsymbol{j}\in\mathcal{N}(i)} \bigotimes_{\mathrm{cg},k=1}^{\nu} Y^{(l_k)}(\vec{\boldsymbol{x}}_{ij_k}/\|\vec{\boldsymbol{x}}_{ij_k}\|)\right\}, \qquad (3)$$

where $\nu \geq 1$ denotes the body order, $\boldsymbol{j} = (j_1, \dots, j_\nu)$ represents all chosen ordered neighbors, and $\vec{\boldsymbol{x}}_{ij_k} := \vec{\boldsymbol{x}}_i - \vec{\boldsymbol{x}}_{j_k}$. When only single-body neighbor is considered (*i.e.*, $\nu = 1$), Eq. (3) forms the basis set for EGNN [38] and HEGNN [41], applicable to degrees $l = 1$ and $l \geq 1$, respectively. In contrast, for multi-body interactions, Eq. (3) corresponds to TFN [28] when the body order $\nu = 1$, or MACE [30] with higher body orders $\nu \geq 1$.



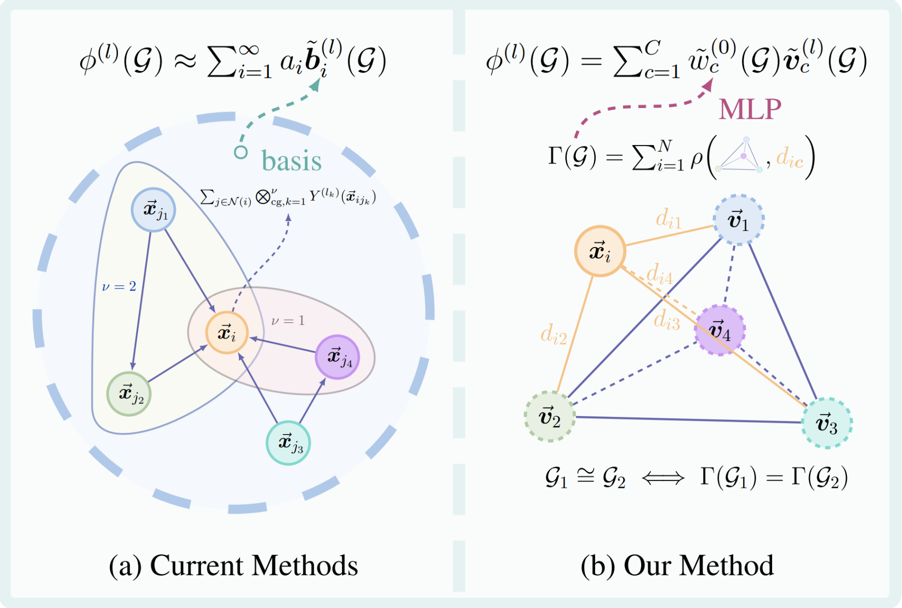
(a) Current Methods (b) Our Method

- Expanding from **single-particle** functions to **multi-particle** functions
- Extensions between function bases require **tensor products**
- Requires **extremely high order** and **model layers** to improve expressiveness
- Extremely **computationally expensive**
- **Complete** under ideal (but completely **unrealistic**) circumstances

Reformulating from Output Space

Theorem 3.2 (Dynamic Method). *Given a geometric graph $\mathcal{G}$, suppose there is a matrix $\tilde{\boldsymbol{V}}^{(l)}(\mathcal{G})$ with C channels of lth-degree steerable features denoted as $\tilde{\boldsymbol{v}}_c^{(l)}(\mathcal{G})$ satisfying* $\operatorname{span}(\tilde{\boldsymbol{V}}^{(l)}(\mathcal{G})) = \mathbb{F}^{(l)}(\mathcal{G}) := \{\tilde{\boldsymbol{f}}^{(l)}(\mathcal{G}) \mid \tilde{\boldsymbol{f}}^{(l)} \in \mathbb{F}^{(l)}\} \subset \mathbb{R}^{2l+1}$. *Then for any lth-degree steerable function $\tilde{\boldsymbol{\phi}}^{(l)} \in \mathbb{F}^{(l)}$, there always exists $\tilde{\boldsymbol{w}}^{(0)}(\mathcal{G}) := [\tilde{w}_c^{(0)}(\mathcal{G})]_{c=1}^{C}$ with C-channel output scalars, such that*

