MCGM: Multi-stage Clustered Global Modeling for Long-range Interactions in Molecules

Haodong Pan[a,1], Yusong Wang[a,1], Nanning Zheng[a], Caigui Jiang[a,*]

[a]*State Key Laboratory of Human-Machine Hybrid Augmented Intelligence, National Engineering Research Center for Visual Information and Applications, and Institute of Artificial Intelligence and Robotics, Xi'an Jiaotong University, , Xi'an, 710049, Shanxi, China*

Abstract

Geometric graph neural networks (GNNs) excel at capturing molecular geometry, yet their locality-biased message passing hampers the modeling of long-range interactions. Current solutions have fundamental limitations: extending cutoff radii causes computational costs to scale cubically with distance; physics-inspired kernels (e.g., Coulomb, dispersion) are often system-specific and lack generality; Fourier-space methods require careful tuning of multiple parameters (e.g., mesh size, k-space cutoff) with added computational overhead. We introduce Multi-stage Clustered Global Modeling (MCGM), a lightweight, plug-and-play module that endows geometric GNNs with hierarchical global context through efficient clustering operations. MCGM builds a multi-resolution hierarchy of atomic clusters, distills global information via dynamic hierarchical clustering, and propagates this context back through learned transformations, ultimately reinforcing atomic features via residual connections. Seamlessly integrated into four diverse backbone architectures, MCGM reduces OE62 energy prediction error by an average of 26.2%. On AQM, MCGM achieves state-of-the-art accuracy (17.0 meV for energy, 4.9 meV/Å for forces) while using 20% fewer parameters than Neural P^3M. Code will be made available upon acceptance.

Keywords:

Geometric GNN, Long-Range Interaction, Multi-stage Clustering

---

[*]Corresponding author. Email:cgjiang@xjtu.edu.cn
[1]These authors contributed equally to this work.

1. Introduction

The intersection of computational physics and pattern recognition has revolutionized molecular modeling, shifting from explicitly solving electronic structure equations to discovering hidden patterns in molecular interactions through deep learning, as systematically reviewed by Reiser et al. [1]. Among these approaches, geometric graph neural networks (GNNs) [2, 3, 4] have emerged as a central framework due to their natural compatibility with molecular topologies and spatial geometry. Unlike traditional Density Functional Theory (DFT) [5], which requires iterative self-consistent field calculations, geometric GNNs enable direct end-to-end learning from data, providing rapid inference without expensive quantum mechanical computations. With $E(3)$-equivariant message passing and strong local geometric priors, GNNs can achieve near-DFT accuracy in energy and force predictions for molecular systems.

Current GNNs predominantly rely on local pattern extraction within fixed-radius neighborhoods, which becomes insufficient when long-range interactions are important for molecular properties. Although effective for small and compact molecules where all relevant interactions fall within typical cut-off radii (5-6 Å, 0.5-0.6 nm), this approach struggles with extended molecular systems. For example, in molecules with more than 50 atoms-common in drug discovery and materials science-important interactions often extend beyond 10 Å(1.0 nm)[6]. Increasing the cutoff radius may partially alleviate this issue, but causes computational costs to scale cubically [7, 8]. In contrast, ignoring long-range interactions introduces systematic errors [9, 10]. Therefore, incorporating efficient and scalable long-range modeling mechanisms has become a key challenge in geometric molecular modeling.

To address the challenge of capturing long-range interactions beyond local neighborhoods, current approaches face distinct limitations: physics-based methods[11, 12] incorporate explicit potentials but lack generality across chemical systems; Fourier-space methods[8, 9, 13] achieve global modeling but require significant computational overhead and complex parameter tuning; and existing hierarchical methods such as LSRM[6] employ fixed fragmentation rules that cannot adapt to diverse molecular topologies. In contrast, MCGM introduces adaptive hierarchical modeling through

dynamic clustering, automatically discovering multi-scale patterns through efficient bounded operations-providing a practical and generalizable solution for long-range molecular interactions.

In this work, we propose **Multi-Stage Clustered Global Modeling (MCGM)**, a pioneering framework that fundamentally reimagines how molecular interactions are modeled hierarchically. Unlike all existing approaches that impose rigid structural assumptions—whether through physical equations, chemical rules, or spatial grids—MCGM achieves adaptive hierarchical decomposition through dynamic clustering in learned representation spaces. The key innovation lies in MCGM's ability to discover, rather than prescribe, molecular organization: at each training epoch, it dynamically reorganizes atoms into hierarchical clusters based on learned representations, discovering multi-scale structural patterns driven entirely by the learning objective without chemical priors. This dynamic hierarchy adapts not only to different molecular systems, but also evolves during training as the model learns richer representations, enabling the automatic discovery of task-relevant structural patterns that fixed hierarchies cannot capture. Remarkably, MCGM achieves this adaptive modeling through efficient hierarchical operations with a bounded computational cost per molecule, avoiding the computational overhead of Fourier-based methods like Neural P^3M[13] that require complex mesh constructions and parameter tuning. Moreover, its plug-and-play design allows seamless integration with any geometric GNN architecture without modifications, demonstrating that hierarchical global modeling can be both powerful and practical. This unique combination of adaptivity, efficiency, and modularity establishes MCGM as a new paradigm for hierarchical molecular interaction modeling.

To demonstrate MCGM's generality and plug-and-play nature, we integrate it into diverse GNN architectures spanning different design philosophies—from simple continuous filters (SchNet[14]) to angular-aware models (DimeNet++[15]) and E(3)-equivariant architectures (PaiNN[4], GemNet-T[16])—showing consistent improvements regardless of the underlying architecture. Our primary comparison is with Neural P^3M [13], the current state-of-the-art for long-range molecular modeling, which uses the same set of backbones for a fair comparison. On OE62 [17], MCGM achieves 26.2% average improvement across these diverse architectures and outperforms Neural P^3M's best

3

result (DimeNet++: 41.5 meV) by reaching 38.7 meV (6.2×10^{-21}J). On AQM [18], MCGM with ViSNet establishes new state-of-the-art results (17.0 meV, 2.7×10^{-21} J for energy; 4.9 meV/Å, 7.8×10^{-12} N for forces) while using 20% fewer parameters than Neural P^{3}M. These results validate that MCGM not only seamlessly integrates with any GNN architecture but also consistently outperforms the best long-range modeling method currently.

Our main contributions are as follows:

- **Adaptive hierarchical framework for molecular modeling.** We propose MCGM, which discovers multi-scale molecular organization through dynamic clustering in learned representation spaces. Unlike existing methods with fixed hierarchies, MCGM identifies structural patterns that evolve during training, enabling adaptive long-range interaction modeling without chemical priors.

