
DualEquiNet: A Dual-Space Hierarchical Equivariant Network for Large Biomolecules

Junjie Xu¹, Jiahao Zhang¹, Mangal Prakash², Xiang Zhang¹, Suhang Wang¹

¹The Pennsylvania State University

²Independent Researcher

{junjiexu, jiahao.zhang, xzz89, szw494}@psu.edu, mangalcrj@gmail.com

Abstract

Geometric graph neural networks (GNNs) that respect $E(3)$ symmetries have achieved strong performance on small molecule modeling, but they face scalability and expressiveness challenges when applied to large biomolecules such as RNA and proteins. These systems require models that can simultaneously capture fine-grained atomic interactions, long-range dependencies across spatially distant components, and biologically relevant hierarchical structure—such as atoms forming residues, which in turn form higher-order domains. Existing geometric GNNs, which typically operate exclusively in either Euclidean or Spherical Harmonics space, are limited in their ability to capture both the fine-scale atomic details and the long-range, symmetry-aware dependencies required for modeling the multi-scale structure of large biomolecules. We introduce DualEquiNet, a Dual-Space Hierarchical Equivariant Network that constructs complementary representations in both Euclidean and Spherical Harmonics spaces to capture local geometry and global symmetry-aware features. DualEquiNet employs bidirectional cross-space message passing and a novel Cross-Space Interaction Pooling mechanism to hierarchically aggregate atomic features into biologically meaningful units, such as residues, enabling efficient and expressive multi-scale modeling for large biomolecular systems. DualEquiNet achieves state-of-the-art performance on multiple existing benchmarks for RNA property prediction and protein modeling, and outperforms prior methods on two newly introduced 3D structural benchmarks demonstrating its broad effectiveness across a range of large biomolecule modeling tasks. The source code is available at <https://github.com/junjie-xu/dualequinnet>.

1 Introduction

Large biomolecules such as RNA and proteins play central roles in biological systems, enabling key functions ranging from enzymatic catalysis to molecular recognition and gene regulation [15, 41, 10, 61, 38, 7]. Recent studies have explored diverse strategies to improve protein and RNA modeling, from geometric deep learning to physics and biology-informed neural architectures [45, 82, 81, 56, 87, 83, 72, 59, 55]. These molecules exist at the atomistic scale, where complex three-dimensional geometries, hierarchical organization (with atoms grouped into residues and domains), and long-range interactions jointly determine function. Crucially, molecular models must respect the underlying physical symmetries: predicted properties should transform consistently under global rotations and translations, remaining invariant for scalar quantities (such as energy) or equivariant for directional quantities (such as dipole moments). Designing computational models that can faithfully capture these geometric and symmetry-aware aspects is essential for accurate biomolecular modeling [36, 71, 5].

Despite recent advances in geometric deep learning, current geometric graph neural networks (GNNs) face two critical limitations when applied to large biomolecular systems. *First*, most existing

geometric GNNs model molecules purely at the atomic level, representing atoms as nodes and bonds or spatial proximities as edges [80, 62, 23, 35]. While effective for small molecules, this atomistic-only approach fails to capture the inherent hierarchical organization in large biomolecules, where atoms are naturally grouped into higher-order units such as nucleotides (in RNA) or amino acids (in proteins). Ignoring this hierarchical structure leads to the loss of essential contextual and relational information, ultimately constraining predictive performance. *Second*, large biomolecules exhibit rich long-range dependencies: distant residues or nucleotides can interact and cooperatively determine global properties [22]. Most existing geometric GNNs focus on local neighborhoods in Euclidean space [62, 63, 12, 16, 6], limiting their ability to capture nonlocal interactions effectively. While some methods incorporate higher-order geometric features, such as using higher-degree spherical harmonics, they often suffer from significant computational overhead [69, 13, 27], making them impractical for large-scale molecular systems.

To address the challenges of hierarchical structure and long-range dependencies in large biomolecules, we propose DualEquiNet, a novel Dual-Space Hierarchical Equivariant Network. Unlike existing methods which either use nonlocal attention using spherical harmonics [26] in Euclidean-space neighborhoods or using high-degree steerable representations within a single geometric space [17, 6], DualEquiNet jointly constructs complementary representations in both Euclidean (EU) and spherical harmonics (SH) spaces: the EU space captures local geometric interactions crucial for atomic-level detail, while the SH space encodes rotation-equivariant features well-suited for long-range structural dependencies. Through bidirectional cross-space message passing, DualEquiNet enables structured information exchange between EU and SH spaces, integrating local and global geometric contexts. Moreover, we introduce a novel Cross-Space Interaction Pooling (CSIP) mechanism that hierarchically aggregates atomic-level features into coarser scales (such as residue-level), expanding the receptive field without requiring deep stacking of atomic layers. By explicitly facilitating interaction between EU and SH representations during pooling, CSIP ensures efficient and expressive multi-scale modeling. Together, these innovations yield a scalable, symmetry-aware framework tailored to the unique architectural and functional demands of large biomolecular systems.

We evaluate DualEquiNet across established RNA and protein property prediction datasets, demonstrating consistent improvements over existing geometric baselines (with average performance gains of 8.4%–33.5% across tasks). To further probe the model’s ability to capture nuanced 3D structural features, we developed two new benchmark datasets for Solvent-Accessible Surface Area (SASA) Prediction and Torsion Angle Prediction, which are crucial for applications in drug discovery and protein engineering [66, 68]. DualEquiNet achieves improvements of 3.1%–28.8% and 2.5%, respectively, over the strongest prior methods on these benchmarks as well. Code and datasets will be released publicly upon paper acceptance.

To sum up, our **main contributions** are: (i) We introduce DualEquiNet, a novel dual-space hierarchical equivariant network that explicitly constructs complementary Euclidean and spherical harmonics (SH) representations, enabling the simultaneous capture of local atomic interactions and long-range structural dependencies; (ii) We propose bidirectional cross-space message passing and a novel Cross-Space Interaction Pooling (CSIP) mechanism, which together allow efficient and expressive hierarchical aggregation from atoms to residues, tailored to large biomolecular structures; and (iii) We introduce two new 3D structural benchmark datasets (SASA and Torsion Angle prediction) and evaluate DualEquiNet across these and other diverse established biomolecular tasks, including RNA and protein property prediction, consistently achieving state-of-the-art results and outperforming prior baselines.

2 Related Works

Invariant GNNs. Rotation-invariant GNNs leverage geometric features such as interatomic distances and angles to produce representations invariant under rigid body motions. SchNet [63] uses continuous-filter convolutions over pairwise distances. DimeNet [30] and GemNet [28] further incorporate angular information, improving directional sensitivity. SphereNet [50] expands the invariant feature set to include distances, angles, and torsions. While effective, such models discard essential orientation-dependent information, motivating the need for equivariant approaches.

Scalarization Equivariant GNNs. Scalarization-based equivariant GNNs aim to preserve $E(3)$ equivariance by decoupling scalar and vector representations. EGNN [62] uses pairwise distances

to compute scalar messages for equivariant updates of coordinates. PaiNN [64] and GVPGNN [39] similarly apply scalar gating and vector decoupling to maintain equivariance efficiently. However, their reliance on scalar guidance can limit expressivity, particularly for modeling complex rotational dependencies [40].

Higher-order Equivariant GNNs. Tensor-product-based methods achieve full SE(3) or SO(3) equivariance via Clebsch–Gordan (CG) tensor algebra. TFN [69] and SE(3)-Transformer [27] construct equivariant convolutions over irreducible representations, enabling expressive geometric reasoning. MACE [9] utilizes a complete basis from the Atomic Cluster Expansion for maximal expressivity. However, these methods often suffer from high computational cost due to CG operations scaling as $\mathcal{O}(L^6)$ with the degree L . Recent methods aim to balance expressivity and efficiency by avoiding explicit CG products. HEGNN [16] and GotenNet [6] incorporate high-degree spherical harmonics but use inner-product-based scalarization to model cross-degree interactions.

Long-range Dependencies. Modeling long-range interactions remains a core challenge in GNNs due to the tension between limited receptive fields and over-smoothing [18, 58]. Recent benchmarks [24, 70, 86] evaluate GNN performance on long-range tasks. Solutions include implicit models (e.g., IGNNs) and Graph Transformers, which relax locality constraints via attention-based architectures. These efforts are complementary to equivariant modeling and may be integrated with geometric priors for improved scalability and fidelity.

A more comprehensive discussion of related work, including technical distinctions from our work, is provided in Appendix A.

3 Methodology

Notations. Each molecule can be represented as a 3D geometric graph, $\mathcal{G} = \{H, X\}$. H contains node features across hierarchical scales, specifically atom-level (e.g. atomic number) and residue-level (e.g. nucleotide type in RNA), denoted by $H = \{H_{atom} \in \mathbb{R}^{N_{atom} \times F_{atom}}, H_{res} \in \mathbb{R}^{N_{res} \times F_{res}}\}$, where N_{atom} and N_{res} are the number of atoms and residues (e.g. the nucleotides in RNA and amino acids in proteins), F_{atom} and F_{res} are the dimensions of atom and residue features. $h_{atom} \in \mathbb{R}^{F_{atom}}$ and $h_{res} \in \mathbb{R}^{F_{res}}$ are the corresponding features of a node. $X \in \mathbb{R}^{N_{atom} \times 3}$ denotes the 3D coordinates of atoms. From initial $\mathcal{G}$, we aim to construct Euclidean (EU) space neighbors $\mathcal{N}_{EU}$ and spherical harmonics (SH) space neighbors $\mathcal{N}_{SH}$. We use uppercase letters to denote matrices and lowercase symbols to represent the corresponding vectors for a given node (e.g. h_i and x_i for node i). To distinguish between invariant and equivariant features, we use regular letters for invariant features (e.g., h) and bold symbols for equivariant features (e.g., $\mathbf{x}$).

Overview of DualEquiNet. Fig. 1(a) gives an illustration of the proposed DualEquiNet. It begins with SH feature initialization, followed by multiple DualEqui layers applied at the atomic level. A Cross-Space Interaction Pooling (CSIP) module then aggregates atomic representations into residue-level features, which are further refined by additional DualEqui layers. Each DualEqui layer first infers EU and SH neighborhoods, then performs both inner-space and cross-space message passing to update the EU coordinates $\mathbf{x}$, invariant features h , and SH features $\mathbf{r}$ (See Fig. 1(b)). EU neighbors are determined based on a distance cutoff, capturing local geometry. SH neighbors are selected based on SH feature similarity, enabling long-range structural connections beyond spatial proximity (See Fig. 1(d)). CSIP aggregates atom-level features within each residue cluster. It performs pooling for h , $\mathbf{x}$, and $\mathbf{r}$, and incorporates cross-space projections to exchange complementary information across spaces (See Fig. 1(c)). Next, we give details of each component.

3.1 Initialization of Dual-Space Features

Most equivariant graph neural networks rely on EU distance between nodes alone to define edges, which struggles to capture long-range dependencies. This often requires increasing distance thresholds or adding more layers, both of which increase computational complexity. Moreover, EU-based message passing assumes that neighboring nodes are close in EU space and share similar features, but this assumption doesn't hold for large biomolecules like proteins and RNA. Nodes may be distant in EU space yet share similar structural features, such as bond angles and torsion, which are better captured in SH space. Thus, relying solely on EU proximity may miss key structural similarities crucial for molecular representation.

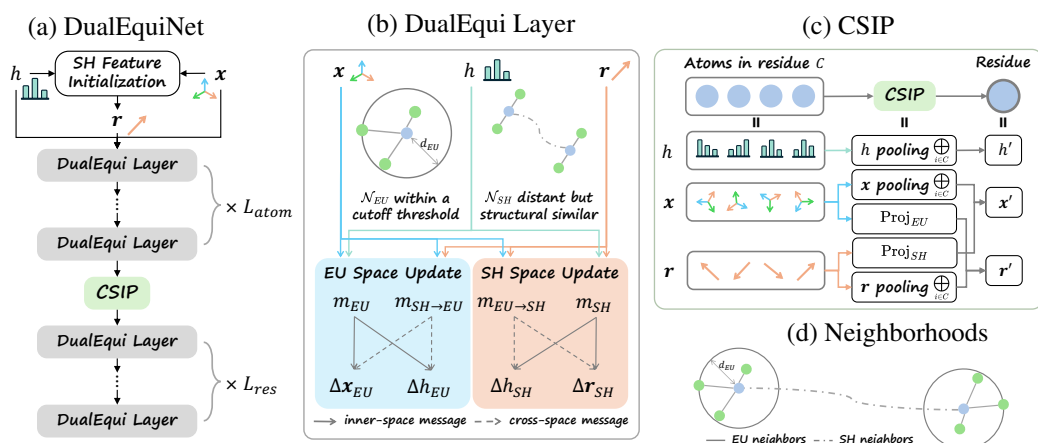


Figure 1: Overview of the proposed DualEquiNet. **(a) DualEquiNet:** It begins with SH feature initialization, followed by multiple DualEqui layers applied at the atomic level. A CSIP module then aggregates atomic representations into residue-level features, which are further refined by additional DualEqui layers. **(b) DualEqui Layer:** Each layer first infers EU and SH neighborhoods, then performs both inner-space and cross-space message passing to update the Euclidean coordinates $\mathbf{x}$, scalar features h , and SH features $\mathbf{r}$. **(c) CSIP:** Cross-Space Interaction Pooling aggregates atom-level features within each residue cluster. It performs pooling for h , $\mathbf{x}$, and $\mathbf{r}$, and incorporates cross-space projections to exchange complementary information across spaces. **(d) Neighborhoods:** EU neighbors are determined based on a distance cutoff, capturing local geometry. SH neighbors are selected based on SH feature similarity, enabling long-range structural connections beyond spatial proximity.

To address these issues, we propose a dual-space initialization strategy that constructs two complementary spaces: EU space for capturing local geometric interactions and SH space for global structural relationships. The EU neighborhood of node i is defined as:

$$\mathcal{N}_{EU}(i) = \{j \mid \|\mathbf{x}_i - \mathbf{x}_j\| \leq d_{EU}\}, \quad (1)$$

where d_{EU} is the Euclidean distance threshold. Next, we initialize the SH space features by aggregating the EU neighbors as follows. For each SH degree $l = 0, 1, \dots, l_{\max}$, the l -th order SH feature for node i is defined as:

$$\mathbf{r}_i^{(l)} = \frac{1}{|\mathcal{N}_{EU}(i)|} \sum_{j \in \mathcal{N}_{EU}(i)} \phi_0([h_i, h_j, \|\mathbf{x}_{ij}\|]) \hat{Y}^{(l)}(\hat{\mathbf{x}}_{ij}), \quad (2)$$

where $\mathbf{x}_{ij} = \mathbf{x}_i - \mathbf{x}_j$ is the direction vector, and $\hat{\mathbf{x}}_{ij} = \mathbf{x}_{ij} / \|\mathbf{x}_{ij}\|$ is the unit direction vector, ϕ_0 is a differentiable message function (e.g., an MLP), and $\hat{Y}^{(l)} \in \mathbb{R}^{2l+1}$ is the normalized SH function of degree l , with unit norm $\|\hat{Y}^{(l)}(\cdot)\| = 1$. The SH basis and corresponding node features up to degree $l_{\max}$ are jointly represented as:

$$\hat{Y}(\hat{\mathbf{x}}_{ij}) = [\hat{Y}^{(0)}(\hat{\mathbf{x}}_{ij}), \hat{Y}^{(1)}(\hat{\mathbf{x}}_{ij}), \dots, \hat{Y}^{(l_{\max})}(\hat{\mathbf{x}}_{ij})], \text{ and } \mathbf{r}_i = [\mathbf{r}_i^{(0)}, \mathbf{r}_i^{(1)}, \dots, \mathbf{r}_i^{(l_{\max})}]. \quad (3)$$

The details of spherical harmonics are introduced in Appendix B. Eq. 2 aims to learn an initialization of the SH space features based on the EU neighbors of the node, after which the SH features are further refined through updates in the SH space. With the initial SH vectors, we define the SH neighborhoods of node i based on the cosine similarity as

$$\mathcal{N}_{SH}(i) = \left\{j \mid \frac{\mathbf{r}_i \cdot \mathbf{r}_j}{\|\mathbf{r}_i\| \cdot \|\mathbf{r}_j\|} \geq d_{SH}\right\}, \quad (4)$$

where d_{SH} is a threshold for SH space. These two neighborhoods—EU and SH—form the basis for equivariant dual-space message passing and cross-space interaction in subsequent model stages. In this way, the SH neighborhood overcomes the distance limitations of EU space and captures relationships between nodes that are distant in EU space.

