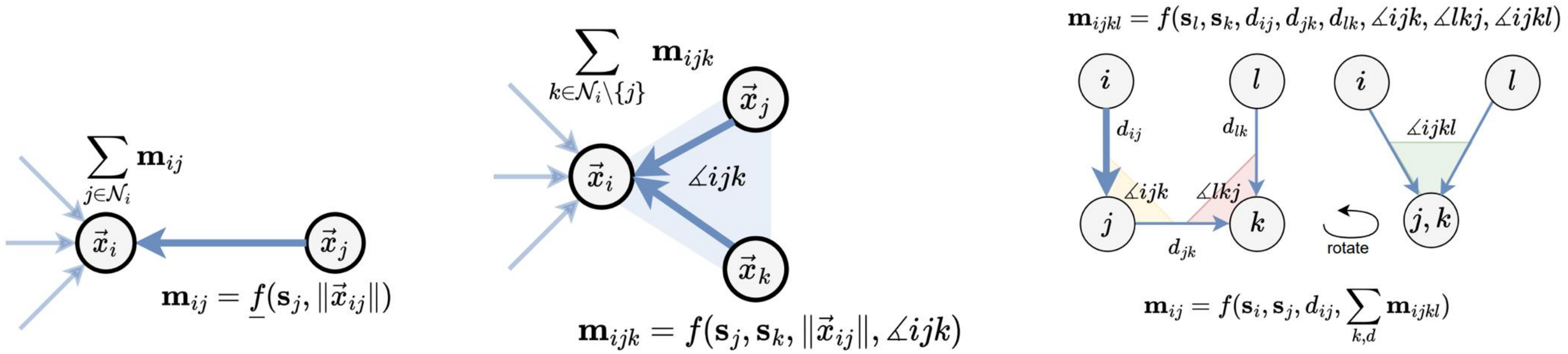
On the Completeness of Invariant Geometric Deep Learning Models

Zian Li, Xiyuan Wang, Shijia Kang, and Muhan Zhang*
Peking University

Introduction

Invariant models—an essential class of geometric deep learning architectures—excel at generating meaningful geometric representations by harnessing **invariant** features within point clouds.



(a) SchNet (b) DimeNet (c) GemNet

However, their theoretical *expressive power* still remains unclear, restricting a deeper understanding of the potential of such models.

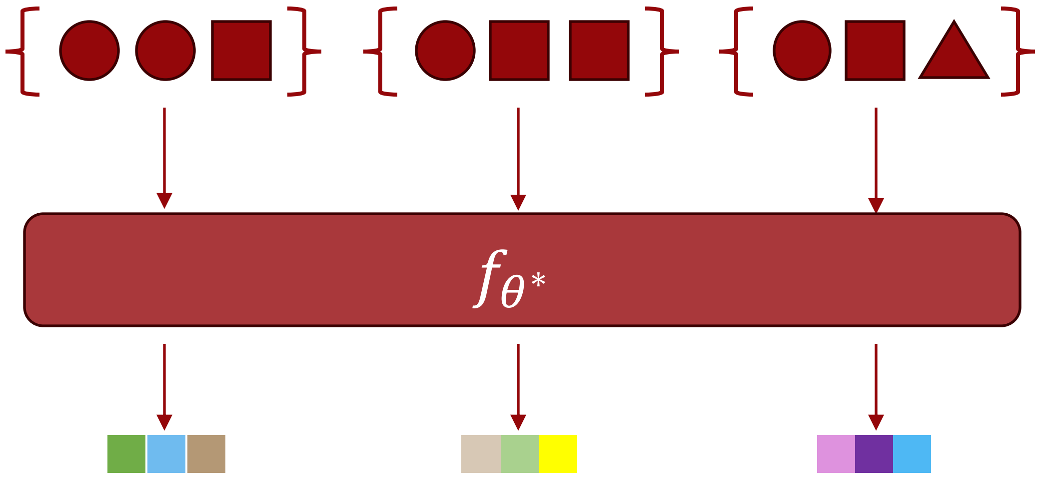
Image credit: A Hitchhiker's Guide to Geometric GNNs for 3D Atomic Systems

Introduction

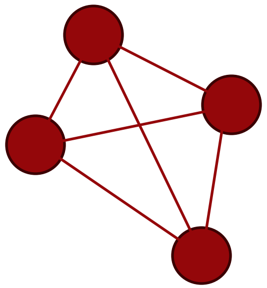
❓ **Question:** ***How expressive are existing invariant models?***

❗ **Assumptions:**

1. ***Injective*** **intermediate functions (especially, multiset functions). [1]**
2. ***Fully-connected*** **graph modeling (i.e., interactions happens among all points). [2]**