- **Consistent performance improvements across architectures.** Extensive experiments demonstrate MCGM's effectiveness on both large-scale (OE62) and mid-scale (AQM) benchmarks, establishing new state-of-the-art results while maintaining parameter efficiency. The consistent gains across diverse backbone architectures validate MCGM's general applicability.

- **Universal plug-and-play design.** MCGM integrates seamlessly with both invariant (SchNet, DimeNet++) and equivariant (PaiNN, GemNet-T, ViSNet) architectures through a unified interface. The modular design requires minimal modifications to existing pipelines, facilitating adoption across different molecular modeling frameworks.

2. Related Work

2.1. Geometric Graph Neural Networks

Geometric GNNs model molecules as graphs with spatial symmetry constraints, categorized into SE(3)-invariant and SE(3)-equivariant architectures. Invariant models [14, 7, 16] operate on scalar quantities like distances and angles, providing rotational invariance but limited receptive fields. Equivariant models [4, 19, 20, 21] maintain

directional features for accurate force prediction. While effective for local interactions within cutoff radii (5-6 Å), both approaches struggle with long-range effects in extended molecular systems where important interactions often exceed 10 Å. To address these limitations, Graph kernel methods naturally provide hierarchical decompositions that complement GNNs' local operations [22], motivating our adaptive clustering approach as an alternative to fixed kernel structures.

2.2. Long-Range Interaction Modeling

Building on these insights, various practical approaches have emerged to capture long-range interactions. Current methods can be categorized into two main strategies, each with inherent trade-offs.

Physics-informed methods incorporate explicit potentials: 4G-HDNNP [23] uses QEq charge equilibration for electrostatics; Ewald-MP [8] integrates Fourier-space summation; LSRM [6] employs chemical fragmentation with relay nodes. While achieving improvements, these methods require domain-specific knowledge and lack generality across chemical systems.

Structural methods enhance global awareness through architectural modifications: Neural-Atom [24] and Equiformer [25] enable all-pair communications but with quadratic complexity; NLA-GNN [26] employs non-local attention to bypass local message-passing limitations; Graphormer [27] uses full connectivity with learnable encodings; Neural P^3M [13] introduces FFT-compatible mesh nodes, achieving strong accuracy on benchmarks but requiring complex parameter tuning (mesh size, k-space cutoff). Recent hierarchical approaches [28, 29, 30, 31] demonstrate the value of adaptive multi-scale modeling, yet rely on fixed decomposition rules that cannot adapt to diverse molecular topologies.

Unlike existing approaches that impose rigid assumptions—whether through physical equations, chemical rules, or spatial grids—MCGM achieves adaptive hierarchical decomposition through dynamic clustering in learned representation spaces. As shown in Table 1, MCGM uniquely combines adaptivity with plug-and-play modularity, automatically discovering task-relevant patterns while maintaining computational efficiency through bounded hierarchical operations.

Table 1: Comparison of long-range modeling approaches.

Method	Physics Prior	Adaptive	Plug-and-Play
Extended Cutoff	No	No	Yes
PhysNet/SpookyNet	Yes	No	No
Ewald-MP	Yes	No	No
Neural P^3M	No	No	Yes
LSRM	No	No	No
MCGM (Ours)	No	Yes	Yes

3. Methodology

3.1. Preliminaries

3.1.1. Problem Definition

In molecular property prediction, a molecule is represented as $\mathcal{M} = \{Z, \mathbf{R}\}$, where $Z = \{z_1, \ldots, z_N\}$ denotes the atomic numbers and $\mathbf{R} = \{\mathbf{r}_1, \ldots, \mathbf{r}_N\}$ with $\mathbf{r}_i \in \mathbb{R}^3$ represents the 3D coordinates of N atoms.

Throughout this paper, we adopt units standard in molecular modeling: distances in Ångströms (1 Å= 0.1 nm = 10^{-10} m), energies in millielectron volts (1 meV = 1.602 $\times 10^{-22}$ J), and forces in meV/Å(1 meV/Å= 1.602×10^{-12} N). These units, while not strictly SI, are universally accepted in computational chemistry and molecular physics literature.

The goal is to learn a function $\mathcal{F}_\theta$ that predicts molecular properties such as potential energy $E \in \mathbb{R}$ and atomic forces $\mathbf{F} \in \mathbb{R}^{3N}$, where forces are defined as the negative gradient of energy with respect to positions. The model is trained to minimize the discrepancy between predictions and reference values computed via DFT.

3.1.2. Geometric Graph Neural Networks

Geometric GNNs model a molecule as a graph $G = (V, E)$, where nodes $v_i \in V$ represent atoms and edges $e_{ij} \in E$ encode pairwise interactions within cutoff radius d_c. Each atom i is characterized by its atomic number z_i and position $\mathbf{r}_i \in \mathbb{R}^3$.

6

The node features are initialized as $\mathbf{h}_i^{(0)} = \text{Embed}(z_i)$ and updated through L layers of message passing:

$$\mathbf{h}_i^{(\ell+1)} = \phi^{(\ell)}\left(\mathbf{h}_i^{(\ell)}, \bigoplus_{j \in \mathcal{N}(i)} \psi^{(\ell)}(\mathbf{h}_i^{(\ell)}, \mathbf{h}_j^{(\ell)}, e_{ij})\right) \tag{1}$$

where $\mathcal{N}(i)$ denotes the neighbors of atom i within cutoff d_c, $\bigoplus$ is a permutation-invariant aggregation (e.g., sum), and ϕ, ψ are learnable functions.

While effective for local interactions, this paradigm struggles with long-range effects beyond d_c. Although stacking more layers cloud theoretically expand the receptive field, this approach suffers from over-smoothing [32, 33] where node features become indistinguishable after excessive aggregation, and computational cost that scales linearly with the number of hops required. These limitations motivate our hierarchical extension that captures long-range interactions without deep stacking.

3.2. Multi-stage Clustered Global Modeling

3.2.1. Overview

MCGM augments geometric GNNs with hierarchical global context through adaptive clustering in learned representation spaces. Given an atomic graph $G^{(0)}$ with N atoms, MCGM constructs a hierarchy of progressively coarser graphs $\{G^{(1)}, ..., G^{(L)}\}$ where $|V^{(\ell+1)}| < |V^{(\ell)}|$. Information flows bidirectionally: local features are aggregated to coarser levels for global context extraction, then disseminated back to augment atomic representations. Unlike methods that rely on fixed hierarchies or physical priors, MCGM's clustering adapts dynamically during training based on learned embeddings, discovering task-relevant multi-scale patterns. The complete architecture is illustrated in Fig. 1, which demonstrates the hierarchical clustering process on a representative molecule.