$$\tilde{\boldsymbol{\phi}}^{(l)}(\mathcal{G}) = \sum_{c=1}^{C} \tilde{w}_c^{(0)}(\mathcal{G})\tilde{\boldsymbol{v}}_c^{(l)}(\mathcal{G}). \tag{4}$$



(a) Current Methods (b) Our Method

- Is it possible, and under what conditions, to achieve **completeness (universal approximation)**?
- How can the required complete scalar function be **efficiently implemented**?
- How can the corresponding **full-rank basis functions** be obtained?

- Transform the problem of **complete scalar functions** into determining **geometric graph isomorphism**.
- Further recast determining geometric graph isomorphism as constructing a **canonical form for geometric graphs**.

Definition 3.3 (Geometric Isomorphism). *Two geometric graphs* $\mathcal{G}(\vec{\boldsymbol{X}}^{(\mathcal{G})}, \boldsymbol{A}^{(\mathcal{G})})$ *and* $\mathcal{H}(\vec{\boldsymbol{X}}^{(\mathcal{H})}, \boldsymbol{A}^{(\mathcal{H})})$ *are called* geometrically isomorphic *if they fulfill both of the following isomorphisms*

1. ***Point Cloud Isomorphism:*** *The two point clouds* $\vec{\boldsymbol{X}}^{(\mathcal{G})}$ *and* $\vec{\boldsymbol{X}}^{(\mathcal{H})}$ *are isomorphic,* i.e., $\exists\sigma \in \mathrm{S}_N, \mathfrak{g} \in \mathrm{E}(3), \forall i, \vec{\boldsymbol{x}}_i^{(\mathcal{G})} = \mathfrak{g} \cdot \vec{\boldsymbol{x}}_{\sigma(i)}^{(\mathcal{H})}$. *Here, all* $\langle \sigma, \mathfrak{g} \rangle$ *make a nonempty set* $\mathbb{M}(\mathcal{G}, \mathcal{H})$.
2. ***Topological Isomorphism:*** *The topological graphs associated with the point clouds are isomorphic,* i.e., $\exists \langle \sigma, \mathfrak{g} \rangle \in \mathbb{M}(\mathcal{G}, \mathcal{H}), \forall i, \forall j, [\boldsymbol{A}_{ij}^{(\mathcal{G})}] = [\boldsymbol{A}_{\sigma(i)\sigma(j)}^{(\mathcal{H})}]$.

Moreover, we denote the geometric isomorphism between $\mathcal{G}$ *and* $\mathcal{H}$ *as* $\mathcal{G} \cong \mathcal{H}$.

Definition 3.4 (Canonical Form of Geometric Graph). *A canonical form of geometric graph is a graph-level scalar function* $\Gamma : (\mathbb{R}^{N\times 3}, \mathbb{R}^{N\times N}) \to \mathbb{R}^H$, *satisfy* $\mathcal{G} \cong \mathcal{H} \iff \Gamma(\mathcal{G}) = \Gamma(\mathcal{H})$.

Construct Canonical Form

Algorithm 3: A canonical form of geometric graphs.

Data: A geometric graph $\mathcal{G}$, and $\psi_{\text{node}}, \psi_{\text{edge}}, \psi_{\text{graph}}$ are DeepSet models.

Result: The canonical form $\Gamma \in \mathbb{R}^H$ of point clouds $\mathcal{G}$.