3.2 DualEqui Layer

The DualEqui Layer integrates geometric information from both EU and SH spaces through a bidirectional message passing scheme. Each layer updates invariant features h , EU coordinates $\mathbf{x}$, and

SH features $\mathbf{r}$ using a combination of within-space and cross-space messages. This design enables the model to simultaneously preserve local geometry and capture global structural patterns.

Euclidean Space Update. In EU space, both the atomic positions $\mathbf{x}$ and invariant features $\mathbf{h}$ are updated using information from two sources: (1) within-space interactions based on EU neighbors, and (2) cross-space interactions from the SH neighbors. Specifically, we compute messages from both the EU and SH neighbors and aggregate them via a multi-head attention mechanism.

Within-space message m_{EU} captures local geometric interactions in the EU space using invariant features and distances, while cross-space message $m_{SH \rightarrow EU}$ models the influence of SH-space structural similarity on the EU update, enabling long-range information flow guided by symmetry-aware features. For each node i and its neighbor j , we have

$$m_{EU,ij} = \phi_{EU}([h_i, h_j, \|\mathbf{x}_{ij}\|]), \quad m_{SH \rightarrow EU,ij} = \psi_{SH \rightarrow EU}([\mathbf{r}_i \odot \mathbf{r}_j, \|\mathbf{x}_{ij}\|]), \quad (5)$$

where $\|\mathbf{x}_{ij}\|$ denotes the Euclidean distance between nodes i and j . The function ϕ_{EU} denotes differentiable functions applied within EU space (e.g., an MLP), and $\psi_{SH \rightarrow EU}$ denotes cross-space transformations. The operator $\odot$ computes degree-wise inner products of SH features to form an invariant representation [17]:

$$\mathbf{r}_i \odot \mathbf{r}_j = [\mathbf{r}_i^{(0)} \cdot \mathbf{r}_j^{(0)}, \mathbf{r}_i^{(1)} \cdot \mathbf{r}_j^{(1)}, \dots, \mathbf{r}_i^{(l_{\max})} \cdot \mathbf{r}_j^{(l_{\max})}]. \quad (6)$$

To combine the messages, we use K -head attention. Each head learns to attend to different aspects of the neighborhood, both in the EU and SH spaces. For the k -th head, the attention weights are computed as:

$$\alpha_{EU,ij}^k = \sigma(\phi_{EU,att}^k(m_{EU,ij})), \quad \beta_{EU,ij}^k = \sigma(\psi_{EU,\beta}^k(m_{SH \rightarrow EU,ij})), \quad (7)$$

where σ is the sigmoid function, and $\phi_{EU,att}^k, \psi_{EU,\beta}^k$ are attention blocks. The invariant feature h_i is updated using a permutation-invariant aggregation $\oplus$ (e.g., sum or mean) over messages from both sets of neighbors:

$$\Delta h_{EU,i} = \frac{1}{K} \sum_{k=1}^K \phi_{upd,EU} \left([h_i, \bigoplus_{j \in \mathcal{N}_{EU}(i)} \alpha_{EU,ij}^k \phi_{EU,h}^k(m_{EU,ij}) + \bigoplus_{j \in \mathcal{N}_{SH}(i)} \beta_{EU,ij}^k \psi_{EU,h}^k(m_{SH \rightarrow EU,ij})] \right), \quad (8)$$

where, $\phi_{EU,h}^k$ and $\psi_{EU,h}^k$ are learnable message transformation networks per head. Similarly, the coordinate update is computed by aggregating directional messages scaled by attention:

$$\Delta \mathbf{x}_{EU,i} = \frac{1}{K} \sum_{k=1}^K \left(\bigoplus_{j \in \mathcal{N}_{EU}(i)} \alpha_{EU,ij}^k \phi_{EU,x}^k(m_{EU,ij}) \mathbf{x}_{ij} + \bigoplus_{j \in \mathcal{N}_{SH}(i)} \beta_{EU,ij}^k \psi_{EU,x}^k(m_{SH \rightarrow EU,ij}) \mathbf{x}_{ij} \right) \quad (9)$$

The EU space update integrates both local geometry and global SH information through learned, attention-weighted message passing. This allows the model to effectively capture rich interactions that span different geometric representations.

Spherical Harmonics Space Update Similarly, in SH space, for each node i and its neighbor j , we compute messages from both the EU and SH neighbors as

$$m_{SH,ij} = \phi_{SH}([h_i, h_j, \mathbf{r}_i \odot \mathbf{r}_j]), \quad m_{EU \rightarrow SH,ij} = \psi_{EU \rightarrow SH}([\mathbf{r}_i \odot \mathbf{r}_j, \|\mathbf{x}_{ij}\|]) \quad (10)$$

We then adopt K -head attention to aggregate the information as

$$\Delta h_{SH,i} = \frac{1}{K} \sum_{k=1}^K \phi_{upd,SH} \left([h_i, \bigoplus_{j \in \mathcal{N}_{SH}(i)} \alpha_{SH,ij}^k \phi_{SH,h}^k(m_{SH,ij}) + \bigoplus_{j \in \mathcal{N}_{EU}(i)} \beta_{SH,ij}^k \psi_{SH,h}^k(m_{EU \rightarrow SH,ij})] \right), \quad (11)$$

where $\alpha_{SH,ij}^k = \sigma(\phi_{SH,att}^k(m_{SH,ij}))$ and $\beta_{SH,ij}^k = \sigma(\psi_{SH,\beta}^k(m_{EU \rightarrow SH,ij}))$. The SH feature is affected by $\mathbf{r}_j$ and $\hat{Y}(\hat{\mathbf{x}}_{ij})$ together. Specifically, for the within SH space update, utilizing $\mathbf{r}_j$ directly captures the structural similarity, including angle and torsion features, among SH neighbors. For cross-space updates from EU to SH, the relative direction of the EU neighbors is crucial. $\hat{Y}(\hat{\mathbf{x}}_{ij})$ explicitly incorporates this geometric directional information from its EU neighbors, potentially impacting SH features with the update given by the following.

$$\Delta \mathbf{r}_{SH,i} = \frac{1}{K} \sum_{k=1}^K \left(\bigoplus_{j \in \mathcal{N}_{SH}(i)} \alpha_{SH,ij}^k \phi_{SH,r}^k(m_{SH,ij}) \mathbf{r}_j + \bigoplus_{j \in \mathcal{N}_{EU}(i)} \beta_{SH,ij}^k \psi_{SH,r}^k(m_{EU \rightarrow SH,ij}) \hat{Y}(\hat{\mathbf{x}}_{ij}) \right) \quad (12)$$

Dual Space Update. We employ a residual connection, and the overall update is given by

$$h_i = h_i + \Delta h_{EU,i} + \Delta h_{SH,i}, \quad \mathbf{x}_i = \mathbf{x}_i + \Delta \mathbf{x}_{EU,i}, \quad \mathbf{r}_i = \mathbf{r}_i + \Delta \mathbf{r}_{SH,i}. \quad (13)$$

3.3 Cross-space Interaction Pooling

To create residue- and molecule-level representations, we use hierarchical pooling over atomic node clusters. The challenge is to maintain both local geometry and global structural context. Standard pooling methods, like average or attention-based aggregation, handle features from different geometric spaces separately. In biomolecular structures, local and global geometry are interconnected; for instance, an RNA nucleotide's torsional configuration (well-represented in SH space) can be affected by nearby atomic arrangements (in EU space), and vice versa.

To address the challenge, we introduce CSIP, a novel hierarchical pooling strategy that facilitates bidirectional information exchange between EU (local) and SH (global) spaces during aggregation. This approach integrates multi-scale geometric information while preserving equivariance. Specifically, we first compute attention weights α_i using only invariant quantities, then the weights are used to perform attention-based pooling as:

$$h' = \frac{1}{|C|} \sum_{i \in C} \alpha_i h_i, \quad \alpha_i = \sigma(\phi_P([h_i, d_i])), \quad (14)$$

where ϕ_P is a neural network applied to invariant features to calculate the attention score, d_i denotes the average pairwise distance from node i to other nodes, and C is the cluster of nodes over which pooling is performed. For example, a nucleotide in RNA or amino acid in protein consists of a cluster of atoms. To enrich the EU representation with global structural cues, we inject projected SH features:

$$\mathbf{x}' = \frac{1}{|C|} \sum_{i \in C} \alpha_i [\mathbf{x}_i + \gamma \cdot \text{Proj}_{SH}(\mathbf{r}_i)], \quad \text{Proj}_{SH}(\mathbf{r}_i) = \sum_{l=1}^{l_{max}} w_l \|\mathbf{r}_i^{(l)}\| \mathbf{x}_i, \quad (15)$$

where $\mathbf{r}_i^{(l)}$ is the SH component at degree l , and w_l are learnable weights. Proj_{SH} converts SH space features $\mathbf{r}$ to EU space by producing an invariant signal summarizing the SH features' norms, which modulates the geometric position via a learnable coefficient γ . Moreover, we enhance SH features with a projection of relative EU coordinates:

$$\mathbf{r}' = \frac{1}{|C|} \sum_{i \in C} \alpha_i [\mathbf{r}_i + \epsilon \cdot \text{Proj}_{EU}(\mathbf{x}_i)], \quad \text{Proj}_{EU}(\mathbf{x}_i) = \hat{Y}(\mathbf{x}_i - \mathbf{x}_{avg}), \quad (16)$$

where $\mathbf{x}_{avg} = \frac{1}{|C|} \sum_{i \in C} \mathbf{x}_i$ centers the input to maintain translation equivariance. The projection $\text{Proj}_{EU}(\mathbf{x}_i)$ converts $\mathbf{x}$ from EU space to SH space. Therefore, we choose to use the SH function $\hat{Y}$. It embeds local relative geometry into the SH domain, controlled by a learnable coefficient ϵ . The pooled features h' , $\mathbf{x}'$, and $\mathbf{r}'$ serve as residue-level inputs for the subsequent DualEqui layers, enabling hierarchical modeling from atoms to residues.

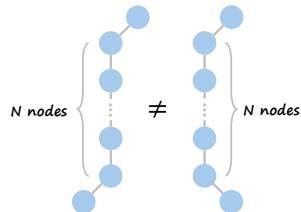
3.4 DualEquiNet

As shown in Figure 1(a), DualEquiNet captures the hierarchical nature of biomolecular structures through two sequential modules: an initial atom-level processing module and a subsequent residue-level module, separated by the CSIP layer. Each module consists of multiple DualEqui layers. Figure 1(b) shows that each DualEqui layer encompasses the operations defined in Equations 1 to 13, and the network-level update is expressed as:

$$h^{l+1}, \mathbf{x}^{l+1}, \mathbf{r}^{l+1} = \text{DualEquiLayer}(h^l, \mathbf{x}^l, \mathbf{r}^l), \quad (17)$$

where the superscript l denotes the layer index, and is distinct from the SH order (l) with parenthesis. There are two types of layers: atom layers and residue layers. The model first applies L_{atom} DualEqui layers to capture fine-grained atomic interactions in both EU and SH spaces. The CSIP module then aggregates these features into residue-level representations, which are further processed by L_{res} residue-level DualEqui layers to capture higher-order structural patterns and long-range dependencies. The input to the first atom layer is $h^0 = h_{atom}$. The first residue layer receives $h^{L_{atom}+1} = [h^{L_{atom}}, h_{res}]$, a concatenation of the final atom-layer output and the CSIP-derived residue features. This hierarchical design enhances both computational efficiency and representational capacity, aligning naturally with the multi-scale structure of biomolecules. The equivariance of the networks is proved in Appendix C.

Table 1: (a) Left: The N -chain task aims to differentiate two N -chain geometric graphs that differ only in the positions of their end nodes. (b) Right: The expressivity in the N -chain task with $N = 10$. Anomalous results are highlighted in red and expected in green. DualEquiNet demonstrates a stronger capability to differentiate the two geometric graphs due to its expressive architecture.



N nodes $\neq$ N nodes

Layers	$\lfloor \frac{N}{2} \rfloor = 5$	$\lfloor \frac{N}{2} \rfloor + 1 = 6$	7	8	9	10
SchNet	50.0±0.0	50.0±0.0	50.0±0.0	50.0±0.0	50.0±0.0	50.0±0.0
EGNN	50.0±0.0	50.0±0.0	50.0±0.0	50.0±0.0	50.0±0.0	50.0±0.0
GVPGNN	50.0±0.0	52.0±9.8	51.0±7.0	53.0±11.9	51.5±8.5	54.5±14.3
TFN	50.0±0.0	50.0±0.0	50.0±0.0	50.0±0.0	50.0±0.0	50.0±0.0
FastEGNN	99.5±5.0	100.0±0.0	91.0±19.2	99.5±5.0	100.0±0.0	51.5±8.5
HEGNN	50.0±0.0	50.0±0.0	50.0±0.0	50.0±0.0	50.0±0.0	50.0±0.0
GotenNet	50.0±0.0	50.0±0.0	50.0±0.0	50.0±0.0	50.0±0.0	50.0±0.0
DualEquiNet $d_{SH}=0.95$	52.5±10.9	51.5±8.5	52.5±10.9	53.5±12.8	99.5±5.0	52.5±13.0
DualEquiNet $d_{SH}=0.90$	98.5±8.5	50.5±5.0	50.5±5.0	99.0±7.0	99.5±5.0	53.5±12.8
DualEquiNet $d_{SH}=0.80$	98.0±9.8	54.0±13.6	100.0±0.0	100.0±0.0	100.0±0.0	100.0±0.0
DualEquiNet $d_{SH}=0.70$	99.0±7.0	100.0±0.0	100.0±0.0	100.0±0.0	100.0±0.0	100.0±0.0
DualEquiNet $d_{SH}=0.60$	100.0±0.0	100.0±0.0	100.0±0.0	100.0±0.0	100.0±0.0	100.0±0.0

4 Experiments and Results

We evaluate DualEquiNet across a suite of tasks to assess its expressivity, ability to capture long-range dependencies, and effectiveness on real-world biomolecular property prediction. Our experiments span synthetic benchmarks and challenging biomolecular tasks, including RNA property prediction, solvent-accessible surface area estimation, and torsion angle regression. The dataset details are in Appendix D.6.

We benchmark DualEquiNet against baselines from various categories: (1)Invariant: SchNet [63]; (2)Scalarization Equivariant: EGNN [62], GVPGNN [39]; (3)Higher-order Equivariant: TFN [69], FastEGNN [85], HEGNN [17], and GotenNet [6], covering a range of popular and recent baselines including invariant, equivariant and low- to high-order tensor methods. A detailed baseline description is provided in Appendix D.7.

4.1 Model Expressivity on Synthetic Dataset

To evaluate DualEquiNet’s ability to capture long-range dependencies, we use the N -chain discrimination task from prior work [40]. This task provides a synthetic yet rigorous benchmark for assessing the expressive power of geometric GNNs, particularly their capacity to propagate and integrate information across distant nodes. In this experiment, as illustrated in Table 1 (left), there are a pair of graphs and each graph consists of a linear chain with N nodes, identical except for their end-node configurations—analogue to chiral molecules. The model is trained on one graph and evaluated on both to test its ability to distinguish the chiral pair. These graphs are $(\lfloor \frac{N}{2} \rfloor + 1)$ -hop distinguishable, meaning a GNN requires at least that many layers to differentiate them as proven in [40]. As N increases, the task becomes more challenging due to oversquashing [4]. While the original Geometric Weisfeiler-Leman (GWL) test uses $N = 4$, we extend this to $N = 10$, requiring at least 6-hop information—mirroring challenges in modeling large biomolecules. Note that in real-world scenarios, large biomolecules can consist of thousands of nodes, making long-range information exchange even more challenging. We vary the SH-space cutoff distance d_{SH} to study its effect on DualEquiNet performance.