3.2.2. Hierarchical Graph Construction

We construct a hierarchy of graphs $\{G^{(0)}, G^{(1)}, ..., G^{(L)}\}$ where $G^{(0)}$ is the original atomic graph with N atoms, and each successive level represents progressively coarser clusters with $|V^{(\ell+1)}| < |V^{(\ell)}|$.

(a) Example of a large molecule

Formula: C60H92N2O6Si2
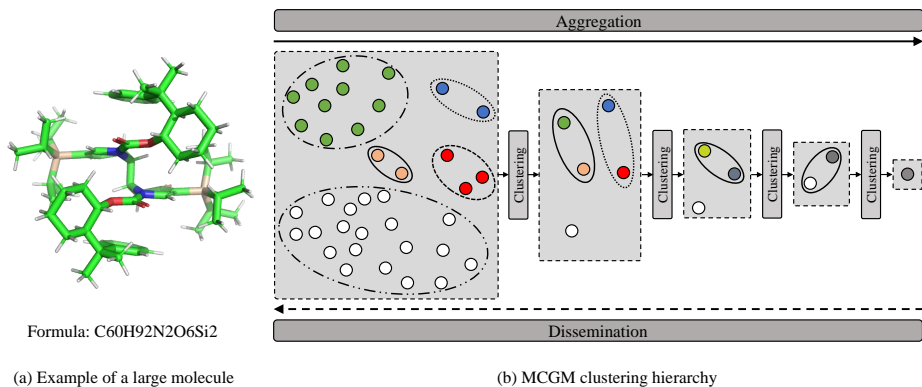
(b) MCGM clustering hierarchy

Figure 1: Overview of MCGM architecture. (a) Example molecule from the OE62 dataset illustrating the scale of structures handled by MCGM. (b) Multi-resolution clustering hierarchy. Dashed ellipses indicate clusters at each level, with nodes color-coded by cluster assignment. Grey boxes denote transitions between resolution levels.

Level 1 (Element-type clustering): Atoms are grouped by element type, providing a strong chemical prior since atoms of different elements exhibit distinct interaction behaviors. This deterministic grouping creates $|V^{(1)}|$ clusters equal to the number of unique elements in the molecule.

Level $\ell > 1$ (Adaptive clustering): We apply K-means++ clustering [34] on the learned node embeddings from level $\ell - 1$, with the number of clusters set to $|V^{(\ell)}| = \max(1, \lfloor |V^{(\ell-1)}|/r \rfloor)$ where r is a reduction ratio (typically 2). Our hierarchical module then re-clusters nodes with a non-differentiable K-means++ algorithm, re-executed at each training epoch, providing an adaptive multi-level decomposition without extra learnable parameters or dense assignment matrices. This contrasts with DiffPool [35], which learns a differentiable soft assignment matrix through an additional pooling GNN at every layer, and with Cluster-GCN [36], which relies on a single offline METIS partition purely to speed up mini-batch training rather than to learn a hierarchical representation.

Edge construction: Within each level ℓ, we connect each node to its assigned

8

cluster center, creating a star topology per cluster. This sparse connectivity pattern enables efficient information propagation between levels. The cluster centers serve as information hubs that aggregate local features and disseminate global context.

Through this hierarchical construction, the model captures interactions at multiple scales: local bonding at the atomic level, fragment-level patterns at intermediate levels, and global molecular context at the coarsest level.

3.2.3. Information Flow Architecture

We enable bidirectional information flow between hierarchical levels through two complementary operations: Aggregation (fine-to-coarse) and Dissemination (coarse-to-fine). This bidirectional exchange allows global patterns to inform local representations and vice versa.

Aggregation computes cluster features by pooling information from member nodes. For a cluster C at level ℓ with members $\{i\}$ from level $\ell - 1$, we compute:

$$\mathbf{h}_C^{(\ell)} = W_{\text{agg}}^{(\ell)} \cdot \text{AvgPool}_{i \in C} \left([\mathbf{h}_i^{(\ell-1)} \| \phi(d_{iC})] \right) \tag{2}$$

where $\|$ denotes concatenation, $d_{iC} = \|\mathbf{r}_i - \mathbf{r}_C\|$ is the Euclidean distance with cluster center position $\mathbf{r}_C = \frac{1}{|C|} \sum_{i \in C} \mathbf{r}_i$, and $\phi(\cdot)$ encodes distances using radial basis functions.

Dissemination propagates cluster information back to member nodes:

$$\tilde{\mathbf{h}}_i^{(\ell-1)} = W_{\text{dis}}^{(\ell)} \cdot [\mathbf{h}_C^{(\ell)} \| \phi(d_{iC})] \tag{3}$$

For the final dissemination from level 1 to the atomic level, we employ a residual connection to preserve local atomic features while incorporating global context:

$$\mathbf{h}_i^{(0)\text{final}} = \mathbf{h}_i^{(0)} + \tilde{\mathbf{h}}_i^{(0)} \tag{4}$$

This architecture enables efficient propagation of information across all scales, as each atom connects only to its assigned cluster center through a sparse star topology.

3.2.4. Integration with GNN Backbones

MCGM functions as a plug-and-play module that can augment any geometric GNN backbone. For invariant architectures (SchNet, DimeNet++), MCGM maintains strict

9

invariance through scalar operations. For equivariant architectures (PaiNN, GemNet-T, ViSNet), MCGM enhances the scalar feature pathway while preserving the backbone's ability to process directional information. Force predictions are obtained through automatic differentiation of the energy with respect to atomic positions, maintaining equivariance in the output. The modular design requires minimal code modification to add hierarchical context to existing message passing.

3.2.5. Energy and Force Prediction

We employ hierarchical energy decomposition where atomic nodes and cluster centers independently contribute to the total energy. Two separate MLPs decode energy contributions: one processes atomic features to capture local interactions (E_i), another processes cluster features for collective effects (E_C). The total energy is:

$$E = \sum_{i \in \text{atoms}} E_i + \sum_{C \in \text{final clusters}} E_C \tag{5}$$

This decomposition enhances interpretability by explicitly separating short-range (atomic) and long-range (cluster) contributions. Forces are obtained via automatic differentiation:

$$\mathbf{F}_i = -\frac{\partial E}{\partial \mathbf{r}_i} \tag{6}$$

ensuring energy conservation and maintaining equivariance for force predictions even when using invariant features.

