WIDER GROUPS, FEWER WEIGHTS: EFFICIENT PSEUDO-ORTHOGONAL EQUIVARIANCE VIA GEOMETRIC ALGEBRAS

Anonymous authors
Paper under double-blind review

ABSTRACT

We introduce a new parameter-efficient method for constructing neural networks that are equivariant with respect to pseudo-orthogonal (or orthogonal) groups, based on geometric (or Clifford) algebras. The approach is based on finding and considering larger groups than the Clifford groups in geometric algebras, which perform the same action as the pseudo-orthogonal (or orthogonal) groups. By constructing networks equivariant to these larger groups, we automatically guarantee their equivariance to the corresponding pseudo-orthogonal (or orthogonal) group. This approach induces an expressiveness-efficiency trade-off: a larger equivariance group leads to stricter constraints on the equivariant layers and thus to fewer required optimizable parameters (weights). We propose and study specific groups for this framework, derive new equivariant mappings, and present several model configurations with varying levels of expressiveness and efficiency. Our models demonstrate strong performance on benchmark tasks including convex hull volume estimation, N-body dynamics prediction, high-energy jet tagging, and prediction of atom positions on the MD17 dataset.

1 INTRODUCTION

Equivariant neural networks are designed to respect the symmetries of a given group, which leads to better generalization by incorporating domain knowledge directly into the architecture (Cohen & Welling, 2016; Bronstein et al., 2021). For example, a neural network is orthogonal equivariant if we can either firstly apply any orthogonal transformation (for example, rotation or reflection) to our data and then apply the model or firstly apply the model and then apply the same transformation and get the same result. Such models have proven effective across diverse domains, especially in computer vision and natural sciences, for example, molecular modeling and generation (Satorras et al., 2021; Liao & Smidt, 2023; Duval et al., 2023; Batzner et al., 2022; Brandstetter et al., 2022), image and point cloud analyses and processing (Worrall et al., 2017; Wang & Jörnsten, 2023; Fuchs et al., 2020), protein modeling (Pepe et al., 2024), particle physics (Bogatskiy et al., 2020; Ruhe et al., 2023a; Finzi et al., 2021), fluid dynamics and electrodynamics (Zhdanov et al., 2024), robotics and motion capturing (Brehmer et al., 2023; Liu et al., 2024a), and other areas.

A common strategy for building these networks is to construct each layer from equivariant functions. One of the recent state-of-the-art approaches of how to construct such functions is in employing the mathematical framework of geometric algebras (GAs), also known as Clifford algebras, instead of matrix formalism and building all the functions on layers as functions of GA elements. GAs generalize complex numbers and quaternions to n-dimensional spaces.[1] They gained prominence in physics through their role in formulating the Dirac equation (Dirac, 1928) and are now successfully applied in physics (Hestenes & Sobczyk, 1984; Doran & Lasenby, 2003), engineering (Dorst et al., 2002; Bayro-Corrochano & Scheuermann, 2010; Perwass, 2009), computer science (Dorst et al., 2007; Hildenbrand, 2018), in particular, computer vision (Bayro-Corrochano, 2019), neural networks (Brandstetter et al., 2022; Ruhe et al., 2023b; Bayro-Corrochano & Buchholz, 1997; Pearson & Bisset, 1994; Buchholz & Sommer, 2008), etc. Their utility in applications stems from a fundamental relation to orthogonal (or, more generally, pseudo-orthogonal) groups. Precisely, there is a well-known theorem (see, for example, Benn & Tucker (1987)) that for any (pseudo-)orthogonal matrix, there exist certain elements of Clifford group in GA that perform the same transformation. Furthermore, GAs offer a unified language for geometric entities (for example, vectors, oriented

[1] We refer the reader to Appendix D for a discussion of the advantages of the GA approach over non-GA methods for constructing equivariant networks.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

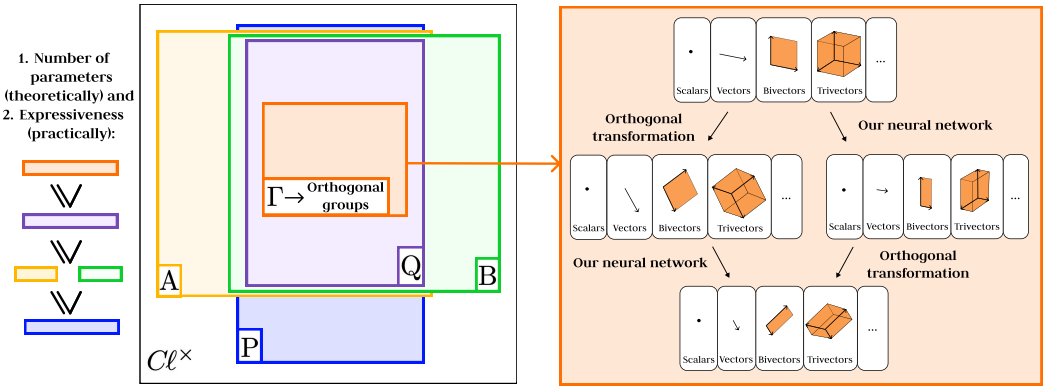


Figure 1: The core idea of our approach. Instead of enforcing equivariance of a neural network directly to an orthogonal group, we enforce it to a larger group (for example, Q, A, B, and P) that contains as a subgroup the Clifford group Γ, which performs the same action as the orthogonal group. This induces a strict symmetry constraint, drastically reducing the number of optimizable parameters while guaranteeing the desired equivariance.

planes, and volumes) and operations between them, making them exceptionally well-suited for geometrically meaningful deep learning (Dorst & De Keninck, 2022).