An injective multiset function maps different multisets to different embeddings

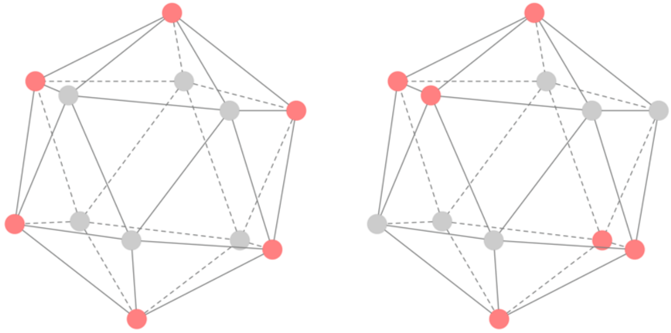


Fully-connected graph modeling

[1] Neural injective functions for multisets, measures and graphs via a finite witness theorem. Tal Amir, et al.
[2] Complete neural networks for complete euclidean graphs. Snir Hordan, et al.

Intuition

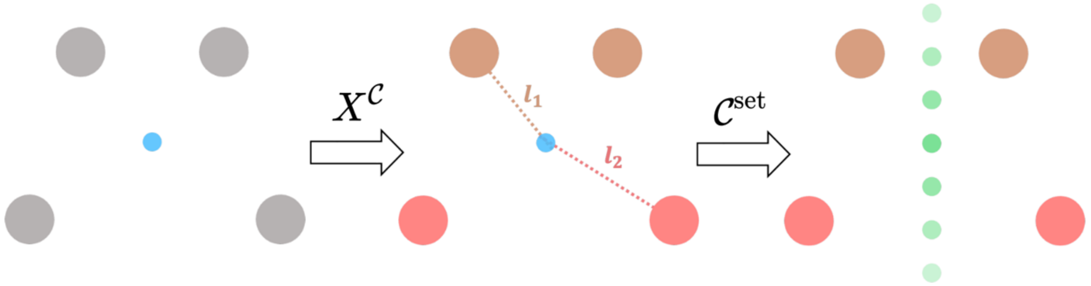
Intuition: The more *symmetric* a point cloud is, the more likely an **invariant** model is to be *"confused"* with another one.



A pair of symmetric point clouds that MPNN (with distance) cannot distinguish

How do we define *symmetry?*

$\mathcal{C}$-asymmetry: Augmenting node features with distance to the geometric center.

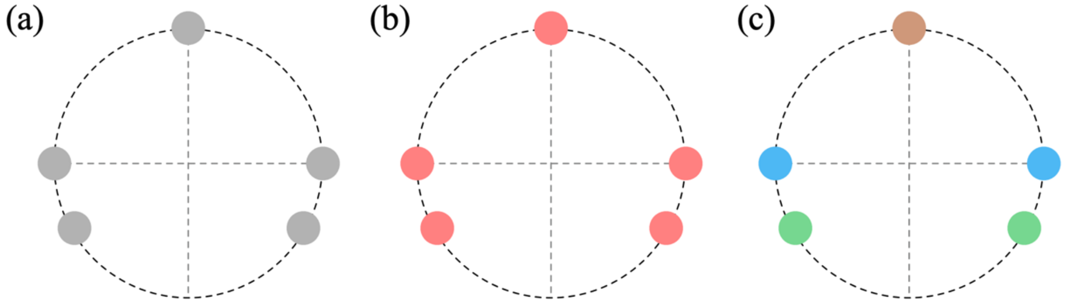


A point cloud that is $\mathcal{C}$-asymmetric.

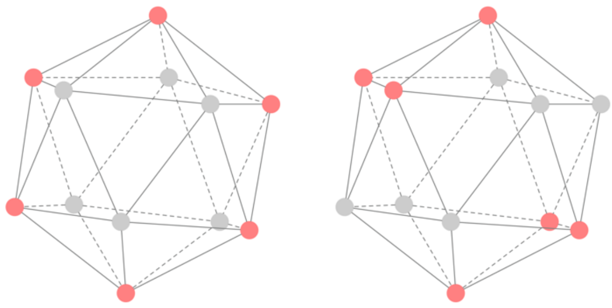
Image credit: Is Distance Matrix Enough for geometric deep learning?

Intuition

📐 $\mathcal{D}$***-asymmetry: Augmenting node features with MPNN (with distance).***



A point cloud that is $\mathcal{C}$-symmetric but $\mathcal{D}$-asymmetric.



