Equivariant Spherical Transformer for Efficient Molecular Modeling

Junyi An[*]
SAIS
Golab
anjunyi@sais.org.cn

Xinyu Lu[*†]
Shanghai Innovation Institute
Xiamen University
xinyulu@stu.xmu.edu.cn

Yunfei Shi
SAIS
Golab
shiyunfei@sais.org.cn

Peijia Lin[†]
Sun Yat-sen University
Shanghai Innovation Institute
linpj6@mail2.sysu.edu.cn

Licheng Xu
SAIS
Golab
xulicheng@sais.org.cn

Qianwei Tang
Nanjing University
cleveland225@163.com

Chao Qu [‡]
Fudan University
SAIS
quchao@fudan.edu.cn

Fenglei Cao [‡]
SAIS, Golab
Fudan Institute of Future Technology
caofenglei@sais.org.cn

Yuan Qi[‡]
Fudan University
SAIS
qiyuan@fudan.edu.cn

Abstract

Equivariant Graph Neural Networks (GNNs) have significantly advanced the modeling of 3D molecular structure by leveraging group representations. However, message passing built only from exactly equivariant operations can have restricted practical expressiveness at a fixed representation degree. We introduce the Equivariant Spherical Transformer (EST), a plug-and-play module that applies Transformer attention to spherical spatial representations obtained from group representations. EST explicitly mixes information across sampled orientations, unlike prior orientation-wise spherical operations, while retaining a strong approximate equivariant inductive bias through near-uniform spherical sampling. As demonstrated by our experiments on challenging benchmarks like OC20, MPtrj and QM9, EST-based models achieve state-of-the-art performance. For the complex molecular systems within OC20, small models empowered by EST can outperform some larger models and those using additional data. We characterize the ideal closed-domain condition under which the continuous operation is exactly equivariant and empirically quantify the finite-sampling accuracy–equivariance–memory trade-off used in practice.
Code: https://github.com/ajy112/EST

1 Introduction

Graph Neural Networks (GNNs) have become indispensable for modeling molecular systems and approximating quantum mechanical calculations, providing critical support for high-impact tasks such as drug discovery [1, 2, 3] and material design [4]. Compared to traditional methods such as Density Functional Theory (DFT), GNNs can predict quantum properties in fractions of a second,

[*]equal contribution. SAIS: Shanghai Academy of AI for Science
[†]This work is done when they are interns at SAIS
[‡]corresponding authors

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

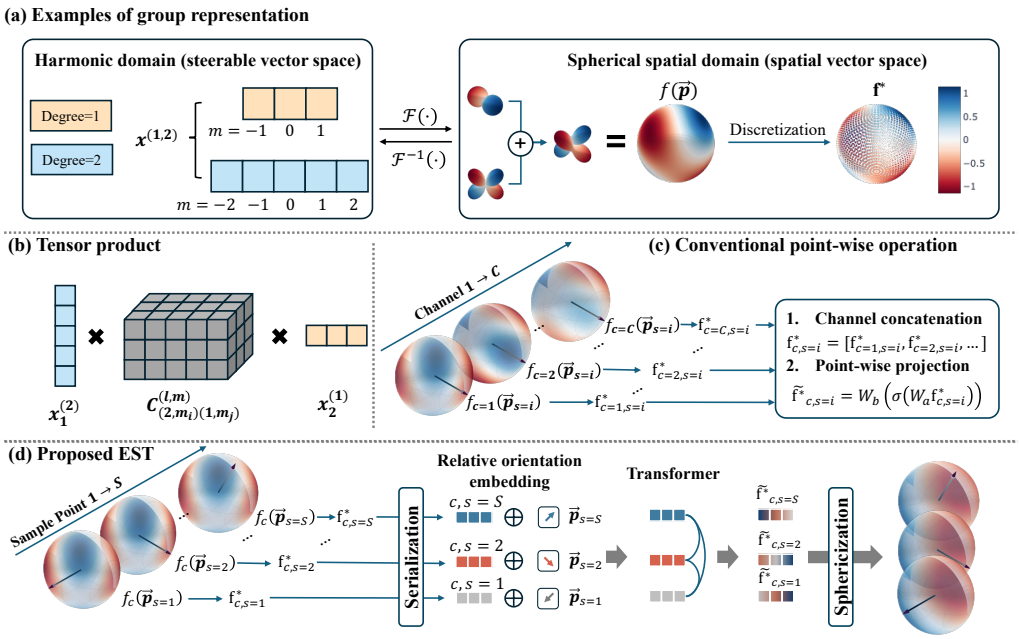


Figure 1: **Overview of operations in the spherical spatial domain.** (a) **Examples of group representation.** A Fourier transform is applied to project steerable vectors onto the spatial domain, which are then stored through sampling (i.e., discretization). (b) **Tensor product operation.** Steerable vectors are combined using the Clebsch-Gordan coefficient $\mathcal{C}$. (c) **Conventional orientation-wise (point-wise) operation.** Features are processed independently at each sampling orientation s. A shared neural network (with linear projection W and activation $\sigma(\cdot)$) acts only across channels ($c = 1, \ldots, C$) within the same orientation $\vec{\mathbf{p}}_{s=i}$, neglecting cross-orientational relationships. (d) **Proposed EST operation.** A Transformer-based architecture that explicitly models dependencies across all pairs of orientations ($s = 1, \ldots, S$), enabling the capture of complex, cross-orientational geometric correlations.

significantly reducing computational cost from hours or days. Among the various GNN architectures, SE(3)-equivariant GNNs [5, 6] are a leading family whose symmetry-aware inductive bias helps capture intricate atomic interactions.

In molecular systems, SE(3)-invariance and SE(3)-equivariance are fundamental constraints. For instance, molecular energies remain constant under rotation, while atomic forces rotate in concert with the molecule equivariantly. Adherent to these priors, early invariant models such as SchNet [7] and HIP-NN [8] utilize scalar interatomic distances in their message-passing blocks, consequently limiting their ability to capture interactions involving triplets or quadruplets of atoms [9]. Directional GNNs [10, 11] address this problem by explicitly incorporating bond angles and dihedral angles. While these models showed performance improvements, their expressiveness remains constrained due to a heavy reliance on handcrafted features. To move beyond these manual descriptors and capture richer geometric information, SE(3)-equivariant GNNs employ group representations as node embeddings and construct steerable message-passing blocks [5, 6, 12, 13, 14], where the tensor product is applied to capture equivariant interactions between group embeddings (see Figure 1(b)). Nevertheless, tensor-product operations provide limited nonlinear mixing at a fixed maximum degree [5, 15, 16, 17]. Cartesian-tensor models organize higher-degree interactions through symmetric Cartesian tensors and irreducible decomposition [18]; they retain exact equivariance but their cost grows with correlation order and angular degree. A complementary line, including SCN and eSCN [19, 20], applies an FT to obtain sampled spherical channels and processes each orientation independently. EquiformerV2 adopts related S2Activation and SO(2)-convolution components [21]. EST shares this spherical representation but changes the central operation: self-attention mixes all sampled orientations and can model dependencies that an orientation-wise MLP cannot (Figure 1(c–d)).

Finite sampling makes EST approximately, rather than exactly, equivariant for arbitrary rotations. This is a deliberate and measured trade-off. Prior work has likewise studied soft, partial, or approximate equivariance when exact constraints limit expressiveness or the symmetry is imperfect

[22, 23, 24, 25]. Our goal is to combine an expressive spatial branch with exact-equivariant operators while disclosing the resulting error and memory cost.

We present EST as a plug-and-play, approximately equivariant module for high-fidelity modeling of atomic interactions. EST converts Fourier spatial representations into orientation sequences and applies a Transformer to capture dependencies across orientations (Figure 1(d)). We use a Fibonacci lattice (FL), optionally refined by repulsion, to reduce finite-sampling error. A controlled replacement of S2Activation in the original SCN isolates the contribution of cross-orientation attention. Across OC20, MPtrj, and QM9, EST improves predictive performance most clearly on complex systems with remaining modeling headroom.

Our contributions are: (i) EST introduces cross-orientation attention into spherical-channel models; (ii) we state the exact closed-domain equivariance condition and measure the approximation induced by finite FL sampling; and (iii) controlled ablations and benchmark experiments characterize the resulting accuracy, expressiveness, equivariance, latency, and memory trade-offs.

2 Preliminaries and Related Work

In this section, we discuss related works and the corresponding mathematical background relevant to equivariant GNNs.

Notations We begin by listing the notation commonly used throughout the paper. **We denote the unit sphere as $\mathbb{S}^2$, where the coordinate of a spherical orientation (i.e. spherical sampling point) $\vec{\mathbf{p}} = (\theta, \varphi)$ is represented by its polar angle θ and azimuth angle φ.** The symbol $\mathbb{R}$ represents the set of real numbers, while $\mathbf{R}$ denotes a rotation matrix for 3D vectors. We use $[\cdot, \cdot]$ or $\oplus$ to indicate the concatenation of the tensors and $\tilde{\cdot}$ to represent the update of the tensors, respectively.

Equivariance and Invariance Given a group $\mathbf{G}$ and a transformation parameter $g \in \mathbf{G}$, a function $\phi : \mathcal{X} \to \mathcal{Y}$ is said to be equivariant to g if it satisfies:

$$\phi(T(g)[x]) = T'(g)[\phi(x)], \tag{1}$$

where $T'(g) : \mathcal{Y} \to \mathcal{Y}$ and $T(g) : \mathcal{X} \to \mathcal{X}$ denote the corresponding transformations on $\mathcal{Y}$ and $\mathcal{X}$, respectively. Invariance is a special case of equivariance where $T'(g)$ is the identity transformation.

Spherical Harmonics and Steerable Vectors Spherical harmonics, a set of orthonormal basis functions defined over the sphere $\mathbb{S}^2$, are commonly employed in equivariant models. The real-valued spherical harmonics are typically denoted as $\{Y^{(l,m)} : \mathbb{S}^2 \to \mathbb{R}\}$, where l represents the degree and m represents the order. For any sample point $\vec{\mathbf{p}}$, we define $\mathbf{Y}^{(l)}(\vec{\mathbf{p}}) = [Y^{(l,-l)}(\vec{\mathbf{p}}), Y^{(l,-l+1)}(\vec{\mathbf{p}}), ..., Y^{(l,l)}(\vec{\mathbf{p}})]$, a vector of size $2l + 1$, and $\mathbf{Y}^{(0\to l)}(\vec{\mathbf{p}}) = [\mathbf{Y}^{(0)}(\vec{\mathbf{p}}), ..., \mathbf{Y}^{(l)}(\vec{\mathbf{p}})]$, a vector of size $(l + 1)^2$.

A key property of spherical harmonics is their behavior under rotation $\mathbf{R} \in SO(3)$:

$$\mathbf{Y}^{(l)}(\mathbf{R}\vec{\mathbf{r}}) = \mathbf{D}^{(l)}(\mathbf{R})\mathbf{Y}^{(l)}(\vec{\mathbf{r}}), \tag{2}$$

where $\mathbf{D}^{(l)}(\mathbf{R})$ is a $(2l + 1) \times (2l + 1)$ matrix known as the Wigner-D matrix of degree l. Notably, $\mathbf{D}^{(1)}(\mathbf{R}) = \mathbf{R}$. Thus, $\mathbf{R}$ and $\mathbf{D}^{(l)}(\mathbf{R})$ can represent $T(g)$ and $T'(g)$ in equation (1). Following the convention in [26, 6], we say that $\mathbf{Y}^{(l)}(\vec{\mathbf{p}})$ is steerable by the Wigner-D matrix of the same degree l. Furthermore, a vector that transforms according to an l-degree Wigner-D matrix is termed an l-degree steerable vector or a type-l vector, residing in the vector space $\mathbb{V}_l$. Further mathematical details are available in Appendix A.2.

Equivariant Architectures Most equivariant models are based on the message-passing neural network (MPNN), the details of which are provided in the Appendix A.1. A common practice in equivariant models is to use steerable vectors as node embedding and encode relative positions $\vec{\mathbf{r}}_{ij}$ using spherical harmonics as edge attributes. TFN [5] and NequIP [27] provide a general framework that learns the interaction between node embeddings and edge attributes through equivariant convolution filters. These filters are composed of equivariant operations such as degree-wise linear layers, Clebsch-Gordan (CG) tensor products, and Gate mechanisms. SEGNN [6] extends this convolution

framework by more non-linear operations to achieve enhanced learning ability. SE(3)-Transformer [28] and Equiformer [12] introduce an attention mechanism that assigns rotation-invariant weights to neighbor attributes, thereby improving the learning of key atomic interactions. Several studies [20, 29, 14] have sought to develop variants of the tensor product operation to reduce computational complexity without significantly compromising expressiveness. Nevertheless, the core operation in these works remains the convolution filter based on the CG tensor product. In contrast, GotenNet [13] proposes an effective tensor-product-free structure that combines neighbor attention with hierarchical refinement mechanisms, achieving SOTA performance in certain molecular benchmarks. Our work tends to explore a distinct tensor-product-free paradigm: we project steerable vectors into the spherical spatial domain, enabling a novel attention mechanism to weight embedding compositions directly. This approach fundamentally diverges from previous attention-based GNNs by shifting the focus from local neighbor-indexing to spatial-domain interactions, providing a more direct way to capture geometric dependencies.

Spherical Fourier Transform Any square-integrable function $f(\vec{\mathbf{p}})$ defined over the sphere $\mathbb{S}^2$ can be expressed in a spherical harmonic basis via the spherical Fourier transform $\mathcal{F}$:

$$f(\vec{\mathbf{p}}) = \mathcal{F}(\mathbf{x}) = \sum_{l=0}^{\infty} \sum_{m=-l}^{l} x^{(l,m)} Y^{(l,m)}(\vec{\mathbf{p}}), \tag{3}$$

where $x^{(l,m)}$ is the spherical Fourier coefficient. Conversely, the coefficient can be obtained from the function $f(\vec{\mathbf{p}})$ via the inverse transform $\mathcal{F}^{-1}$:

$$x^{(l,m)} = \mathcal{F}^{-1}(f(\vec{\mathbf{p}})) = \int_{\Omega} f(\vec{\mathbf{p}}) Y^{(l,m)*}(\vec{\mathbf{p}}) d\Omega, \tag{4}$$

where $d\Omega$ denotes the unit sphere and $Y^{(l,m)*}(\cdot)$ is the conjugate of the spherical harmonic function. The function $f(\vec{\mathbf{p}})$ and its coefficients $\mathbf{x}$ are referred to as the representations in the spherical spatial domain and the harmonic domain, respectively. These processes are illustrated in Figure 1(a). SCN [19] introduced an approach that applies a simple point-wise neural network in the spherical spatial domain. This technique has been subsequently adopted by several works [20, 21, 30] within their message-passing frameworks. However, point-wise operations inherently limit expressiveness as they process each point in isolation, failing to account for the relationships and structural correlations between different points in the spatial domain. In this work, we empower models to learn across the entire spherical spatial domain, thereby capturing complex cross-orientational dependencies.

3 Model

In this section, we first make a review of the tensor product to detail our motivation. Then, we introduce the formulation of our EST and clarify its properties.