The wonderful work (Ruhe et al., 2023a) paved a new wave of application GAs to equivariant neural networks. They propose Clifford Group Equivariant Neural Networks (CGENN), which are equivariant with respect to Clifford groups in GAs and the corresponding (pseudo-)orthogonal groups. This work has a lot of applications and developments: GA transformer with applications (Brehmer et al., 2023), equivariant message passing on simplicial complexes (Liu et al., 2024a), convolutional neural networks with GA applied in physics (Zhdanov et al., 2024), metric learning for CGENN (Ali et al., 2024), applications in protein structure estimation (Pepe et al., 2024). The recent paper Filimoshina & Shirokov (2025) proposes to modify CGENN layers so that they have smaller number of optimizable parameters. Surprisingly, their model GLGENN (Generalized Lipschitz Group Equivariant Neural Networks) outperforms CGENN in lots of settings but with significantly smaller number of weights, especially in the cases of small training datasets, where overfitting is a concern.

In this paper, we generalize this insight into a unified framework. Our idea is to enforce model equivariance with respect to the larger groups containing orthogonal groups as subgroups; this automatically ensures orthogonal equivariance. For this purpose, we suggest and consider several groups in GAs that contain the Clifford group as a subgroup; some of their analogues are considered in Shirokov (2021); Filimoshina & Shirokov (2024). Our approach guarantees the desired equivariance while introducing a tunable trade-off between parameter efficiency and expressiveness (see Figure 1). A key advantage of our approach is the ability to select and compose a configuration—from highly efficient large-group models to highly expressive small-group models—based on the complexity of the task and the size of the available train dataset.

Our main contributions:

- **Generalized equivariance approach:** We introduce a principled approach to achieving pseudo-orthogonal equivariance by enforcing equivariance with respect to novel, larger groups defined within geometric algebras (GAs). We introduce and study these groups and prove equivariance of mappings with respect to them (see Table 1).
- **Configurations of GA-based equivariant models:** We construct a general framework for GA-based networks where the choice of the equivariance group provides a trade-off between parameter efficiency and expressiveness. The larger is the group – the stricter are symmetry constraints – the smaller number of optimizable parameters is required. This yields a hierarchy of models, from highly efficient (large-group) to highly expressive (small-group), all guaranteeing pseudo-orthogonal equivariance.
- **Expressiveness-efficiency trade-off evaluation:** We design, implement, and test neural networks architectures equivariant with respect to the new groups and pseudo-orthogonal groups as well. We evaluate the trade-off in four benchmark experiments and provide guidelines for configuration selection. Code is available in a repository in the supplementary materials.

This paper is organized as follows. Section 2 reviews the mathematical background. In Section 3, we introduce novel classes of larger groups within GAs and prove equivariance of key mappings with respect to these groups and the corresponding orthogonal groups. We design layers based on these mappings in Section 4 and discuss how they can be composed into configurations equivariant with respect to different groups in Section 5. In Section 6, we evaluate new configurations in convex hull volume estimation, benchmarking regression task, N-body dynamics simulation, high-energy jets categorization, and atom motion prediction.

2 Mathematical Background

2.1 Orthogonal groups equivariance

Suppose V is an n-dimensional vector space equipped with a non-degenerate or degenerate quadratic form $\mathfrak{q}$ of signature (p, q, r): it has p positive, q negative, and r zero eigenvalues, where $p+q+r = n \geq 1$. The vector space V can be real $\mathbb{R}^{p,q,r}$ or complex $\mathbb{C}^{p+q,0,r}$. We use $\mathbb{F}$ to denote the field of real numbers $\mathbb{R}$ in the first case and complex numbers $\mathbb{C}$ in the second case. In this work, we are interested in the degenerate and non-degenerate *pseudo-orthogonal group* (in the real case $V = \mathbb{R}^{p,q,r}$) or *complex orthogonal group* (in the complex case $V = \mathbb{C}^{p+q,0,r}$):

$$\mathrm{O}(V, \mathfrak{q}) \quad := \quad \{\Phi : V \to V : \quad \Phi \text{ is linear, invertible}, \quad \mathfrak{q}(\Phi(v)) = \mathfrak{q}(v), \quad \forall v \in V\}, \tag{1}$$

which is the Lie group of all linear transformations of V that leave invariant $\mathfrak{q}$. We also consider its subgroup called *special pseudo-orthogonal group* (in the real case) or *complex special orthogonal group* (in the complex case): $\mathrm{SO}(V, \mathfrak{q}) := \{\Phi \in \mathrm{O}(V, \mathfrak{q}) : \det(\Phi) = 1\}$. When considering both cases, we refer to $\mathrm{O}(V, \mathfrak{q})$ and $\mathrm{SO}(V, \mathfrak{q})$ as the *orthogonal* and *special orthogonal groups*.