A pair of symmetric point clouds that MPNN (with distance) cannot distinguish

💡 **Both $\mathcal{C}$- and $\mathcal{D}$-symmetric!**

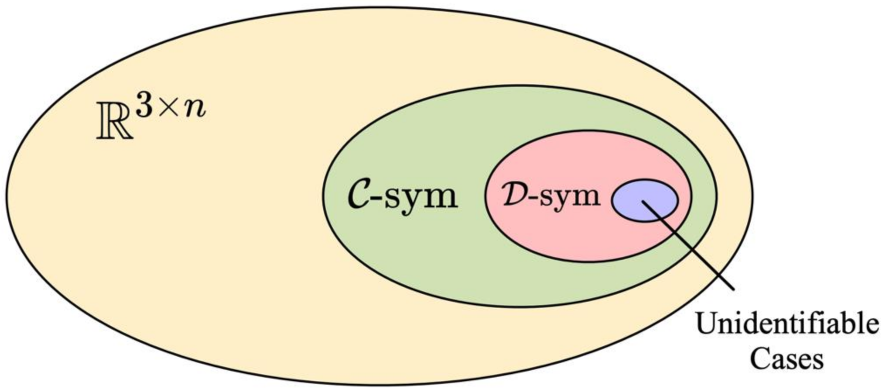
Image credit: Is Distance Matrix Enough for geometric deep learning?

5

DisGNN is near-complete

🔍 **Conclusion 1: Message-Passing GNN with distance (aka, DisGNN) is *nearly complete***

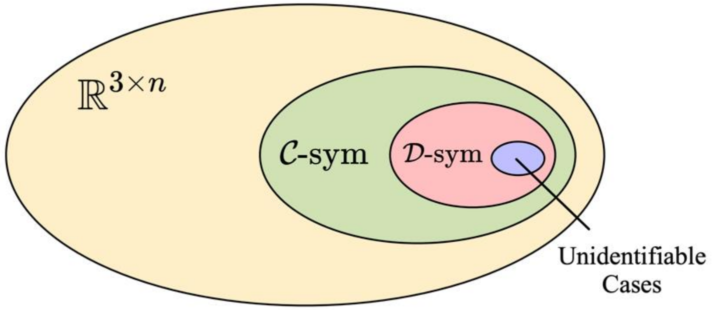
– all $\mathcal{C}$-asymmetry and $\mathcal{D}$-asymmetry point clouds can be identified!



Identify: *For a point cloud P_1, if $f(P_1) \neq f(P_2)$ holds for **all** non-isometric $P_2 \rightarrow f$ can identify P_1.*

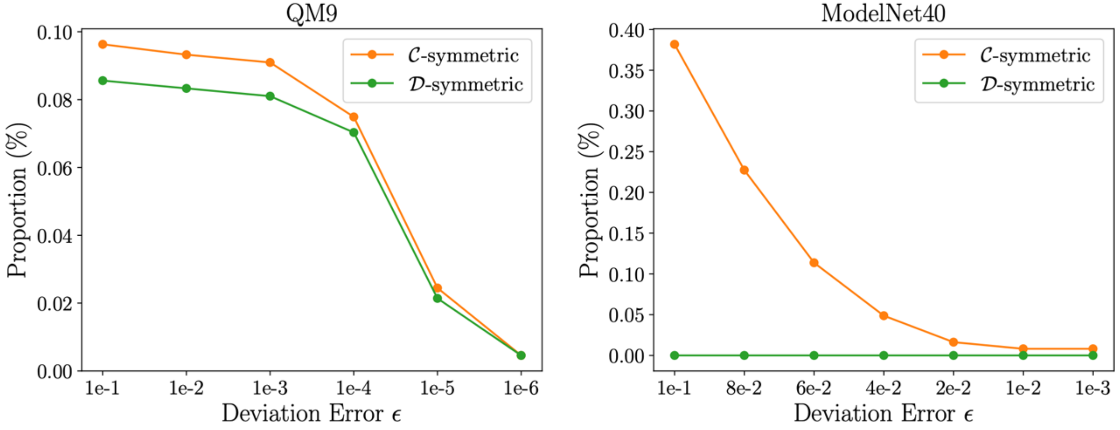
DisGNN is near-complete

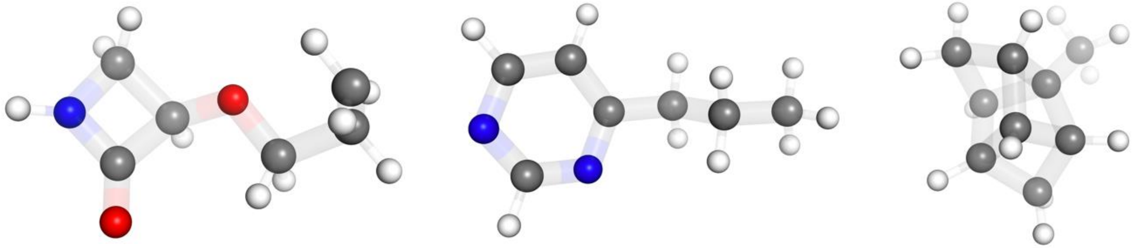
❓ **Explaination of "near-complete"?**



🔍 **Theoretically:** *$\mathcal{C}$-symmetric* and *$\mathcal{D}$-symmetric* and *unidentifiable* case spaces' **Lebesgue measure on $\mathbb{R}^{n\times3}$ are all 0 !**