`// ` $\mathcal{O}(N^4)$`, traverse all permutations.`

$\mathbb{T} \leftarrow \varnothing$;

for *any ordered set containing four non-coplanar points* $\vec{\boldsymbol{U}}_{\boldsymbol{\alpha}} \leftarrow \{\vec{\boldsymbol{u}}_{\alpha_i}\}_{i=1}^{4}$ *in* $\mathcal{G}$ **do**

  for $\vec{\boldsymbol{u}}_i \in \vec{\boldsymbol{X}}^{(\mathcal{G})}$ **do**

    `// ` $\mathcal{O}(N)$`, get a 4-channel scalar vector.`

    $\boldsymbol{d}_i \leftarrow (\|\vec{\boldsymbol{u}}_i - \vec{\boldsymbol{u}}_{\alpha_1}\|, \|\vec{\boldsymbol{u}}_i - \vec{\boldsymbol{u}}_{\alpha_2}\|, \|\vec{\boldsymbol{u}}_i - \vec{\boldsymbol{u}}_{\alpha_3}\|, \|\vec{\boldsymbol{u}}_i - \vec{\boldsymbol{u}}_{\alpha_4}\|)$;

  end

  `// ` $\mathcal{O}(N+N^2)$`, convert point sets and edge sets into scalar sets.`

  $\mathbb{D} \leftarrow \texttt{set}([\boldsymbol{d}_i]_{i=1}^{N}), \mathbb{E} \leftarrow \texttt{set}([\boldsymbol{d}_i, \boldsymbol{d}_j, \boldsymbol{e}_{ij}]_{\langle i,j\rangle \in \mathcal{E}})$;

  `// ` $\mathcal{O}(1)$`, decentralization of the four reference points.`

  $\vec{\boldsymbol{U}}_{\boldsymbol{\alpha}} \leftarrow (\boldsymbol{I}_{4\times 4} - \frac{1}{4}\mathbf{1}_{4\times 4})\vec{\boldsymbol{U}}_{\boldsymbol{\alpha}}$;

  `// ` $\mathcal{O}(N+N^2)$`, get the embedding based on current four points.`

  $\Gamma_{\boldsymbol{\alpha}} \leftarrow \texttt{concat}(\vec{\boldsymbol{U}}_{\boldsymbol{\alpha}}^{\top}\vec{\boldsymbol{U}}_{\boldsymbol{\alpha}}, \psi_{\text{node}}(\mathbb{D}), \psi_{\text{edge}}(\mathbb{E}))$;

  $\mathbb{T} \leftarrow \mathbb{T} \cup \{\Gamma_{\boldsymbol{\alpha}}\}$;

end

$\Gamma \longleftarrow \psi_{\text{graph}}(\mathbb{T})$;

return Γ;

A Faster Method

Using **virtual nodes** to bypass **quadratic traversal**, where the virtual nodes could be generated via models like FastEGNN.

Algorithm 4: A faster method to construct canonical form.

Data: A geometric graph $\mathcal{G}$, and $\psi_{\text{node}}, \psi_{\text{edge}}$ are DeepSet models.

Result: The canonical form $\Gamma \in \mathbb{R}^H$ of point clouds $\mathcal{G}$.

`// Get four non-coplanar reference points via generation.`

$\vec{\boldsymbol{V}} \leftarrow \zeta(\mathcal{G})$;

for $\vec{\boldsymbol{u}}_i \in \vec{\boldsymbol{X}}^{(\mathcal{G})}$ **do**

    `//` $\mathcal{O}(N)$`, get a 4-channel scalar vector.`

    $\boldsymbol{d}_i \leftarrow (\|\vec{\boldsymbol{u}}_i - \vec{\boldsymbol{v}}_1\|, \|\vec{\boldsymbol{u}}_i - \vec{\boldsymbol{v}}_2\|, \|\vec{\boldsymbol{u}}_i - \vec{\boldsymbol{v}}_3\|, \|\vec{\boldsymbol{u}}_i - \vec{\boldsymbol{v}}_4\|)$;

end

`//` $\mathcal{O}(N+N^2)$`, convert point sets and edge sets into scalar sets.`

$\mathbb{D} \leftarrow \texttt{set}([\boldsymbol{d}_i]_{i=1}^N)$, $\mathbb{E} \leftarrow \texttt{set}([\boldsymbol{d}_i, \boldsymbol{d}_j, \boldsymbol{e}_{ij}]_{\langle i,j\rangle \in \mathcal{E}})$;