Suppose G is a group and $\circ$ is its action on some sets X and Y. A mapping (or a neural network) $L : X \to Y$ is called *equivariant* with respect to the group G (or G-*equivariant*) iff it commutes with the group's action: $L(g \circ x) = g \circ L(x)$ for any $g \in G$ and $x \in X$. This work focuses on $G = \mathrm{O}(V, \mathfrak{q})$-equivariance. A neural network is $\mathrm{O}(V, \mathfrak{q})$-equivariant iff applying any orthogonal transformation $g \in \mathrm{O}(V, \mathfrak{q})$ to the input and then the network yields the same result as applying the network first and then g to the output.

2.2 Geometric (Clifford) algebras

Let us consider a *geometric algebra* (*GA*, also known as *Clifford algebra*) $C\ell_{p,q,r} = C\ell(V)$, where the vector space V and its quadratic form $\mathfrak{q}$ are defined as above (see Appendix E.1 and Lounesto (1997); Helmstetter & Micali (2008); Porteous (1995) for a detailed construction). We denote the matrix of $\mathfrak{q}$ in a normal form as η, which is a diagonal matrix with p times 1, q times -1, and r times 0 on the diagonal in the real case $C\ell(\mathbb{R}^{p,q,r})$, and $p+q$ times 1 and r times 0 on the diagonal in the complex case $C\ell(\mathbb{C}^{p+q,0,r})$. For brevity, we use the notation $C\ell$ to refer to the GA in both cases. Let $e_1, \ldots, e_n$ be an orthogonal basis of V. These basis vectors serve as *generators* of $C\ell$ and, by definition, satisfy the following conditions: $e_a e_b + e_b e_a = 2\eta_{ab} e$, $a, b = 1, \ldots, n$, where $e \equiv 1$ is the identity element of $C\ell$ and η_{ab} denotes the (a, b)-th entry of η. This relation implies that distinct generators anticommute ($e_a e_b = -e_b e_a$ for $a \neq b$) and square to either $+e$, $-e$, or 0 (precisely, in the real case, $(e_a)^2 = +e$ for $a = 1, \ldots, p$; $(e_a)^2 = -e$ for $a = p+1, \ldots, p+q$; and $(e_a)^2 = 0$ for $a = p+q+1, \ldots, n$). $C\ell$ is a 2^n-dimensional vector space. Its basis consists of all distinct ordered products of the generators, denoted by $e, e_1, \ldots, e_n$, $e_{12} := e_1 e_2$, $\ldots$, $e_{123} := e_1 e_2 e_3$, $\ldots, e_{1\ldots n} := e_1 \cdots e_n$, where for any ordered multi-index $a_1 < \cdots < a_k$, we define $e_{a_1 \ldots a_k} := e_{a_1} \cdots e_{a_k}$. An arbitrary element of $C\ell$, called *multivector*, is a linear combination of the basis elements: $U = ue + \sum_{a=1}^{n} u_a e_a + \sum_{a<b} u_{ab} e_{ab} + \sum_{a<b<c} u_{abc} e_{abc} + \cdots + u_{1\ldots n} e_{1\ldots n}$, where $u, \ldots, u_{1\ldots n} \in \mathbb{F}$ are scalars.

The name GA stems from the rich geometric interpretation of its elements (Lounesto, 1997; Dorst & De Keninck, 2022). GA is naturally decomposed into *grade subspaces* $C\ell^k$, which elements are linear combinations of basis elements $e_{a_1 \ldots a_k}$ with ordered multi-indices of length k. The grade-0 subspace $C\ell^0 \equiv \mathbb{F}$ is a one-dimensional space of *scalars*. The grade-1 subspace $C\ell^1 \equiv V$ consists of *vectors*. The grade-2 and grade-3 subspaces, $C\ell^2$ and $C\ell^3$, contain *bivectors* and *trivectors*, which represent oriented planes and volumes, respectively. In general, elements of $C\ell^k$, $k = 0, \ldots, n$, are called k-*vectors*. In Figure 2, we provide a visual representation of these objects.

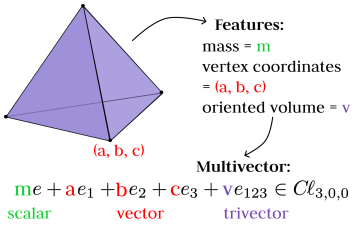


Figure 2: Standard geometric data can easily be embedded into GA.