3.1 Review of traditional equivariant operations

A common wisdom in equivariant models is to incorporate steerable vectors with multiple degrees for node or edge features and then employ CG tensor products $\otimes : \mathbb{V}_{l_1} \times \mathbb{V}_{l_2} \to \mathbb{V}_l$, which are defined as:

$$(\mathbf{x}_1 \otimes \mathbf{x}_2)^{(l,m)} = \sum_{m_1=-l_1}^{l_1} \sum_{m_2=-l_2}^{l_2} \mathcal{C}^{(l,m)}_{(l_1,m_1)(l_2,m_2)} \mathbf{x}_1^{(l_1,m_1)} \mathbf{x}_2^{(l_2,m_2)}, \tag{5}$$

where $\mathcal{C}^{(l,m)}_{(l_1,m_1)(l_2,m_2)}$ are the CG coefficients, a sparse tensor yielding non-zero terms when $|l_1 - l_2| \leq l \leq (l_1 + l_2)$. These products enable interactions between various combinations of l_1 and l_2, crucial for the universal expressiveness of MPNNs [15, 16]. However, CG tensor products provide limited non-linearity [6], often requiring the stacking of multiple layers to capture complex molecular features. Furthermore, their high complexity makes it difficult to apply to learning large-scale cross-channel interactions [12], especially in message block. While techniques like Gate mechanisms [6, 12] and geometry-aware tensor attention [13] can be used to mitigate this limitation, they apply non-linear transformations to the invariant ($l = 0$) representations. Higher-degree representations ($l > 0$) are then updated through a unified scaling multiplication:

$$C_{inv} = \mathrm{NL}(\mathbf{x}^{(0)}, \sum_{l=1}^{L} \mathrm{d}(\mathbf{x}^{(l)})), \quad \text{and} \quad \tilde{\mathbf{x}}^{(l)} = C_{inv} \cdot \mathbf{x}^{(l)}, \tag{6}$$

where C_{inv} is an invariant scalar, $\mathrm{NL}(\cdot)$ is a non-linear network, and $\mathrm{d}(\cdot) : \mathbb{V}_l \to \mathbb{V}_0$ maps equivariant features to invariants, such as an inner product. This indirect exchange of information between higher-degree ($l > 0$) representations still limits the overall expressiveness.

3.2 Equivariant Spherical Transformer

3.2.1 Node embedding

Steerable Representation For representation with steerable vectors, we first define a maximal degree L. Each node n's embedding $\mathbf{x}_n$ comprises C channels, and each channel c is a concatenation of steerable vectors from degree 0 to L: $\mathbf{x}_{n,c}^{(0\to L)} = [x_{n,c}^{(0)}, \mathbf{x}_{n,c}^{(1)}, ..., \mathbf{x}_{n,c}^{(L)}]$. As a result, the dimension of a steerable node embedding is $(L+1)^2 \times C$.

Spatial Representation Steerable representations $\mathbf{x}_{n,c}$ can be transformed into a spherical function $f_{n,c}(\vec{\mathbf{p}})$ via equation (3), with the summation truncated at a maximum degree L. Furthermore, $f_{n,c}(\vec{\mathbf{p}})$ is discretely represented by sampling S points on the sphere, yielding a spatial node embedding $\mathbf{f}_n^* \in \mathbb{R}^{S\times C}$. It can be expressed as the concatenation over sampled points:

$$\mathbf{f}_{n,c}^* = \Big[(\mathbf{x}_{n,c}^{(0\to L)})^T \mathbf{Y}^{(0\to L)}(\vec{\mathbf{p}}_1),\ (\mathbf{x}_{n,c}^{(0\to L)})^T \mathbf{Y}^{(0\to L)}(\vec{\mathbf{p}}_2), \ldots,\ (\mathbf{x}_{n,c}^{(0\to L)})^T \mathbf{Y}^{(0\to L)}(\vec{\mathbf{p}}_S)\Big]. \quad (7)$$

We use the term $\mathbf{f}_{n,c,s}^*$ to denote the signal at the sampled point $\vec{\mathbf{p}}_s$ of channel c of node n. The inverse transform in equation (4) can convert $\mathbf{f}_n^*$ back to $\mathbf{x}_n$.

The Significance of Operating in the Spatial Domain In the spatial representation, the explicit degree and order are no longer directly accessible. Instead, higher-degree information is encoded as geometric features distributed across the spherical surface, as illustrated in Figure 1(a). Intuitively, **by modeling the interdependencies of these geometric features, the model can effectively exchange information between implicit degrees, thereby approximating the capability of the tensor product.** SCN/eSCN-style S2Activation processes channels independently at each orientation [19, 20]; EST instead models dependencies among all orientations. We analyze two distinct properties: exact equivariance for an ideal rotationally closed domain and approximate equivariance after finite sampling, together with evidence that cross-orientation attention increases practical expressiveness.

3.2.2 Spherical Attention

To prepare the data, we flatten all sampled points from $\mathbf{f}^*$ into a sequential format, where each point s represented by a C-dimensional feature vector. We then apply an attention mechanism [31]:

$$a_{s=i,s=j} = \frac{\exp(\mathbf{Q}_{s=i}\mathbf{K}_{s=j}^T/\sqrt{C})}{\sum_{k=1}^{S}\exp(\mathbf{Q}_{s=i}\mathbf{K}_{s=k}^T/\sqrt{C})}, \text{and} \quad \tilde{\mathbf{f}}_{s=i}^* = \sum_{j=1}^{S} a_{s=i,s=j}\mathbf{V}_{s=j}. \quad (8)$$

where the $\mathbf{Q}$, $\mathbf{K}$, and $\mathbf{V}$ matrices are obtained through point-wise linear transformations of $\mathbf{f}_{s=i}^*$. The spherical attention is the core of EST. We also incorporate the pre-LayerNorm [32] and feedforward networks (FFN) in EST. As a fundamental building block, EST focuses on interactions within an equivariant representation while preserving input-output dimensionality.

Equivariance and sampling strategies We first state the exact result on the closed domain required by the proof, and then distinguish it from the finite sampling used in implementations.

Theorem 3.1. *Let $\Omega \subseteq \mathbb{S}^2$ be the domain of the spherical transform, equipped with the rotation-invariant surface measure, and assume that it is closed under every $\mathbf{R} \in SO(3)$, i.e., $\mathbf{R}\Omega = \Omega$. If spherical attention is evaluated over Ω before the inverse transform, then the continuous EST operator is $SO(3)$-equivariant:*

$$\mathcal{F}^{-1}\big(\mathrm{EST}\big(\mathcal{F}(\mathbf{D}\mathbf{x})\big)\big) = \mathbf{D}\mathcal{F}^{-1}\big(\mathrm{EST}\big(\mathcal{F}(\mathbf{x})\big)\big), \quad (9)$$

where $\mathbf{D}$ is the Wigner-D representation associated with $\mathbf{R}$.

The proof is provided in Appendix B.1. The theorem applies to the closed continuous domain Ω (in our setting, the full sphere), not to an arbitrary finite point set. A finite set $\mathcal{P} = \{\vec{\mathbf{p}}_s\}_{s=1}^S$ is

closed only under its grid-symmetry subgroup, so arbitrary rotations introduce discretization error. We therefore seek near-uniform samples that make the quadrature and nearest rotated-point mismatch small. Compared with the latitude-dependent e3nn grid [33], we use Fibonacci Lattice (FL) sampling [34]:

$$\vec{\mathbf{p}}_s = [\sqrt{1-z_s^2}\cos(2s\pi/\lambda), \sqrt{1-z_s^2}\sin(2s\pi/\lambda), z_s], \tag{10}$$

where $z_s = \frac{2s-1}{S} - 1$ and $\lambda = \frac{1+\sqrt{5}}{2}$ denotes the golden ratio. Equation (10) ensures that the sampled points are distributed across the spherical surface with approximately uniform density (see Figure 3(b)). While FL sampling provides high uniformity, we further refine this distribution by simulating point motion on the sphere via consistent repulsions. This process resembles a simple molecular dynamics simulation where each "atom-like" point is confined to the spherical surface, with repulsive forces determined by inter-point distances. Following FL sampling, we define the self-conjugate basis for the FT and its inverse. Further details regarding the sampling methods and the basis are provided in Appendix B.1. Furthermore, our ablation experiments (Table 6) demonstrate that sampling methods with poor uniformity can severely compromise equivariance.

Expressiveness As demonstrated by [15, 16, 17], the expressiveness of CG tensor product-based modules is constrained by the maximum degree L, a finding corroborated by ablation experiments. In contrast, our proposed EST module can project steerable representations of any degree onto the sphere and perform flexible, learnable operations. This allows the resulting features to be reprojected back to any desired degree, thereby potentially overcoming the expressiveness limitations.

Claim 3.2 (Unproven expressiveness intuition)**.** Under information-preserving spherical transforms, cross-orientation Transformer mixing may approximate the local functions represented by CG tensor products and also represent functions outside the corresponding fixed-degree local class.

Claim 3.2 is not a proved set-inclusion theorem. It is intuition supported by the universal-approximation argument in Appendix B.2, the controlled fitting experiment in Appendix B.2, and the symmetry task in Table 5. The transform itself is information-preserving once the sampling meets the relevant band-limit/Nyquist requirement (we use $S \geq (2L)^2$) [19].

Relative Orientation Embedding In basic form, spherical attention determines the dependencies between points based solely on their features, ignoring their relative positions in the input sequence. This means the output feature of a given point remains unchanged even if the order of all other points is altered (Figure 4). Such behavior is undesirable, as different point orderings correspond to distinct spherical functions. To enable the model to distinguish between these configurations, we introduce a relative orientation embedding by augmenting the query ($\mathbf{Q}$) and key ($\mathbf{K}$) vectors with point information:

$$\mathbf{Q}_{s=i} := [\mathbf{Q}_{s=i}, \vec{\mathbf{p}}_{s=i}], \quad \mathbf{K}_{s=j} := [\mathbf{K}_{s=j}, \vec{\mathbf{p}}_{s=j}], \quad \forall i, j \in \{1, \ldots, S\}. \tag{11}$$

Consequently, the inner product between $\mathbf{Q}_{s=i}$ and $\mathbf{K}_{s=j}$ now incorporates a term related to the orientation-aware inner product $\vec{\mathbf{p}}_{s_i}^T \vec{\mathbf{p}}_{s_j}$. Crucially, this augmentation does not compromise the equivariant inductive bias of EST, as proven in Appendix B.3. Our approach shares conceptual similarities with Rotary Position Embeddings (RoPE) [35] in Natural Language Processing (NLP), where the relative position embedding is invariant under transformation. However, our method fundamentally differs from RoPE by addressing rotation invariance instead of permutation invariance.

3.2.3 Integrating EST into Equivariant Architectures

The modular design of EST enables seamless integration into existing equivariant neural networks, circumventing complex hyperparameter tuning while leveraging the inherent strengths of diverse architectures. **Given ESTs high expressivity and the proficiency of Transformers in multi-channel processing, we incorporate this framework into the update blocks of various models to handle complex message aggregation.** Furthermore, because most standard update blocks are architecturally minimalist, EST can be easily deployed as a versatile, plug-and-play enhancement.

In our implementation, the base update and an EST branch run in parallel and are combined by learned invariant weights. We call them steerable and spatial experts only as shorthand; this is not a sparse or routed language-model MoE. The mixed branch compensates for finite-sampling error while adding cross-orientation capacity. Appendix D.4 shows that a minimal $1 + 1$ mixture is more effective than merely adding same-type branches.

Table 1: S2EF validation results. λ_E is the coefficient of the energy loss. "$^\tau$" denotes the test results.

Dataset	Model	Number of parameters	Throughput Samples/s	Energy MAE (meV) ↓	Force MAE (meV/Å) ↓
All+MD	EScAIP-Small	83M	-	**229**	15.1
	EquiformerV2 ($\lambda_E = 4$, 8 layers)	31M	7.1	232	16.3
All	SchNet	9.1M	-	549	56.8
	DimeNet++-L-F+E	10.7M	4.6	515	32.8
	SpinConv	8.5M	6.0	371	41.2
	GemNet-dT	32M	25.8	315	27.2
	GemNet-OC	39M	18.3	244	21.7
	SCN$^\tau$	271M	-	244	17.7
	eSCN$^\tau$	200M	2.9	242	17.1
	EquiformerV2 ($\lambda_E = 2$, 20 layers)	153M	1.8	236	15.7
	EquiformerV2 + EST ($\lambda_E = 4$, 8 layers)	45M	6.8	**229**	15.8
	EquiformerV2 + EST ($\lambda_E = 2$, 8 layers)	45M	6.8	232	**15.0**

Table 2: IS2RE results of models trained on IS2RE training dataset. Bold and underline indicate the best result, and the second best result, respectively.

Model	Energy MAE (meV) ↓				EwT (%) ↑			
	ID	OOD Ads	OOD Cat	OOD Both	ID	OOD Ads	OOD Cat	OOD Both
SchNet	639	734	662	704	2.96	2.33	2.94	2.21
PaiNN	575	783	604	743	3.46	1.97	3.46	2.28
TFN	584	766	636	700	4.32	2.51	4.55	2.66
DimeNet++	562	725	576	661	4.25	2.07	4.10	2.41
GemNet-dT	527	758	549	702	4.59	2.09	4.47	2.28
GemNet-OC	560	711	576	671	4.15	2.29	3.85	2.28
SphereNet	563	703	571	638	4.47	2.29	4.09	2.41
SEGNN	533	692	537	679	**5.37**	2.46	<u>4.91</u>	2.63
SCN	516	**643**	530	<u>604</u>	4.92	**2.71**	4.42	**2.76**
Equiformer	<u>504</u>	688	<u>521</u>	630	5.14	2.41	4.67	<u>2.69</u>
+ EST	**501**	<u>652</u>	**502**	**578**	<u>5.16</u>	<u>2.67</u>	**5.16**	**2.76**

4 Experiments

In this section, we construct experiments to investigate the efficacy of the proposed EST. We evaluate its performance on the OC20 (S2EF and IS2RE), MPtrj and QM9 benchmarks. Our analysis involves a comparison with a wide range of baseline models, including SchNet [7], PaiNN [36], SEGNN [6], TFN [5], DimeNet++ [37], Equiformer [12], EquiformerV2 [21] and EScAIP [38]. Specifically for the OC20 benchmark, we extend comparison to include SpinConv [39], GemNet [11, 40], SphereNet [41], SCN [19] and eSCN [20]. For the MPtrj benchmark, our comparison includes CHGNet [3] and MACE [42]. Similarly, our QM9 experiments are augmented with comparisons to TorchMD-NET [43], EQGAT [44] and GotenNet [13]. The baseline configurations can be found in Appendix D.1.

4.1 OC20 Results

Dataset and Configurations The OC20 dataset, comprising over 130 million molecular structures, covers a broad spectrum of materials and adsorbates. We evaluate models on two core subdatasets: S2EF and IS2RE. All configurations are detailed in Appendix D.2. Notably, We did not use very deep models. For instance, our S2EF experiments utilize a 8-layer architecture, and IS2RE experiments employ a 6-layer one, significantly fewer than many advanced methods.

S2EF Results Using EquiformerV2 as the base model, we implemented an EST-based variant and conducted training on the S2EF-All dataset. Each validation set was divided into four similarly-sized subsets: In-Domain (ID), Out-of-Domain Catalysis (OOD Cat), Out-of-Domain Both (OOD Both). Consistent with prior work [20, 21], the best model was selected based on its performance on the ID validation subset during training and subsequently evaluated across all validation subsets. We compared our results with two variants of the EquiformerV2 base model: a deep 20-layer architecture and one trained with additional MD dataset. As shown in Table 1, EST outperformed both EquiformerV2 variants in energy and forces prediction. Furthermore, EST achieved competitive results in force prediction, surpassing several bigger baselines including EScAIP, SCN and eSCN.

Table 3: Results on Mptrj dataset.

Model	Energy (meV/atom) ↓	Forces (meV/Å) ↓	Stress (meV/Å^3) ↓
CHGNet	33	79	2.19
MACE	20	45	-
EquiformerV2-S	12.4	32.22	1.55
+ DeNS	11.43	31.67	1.44
+ EST	**10.80**	**28.85**	**1.36**

Table 4: Results on QM9 dataset for various properties. † denotes using different data partitions. Gray indicates the optimal value in the comparison between EST and the base model.