As shown in Table 1, most baselines fail to distinguish the N -chain graphs, performing no better than random guessing even when stacking more than $\lfloor \frac{N}{2} \rfloor + 1$ layers. FastEGNN is an exception due to its global virtual nodes. In contrast, DualEquiNet achieves higher accuracy even with fewer than $\lfloor \frac{N}{2} \rfloor + 1$ layers by leveraging SH-space neighbors for global context. Performance improves as d_{SH} decreases, reaching near-perfect accuracy at $d_{SH} \leq 0.7$, demonstrating the effectiveness of SH-space communication in overcoming long-range limitations with lower SH cutoff distance signifying more SH-space neighbors. We also validate expressivity on another rotational symmetry task (Appendix D.4) showing similar results, confirming the benefit of high-order SH features, in agreement with HEGNN [17] and prior analyses [40].

4.2 RNA Property Prediction

We evaluate DualEquiNet on RNA property prediction tasks because RNA molecules hold significant therapeutic potential, and accurate in-silico modeling can greatly reduce experimental costs and

Table 2: Performance of RNA property prediction (RMSE±standard deviation (std.)). The best-performing model is highlighted in bold. DualEquiNet outperforms other baselines on each metric.

Method	COVID				Ribonanza			Tc-Ribo
	Reactivity	pH10	Mg pH10	Avg.	DMS	2A3	Avg.	Factor
SchNet	0.451±0.006	0.633±0.012	0.518±0.010	0.534±0.011	0.909±0.018	0.966±0.004	0.937±0.012	0.737±0.002
EGNN	0.453±0.006	0.634±0.013	0.520±0.010	0.535±0.011	0.912±0.017	0.968±0.005	0.940±0.012	0.720±0.022
GVPGNN	0.448±0.006	0.631±0.012	0.515±0.009	0.531±0.011	0.901±0.015	0.963±0.004	0.932±0.010	0.736±0.003
TFN	0.453±0.006	0.634±0.012	0.520±0.010	0.535±0.011	0.911±0.018	0.968±0.004	0.939±0.012	0.694±0.008
FastEGNN	0.390±0.019	0.582±0.029	0.463±0.027	0.478±0.031	0.808±0.022	0.869±0.015	0.839±0.020	0.731±0.006
HEGNN	0.450±0.005	0.632±0.012	0.517±0.009	0.533±0.011	0.904±0.015	0.964±0.004	0.934±0.010	0.729±0.007
GotenNet	0.388±0.002	0.582±0.011	0.455±0.008	0.475±0.009	0.832±0.017	0.882±0.009	0.857±0.013	0.718±0.002
EGNN + Pooling	0.280±0.006	0.469±0.006	0.362±0.010	0.370±0.009	0.688±0.028	0.534±0.014	0.611±0.010	0.699±0.016
SchNet + Pooling	0.302±0.006	0.495±0.010	0.384±0.010	0.394±0.010	0.721±0.019	0.642±0.004	0.681±0.011	0.702±0.007
FastEGNN + Pooling	0.379±0.002	0.566±0.009	0.454±0.008	0.467±0.007	0.773±0.024	0.819±0.006	0.796±0.013	0.721±0.013
DualEquiNet	0.272±0.004	0.448±0.010	0.340±0.005	0.353±0.006	0.611±0.025	0.505±0.024	0.558±0.012	0.636±0.049

time [21]. RNA molecules pose unique modeling challenges due to their large size, complex secondary and tertiary structures involving both short-range and long-range interactions [67], and the necessity for equivariant representations respecting rotations and translations [57]. An effective RNA modeling approach must efficiently integrate both local atomic interactions and global structural dependencies.

We use three RNA datasets consisting of thousands of atoms: (i) **CovidVaccine** [74], which provides nucleotide-level reactivity and degradation labels relevant for vaccine stability; (ii) **Ribonanza** [37], which includes nucleotide-level reactivity under different compounds to assess RNA folding and flexibility; and (iii) **Tc-ribo** [33], which contains RNA-level regulatory behavior in response to tetracycline. 3D RNA structures are generated using RhoFold [65] following prior works [57].

We follow the experimental protocol in [57] to use eta-theta pseudotorsional backbone [73] as input and use Root Mean Squared Error (RMSE) as our performance metric. All models are evaluated using consistent train-validation-test splits (8:1:1), with hyperparameters selected based on the best validation performance, optimized via Optuna [3] to ensure fair comparisons.

As shown in Table 2, DualEquiNet outperforms all baselines, reducing RMSE by 25.7%, 33.5%, and 8.4% on CovidVaccine, Ribonanza, and Tc-ribo, respectively compared to best baseline. This highlights DualEquiNet’s strength in modeling complex RNA interactions, especially long-range dependencies, making it a strong candidate for RNA property prediction tasks. Note that the pooling in Table 2 is the standard hierarchical pooling (e.g., atom-to-nucleotide for RNA, atom-to-amino acid for proteins).

4.3 Protein and RNA Solvent-Accessible Surface Area (SASA) prediction

Solvent-accessible surface area (SASA) measures how exposed atoms or residues are to solvent, reflecting a large biomolecule’s 3D conformation and influencing interactions and stability [66, 54]. Accurate SASA prediction is crucial for understanding and engineering molecular behavior [68].

For protein SASA prediction, we use the existing mRFP protein dataset from [68], with 3D protein structures generated using AlphaFold[41]. This dataset contains ground-truth SASA values, offering a reliable benchmark for model evaluation.

Additionally, we introduce a new SASA benchmark dataset for RNA, sourced from RNASolo [2], which includes experimentally determined 3D structures of RNA molecules. During preprocessing, we choose RNA molecules with sequence lengths ranging from 10 to 400 and exclude RNA sequences with multiple chains or non-AUGC nucleotides. Ground-truth residue-level SASA values are computed using DSSR [51]. Together with the existing protein SASA benchmark, this new RNA dataset provides a comprehensive benchmark for assessing models’ ability to capture detailed structural representations of both RNA and protein.

Table 3: Performance on SASA Prediction (RMSE±std).

	RNASolo	mRFP Protein (×10)
SchNet	48.560±2.187	8.370±0.963
EGNN	33.011±2.323	9.054±1.303
GVPGNN	37.704±1.050	20.351±0.964
TFN	51.003±1.214	20.176±1.277
FastEGNN	37.251±1.028	12.282±1.310
HEGNN	37.782±1.795	12.382±3.525
GotenNet	34.966±1.671	8.743±1.558
DualEquiNet	31.972±0.859	5.959±1.160

Table 4: Performance on TorsionAngle dataset with 7 predicted angles. (Circular MAE $\pm$ std.).

	Avg.	α	β	γ	δ	ϵ	ζ	χ
SchNet	18.54 $\pm$ 0.86	31.38 $\pm$ 1.44	17.54 $\pm$ 0.20	24.44 $\pm$ 1.35	7.68 $\pm$ 0.25	15.48 $\pm$ 0.56	20.50 $\pm$ 1.30	12.78 $\pm$ 0.63
EGNN	17.79 $\pm$ 0.61	29.62 $\pm$ 0.98	17.25 $\pm$ 0.26	22.96 $\pm$ 0.79	7.54 $\pm$ 0.16	15.17 $\pm$ 0.67	19.72 $\pm$ 1.11	12.29 $\pm$ 0.58
GVPGNN	17.56 $\pm$ 0.21	29.69 $\pm$ 0.41	16.94 $\pm$ 0.25	22.77 $\pm$ 0.49	7.02 $\pm$ 0.26	15.02 $\pm$ 0.57	19.63 $\pm$ 0.54	11.85 $\pm$ 0.15
TFN	21.97 $\pm$ 0.45	36.52 $\pm$ 0.62	18.19 $\pm$ 0.40	27.63 $\pm$ 0.66	11.63 $\pm$ 0.43	17.64 $\pm$ 0.79	26.59 $\pm$ 0.77	15.57 $\pm$ 0.74
FastEGNN	18.89 $\pm$ 0.22	31.96 $\pm$ 0.58	17.43 $\pm$ 0.40	24.64 $\pm$ 0.60	8.47 $\pm$ 0.45	15.66 $\pm$ 0.45	21.31 $\pm$ 0.33	12.78 $\pm$ 0.13
HEGNN	18.41 $\pm$ 0.22	31.12 $\pm$ 0.59	17.49 $\pm$ 0.21	23.78 $\pm$ 0.69	7.91 $\pm$ 0.66	15.35 $\pm$ 0.46	20.54 $\pm$ 0.18	12.67 $\pm$ 0.26
GotenNet	16.27 $\pm$ 1.19	26.88$\pm$1.57	15.90 $\pm$ 0.60	21.06$\pm$1.34	6.72 $\pm$ 0.35	13.85 $\pm$ 0.64	17.74 $\pm$ 1.63	10.96 $\pm$ 0.84
DualEquiNet	15.87$\pm$2.03	26.89 $\pm$ 3.00	15.48$\pm$1.17	21.12 $\pm$ 2.68	6.30$\pm$0.45	13.68$\pm$1.29	17.03$\pm$1.84	10.60$\pm$1.32

The baselines and data splits are consistent with those in Section 4.2 and we use Optuna for optimal hyperparameter search. RMSE to the ground truth SASA values is reported. Table 3 presents performance results on the SASA benchmarks. DualEquiNet consistently achieves the lowest prediction error across both RNA and protein datasets, demonstrating its superior ability to capture and refine both local and global structural features through its dual-space hierarchical representations.

4.4 RNA Torsion Angle Prediction

Torsion angles define rotations around specific bonds, shaping the 3D conformation of biomolecules and influencing their secondary and tertiary structures, interactions, and functions. Accurate prediction of these angles is essential for modeling structural geometry. While sequence modeling techniques have been proposed for RNA torsion angle prediction [1, 11], they rely solely on nucleotide sequences and ignore the 3D structural context. In contrast, torsion angles are fundamentally determined by spatial geometry (more details in Appendix D.6).

Similar to SASA experiments, the **TorsionAngle** dataset uses ground-truth RNA sequences and structures from RNASolo and torsion angles are computed using DSSR. Consistent with [1], we adopt the circular Mean Absolute Error (MAE) loss. To predict seven torsion angles ($\alpha, \beta, \gamma, \delta, \epsilon, \zeta, \chi$), the model outputs 14 values—sin and cos for each angle. The angle θ is recovered via $\theta = \arctan(\sin(\theta), \cos(\theta))$, and the circular MAE is computed as $\text{MAE}(\theta) = \frac{1}{N_{nt}} \sum_{i=1}^{N_{nt}} \min(\Delta\theta_i, 360^\circ - \Delta\theta_i)$ where $\Delta\theta_i = |\theta_{\text{pred},i} - \theta_{\text{true},i}|$ and N_{nt} is the number of predicted nucleotides.

Results presented in Table 4 illustrate the performance of various models across different torsion angles prediction. Our DualEquiNet achieves the lowest average error and performs best for 5 out of 7 angles prediction. These findings highlight that DualEquiNet effectively learns 3D structural representations, enabling accurate prediction of torsion angles based on molecular geometry.

4.5 Ablation and Neighborhood Dynamics: Importance of Dual-Space Learning

To quantify the contributions of key components in DualEquiNet, we perform an ablation study across large-scale RNA property datasets. We evaluate four configurations: (i) the full DualEquiNet model; (ii)

Table 5: Ablation Study on RNA Property datasets (RMSE $\pm$ std.).

Model	COVID $\downarrow$	Ribonanza $\downarrow$	Tc-Ribo $\downarrow$
DualEquiNet	0.353$\pm$0.006	0.558$\pm$0.123	0.636$\pm$0.049
w/o CSIP	0.356 $\pm$ 0.002	0.586 $\pm$ 0.016	0.682 $\pm$ 0.026
w/o CSIP/Cross	0.356 $\pm$ 0.013	0.591 $\pm$ 0.081	0.698 $\pm$ 0.010
w/o CSIP/Cross/Dual	0.362 $\pm$ 0.010	0.592 $\pm$ 0.028	0.699 $\pm$ 0.010

w/o CSIP, which replaces contextual structure-informed pooling (CSIP) with standard mean pooling; (iii) **w/o CSIP/Cross**, which disables cross-space message passing but retains dual-space representations; and (iv) **w/o CSIP/Cross/Dual**, a single-space Euclidean model. As shown in Table 5, each component enhances performance, with CSIP particularly benefiting the Ribonanza and Tc-ribo datasets, and the dual-space mechanism notably improving performance on COVID. These results confirm that combining local and global geometric contexts through dual-space learning and cross-space communication is essential for generalization.

More experiments about the neighborhood analysis, time complexity analysis, and expressivity on symmetric structures are in Appendix D.

5 Conclusion

We introduce DualEquiNet, a novel framework for modeling large biomolecules by hierarchically integrating Euclidean and spherical harmonics representations across atomic, residue (nucleotide/amino acid), and molecular levels. DualEquiNet is $E(3)$ equivariant and effectively captures both local and global structural dependencies. Extensive experiments on real-world datasets demonstrate state-of-the-art performance over existing geometric GNNs, highlighting the benefits of dual-space representations and hierarchical learning for large biomolecular modeling. Although DualEquiNet is effective across diverse tasks, its performance may depend on the quality of input 3D structures, which can vary across datasets and affect generalization. Future work will explore integrating structure prediction into the pipeline and extending the framework to broader molecular applications.

Acknowledgments

This material is based upon work supported by, or in part by, the Army Research Office (ARO) under grant number W911NF-21-10198 and the Cisco Faculty Research Award. The findings in this paper do not necessarily reflect the view of the funding agencies.

References

- [1] Abrar Rahman Abir, Md Toki Tahmid, Rafiqul Islam Rayan, and M Saifur Rahman. DeepRNA-twist: Language model guided rna torsion angle prediction with attention-inception network. *bioRxiv*, pages 2024–10, 2024.
- [2] Bartosz Adamczyk, Maciej Antczak, and Marta Szachniuk. Rnasolo: a repository of cleaned pdb-derived rna 3d structures. *Bioinformatics*, 38(14):3668–3670, 2022.
- [3] Takuya Akiba, Shotaro Sano, Toshihiko Yanase, Takeru Ohta, and Masanori Koyama. Optuna: A next-generation hyperparameter optimization framework. In *Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery & data mining*, pages 2623–2631, 2019.
- [4] Uri Alon and Eran Yahav. On the bottleneck of graph neural networks and its practical implications. In *International Conference on Learning Representations*, 2021.
- [5] Kenneth Atz, Francesca Grisoni, and Gisbert Schneider. Geometric deep learning on molecular representations. *Nature Machine Intelligence*, 3(12):1023–1032, 2021.
- [6] Sarp Aykent and Tian Xia. Gotennet: Rethinking efficient 3d equivariant graph neural networks. In *The Thirteenth International Conference on Learning Representations*, 2025.
- [7] David Baker and Andrej Sali. Protein structure prediction and structural genomics. *Science*, 294(5540):93–96, 2001.
- [8] Justin Baker, Qingsong Wang, Cory D Hauck, and Bao Wang. Implicit graph neural networks: A monotone operator viewpoint. In *International Conference on Machine Learning*, pages 1521–1548. PMLR, 2023.
- [9] Ilyes Batatia, David P Kovacs, Gregor Simm, Christoph Ortner, and Gábor Csányi. Mace: Higher order equivariant message passing neural networks for fast and accurate force fields. *Advances in Neural Information Processing Systems*, 35:11423–11436, 2022.
- [10] Helen M Berman, John Westbrook, Zukang Feng, Gary Gilliland, Talapady N Bhat, Helge Weissig, Ilya N Shindyalov, and Philip E Bourne. The protein data bank. *Nucleic acids research*, 28(1):235–242, 2000.
- [11] Clément Bernard, Guillaume Postic, Sahar Ghannay, and Fariza Tahi. Rna-torsionbert: leveraging language models for rna 3d torsion angles prediction. *Bioinformatics*, 41(1):btaf004, 2025.