🔍 **Empirically:** *such highly symmetric point clouds are quite* ***rare in real-world settings.***

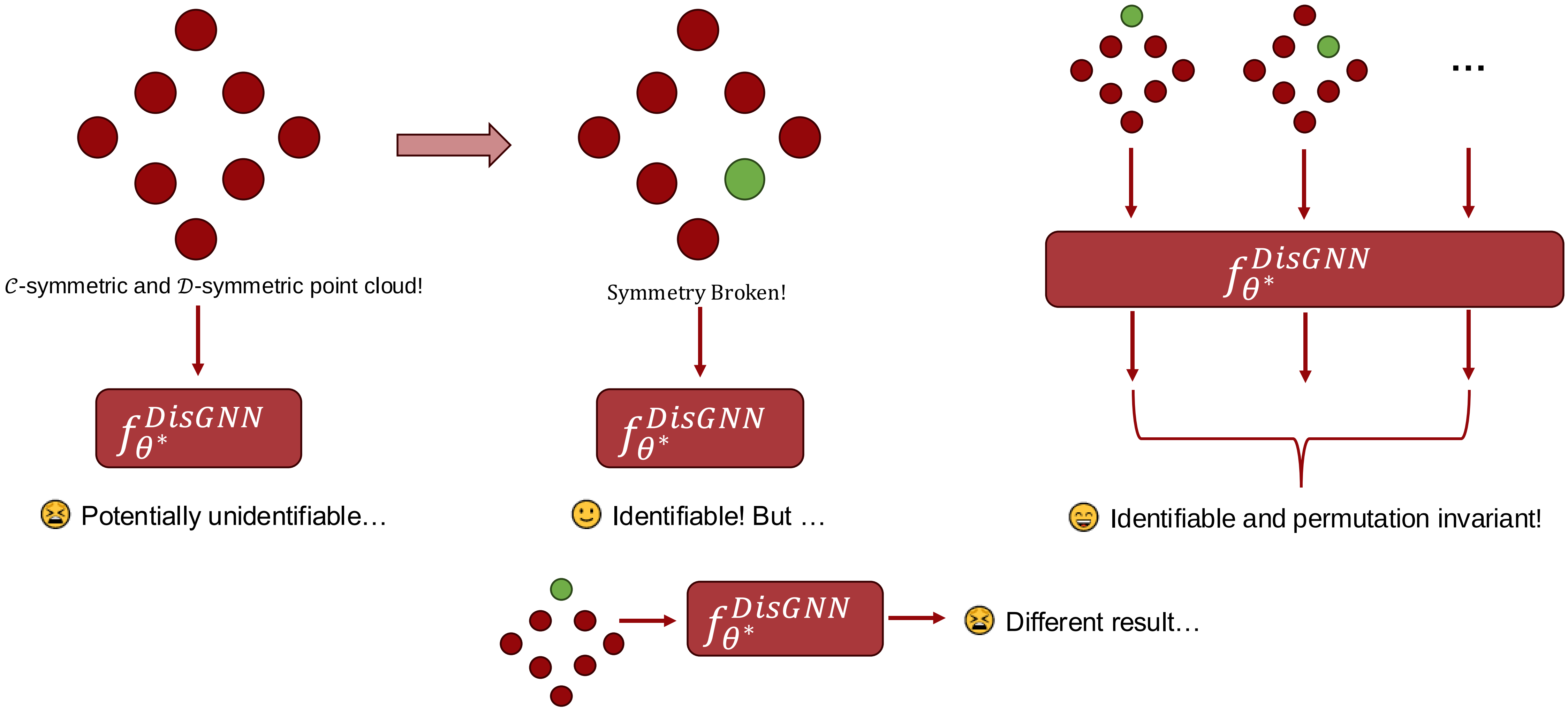




Molecules represented as point clouds –-- all can be identified!