Task Units	α $bohr^3$	$\Delta\varepsilon$ meV	ε_{HOMO} meV	ε_{LUMO} meV	μ D	C_v cal/(mol K)	G meV	H meV	R^2 $bohr^3$	U meV	U_0 meV	$ZPVE$ meV
SchNet	.235	63	41	34	.033	.033	14	14	.073	19	14	1.70
TFN†	.223	58	40	38	.064	.101	-	-	-	-	-	-
DimeNet++	.044	33	25	20	.030	.023	8	7	.331	6	6	1.21
PaiNN	.045	46	28	20	.012	.024	7.35	5.98	.066	5.83	5.85	1.28
TorchMD-NET	.059	36	20	18	.011	.026	7.62	6.16	.033	6.38	6.15	1.84
SEGNN†	.060	42	24	21	.023	.031	15	16	.660	13	15	1.62
EQGAT	.053	32	20	16	.011	.024	23	24	.382	25	25	2.00
EquiformerV2	.050	29	14	13	.010	.023	7.57	6.22	.186	6.49	6.17	1.47
Equiformer	.046	30	15	14	.011	.023	7.63	6.63	.251	6.74	6.59	1.26
+ EST	.041	29	14	13	.011	.021	7.18	6.17	.227	6.35	6.32	1.27
GotenNet-S	.033	**21.2**	16.9	13.9	.0075	.020	5.50	**3.70**	**.029**	**3.67**	3.71	**1.09**
+ EST	**.030**	31.2	**16.7**	**13.3**	**.0070**	**.019**	**5.40**	3.71	**.029**	3.70	**3.70**	1.19

IS2RE Results We trained our model directly on the IS2RE training set for energy prediction. Related results, including the Mean Absolute Error (MAE) for energy and the percentage of Energies Within a Threshold (EwT) for each model, are summarized in Table 2. We use a 6-layer Equiformer as base model. EST achieves SOTA performance on ID, OOD Cat, OOD Both and the second best on OOD Ads. It is notable that the current SOTA model on OOD Ads, SCN, employs complex 16-layer architecture with over 100M parameters while EST achieves closely competitive performance using a unified 6-layer structure comprising just 32.47 million parameters. Furthermore, we observe that EST consistently surpasses the base model (Equiformer) across all evaluated metrics, providing further evidence for the effectiveness of the proposed EST module.

4.2 MPtrj Results

Dataset and Configurations The MPtrj benchmark consists of a large-scale collection of material molecular dynamics trajectories for predicting energy, force, and stress. We integrate EST into a lightweight 8-layer EquiformerV2. Note that we compare our approach against the EquiformerV2 and DeNS [45] baselines as reported in [46], the latter being an efficient self-supervised enhancement technique. Other configurations are detailed in Appendix D.2.

Results Note that molecules in MPtrj show complex interaction where metallic systems typically exhibit complex electronic orbital structures and interfacial properties. Remarkablely, From Table 3, we observe that EST significantly outperform other baselines on all the metrics, showing a stronger expressiveness. Furthermore, due to discrepancies and ambiguities in the hyperparameter configurations of prior work, we also reproduced the results of the base model in the Appendix D.4, where EST-based model still demonstrates significant improvement. Additional NVE dynamics, force-equivariance, geometry-relaxation, and vibrational tests in Appendix C.4 show that EST remains applicable to interatomic-potential workflows, while also revealing the non-conservativity limitation of a direct force head.

4.3 QM9 Results

Dataset and Configurations The QM9 benchmark comprises quantum chemical properties for a relevant, consistent, and comprehensive chemical space of 134k equilibrium small organic molecules containing up to 29 atoms. We developed two model architectures: the first model, denoted as Equiformer+EST, integrates Equiformer's message block with EST's update block. The second model, denoted as GotenNet+EST, integrates GotenNet's message block with EST's update block. For additional configuration specifics, please refer to Appendix D.2.

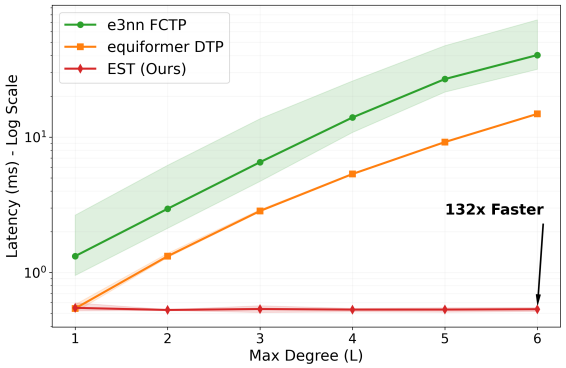


Figure 2: **Scaling analysis of inference latency.** EST, FCTP, and DTP are compared across maximum degree L and channel dimension C. Curves show the mean over $C \in \{32, 64, 128, 256, 512\}$ and shaded regions show the range. EST achieves a 132× speedup over FCTP at $L = 6, C = 512$.

Results QM9 is small and contains only equilibrium molecules with at most nine heavy atoms, so performance is comparatively saturated and symmetry is a strong regularizer. Equiformer+EST improves ten of twelve tasks, but the gains are attenuated for deeper GotenNet backbones. Our depth ablation is consistent with two contributing effects: less remaining accuracy headroom and accumulation of finite-sampling equivariance error with depth. We report this as a limitation and interpretation rather than a conclusively isolated causal mechanism.

4.4 Ablation study

In this section, we explore three key properties of EST: expressiveness, equivariance and latency. The other experiments can be found in Appendix D.4, where we supplement theoretical experiments on expressiveness and comprehensively investigate the influence of building components, including self-attention, spherical relative orientation embedding and hybrid experts.

Controlled replacement In the original SCN on OC20-S2EF-200K, replacing only its orientation-wise grid MLP with EST improves force MAE from 0.05172 to 0.04931, force cosine from 0.26900 to 0.29511, and force-magnitude error from 0.08147 to 0.07597, while energy MAE increases from 0.54533 to 0.56208. Appendix C.1 gives the matched settings and full result.

Expressiveness Following [16], we employ evaluation metrics that distinguish n-fold symmetric structures to precisely assess EST's expressiveness. We first construct a single-layer message-passing layer based on a EST-only structure, consistent with tensor product-based operations (TFN, MACE). Additionally, we remove the steerable FFN for a clear comparison. As demonstrated in [15, 16], the expressiveness of tensor product-based models is limited by the maximum degree L, failing to perfectly distinguish n-fold symmetric structures when $n > L$. From Table 5, we observe that EST significantly breaks through this theoretical boundary. It allows the model to distinguish symmetric structures even with very high fold (e.g., using 1-degree EST for 100-fold structures).

Table 5: Experiments on Rotationally symmetric structures.

GNN Layer	3 fold	10 fold	100 fold	GNN Layer	2 fold	3 fold	5 fold	10 fold
$\text{EST}_{L=1}$	**100.0** ś 0.0	**100.0** ś 0.0	**100.0** ś 0.0	$\text{TFN/MACE}_{L=1}$	50.0 ś 0.0	50.0 ś 0.0	50.0 ś 0.0	50.0 ś 0.0
$\text{EST}_{L=2}$	**100.0** ś 0.0	**100.0** ś 0.0	**100.0** ś 0.0	$\text{TFN/MACE}_{L=2}$	**100.0** ś 0.0	50.0 ś 0.0	50.0 ś 0.0	50.0 ś 0.0
$\text{EST}_{L=3}$	**100.0** ś 0.0	**100.0** ś 0.0	**100.0** ś 0.0	$\text{TFN/MACE}_{L=3}$	**100.0** ś 0.0	**100.0** ś 0.0	50.0 ś 0.0	50.0 ś 0.0
$\text{EST}_{L=5}$	**100.0** ś 0.0	**100.0** ś 0.0	**100.0** ś 0.0	$\text{TFN/MACE}_{L=5}$	**100.0** ś 0.0	**100.0** ś 0.0	**100.0** ś 0.0	50.0 ś 0.0
$\text{EST}_{L=10}$	**100.0** ś 0.0	**100.0** ś 0.0	**100.0** ś 0.0	$\text{TFN/MACE}_{L=10}$	**100.0** ś 0.0	**100.0** ś 0.0	**100.0** ś 0.0	**100.0** ś 0.0

Equivariance Finite EST is approximately equivariant. We measure the mean absolute output difference before and after random rotations on 1,000 QM9 molecules using untrained 1–6 layer EST-only networks. Table 6 shows that error decreases with more uniform or denser sampling; a trained 6-layer model gives $1.8 \times 10^{-5} \pm 0.7 \times 10^{-5}$ averaged across QM9 properties. Independent

Table 6: Equivariance evaluation with different sampling strategies.

Sampling Strategy/Number	1-layer	2-layer	3-layer	4-layer	5-layer	6-layer
FL / $S = 64$, w/o optim.	0.0010	0.0011	0.0010	0.0018	0.0022	0.0024
FL / $S = 64$, w optim.	0.0002	0.0001	0.0003	0.0003	0.0002	0.0002
FL / $S = 256$, w/o optim.	0.0002	0.0002	0.0001	0.0002	0.0001	0.0003
e3nn / $S = 210$, w/o optim.	0.0084	0.1006	0.0366	0.0593	0.0046	0.0199

geometric metrics in Appendix C.2 confirm that FL and repulsion-refined FL are substantially more uniform than the e3nn grid.

Sampling-resolution guidance Appendix C.3 jointly reports accuracy, equivariance error, runtime, and peak memory. At $S = 128$, EST uses 2,538 MiB versus 1,624 MiB for the baseline; $S = 512$ uses 12,424 MiB while improving α MAE by only 0.0003. We recommend $S = \max\{64, (2L)^2\}$, increasing it only when stricter equivariance or scarce-data regularization justifies the quadratic attention-memory cost.

Latency The theoretically analysis of EST's complexity is privided in Appendix B.4. Here, we evaluate the computational efficiency of our EST against the Fully Connected Tensor Product (FCTP) and the Depthwise Tensor Product (DTP). We benchmarked the feed-forward latency of these operators across varying channel dimensions (C) and maximum degree (L). All experiments were conducted on a server equipped with an Intel(R) Xeon(R) Platinum 8558 CPU and an NVIDIA H200 GPU. Detailed results are recorded in Table 23. As illustrated in Fig. 2, the computational overhead of FCTP and DTP grows exponentially with L, whereas the EST remains remarkably stable. Furthermore, FCTP exhibits high sensitivity to channel scaling. Its latency increases sharply as C expands from 32 to 512. In contrast, both DTP and EST demonstrate superior scalability with respect to channel depth. Notably, at $C = 512$ and $L = 6$, the EST achieves an approximate $132\times$ speedup over the traditional FCTP. This efficiency stems not only from our architectural design but also from the integration of deep learning engineering and infrastructure, e.g., hardware-friendly Transformations.

5 Conclusion

We present EST, a plug-and-play spherical Transformer that adds cross-orientation interactions to equivariant molecular models. The continuous operator is exactly equivariant on a rotation-closed domain, whereas the finite implementation is approximately equivariant. A controlled replacement in SCN isolates gains from cross-orientation attention, and experiments on OC20, MPtrj, and QM9 show the largest benefits on complex tasks with modeling headroom. NVE, relaxation, and vibrational tests further support applying EST in dynamical and structural workflows, particularly with conservative gradient forces. Limitations are the $O(S^2)$ attention memory, finite-sampling equivariance error, increased non-conservativity with a direct force head, and attenuated gains on deeper or saturated backbones. The recommended $S = \max\{64, (2L)^2\}$ balances accuracy and equivariance against memory.

Acknowledgments

We thank our collaborators for their valuable discussions and support. We also thank SAIS and the Golab platform for providing the research environment and infrastructure that supported this work.

References

[1] Andrew W Senior, Richard Evans, John Jumper, James Kirkpatrick, Laurent Sifre, Tim Green, Chongli Qin, Augustin Žídek, Alexander WR Nelson, Alex Bridgland, et al. Improved protein structure prediction using potentials from deep learning. *Nature*, 577(7792):706–710, 2020.

[2] Dávid Péter Kovács, J Harry Moore, Nicholas J Browning, Ilyes Batatia, Joshua T Horton, Yixuan Pu, Venkat Kapil, William C Witt, Ioan-Bogdan Magdau, Daniel J Cole, et al. Mace-

off: Short-range transferable machine learning force fields for organic molecules. *Journal of the American Chemical Society*, 147(21):17598–17611, 2025.

[3] Bowen Deng, Peichen Zhong, KyuJung Jun, Janosh Riebesell, Kevin Han, Christopher J Bartel, and Gerbrand Ceder. Chgnet as a pretrained universal neural network potential for charge-informed atomistic modelling. *Nature Machine Intelligence*, 5(9):1031–1041, 2023.

[4] C Lawrence Zitnick, Lowik Chanussot, Abhishek Das, Siddharth Goyal, Javier Heras-Domingo, Caleb Ho, Weihua Hu, Thibaut Lavril, Aini Palizhati, Morgane Riviere, et al. An introduction to electrocatalyst design using machine learning for renewable energy storage. *arXiv preprint arXiv:2010.09435*, 2020.

[5] 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. *Neurips 2018*, 2018.

[6] Johannes Brandstetter, Rob Hesselink, Elise van der Pol, Erik J Bekkers, and Max Welling. Geometric and physical quantities improve e (3) equivariant message passing. *The Tenth International Conference on Learning Representations*, 2022.

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

[8] Nicholas Lubbers, Justin S Smith, and Kipton Barros. Hierarchical modeling of molecular energies using a deep neural network. *The Journal of chemical physics*, 148(24):241715, 2018.

[9] Benjamin Kurt Miller, Mario Geiger, Tess E Smidt, and Frank Noé. Relevance of rotationally equivariant convolutions for predicting molecular properties. *arXiv preprint arXiv:2008.08461*, 2020.

[10] Johannes Gasteiger, Janek Groß, and Stephan Günnemann. Directional message passing for molecular graphs. *ICLR2020*, 2020.

[11] 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.

[12] Yi-Lun Liao and Tess Smidt. Equiformer: Equivariant graph attention transformer for 3d atomistic graphs. In *The Eleventh International Conference on Learning Representations*, 2023.

[13] Sarp Aykent and Tian Xia. Gotennet: Rethinking efficient 3d equivariant graph neural networks. In *The Thirteenth International Conference on Learning Representations (ICLR)*, 2025.

[14] Yunyang Li, Lin Huang, Zhihao Ding, Xinran Wei, Chu Wang, Han Yang, Zun Wang, Chang Liu, Yu Shi, Peiran Jin, et al. E2former: An efficient and equivariant transformer with linear-scaling tensor products. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025.

[15] Nadav Dym and Haggai Maron. On the universality of rotation equivariant point cloud networks. In *International Conference on Learning Representations*, 2021.

[16] Chaitanya K Joshi, Cristian Bodnar, Simon V Mathis, Taco Cohen, and Pietro Liò. On the expressive power of geometric graph neural networks. *arXiv preprint arXiv:2301.09308*, 2023.

[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? *Advances in Neural Information Processing Systems*, 37:26238–26266, 2024.

[18] Junjie Wang, Yong Wang, Haoting Zhang, Ziyang Yang, Zhixin Liang, Jiuyang Shi, Hui-Tian Wang, Dingyu Xing, and Jian Sun. E (n)-equivariant cartesian tensor message passing interatomic potential. *Nature communications*, 15(1):7607, 2024.

[19] C Lawrence Zitnick, Abhishek Das, Adeesh Kolluru, Janice Lan, Muhammed Shuaibi, Anuroop Sriram, Zachary Ulissi, and Brandon Wood. Spherical channels for modeling atomic interactions. *Neurips 2022*, 2022.

[20] Saro Passaro and C Lawrence Zitnick. Reducing SO(3) convolutions to SO(2) for efficient equivariant gnns. volume 202 of *Proceedings of Machine Learning Research*, pages 27420–27438. PMLR, 2023.

[21] Yi-Lun Liao, Brandon M Wood, Abhishek Das, and Tess Smidt. Equiformerv2: Improved equivariant transformer for scaling to higher-degree representations. In *The Twelfth International Conference on Learning Representations*, 2024.

[22] Marc Finzi, Gregory Benton, and Andrew G Wilson. Residual pathway priors for soft equivariance constraints. *Advances in Neural Information Processing Systems*, 34:30037–30049, 2021.

[23] Tycho van der Ouderaa, David W Romero, and Mark van der Wilk. Relaxing equivariance constraints with non-stationary continuous filters. *Advances in Neural Information Processing Systems*, 35:33818–33830, 2022.

[24] Rui Wang, Robin Walters, and Rose Yu. Approximately equivariant networks for imperfectly symmetric dynamics. In *International Conference on Machine Learning*, pages 23078–23091, 2022.

[25] David W. Romero and Suhas Lohit. Learning partial equivariances from data. In *Advances in Neural Information Processing Systems*, 2022.

[26] Ines Chami, Zhitao Ying, Christopher Ré, and Jure Leskovec. Hyperbolic graph convolutional neural networks. *Advances in neural information processing systems*, 32, 2019.