4. Experiment

4.1. Experimental Setup

4.1.1. Datasets

OE62 [17] comprises approximately 62,000 organic molecules with 16 chemical elements and molecular sizes ranging from a few atoms to over 200. The dataset features molecules with spatial extents often exceeding 20 Å, making long-range interactions crucial for accurate energy prediction. Ground-truth energies are computed using DFT with the PBE functional [37]. We adopt the same preprocessing and data split as in [8].

AQM [18] contains gas-phase conformations of drug-like molecules with up to 54 heavy atoms. We use the AQM-gas subset with 59,783 conformations from 1,653 molecules. Reference calculations employ DFTB3+MBD [38], incorporating many-body dispersion to accurately model long-range interactions. We use an 80/10/10 train/validation/test split with the same preprocessing pipeline as OE62.

4.1.2. Baselines and Evaluation Metrics

We evaluate MCGM against several categories of approaches for modeling long-range interactions:

Architecture-agnostic improvements: (1) *Embeddings*: Increasing hidden dimensions to enhance model capacity without architectural changes. (2) *Extended Cutoff*: Expanding the interaction radius, which increases computational cost cubically with distance.

Specialized long-range methods: (3) *SchNet-LR*: A variant introduced in [8] that augments SchNet with a pairwise long-range block for interactions beyond the standard cutoff. (4) *Ewald MP* [8]: Incorporates Ewald summation for long-range interactions through Fourier-space calculations. (5) *Neural P³M* [13]: Introduces learnable mesh nodes that propagate global information via FFT-compatible operations. (6) *Range variants* [39]: Employ relaying attention nodes as intermediaries for global information exchange.

To demonstrate generality, we integrate MCGM with diverse GNN backbones: SchNet [14] (continuous filters), PaiNN [4] (equivariant), DimeNet++ [15] (angular-aware), GemNet-T [16] (higher-order invariants), and ViSNet [21] (lightweight equivariant). All models are evaluated using mean absolute error (MAE) on test sets. Baseline results are taken from their respective papers [8, 13, 39].

4.1.3. Implementation Details

MCGM employs K-means clustering with K-means++ initialization and a reduction ratio of 2 between hierarchical levels. Clustering is performed for 10 iterations with early stopping (tolerance 1e-4).

For OE62, we train models using L1 loss on energy predictions only. Optimization

uses AdamW [40] with learning rates of 5e-4 (SchNet), 1e-4 (PaiNN, DimeNet++), and 3e-4 (GemNet-T), with cosine annealing and warmup. Training uses early stopping (patience 50) and ReduceLROnPlateau scheduler (factor 0.8, patience 10).

For AQM, we jointly optimize energy and force predictions using weighted MSE loss with $\lambda_E = 0.01$ and $\lambda_F = 0.99$. Training uses AdamW with a learning rate of 1e-4, 10,000 warmup steps, and early stopping (patience 150).

All models use a cutoff radius of 6.0 Å for atomic interactions and 4.0 Å for cluster interactions. Batch sizes range from 8-64 for OE62 and 32 for AQM. We run all experiments with three random seeds (0, 1, 2) and report mean ± standard deviation. Other hyperparameters follow the original papers.

4.2. Main Results

4.2.1. Results on OE62

To systematically assess MCGM's impact and generality, we integrate it into four widely used GNN architectures representing diverse geometric learning strategies: continuous-filter convolution (SchNet), equivariant message passing (PaiNN), explicit angular representations (DimeNet++), and higher-order invariants (GemNet-T).

Table 2 presents the results. MCGM consistently improves all backbones, achieving an average relative improvement of 26.2% on the OE62 test set. The improvement pattern is revealing: simpler architectures benefit most from MCGM's hierarchical modeling, with SchNet showing a 50.0% improvement (131.3 to 65.6±0.8 meV), while sophisticated models like GemNet-T exhibit smaller but significant gains of 11.9% (to 46.8±0.4 meV). This suggests MCGM effectively complements existing architectural innovations. DimeNet++-MCGM achieves the best absolute performance at 38.7±0.5 meV, surpassing the previous state-of-the-art Neural P^3M variant (41.5 meV).

Compared to other long-range approaches, MCGM demonstrates consistent advantages. The extended cutoff approach, while conceptually simple, shows limited effectiveness (e.g., SchNet-Cutoff: 254.8 meV vs baseline 131.3 meV), indicating that increased computational cost does not guarantee improved accuracy. More sophisticated methods like Ewald MP (81.1 meV) and SchNet-LR (89.2 meV) achieve meaningful improvements, yet MCGM (65.6 meV) surpasses them all. Notably, although Neural

P³M slightly edges out MCGM on PaiNN (52.9 vs 53.9±0.6 meV), MCGM achieves competitive accuracy with 24% faster inference (1.64 vs 2.17 ms), offering a superior accuracy-efficiency trade-off.

Computationally, MCGM maintains efficiency across architectures. Inference overhead remains modest: 0.47 ms for SchNet (vs 0.13 baseline), 2.12 ms for DimeNet++ (vs 1.99), and 3.23 ms for GemNet-T (vs 3.07). These runtimes consistently match or outperform other long-range methods while achieving superior accuracy. The low standard deviations (≤0.8 meV) across three random seeds demonstrate the stability of MCGM's improvements.

4.2.2. Results on AQM

For the AQM dataset, we evaluate MCGM integrated with ViSNet, a lightweight equivariant architecture optimized for force prediction—a critical requirement for this benchmark. Table 3 presents the results.

ViSNet-MCGM achieves state-of-the-art performance on both tasks, with energy MAE of 17.0±0.3 meV and force MAE of 4.9±0.1 meV/Å. This represents substantial improvements over existing long-range methods: 62.7% better than SchNet-Ewald (45.6 meV) and 38.8% better than SchNet-Range (27.8 meV) for energy prediction. Compared to the previous best method PaiNN-Range, ViSNet-MCGM reduces energy MAE by 12.8% (from 19.5 to 17.0 meV) and force MAE by 36.4% (from 7.7 to 4.9 meV/Å).

The balanced performance across both energy and force predictions is particularly noteworthy. ViSNet-MCGM maintains excellent force accuracy (74.6% improvement over SchNet-Ewald) while achieving the best energy performance, demonstrating MCGM's effectiveness in capturing both local gradients and global interactions. The larger relative improvements on AQM compared to OE62 (62.7% vs 26.2% average) suggest MCGM particularly excels when many-body dispersion interactions are dominant, as explicitly modeled in AQM's DFTB3+MBD reference calculations. These results validate MCGM's hierarchical approach for learning complex long-range effects in drug-like molecules with up to 54 heavy atoms.