Let us provide several important examples of GAs. For working with Euclidean spaces, $C\ell_n := C\ell_{n,0,0}$ are considered; particularly, $C\ell_3$ provides a foundational model for 3D geometry. For spaces with non-degenerate metrics of indefinite signature, the *non-degenerate* algebras $C\ell_{p,q} := C\ell_{p,q,0}$ are essential, particularly $C\ell_{1,3}$ models Minkowski spacetime in physics. Another example is the *degenerate* algebra $C\ell_{0,0,r}$, which is also called *exterior (Grassmann) algebra* and denoted by $\Lambda = \Lambda_r$. We can consider Λ_r as a subalgebra in any $C\ell_{p,q,r}$.

Consider *even* and *odd* subspaces: $C\ell^{(0)} := \bigoplus_{k=0 \bmod 2} C\ell^k$, $C\ell^{(1)} = \bigoplus_{k=1 \bmod 2} C\ell^k$. Consider the four *subspaces of quaternion types*: $C\ell^{\overline{m}} := \bigoplus_{k=m \bmod 4} C\ell^k$, $m = 0, 1, 2, 3$. We denote $C\ell^{\overline{ml}} := C\ell^{\overline{m}} \oplus C\ell^{\overline{l}}$ for $m < l$. Any $U \in C\ell$ can be projected onto these subspaces. We denote these projections using angle brackets $\langle\cdot\rangle$ with corresponding indices[2]:

$$U = \sum_{k=0}^{n} \langle U\rangle_k = \sum_{m=0}^{3} \langle U\rangle_{\overline{m}} = \langle U\rangle_{(0)} + \langle U\rangle_{(1)} = \langle U\rangle_{\overline{01}} + \langle U\rangle_{\overline{23}} = \langle U\rangle_{\overline{12}} + \langle U\rangle_{\overline{03}}, \tag{2}$$

where $\langle U\rangle_k \in C\ell^k$, $\langle U\rangle_{\overline{m}} \in C\ell^{\overline{m}}$, $\langle U\rangle_{(l)} \in C\ell^{(l)}$, $\langle U\rangle_{\overline{ml}} \in C\ell^{\overline{ml}}$. These decompositions onto direct sums with step sizes 2 and 4 are motivated by the fundamental[3] operations of *grade involution* $\widehat{\ }$, *reversion* $\widetilde{\ }$, and their composition $\widehat{\widetilde{\ }}$ called *Clifford conjugation* in GAs. They are defined as

$$\widehat{U} := \langle U\rangle_{(0)} - \langle U\rangle_{(1)}, \quad \widetilde{U} := \langle U\rangle_{\overline{01}} - \langle U\rangle_{\overline{23}}, \quad \widehat{\widetilde{U}} := \langle U\rangle_{\overline{03}} - \langle U\rangle_{\overline{12}}, \qquad \forall U \in C\ell. \tag{3}$$

To facilitate orientation, we present a roadmap of main definitions in Figure 4.

3 Theoretical results

3.1 New way to achieve $\mathrm{O}(V, \mathfrak{q})$-equivariance

In this subsection, we introduce a principled approach to achieving $\mathrm{O}(V, \mathfrak{q})$-equivariance by enforcing equivariance with respect to novel, larger groups in GA. The detailed proofs of all statements are provided in Appendix E.5. Let $\mathrm{K}^\times$ denote the set of all invertible elements within any set K. Our approach builds upon the well-known Clifford group Γ (see, for example, Crumeyrolle (1990); Benn & Tucker (1987); Lounesto (1997)), defined as the group of invertible elements that preserve the grade-1 subspace $C\ell^1$:

$$\Gamma := \{T \in C\ell^\times : \quad \mathrm{ad}_T(C\ell^1) := TC\ell^1T^{-1} = C\ell^1\}, \tag{4}$$

where ad is the *adjoint representation* acting on the group of all invertible elements $\mathrm{ad} : C\ell^\times \to \mathrm{Aut}(C\ell)$ as $T \mapsto \mathrm{ad}_T : C\ell \to C\ell$, where $\mathrm{ad}_T(U) := TUT^{-1}$ acts as a similarity transformation[4] for any $U \in C\ell$. It satisfies linearity, scalar invariance, and multiplicativity (see Lemma E.2 for details). The Clifford group Γ, often considered alongside with the *Lipschitz group* in the context of *spin groups* (see, for example, Ablamowicz (1986); Porteous (1995); Benn & Tucker (1987); Lounesto (1997)), has significant applications in physics (see Appendix E.4 for details). These groups are of great interest because their elements perform the same action as orthogonal matrices.