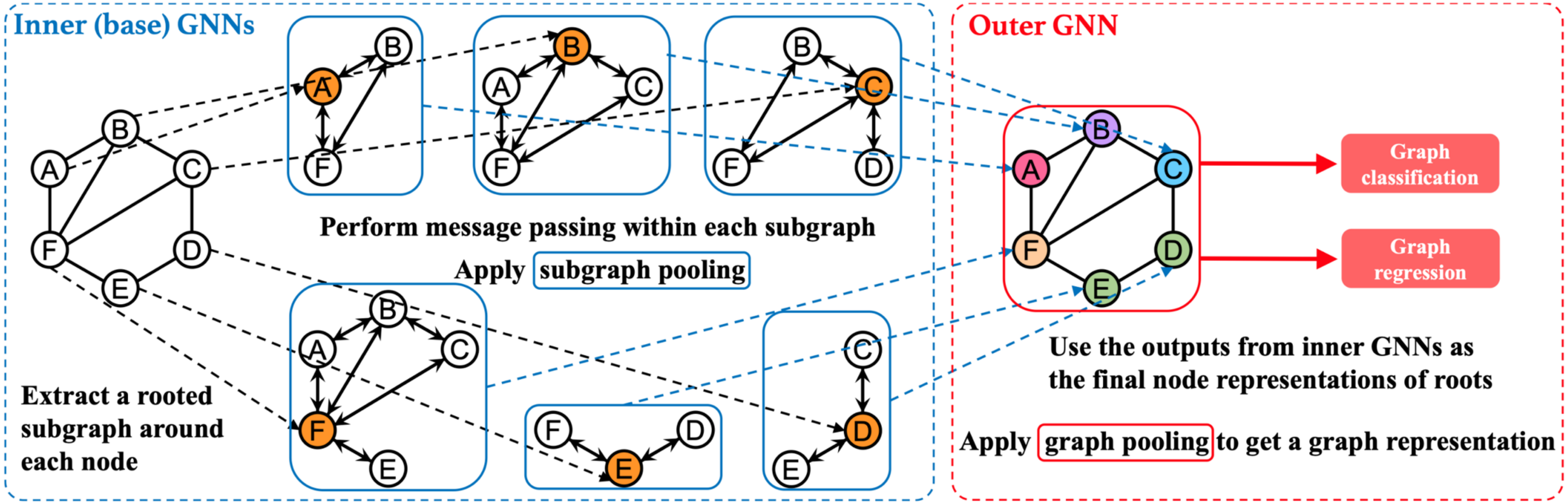
To Break Symmetry?

❓ **Can we break symmetry for a highly symmetric point cloud and further identify it?**



GeoNGNN

❗ **This is exactly the geometric version of the *simplest* subgraph GNN -- NGNN!**



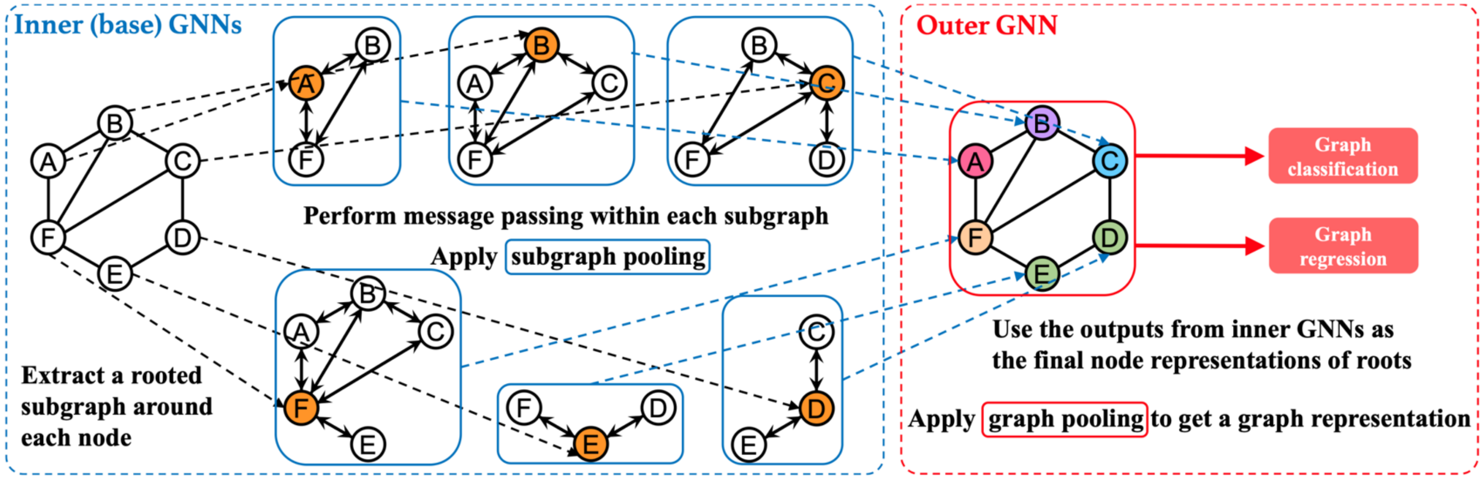
Framework of original NGNN

Image credit: Nested Graph Neural Network

GeoNGNN

Definition: *GeoNGNN*

- ***Graph Construction:*** *The point cloud is treated as a distance graph with a cutoff radius* r_{cutoff}.
- ***Base GNN:*** *Both inner and outer layers use DisGNN, with* N_{in} *and* N_{out} *layers respectively.*
- ***Subgraph Construction:*** *For each node* i*, its ego subgraph includes all nodes and edges within Euclidean distance* r_{sub} *with node* i *explicitly marked.*