Table 2: Energy prediction MAE and computational cost on the OE62 dataset. Best-performing values are highlighted in **bold**. Energy units: 1 meV = 1.602×10^{-22} J. Runtime is reported in milliseconds per molecular structure. Rel. indicates the relative improvement in percentage, and Fwd + Bwd refers to the total cost of a complete forward and backward pass.

Model	Variant	OE62-val		OE62-test		Forward		Fwd + Bwd	
		MAE	Rel.	MAE	Rel.	Runtime	Rel.	Runtime	Rel.
		meV ↓	% ↑	meV ↓	% ↑	ms ↓	% ↓	ms ↓	% ↓
SchNet	Baseline	133.5	–	131.3	–	0.13	–	0.28	–
	Embeddings	144.7	-8.4	136.7	-4.1	0.14	15.2	0.33	17.8
	Cutoff	257.4	-92.8	254.8	-94.1	0.14	13.6	0.31	11.6
	SchNet-LR	86.6	35.1	89.2	32.1	0.32	156.0	0.75	171.7
	Ewald	79.2	40.7	81.1	38.2	0.70	461.6	1.03	271.4
	Neural P³M	70.2	47.4	69.1	47.4	0.37	184.6	0.57	103.6
	MCGM	**66.6±0.3**	**50.1**	**65.6±0.8**	**50.0**	0.47±0.01	261.5	1.05±0.01	275.0
PaiNN	Baseline	61.4	–	63.3	–	1.52	–	3.16	–
	Embeddings	63.5	-3.4	63.1	-0.2	1.54	1.4	3.28	3.8
	Cutoff	65.1	-6.0	64.4	-2.2	1.84	20.9	3.91	23.6
	SchNet-LR	58.3	5.1	58.2	7.7	1.84	20.7	4.21	33.1
	Ewald	57.9	5.7	59.7	5.7	2.29	50.5	4.57	44.4
	Neural P³M	**54.1**	**11.9**	**52.9**	**16.4**	2.17	42.8	4.19	32.6
	MCGM	56.3±0.5	8.3	53.9±0.6	14.8	1.64±0.01	7.9	6.95±0.19	119.9
DimeNet++	Baseline	51.2	–	53.8	–	1.99	–	4.26	–
	Embeddings	50.4	1.6	53.4	0.7	2.25	12.9	4.93	15.8
	Cutoff	48.3	5.7	48.1	10.6	2.68	34.7	6.10	43.4
	SchNet-LR	51.4	-0.5	54.4	-1.1	2.37	19.0	4.73	11.2
	Ewald	46.5	9.2	48.1	10.6	2.70	35.5	5.93	39.5
	Neural P³M	40.9	20.1	41.5	22.9	3.11	56.3	5.62	31.9
	MCGM	**40.0±0.3**	**21.9**	**38.7±0.5**	**28.1**	2.12±0.02	6.5	8.01±0.11	88.0
GemNet-T	Baseline	51.5	–	53.1	–	3.07	–	6.96	–
	Embeddings	52.7	-2.3	53.9	-1.5	3.11	1.5	6.98	0.4
	Cutoff	47.8	7.2	47.7	10.2	4.02	31.2	8.88	27.7
	SchNet-LR	51.2	0.6	52.8	0.5	3.32	8.3	7.73	11.1
	Ewald	47.4	8.0	47.5	10.5	4.05	32.0	8.86	27.4
	Neural P³M	**47.2**	**8.3**	47.4	10.7	3.93	28.0	7.71	10.8
	MCGM	48.7±0.5	5.4	**46.8±0.4**	**11.9**	3.23±0.01	5.2	8.70±0.35	25.0

Table 3: Energy MAE and Force MAE on the AQM dataset (best in **bold**). Energy units: 1 meV = 1.602 × 10^{-22} J; Forces: 1 meV/Å= 1.602×10^{-12} N.

Method	Energy		Forces	
	MAE	Rel.	MAE	Rel.
	meV ↓	% ↑	meV/Å ↓	% ↑
SchNet-Ewald	45.6	–	19.3	–
SchNet-Range	27.8	39.0	12.9	33.2
PaiNN-Ewald	23.3	48.9	8.8	54.4
PaiNN-Range	19.5	57.2	7.7	60.1
ViSNet-MCGM	**17.0**±0.3	**62.7**	**4.9**±0.1	**74.6**

4.3. Efficiency Analysis

We evaluate the computational efficiency of ViSNet-MCGM against ViSNet-Neural P^3M's architecture on AQM. Table 4 reports model size and memory consumption.

ViSNet-MCGM uses 20% fewer parameters than Neural P^3M (2.8×10^6 vs 3.5×10^6). While runtime memory reduction is modest (5.4%) due to activation storage and framework overhead dominating total memory usage, fewer parameters provide important benefits: faster convergence during training, reduced overfitting risk, and simplified deployment.

The efficiency stems from architectural streamlining. Neural P^3M requires auxiliary mesh nodes with dedicated interaction layers and parameter-heavy continuous filters. In contrast, MCGM operates directly on adaptive cluster hierarchies using lightweight linear transformations, eliminating redundant components while improving predictive performance.

These results demonstrate that effective long-range modeling can be achieved through architectural simplicity rather than complexity, making MCGM particularly suitable for resource-conscious applications.

Table 4: Model size and memory usage on AQM. Measurements on RTX 4090 with batch size 1. Best values in **bold**.

Model	Params	Train Mem. (MB)	Infer. Mem. (MB)
ViSNet-Neural P^3M	3.5×10^6	362.8	361.2
ViSNet-MCGM	$\mathbf{2.8 \times 10^6}$	**343.1**	**341.6**

4.4. Ablation Studies

We investigate the impact of clustering algorithms on MCGM's performance. While our hierarchical framework is designed to be flexible, the clustering quality affects the learned representations.

Table 5 compares different clustering methods using SchNet-MCGM on OE62. K-means++ achieves the best performance (65.6 meV), with alternatives showing degraded accuracy: spectral (71.1 meV), random (72.9 meV), and random-balanced (74.9 meV). The 5.5-9.3 meV performance gaps demonstrate that while MCGM remains functional with various clustering schemes, K-means++ provides superior hierarchical decomposition.

This performance difference likely stems from K-means++ creating spatially coherent clusters that preserve local chemical environments while enabling effective cross-scale communication. Random clustering disrupts molecular structure, hindering the model's ability to learn meaningful hierarchical representations. These results validate our choice of K-means++ as an effective balance between computational efficiency and clustering quality.