We denote by $\Lambda^1 := C\ell^1_{0,0,r}$ the *radical subspace* of $C\ell$, that is, the span of generators that square to zero, and by $\mathrm{O}_{\Lambda^1}(V, \mathfrak{q}) := \{\Phi \in \mathrm{O}(V, \mathfrak{q}) : \ \Phi|_{\Lambda^1} = \mathrm{id}_{\Lambda^1}\}$ the *restricted orthogonal group*[5], with its special analogue $\mathrm{SO}_{\Lambda^1}(V, \mathfrak{q})$ defined similarly. We also denote by Z the center of $C\ell$:

$$\mathrm{Z} = \Lambda^{(0)} \oplus C\ell^n \ \text{ if } n \text{ is odd}; \quad \mathrm{Z} = \Lambda^{(0)} \ \text{ if } n \text{ is even}; \tag{5}$$

where $\Lambda^{(0)} = C\ell^{(0)} \cap \Lambda$ is the even subspace of the Grassmann subalgebra.

Theorem 3.1 *Benn & Tucker (1987); Marchuk & Shirokov (2020) The adjoint representation* ad *induces a group isomorphism*

$$\mathrm{ad} : \Gamma \,/\mathrm{Z}^\times \ \cong \ \begin{cases} \mathrm{O}_{\Lambda^1}(V, \mathfrak{q}), & \textit{if } n \textit{ is even}, \\ \mathrm{SO}_{\Lambda^1}(V, \mathfrak{q}), & \textit{if } n \textit{ is odd}. \end{cases} \tag{6}$$

[2] Note that these five expressions are equivalent ways of representing the same multivector U.

[3] See Appendix E.2 for the details.

[4] Here T is an element of $C\ell$, not a matrix; T^{-1} is an inverse with respect to the geometric product.

[5] In the non-degenerate case $r = 0$ the subspace Λ^1 is trivial, so $\mathrm{O}_{\Lambda^1}(V, \mathfrak{q}) = \mathrm{O}(V, \mathfrak{q})$ and $\mathrm{SO}_{\Lambda^1}(V, \mathfrak{q}) = \mathrm{SO}(V, \mathfrak{q})$. All our experiments use $r = 0$; thus the models are strictly $\mathrm{O}(V, \mathfrak{q})$-equivariant. In the degenerate case $r > 0$, we prove equivariance with respect to $\mathrm{O}_{\Lambda^1}(V, \mathfrak{q})$, which is the most geometrically meaningful subgroup of $\mathrm{O}(V, \mathfrak{q})$.

Equivalently, for any orthogonal matrix $P = (p_{ab})$ in $\mathrm{O}_{\Lambda^1}(V, \mathfrak{q})$ (if n is even) or in $\mathrm{SO}_{\Lambda^1}(V, \mathfrak{q})$ (if n is odd), there exists $T \in \Gamma$ such that $Te_aT^{-1} = \sum_{b=1}^{n} p_{ab}e_b$ for all $a = 1, \dots, n$. Moreover, this T is unique up to multiplication by an element of $\mathrm{Z}^\times$.

This fundamental relation allows us to move from the matrix formalism to the GA formalism. By identifying mappings that are Γ-equivariant, we automatically obtain mappings that are equivariant with respect to $\mathrm{O}_{\Lambda^1}(V, \mathfrak{q})$ (or $\mathrm{SO}_{\Lambda^1}(V, \mathfrak{q})$). Crucially, we can extend this principle to leverage even larger groups. The following theorem shows that equivariance is preserved under subgroup restrictions:

Theorem 3.2 *If a mapping $L : X \to Y$ is equivariant with respect to a group H that contains the other group G as subgroup ($\mathrm{G} \subseteq \mathrm{H}$), then it is equivariant with respect to G as well:*

$$L(h \circ x) = h \circ L(x),\ \forall x \in X,\ \forall h \in \mathrm{H} \quad \Rightarrow \quad L(g \circ x) = g \circ L(x),\ \forall x \in X,\ \forall g \in \mathrm{G}. \tag{7}$$

Proof. Suppose $L(h \circ x) = h \circ L(x)$ for any $x \in X$ and $h \in \mathrm{H}$; then this equality holds, in particular, for any $g \in \mathrm{G} \subseteq \mathrm{H}$, and we get the statement. □

Consequently, if a mapping is equivariant with respect to a group H that contains the Clifford group Γ as a subgroup ($\Gamma \subseteq \mathrm{H}$), then it is equivariant with respect to Γ as well. A natural strategy for defining a group that contains the Clifford group is to generalize its defining property. Specifically, we can replace the grade-1 subspace $C\ell^1$ in the definition (4) with a larger space, which can be constructed as a direct sum of selected grade subspaces $C\ell^k$ that includes $C\ell^1$, and consider the group preserving this new space under the adjoint action. Namely, we suggest considering the groups preserving the following fundamental subspaces of GA: the odd subspace $C\ell^{(1)}$, the subspaces $C\ell^{\overline{01}}$ and $C\ell^{\overline{12}}$ determined by the reversion and Clifford conjugation, and their intersection $C\ell^{\overline{1}}$ (all contain $C\ell^1$ as a subspace):

$$\begin{aligned} \mathrm{P} &:= \{T \in C\ell^\times : \quad TC\ell^{(1)}T^{-1} = C\ell^{(1)}\}, \qquad \mathrm{Q} := \{T \in C\ell^\times : \quad TC\ell^{\overline{1}}T^{-1} = C\ell^{\overline{1}}\}, \\ \mathrm{A} &:= \{T \in C\ell^\times : \quad TC\ell^{\overline{01}}T^{-1} = C\ell^{\overline{01}}\}, \qquad \mathrm{B} := \{T \in C\ell^\times : \quad TC\ell^{\overline{12}}T^{-1} = C\ell^{\overline{12}}\}. \end{aligned} \tag{8}$$

The groups P, Q, A, and B [6] can be viewed as natural generalizations of the Clifford group Γ (4).