- [12] Johannes Brandstetter, Rob Hesselink, Elise van der Pol, Erik J Bekkers, and Max Welling. Geometric and physical quantities improve e (3) equivariant message passing. *arXiv preprint arXiv:2110.02905*, 2021.
- [13] Johannes Brandstetter, Rob Hesselink, Elise van der Pol, Erik J. Bekkers, and Max Welling. Geometric and Physical Quantities Improve E(3) Equivariant Message Passing, March 2022. arXiv:2110.02905 [cs, stat].
- [14] Michael M Bronstein, Joan Bruna, Taco Cohen, and Petar Veličković. Geometric deep learning: Grids, groups, graphs, geodesics, and gauges. *arXiv preprint arXiv:2104.13478*, 2021.
- [15] Jamie H Cate, Anne R Gooding, Elaine Podell, Kaihong Zhou, Barbara L Golden, Craig E Kundrot, Thomas R Cech, and Jennifer A Doudna. Crystal structure of a group i ribozyme domain: principles of rna packing. *Science*, 273(5282):1678–1685, 1996.
- [16] Jiacheng Cen, Anyi Li, Ning Lin, Yuxiang Ren, Zihe Wang, and Wenbing Huang. Are high-degree representations really unnecessary in equivariant graph neural networks? In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*.
- [17] Jiacheng Cen, Anyi Li, Ning Lin, Yuxiang Ren, Zihe Wang, and Wenbing Huang. Are high-degree representations really unnecessary in equivariant graph neural networks? *arXiv preprint arXiv:2410.11443*, 2024.
- [18] Ming Chen, Zhewei Wei, Zengfeng Huang, Bolin Ding, and Yaliang Li. Simple and deep graph convolutional networks. In *International conference on machine learning*, pages 1725–1735. PMLR, 2020.
- [19] Qi Chen, Yifei Wang, Yisen Wang, Jianlong Chang, Qi Tian, Jiansheng Yang, and Zhouchen Lin. Efficient and scalable implicit graph neural networks with virtual equilibrium. In *2022 IEEE International Conference on Big Data (Big Data)*, pages 864–873. IEEE, 2022.
- [20] Qi Chen, Yifei Wang, Yisen Wang, Jiansheng Yang, and Zhouchen Lin. Optimization-induced graph implicit nonlinear diffusion. In *International Conference on Machine Learning*, pages 3648–3661. PMLR, 2022.
- [21] Tulsi Ram Damase, Roman Sukhovshin, Christian Boada, Francesca Taraballi, Roderic I Pettigrew, and John P Cooke. The limitless future of rna therapeutics. *Frontiers in bioengineering and biotechnology*, 9:628137, 2021.
- [22] Jie Deng, Xianyang Fang, Lin Huang, Shanshan Li, Lilei Xu, Keqiong Ye, Jinsong Zhang, Kaiming Zhang, and Qiangfeng Cliff Zhang. Rna structure determination: From 2d to 3d. *Fundamental Research*, 3(5):727–737, 2023.
- [23] Alexandre Duval, Simon V Mathis, Chaitanya K Joshi, Victor Schmidt, Santiago Miret, Fragkiskos D Malliaros, Taco Cohen, Pietro Liò, Yoshua Bengio, and Michael Bronstein. A hitchhiker’s guide to geometric gnns for 3d atomic systems. *arXiv preprint arXiv:2312.07511*, 2023.
- [24] Vijay Prakash Dwivedi, Ladislav Rampásek, Michael Galkin, Ali Parviz, Guy Wolf, Anh Tuan Luu, and Dominique Beaini. Long range graph benchmark. *Advances in Neural Information Processing Systems*, 35:22326–22340, 2022.
- [25] Matthias Fey and Jan E. Lenssen. Fast graph representation learning with PyTorch Geometric. In *ICLR Workshop on Representation Learning on Graphs and Manifolds*, 2019.
- [26] J Thorben Frank, Oliver T Unke, Klaus-Robert Müller, and Stefan Chmiela. A euclidean transformer for fast and stable machine learned force fields. *Nature Communications*, 15(1):6539, 2024.
- [27] Fabian Fuchs, Daniel Worrall, Volker Fischer, and Max Welling. Se (3)-transformers: 3d roto-translation equivariant attention networks. *Advances in neural information processing systems*, 33:1970–1981, 2020.

- [28] Johannes Gasteiger, Florian Becker, and Stephan Günnemann. Gemnet: Universal directional graph neural networks for molecules. *Advances in Neural Information Processing Systems*, 34:6790–6802, 2021.
- [29] Johannes Gasteiger, Shankari Giri, Johannes T Margraf, and Stephan Günnemann. Fast and uncertainty-aware directional message passing for non-equilibrium molecules. *arXiv preprint arXiv:2011.14115*, 2020.
- [30] Johannes Gasteiger, Janek Groß, and Stephan Günnemann. Directional message passing for molecular graphs. *arXiv preprint arXiv:2003.03123*, 2020.
- [31] Mario Geiger and Tess Smidt. e3nn: Euclidean neural networks. *arXiv preprint arXiv:2207.09453*, 2022.
- [32] Simon Geisler, Yujia Li, Daniel J Mankowitz, Ali Taylan Cemgil, Stephan Günnemann, and Cosmin Paduraru. Transformers meet directed graphs. In *International conference on machine learning*, pages 11144–11172. PMLR, 2023.
- [33] Ann-Christin Groher, Sven Jager, Christopher Schneider, Florian Groher, Kay Hamacher, and Beatrix Suess. Tuning the performance of synthetic riboswitches using machine learning. *ACS synthetic biology*, 8(1):34–44, 2018.
- [34] Fangda Gu, Heng Chang, Wenwu Zhu, Somayeh Sojoudi, and Laurent El Ghaoui. Implicit graph neural networks. *Advances in neural information processing systems*, 33:11984–11995, 2020.
- [35] Jiaqi Han, Jiacheng Cen, Liming Wu, Zongzhao Li, Xiangzhe Kong, Rui Jiao, Ziyang Yu, Tingyang Xu, Fandi Wu, Zihe Wang, et al. A survey of geometric graph neural networks: Data structures, models and applications. *arXiv preprint arXiv:2403.00485*, 2024.
- [36] Jiaqi Han, Yu Rong, Tingyang Xu, and Wenbing Huang. Geometrically equivariant graph neural networks: A survey. *arXiv preprint arXiv:2202.07230*, 2022.
- [37] Shujun He, Rui Huang, Jill Townley, Rachael C Kretsch, Thomas G Karagianes, David BT Cox, Hamish Blair, Dmitry Penzar, Valeriy Vyaltsev, Elizaveta Aristova, et al. Ribonanza: deep learning of rna structure through dual crowdsourcing. *bioRxiv*, 2024.
- [38] Yi Jiang, Wanchao Yin, and H Eric Xu. Rna-dependent rna polymerase: Structure, mechanism, and drug discovery for covid-19. *Biochemical and biophysical research communications*, 538:47–53, 2021.
- [39] Bowen Jing, Stephan Eismann, Patricia Suriana, Raphael JL Townshend, and Ron Dror. Learning from protein structure with geometric vector perceptrons. *arXiv preprint arXiv:2009.01411*, 2020.
- [40] Chaitanya K. Joshi, Cristian Bodnar, Simon V. Mathis, Taco Cohen, and Pietro Liò. On the expressive power of geometric graph neural networks. In *International Conference on Machine Learning*, 2023.
- [41] John Jumper, Richard Evans, Alexander Pritzel, Tim Green, Michael Figurnov, Olaf Ronneberger, Kathryn Tunyasuvunakool, Russ Bates, Augustin Žídek, Anna Potapenko, et al. Highly accurate protein structure prediction with alphafold. *nature*, 596(7873):583–589, 2021.
- [42] Johannes Klicpera, Janek Groß, Stephan Günnemann, et al. Directional message passing for molecular graphs. In *ICLR*, pages 1–13, 2020.
- [43] Risi Kondor and Shubhendu Trivedi. On the generalization of equivariance and convolution in neural networks to the action of compact groups. In *International conference on machine learning*, pages 2747–2755. PMLR, 2018.
- [44] Arthur Kosmala, Johannes Gasteiger, Nicholas Gao, and Stephan Günnemann. Ewald-based long-range message passing for molecular graphs. In *International Conference on Machine Learning*, pages 17544–17563. PMLR, 2023.

- [45] Chenao Li, Shuo Yan, and Enyan Dai. Unizyme: A unified protein cleavage site predictor enhanced with enzyme active-site knowledge. *arXiv preprint arXiv:2502.06914*, 2025.
- [46] Mingjie Li, Yifei Wang, Yisen Wang, and Zhouchen Lin. Unbiased stochastic proximal solver for graph neural networks with equilibrium states. In *The Eleventh International Conference on Learning Representations*, 2022.
- [47] Xuan Li, Zhanke Zhou, Jiangchao Yao, Yu Rong, Lu Zhang, and Bo Han. Neural atoms: Propagating long-range interaction in molecular graphs through efficient communication channel. *arXiv preprint arXiv:2311.01276*, 2023.
- [48] Juncheng Liu, Bryan Hooi, Kenji Kawaguchi, Yiwei Wang, Chaosheng Dong, and Xiaokui Xiao. Scalable and effective implicit graph neural networks on large graphs. In *The Twelfth International Conference on Learning Representations*, 2024.
- [49] Juncheng Liu, Bryan Hooi, Kenji Kawaguchi, and Xiaokui Xiao. Mgnni: Multiscale graph neural networks with implicit layers. *Advances in Neural Information Processing Systems*, 35:21358–21370, 2022.
- [50] Yi Liu, Limei Wang, Meng Liu, Yuchao Lin, Xuan Zhang, Bora Oztekin, and Shuiwang Ji. Spherical message passing for 3d molecular graphs. In *International Conference on Learning Representations*, 2022.
- [51] Xiang-Jun Lu, Harmen J Bussemaker, and Wilma K Olson. Dssr: an integrated software tool for dissecting the spatial structure of rna. *Nucleic acids research*, 43(21):e142–e142, 2015.
- [52] Yuankai Luo, Hongkang Li, Lei Shi, and Xiao-Ming Wu. Enhancing graph transformers with hierarchical distance structural encoding. *Advances in Neural Information Processing Systems*, 37:57150–57182, 2024.
- [53] Yuankai Luo, Veronika Thost, and Lei Shi. Transformers over directed acyclic graphs. *Advances in Neural Information Processing Systems*, 36:47764–47782, 2023.
- [54] Avinash Mishra, Shoba Ranganathan, B Jayaram, and Abdul Sattar. Role of solvent accessibility for aggregation-prone patches in protein folding. *Scientific reports*, 8(1):12896, 2018.
- [55] Amina Mollaysa, Artem Moskalev, Pushpak Pati, Tommaso Mansi, Mangal Prakash, and Rui Liao. Biolangfusion: Multimodal fusion of dna, mrna, and protein language models, 2025.
- [56] Artem Moskalev, Mangal Prakash, Junjie Xu, Tianyu Cui, Rui Liao, and Tommaso Mansi. Geometric hyena networks for large-scale equivariant learning. In *Forty-second International Conference on Machine Learning*, 2025.
- [57] Artem Moskalev, Mangal Prakash, Junjie Xu, Rui Liao, and Tommaso Mansi. SE(3)-hyena operator for scalable equivariant learning, 2024.
- [58] Khang Nguyen, Nong Minh Hieu, Vinh Duc Nguyen, Nhat Ho, Stanley Osher, and Tan Minh Nguyen. Revisiting over-smoothing and over-squashing using ollivier-ricci curvature. In *International Conference on Machine Learning*, pages 25956–25979. PMLR, 2023.
- [59] Mangal Prakash, Artem Moskalev, Peter DiMaggio Jr., Steven Combs, Tommaso Mansi, Justin Scheer, and Rui Liao. Bridging biomolecular modalities for knowledge transfer in bio-language models. In *Neurips 2024 Workshop Foundation Models for Science: Progress, Opportunities, and Challenges*, 2024.
- [60] Ladislav Rampásek, Mikhail Galkin, Vijay Prakash Dwivedi, Anh Tuan Luu, Guy Wolf, and Dominique Beaini. Recipe for a General, Powerful, Scalable Graph Transformer. *Advances in Neural Information Processing Systems*, 35, 2022.
- [61] Ugur Sahin, Katalin Karikó, and Özlem Türeci. mrna-based therapeutics—developing a new class of drugs. *Nature reviews Drug discovery*, 13(10):759–780, 2014.
- [62] Victor Garcia Satorras, Emiel Hooijboom, and Max Welling. E (n) equivariant graph neural networks. In *International conference on machine learning*, pages 9323–9332. PMLR, 2021.

- [63] Kristof Schütt, Pieter-Jan Kindermans, Huziel Enoc Saucedo Felix, Stefan Chmiela, Alexandre Tkatchenko, and Klaus-Robert Müller. Schnet: A continuous-filter convolutional neural network for modeling quantum interactions. *Advances in neural information processing systems*, 30, 2017.
- [64] Kristof Schütt, Oliver Unke, and Michael Gastegger. Equivariant message passing for the prediction of tensorial properties and molecular spectra. In *International Conference on Machine Learning*, pages 9377–9388. PMLR, 2021.
- [65] Tao Shen, Zhihang Hu, Zhangzhi Peng, Jiayang Chen, Peng Xiong, Liang Hong, Liangzhen Zheng, Yixuan Wang, Irwin King, Sheng Wang, et al. E2efold-3d: End-to-end deep learning method for accurate de novo rna 3d structure prediction. *arXiv preprint arXiv:2207.01586*, 2022.
- [66] Andrew Shrake and John A Rupley. Environment and exposure to solvent of protein atoms. lysozyme and insulin. *Journal of molecular biology*, 79(2):351–371, 1973.
- [67] Jiri Sponer, Giovanni Bussi, Miroslav Krepl, Pavel Banašš, Sandro Bottaro, Richard A Cunha, Alejandro Gil-Ley, Giovanni Pinamonti, Simón Poblete, Petr Jurecka, et al. Rna structural dynamics as captured by molecular simulations: a comprehensive overview. *Chemical reviews*, 118(8):4177–4338, 2018.
- [68] Samuel Stanton, Wesley Maddox, Nate Gruver, Phillip Maffettone, Emily Delaney, Peyton Greenside, and Andrew Gordon Wilson. Accelerating bayesian optimization for biological sequence design with denoising autoencoders. In *International conference on machine learning*, pages 20459–20478. PMLR, 2022.
- [69] 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.
- [70] Jan Tönshoff, Martin Ritzert, Eran Rosenbluth, and Martin Grohe. Where did the gap go? reassessing the long-range graph benchmark. *Transactions on Machine Learning Research*, 2024.
- [71] Raphael JL Townshend, Stephan Eismann, Andrew M Watkins, Ramya Rangan, Masha Karelina, Rhiju Das, and Ron O Dror. Geometric deep learning of rna structure. *Science*, 373(6558):1047–1051, 2021.
- [72] Max van Spengler, Artem Moskalev, Tommaso Mansi, Mangal Prakash, and Rui Liao. Hyperhelm: Hyperbolic hierarchy encoding for mrna language modeling, 2025.
- [73] Leven M Wadley, Kevin S Keating, Carlos M Duarte, and Anna Marie Pyle. Evaluating and learning from rna pseudotorsional space: quantitative validation of a reduced representation for rna structure. *Journal of molecular biology*, 372(4):942–957, 2007.
- [74] Hannah K Wayment-Steele, Wipapat Kladwang, Andrew M Watkins, Do Soon Kim, Bojan Tunguz, Walter Reade, Maggie Demkin, Jonathan Romano, Roger Wellington-Oguri, John J Nicol, et al. Deep learning models for predicting rna degradation via dual crowdsourcing. *Nature Machine Intelligence*, 4(12):1174–1184, 2022.
- [75] Qitian Wu, Wentao Zhao, Zenan Li, David Wipf, and Junchi Yan. Nodeformer: A scalable graph structure learning transformer for node classification. In *Advances in Neural Information Processing Systems*, 2022.
- [76] Qitian Wu, Wentao Zhao, Chenxiao Yang, Hengrui Zhang, Fan Nie, Haitian Jiang, Yatao Bian, and Junchi Yan. Sgformer: Simplifying and empowering transformers for large-graph representations. *Advances in Neural Information Processing Systems*, 36:64753–64773, 2023.
- [77] Zhanghao Wu, Paras Jain, Matthew Wright, Azalia Mirhoseini, Joseph E Gonzalez, and Ion Stoica. Representing long-range context for graph neural networks with global attention. *Advances in neural information processing systems*, 34:13266–13279, 2021.