[27] Simon Batzner, Albert Musaelian, Lixin Sun, Mario Geiger, Jonathan P Mailoa, Mordechai Kornbluth, Nicola Molinari, Tess E Smidt, and Boris Kozinsky. E (3)-equivariant graph neural networks for data-efficient and accurate interatomic potentials. *Nature communications*, 13(1):1–11, 2022.

[28] 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.

[29] Shengjie Luo, Tianlang Chen, and Aditi S. Krishnapriyan. Enabling efficient equivariant operations in the fourier basis via gaunt tensor products. In *The Twelfth International Conference on Learning Representations (ICLR)*, 2024.

[30] Xiang Fu, Brandon M Wood, Luis Barroso-Luque, Daniel S Levine, Meng Gao, Misko Dzamba, and C Lawrence Zitnick. Learning smooth and expressive interatomic potentials for physical property prediction. *arXiv preprint arXiv:2502.12147*, 2025.

[31] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

[32] Ruibin Xiong, Yunchang Yang, Di He, Kai Zheng, Shuxin Zheng, Chen Xing, Huishuai Zhang, Yanyan Lan, Liwei Wang, and Tieyan Liu. On layer normalization in the transformer architecture. In *International conference on machine learning*, pages 10524–10533, 2020.

[33] Mario Geiger, Tess Smidt, Alby M., Benjamin Kurt Miller, Wouter Boomsma, Bradley Dice, Kostiantyn Lapchevskyi, Maurice Weiler, Micha Tyszkiewicz, Simon Batzner, Dylan Madisetti, Martin Uhrin, Jes Frellsen, Nuri Jung, Sophia Sanborn, Mingjian Wen, Josh Rackers, Marcel Rød, and Michael Bailey. Euclidean neural networks: e3nn, April 2022.

[34] Álvaro González. Measurement of areas on a sphere using fibonacci and latitude–longitude lattices. *Mathematical geosciences*, 42:49–64, 2010.

[35] Jianlin Su, Murtadha Ahmed, Yu Lu, Shengfeng Pan, Wen Bo, and Yunfeng Liu. Roformer: Enhanced transformer with rotary position embedding. *Neurocomputing*, 568:127063, 2024.

[36] 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, 2021.

[37] Johannes Klicpera, 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.

[38] Eric Qu and Aditi Krishnapriyan. The importance of being scalable: Improving the speed and accuracy of neural network interatomic potentials across chemical domains. *Advances in Neural Information Processing Systems*, 37:139030–139053, 2024.

[39] Muhammed Shuaibi, Adeesh Kolluru, Abhishek Das, Aditya Grover, Anuroop Sriram, Zachary Ulissi, and C Lawrence Zitnick. Rotation invariant graph neural networks using spin convolutions. *arXiv preprint arXiv:2106.09575*, 2021.

[40] Johannes Gasteiger, Muhammed Shuaibi, Anuroop Sriram, Stephan Günnemann, Zachary Ward Ulissi, C. Lawrence Zitnick, and Abhishek Das. Gemnet-OC: Developing graph neural networks for large and diverse molecular simulation datasets. *Transactions on Machine Learning Research*, 2022.

[41] Yi Liu, Limei Wang, Meng Liu, Xuan Zhang, Bora Oztekin, and Shuiwang Ji. Spherical message passing for 3d graph networks. *ICLR2022*, 2021.

[42] 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.

[43] Philipp Thölke and Gianni De Fabritiis. Equivariant transformers for neural network based molecular potentials. In *International Conference on Learning Representations*, 2022.

[44] Tuan Le, Frank Noé, and Djork-Arné Clevert. Equivariant graph attention networks for molecular property prediction. *arXiv preprint arXiv:2202.09891*, 2022.

[45] Yi-Lun Liao, Tess Smidt, Muhammed Shuaibi, and Abhishek Das. Generalizing denoising to non-equilibrium structures improves equivariant force fields. *arXiv preprint arXiv:2403.09549*, 2024.

[46] Luis Barroso-Luque, Muhammed Shuaibi, Xiang Fu, Brandon M Wood, Misko Dzamba, Meng Gao, Ammar Rizvi, C Lawrence Zitnick, and Zachary W Ulissi. Open materials 2024 (omat24) inorganic materials dataset and models. *arXiv preprint arXiv:2410.12771*, 2024.

[47] Ilyes Batatia, Philipp Benner, Yuan Chiang, Alin M Elena, Dávid P Kovács, Janosh Riebesell, Xavier R Advincula, Mark Asta, Matthew Avaylon, William J Baldwin, et al. A foundation model for atomistic materials chemistry. *arXiv preprint arXiv:2401.00096*, 2023.

APPENDIX

A The Mathematics **14**

A.1 Message Passing Neural Networks 14

A.2 The Mathematics of Spherical Harmonics 15

A.3 Relationship Between Expressive Power and Equivariant Operations . 18

B Proofs and Details For Section 3 **18**

B.1 Proof for Equivariance of EST 18

B.2 Approximation Argument for the Expressiveness Intuition 21

B.3 Equivariance of Relative Orientation Embedding 22

B.4 Complexity Analysis 22

C Controlled Ablations and Sampling Trade-offs **23**

C.1 Controlled replacement in SCN 23

C.2 Sampling uniformity 24

C.3 Resolution, accuracy, equivariance, and memory 24

C.4 Physical validity for dynamics and structure optimization 24

D Details of Experiments and Supplementary Experiments **25**

D.1 Implementation Details of Baselines 25

D.2 Implementation Details of EST Experiments 27

D.3 Supplementary GotenNet Experiments 27

D.4 Supplementary Experiments 28

E Hybrid Branch and Architecture Details **30**

E.1 Integration into equivariant architectures 30

A The Mathematics

A.1 Message Passing Neural Networks

Consider a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, with nodes $v_i \in \mathcal{V}$ and edges $e_{ij} \in \mathcal{E}$. Each node v_i has an embedding $\mathbf{x}_i$ and an attribute $\mathbf{z}_i$, and each edge e_{ij} has an attribute $\mathbf{a}_{ij}$. The Message Passing Neural Network (MPNN) [7], a specific type of GNN, updates node embeddings through a message block $\mathbf{M}(\cdot)$ and an update block $\mathbf{U}(\cdot)$ via the following steps:

$$\mathbf{m}_{ij} = \mathbf{M}(\mathbf{x}_i, \mathbf{x}_j, \mathbf{z}_i, \mathbf{z}_j, \mathbf{a}_{ij}, \vec{\mathbf{r}}_{ij}), \quad \text{and} \quad \tilde{\mathbf{x}}_i = \mathbf{U}(\mathbf{x}_i, \sum_{j \in \mathcal{N}(i)} \mathbf{m}_{ij}), \tag{12}$$

where we use $\vec{\mathbf{r}}_{ij}$ to denote the relative position of node i and node j in 3D space, and the neighborhood $\mathcal{N}(i)$ is typically defined by a cutoff radius: $\mathcal{N}(i) = \{j \mid \|\vec{\mathbf{r}}_{ij}\| \leq r_{cut}\}$. The node attribute $\mathbf{z}_i$ contains atomic information such as the atomic type, and the edge attribute $\mathbf{a}_{ij}$ contains atomic pair information such as distance and bond type. By stacking multiple message and update blocks, the final node embeddings can be used to model atomic interactions or represent molecular properties.

A.2 The Mathematics of Spherical Harmonics

A.2.1 The Properties of Spherical Harmonics

The spherical harmonics $Y^{(l,m)}(\theta,\varphi)$ are the angular portion of the solution to Laplace's equation in spherical coordinates where azimuthal symmetry is not present. Some care must be taken in identifying the notational convention being used. In this entry, θ is taken as the polar (colatitudinal) coordinate with θ in $[0,\pi]$, and φ as the azimuthal (longitudinal) coordinate with φ in $[0,2\pi)$.

Spherical harmonics satisfy the spherical harmonic differential equation, which is given by the angular part of Laplace's equation in spherical coordinates. If we define the solution of Laplace's equation as $F=\Phi(\varphi)\Theta(\theta)$, the equation can be transformed as:

$$\frac{\Phi(\varphi)}{\sin\theta}\frac{d}{d\theta}\left(\sin\theta\frac{d\Theta}{d\theta}\right)+\frac{\Theta(\theta)}{\sin^2\theta}\frac{d^2\Phi(\varphi)}{d\varphi^2}+l(l+1)\Theta(\theta)\Phi(\varphi)=0 \tag{13}$$

Here we omit the derivation process and just show the result. The (complex-value) spherical harmonics are defined by:

$$Y^{(l,m)}(\theta,\varphi)\equiv\sqrt{\frac{2l+1}{4\pi}\frac{(l-m)!}{(l+m)!}}P^{(l,m)}(\cos\theta)e^{im\varphi}, \tag{14}$$

where $P^{(l,m)}(\cos\theta)$ is an associated Legendre polynomial. Spherical harmonics are integral basis, which satisfy:

$$\begin{aligned}&\int_0^{2\pi}\int_0^{\pi}Y^{(l_1,m_1)}(\theta,\varphi)Y^{(l_2,m_2)}(\theta,\varphi)Y^{(l_3,m_3)}(\theta,\varphi)\sin\theta d\theta d\varphi\\&=\sqrt{\tfrac{(2l_1+1)(2l_2+1)(2l_3+1)}{4\pi}}\begin{pmatrix}l_1&l_2&l_3\\0&0&0\end{pmatrix}\begin{pmatrix}l_1&l_2&l_3\\m_1&m_2&m_3\end{pmatrix},\end{aligned} \tag{15}$$

where $\begin{pmatrix}l_1&l_2&l_3\\m_1&m_2&m_3\end{pmatrix}$ is a Wigner 3j-symbol (which is related to the Clebsch-Gordan coefficients). We list a few spherical harmonics which are:

$$\begin{aligned}Y^{(0,0)}(\theta,\varphi)&=\frac{1}{2}\sqrt{\frac{1}{\pi}},\\Y^{(1,-1)}(\theta,\varphi)&=\frac{1}{2}\sqrt{\frac{3}{2\pi}}\sin\theta e^{-i\varphi},\\Y^{(1,0)}(\theta,\varphi)&=\frac{1}{2}\sqrt{\frac{3}{\pi}}\cos\theta,\\Y^{(1,1)}(\theta,\varphi)&=\frac{-1}{2}\sqrt{\frac{3}{2\pi}}\sin\theta e^{i\varphi},\\Y^{(2,-2)}(\theta,\varphi)&=\frac{1}{4}\sqrt{\frac{15}{2\pi}}\sin^2\theta e^{-2i\varphi},\\Y^{(2,-1)}(\theta,\varphi)&=\frac{1}{2}\sqrt{\frac{15}{2\pi}}\sin\theta\cos\theta e^{-i\varphi},\\Y^{(2,0)}(\theta,\varphi)&=\frac{1}{4}\sqrt{\frac{5}{\pi}}\left(3\cos^2\theta-1\right),\\Y^{(2,1)}(\theta,\varphi)&=\frac{-1}{2}\sqrt{\frac{15}{2\pi}}\sin\theta\cos\theta e^{i\varphi},\\Y^{(2,2)}(\theta,\varphi)&=\frac{1}{4}\sqrt{\frac{15}{2\pi}}\sin^2\theta e^{2i\varphi},\end{aligned} \tag{16}$$

In this work, we use the real-value spherical harmonics rather than the complex-value one, which can be written as :

$$
\begin{aligned}
Y^{0,0}(\theta,\varphi) &= \sqrt{\frac{1}{4\pi}}, \\
Y^{(1,-1)}(\theta,\varphi) &= \sqrt{\frac{3}{4\pi}}\sin\varphi\sin\theta, \\
Y^{(1,0)}(\theta,\varphi) &= \sqrt{\frac{3}{4\pi}}\cos\theta, \\
Y^{(1,1)}(\theta,\varphi) &= \sqrt{\frac{3}{4\pi}}\cos\varphi\sin\theta, \\
Y^{(2,-2)}(\theta,\varphi) &= \sqrt{\frac{15}{16\pi}}\sin(2\varphi)\sin^2\theta, \\
Y^{(2,-1)}(\theta,\varphi) &= \sqrt{\frac{15}{4\pi}}\sin\varphi\sin\theta\cos\theta, \\
Y^{(2,0)}(\theta,\varphi) &= \sqrt{\frac{5}{16\pi}}(3\cos^2\theta-1), \\
Y^{(2,1)}(\theta,\varphi) &= \sqrt{\frac{15}{4\pi}}\cos\varphi\sin\theta\cos\theta, \\
Y^{(2,2)}(\theta,\varphi) &= \sqrt{\frac{15}{16\pi}}\cos(2\varphi)\sin^2\theta.
\end{aligned}
\tag{17}
$$

A.2.2 Fourier transformation over $\mathbb{S}^2$

In the main paper, we show that any square-integrable function $f(\cdot)$ can thus be expanded as a linear combination of spherical harmonics:

$$
f(\vec{\mathbf{p}}) = \sum_{l=0}^{\infty}\sum_{m=-l}^{l}\mathbf{x}^{(l,m)}Y^{(l,m)}(\vec{\mathbf{p}}), \tag{18}
$$

where $\vec{\mathbf{p}} = (\theta,\varphi)$ denotes the orientations, like what we do in the main paper. The coefficient $\mathbf{x}^{(l,m)}$ can be obtained by the inverse transformation over $\mathbb{S}^2$, which is

$$
\mathbf{x}^{l,m} = \int_{\Omega} f(\vec{\mathbf{p}})Y^{(l,m)*}(\vec{\mathbf{p}})d\Omega = \int_0^{2\pi}d\varphi\int_0^{\pi}d\theta\sin\theta f(\vec{\mathbf{p}})Y^{(l,m)*}(\vec{\mathbf{p}}). \tag{19}
$$

Using the fact $\mathbf{Y}^l(\mathbf{R}\vec{\mathbf{p}}) = \mathbf{D}^l(\mathbf{R})\mathbf{Y}^l(\vec{\mathbf{p}})$, and equation (18), we know

$$
f(\mathbf{R}\vec{\mathbf{p}}) = \sum_{l=0}^{\infty}\sum_{m=-l}^{l}\mathbf{x}^{(l,m)}Y^{(l,m)}(\mathbf{R}\vec{\mathbf{p}}) = \sum_{l=0}^{\infty}\sum_{m=-l}^{l}\mathbf{x}^{(l,m)}\mathbf{D}Y^{(l,m)}(\vec{\mathbf{p}}). \tag{20}
$$

Therefore, we can get the conclusion that spatial representation $f(\mathbf{R}\vec{\mathbf{P}})$ and $f(\vec{\mathbf{P}})$ is steerable, which can be represented by

$$
f(\mathbf{R}\vec{\mathbf{p}}) = \mathbf{D}^{-1}\mathbf{x} = \mathbf{D}^T\mathbf{x}. \tag{21}
$$

A.2.3 The Relationship Between Spherical Harmonics and Wigner-D Matrix

One can also think of standard spherical harmonics as functions $\tilde{Y}_m^{(l)} : \mathrm{SO}(3) \to \mathbb{R}$ on $\mathrm{SO}(3)$ that are invariant to a sub-group of rotations via

$$
Y^{(l,m)}(\mathbf{n}_{\alpha,\beta}) = Y^{(l,m)}(\mathbf{R}_{\alpha,\beta,\gamma}\mathbf{n}_x) =: \tilde{Y}^{(l,m)}(\mathbf{R}_{\alpha,\beta,\gamma})\ .
$$

Then, by definition, $\tilde{Y}(l,m)$ is invariant with respect to rotation angle γ, i.e.., $\forall_{\gamma\in[0,2\pi)}$: $\tilde{Y}^{(l,m)}(\mathbf{R}_{\alpha,\beta,\gamma}) = \tilde{Y}^{(l,m)}(\mathbf{R}_{\alpha,\beta,0})$. This viewpoint of regarding the spherical harmonics as γ-invariant functions on $\mathrm{O}(3)$ helps us to draw the connection to the Wigner-D functions $D_{mn}^{(l)}$ that

make up the $2l+1 \times 2l+1$ elements of the Wigner-D matrices. Namely, the $n=0$ column of Wigner-D functions are also γ-invariant and, in fact, correspond (up to a normalisation factor) to the spherical harmonics via

$$Y^{(l,m)}(\mathbf{n}_{\alpha,\beta}) = \frac{1}{\sqrt{2l+1}} D^{(l)}_{m0}(\mathbf{R}_{\alpha,\beta,\gamma}) \,. \tag{22}$$