`//` $\mathcal{O}(1)$`, decentralization of the four reference points.`

$\vec{\boldsymbol{V}} \leftarrow (\boldsymbol{I}_{4\times 4} - \frac{1}{4}\mathbf{1}_{4\times 4})\vec{\boldsymbol{V}}$;

`//` $\mathcal{O}(N+N^2)$`, get the embedding based on current four points.`

$\Gamma \leftarrow \texttt{concat}(\vec{\boldsymbol{V}}^\top \vec{\boldsymbol{V}}, \psi_{\text{node}}(\mathbb{D}), \psi_{\text{edge}}(\mathbb{E}))$;

return Γ;

Expressive Experiments

Expressive Experiments of GWL-test and IASR test, together with the ability to determine chirality.

Table 1: *The Completeness Test.*

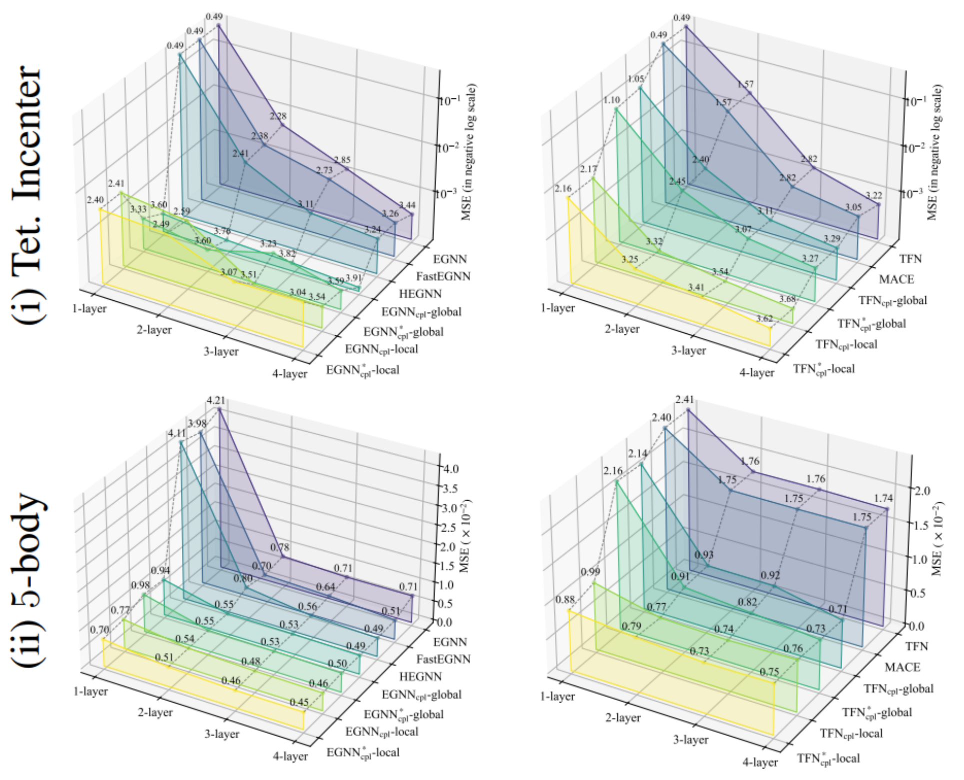
GNN Layer	The Completeness Test 2-body (Table. 3 in GWL)	3-body (Fig. 2(b) in IASR)
$\text{SchNet}_{\text{2-body}}$	50.0 ± 0.0	50.0 ± 0.0
$\text{EGNN}_{\text{2-body}}$	50.0 ± 0.0	50.0 ± 0.0
$\text{GVP-GNN}_{\text{3-body}}$	**100.0** ± 0.0	50.0 ± 0.0
$\text{TFN}_{\text{2-body}}$	50.0 ± 0.0	50.0 ± 0.0
$\text{MACE}_{\text{3-body}}$	**100.0** ± 0.0	50.0 ± 0.0
$\text{MACE}_{\text{4-body}}$	**100.0** ± 0.0	**100.0** ± 0.0
$\text{Basic}_{\text{cpl}}$	**100.0** ± 0.0	**100.0** ± 0.0
$\text{SchNet}_{\text{cpl}}$	**100.0** ± 0.0	**100.0** ± 0.0
EGNN_{cpl}	**100.0** ± 0.0	**100.0** ± 0.0
$\text{GVP-GNN}_{\text{cpl}}$	**100.0** ± 0.0	**100.0** ± 0.0
TFN_{cpl}	**100.0** ± 0.0	**100.0** ± 0.0