- [78] Luis Wyss, Vincent Mallet, Wissam Karroucha, Karsten Borgwardt, and Carlos Oliver. A comprehensive benchmark for rna 3d structure-function modeling. *arXiv preprint arXiv:2503.21681*, 2025.
- [79] Junjie Xu, Enyan Dai, Dongsheng Luo, Xiang Zhang, and Suhang Wang. Shape-aware graph spectral learning. In *Proceedings of the 33rd ACM International Conference on Information and Knowledge Management*, pages 2692–2701, 2024.
- [80] Junjie Xu, Artem Moskalev, Tommaso Mansi, Mangal Prakash, and Rui Liao. Beyond sequence: Impact of geometric context for RNA property prediction. In *The Thirteenth International Conference on Learning Representations*, 2025.
- [81] Junjie Xu, Artem Moskalev, Tommaso Mansi, Mangal Prakash, and Rui Liao. HARMONY: A multi-representation framework for RNA property prediction. In *ICLR 2025 Workshop on AI for Nucleic Acids*, 2025.
- [82] Shuo Yan, Yuliang Yan, Bin Ma, Chenao Li, Haochun Tang, Jiahua Lu, Minhua Lin, Yuyuan Feng, Hui Xiong, and Enyan Dai. Protap: A benchmark for protein modeling on realistic downstream applications. *arXiv preprint arXiv:2506.02052*, 2025.
- [83] Mehdi Yazdani-Jahromi, Mangal Prakash, Tommaso Mansi, Artem Moskalev, and Rui Liao. HELM: Hierarchical encoding for mRNA language modeling. In *The Thirteenth International Conference on Learning Representations*, 2025.
- [84] Chengxuan Ying, Tianle Cai, Shengjie Luo, Shuxin Zheng, Guolin Ke, Di He, Yanming Shen, and Tie-Yan Liu. Do transformers really perform badly for graph representation? *Advances in neural information processing systems*, 34:28877–28888, 2021.
- [85] Yuelin Zhang, Jiacheng Cen, Jiaqi Han, Zhiqiang Zhang, Jun Zhou, and Wenbing Huang. Improving equivariant graph neural networks on large geometric graphs via virtual nodes learning. In *Forty-first International Conference on Machine Learning*.
- [86] Dongzhuoran Zhou, Evgeny Kharlamov, and Egor V. Kostylev. GLora: A benchmark to evaluate the ability to learn long-range dependencies in graphs. In *The Thirteenth International Conference on Learning Representations*, 2025.
- [87] Ruipeng Zhou, Ming Yang, Yi Li, Xin Zheng, Alan Wee-Chung Liew, Shirui Pan, and Yanqing Guo. Efficient and diverse de novo protein backbone design with se (3)-equivariant diffusion. In *Pacific-Asia Conference on Knowledge Discovery and Data Mining*, pages 408–420. Springer, 2025.

A Related Work

A.1 Invariant GNNs

Invariant GNNs learn molecular representations using rotation-invariant geometric features, ensuring predictions remain unchanged under rigid transformations. SchNet employs continuous-filter convolutions over interatomic distances to model molecular properties effectively [63]. DimeNet extends SchNet by integrating angular features, explicitly considering bond angles to capture directional information [30, 29, 42]. GemNet further enhances expressiveness by utilizing multi-hop distance–angle interactions, implicitly encoding higher-order geometric features [28]. SphereNet incorporates a comprehensive set of invariant scalars, including distances, angles, and torsion angles, to robustly represent molecular structures [50].

Despite their effectiveness, invariant GNNs inherently lose crucial directional and orientational information, motivating the development of equivariant models capable of explicitly encoding full geometric contexts.

A.2 Scalarization Equivariant GNNs

Scalarization equivariant graph neural networks maintain E(3) equivariance by explicitly handling vector-valued features through scalarized updates. EGNN [62] leverages pairwise distances to compute scalar-valued messages, enabling direct equivariant updates to atomic positions without high-order tensor operations. GVPGNN [39] introduces geometric vector perceptrons, managing scalar and vector embeddings separately and using scalar gating to achieve SO(3)-equivariant transformations. PaiNN [64] similarly utilizes scalarized vector updates, applying distance-based filters to coordinate features, thus ensuring consistent equivariant message passing.

These models share the common technique of scalarization, employing invariant scalars to guide separate, equivariant updates of vector or positional features. However, scalarization methods inherently lack the expressive power provided by high-degree spherical tensor representations, potentially limiting their ability to capture complex rotational relationships and subtle structural differences in molecular systems [40].

A.3 Higher-degree Equivariant GNNs

CG-based Tensor Product Several equivariant GNNs achieve expressivity by using high-order irreducible representations combined via Clebsch–Gordan tensor products. Tensor Field Networks [69] (TFN) introduced this paradigm by building SE(3)-equivariant convolutional layers from CG-based feature tensor products, enabling rotation-equivariant learning with theoretically maximal expressiveness at a significant computational cost. The SE(3)-Transformer [27] extended TFN by incorporating a self-attention mechanism on irreducible feature spaces, further improving representation power. SEGNN [12] construct streeable MLPs by using the tensor product so that formalize an equivariant message aggregation. MACE [9] uses high-order tensor features to form a complete basis of scalar invariants through the Atomic Cluster Expansion, capturing arbitrary many-body interactions. However, it also incurs significant overhead due to numerous CG tensor product computations.

Efficient Implementation Notably, the reliance on full CG tensor products leads to steep scaling in complexity (e.g. $O(L^6)$ for degree L features). Due to the high computation cost of CG-based tensor product, more recent approaches aim to retain high-degree expressivity in a more efficient way. HEGNN [16] incorporate higher-degree spherical harmonics features but using EGNN-like inner-product scalarization trick to handle cross-degree interactions. Similarly, GotenNet [6] introduces high-order spherical tensor in a unified attention architecture that eschews explicit CG transforms; instead, it updates features via inner products on steerable tensors. Our DualEqui also follows this paradigm: it combines spherical harmonic feature channels with dual scalarization steps to capture rich 3D geometric information while keeping computations tractable.

A.4 Long-Range Dependencies in GNNs

Learning long-range dependencies is a key challenge in graph machine learning, arising from the fundamental trade-off between under-reaching (i.e., insufficient GNN layers leading to small

receptive fields) and over-smoothing (i.e., excessive layers causing feature homogenization) [18, 58]. A variety of benchmarks [24, 70, 86, 79] have been proposed to study this problem, along with several representative solutions, including Implicit Graph Neural Networks (IGNNs) and Graph Transformers (GTs). IGNNs define message-passing through fixed-point equations [34, 49] or unfolded optimization steps [20, 8], allowing stable convergence to an equilibrium state even with infinite layers. Efforts to improve their scalability have focused on accelerating convergence [19, 8] and efficient neighbor sampling [48, 46], making IGNNs more practical for large-scale applications. GTs leverage global attention mechanisms [77, 84, 60] to capture long-range dependencies while incorporating structural encodings to preserve graph properties. Recent works have improved their scalability [75, 76, 52] and extended their applicability to directed graphs [32, 53], broadening their impact in graph learning.

B Basics of Group Theory, Equivariance, and Spherical Harmonics

In this section, we provide an introduction to the mathematical background used in this paper. For readers interested in more details, we refer them to the Guide to Geometric GNNs [23] and the e3nn documentation [31].

B.1 Groups and Symmetry

A group is an algebraic structure consisting of a set $\mathcal{G}$ together with a binary operation (composition) that satisfies four fundamental axioms:

- Closure: For any $a, b \in \mathcal{G}$, the composite $a \cdot b$ is also in $\mathcal{G}$.
- Identity: There is an identity element $e \in \mathcal{G}$ such that $e \cdot a = a \cdot e = a$ for every $a \in \mathcal{G}$.
- Inverse: For each $a \in \mathcal{G}$, there exists an inverse element $a^{-1} \in \mathcal{G}$ with $a \cdot a^{-1} = a^{-1} \cdot a = e$.
- Associativity: For all $a, b, c \in \mathcal{G}$, we have $(a \cdot b) \cdot c = a \cdot (b \cdot c)$.

Groups are often used to describe symmetries of objects or spaces. For example, the set of all rotations about a fixed point forms a group, as does the set of all translations in the plane. The symmetries of an object form a group under composition of transformations.

The Euclidean Group $E(3)$: In geometric deep learning, an important example is the Euclidean group in three dimensions, $E(3)$, which captures the symmetries of 3D Euclidean space (i.e., all rigid motions in $\mathbb{R}^3$). An element of $E(3)$ can be represented as a pair $(\mathbf{R}, \mathbf{t})$ where $\mathbf{R}$ is a 3×3 rotation matrix and $\mathbf{t} \in \mathbb{R}^3$ is a translation vector. Composition in $E(3)$ is done by composing rotations and adding translations: $(\mathbf{R}_1, \mathbf{t}_1) \cdot (\mathbf{R}_2, \mathbf{t}_2) = (\mathbf{R}_1 \mathbf{R}_2, \mathbf{R}_1 \mathbf{t}_2 + \mathbf{t}_1)$. This group is essentially the semi-direct product of translations and rotations. Intuitively, applying an $E(3)$ transformation to a point $\mathbf{x} \in \mathbb{R}^3$ yields $\mathbf{R}\mathbf{x} + \mathbf{t}$, which is just a rotated and translated version of the point. $E(3)$ is the symmetry group of any rigid 3D object, meaning that if an object (or a dataset) is moved or rotated in space, it remains essentially the “same” object – a concept captured by invariance or equivariance (discussed in B.2). Many physical or geometric tasks exhibit $E(3)$ symmetry. Hence, incorporating $E(3)$ into machine learning models is a powerful inductive bias [14, 43].

The Rotation Group $SO(3)$: The special orthogonal group $SO(3)$ is the subgroup of $E(3)$ consisting of rotations about the origin (with no translations). Formally, $SO(3)$ is the set of all 3×3 orthogonal matrices with determinant 1. Every element $\mathbf{R} \in SO(3)$ represents a rotation in 3D space. $SO(3)$ plays a central role in describing angular symmetries. For example, the orientation of an object in space can be changed by an element of $SO(3)$. $SO(3)$ is non-abelian (the order of rotations matters), unlike the translation sub-group (which is abelian). In many applications (like molecular modeling or 3D vision), only rotations need to be considered (e.g., when comparing shapes regardless of orientation), making $SO(3)$ a crucial symmetry group. Models that respect $SO(3)$ symmetry avoid to learn the same feature in all rotated forms and group-theoretic approaches provide a unified framework to incorporate such symmetries into neural networks.

B.2 Equivariance and Invariance in Geometric Learning

Equivariance to Euclidean transformations is a desirable property in geometric learning tasks involving 3D molecular structures. Specifically, a function $f: \mathcal{X} \rightarrow \mathcal{Y}$ is said to be *equivariant* with respect to

a group $\mathfrak{G}$ (e.g., the Euclidean group $E(3)$) if, for any transformation $\mathfrak{g} \in \mathfrak{G}$ and any input $\mathbf{x} \in \mathcal{X}$, there exists a corresponding transformation $\mathfrak{g}$ acting on the output space such that

$$f(\mathfrak{g} \cdot \mathbf{x}) = \mathfrak{g} \cdot f(\mathbf{x}). \quad (18)$$

For $E(3)$, this means that if the input (e.g., atomic coordinates or features) is rotated, translated, or reflected, the output of the model transforms in a consistent and predictable manner. This property ensures that the model's predictions are physically meaningful and invariant to arbitrary choices of coordinate systems—a key requirement for tasks involving 3D biomolecular data.

To achieve rotational equivariance in 3D, spherical harmonics are commonly used to represent directional features on the unit sphere. Spherical harmonics form an orthonormal basis for square-integrable functions defined on the sphere and serve as the irreducible representations of the rotation group $SO(3)$. A key property of these representations is their equivariant transformation under rotations. For a unit direction vector $\hat{\mathbf{x}}$ and a rotation matrix $\mathbf{R}$, the spherical harmonics basis of degree l transforms according to:

$$Y^{(l)}(\mathbf{R}\hat{\mathbf{x}}) = \mathcal{D}^{(l)}(\mathbf{R}) Y^{(l)}(\hat{\mathbf{x}}), \quad (19)$$

where $Y^{(l)}(\cdot)$ is the spherical harmonics of degree l , and $\mathcal{D}^{(l)}(\mathbf{R}) \in \mathbb{R}^{(2l+1) \times (2l+1)}$ is the Wigner D -matrix corresponding to rotation $\mathbf{R}$. This property enables the construction of rotation-equivariant functions by combining SH features with these bases, making them particularly suitable for encoding angular dependencies in molecular systems. We introduce more background in Appendix D.

Invariance vs. Equivariance: Invariant functions are a special case of equivariant functions. We say $f: \mathcal{X} \rightarrow \mathcal{Y}$ is invariant to $\mathfrak{G}$ if $f(\mathfrak{g} \cdot \mathbf{x}) = f(\mathbf{x})$ for all $\mathfrak{g} \in \mathfrak{G}$, i.e. the output is unchanged by the transformation. This corresponds to equation 18 where the $\mathfrak{G}$ -action on $\mathcal{Y}$ is trivial (identity for all $\mathfrak{g}$). Invariance discards all information about the transformation (losing equivariant detail), whereas equivariance preserves information in a structured way. Generally, equivariance is more powerful than invariance when building deep representations because it allows higher layers to reason about what transformation occurred, not just that something is symmetric.

B.3 SH Space Captures Global Information

Spherical Harmonics (SH) provide an orthonormal basis for functions defined on a sphere, similar to how the Fourier series represents periodic functions. In this section, we draw a parallel between Fourier decomposition in one dimension and SH expansion on the sphere, highlighting how increasing the expansion order refines function approximation and why SH captures global structural information.

B.3.1 Fourier Series vs. Spherical Harmonics

A well-known tool for analyzing periodic functions is the Fourier series, which decomposes a function into a sum of sinusoidal components. Given a periodic function $f(x)$ defined on $x \in [-\pi, \pi]$, its Fourier series expansion is:

$$f(x) = a_0 + \sum_{n=1}^{\infty} (a_n \cos(nx) + b_n \sin(nx)), \quad (20)$$

where the coefficients are computed as:

$$a_n = \frac{1}{\pi} \int_{-\pi}^{\pi} f(x) \cos(nx) dx, \quad b_n = \frac{1}{\pi} \int_{-\pi}^{\pi} f(x) \sin(nx) dx. \quad (21)$$

By truncating the expansion at order N , we obtain an approximation:

$$f_{\text{approx}}(x) = a_0 + \sum_{n=1}^N (a_n \cos(nx) + b_n \sin(nx)). \quad (22)$$

As $N \rightarrow \infty$, the approximation converges to the true function $f(x)$. In this representation:

- **Low-frequency terms** ($n = 1, 2$) capture smooth, large-scale structures.
- **High-frequency terms** ($n \gg 2$) refine the function by encoding local variations.