A.2.4 Equivariance of Clebsch-Gordan Tensor Product

The Clebsch-Gordan Tensor Product shows a strict equivariance for different group representations, which make the mixture representations transformed equivariant based on Wigner-D matrices. We use $\mathbf{D}^{l,m'm}$ to denote the element of Wigner-D matrix in the degree l. The Clebsch-Gordan coefficient satisfies:

$$\begin{aligned} &\textstyle\sum_{m_1',m_2'} \mathcal{C}^{(l_0,m_0)}_{(l_1,m_1')(l_2,m_2')} \mathbf{D}^{(l_1,m_1'm_1)}(g) \mathbf{D}^{(l_2,m_2'm_2)}(g) \\ &= \textstyle\sum_{m_0'} \mathbf{D}^{(l_0,m_0'm_0)}(g) \mathcal{C}^{(l_0,m_0')}_{(l_1,m_1)(l_2,m_2)} \end{aligned} \tag{23}$$

Therefore, the spherical harmonics can be combined equivariantly by CG Tensor Product:

$$\begin{aligned} &CG\left(\sum_{m_1'} \mathbf{D}^{(l_1,m_1m_1')}(g) Y^{(l_1,m_1')}, \sum_{m_2'} \mathbf{D}^{(l_2,m_2m_2')}(g) Y^{(l_2,m_2')}\right) \\ =& \sum_{m_1,m_2} \mathcal{C}^{(l_0,m_0)}_{(l_1,m_1)(l_2,m_2)} \sum_{m_1'} \mathbf{D}^{(l_1,m_1m_1')}(g) Y^{(l_1,m_1')} \sum_{m_2'} \mathbf{D}^{(l_2,m_2m_2')}(g) Y^{(l_2,m_2')} \\ =& \sum_{m_0'} \mathbf{D}^{(l_0,m_0m_0')}(g) \sum_{m_1,m_2} \mathcal{C}^{(l_0m_0')}_{(l_1,m_1)(l_2,m_2)} Y^{(l_1,m_1')} Y^{(l_2,m_2')} \\ =& \sum_{m_0'} \mathbf{D}^{(l_0,m_0m_0')}(g) CG\left(Y^{(l_1,m_1')}, Y^{(l_2,m_2')}\right). \end{aligned} \tag{24}$$

Here, we omit the input argument of the spherical harmonics, which can represent any direction on the sphere. Equation (24) represents a relationship between scalar. If we transform the scalar to vector or matrix like what we do in equation (2), equation (24) is equal to

$$(\mathbf{D}^{l_1}\mathbf{u} \otimes \mathbf{D}^{l_2}\mathbf{v})^{l_0} = \mathbf{D}^{l_0}(\mathbf{u} \otimes \mathbf{v})^{l_0}. \tag{25}$$

The tensor CG product mixes two representations to a new representation under special rule $|l_1 - l_2| \leq l \leq (l_1 + l_2)$. For example, 1.two type-0 vectors will only generate a type-0 representations; 2.type-l_1 and type-l_2 can generate type-$l_1 + l_2$ vector at most. Note that some widely-used products are related to tensor product: scalar product ($l_1 = 0, l_2 = 1, l = 1$), dot product ($l_1 = 1, l_2 = 1, l = 0$) and cross product ($l_1 = 1, l_2 = 1, l = 1$). However, for each element with $l > 0$, there are multi mathematical operation for the connection with weights. The relation between number of operations and degree is quadratic. Thus, as degree increases, the amount of computation increases significantly, making calculation of the CG tensor product slow for higher order irreps. This statement can be proven by the implementation of e3nn (o3.FullyConnectedTensorProduct).

A.2.5 Learnable Parameters in Tensor Product

Previous works utilize the e3nn library [33] to implement the corresponding tensor product. It is crucial to emphasize that the formulation of CG tensor product is devoid of any learnable parameters, as CG coefficients remain constant. In the context of e3nn, learnable parameters are introduced into each path, represented as $w(\mathbf{u}^{l_1} \otimes \mathbf{v}^{l_2})$. Importantly, these learnable parameters will not destory the equivariance of each path. However, they are limited in capturing directional information. In equivariant models, the original CG tensor product primarily captures directional information. We have previously mentioned our replacement of the CG tensor product with learnable modules. It is worth noting that our focus lies on the CG coefficients rather than the learnable parameters in the e3nn implementation.

A.2.6 Gate Activation and Normalization

Gate Activation. In equivariant models, the Gate activation combines two sets of group representations. The first set consists of scalar steerable vector ($l = 0$), which are passed through standard

activation functions such as sigmoid, ReLU and SiLU. The second set comprises higher-order steerable vector (($l > 0$)), which are multiplied by an additional set of scalar steerable vector that are introduced solely for the purpose of the activation layer. These scalar steerable vector are also passed through activation functions.

Normalization. Normalization is a technique commonly used in neural networks to normalize the activations within each layer. It helps stabilize and accelerate the training process by reducing the internal covariate shift, which refers to the change in the distribution of layer inputs during training.

The normalization process involves computing the mean and variance across the channels. In equivariant normalization, the variance is computed using the root mean square value of the L2-norm of each type-l vector. Additionally, this normalization removes the mean term. The normalized activations are then passed through a learnable affine transformation without a learnable bias, which enables the network to adjust the mean and variance based on the specific task requirements.

A.3 Relationship Between Expressive Power and Equivariant Operations

In [15], Theorem 2 establishes the universality of equivariant networks based on the TFN structure:

Theorem. For all $n \in \mathbb{N}$, $\mathbf{l}_T \in {\mathbb{N}_+}^*$,

- 1. For $D \in \mathbb{N}_+$, every G-equivariant polynomial $p : \mathbb{R}^{3\times n} \to W_{\mathbf{l}_T}^n$ of degree D is in $F_{C(D),D}^{TFN}$.
- 2. Every continuous G-equivariant function can be approximated uniformly on compact sets by functions in $\cup_{D\in\mathbb{N}_+} F_{C(D),D}^{TFN}$.

Here, n represents the number of input points (or nodes), $\mathbf{l}_T$ represents the degree of the approximated G-equivariant function, C represents the number of channels, and D represents the degree of the TFN (Tensor Field Network) structure, which is equivalent to the term l used in our method. The TFN structure consists of two layers, including convolution and self-interaction. Self-interaction involves equivariant linear functions. The convolution operation calculates the CG tensor product between different steerable representations, which is a fundamental operation for transforming directional information. Most equivariant models based on group representations use a similar approach (CG tensor product) to capture directional features. Therefore, the theorem mentioned above also applies to building blocks based on CG tensor products, such as SEGNN [6] and Equiformer [12].

It is important to note that achieving an infinite degree in practice is not feasible. However, equivariant models based on group representations can enhance their expressive power by increasing the number of maximal degrees [15]. In their evaluation of expressive power, as presented in [16], the authors utilize the GWL (geometric Weisfeiler-Leman) graph isomorphism test. In Table 2 of their work, it is evident that equivariant models with a maximal degree denoted as L are incapable of distinguishing n-fold symmetric structures when n exceeds the value of L.

B Proofs and Details For Section 3

B.1 Proof for Equivariance of EST

We prove Theorem 3.1 for a rotation-closed continuous domain Ω with rotation-invariant measure.

Proof. Recall that FT can be represented by

$$f(\vec{\mathbf{p}}) = \sum_{l=0}^{L} \sum_{m=-l}^{l} x^{(l,m)} Y^{(l,m)}(\vec{\mathbf{p}}) = (\mathbf{x}^{(0\to L)})^T \mathbf{Y}^{(0\to L)}(\vec{\mathbf{p}}). \tag{26}$$

When we apply a random rotation $\mathbf{D}^{-1}$ to $\mathbf{x}$, we obtain

$$
\begin{aligned}
&(\mathbf{D}^{-1}\mathbf{x}^{(0\to L)})^T\mathbf{Y}^{(0\to L)}(\vec{\mathbf{p}}) && (27)\\
=&(\mathbf{x}^{(0\to L)})^T\mathbf{D}\mathbf{Y}^{(0\to L)}(\vec{\mathbf{p}}) && (28)\\
=&(\mathbf{x}^{(0\to L)})^T\mathbf{Y}^{(0\to L)}(\mathbf{R}\vec{\mathbf{p}}) && (29)\\
=&f(\mathbf{R}\vec{\mathbf{p}}), && (30)
\end{aligned}
$$

where $\mathbf{R}$ and $\mathbf{D}$ share the same transformation parameters.

Given a spatial representation $f(\vec{\mathbf{p}})$ transformed from a steerable representation $\mathbf{x}$, spherical attention can be represented as:

$$
\tilde{f}(\vec{\mathbf{p}}_1) = \int_{\vec{\mathbf{p}}_2\in\Omega} a\big(f^Q(\vec{\mathbf{p}}_1), f^K(\vec{\mathbf{p}}_2)\big) f^V(\vec{\mathbf{p}}_2) d\vec{\mathbf{p}}_2, \tag{31}
$$

where $a(\cdot)$ denotes the operation to compute attention coefficients. Terms Q, K, V denote three independent linear transformations. When the origin representation $\mathbf{x}$ is rotation by a Wigner-D matrix $\mathbf{D}^{-1}$, the representation $f(\vec{\mathbf{p}})$ is transformed to $f(\mathbf{R}\vec{\mathbf{p}})$. Therefore, the attention results are changed to

$$
\begin{aligned}
&\int_{\mathbf{R}\vec{\mathbf{p}}_2\in\Omega} a\big(f^Q(\mathbf{R}\vec{\mathbf{p}}_1), f^K(\mathbf{R}\vec{\mathbf{p}}_2)\big) f^V(\mathbf{R}\vec{\mathbf{p}}_2) d\mathbf{R}\vec{\mathbf{p}}_2\\
=&\int_{\mathbf{R}\vec{\mathbf{p}}_2\in\Omega} \frac{\exp\big(f^Q(\mathbf{R}\vec{\mathbf{p}}_1) * f^K(\mathbf{R}\vec{\mathbf{p}}_2)\big)}{\int_{\mathbf{R}\vec{\mathbf{p}}_3\in\Omega}\exp\big(f^Q(\mathbf{R}\vec{\mathbf{p}}_1) * f^K(\mathbf{R}\vec{\mathbf{p}}_3)\big) d\mathbf{R}\vec{\mathbf{p}}_3} f^V(\mathbf{R}\vec{\mathbf{p}}_2) d\mathbf{R}\vec{\mathbf{p}}_2\\
=&\int_{\mathbf{R}\vec{\mathbf{p}}_2\in\Omega} \frac{\exp\big(f^Q(\mathbf{R}\vec{\mathbf{p}}_1) * f^K(\vec{\mathbf{p}}_2)\big)}{\int_{\vec{\mathbf{p}}_3\in\Omega}\exp\big(f^Q(\mathbf{R}\vec{\mathbf{p}}_1) * f^K(\vec{\mathbf{p}}_3)\big) d\vec{\mathbf{p}}_3} f^V(\vec{\mathbf{p}}_2) d\vec{\mathbf{p}}_3\\
=&\int_{\vec{\mathbf{p}}_2\in\Omega} \frac{\exp\big(f^Q(\mathbf{R}\vec{\mathbf{p}}_1) * f^K(\vec{\mathbf{p}}_2)\big)}{\int_{\vec{\mathbf{p}}_3\in\Omega}\exp\big(f^Q(\mathbf{R}\vec{\mathbf{p}}_1) * f^K(\vec{\mathbf{p}}_3)\big) d\vec{\mathbf{p}}_3} f^V(\vec{\mathbf{p}}_2) d\vec{\mathbf{p}}_2\\
=&\int_{\vec{\mathbf{p}}_2\in\Omega} a\big(f^Q(\mathbf{R}\vec{\mathbf{p}}_1), f^K(\vec{\mathbf{p}}_2)\big) f^V(\vec{\mathbf{p}}_2) d\vec{\mathbf{p}}_2\\
=&\tilde{f}(\mathbf{R}\vec{\mathbf{p}}_1).
\end{aligned} \tag{32}
$$

The changes of variables above are valid because $\mathbf{R}\Omega = \Omega$ and rotations preserve surface measure. Therefore, the output of spherical attention remains steerable after rotation. If we transform $\tilde{f}(\mathbf{R}\vec{\mathbf{p}}_1)$ into its steerable representation, denoted as $\tilde{\mathbf{x}}$, the following relationship holds:

$$
\tilde{\mathbf{x}} = \mathbf{D}^{-1}\mathcal{F}^{-1}\big(\mathrm{SA}(\mathcal{F}(\mathbf{x}))\big) = \mathcal{F}^{-1}\big(\mathrm{SA}(\mathcal{F}(\mathbf{D}^{-1}\mathbf{x}))\big) \tag{33}
$$

Thus spherical attention is equivariant on the closed continuous domain. A shared pointwise spherical FFN also commutes with rotation because it applies the same function at every orientation, completing the proof for the continuous EST operator.

In the derivation above, the rotational term is effectively omitted in the integration limits because the sphere is a symmetric set that is invariant under any rotation. Specifically, the continuous spherical surface Ω satisfies $\mathbf{R}\Omega \equiv \Omega$ for any $\mathbf{R} \in SO(3)$. This implies that for any direction $\vec{\mathbf{p}} \in \Omega$, the rotated point $\mathbf{R}\vec{\mathbf{p}}$ remains in Ω, and this mapping is bijective. Since the Jacobian determinant of a rotation matrix is 1, the integral measure is invariant, i.e., $d(\mathbf{R}\vec{\mathbf{p}}) = d\vec{\mathbf{p}}$. Thus, integrating over the rotated coordinates yields the same result as integrating over the original coordinates.

To illustrate this, consider a 2D analog: a periodic function on a circle $y = f(\theta)$. The integral over the circle is $\int_0^{2\pi} f(\theta)\, d\theta$. If we rotate the domain by a constant C, the function becomes $f(\theta + C)$. Due to the periodicity of the circle ($f(2\pi+\theta) = f(\theta)$), the integral $\int_0^{2\pi} f(\theta+C)\, d\theta$ yields the same result as the original integral. The 3D spherical case follows the same principle: rotation changes the relative positions of points, but the integral over the entire closed surface remains invariant.

Conclusion of Proof: Since $\tilde{f}(\mathbf{R}\vec{\mathbf{p}}_1)$ is the rotated output, if we transform it back into its steerable representation $\tilde{\mathbf{x}}$, the following relationship holds:

$$
\tilde{\mathbf{x}} = \mathbf{D}^{-1}\mathcal{F}^{-1}\big(\mathrm{SA}(\mathcal{F}(\mathbf{x}))\big) = \mathcal{F}^{-1}\big(\mathrm{SA}(\mathcal{F}(\mathbf{D}^{-1}\mathbf{x}))\big). \tag{34}
$$

Thus, the equivariance of spherical attention holds. Moreover, the spherical FFN is inherently equivariant because the function $\tilde{f}(\vec{\mathbf{p}}_1) = \mathrm{FFN}\big(f(\vec{\mathbf{p}}_1)\big)$ operates locally on each orientation. Therefore, the entire EST framework is equivariant. □

Discussion on Finite Sampling and Equivariance: Theorem 3.1 establishes exact equivariance only for the closed continuous domain. Practical EST replaces the integral by a finite quadrature and is therefore approximately equivariant for arbitrary rotations.