Table 2: *The Chirality Test.*

	# Color	# TP	The Chirality Test Fig. 4(a)	Fig. 4(b)	Fig. 4(c)
Basic	$\varnothing$		50.0 ± 0.0	50.0 ± 0.0	50.0 ± 0.0
	$\oplus$		**100.0** ± 0.0	**100.0** ± 0.0	50.0 ± 0.0
	$\otimes$		**100.0** ± 0.0	**100.0** ± 0.0	**100.0** ± 0.0
	$\varnothing$	✓	75.0 ±15.0	95.0 ±15.0	**100.0** ± 0.0
	$\oplus$	✓	**100.0** ± 0.0	**100.0** ± 0.0	**100.0** ± 0.0
	$\otimes$	✓	**100.0** ± 0.0	**100.0** ± 0.0	**100.0** ± 0.0
EGNN	$\varnothing$		50.0 ± 0.0	50.0 ± 0.0	50.0 ± 0.0
	$\oplus$		**100.0** ± 0.0	**100.0** ± 0.0	50.0 ± 0.0
	$\otimes$		**100.0** ± 0.0	**100.0** ± 0.0	**100.0** ± 0.0
	$\varnothing$	✓	**100.0** ± 0.0	**100.0** ± 0.0	**100.0** ± 0.0
	$\oplus$	✓	**100.0** ± 0.0	**100.0** ± 0.0	**100.0** ± 0.0
	$\otimes$	✓	**100.0** ± 0.0	**100.0** ± 0.0	**100.0** ± 0.0

Synthetic Dataset

A dataset with **graph-level** target (Tetrahedron Center Prediction) & A dataset with **node-level** target (5-body).



(a) EGNN$_{cpl}$ Model (b) TFN$_{cpl}$ Model

Figure 3: Visualization of MSE loss.

Table 4: MSE loss on 5-body system.

	MSE Loss on 5-body system ($\times 10^{-2}$)			
	1-layer	2-layer	3-layer	4-layer
EGNN	4.214	0.780	0.710	0.712
FastEGNN	3.983	0.705	0.640	0.509
HEGNN	4.114	0.801	0.561	0.489
TFN	2.411	1.758	1.758	1.739
MACE	2.403	1.754	1.746	1.746
Equiformer	0.805	0.682	<u>0.465</u>	0.657
EGNN$_{cpl}$-global	0.943	0.546	0.530	0.492
EGNN$^*_{cpl}$-global	0.985	0.554	0.533	0.498
EGNN$_{cpl}$-local	<u>0.768</u>	<u>0.537</u>	0.481	<u>0.458</u>
EGNN$^*_{cpl}$-local	**0.703**	**0.513**	**0.455**	**0.450**
TFN$_{cpl}$-global	2.144	0.934	0.916	0.708
TFN$^*_{cpl}$-global	2.163	0.910	0.825	0.734
TFN$_{cpl}$-local	0.988	0.766	0.738	0.761
TFN$^*_{cpl}$-local	0.883	0.792	0.734	0.746

Large-scale & Real-world Dataset

Table 5: 100-body dataset.