Analogously and Intuitively, SH expansion generalizes this idea to functions defined on the sphere. For a function $f(\theta, \phi)$ defined on the unit sphere, the **SH decomposition** is:

$$f(\theta, \phi) = \sum_{l=0}^{\infty} \sum_{m=-l}^l c_m^{(l)} Y_m^{(l)}(\theta, \phi), \quad (23)$$

where the coefficients are obtained via:

$$c_m^{(l)} = \int_{\mathbb{S}^2} f(\theta, \phi) Y_m^{(l)}(\theta, \phi) \sin \theta \, d\theta \, d\phi. \quad (24)$$

Truncating the expansion at order L provides an approximation:

$$f_{\text{approx}}(\theta, \phi) = \sum_{l=0}^L \sum_{m=-l}^l c_m^{(l)} Y_m^{(l)}(\theta, \phi). \quad (25)$$

As $L \rightarrow \infty$, this converges to the true function $f(\theta, \phi)$. The role of L in SH is similar to N in Fourier analysis:

- **Low-degree terms** ($l = 0, 1, 2$) capture smooth, global variations.
- **High-degree terms** ($l \gg 2$) encode fine-grained local details.

B.3.2 Why Spherical Harmonics Capture Global Information

Spherical harmonics enable long-range information encoding because they provide a global basis rather than relying on local neighborhoods. Unlike message-passing GNNs, which aggregates information only from spatially close neighbors, SH representations allow direct interactions between distant nodes that share similar SH coefficients.

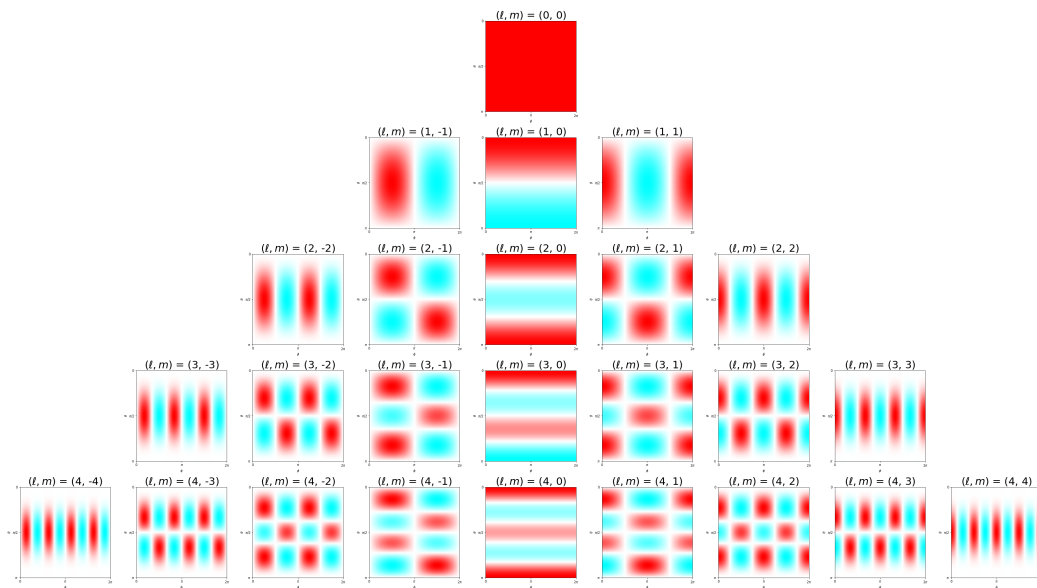


Figure 2: Visualization of real Spherical Harmonics table.¹

Figure 2 provides a visualization of the first few real-valued Spherical Harmonics. The figure illustrates how the functions oscillate across different degrees. Lower-degree harmonics (l) exhibit slower variations, while higher-degree harmonics demonstrate more rapid oscillations along the axes, reflecting their ability to capture different scales of geometric features. To be more specific, when

- $l = 0$ (**Monopole**): Captures the global molecular shape.

¹Image source: Wikipedia

- $l = 1$ (**Dipole**): Represents molecular orientation.
- $l = 2$ (**Quadrupole**): Encodes bond angles and torsional interactions.
- **Higher l values**: Capture localized atomic-scale interactions.

Since low-degree SH functions change gradually, functions dominated by low-degree SH terms exhibit long-range correlations, as information propagates smoothly across extended regions of the domain.

Mathematical Justification. The SH functions are eigenfunctions of the Laplace-Beltrami operator:

$$\Delta_{\mathbb{S}^2} Y_m^{(l)} = -l(l+1)Y_m^{(l)}. \quad (26)$$

Since the eigenvalues $-l(l+1)$ control how rapidly the functions oscillate, low-degree SH components ($l = 0, 1, 2$) vary slowly and thus retain long-range structural features.

SH-Based Neighborhoods Enable Long-Range Dependencies. In our framework, nodes interact based on the similarity of their spherical harmonics (SH) feature representations, independent of their Euclidean spatial distance. This enables two spatially distant nodes to have strong interactions if they share similar local geometric environments (e.g., comparable bond angles and torsions with their respective neighbors), as captured by their SH representations.

Traditional message passing in molecular graphs is inherently local, where information propagates only between spatially adjacent nodes. To capture long-range dependencies, these methods require multiple message passing steps, which can be computationally expensive and may lead to information dilution.

Our SH-based approach naturally encodes global structural dependencies while maintaining computational efficiency. This is possible because low-degree spherical harmonics have broad spatial support, creating smooth, global basis functions that span large regions of the molecular structure. As a result, nodes with similar SH coefficients maintain strong correlations regardless of their spatial separation.

This framework effectively captures long-range molecular interactions without requiring explicit long-range message passing, as the similarity in SH representations automatically establishes meaningful connections between distant nodes that share similar local geometric environments.

C Proof of Equivariance

In this section, we formally demonstrate the E(3)-equivariance property of our proposed DualEqui model. Specifically, we begin by proving the invariance property of the message functions, showing that they are built upon fundamental invariant quantities. Next, we analyze the forward computation of the DualEqui model and establish the equivariance property.

Notations. In this proof, we let $\mathbf{X} \in \mathbb{R}^{N \times 3}$ represent the set of all input 3D atomic coordinates, where each row $\mathbf{x}_i \in \mathbb{R}^3$ (for $i \in \{1, \dots, N\}$) corresponds to one specific atomic position within $\mathbf{X}$. We represent an arbitrary E(3) transformation using a rotation matrix $\mathbf{R} \in \mathbb{R}^{3 \times 3}$ and a translation vector $\mathbf{t} \in \mathbb{R}^3$.

For convenience, we let z be an arbitrary intermediate variable in our model computation (e.g., message $m_{EU,ij}$ in Eq. (5)). Occasionally, we adopt a functional notation, expressing it as a function of all input 3D coordinates, i.e., $z(\mathbf{X})$. This allows us to denote the model output under E(3) transformations as $z(\mathbf{R}\mathbf{X} + \mathbf{t})$ and verify whether equivariance or invariance holds. By default, we consider the original non-functional form of intermediate variables z as shorthand notation for $z(\mathbf{R}\mathbf{X} + \mathbf{t})$, where $\mathbf{R} = \mathbf{I}$ and $\mathbf{t} = 0$.

C.1 Basic Invariant Quantities

The equivariance of the DualEqui model is grounded in the invariance of several fundamental quantities. Our message functions in Eq. (5) and Eq. (10) are computed using two basic invariant quantities, whose invariance is shown below.

Euclidean Distance $\|\mathbf{x}_{ij}\|$. Let $\mathbf{x}_i, \mathbf{x}_j$ be any pair of 3D coordinate vectors. When an E(3) transformation is applied to both $\mathbf{x}_i$ and $\mathbf{x}_j$, the Euclidean distance is invariant due to the orthogonality of rotation matrices:

$$\begin{aligned}\|(\mathbf{R}\mathbf{x}_i + \mathbf{t}) - (\mathbf{R}\mathbf{x}_j + \mathbf{t})\| &= \|\mathbf{R}(\mathbf{x}_i - \mathbf{x}_j)\| \\ &= \sqrt{(\mathbf{x}_i - \mathbf{x}_j)^\top \mathbf{R}^\top \cdot \mathbf{R}(\mathbf{x}_i - \mathbf{x}_j)} \\ &= \|\mathbf{x}_i - \mathbf{x}_j\| \\ &= \|\mathbf{x}_{ij}\|.\end{aligned}$$

Spherical Harmonics Feature $\mathbf{r}_i \odot \mathbf{r}_j$. To show the invariance of $\mathbf{r}_i \odot \mathbf{r}_j$, we first consider the equivariance of an arbitrary $\mathbf{r}_i^l$ for node $i \in \{1, \dots, N\}$ and order $l \in \{0, \dots, l_{max}\}$ under E(3) transformations. Specifically, we have:

$$\begin{aligned}\mathbf{r}_i^{(l)}(\mathbf{R}\mathbf{X} + \mathbf{t}) &= \frac{1}{|\mathcal{N}_{EU}(i)|} \sum_{j \in \mathcal{N}_{EU}(i)} \phi_0([h_i, h_j, \|\mathbf{R}\mathbf{x}_{ij}\|]) \hat{Y}^l(\mathbf{R}\hat{\mathbf{x}}_{ij}) \\ &= \frac{1}{|\mathcal{N}_{EU}(i)|} \sum_{j \in \mathcal{N}_{EU}(i)} \phi_0([h_i, h_j, \|\mathbf{x}_{ij}\|]) D^{(l)}(\mathbf{R}) \hat{Y}^l(\hat{\mathbf{x}}_{ij}) \\ &= D^{(l)}(\mathbf{R}) \cdot \left(\frac{1}{|\mathcal{N}_{EU}(i)|} \sum_{j \in \mathcal{N}_{EU}(i)} \phi_0([h_i, h_j, \|\mathbf{x}_{ij}\|]) \hat{Y}^l(\hat{\mathbf{x}}_{ij}) \right) \\ &= D^{(l)}(\mathbf{R}) \mathbf{r}_i^{(l)},\end{aligned}\tag{27}$$

where $D^{(l)}(\mathbf{R})$ denotes the l -th order Wigner-D matrix, which denotes the irreducible representation of the rotation group SO(3). Specifically, all the Wigner-D matrices are unitary, i.e., $D^{(l)}(\mathbf{R})^\top D^{(l)}(\mathbf{R}) = \mathbf{I}$.

Thus, since Eq. (3) already establishes that

$$\mathbf{r}_i = [\mathbf{r}_i^{(0)}, \mathbf{r}_i^{(1)}, \dots, \mathbf{r}_i^{(l_{max})}].$$

We can then directly obtain

$$\mathbf{r}_i(\mathbf{R}\mathbf{X} + \mathbf{t}) = [D^{(0)}(\mathbf{R})\mathbf{r}_i^{(0)}, D^{(1)}(\mathbf{R})\mathbf{r}_i^{(1)}, \dots, D^{(l_{max})}(\mathbf{R})\mathbf{r}_i^{(l_{max})}].$$

Therefore, Since the Wigner-D matrices are unitary, the invariance of $\mathbf{r}_i \odot \mathbf{r}_j$ follows:

$$\begin{aligned}&\mathbf{r}_i(\mathbf{R}\mathbf{X} + \mathbf{t}) \odot \mathbf{r}_j(\mathbf{R}\mathbf{X} + \mathbf{t}) \\ &= \left[\left(\mathbf{r}_i^{(0)} D^{(0)}(\mathbf{R}) \right)^\top \cdot \left(D^{(0)}(\mathbf{R}) \mathbf{r}_j^{(0)} \right), \dots, \left(\mathbf{r}_i^{(l_{max})} D^{(l_{max})}(\mathbf{R}) \right)^\top \cdot \left(D^{(l_{max})}(\mathbf{R}) \mathbf{r}_j^{(l_{max})} \right) \right] \\ &= \left[(\mathbf{r}_i^{(0)})^\top \mathbf{r}_j^{(0)}, \dots, (\mathbf{r}_i^{(l_{max})})^\top \mathbf{r}_j^{(l_{max})} \right] \\ &= \mathbf{r}_i \odot \mathbf{r}_j.\end{aligned}$$

C.2 Invariance of Message Functions

Based on the invariance of the above quantities, the invariance of the message functions follows directly from their definitions. Recall Eq. (5) and Eq. (10):

$$\begin{aligned}m_{EU,ij} &:= \phi_{EU}([h_i, h_j, \|\mathbf{x}_{ij}\|]), \\ m_{SH,ij} &:= \phi_{EU}([h_i, h_j, \mathbf{r}_i \odot \mathbf{r}_j]), \\ m_{EU \rightarrow SH,ij} &:= \psi_{EU \rightarrow SH}([\mathbf{r}_i \odot \mathbf{r}_j, \|\mathbf{x}_{ij}\|]), \\ m_{SH \rightarrow E,ij} &:= \psi_{SH \rightarrow EU}([\mathbf{r}_i \odot \mathbf{r}_j, \|\mathbf{x}_{ij}\|]).\end{aligned}$$

Since h_i and h_j are transformation-invariant scalars, and all the message functions use only invariant quantities ($\|\mathbf{x}_{ij}\|$ and $\mathbf{r}_i \odot \mathbf{r}_j$), we conclude that all the message functions are invariant.

C.3 Equivariance of the DualEqui Model

We now prove that the entire DualEqui model is equivariant with respect to E(3) transformations.

Initialization. In DualEqui, each node i is represented by two types of coordinate vectors: $\mathbf{x}_i$ in Euclidean space and $\mathbf{r}_i$ in Spherical Harmonics space. The equivariance of $\mathbf{x}_i$ is immediate, and, as shown in Eq. (27), each $\mathbf{r}_i^l$ is equivariant. Hence, the initialized node representations are equivariant.

Euclidean Space Update. In Euclidean space, we update the scalar features h_i and the position vector $\mathbf{x}_i$ using updates $\Delta h_{EU,i}$ and $\Delta \mathbf{x}_{EU,i}$. The scalar update is given by Eq. (8):

$$\Delta h_{EU,i} := \frac{1}{K} \sum_{k=1}^K \phi_{upd,EU} \left(\left[h_i, \bigoplus_{j \in \mathcal{N}_{EU}(i)} \alpha_{EU,ij}^k \phi_{EU,h}^k(m_{EU,ij}) + \bigoplus_{j \in \mathcal{N}_{SH}(i)} \beta_{EU,ij}^k \psi_{EU,h}^k(m_{SH \rightarrow EU,ij}) \right] \right),$$

with attention scores defined as:

$$\alpha_{EU,ij}^k := \sigma(\phi_{EU,att}^k(m_{EU,ij})), \quad \beta_{EU,ij}^k := \sigma(\psi_{EU,\beta}^k(m_{SH \rightarrow E,ij})).$$

Since these computations involve only invariant quantities, $\Delta h_{EU,i}$ is invariant.

Similarly, the position update from Eq. (9) is:

$$\Delta \mathbf{x}_{EU,i} = \frac{1}{K} \sum_{k=1}^K \left(\bigoplus_{j \in \mathcal{N}_{EU}(i)} \alpha_{EU,ij}^k \phi_{EU,x}^k(m_{EU,ij}) \mathbf{x}_{ij} + \bigoplus_{j \in \mathcal{N}_{SH}(i)} \beta_{EU,ij}^k \psi_{EU,x}^k(m_{SH \rightarrow E,ij}) \mathbf{x}_{ij} \right).$$

Under an E(3) transformation, the scalar weights are invariant and the aggregation $\bigoplus$ (i.e., sum or average) is linear, hence:

$$\Delta \mathbf{x}_{EU,i}(\mathbf{R}\mathbf{X} + \mathbf{t}) = \mathbf{R}\Delta \mathbf{x}_{EU,i},$$

which proves that $\Delta \mathbf{x}_{EU,i}$ is equivariant.