Framework of original NGNN

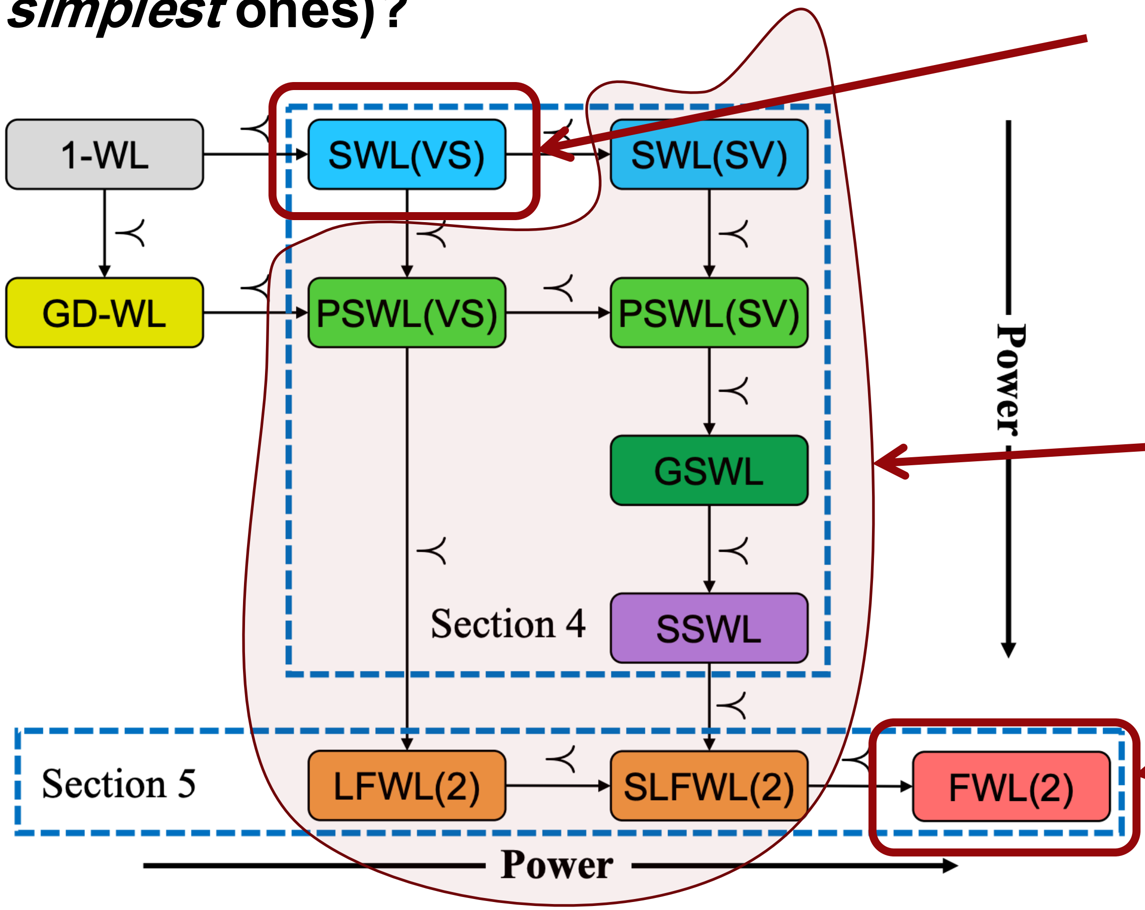
Image credit: Nested Graph Neural Network

🔍 **Conclusion 2:** When the design conditions are satisfied, GeoNGNN achieves E(3)-completeness:

- Point clouds are modeled as fully-connected graphs ($r_{cutoff} = +\infty$).
- Subgraphs span the entire graph ($r_{sub} = +\infty$).
- $N_{in} \geq 5, N_{out} \geq 0.$

Completeness of GeoNGNN

❓How about geometric versions of *other subgraph GNNs* (given that NGNN is one of the *simplest* ones)?



NGNN falls in. GeoNGNN is complete!

OSAN, GNN-AK, DSS-GNN, GNN-AK-ctx, SUN, RelGN(2) ... fall in!

2-F-DisGNN is complete!

Image credit: A Complete Expressiveness Hierarchy for Subgraph GNNs via Subgraph Weisfeiler-Lehman Tests