In the discrete case, the attention operation in equation (32) is approximated as:

$$\tilde{f}(\vec{\mathbf{p}}_1) = \sum_{\vec{\mathbf{p}}_2} a\big(f^Q(\vec{\mathbf{p}}_1), f^K(\vec{\mathbf{p}}_2)\big) f^V(\vec{\mathbf{p}}_2). \tag{35}$$

After introducing random rotations, this becomes:

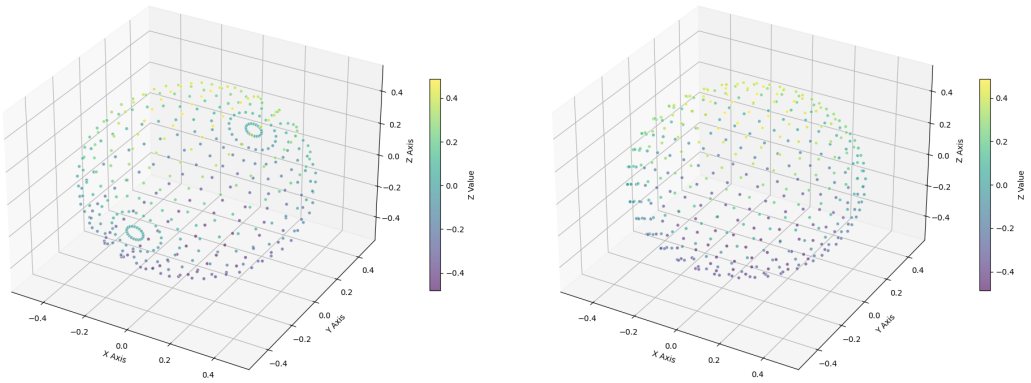
$$\tilde{f}(\mathbf{R}\vec{\mathbf{p}}_1) = \sum_{\vec{\mathbf{p}}_3} a\big(f^Q(\mathbf{R}\vec{\mathbf{p}}_1), f^K(\vec{\mathbf{p}}_3)\big) f^V(\vec{\mathbf{p}}_3). \tag{36}$$

A finite point set cannot be closed under all of $SO(3)$. Exact discrete equivariance holds only for rotations in the grid-symmetry subgroup that permute all samples; other rotations incur quadrature and interpolation mismatch.

For a grid-symmetry rotation, equation (36) equals equation (35) up to permutation. Uniformity alone does not create exact closure; it reduces the numerical mismatch for rotations outside this subgroup.

To mitigate this, we employ approximately uniform sampling methods, such as Fibonacci Lattices. Furthermore, as established in prior literature, neural networks are capable of approximating continuous equivariant functions. Rotations not strictly covered by the discrete sampling grid are handled via training, allowing the model to learn approximate equivariance. Our experiments confirm that the equivariance error reduces significantly after training without requiring explicit data augmentation.

Sampling strategies Most previous works use the e3nn implementation for spherical Fourier transform. However, it significantly destroy the uniformity of spherical sampling, which is illustrated in Figure 3(a). In contrast, Fibonacci Lattices (FL) do not directly divide the polar angle and azimuth angle into a grid. Instead, they select sampling points on the sphere in a spiral pattern. As shown in Figure 3(b), FL tends to achieve more uniform sampling, thereby improving the equivariance of EST. This is also consistent with the results observed in Table 6.



(a) e3nn Sampling (b) Fibonacci Lattices Sampling

Figure 3: Two spherical sampling strategies.

Enhanced Uniformity Building on the FL sampling, we further propose a scattering optimization strategy similar to molecular dynamics. We define a distance-based interaction force between every

two points: the greater the distance, the smaller the force. Then, we iterate two processes: 1. Simultaneously optimize the three-dimensional coordinates of all points through the interaction forces; 2. Re-project all points back onto the sphere. After multiple rounds of iteration, the spherical sampling points can become more uniform. It is worth noting that both the FL sampling and the dynamics simulation are conducted during the model initialization phase. Once we obtain the uniform sampling points, we save their results and directly read them in the model's feedforward process.

Self-conjugate Fourier Basis The orthogonality of the basis functions in equation (3) is defined by $\int_\Omega Y^{(l,m)}(\vec{\mathbf{p}})Y^{(l',m')*}(\vec{\mathbf{p}})d\Omega = \delta_{ll'}\delta_{mm'}$, where δ_{ij} is the Kronecker delta. This work utilizes real-valued spherical harmonics, which possess the property of being self-conjugate: $Y^{(l,m)*}(\vec{\mathbf{p}}) = Y^{(l,m)}(\vec{\mathbf{p}})$. Then we transform equation (4) to a discrete inverse FT:

$$x^{l,m} = \sum_{s=1}^{S} \lambda^{(l,m)} \mathbf{f}_s^* Y^{(l,m)*}(\vec{\mathbf{p}}_s). \tag{37}$$

To ensure orthogonality in the code implementation, we adopt the simple sum of squares rule to define the quadrature weights: $\lambda^{(l,m)} = \frac{1}{\sum_{s=1}^{S} |Y^{(l,m)}(\vec{\mathbf{p}}_s)|^2}$.

B.2 Approximation Argument for the Expressiveness Intuition

Claim 3.2 is an unproven intuition rather than a formal function-class inclusion. This section explains why a sufficiently expressive spatial network can approximate CG-induced local products; Appendix B.2 supplies controlled empirical support and counterexamples showing additional cross-orientation capacity.

Given two steerable vectors $\mathbf{u} \in \mathbb{V}_{0\to l_1}$ and $\mathbf{v} \in \mathbb{V}_{0\to l_2}$, and we define the CG tensor product result is $\mathbf{w} \in \mathbb{V}_{l_0}$, where $l_0 <= l_1 + l_2$. The spatial representations of $\mathbf{w}$ can be represented as:

$$\sum_{l_0,m_0} \mathbf{w}^{(l_0,m_0)} Y^{(l_0,m_0)} = \sum_{l_0,m_0} (\mathbf{u}^{(0\to l_1)^T} \mathbf{C}^{(l_0,m_0)}_{(0\to l_1),(0\to l_2)} \mathbf{v}^{(0\to l_2)}) Y^{(l_0,m_0)}, \tag{38}$$

where $\mathbf{C}^{(l_0,m_0)}_{(0\to l_1),(0\to l_2)}$ is a matrix including the whole CG coefficients corresponding to degree l_0 and order m_0. We temporarily ignore the input of the spherical harmonics. The spherical Transformer use the multiplication between spatial representations of $\mathbf{u}$ and $\mathbf{v}$:

$$\begin{aligned} &\sum_{l_1,m_1} \mathbf{u}^{(l_1,m_1)} Y^{(l_1,m_1)} \sum_{l_2,m_2} \mathbf{v}^{(l_2,m_2)} Y^{(l_2,m_2)} \\ =&\sum_{l_1,m_1,l_2,m_2} \mathbf{u}^{(l_1,m_1)} Y^{(l_1,m_1)} \mathbf{v}^{(l_2,m_2)} Y^{(l_2,m_2)} \\ =&\mathbf{u}^{(0\to l_1)^T} (\mathbf{Y}^{(0\to l_1)^T} \mathbf{Y}^{(0\to l_2)}) \mathbf{v}^{(0\to l_2)} \end{aligned} \tag{39}$$

Recall that CG coefficients are in fact the expansion coefficients of a product of two spherical harmonics in terms of a single spherical harmonic (see equation (15)):

$$Y^{(l_1,m_1)} Y^{(l_2,m_2)} = \sum_{l_0,m_0} \sqrt{\frac{(2l_1+1)(2l_2+1)}{4\pi(2l_0+1)}} \mathcal{C}^{(0,0)}_{(l_1,m_1)(l_2,m_2)} \mathcal{C}^{(l_0,m_0)}_{(l_1,m_1)(l_2,m_2)} Y^{l_0,m_0}, \tag{40}$$

equation (39) can be transformed to

$$\begin{aligned} &\mathbf{u}^{(0\to l_1)^T} (\mathbf{Y}^{(0\to l_1)^T} \mathbf{Y}^{(0\to l_2)}) \mathbf{v}^{(0\to l_2)} \\ =&\sum_{l_0,m_0} \mathbf{u}^{(0\to l_1)^T} \mathbf{H}^{(l_0,m_0)}_{(0\to l_1),(0\to l_2)} \mathbf{v}^{(0\to l_2)} Y^{(l_0,m_0)}, \end{aligned} \tag{41}$$

where $\mathbf{H}^{(l_0,m_0)}_{(l_1,m_1),(l_2,m_2)} = \sqrt{\frac{(2l_1+1)(2l_2+1)}{4\pi(2l_0+1)}} \mathcal{C}^{(0,0)}_{(l_1,m_1)(l_2,m_2)} \mathcal{C}^{(l_0,m_0)}_{(l_1,m_1)(l_2,m_2)}$. Compared to $\mathbf{C}^{(l_0,m_0)}_{(0\to l_1),(0\to l_2)}$, the term $\mathbf{H}^{(l_0,m_0)}_{(0\to l_1),(0\to l_2)}$ introduces an additional constant term that can be approximated by linear layers and the FFNs. Therefore, equation (39) is equivalent to equation (38). On

the other hand, equation (39) is inherently part of the Transformer architecture: when relationships between different orientations are masked and the $\mathbf{V}$ vectors are taken as constant vectors, the Transformer can naturally reduce to equation (39).

Through the above process, we find that the spherical representation of the output from the CG tensor product is fundamentally a special case of EST. Furthermore, *EST can capture dependencies between different orientations, which is particularly beneficial for approximating higher degree spherical harmonic representations. For instance, at a orientation* (θ, φ)*, higher-degree spherical harmonics may involve contributions from other orientations such as* $(2\theta, \varphi), (\theta, 2\varphi), (2\theta, 2\varphi)$ *(see* $Y^{(2)}$ *in equation* (17)*). The Transformer in EST can directly combine these lower-degree terms in these orientations to approximate the higher-order terms at the orientation* (θ, φ)*.*

We propose an alternative interpretation of the tensor product and spherical harmonics. The traditional tensor product can combine steerable vectors of any degree combination, and its functional form is bilinear. When we transform these steerable vectors to the spatial domain, the information from the different vectors, regardless of their degree, is projected onto a sphere. This projection allows a Transformer structure on a sphere to effectively capture the dependencies between the input steerable vectors. Furthermore, the Spherical Fourier Transform ($\mathcal{F}$) and its inverse ($\mathcal{F}^{-1}$) can be expressed as two matrix multiplications: $\mathcal{F}(\mathbf{x}) = W_{FT}\mathbf{x} = \mathbf{f}^*and\mathcal{F}^{-1}(\mathbf{f}^*) = W_{IFT}\mathbf{f}^* = \mathbf{x}$. Here, W_{FT} and W_{IFT} are matrices of size $S \times (L+1)^2$ and $(L+1)^2 \times S$, respectively, which contain the Fourier basis. The vectors $\mathbf{x}$ and $\mathbf{f}^*$ are the steerable representation and spherical representation, with sizes $(L+1)^2 \times 1$ and $S \times 1$.

Controlled fitting evidence. We compare parameter-matched local FCTP (10,033 parameters), S2Activation (9,793), and EST (8,744) students on analytic local targets and on a nonlocal one-head attention target. Values below are test NRMSE over three seeds. The local rows support approximate correspondence between spectral CG products and pointwise spherical products at finite bandlimit. On the nonlocal target, explicit EST weights reproduce the teacher with maximum absolute error 1.19×10^{-7}, while the trained EST reaches 0.001872 NRMSE versus 0.418440 for FCTP and 0.435842 for S2Activation. This is controlled evidence for the intuition in Claim 3.2, not a proof of unrestricted function-class inclusion.

Table 7: Controlled expressiveness fitting (test NRMSE, mean $\pm$ std.).

Target	FCTP	EST	S2Activation
FCTP-local	0.033470 ± 0.000085	0.021227 ± 0.000748	0.018035 ± 0.000880
S2Activation-local	0.034956 ± 0.000022	0.017678 ± 0.000408	0.015832 ± 0.001622
EST-attention	0.418440 ± 0.000845	$\mathbf{0.001872 \pm 0.000088}$	0.435842 ± 0.002330

B.3 Equivariance of Relative Orientation Embedding

The equivariance of relative orientation embedding can be easily proven with the same way in equation (32), where $a\big(f^Q(\mathbf{R}\vec{\mathbf{p}}_1), f^K(\mathbf{R}\vec{\mathbf{p}}_2)\big)$ is transformed to

$$\frac{\exp\big(f^Q(\mathbf{R}\vec{\mathbf{p}}_1) * f^K(\mathbf{R}\vec{\mathbf{p}}_2) + (\mathbf{R}\vec{\mathbf{p}}_1)^T(\mathbf{R}\vec{\mathbf{p}}_2)\big)}{\int_{\mathbf{R}\vec{\mathbf{p}}_3 \in \Omega} \exp\big(f^Q(\mathbf{R}\vec{\mathbf{p}}_1) * f^K(\mathbf{R}\vec{\mathbf{p}}_3) + (\mathbf{R}\vec{\mathbf{p}}_1)^T(\mathbf{R}\vec{\mathbf{p}}_3)\big) d\mathbf{R}\vec{\mathbf{p}}_3}. \tag{42}$$

Due to the spherical symmetry, we can still eliminate the rotation $\mathbf{R}$ acting on $\mathbf{p}_2$ and $\mathbf{p}_3$. Therefore, after incorporating the Relative Orientation Embedding, EST retains its equivariance.

We further elaborate on the contribution of the relative orientation embedding. In Figure 4(left), EST without positional encoding cannot distinguish locally restructured spherical functions, but it can after using positional encoding (Figure 4(middle)). In Figure 4 (right), the relative positional encoding ensures the overall equivariance of the spherical functions.

B.4 Complexity Analysis

The complexity of the CG tensor product is related to the maximum degree L: $\mathcal{O}(L^6)$. In contrast, the complexity of the Transformer in EST is related to the spherical sampling number S:

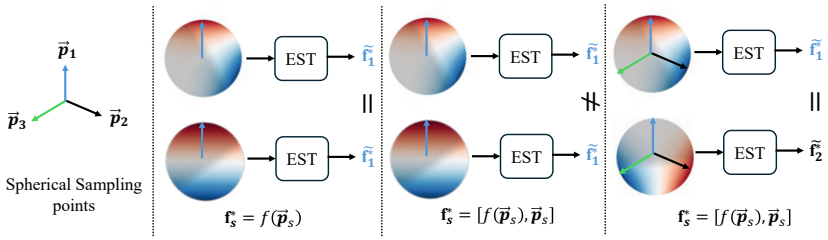


Figure 4: **An example of relative orientation embedding.** The three figures on the right illustrate distinct scenarios: the left and middle panels represent locally restructured spherical functions where the set of orientations has been reordered. Theoretically, the values in $\vec{\mathbf{p}}_1$ for these two spherical functions should differ after feeding EST; however, an EST without relative orientation embedding cannot distinguish between them. In contrast, the right panel illustrates globally rotated spherical functions. In this case, the spherical functions are identical under rotation, meaning the value in $\vec{\mathbf{p}}_1$ for the upper function should theoretically match the value in $\vec{\mathbf{p}}_2$ for the lower function after feeding EST.

$\mathcal{O}(S^2)$ (self-attention). Based on the Nyquist sampling rate, the minimum number of samples on the spatial domain is at least $(2L)^2$, meaning the complexity of the Transformer in EST is $\mathcal{O}(16L^4)$. The complexity of FFT and IFFT is $2 \cdot S \cdot L^2 \geq 8L^4$. Therefore, the total complexity of EST is $\mathcal{O}(24L^4)$. When $L > 4$, EST theoretically has lower computational cost. Moreover, in practical implementation, we also need to consider the impact of channel expansion on computational complexity. **Transformers can directly fuse multi-channel signals, allowing the channel dimension C in EST to be freely scaled, with complexity growing linearly with respect to C: $\mathcal{O}(24CL^4)$**. In contrast, the CG tensor product operation can only handle two single-channel irreducible representations at a time. When extending to multiple channels, if we consider pairwise interactions between all channels, the complexity increases to $\mathcal{O}(C^2L^6)$. Some methods only consider interactions between a subset of channels (e.g., Equiformer DTP), reducing the complexity to $\mathcal{O}(CL^6)$, but at the cost of reduced representational capacity. In summary, EST offers greater flexibility in channel expansion while maintaining lower computational complexity.

EST employs a standard fixed-length Transformer in its implementation, where the sequence length is S, which enables it to benefit from various engineering optimizations. The results we presented in Figure 2 already demonstrate that it can gain advantages from some hardware-friendly operations, such as

```
import torch.nn.functional as F
out = F.scaled_dot_product_attention(q, k, v)
```

For larger models, such benefits will be even more extensive, such as using Flash Attention and so on.