Spherical Harmonics Space Update. In Spherical Harmonics space, we update the scalar features h_i with $\Delta h_{SH,i}$ and then update the Spherical Harmonics features $\mathbf{r}_i$ with $\Delta \mathbf{r}_{SH,i}$. The scalar update is defined in Eq. (11) as:

$$\Delta h_{SH,i} := \frac{1}{K} \sum_{k=1}^K \phi_{upd,SH} \left(\left[h_i, \bigoplus_{j \in \mathcal{N}_{SH}(i)} \alpha_{SH,ij}^k \phi_{SH,h}^k(m_{SH,ij}) + \bigoplus_{j \in \mathcal{N}_{EU}(i)} \beta_{SH,ij}^k \psi_{SH,h}^k(m_{EU \rightarrow SH,ij}) \right] \right),$$

where the attention scores are defined as:

$$\alpha_{SH,ij}^k = \sigma(\phi_{SH,att}^k(m_{SH,ij})), \quad \beta_{SH,ij}^k = \sigma(\psi_{SH,\beta}^k(m_{EU \rightarrow SH,ij})).$$

As these updates involve only invariant message functions, $\Delta h_{SH,i}$ is invariant.

For the Spherical Harmonics feature update, we have from Eq. (12):

$$\Delta \mathbf{r}_{SH,i} = \frac{1}{K} \sum_{k=1}^K \left(\bigoplus_{j \in \mathcal{N}_{SH}(i)} \alpha_{SH,ij}^k \phi_{SH,r}^k(m_{SH,ij}) \mathbf{r}_j + \bigoplus_{j \in \mathcal{N}_{EU}(i)} \beta_{SH,ij}^k \psi_{SH,r}^k(m_{EU \rightarrow SH,ij}) \hat{Y}(\hat{\mathbf{x}}_{ij}) \right),$$

where both $\mathbf{r}_j$ and $\hat{Y}(\hat{\mathbf{x}}_{ij})$ are equivariant, transforming under the block diagonal concatenation of all Wigner-D matrices $[D^{(0)}(\mathbf{R}), D^{(1)}(\mathbf{R}), \dots, D^{(l_{max})}(\mathbf{R})]$. Therefore,

$$\Delta \mathbf{r}_{SH,i}(\mathbf{R}\mathbf{X} + \mathbf{t}) = [D^{(0)}(\mathbf{R}), D^{(1)}(\mathbf{R}), \dots, D^{(l_{max})}(\mathbf{R})] \Delta \mathbf{r}_{SH,i},$$

which shows that $\Delta \mathbf{r}_{SH,i}$ is equivariant.

Cross-space Interaction Pooling. In the cross-space interaction pooling stage, we use a biomolecular structure-aware hierarchical pooling strategy to aggregate nodes within a cluster C in both Euclidean space ($\mathbf{x}'$) and Spherical Harmonics space ($\mathbf{r}'$), along with scalar features h' . Since the scalar features are invariant, we focus on the pooling in both vector spaces. For the Euclidean space pooling in Eq. (15), we have:

$$\begin{aligned}
\mathbf{x}'(\mathbf{R}\mathbf{X} + \mathbf{t}) &= \frac{1}{|C|} \sum_{i \in C} \alpha_i (\mathbf{R}\mathbf{x}_i + \gamma \cdot \text{Proj}_{SH}(\mathbf{r}_i(\mathbf{R}\mathbf{X} + \mathbf{t}))) \\
&= \frac{1}{|C|} \sum_{i \in C} \alpha_i \left(\mathbf{R}\mathbf{x}_i + \gamma \cdot \sum_{l=1}^L w_l \|D^{(l)}(\mathbf{R})\mathbf{r}_i^l\| \mathbf{R}\mathbf{x}_i \right) \\
&= \frac{1}{|C|} \sum_{i \in C} \alpha_i \left(\mathbf{R}\mathbf{x}_i + \gamma \cdot \sum_{l=1}^L w_l \|\mathbf{r}_i^l\| \mathbf{R}\mathbf{x}_i \right) \\
&= \mathbf{R} \cdot \frac{1}{|C|} \left(\sum_{i \in C} \alpha_i \left(\mathbf{x}_i + \gamma \cdot \sum_{l=1}^L w_l \|\mathbf{r}_i^l\| \mathbf{x}_i \right) \right) \\
&= \mathbf{R}\mathbf{x}',
\end{aligned}$$

where $D^{(l)}(\mathbf{R})$ denotes the l -order Wigner-D matrix.

Similarly, for the Spherical Harmonics space pooling in Eq. (16):

$$\begin{aligned}
\mathbf{r}'(\mathbf{R}\mathbf{X} + \mathbf{t}) &= \frac{1}{|C|} \sum_{i \in C} \alpha_i (\mathbf{r}_i(\mathbf{R}\mathbf{X} + \mathbf{t}) + \epsilon \cdot \text{Proj}_{EU}(\mathbf{R}\mathbf{x}_i + \mathbf{t})) \\
&= \frac{1}{|C|} \sum_{i \in C} \alpha_i \left(\mathbf{r}_i(\mathbf{R}\mathbf{X} + \mathbf{t}) + \epsilon \cdot \hat{Y}(\mathbf{R}(\mathbf{x}_i - \mathbf{x}_{\text{avg}})) \right) \\
&= \frac{1}{|C|} \sum_{i \in C} \alpha_i \left(D(\mathbf{R})\mathbf{r}_i + \epsilon \cdot D(\mathbf{R})\hat{Y}(\mathbf{x}_i - \mathbf{x}_{\text{avg}}) \right) \\
&= D(\mathbf{R}) \cdot \frac{1}{|C|} \left(\sum_{i \in C} \alpha_i \left(\mathbf{r}_i + \epsilon \cdot \hat{Y}(\mathbf{x}_i - \mathbf{x}_{\text{avg}}) \right) \right) \\
&= D(\mathbf{R})\mathbf{r}',
\end{aligned}$$

where $D(\mathbf{R})$ denotes the block diagonal concatenation of all Wigner-D matrices, i.e., $D(\mathbf{R}) := [D^{(0)}(\mathbf{R}), D^{(1)}(\mathbf{R}), \dots, D^{(l_{\text{max}})}(\mathbf{R})]$. Specifically, $\mathbf{x}_{\text{avg}} := \frac{1}{|C|} \sum_{i \in C} \mathbf{x}_i$ in the equation above denotes the center of all input 3D coordinates, and preserves the translation invariance.

To summarize, we have shown that all node position vectors $\mathbf{x}_i$ and $\mathbf{r}_i$, and their updates, are equivariant, and that the pooling operations preserve this equivariance. Thus, traversing the entire model structure of DualEqui, we conclude that DualEqui is equivariant with respect to E(3) transformations.

D More Experiments and Details

D.1 rnaglib Datasets Performance

We conduct additional experiments on the RNAGo and RNAProt datasets from rnaglib [78], shown in Table 6. DualEquiNet achieves state-of-the-art performance on the datasets, further validating its effectiveness.

D.2 Comparison with Long-range Baselines

To separate the performance gains and provide stronger evidence of the effectiveness of the proposed method, we compare the performance with models that are made for long-range tasks and also use a dual-space design. Specifically, we select Neural Atoms [47] and Ewald-based message passing [44] and show their performance in Table 7.

Table 6: Performance comparison on RNAGo and RNAProt datasets.

Model	RNAGo (Jaccard Index $\uparrow$)	RNAProt (Accuracy $\uparrow$)
SchNet	0.613 $\pm$ 0.020	0.709 $\pm$ 0.005
EGNN	0.646 $\pm$ 0.060	0.720 $\pm$ 0.009
GVPGNN	0.632 $\pm$ 0.048	0.710 $\pm$ 0.005
TFN	0.651 $\pm$ 0.185	0.697 $\pm$ 0.005
FastEGNN	0.527 $\pm$ 0.159	0.718 $\pm$ 0.010
HEGNN	0.661 $\pm$ 0.038	0.735 $\pm$ 0.016
GotenNet	0.555 $\pm$ 0.170	0.709 $\pm$ 0.006
DualEquiNet	0.681$\pm$0.047	0.749$\pm$0.015

Table 7: Performance comparison with long-range baselines. (RMSE $\pm$ std.)

Method	COVID					Ribonanza		Tc-Ribo
	Reactivity	pH10	Mg pH10	Avg.	DMS	2A3	Avg.	Factor
SchNet + Ewald	0.392 $\pm$ 0.004	0.583 $\pm$ 0.010	0.459 $\pm$ 0.008	0.478 $\pm$ 0.009	0.845 $\pm$ 0.020	0.884 $\pm$ 0.002	0.865 $\pm$ 0.014	0.736 $\pm$ 0.004
SchNet + Neural Atom	0.409 $\pm$ 0.004	0.596 $\pm$ 0.009	0.481 $\pm$ 0.006	0.495 $\pm$ 0.008	0.854 $\pm$ 0.015	0.919 $\pm$ 0.009	0.886 $\pm$ 0.012	0.726 $\pm$ 0.002
Dimnet + Ewald	0.403 $\pm$ 0.005	0.594 $\pm$ 0.006	0.480 $\pm$ 0.004	0.492 $\pm$ 0.005	0.828 $\pm$ 0.011	0.896 $\pm$ 0.011	0.862 $\pm$ 0.003	0.676 $\pm$ 0.044
Dimnet + Neural Atom	0.411 $\pm$ 0.009	0.602 $\pm$ 0.016	0.489 $\pm$ 0.013	0.501 $\pm$ 0.015	0.825 $\pm$ 0.010	0.892 $\pm$ 0.013	0.859 $\pm$ 0.005	0.679 $\pm$ 0.006
DualEquiNet	0.272$\pm$0.004	0.448$\pm$0.010	0.340$\pm$0.005	0.353$\pm$0.006	0.611$\pm$0.025	0.505$\pm$0.024	0.558$\pm$0.012	0.636$\pm$0.049

D.3 Neighborhood Analysis

To better understand how DualEqui captures both local and global geometric contexts, we analyze how neighborhoods evolve across layers in the EU and SH spaces. We use the experiment of DualEqui with 4 atom layers on the Covid Vaccine dataset as a case study. At each layer, EU and SH neighborhoods are dynamically redefined based on updated 3D coordinates and SH features, respectively. We compute 4 statistics:

- (1) the average distance of coordinates between each node and its EU neighbors;

$$dis_{EU} = \frac{1}{N_{atom}} \sum_{i=1}^{N_{atom}} \frac{1}{|\mathcal{N}_{EU}(i)|} \sum_{j \in \mathcal{N}_{EU}(i)} \|\mathbf{x}_i - \mathbf{x}_j\| \quad (28)$$

- (2) the average coordinates distance between each node and its SH neighbors;

$$dis_{SH} = \frac{1}{N_{atom}} \sum_{i=1}^{N_{atom}} \frac{1}{|\mathcal{N}_{SH}(i)|} \sum_{j \in \mathcal{N}_{SH}(i)} \|\mathbf{x}_i - \mathbf{x}_j\| \quad (29)$$

- (3) the average cosine similarity of SH features between each node and its EU neighbors;

$$cos_{EU} = \frac{1}{N_{atom}} \sum_{i=1}^{N_{atom}} \frac{1}{|\mathcal{N}_{EU}(i)|} \sum_{j \in \mathcal{N}_{EU}(i)} \frac{\mathbf{r}_i \cdot \mathbf{r}_j}{\|\mathbf{r}_i\| \cdot \|\mathbf{r}_j\|} \quad (30)$$

- (4) the average cosine similarity of SH features between each node and its SH neighbors.

$$cos_{SH} = \frac{1}{N_{atom}} \sum_{i=1}^{N_{atom}} \frac{1}{|\mathcal{N}_{SH}(i)|} \sum_{j \in \mathcal{N}_{SH}(i)} \frac{\mathbf{r}_i \cdot \mathbf{r}_j}{\|\mathbf{r}_i\| \cdot \|\mathbf{r}_j\|} \quad (31)$$

The distances of coordinates are plotted in Fig. 3(a), and the cosine similarity of SH features are plotted in Fig. 3(b). Results reveal that the average distance of EU neighbors (dis_{EU}) and the SH features cosine similarity of SH neighbors (cos_{SH}) remain stable across layers. This behavior is expected since EU neighborhoods are constructed using a fixed cutoff d_{EU} based on spatial proximity, while SH neighborhoods are selected using a threshold d_{SH} on SH feature cosine similarity. In this experiment, we use $d_{EU} = 4.8$ and $d_{SH} = 0.97$.

In contrast, we observe clear trends in the other two metrics. The average coordinates distance of SH neighbors (dis_{SH}) decreases significantly over layers (Fig.3(a), orange line), indicating that nodes

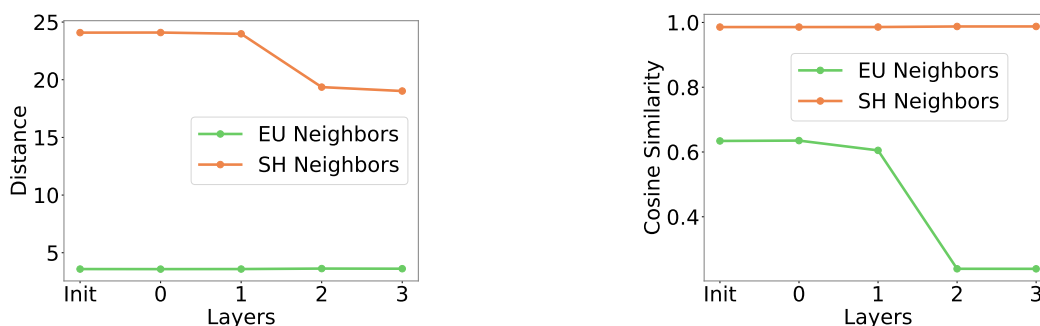


Figure 3: (a) Left: Average distance variation of EU and SH neighborhoods across layers. (b) Right: Average similarity variation of EU and SH neighborhoods across layers.

with similar SH features—which initially may be far apart in EU space—gradually become closer in Euclidean distance. This trend reflects the impact of the cross-space update in Eq.9 and 12. When we update the SH features r , we also consider the EU coordinates, which brings distant SH neighbors closer in EU space. In practice, we can imagine the scenario in biomolecules where spatially distant residues can interact through folding.

Meanwhile, the SH features cosine similarity of EU neighbors ($\cos_{EU}$) drops across layers (Fig.3(b), green line), implying that while EU neighbors may remain close in EU space, their SH features increasingly diverge. According to Eq.2, the SH neighbors are initialized according to the EU neighbors at first and therefore have a higher similarity score. But after several layers, the SH features are updated to differentiate local nodes based on their broader geometric context or functional roles. This is relevant in structured biomolecules where nearby atoms may belong to different residues or to distinct structural regions.

Thus, the observed dynamics validate the model’s dual-space architecture: SH space helps bridge distant but functionally related parts of the molecule, while EU space preserves local geometry but allows for semantic diversity.

D.4 Expressivity on Rotationally Symmetric Structures

An L -fold symmetric structure is one that remains unchanged under rotation by $360^\circ/L$, exhibiting identical appearance at each of L evenly spaced angles around a central axis. Joshi et al. [40] demonstrate that spherical tensors of order L cannot discern the orientation of structures with a rotational symmetry greater than L -fold.

Table 8 presents the performance of various models on the L -fold symmetry classification task for $L = 2, 3, 5$, and 10. As expected, SchNet, EGNN, and FastEGNN fail to solve the task across all settings due to their lack of higher-order spherical tensors. TFN and HEGNN exhibit correct behavior when equipped with sufficiently high spherical harmonic order (e.g., $l_{\max} \geq 2$), aligning with theoretical expectations from GWL test.

DualEqui achieves similarly expressive results. When $l_{\max} \geq 2$, it consistently attains 100% accuracy when $L \leq l_{\max}$, matching the expressive power of TFN/HEGNN.

D.5 Computational Efficiency Analysis

To assess the computational efficiency of DualEquiNet, we compare the training time per epoch and memory usage across several baseline models, as shown in Table 9. While DualEquiNet introduces additional overhead due to its dual-space message passing and spherical harmonics computations, it remains significantly more efficient than high-order tensor-based models like TFN, with training time comparable to FastEGNN and GotenNet. This demonstrates that our design strikes a favorable balance between computational cost and model expressivity, enabling scalable training on large biomolecular systems.