Table 5: Impact of clustering algorithms on SchNet-MCGM energy prediction. Results on OE62 test set (MAE in meV). Energy: 1 Å= 0.1 nm = 10^{-10} m.

Metric	Clustering Method			
	K-means++	Spectral	Random Balanced	Random
Energy	**65.6**	71.1	74.9	72.9

5. Conclusion

We presented MCGM, a hierarchical framework that enhances long-range interaction modeling in molecular property prediction. By introducing adaptive multi-stage clustering with star-shaped Aggregation-Dissemination, MCGM enables efficient global context modeling through sparse hierarchical connections.

Our extensive evaluation demonstrates MCGM's effectiveness and generality. On OE62, MCGM improves four diverse GNN architectures by an average of 26.2%, with DimeNet++-MCGM achieving 38.7 meV MAE—surpassing the previous state-of-the-art. On AQM, ViSNet-MCGM attains 17.0 meV for energy and 4.9 meV/Åfor forces, outperforming all baselines while using 20% fewer parameters than comparable methods. The consistent improvements across different backbones (SchNet, PaiNN, DimeNet++, GemNet-T, ViSNet) confirm MCGM's plug-and-play applicability.

MCGM's design philosophy—achieving long-range modeling through architectural simplicity rather than auxiliary structures—offers a practical path for incorporating global interactions into existing molecular simulation pipelines. Beyond molecular modeling, the hierarchical clustering framework may benefit other domains requiring multi-scale spatial modeling, such as point cloud processing and 3D shape understanding. Future work could explore learnable clustering strategies, investigate community-aware pre-training approaches [41] to enhance structural awareness, and validate MCGM's effectiveness across these diverse applications.

Appendix A. Algorithm Details

Appendix A.1. K-means Clustering

Algorithm A.1 presents the hierarchical graph clustering procedure used in MCGM.

Appendix A.2. Aggregation and Dissemination Units

Algorithms A.2 and A.3 detail the aggregation and dissemination procedures.

Algorithm A.1: Hierarchical Graph Clustering

Input: Node features $X \in \mathbb{R}^{N \times d}$; batch index b;

reduction ratio $r = 2$; tolerance $\varepsilon = 10^{-4}$; max iterations $T = 10$.

Output: Cluster assignments $c \in \{0, \ldots, K-1\}^N$.

1   $B \leftarrow$ number of graphs in batch

2   **foreach** *graph g in batch* **do**

3      $n_g \leftarrow$ number of nodes in graph g

4      $k_g \leftarrow \lceil n_g/r \rceil$ // Clusters for this graph

5   Initialize centroids C_g using K-means++ for graph g

6   **end**

7   **for** $t = 1$ **to** T **do**

8      $C_{\text{old}} \leftarrow C$

9   **foreach** *node i* **do**

10   Assign i to nearest centroid in its graph

11   **end**

12   Update centroids as cluster means

13   **if** *any cluster is empty* **then**

14   Reinitialize empty centroid randomly

15   **end**

16   **if** *all clusters have single node* **then**

17   **return** c // Early stop

18   **end**

19   **if** $\|C - C_{old}\| \leq \varepsilon$ **then**

20   **return** c

21   **end**

22   **end**

23   **return** c

Algorithm A.2: Aggregation Unit

Input: Node features $F_f \in \mathbb{R}^{N \times d}$; positions $P_f \in \mathbb{R}^{N \times 3}$;

cluster assignments $c \in \{0, \ldots, K-1\}^N$; total clusters K;

cutoff radius γ; number of RBFs N_R.

Output: Cluster features $F_c \in \mathbb{R}^{K \times d'}$; centroids $P_c \in \mathbb{R}^{K \times 3}$.

```
// Compute cluster centroids
```

1 **for** $k = 0$ **to** $K - 1$ **do**

2      $\mathcal{I}_k \leftarrow \{i : c_i = k\}$

3   **if** $\mathcal{I}_k = \varnothing$ **then continue**

4      $P_{c,k} \leftarrow \frac{1}{|\mathcal{I}_k|} \sum_{i \in \mathcal{I}_k} P_{f,i}$

5 **end**

```
// Aggregate features to clusters
```

6 **for** $k = 0$ **to** $K - 1$ **do**

7      $\mathcal{I}_k \leftarrow \{i : c_i = k\}$

8   **if** $\mathcal{I}_k \neq \varnothing$ **then**

9   **foreach** $i \in \mathcal{I}_k$ **do**

10             $r_i \leftarrow \|P_{f,i} - P_{c,k}\|_2$

11             $e_i \leftarrow \text{RBF}(r_i; \gamma, N_R)$

12             $z_i \leftarrow [F_{f,i} \,\|\, e_i]$

13   **end**

14          $F_{c,k} \leftarrow W_{\text{agg}} \cdot \frac{1}{|\mathcal{I}_k|} \sum_{i \in \mathcal{I}_k} z_i + b$

15   **end**

16 **end**

17 **return** (F_c, P_c)

Algorithm A.3: Dissemination Unit

Input: Cluster features $F_c \in \mathbb{R}^{K \times d}$; cluster positions $P_c \in \mathbb{R}^{K \times 3}$;
node positions $P_f \in \mathbb{R}^{N \times 3}$; cluster assignments $c \in \{0, \ldots, K-1\}^N$;
cutoff radius γ; number of RBFs N_R.

Output: Updated node features $F'_f \in \mathbb{R}^{N \times d'}$.

```
// Disseminate cluster information to nodes
```

1 **for** $i = 0$ **to** $N-1$ **do**
2      $k \leftarrow c_i$ `// Get cluster index for node i`
3      $r_i \leftarrow \|P_{f,i} - P_{c,k}\|_2$
4      $e_i \leftarrow \text{RBF}(r_i; \gamma, N_R)$
5      $z_i \leftarrow [F_{c,k} \| e_i]$
6      $F'_{f,i} \leftarrow W_{\text{dis}} \cdot z_i + b$
7 **end**
8 **return** F'_f

Appendix B. Integration with GNN Backbones

MCGM seamlessly integrates with existing GNN architectures by introducing hierarchical clustering and cross-scale communication modules. The integration preserves the original architecture's properties while enabling long-range interactions.