Completeness of GeoNGNN

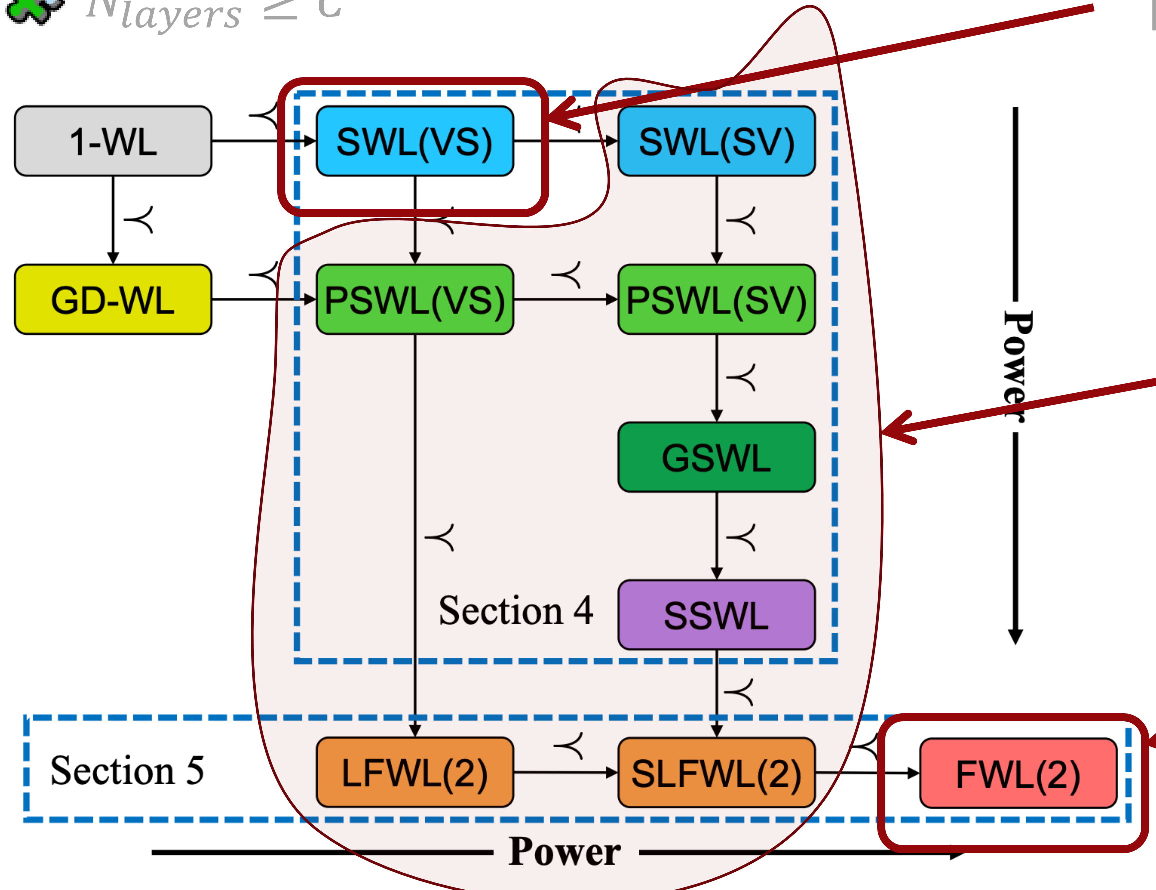
Conclusion 3: All general geometric subgraph GNNs are complete when:

- Point clouds are modeled as fully-connected graphs ($r_{cutoff} = +\infty$).
- Interactions span the entire graph ($r_{sub} = +\infty$).
- $N_{layers} \geq C$

Completeness of GeoNGNN

Conclusion 3: All general geometric subgraph GNNs are complete when:

- Point clouds are modeled as fully-connected graphs ($r_{cutoff} = +\infty$).
- Interactions span the entire graph ($r_{sub} = +\infty$).
- $N_{layers} \geq C$



NGNN falls in. GeoNGNN is complete!

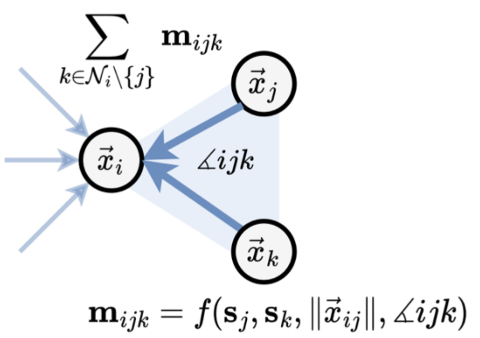
OSAN, GNN-AK, DSS-GNN, GNN-AK-ctx, SUN, RelGN(2) ... fall in! **Their geometric counterparts are complete!**

2-F-DisGNN is complete!