Lemma 3.3 *The groups P, Q, A, and B contain the Clifford group as a subgroup: $\Gamma \subseteq \mathrm{P}, \mathrm{Q}, \mathrm{A}, \mathrm{B}$.*

Proof. If $T \in \Gamma$, then by Theorem E.5, T preserves each grade-k subspace: $TC\ell^kT^{-1} = C\ell^k$ for all $k = 0, \dots, n$. Thus, T preserves any direct sum of these subspaces. □

To facilitate the construction of mappings equivariant to these groups P, Q, A, and B, we establish equivalent definitions (9) for them that avoid similarity transformations. Instead, these new definitions are based on the concepts of norm functions, the center Z (5) of $C\ell$, reversion, Clifford conjugation, which serve as inspiration for construction of new equivariant mappings containing these concepts and significantly simplifies the further proofs.

Here and below, by *norm functions* of a multivector $T \in C\ell$ we mean the products[7] $\widetilde{T}T$ and $\widehat{\widetilde{T}}T$, which, for $T \in \Gamma$, are invertible scalars (in the case of the non-degenerate algebra and even n).

Theorem 3.4 *The groups P, A, B, Q can be equivalently defined [8]:*

$$\begin{gathered} \mathrm{P} = (C\ell^{(0)\times} \cup C\ell^{(1)\times})\mathrm{Z}^\times, \quad \mathrm{A} = \{T \in C\ell^\times : \widetilde{T}T \in \mathrm{Z}^\times\}, \quad \mathrm{B} = \{T \in C\ell^\times : \widehat{\widetilde{T}}T \in \mathrm{Z}^\times\}, \\ \mathrm{Q} = \{T \in C\ell^\times : \quad \widetilde{T}T \in \mathrm{Z}^\times, \quad \widehat{\widetilde{T}}T \in \mathrm{Z}^\times\} = \mathrm{A} \cap \mathrm{B} = \mathrm{P} \cap \mathrm{A} = \mathrm{P} \cap \mathrm{B}. \end{gathered} \tag{9}$$

The proof is based on the connection of these groups with the centralizers of certain subspaces and is given in Appendix E.5. ***Key takeaways from the subsection:*** 1) $\mathrm{O}(V, \mathfrak{q})$-equivariance can be achieved by forcing a mapping to be Γ- or P-, A-, B-, Q-equivariant (Theorems 3.1–3.2 and Lemma 3.3). 2) The groups P, A, B, Q can be defined either by preservation of fundamental GA subspaces under the similarity transformation ad_T (8), or using the notion of center, norm functions, etc. (Theorem 3.4).

[6] Naming convention: P stands for parity (preserving the odd grading), A and B denote the two symmetric cases (using the first letters of the alphabet), and Q refers to the quaternion-type subspace $C\ell^{\overline{1}}$.

[7] These functions are used in the theory of spin groups (Lounesto, 1997; Benn & Tucker, 1987; Crumeyrolle, 1990), see (42)–(46).

[8] Elements of P have the form UV, where $U \in C\ell^{(0)\times} \cup C\ell^{(1)\times}$ is even or odd, and $V \in \mathrm{Z}^\times$.

3.2 NEW EQUIVARIANT MAPPINGS

This section establishes equivariance of several fundamental GA mappings with respect to the groups P, A, B, Q (9), Γ (4), and the group of invertible elements $C\ell^\times$. The proof of Theorem 3.5 is based on equivalent definitions of P, A, B, Q and linearity and multiplicativity of ad_T and is provided in Appendix E.6. These mappings serve for constructing equivariant neural network layers in Section 4.

Theorem 3.5 *The mappings from Table 1 are equivariant with respect to the corresponding groups. For example,* $\mathrm{ad}_T(\langle U\rangle_{(s)}) = \langle \mathrm{ad}_T(U)\rangle_{(s)}$*, where* $s = 0, 1$*, for any* $U \in C\ell$ *and* $T \in \mathrm{P}$.

Table 1: The first column specifies the mapping, the second lists the groups with respect to which it is equivariant, the third provides the theorems where this is stated.