C Controlled Ablations and Sampling Trade-offs

C.1 Controlled replacement in SCN

We isolate cross-orientation attention in the original SCN implementation on OC20-S2EF-200K by replacing only its orientation-wise S2Activation with EST. We keep $L = 3$, six interaction blocks, 256 hidden channels, 64 spherical channels, 20 neighbors, a 6 Å cutoff, and 60k training steps. EST increases parameters from 3.933M to 4.282M ($+8.9\%$). The SCN replacement reduces force MAE by 4.7%, improves force cosine by 9.7%, and reduces force-magnitude error by 6.7%, while energy MAE increases by 3.1%. Thus the controlled gain is concentrated on force modeling and cannot be attributed to a new spherical projection, which SCN already uses.

Table 8: Controlled SCN replacement on OC20-S2EF-200K. Only S2Activation is replaced by EST.

Model	Energy MAE ↓	Force MAE ↓	Force cosine ↑	Force magnitude error ↓
SCN-original	**0.54533**	0.05172	0.26900	0.08147
SCN–EST	0.56208	**0.04931**	**0.29511**	**0.07597**

C.2 Sampling uniformity

We evaluate equal-size grids at $S \in \{16, 36, 64, 100, 144, 196\}$ using nearest-neighbor angular-distance coefficient of variation (NN-CV), spherical-Voronoi area coefficient of variation (Area-CV), and spherical-cap discrepancy (Cap Error); lower is better. Table 9 averages each metric across the six resolutions. Repulsion-refined FL ranks first overall at every resolution; plain FL ranks second.

Table 9: Sphere-sampling uniformity averaged over six equal-S comparisons.

Sampling method	NN-CV ↓	Area-CV ↓	Cap Error ↓	First place
e3nn grid	0.2934	0.4348	0.1511	0/6
Fibonacci	0.1091	0.0666	0.0691	0/6
Fibonacci + repulsion	**0.0339**	**0.0337**	**0.0646**	**6/6**

C.3 Resolution, accuracy, equivariance, and memory

Table 10 uses a fixed QM9 configuration, batch size 4, and an RTX 4090. Accuracy and equivariance improve with S, but accuracy saturates rapidly while attention memory grows strongly. The Equiformer baseline uses 1,624 MiB. At the paper setting $S = 128$, EST improves α MAE from 0.0460 to 0.0408 with equivariance error 1.05×10^{-5} and 2,538 MiB peak memory; increasing to $S = 512$ changes MAE by only 0.0003 but raises memory to 12,424 MiB. In the large EquiformerV2 experiment, adding EST raises same-batch single-GPU memory from 66,570 to 71,350 MiB; the smaller MPtrj global batch size was partly due to this overhead and limited GPU availability. Practical guidance is therefore $S = \max\{64, (2L)^2\}$: the Nyquist term preserves the represented bandlimit and the lower bound avoids the large equivariance error observed at very coarse sampling. With abundant data, larger S usually yields marginal accuracy gains; with scarce data or stricter equivariance requirements, S can be increased subject to memory limits. Expressiveness is primarily adjusted through L and the EST branch, whereas S primarily controls discretization error and attention memory.

Table 10: Sampling-resolution trade-off on QM9.

Model	α MAE ↓	Eq. error ↓	Time/batch	Peak memory
EqV baseline	0.0460	8.482×10^{-7}	223 ms	1,624 MiB
EST, $S = 64$	0.0414	7.085×10^{-4}	246 ms	1,944 MiB
EST, $S = 128$	0.0408	1.050×10^{-5}	255 ms	2,538 MiB
EST, $S = 256$	0.0407	8.379×10^{-6}	268 ms	4,790 MiB
EST, $S = 512$	0.0405	3.501×10^{-6}	290 ms	12,424 MiB

C.4 Physical validity for dynamics and structure optimization

We evaluate EST in two representative interatomic-potential settings: MPtrj with an independently predicted force head, and MD17 with conservative forces computed as $\mathbf{F} = -\nabla E$. For NVE molecular dynamics, initial velocities are sampled from a 300 K Maxwell–Boltzmann distribution with center-of-mass translation and rotation removed. Energy drift is the linear slope of total energy after discarding the first 20% of each trajectory. On eight MD17 systems, EST reduces the mean absolute drift from 0.036 to 0.022 meV/ps/atom, and both models remain below 0.16 meV/ps/atom on every system.

Table 11 summarizes force-level and downstream physical checks. Under random rotations of approximately 100 MPtrj structures, both models retain cosine similarity above 0.9998 and force equivariance MAE below 0.6 meV/Å. All perturbed structures converge under L-BFGS relaxation, and finite-difference Hessians introduce neither imaginary frequencies nor incorrect mode counts. These results support using EST for molecular dynamics, relaxation, and vibrational workflows when paired with an appropriate force construction.

Table 11: Physical-validity tests. EqV1/2 denote the corresponding Equiformer baselines.

Setting	Metric	Baseline	Baseline + EST
MD17, gradient force	Mean |NVE drift| (meV/ps/atom; 8 systems)	0.036	**0.022**
MPtrj, direct force	Force equivariance MAE (meV/Å; $\sim$100 structures)	**0.359**	0.578
MPtrj, direct force	Force cosine similarity	**0.99988**	0.99987
MPtrj, direct force	Jacobian asymmetry $\|J_{\text{anti}}\|_F / \|J\|_F$	**0.258**	0.464
MPtrj relaxation	Converged / average $F_{\max}$ (eV/Å)	16/16 / 0.041	**16/16 / 0.038**
MD17 relaxation	Converged / average $F_{\max}$ (eV/Å)	8/8 / 0.014	**8/8 / 0.011**
MPtrj vibrations	Imaginary frequencies / correct mode count	0/16 / 16/16	0/16 / 16/16

The direct-force MPtrj setting is not conservative for either model. EST increases the Jacobian-asymmetry metric from 0.258 to 0.464, and per-system MPtrj NVE drift is mixed rather than consistently improved. We therefore do not claim that EST repairs non-conservative force heads. Instead, the tests show that force equivariance, relaxation convergence, and vibrational stability remain practically intact, while conservative gradient forces on MD17 retain excellent long-term energy behavior.

D Details of Experiments and Supplementary Experiments

D.1 Implementation Details of Baselines

Table 12: Hyper-parameters for the EST model setting on OC20 S2EF and OC20 IS2RE experiments.

Hyper-parameters	S2EF-ALL	IS2RE
Optimizer	AdamW	AdamW
Learning rate scheduling	Cosine learning rate with linear warmup	Cosine learning rate with linear warmup
Warmup epochs	0.01	2
Maximum learning rate	4×10^{-4}	2×10^{-4}
Batch size	256	32
Number of epochs	4	20
Weight decay	1×10^{-3}	1×10^{-3}
Dropout rate	0.1	0.2
Energy coefficient λ_E	$4, 2$	1
Force coefficient λ_F	100	-
Gradient clipping norm threshold	100	100
Model EMA decay	0.999	0.999
Cutoff radius (Å)	12	5
Maximum number of neighbors	20	500
Number of radial bases	600	128
Dimension of hidden scalar features in radial functions	128	64
Maximum degree L_{max}	6	1
Maximum order M_{max}	2	1
Number of Layers	8	6
Node embedding dimension	128	$(256, l = 0), (128, l = 1)$
Intermediate dimension during the Fourier transform	128	256
Intermediate dimension and the number of steerable FFN	$128, 6$	$[(768, l = 0), (384, l = 1)], 5$
Intermediate dimension and the number of spherical FFN	$512, 4$	$512, 5$
Number of spherical point samples	156	128

In the S2EF experiments, the results of baselines in Table 1 follow [21], where each model is trained in official configuration. Most of these configurations can be found in Fairchem repository, and we also follow its code framework to construct our OC20 experiments. In the IS2RE experiment, the results in Table 2 of baselines follow [12] and [19]. In the MPtrj experiments, the results are from [47, 3, 46]. In the QM9 experiments, the results in Table 4 of baselines follow [21] and [13].

Table 13: Hyper-parameters for Mptrj dataset.

Hyper-parameters	EquiformerV2	EST
Optimizer	AdamW	AdamW
Learning rate scheduling	Cosine learning rate with linear warmup	Cosine learning rate with linear warmup
Warmup epochs	0.1	0.1
Warmup factor	0.2	0.2
Maximum learning rate	4×10^{-4}	4×10^{-4}
Batch size	$512, 256$	256
Number of epochs	100	100
Weight decay	1×10^{-3}	1×10^{-3}
Model EMA decay	0.999	0.999
Number of radial bases	512 for Gaussian radial basis	
Cutoff radius (Å)	6	6
L_{max}	4	4
Number of layers	8	8
Node dimension	128	128
Energy loss coefficient	5	5
Force loss coefficient	20	20
Isotropic stress loss coefficient	5	5
Anisotropic stress loss coefficient	5	5
Intermediate dimension during the Fourier transform		512
Intermediate dimension and the number of steerable FFN		$128, 4$
Intermediate dimension and the number of spherical FFN		$128, 4$
Number of spherical point samples		72

Table 14: Hyper-parameters for QM9 dataset.

Hyper-parameters	Full-EST model (Figure 5)	EST+Equiformer
Optimizer	AdamW	AdamW
Learning rate scheduling	Cosine learning rate with linear warmup	Cosine learning rate with linear warmup
Warmup steps	5	5
Maximum learning rate	5×10^{-4}, 2×10^{-4}	5×10^{-4}, 1.5×10^{-4}
Batch size	$128, 64$	$128, 64$
Max training epochs	$350, 700$	$300, 600$
Weight decay	$5 \times 10^{-3}, 0$	$5 \times 10^{-3}, 0$
Dropout rate	$0.0, 0.2$	$0.0, 0.2$
Number of radial bases	128 for Gaussian radial basis 8 for radial bessel basis	
Cutoff radius (Å)	5	5
L_{max}	2	2
Number of layers	6	6
Node dimension	128	$(128, l = 0), (64, l = 1), (32, l = 2)$
Spherical harmonics embedding dimension	$(1, l = 0), (1, l = 1), (1, l = 2)$	$(1, l = 0), (1, l = 1), (1, l = 2)$
Intermediate dimension during the Fourier transform	128	128
Intermediate dimension and the number of steerable FFN	$5, 128$	$6, (768, l = 0), (384, l = 1)$
Intermediate dimension and the number of spherical FFN	$5, 512$	$4, 512$
Number of spherical point samples	$200, 128$	128

D.2 Implementation Details of EST Experiments

S2EF and IS2RE In our experiments on OC20, we adopt two hybrid models: S2EF combines the message module from EquiformerV2 with the update module of EST, while IS2RE combines the message module of Equiformer with the update module of EST. To ensure a fair comparison, all training configurations are aligned with those of the original EquiformerV2 and Equiformer. The hyperparameters specific to the EST architecture include the number of spherical sampling points, the number and dimension of experts in the steerable FFN, and the number of experts in the spherical FFN. Detailed configurations for all models are summarized in Table 12. The experiment on S2EF is conducted on 32 NVIDIA A100 GPUs and the experiment on IS2RE is conducted on 8 NVIDIA A100 GPUs. For comparison with SCN and eSCN, we report their test results due to the absence of publicly available validation results; previous work [21] indicates that the validation and test sets have a similar distribution and the test results are usually slightly better than the validation results.

MPtrj In our experiments on MPtrj, we follow most of the configuration from prior work [46]. Due to limited computational resources, we reduced the batch size from 512 to 256. To ensure a fair comparison, we also reproduced the base model (EquiformerV2) under the same configuration in subsequent experiments. The settings of other parameters are shown in Table 13.

QM9 In the QM9 experiments, we design three EST variants: 1) a fully EST architecture (Table 24) with both message and update blocks; 2) a hybrid model combining the message block of Equiformer and the update block of EST. We adopt similar configurations to base model as shown in Table 14 and Table 15. Note that Equiformer employs two different configurations for the following properties: $\alpha, \Delta\varepsilon, \varepsilon_{\text{HOMO}}, \varepsilon_{\text{LUMO}}, \mu, C_v, R^2$, ZPVE, and G, H, U, U_0. We follow the same strategy in Table 14. GotenNet also has a similar division, and we follow its configuration.

Table 15: Hyper-parameters for the EST+GotenNet models on QM9 experiments.

Hyper-parameters	EST (with GATA)	
properties	α, $\Delta\varepsilon$, ε_{HOMO}, ε_{LUMO}, μ, C_v, R^2	G, H, U, U_0, U, ZPVE
Loss function	MAE	MSE
Warmup steps	0	10000
Weight decay	0	0.01
Intermediate dimension and the number of steerable FFN	512, 1	512, 4
Intermediate dimension and the number of spherical FFN	512, 1	512, 4
Learning rate scheduling	linear warmup with reduce on plateau	
Optimizer	AdamW	
Maximum learning rate	1×10^{-4}	
Batch size	32	
Max training epochs	1000	
Dropout rate	0.1	
Number of RBFs	64	
Cutoff radius (Å)	5	
Lmax	2	
Number of layers	4	
Node dimension	256	
Edge dimension	256	
Number of attention heads	8	
Number of spherical point samples	64	

D.3 Supplementary GotenNet Experiments

We offer further experiments on GotenNet. From Table 16, two principal observations emerge: first, EST improvements are inconsistent, primarily benefiting non-energy properties (e.g., α, μ, ε_{HOMO}) while struggling with core energy-related tasks (G, H, U, U_0). Second, the gains from EST diminish as GotenNets depth increases. In GotenNet-S, EST improves performance on six tasks, but yields no gains in the deeper GotenNet-L. This suggests that as deeper architectures potentially accumulate equivariance errors. While increasing FL sampling number could restore equivariance, it would incur substantial computational costs.

Table 16: Results on QM9 dataset for various properties. † denotes using different data partitions.

Task Units	α $bohr^3$	$\Delta\varepsilon$ meV	ε_{HOMO} meV	ε_{LUMO} meV	μ D	C_v cal/(mol K)	G meV	H meV	R^2 $bohr^3$	U meV	U_0 meV	$ZPVE$ meV
GotenNet-B	.032	20.5	15.2	13.0	.0072	**.019**	5.19	3.44	.027	3.49	3.43	1.09
+ EST	.030	31.6	15.0	13.1	.0071	**.019**	5.13	3.60	.028	3.58	3.55	1.17
GotenNet-L	**.028**	**19.8**	**13.4**	**12.2**	**.0067**	**.019**	**4.98**	**3.30**	**.024**	**3.41**	**3.37**	**1.08**
+ EST	**.028**	30.8	13.7	12.8	.0070	**.019**	5.26	3.41	.027	3.50	3.53	1.18

D.4 Supplementary Experiments

Theoretical experiments on expressiveness To rigorously evaluate the representational capacity of steerable architectures, we construct a synthetic dataset based on randomized tensor product operations. Let $\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3 \in \mathbb{R}^3$ denote three independently sampled 3D vectors, corresponding to irreducible representations of degree $l = 1$. The target output $\mathbf{y}$ is generated via a randomized Fully Connected Tensor Product (FCTP) network. To ensure comprehensive coverage of spherical harmonic degrees, we cascade two FCTP operations: $\mathbf{x} = \mathrm{FCTP}(\mathbf{x}_1, \mathbf{x}_2)$ yields an intermediate steerable feature spanning degrees $l \in \{0, 1, 2\}$, which is subsequently combined with $\mathbf{x}_3$ via $\mathbf{y} = \mathrm{FCTP}(\mathbf{x}, \mathbf{x}_3)$ to produce the final output spanning $l \in \{0, 1, 3\}$. A total of $N = 1000$ samples are generated and partitioned into training and validation sets with a $9 : 1$ ratio.

We evaluate four distinct architectural variants:

1. **FCTP-based model**: Mirrors the dataset generation pipeline but employs expanded channel dimensions for enhanced capacity.
2. **EST**: A single-layer Equivariant Spherical Transformer.
3. **S2Activation**: An ablated EST variant with the cross-orientational attention mechanism removed.
4. **S2MLP**: A baseline that flattens all spherical sampling coordinates and channel dimensions into a monolithic vector, processed by a conventional multi-layer perceptron. This design discards explicit geometric structure.