	MSE Loss ($\times 10^{-2}$)
EGNN	1.36
FastEGNN	1.10
$\text{TFN}_{l\leq 2}$	3.77
$\text{MACE}_{l\leq 2}$	3.83
$\text{Equiformer}_{l\leq 2}$	0.90
$\text{HEGNN}_{l\leq 1}$	1.13
$\text{HEGNN}_{l\leq 2}$	0.97
$\text{HEGNN}_{l\leq 3}$	0.94
$\text{HEGNN}_{l\leq 6}$	$\underline{0.86}$
EGNN_{cpl}-global	0.98
EGNN_{cpl}-local	**0.73**
$\text{TFN}_{\text{cpl}}\text{-global}_{l\leq 2}$	1.78
$\text{TFN}_{\text{cpl}}\text{-local}_{l\leq 2}$	1.73

Table 6: Prediction error ($\times 10^{-2}$) on MD17 dataset (3 runs).

	Aspirin	Benzene	Ethanol	Malonal.	Naph.	Salicylic	Toluene	Uracil
EGNN	$14.41_{\pm 0.15}$	$62.40_{\pm 0.53}$	$4.64_{\pm 0.01}$	$13.64_{\pm 0.01}$	$0.47_{\pm 0.02}$	$1.02_{\pm 0.02}$	$11.78_{\pm 0.07}$	$0.64_{\pm 0.01}$
FastEGNN	$9.81_{\pm 0.11}$	$60.84_{\pm 0.14}$	$4.65_{\pm 0.00}$	$12.82_{\pm 0.02}$	$0.38_{\pm 0.00}$	$1.05_{\pm 0.08}$	$10.88_{\pm 0.08}$	$0.56_{\pm 0.01}$
$\text{TFN}_{l\leq 2}$	$12.37_{\pm 0.18}$	$58.48_{\pm 1.98}$	$4.81_{\pm 0.04}$	$13.62_{\pm 0.08}$	$0.49_{\pm 0.01}$	$1.03_{\pm 0.02}$	$10.89_{\pm 0.01}$	$0.84_{\pm 0.02}$
$\text{MACE}_{l\leq 2}$	$10.43_{\pm 0.44}$	$59.71_{\pm 2.21}$	$4.83_{\pm 0.03}$	$13.78_{\pm 0.04}$	$0.44_{\pm 0.02}$	$0.94_{\pm 0.01}$	$10.20_{\pm 0.11}$	$0.74_{\pm 0.01}$
$\text{Equiformer}_{l\leq 2}$	$9.84_{\pm 0.10}$	$\mathbf{33.28}_{\pm 0.15}$	$4.69_{\pm 0.03}$	$13.06_{\pm 0.04}$	$\underline{0.34}_{\pm 0.01}$	$0.86_{\pm 0.01}$	$\mathbf{9.50}_{\pm 0.09}$	$0.57_{\pm 0.01}$
$\text{HEGNN}_{l\leq 1}$	$10.32_{\pm 0.58}$	$62.53_{\pm 7.62}$	$\underline{4.63}_{\pm 0.01}$	$12.85_{\pm 0.01}$	$0.38_{\pm 0.01}$	$0.90_{\pm 0.05}$	$10.56_{\pm 0.10}$	$0.56_{\pm 0.02}$
$\text{HEGNN}_{l\leq 2}$	$10.04_{\pm 0.45}$	$61.80_{\pm 5.92}$	$\underline{4.63}_{\pm 0.01}$	$12.85_{\pm 0.01}$	$0.39_{\pm 0.01}$	$0.91_{\pm 0.06}$	$10.56_{\pm 0.05}$	$0.55_{\pm 0.01}$
$\text{HEGNN}_{l\leq 3}$	$10.20_{\pm 0.23}$	$62.82_{\pm 4.25}$	$\underline{4.63}_{\pm 0.01}$	$12.85_{\pm 0.02}$	$0.37_{\pm 0.01}$	$0.94_{\pm 0.10}$	$10.55_{\pm 0.16}$	$\mathbf{0.52}_{\pm 0.01}$
$\text{HEGNN}_{l\leq 6}$	$9.94_{\pm 0.07}$	$59.93_{\pm 5.21}$	$4.62_{\pm 0.01}$	$12.85_{\pm 0.01}$	$0.37_{\pm 0.02}$	$0.88_{\pm 0.02}$	$10.56_{\pm 0.33}$	$0.54_{\pm 0.01}$
EGNN_{cpl}-global	$9.60_{\pm 0.09}$	$58.24_{\pm 1.40}$	$4.64_{\pm 0.01}$	$12.85_{\pm 0.01}$	$0.39_{\pm 0.01}$	$0.95_{\pm 0.05}$	$10.37_{\pm 0.16}$	$0.56_{\pm 0.02}$
EGNN_{cpl}-local	$\underline{9.52}_{\pm 0.42}$	$\underline{44.90}_{\pm 1.53}$	$\mathbf{4.62}_{\pm 0.00}$	$\mathbf{12.80}_{\pm 0.02}$	$0.36_{\pm 0.02}$	$0.94_{\pm 0.05}$	$\underline{10.21}_{\pm 0.06}$	$0.57_{\pm 0.00}$
$\text{TFN}_{\text{cpl}}\text{-global}_{l\leq 2}$	$\mathbf{9.49}_{\pm 0.04}$	$58.24_{\pm 0.42}$	$\underline{4.63}_{\pm 0.00}$	$\underline{12.82}_{\pm 0.00}$	$\mathbf{0.33}_{\pm 0.00}$	$\mathbf{0.80}_{\pm 0.00}$	$10.24_{\pm 0.02}$	$\underline{0.53}_{\pm 0.00}$
$\text{TFN}_{\text{cpl}}\text{-local}_{l\leq 2}$	$\underline{9.52}_{\pm 0.07}$	$48.77_{\pm 6.51}$	$4.64_{\pm 0.00}$	$12.83_{\pm 0.02}$	$\underline{0.34}_{\pm 0.00}$	$\underline{0.81}_{\pm 0.01}$	$10.95_{\pm 0.01}$	$\underline{0.53}_{\pm 0.00}$