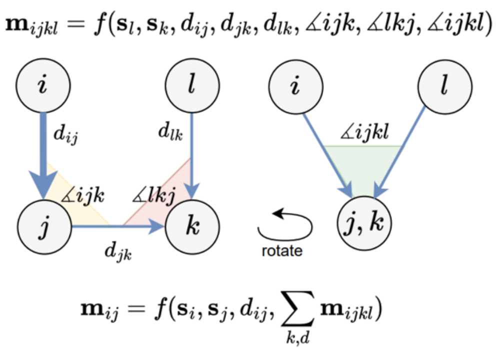
Image credit: A Complete Expressiveness Hierarchy for Subgraph GNNs via Subgraph Weisfeiler-Lehman Tests

Completeness of Well-established Models

❗ **DimeNet, GemNet and SphereNet are popular invariant geometric models …**



(b) DimeNet



(c) GemNet

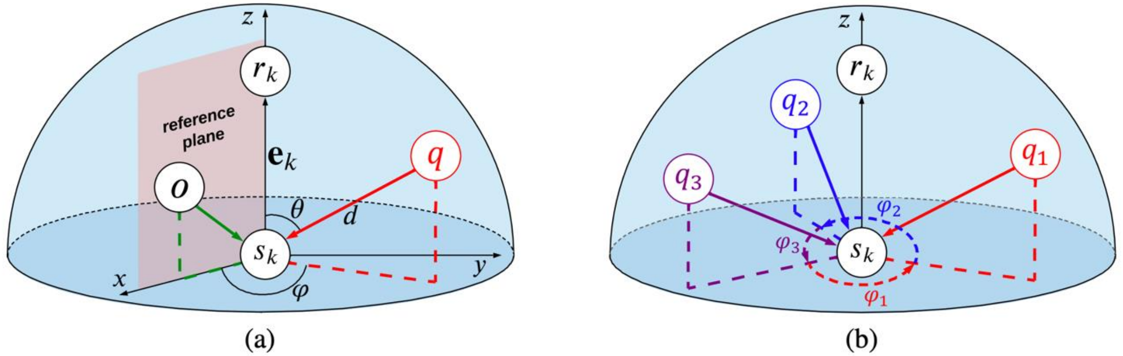


Figure 2: (a). The message aggregation scheme for the spherical message passing. (b). An illustration for computing torsion angles in the spherical message passing architecture.

🤔 **They tracks *edge representations* h_{ij} for edge (i, j)**

🤔 **GeoNGNN tracks subgraph-node representation h_{ij} for node j in subgraph i**

💡 **They can be mathematically aligned!**

Image credit:
[1] A Hitchhiker's Guide to Geometric GNNs for 3D Atomic Systems
[2] SphereNet

Completeness of Well-established Models

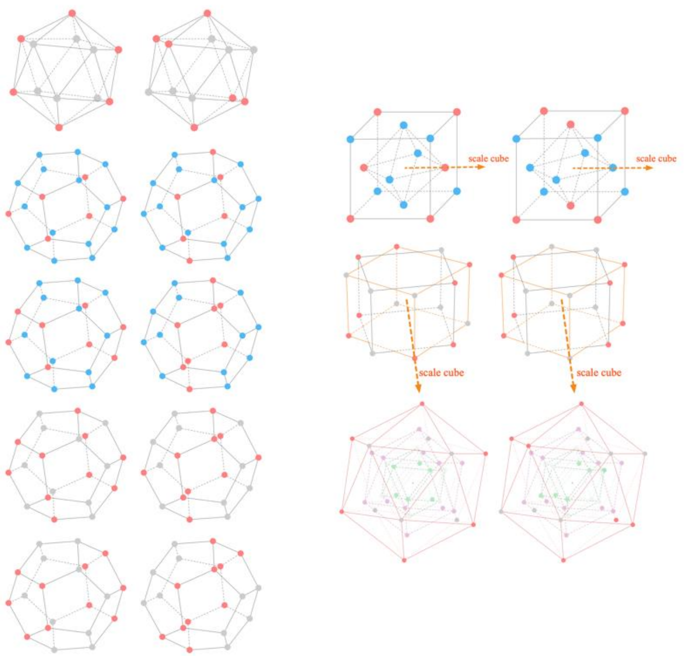
💡 **Intuition: To *implement* GeoNGNN with DimeNet/GemNet/SphereNet based on the mathematical *alignment* of the representations they track!**

🔍 **Conclusion 4:** When the design conditions are satisfied, **DimeNet/GemNet/SphereNet** achieves E(3)-completeness:

- They initialize and update all *edge* representations ($r_{\mathrm{emb}} = +\infty$)
- They interact with all neighbors, ($r_{int} = +\infty$)
- $N_{layers} \geq C$

Evaluations on "Confusable" Point Clouds

PEKING UNIVERSITY



Highly *symmetric* counterexamples taken from [1]
For each pair of point clouds, DisGNN *cannot distinguish* them.

Table 1: Separation results on the constructed geometric expressiveness dataset. Models for which we have theoretically established completeness are highlighted in gray.

	Invaraint						Equivaraint	
	SchNet	DisGNN	DimeNet	SphereNet	GemNet	GeoNGNN	PaiNN	MACE
Isolated (10 cases)	0%	0%	100%	100%	100%	100%	100%	100%
Combined (7 cases)	0%	0%	100%	100%	100%	100%	100%	100%

Image credit: [1] Is Distance Matrix Enough for Geometric Deep Learning

北京大學 PEKING UNIVERSITY

ICLR

Thanks for listening!