MAPPINGS	GROUPS	THEOREMS
Polynomials $F \in \mathbb{F}[U_1, \dots, U_l]$	$C\ell^\times$, P, A, B, Q, Γ	Th. E.11
Projection $\langle\cdot\rangle_0$ onto $C\ell^0$	$C\ell^\times$, P, A, B, Q, Γ	Th. E.14
Projections $\langle\cdot\rangle_k$ onto $C\ell^k$, $k = 1,\dots,n$	Γ	(Ruhe et al., 2023a)
Projections $\langle\cdot\rangle_{\overline{m}}$ onto $C\ell^{\overline{m}}$, $m = 0, 1, 2, 3$	Q, Γ	(Filimoshina & Shirokov, 2025)
$\langle\cdot\rangle_{(0)}$, $\langle\cdot\rangle_{(1)}$; grade involution $\widehat{\ }$; function $\widehat{T}T$	P, Q, Γ	Th. E.10, E.12, E.13
$\langle\cdot\rangle_{\overline{01}}$, $\langle\cdot\rangle_{\overline{23}}$; reversion $\widetilde{\ }$; function $\widetilde{T}T$	A, Q, Γ	Th. E.10, E.12, E.13
$\langle\cdot\rangle_{\overline{12}}$, $\langle\cdot\rangle_{\overline{03}}$; Clifford conjugation $\widehat{\widetilde{\ }}$; function $\widehat{\widetilde{T}}T$	B, Q, Γ	Th. E.10, E.12, E.13

4 EQUIVARIANT LAYERS

This section presents a general framework for constructing equivariant layers using GAs. This framework generalizes prior state-of-the-art architectures: CGENN (Ruhe et al., 2023a) and GLGENN (Filimoshina & Shirokov, 2025). More detailed description can be found in Appendix F; the number of parameters for each layer is in Table 4. In this and the following sections, we consider the real GAs $C\ell(\mathbb{R}^{p,q,r})$. The inputs and outputs of our networks are represented as GA elements (multivectors), and all layer operations are multivector-valued. Almost any geometric-based data can be represented as multivectors (see example in Subsection 6). Suppose each input data object has l channels $x_1, \dots, x_l \in C\ell$. We can represent any $x_{c_{in}}$, $c_{in} = 1, \dots, l$, as sums of projections (2). For a unified description of the layers, we introduce an auxiliary notation for the set of subspace indices associated with each group:

$$\mathrm{S} := \begin{cases} \{(0), (1)\}, & \text{for P;} \qquad \{\overline{01}, \overline{23}\}, \text{for A;} \qquad \{\overline{12}, \overline{03}\}, \text{ for B;} \\ \{\overline{0}, \overline{1}, \overline{2}, \overline{3}\}, & \text{for Q (GLGENN);} \qquad \{0, 1, \dots, n\}, \text{ for } \Gamma \text{ (CGENN).} \end{cases} \tag{10}$$

GA-linear layers We now introduce GA-linear layers, where we parameterize linear combinations of projections of input multivectors onto specific $C\ell$ subspaces:

$$\mathrm{GALinear}(x_1, \dots, x_l, c_{out}, \mathrm{G}) := \sum_{s\in\mathrm{S}, c_{in}\in\{1,\dots,l\}} \phi_{c_{out}c_{in}s}\langle x_{c_{in}}\rangle_s, \tag{11}$$

where G specifies the desired equivariance group (P, A, B, Q, or Γ), S (10) is the set of subspace indices associated with that group, and $\phi_{c_{out}c_{in}s} \in \mathbb{R}$ are optimizable parameters.

GA-geometric product layers To introduce non-linearities consistent with GAs, we parameterize geometric products equivariant with respect to different groups. First, for the input channels $x_1, \dots, x_l$, we calculate their linear combinations using GA-linear layer as $z_i := \mathrm{GALinear}(x_1, \dots, x_l, c_i, \mathrm{G})$, $i = 1, \dots, l$. We then define a parameterized geometric product for each pair $(x_{c_{in}}, z_{c_{in}})$, $c_{in} = 1, \dots, l$:

$$\mathrm{GAProduct}(x_{c_{in}}, z_{c_{in}}, \mathrm{G}) := \sum_{s_1,s_2,s_3\in\mathrm{S}} \phi_{s_1s_2s_3}\langle\langle x_{c_{in}}\rangle_{s_1}\langle z_{c_{in}}\rangle_{s_2}\rangle_{s_3}, \tag{12}$$

where G specifies the equivariance group and $\phi_{s_1s_2s_3} \in \mathbb{R}$ are optimizable weights. The layer defined in (12) preserves the number of input and output channels (l). For greater expressiveness, we define fully connected layer, where the number of output channels may vary: $\mathrm{GALinear}(\mathrm{GAProduct}(x_1, z_1, \mathrm{G}), \dots, \mathrm{GAProduct}(x_l, z_l, \mathrm{G}), c_{out}, \mathrm{G})$.