Table 7: Results on Water-3D-mini.

	MSE Loss on Water-3D-mini ($\times 10^{-4}$)			
	1-layer	2-layer	3-layer	4-layer
EGNN	4.904	4.323	3.649	3.338
FastEGNN	4.885	4.332	3.782	3.259
HEGNN	4.885	4.138	3.519	3.287
EGNN_{cpl}-global	4.368	3.711	3.294	3.248
EGNN_{cpl}-local	**3.611**	**3.320**	**2.803**	**2.495**

- Large-scale Dataset: 100-body, Water-3D-mini (>8,000 nodes)
- Real-world Dataset: MD-17, Water-3D-mini

Reference

[1] Joshi C K, Bodnar C, Mathis S V, et al. On the expressive power of geometric graph neural networks[C]//International conference on machine learning. PMLR, 2023: 15330-15355.

[2] Drautz R. Atomic cluster expansion for accurate and transferable interatomic potentials[J]. Physical Review B, 2019, 99(1): 014104.

[3] Brandstetter J, Hesselink R, van der Pol E, et al. Geometric and physical quantities improve e (3) equivariant message passing[J]. arXiv preprint arXiv:2110.02905, 2021.

[4] Cen J, Li A, Lin N, et al. Are high-degree representations really unnecessary in equivariant graph neural networks?[J]. Advances in Neural Information Processing Systems, 2024, 37: 26238-26266.