Appendix B.1. SchNet

SchNet uses continuous-filter convolutions with radial basis functions. MCGM integrates after each SchNet layer, where atomic features $h^{(l)}$ and cluster features $h_C^{(l)}$ interact through aggregation-dissemination:

$$h^{(l+1)} = h^{(l)} + \Delta h_{\text{SchNet}}^{(l)} + W_{\text{dis}} \cdot [h_C^{(l)} \| e_{Ci}] \tag{B.1}$$

$$h_C^{(l+1)} = h_C^{(l)} + W_{\text{agg}} \cdot \text{mean}_{i \in C}([h_i^{(l)} \| e_{iC}]) \tag{B.2}$$

where $e_{iC} = \text{RBF}(\|\mathbf{r}_i - \mathbf{r}_C\|)$ encodes distances between atoms and cluster centers.

Appendix B.2. PaiNN

PaiNN maintains scalar (h) and vector ($\vec{v}$) features for equivariance. MCGM operates only on scalar features to preserve rotational equivariance:

$$h^{(l+1)}, \vec{v}^{(l+1)} = \text{PaiNN}(h^{(l)}, \vec{v}^{(l)}) \tag{B.3}$$

$$h^{(l+1)} = h^{(l+1)} + W_{\text{dis}} \cdot [h_C^{(l)} \| e_{Ci}] \tag{B.4}$$

Vector features remain within their resolution level, while scalar features carry long-range information.

Appendix B.3. DimeNet++

DimeNet++ uses directional message passing with angular information. MCGM integrates at the node level after message aggregation:

$$h^{(l+1)} = h^{(l)} + \Delta h_{\text{DimeNet}}^{(l)} + W_{\text{dis}} \cdot [h_C^{(l)} \| e_{Ci}] \tag{B.5}$$

$$h_C^{(l+1)} = h_C^{(l)} + W_{\text{agg}} \cdot \text{mean}_{i \in C}([h_i^{(l)} \| e_{iC}]) \tag{B.6}$$

where $\Delta h_{\text{DimeNet}}^{(l)}$ includes the directional message passing with angular features. Edge features in DimeNet++ are updated internally within the model and do not directly interact with MCGM.

Appendix B.4. GemNet-T and ViSNet

Both GemNet-T and ViSNet employ sophisticated geometric features that require special handling to preserve their properties.

GemNet-T uses triplet-based message passing with angular information. MCGM operates on node embeddings while preserving edge-level angular features:

$$h^{(l+1)}, m^{(l+1)} = \text{GemNet-T}(h^{(l)}, m^{(l)}) \tag{B.7}$$

$$h^{(l+1)} = h^{(l+1)} + W_{\text{dis}} \cdot [h_C^{(l)} \| e_{Ci}] \tag{B.8}$$

ViSNet combines scalar and vector features with directional units. Similar to PaiNN, MCGM acts only on scalar features to maintain equivariance:

$$h^{(l+1)}, \vec{v}^{(l+1)}, f^{(l+1)} = \text{ViSNet}(h^{(l)}, \vec{v}^{(l)}, f^{(l)}) \tag{B.9}$$

$$h^{(l+1)} = h^{(l+1)} + W_{\text{dis}} \cdot [h_C^{(l)} \| e_{Ci}] \tag{B.10}$$

where $f^{(l)}$ represents edge features with angular information.

Table B.1: Key hyperparameters for MCGM experiments. Distance unit: 1 Å = 0.1 nm.

Parameter	OE62	AQM
Clustering iterations	10	10
Reduction ratio	2	2
Atom cutoff (Å)	6.0	4.0
Cluster cutoff (Å)	4.0	4.0
Learning rate	1e-4 to 5e-4	1e-4
Batch size	8-64	32
Early stopping patience	50	150
Loss function	L1	MSE (λ_E=0.01, λ_F=0.99)

Appendix B.5. Key Hyperparameters

The key hyperparameters used in our experiments are summarized in Table B.1.

References

[1] P. Reiser, M. Neubert, A. Eberhard, L. Torresi, C. Zhou, C. Shao, H. Metni, C. van Hoesel, H. Schopmans, T. Sommer, et al., Graph neural networks for materials science and chemistry, Communications Materials 3 (1) (2022) 93.

[2] S. Batzner, A. Musaelian, L. Sun, M. Geiger, J. P. Mailoa, M. Kornbluth, N. Molinari, T. E. Smidt, B. Kozinsky, E (3)-equivariant graph neural networks for data-efficient and accurate interatomic potentials, Nature communications 13 (1) (2022) 2453.

[3] F. Ekström Kelvinius, D. Georgiev, A. Toshev, J. Gasteiger, Accelerating molecular graph neural networks via knowledge distillation, Advances in Neural Information Processing Systems 36 (2023) 25761–25792.

[4] K. Schütt, O. Unke, M. Gastegger, Equivariant message passing for the prediction of tensorial properties and molecular spectra, in: International Conference on Machine Learning, PMLR, 2021, pp. 9377–9388.

[5] W. Kohn, L. J. Sham, Self-consistent equations including exchange and correlation effects, Physical review 140 (4A) (1965) A1133.

[6] Y. Li, Y. Wang, L. Huang, H. Yang, X. Wei, J. Zhang, T. Wang, Z. Wang, B. Shao, T.-Y. Liu, Long-short-range message-passing: A physics-informed framework to capture non-local interaction for scalable molecular dynamics simulation, in: The Twelfth International Conference on Learning Representations, 2024, p. rvDQt-dMnOl.
URL https://openreview.net/forum?id=rvDQtdMnOl

[7] J. Gasteiger, J. Groß, S. Günnemann, Directional message passing for molecular graphs, arXiv preprint arXiv:2003.03123 (2020).

[8] A. Kosmala, J. Gasteiger, N. Gao, S. Günnemann, Ewald-based long-range message passing for molecular graphs, in: International Conference on Machine Learning, PMLR, 2023, pp. 17544–17563.

[9] B. Cheng, Latent ewald summation for machine learning of long-range interactions, npj Computational Materials 11 (1) (2025) 80.

[10] C. G. Staacke, H. H. Heenen, C. Scheurer, G. Csányi, K. Reuter, J. T. Margraf, On the role of long-range electrostatics in machine-learned interatomic potentials for complex battery materials, ACS Applied Energy Materials 4 (11) (2021) 12562–12569.

[11] O. T. Unke, M. Meuwly, Physnet: A neural network for predicting energies, forces, dipole moments, and partial charges, Journal of chemical theory and computation 15 (6) (2019) 3678–3693.

[12] O. T. Unke, S. Chmiela, M. Gastegger, K. T. Schütt, H. E. Sauceda, K.-R. Müller, Spookynet: Learning force fields with electronic degrees of freedom and nonlocal effects, Nature communications 12 (1) (2021) 7273.