All models are optimized to regress $\mathbf{y}$ using the Mean Absolute Error (MAE) as the objective function. Each configuration is trained for a sufficient number of epochs to ensure convergence, and the procedure is repeated across 10 independent random seeds to assess statistical stability. The resulting training and validation losses are reported in Table 17. Across all repetitions, the observed performance trends remain highly consistent: the FCTP and EST architectures achieve nearly identical losses, the S2MLP baseline attains marginally lower training loss but exhibits a validation gap, while the S2Activation variant demonstrates substantially elevated error.

Table 17: Training and validation MAE on the FCTP-generated dataset. Results are averaged over 10 independent runs.

Metric	**FCTP**	**EST**	**S2Activation**	**S2MLP**
Train loss	2.58×10^{-4}	2.60×10^{-4}	3.31×10^{-4}	2.50×10^{-4}
Valid loss	2.50×10^{-4}	2.52×10^{-4}	3.28×10^{-4}	2.68×10^{-4}

These findings substantiate our theoretical claim that the EST framework can effectively approximate arbitrary randomized FCTP operations. The S2MLP baseline, benefiting from an unconstrained architecture, attains the lowest training loss, reflecting its superior expressive capacity in unconstrained function approximation. However, its validation loss exceeds the training loss, indicating a tendency to overfit by memorizing dataset-specific patterns rather than learning an equivariant mapping. This behavior underscores a fundamental limitation of non-equivariant architectures in geometric learning tasks: they may exploit non-equivariant shortcuts to fit training data at the expense of generalization. Conversely, the S2Activation model consistently underperforms relative to both FCTP and EST across all experimental runs, evidencing its comparatively limited representational capacity.

The residual non-zero loss observed even in the FCTP-to-FCTP fitting task arises from the intrinsic stochasticity of the randomized tensor product generation process. For instance, the Clebsch–Gordan tensor product mapping two $l = 1$ irreducible representations to an $l = 1$ output is mathematically isomorphic to the vector cross product, which becomes degenerate when the input vectors are collinear or vanishing. To mitigate such multi-solution ambiguities and increase task complexity, we deliberately employ three distinct 3D vectors as inputs. Consequently, the generation process introduces irreducible stochastic components that cannot be perfectly reconstructed by a deterministic model, establishing a non-zero lower bound on the achievable loss.

To further probe the expressivity of the EST module, we conduct a complementary experiment using a dataset generated by a randomized EST layer. The corresponding performance metrics are summarized in Table 18. Notably, the unconstrained and semi-constrained architectures (EST, S2Activation, and S2MLP) achieve substantially lower losses, whereas the FCTP model exhibits a markedly higher error. This disparity strongly suggests that the EST module synthesizes representational features that transcend the functional space spanned by conventional tensor product operations.

Table 18: Training and validation MAE on the EST-generated dataset.

Metric	**FCTP**	**EST**	**S2Activation**	**S2MLP**
Train loss	7.06×10^{-3}	3.17×10^{-6}	8.04×10^{-5}	6.97×10^{-5}
Valid loss	7.06×10^{-3}	3.21×10^{-6}	7.96×10^{-5}	9.13×10^{-5}

All experimental scripts, data generation pipelines, and training routines are provided in `rebuttal/est_vs_tp.ipynb`. To facilitate independent verification of the architectural equivalence between EST and FCTP, trained model checkpoints can be serialized via `torch.save` for direct parameter inspection and cross-architecture comparison.

Table 19: The results of different expert configurations on OC20 IS2RE dataset.

Model	energy (ID) MAE	energy (OOD Ads) MAE	energy (OOD Cat) MAE	energy (OOD Both) MAE	average
Equiformer (1 + 0)	504	688	521	630	586
Equiformer + EST (5 + 5)	501	652	502	578	558
Equiformer + EST (0 + 1)	507	664	513	593	569
Equiformer + EST (1 + 1)	502	657	504	588	563

Table 20: Results of different expert configurations on QM9 dataset.

Model	α	ε_{HOMO}	U_0
Equiformer (1 + 0)	0.046	15	6.59
Equiformer + EST (4 + 6)	0.041	14	5.64
Equiformer + EST (0 + 1)	0.045	15	6.85
Equiformer + EST (1 + 1)	0.042	14	5.73
Equiformer + EST (10 + 0)	0.043	15	6.47
Equiformer + EST (8 + 0)	0.044	15	6.41
Equiformer + EST (4 + 0)	0.050	15	6.68
Equiformer + EST (2 + 0)	0.046	15	6.74

Different expert configurations To further understand the contributions of EST and MoE, we conducted experiments on OC20 IS2RE and QM9 using various expert configurations. $(x + y)$ denotes x steerable experts and y spherical experts. As shown in Table 19 and Table 20, we have several findings: 1.The update block using only spherical attention still achieves improvements. The minimal MoE configuration (1 steerable expert and 1 spherical attention expert) achieves results close to the multi-expert configuration.

Building Blocks We conducted ablation studies to validate the effectiveness of individual building blocks within EST. Specically, in Table 21, removing the steerable FFN means all experts are replaced by the spherical FFN, removing the spherical FFN means all experts are replaced by the steerable FFN and removing FL Sampling refers to using the Fourier transform from e3nn instead. We used the prediction of the α property on QM9 as the core task for evaluation. As shown in Table 21, all components positively contribute to the overall performance. We draw several conclusions:

Table 21: Ablation studies for modules in EST.

Building blocks in EST					α MAE
LayerNorm	SA	Steerable FFN	Spherical FFN	FL Sampling	$bohr^3$
-	✓	✓	✓	✓	0.046
-	-	✓	✓	✓	0.044
✓	✓	-	✓	✓	0.045
-	-	✓	-	✓	0.043
✓	✓	✓	✓	-	0.053
✓	✓	✓	✓	✓	**0.041**

Table 22: Results on Mptrj dataset. EquiformerV2 is replicated with the configuration in Table 13.

model	energy (meV/atom) ↓	forces (meV/Å) ↓	stress (meV/Å^3)↓	forces cos ↑
EquiformerV2-S (Repetition)	11.1	29.48	1.39	0.74
EquiformerV2-S + EST	10.80	28.85	1.36	0.74

1. The **SA module with layer normalization** effectively improves overall performance.
2. **Mixing steerable and spherical experts** helps strike a balance between equivariance and expressive power, leading to better generalization performance.
3. **FL Sampling is crucial for EST**. Disrupting equivariance without it significantly harms the results on QM9.

Replication of EquiformerV2 on the MPtrj Benchmark In our MPtrj experiments, we reproduced EquiformerV2 using the original configurations. As shown in Table 22, our replicated EquiformerV2 slightly outperforms the originally reported results. This improvement is likely attributable to a more conservative learning rate, which prevents the model from converging to suboptimal local minima. Furthermore, under these optimized baseline conditions, EST consistently yields significant performance gains, further validating its effectiveness in modeling complex molecular systems.

E Hybrid Branch and Architecture Details

The term "hybrid experts" denotes parallel function branches, not a large-language-model-style sparse MoE. Given an aggregated steerable message $\mathbf{m}$, the base model applies an exactly equivariant steerable FFN, while EST applies spherical attention and a spatial FFN after the Fourier transform. Invariant gate weights combine the branch outputs with the residual:

$$\tilde{\mathbf{m}} = \mathbf{m} + \sum_{i=1}^{N_{\text{steer}}} g_i \, \text{FFN}_i^{\text{steer}}(\mathbf{m}) + \sum_{j=1}^{N_{\text{spatial}}} g_j \, \mathcal{F}^{-1}\Big(\text{FFN}_j^{\text{spatial}}(\text{EST}(\mathcal{F}(\mathbf{m})))\Big), \tag{43}$$

where the gates are computed from the invariant $l = 0$ channels. Our primary models use dense, small mixtures rather than top-k routing. The purpose is to retain an exact-equivariant path alongside the expressive, finite-sampling EST path; it is not to scale parameter count through conditional computation. The ablations in Appendix D.4 show that adding same-type branches yields limited gains, whereas a minimal mixed $1 + 1$ configuration is consistently useful.

E.1 Integration into equivariant architectures

EST can be inserted into either the edge-message or node-update block (Figure 5). We use it mainly in update blocks because attention cost then scales with nodes rather than edges. A full-EST message block provides little additional QM9 benefit while increasing edge-dependent computation; Table 24 records this comparison.

Table 23: Efficiency benchmark of the proposed EST compared to DTP and FCTP baselines. Latency is reported in milliseconds (ms) under GPU acceleration across varying maximum degrees (L) and channels (C). Each data point represents the mean execution time across 50 independent trials to ensure statistical reliability.

L	Channels	EST Latency (ms)	DTP Latency (ms)	FCTP Latency (ms)	Speedup vs DTP	Speedup vs FCTP
1	32	0.598	0.517	1.051	0.865	1.757
1	64	0.540	0.512	0.975	0.948	1.805
1	128	0.555	0.565	0.953	1.019	1.718
1	256	0.528	0.581	0.972	1.100	1.839
1	512	0.522	0.538	2.654	1.031	5.081
2	32	0.534	1.327	2.171	2.488	4.069
2	64	0.528	1.291	2.125	2.445	4.023
2	128	0.525	1.292	2.125	2.459	4.044
2	256	0.530	1.298	2.158	2.450	4.074
2	512	0.531	1.387	6.197	2.612	11.669
3	32	0.569	2.817	4.714	4.948	8.281
3	64	0.542	2.920	4.774	5.389	8.811
3	128	0.537	2.844	4.758	5.294	8.858
3	256	0.508	2.822	4.722	5.554	9.293
3	512	0.534	2.833	13.658	5.302	25.562
4	32	0.540	5.299	10.826	9.816	20.054
4	64	0.528	5.328	10.985	10.085	20.793
4	128	0.537	5.334	10.895	9.938	20.300
4	256	0.514	5.392	11.054	10.496	21.516
4	512	0.545	5.327	25.955	9.781	47.657
5	32	0.550	9.216	21.509	16.761	39.119
5	64	0.541	9.197	21.508	16.991	39.734
5	128	0.511	9.170	21.671	17.955	42.431
5	256	0.516	9.169	21.932	17.780	42.527
5	512	0.547	9.167	47.309	16.756	86.480
6	32	0.544	14.888	31.668	27.346	58.167
6	64	0.543	14.787	31.743	27.208	58.408
6	128	0.517	14.809	31.834	28.644	61.574
6	256	0.524	14.898	32.796	28.430	62.586
6	512	0.552	14.745	73.367	26.701	**132.857**

Table 24: QM9 comparison of update-only and full EST variants.

Task	α	$\Delta\varepsilon$	ε_{HOMO}	ε_{LUMO}	μ	C_v	G	H	R^2	U	U_0	ZPVE
Equiformer + EST update	.041	29	14	13	.011	.021	7.18	6.17	.227	6.35	6.32	1.27
Full EST	.042	28	13.4	12.4	.0113	.022	7.03	5.94	.298	5.92	5.64	1.31

NeurIPS Paper Checklist

The checklist is designed to encourage best practices for responsible machine learning research, addressing issues of reproducibility, transparency, research ethics, and societal impact. Do not remove the checklist: **The papers not including the checklist will be desk rejected.** The checklist should follow the references and follow the (optional) supplemental material. The checklist does NOT count towards the page limit.

Please read the checklist guidelines carefully for information on how to answer these questions. For each question in the checklist:

- You should answer [Yes], [No], or [N/A].
- [N/A] means either that the question is Not Applicable for that particular paper or the relevant information is Not Available.
- Please provide a short (1–2 sentence) justification right after your answer (even for [N/A]).

The checklist answers are an integral part of your paper submission. They are visible to the reviewers, area chairs, senior area chairs, and ethics reviewers. You will also be asked to include it (after eventual revisions) with the final version of your paper, and its final version will be published with the paper.

The reviewers of your paper will be asked to use the checklist as one of the factors in their evaluation. While [Yes] is generally preferable to [No], it is perfectly acceptable to answer [No] provided a proper justification is given (e.g., error bars are not reported because it would be too computationally

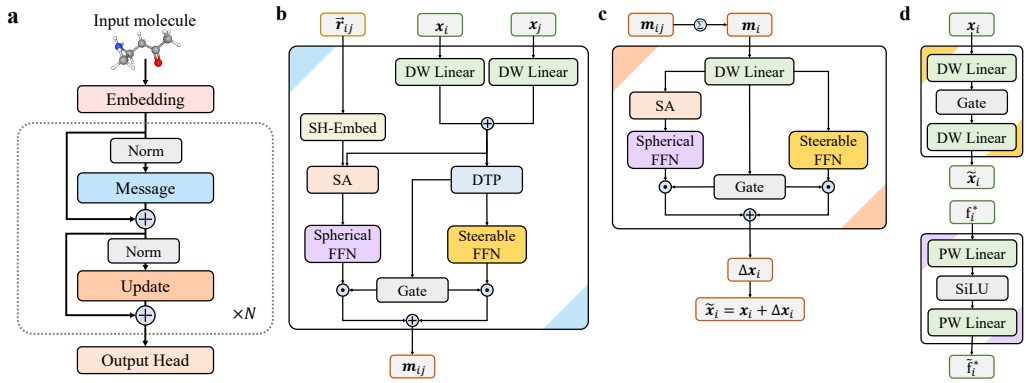


Figure 5: **The architecture and building blocks of EST.** SH and DTP denote spherical harmonic embedding and depth-wise tensor product [12]. Fourier and inverse Fourier transforms are omitted for clarity. (a) Overall architecture. (b) Message block. (c) Update block. (d) Parallel steerable and spatial branches.

expensive" or "we were unable to find the license for the dataset we used"). In general, answering [No] or [N/A] is not grounds for rejection. While the questions are phrased in a binary way, we acknowledge that the true answer is often more nuanced, so please just use your best judgment and write a justification to elaborate. All supporting evidence can appear either in the main paper or the supplemental material, provided in appendix. If you answer [Yes] to a question, in the justification please point to the section(s) where related material for the question can be found.

IMPORTANT, please:

- **Delete this instruction block, but keep the section heading "NeurIPS Paper Checklist"**,
- **Keep the checklist subsection headings, questions/answers and guidelines below.**
- **Do not modify the questions and only use the provided macros for your answers**.

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: Yes. Our main contributions are also detailed in Section 3. Also see Section 4 and Appendix B for more theoretical and experimental evidence.

   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: We discuss limitations in Section 5.

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 assumptions and proof of Theorem 3.1, including the required rotation-closed domain, are provided in Appendix B.1. Claim 3.2 is explicitly labeled as unproven intuition and is supported by an approximation argument and controlled experiments in Appendix B.2, rather than presented as a theorem.

   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: All hyper-parameters for the experiments in this paper are provided in Section 4, Appendix D.1 and D.2.

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: [No]

   Justification: The paper uses existing open-source datasets, and the authors plan to release the EST architecture, training scripts, configurations, environment instructions, and commands. However, a public repository URL and release license were not available for verification when finalizing this checklist. This answer should be changed to Yes once the public release is linked and checked.

   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, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

   Answer: [Yes]

   Justification: The training and test details can be found in Appendix D.2, including data splits, hyperparameters, how they were chosen, type of optimizer, etc.

   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: The experiments supporting the main motivation of the paper (Table 4) include error bars. The experiments on OC20 do not include error bars due to high computational costs. However, we analyze the potential error bar situation in Appendix D.3. The experiments on QM9 have been repeated multiple times, and the results consistently converge within a very small range, which we also discuss in Appendix D.3.

   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: The required computing resources are provided in the Appendix D.2 and the anonymized code repository.

   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: We have reviewed and adhered to 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: [N/A]

    Justification: Due to the low-level nature of our method, we do not see it directly facilitating any negative social impacts.

    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: [N/A]

    Justification: The model proposed in this paper does not pose risks of high-risk 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: [No]

    Justification: The dataset and baseline sources are cited, but the exact license names for every external asset and the license of the planned code release are not currently stated in the manuscript. These should be verified in the public repository before submission.

    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: [No]

    Justification: The planned public release introduces the EST implementation, model configurations, and training scripts. The repository README, exact reproduction commands, and license should be verified before changing this answer to Yes.

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: [N/A]

    Justification: The paper does not involve crowdsourcing nor research with 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: [N/A]

    Justification: The paper does not involve crowdsourcing nor research with human subjects.

    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: [N/A]

    Justification: This paper does not involve LLMs as any important, original, or non-standard components.

    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.