Table 8: The expressivity in the L -fold task with $L = 2, 3, 5, 10$. Anomalous results are highlighted in red and expected in green. DualEqui using the high-order spherical harmonic features is as expressive as TFN and HEGNN.

Folds $L =$	2	3	5	10
SchNet	50.0±0.0	50.0±0.0	50.0±0.0	50.0±0.0
EGNN	50.0±0.0	50.0±0.0	50.0±0.0	50.0±0.0
GVPINN	91.5±18.8	50.5±5.0	66.5±23.5	50.0±0.0
FastEGNN	50.0±0.0	50.0±0.0	50.0±0.0	50.0±0.0
TFN/HEGNN $l_{max} = 1$	50.0±0.0	50.0±0.0	50.0±0.0	50.0±0.0
TFN/HEGNN $l_{max} = 2$	100.0±0.0	50.0±0.0	50.0±0.0	50.0±0.0
TFN/HEGNN $l_{max} = 3$	100.0±0.0	100.0±0.0	50.0±0.0	50.0±0.0
TFN/HEGNN $l_{max} = 5$	100.0±0.0	100.0±0.0	100.0±0.0	50.0±0.0
TFN/HEGNN $l_{max} = 10$	100.0±0.0	100.0±0.0	100.0±0.0	100.0±0.0
DualEqui $l_{max} = 1$	50.5±5.0	50.0±0.0	50.0±0.0	50.5±5.0
DualEqui $l_{max} = 2$	100.0±0.0	50.0±0.0	63.5±22.2	95.5±14.3
DualEqui $l_{max} = 3$	100.0±0.0	100.0±0.0	55.0±15.0	60.0±20.0
DualEqui $l_{max} = 5$	100.0±0.0	100.0±0.0	100.0±0.0	50.0±0.0
DualEqui $l_{max} = 10$	100.0±0.0	100.0±0.0	100.0±0.0	100.0±0.0

Table 9: Training time and memory usage of each model.

Model	Train Time (s/epoch)	Memory (MB)
SchNet	4.361	628
EGNN	6.240	608
GVPINN	11.172	10160
TFN	56.235	43970
FastEGNN	11.755	1954
HEGNN	6.320	1232
GotenNet	10.332	5894
DualEqui	10.990	2436

D.6 Datasets Details

COVID Vaccine The OpenVaccine COVID-19 dataset [74] consists of 4,082 synthetic RNA sequences, each 107-130 nucleotides long. It was developed to support the predictive modeling of RNA stability for mRNA vaccine development. Each nucleotide is annotated with three quantitative properties: reactivity, degradation under pH10, and degradation under Mg pH10 conditions. These measurements reflect how susceptible each nucleotide is to chemical degradation and serve as surrogates for RNA structural stability. Sequences with low signal-to-noise ratio ($\text{SNR} < 1$) are filtered out to ensure label reliability, following prior best practices. This dataset is especially suited for testing fine-grained, residue-level predictions. In our experiments, we infer the 3D atomic structure for each sequence using RhoFold [65].

Ribonanza The Ribonanza-2k dataset [37] includes 2,260 RNA sequences, each around 177 nucleotides in length, annotated at the nucleotide level with reactivity values obtained from two chemical probing agents: DMS and 2A3. These chemical modifications reflect the accessibility and flexibility of each nucleotide, thereby encoding aspects of RNA's secondary and tertiary structures. The labels thus capture nuanced differences in base-pairing and folding landscapes across diverse RNA sequences, making the dataset a benchmark for modeling structural flexibility.

Tc-Riboswitches The Tc-Riboswitches dataset [33] consists of 355 mRNA sequences, each 67-73 nucleotides long, paired with graph-level regression labels reflecting gene regulatory behavior in response to tetracycline binding. Specifically, the label represents the expression shift induced by the presence of a tetracycline switching factor, an RNA-based synthetic regulatory system widely

Table 10: Dataset Statistics.

	COVID	Ribonanza	Tc-Riboswitches	RNASolo SASA	mRFP Protein SASA	TorsionAngle
Task Level	residue-level reactivity & degradation	residue-level reactivity	graph-level switching factor	residue-level SASA Value	graph-level SASA Value	residue-level torsion angles
Target						
# Sequences	4082	2260	355	1018	718	1018
Sequence Length	107 - 130	177	66 - 75	10 - 400	216 - 217	10 - 400
# Labels	3	2	1	1	1	7
# Avg. Atoms	363	531	216	249	649	249

studied in gene therapy and synthetic biology. Unlike COVIDVaccine and Ribonanza, which have residue-level labels, TcRibo focuses on sequence-level prediction, allowing evaluation of how local structure aggregates into global function.

RNASolo SASA The SASA dataset consists of RNA structures annotated with Solvent-Accessible Surface Area (SASA) values for each nucleotide. SASA quantifies the extent to which each nucleotide is exposed to solvent, providing important insights into RNA folding, stability, and interaction potential. We compute per-nucleotide SASA values by aggregating atomic-level SASA from experimentally resolved PDB structures in the RNASolo database [2]. These values are extracted using DSSR, which internally implements the Shrake and Rupley algorithm [66]. This method models solvent accessibility by rolling a spherical probe over the van der Waals surface of each atom, and then estimating the exposed area based on point sampling. The resulting SASA values reflect the 3D packing and surface topology of RNA molecules. Unlike earlier tasks focused on sequence-derived stability, RNASolo-SASA emphasizes the role of 3D structural precision and surface geometry. The dataset includes a diverse set of RNA structures with variable lengths and topologies.

mRFP Protein SASA We utilize the mRFP protein dataset introduced by Stanton et al.[68] for the task of SASA prediction. This dataset was designed as part of a realistic large-molecule sequence design benchmark, aiming to simulate practical protein engineering objectives, consisting of approximately 200-residue red fluorescent proteins (RFPs), where each protein sequence is annotated with ground-truth SASA values. We randomly select 718 proteins and the ground-truth 3D structures are predicted by AlphaFold2 [41].

TorsionAngle The TorsionAngle dataset is designed to assess a model’s ability to recover fine-grained structural geometry by predicting residue-level torsion angles, including backbone and glycosidic dihedrals. These angles—such as α , β , γ , δ , ϵ , ζ , and χ —are crucial descriptors of RNA conformation and govern higher-order folding. We extract these angles from RNA structures in the RNASolo database also using the DSSR, which computes torsions directly from atomic coordinates. Each sample in the dataset contains the full 3D atomic structure and a set of seven torsion angles per nucleotide. Because RNA torsions are sensitive to both local backbone rigidity and global conformation, this dataset provides a challenging test for models to capture both short-range and long-range structural dependencies. All angles are normalized into the range $[-180^\circ, 180^\circ]$, and training uses a circular MAE loss described in Section 4.4.

Here, we summarize all the datasets in Table 10.

D.7 Comparison of Baselines

SchNet is an invariant GNN using continuous-filter convolutional layers based on pairwise distances expanded in radial bases. It employs only scalar features, making predictions rotation- and translation-invariant. It lacks higher-order directional features and handles long-range interactions primarily through stacking layers or increasing cutoff distances.

EGNN achieves E(3)-equivariance by scalarizing messages computed from pairwise distances and features to scale relative position vectors, updating Cartesian coordinates. Unlike SchNet, it explicitly uses first-order geometric (Cartesian) features, capturing directional information. Long-range interactions are possible via dense connections, though it can become computationally expensive for large systems.

GVP-GNN maintains separate scalar and Cartesian vector features, updating them with an equivariant vector perceptron that couples vector updates with scalar invariants. This ensures SE(3)-equivariance and improves geometric expressiveness over EGNN. It remains constrained to first-order Cartesian vectors and, like EGNN, relies on standard message passing for long-range information.

TFN utilizes spherical harmonics and Clebsch–Gordan tensor products to explicitly represent higher-order geometric correlations, achieving rigorous SE(3)-equivariance. While highly expressive, TFN’s complexity is significantly greater than scalar-based methods. It primarily captures local interactions, with long-range interactions requiring deep stacking or additional global modules.

FastEGNN extends EGNN by incorporating virtual nodes that aggregate and distribute global information, enhancing long-range communication without substantially increasing computational complexity. It preserves EGNN’s E(3)-equivariance and Cartesian features, improving efficiency on larger or more distant interactions while maintaining EGNN’s directional limitations.

HEGNN enhances EGNN’s framework by integrating higher-order spherical harmonic features (beyond first-order vectors). It scalarizes tensor interactions for computational efficiency, achieving improved geometric expressiveness. While computationally manageable, it remains primarily focused on local interactions unless combined with specialized long-range mechanisms.

GotenNet employs geometry-aware attention with learned higher-order Cartesian tensors, avoiding explicit spherical harmonic computations while maintaining E(3)-equivariance. Its attention mechanism efficiently captures long-range interactions and high-order geometric patterns, combining the benefits of expressive tensors with scalable global communication.

Table 11 summarizes key properties of baseline models in terms of equivariance, feature representation, spherical harmonic usage, and ability to model long-range interactions. While many recent models achieve E(3)-equivariance, they differ in expressivity and scalability. Invariant models like SchNet rely solely on scalar features and lack the capacity to model directional or long-range dependencies. Scalarization-based equivariant models such as EGNN and GVP-GNN use Cartesian vectors but are limited to local neighborhoods. High-order methods like TFN offer expressive SH-based tensors but suffer from inefficient tensor product calculations. In contrast, HEGNN, GotenNet, and DualEqui provide efficient SH updates, with DualEqui and FastEGNN uniquely supporting long-range modeling. DualEqui stands out for combining spherical representations with long-range communication, enabled by its cross-space design.

Table 11: Comparison of baseline geometric GNN models. “Long-Range” indicates whether the model has a built-in mechanism to capture long-range interactions efficiently. Designs with higher expressivity or efficiency are highlighted in green.

Model	Equivariance	Feature Type	Efficient SH Update	Long-Range
SchNet	Invariant	Scalar		×
EGNN	Equivariant	Cartesian		×
GVP-GNN	Equivariant	Cartesian		×
TFN	Equivariant	Spherical	×	×
FastEGNN	Equivariant	Cartesian		✓
HEGNN	Equivariant	Spherical	✓	×
GotenNet	Equivariant	Spherical	✓	×
DualEqui	Equivariant	Spherical	✓	✓

D.8 Reproducibility

All experiments in this work are conducted under unified and controlled conditions to ensure fair comparison and reproducibility. For each dataset, we adopt an 8:1:1 split for training, validation, and testing. All reported results are averaged over 5 independent random splits, with standard deviations included. For each run, the best-performing model is selected based on validation set performance and evaluated on the test set. Our implementation is built on PyTorch and PyTorch Geometric [25]. All code and datasets will be publicly available upon acceptance.

We apply consistent hyperparameter tuning for all models—including baselines and our DualEqui—using the Optuna [3] framework with 100 trials per method. Search spaces and best config-

Table 12: Optimal hyperparameters of DualEqui on each dataset.

Dataset	Covid	Ribonanza	Tcribo	RNASolo SASA	mRFP Protein SASA	TorsionAngle
L_{atom}	4	1	3	4	1	3
L_{nt}	3	4	3	1	1	4
# heads	2	8	4	6	3	5
hidden	72	128	84	150	174	140
atom d_{EU}	4.82	3.93	3.03	10.44	3.68	8.93
residue d_{EU}	26.4	30.61	122.64	47.02	116.39	42.45
lr	0.0030	0.0015	0.0002	0.0002	0.0005	0.0013
weight decay	1.37e-6	5.65e-8	1.17e-7	1.37e-5	3.00e-7	2.10e-7
l_{max}	2	2	2	2	3	3
atom d_{SH}	0.97	1.0	0.95	1.00	0.96	0.99
residue d_{SH}	0.97	0.95	0.92	0.95	0.95	0.91
batch size	32	25	32	16	16	16
epochs	1000	1000	1000	1000	1000	1000

urations are detailed in Appendix D.5. To ensure consistency, we fix random seeds for all runs (PyTorch, NumPy, CUDA) and repeat each experiment across 5 different seeds. All experiments were conducted on a single NVIDIA A6000 GPU with a number of parameters limit up to 4×10^6 . In Table 12, we provide our searched optimal hyperparameters of DualEqui on each dataset. More detailed training scripts will be released with the code.

NeurIPS Paper Checklist

1. Claims

Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

Answer: [\[Yes\]](#)

Justification: The claims in the abstract and introduction clearly reflect the method's dual-space design, hierarchical modeling, and strong benchmark performance.

Guidelines:

- The answer NA means that the abstract and introduction do not include the claims made in the paper.
- The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
- The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
- It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. Limitations

Question: Does the paper discuss the limitations of the work performed by the authors?

Answer: [\[Yes\]](#)

Justification: The paper discusses model limitations in Conclusion.

Guidelines:

- The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
- The authors are encouraged to create a separate "Limitations" section in their paper.
- The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
- The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
- The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
- The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
- If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
- While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. Theory assumptions and proofs

Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [\[Yes\]](#)

Justification: The paper includes a formal proof of equivariance in Appendix to support theoretical claims.

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. Experimental result reproducibility

Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

Answer: [Yes]

Justification: The experimental details are revealed in Appendix, including the optimal hyperparameter

Guidelines:

- The answer NA means that the paper does not include experiments.
- If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general, releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
 - (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
 - (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
 - (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
 - (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. Open access to data and code

Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [\[Yes\]](#)

Justification: All code will be released upon acceptance.

Guidelines:

- The answer NA means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (<https://nips.cc/public/guides/CodeSubmissionPolicy>) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so “No” is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (<https://nips.cc/public/guides/CodeSubmissionPolicy>) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. Experimental setting/details

Question: Does the paper specify all the training and test details (e.g., data splits, hyper-parameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

Answer: [\[Yes\]](#)

Justification: All of them are mentioned in experimental details.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
- The full details can be provided either with the code, in appendix, or as supplemental material.

7. Experiment statistical significance

Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

Answer: [\[Yes\]](#)

Justification: All experiment results come with standard deviation.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)

- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. Experiments compute resources

Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

Answer: [Yes]

Justification: All of them are mentioned in experimental details.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
- The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
- The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. Code of ethics

Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics <https://neurips.cc/public/EthicsGuidelines>?

Answer: [Yes]

Justification: The paper is with the NeurIPS Code of Ethics

Guidelines:

- The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
- If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
- The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. Broader impacts

Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

Answer: [NA]

Justification: The work focuses purely on methodological and scientific advancements in molecular modeling, without direct societal or ethical implications requiring discussion.

Guidelines:

- The answer NA means that there is no societal impact of the work performed.
- If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.

- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. Safeguards

Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

Answer: [NA]

Justification: The proposed model and datasets do not pose high risk for misuse.

Guidelines:

- The answer NA means that the paper poses no such risks.
- Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. Licenses for existing assets

Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

Answer: [Yes]

Justification: All of them are cited and credited.

Guidelines:

- The answer NA means that the paper does not use existing assets.
- The authors should cite the original paper that produced the code package or dataset.
- The authors should state which version of the asset is used and, if possible, include a URL.
- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, paperswithcode.com/datasets has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.

- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

Answer: [\[Yes\]](#)

Justification: The paper introduces two new benchmark datasets, with detailed descriptions provided in Section 4 and Appendix D.4. The new datasets will be released together with the code.

Guidelines:

- The answer NA means that the paper does not release new assets.
- Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
- The paper should discuss whether and how consent was obtained from people whose asset is used.
- At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

Answer: [\[NA\]](#)

Justification: There is no human subjects.

Guidelines:

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
- Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
- According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

Answer: [\[NA\]](#)

Justification: The study does not involve human participants and thus does not require IRB approval or risk disclosure.

Guidelines:

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.

- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does not impact the core methodology, scientific rigorousness, or originality of the research, declaration is not required.

Answer: [NA]

Justification: The LLM is used only for writing, editing, or formatting purposes and does not impact the core methodology, scientific rigorousness, or originality of the research, declaration is not required.

Guidelines:

- The answer NA means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
- Please refer to our LLM policy (<https://neurips.cc/Conferences/2025/LLM>) for what should or should not be described.