[13] Y. Wang, C. Cheng, S. Li, Y. Ren, B. Shao, G. Liu, P.-A. Heng, N. Zheng, Neural p^3m: A long-range interaction modeling enhancer for geometric gnns, Advances in Neural Information Processing Systems 37 (2024) 120336–120365.

[14] K. Schütt, P.-J. Kindermans, H. E. Sauceda Felix, S. Chmiela, A. Tkatchenko, K.-R. Müller, Schnet: A continuous-filter convolutional neural network for modeling quantum interactions, Advances in neural information processing systems 30 (2017).

[15] J. Gasteiger, S. Giri, J. T. Margraf, S. Günnemann, Fast and uncertainty-aware directional message passing for non-equilibrium molecules, arXiv preprint arXiv:2011.14115 (2020).

[16] J. Gasteiger, F. Becker, S. Günnemann, Gemnet: Universal directional graph neural networks for molecules, Advances in Neural Information Processing Systems 34 (2021) 6790–6802.

[17] A. Stuke, C. Kunkel, D. Golze, M. Todorović, J. T. Margraf, K. Reuter, P. Rinke, H. Oberhofer, Atomic structures and orbital energies of 61,489 crystal-forming organic molecules, Scientific data 7 (1) (2020) 58.

[18] L. Medrano Sandonas, D. Van Rompaey, A. Fallani, M. Hilfiker, D. Hahn, L. Perez-Benito, J. Verhoeven, G. Tresadern, J. Kurt Wegner, H. Ceulemans, et al., Dataset for quantum-mechanical exploration of conformers and solvent effects in large drug-like molecules, Scientific Data 11 (1) (2024) 742.

[19] V. G. Satorras, E. Hoogeboom, M. Welling, E (n) equivariant graph neural networks, in: International conference on machine learning, PMLR, 2021, pp. 9323–9332.

[20] A. Musaelian, S. Batzner, A. Johansson, L. Sun, C. J. Owen, M. Kornbluth, B. Kozinsky, Learning local equivariant representations for large-scale atomistic dynamics, Nature Communications 14 (1) (2023) 579.

[21] Y. Wang, T. Wang, S. Li, X. He, M. Li, Z. Wang, N. Zheng, B. Shao, T.-Y. Liu, Enhancing geometric representations for molecules with equivariant vector-scalar interactive message passing, Nature Communications 15 (1) (2024) 313.

[22] L. Xu, J. Peng, X. Jiang, E. Chen, B. Luo, Graph neural network based on graph kernel: A survey, Pattern Recognition 161 (2025) 111307.

[23] T. W. Ko, J. A. Finkler, S. Goedecker, J. Behler, A fourth-generation high-dimensional neural network potential with accurate electrostatics including non-local charge transfer, Nature communications 12 (1) (2021) 398.

[24] X. Li, Z. Zhou, J. Yao, Y. Rong, L. Zhang, B. Han, Neural atoms: Propagating long-range interaction in molecular graphs through efficient communication channel, arXiv preprint arXiv:2311.01276 (2023).

[25] Y.-L. Liao, T. Smidt, Equiformer: Equivariant graph attention transformer for 3d atomistic graphs, arXiv preprint arXiv:2206.11990 (2022).

[26] S. Wang, G. Cao, W. Cao, Y. Li, Nla-gnn: Non-local information aggregated graph neural network for heterogeneous graph embedding, Pattern Recognition 158 (2025) 110940.

[27] C. Ying, T. Cai, S. Luo, S. Zheng, G. Ke, D. He, Y. Shen, T.-Y. Liu, Do transformers really perform badly for graph representation?, Advances in neural information processing systems 34 (2021) 28877–28888.

[28] G. Lin, W. Wei, X. Kang, K. Liao, E. Zhang, Deep graph layer information mining convolutional network, Pattern Recognition 154 (2024) 110593.

[29] W. Chen, W. Yan, W. Wang, Adaptive propagation deep graph neural networks, Pattern Recognition 154 (2024) 110607.

[30] G. Ai, H. Yan, H. Wang, X. Li, A2gcn: Graph convolutional networks with adaptive frequency and arbitrary order, Pattern Recognition 156 (2024) 110764.

[31] Q. Guo, X. Yang, M. Li, Y. Qian, Collaborative graph neural networks for augmented graphs: A local-to-global perspective, Pattern Recognition 158 (2025) 111020.

[32] Q. Li, Z. Han, X.-M. Wu, Deeper insights into graph convolutional networks for semi-supervised learning, Proceedings of the AAAI Conference on Artificial Intelligence 32 (01 2018). doi:10.1609/aaai.v32i1.11604.

[33] K. Oono, T. Suzuki, Graph neural networks exponentially lose expressive power for node classification, in: International Conference on Learning Representations, 2020, p. S1ldO2EFPr.
URL `https://openreview.net/forum?id=S1ldO2EFPr`

[34] D. Arthur, S. Vassilvitskii, k-means++: the advantages of careful seeding, in: Proceedings of the Eighteenth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA '07, Society for Industrial and Applied Mathematics, USA, 2007, p. 1027–1035.

[35] Z. Ying, J. You, C. Morris, X. Ren, W. Hamilton, J. Leskovec, Hierarchical graph representation learning with differentiable pooling, Advances in neural information processing systems 31 (2018).

[36] W.-L. Chiang, X. Liu, S. Si, Y. Li, S. Bengio, C.-J. Hsieh, Cluster-gcn: An efficient algorithm for training deep and large graph convolutional networks, in: Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery & data mining, 2019, pp. 257–266.

[37] J. P. Perdew, K. Burke, M. Ernzerhof, Generalized gradient approximation made simple, Physical review letters 77 (18) (1996) 3865.

[38] M. Mortazavi, J. G. Brandenburg, R. J. Maurer, A. Tkatchenko, Structure and stability of molecular crystals with many-body dispersion-inclusive density functional tight binding, The journal of physical chemistry letters 9 (2) (2018) 399–405.

[39] A. Caruso, J. Venturin, L. Giambagli, E. Rolando, F. Noé, C. Clementi, Extending the range of graph neural networks: Relaying attention nodes for global encoding, arXiv preprint arXiv:2502.13797 (2025).

[40] I. Loshchilov, F. Hutter, Decoupled weight decay regularization, in: International Conference on Learning Representations, 2019, p. Bkg6RiCqY7.
URL `https://openreview.net/forum?id=Bkg6RiCqY7`

[41] Z. Huang, W. Zhou, Y. Jiang, Z. Jia, L. Lü, Y. Ma, An efficient community-aware pre-training method for graph neural networks, Pattern Recognition (2025) 112340.