GA-normalization layers To enhance numerical stability, we introduce a normalization framework applied to layer inputs prior to the GA-geometric product layer. For each input channel $x_{c_{in}}$, $c_{in} = 1, \dots, l$, we compute:

$$\mathrm{GANormalization}(x_{c_{in}}, \mathrm{G}) := \sum_{s\in\mathrm{S}} \langle x_{c_{in}}\rangle_s \ \left[\sigma(\phi_{c_{in}s})(\mathrm{Norm}(\langle x_{c_{in}}\rangle_s) - 1) + 1\right]^{-1}, \tag{13}$$

where G specifies the desired equivariance group, $\phi_{c_{in}s} \in \mathbb{R}$ are optimizable weights, σ is the logistic sigmoid function, and Norm is a norm function that depends on the group G: Norm(x) $= \sqrt{|\langle \widetilde{x}x \rangle_0|}$ can be used for A, Q, Γ; Norm(x)$= \sqrt{|\langle \widehat{\widetilde{x}}x \rangle_0|}$ can be applied for B, Q, Γ; and Norm(x) $= \sqrt{|\langle \widehat{x}x \rangle_0|}$ can be applied for P, Q, Γ. [9]

The mappings (11), (12), and (13) are equivariant with respect to the corresponding group G = P, A, B, Q, Γ, since they are compositions of polynomials, projections, and norm functions, which are equivariant by Table 1. We also parameterize linear combinations and geometric products of projections onto the direct sums of grade subspaces with step size 3 and call them *Triple-linear layers* and *Triple-geometric product layers* (they are equivariant with respect to Γ but contain less parameters than CGENN).

5 Configurations and trade-offs

We can construct networks by combining GA-linear, GA-geometric product, and GA-normalization layers in various sequences, with each layer chosen to be equivariant to any of P, A, B, Q, or Γ. We define a *configuration* (config) as a specific sequence of these layers. There are two strategies: 1) *Homogeneous config*: all layers are chosen to be equivariant with respect to the same group (for example, all layers (11)–(13) use G = A). The resulting network is fully equivariant with respect to that group. 2) *Heterogeneous config*: layers equivariant with respect to different groups are mixed within a single network (for example, GALinear (11) with G = A followed by GAProduct (12) with G = Q). The overall equivariance of the network is constrained to the intersection of all groups in its layers. For example, mixing A-, Q-, and Γ-equivariant layers results in a network equivariant with respect to $\mathrm{A} \cap \mathrm{Q} \cap \Gamma = \Gamma$ (see Euler diagram in Figure 1).

Any config is automatically equivariant to $\mathrm{O}_{\Lambda^1}(V, \mathfrak{q})$ in even-dimensional spaces (n is even) and to $\mathrm{SO}_{\Lambda^1}(V, \mathfrak{q})$ in odd-dimensional spaces (n is odd). If $\mathrm{O}_{\Lambda^1}(V, \mathfrak{q})$ equivariance is required in odd-dimensional spaces, it can still be achieved by using not all layers in a config. The choice of the config induces trade-off between expressiveness and parameter efficiency. The larger the group, the fewer weights are required to parameterize the corresponding layers (the relative 'size' of these groups is visualized in Figure 1; we consider P, A, and B to be 'large' groups, Q a 'medium' group, and Γ a 'small' group). However, a larger group imposes stricter equivariance constraints, forcing the model to be equivariant to transformations beyond $\mathrm{O}_{\Lambda^1}(V, \mathfrak{q})$. Consequently, larger groups result in less expressive layers. An important corollary is that networks equivariant with respect to smaller groups are more expressive but prone to overfitting, particularly on small datasets. To provide guidance for selecting a config, we present a decision flowchart in Figure 5 and summarize our heuristics in Table 5.

6 Experimental results

We compare our homogeneous configs using A-, B-, P-equivariant layers and Triple-layers (we call them A-, B-, P-, Triple-configs), as well as heterogeneous configs mixing the groups which demonstrated strong performance, against state-of-the-art models for these tasks: GA-based models GLGENN ((Filimoshina & Shirokov, 2025), corresponds to Q-config) and CGENN ((Ruhe et al., 2023a), Γ-config) and several non-GA models. We report mean $\pm$ standard deviation (with ddof=1) on test over 20 runs for $\mathrm{O}(6)$-Convex Hull, $\mathrm{O}(8)$-Regression, and N-Body experiments, and over 5 runs for $\mathrm{O}(1,3)$-Top Tagging. All experimental details and discussions can be found in Appendix G and our code. Computational cost trade-off discussion is in Appendix H. Note that empirically, the benefit of using wider equivariance groups cannot be matched by simply adding traditional regularizers to a less constrained model (see Appendix I).

$\mathrm{O}(6)$-Convex Hull Volume Estimation We consider a benchmark equivariant task of estimation volume of a convex hull generated by 16 points in $\mathbb{R}^6$ (Ruhe et al., 2023a; Liu et al., 2024a; Filimoshina & Shirokov, 2025). Points are sampled from a standard Gaussian distribution; the volume is $\mathrm{O}(6)$-invariant. Each point is embedded into $C\ell_6$ as a grade-1 vector. We compare A-, P-, B-, Triple-; baseline Q- and Γ-; A-Q-Triple-, Triple-Γ-, and Triple-Q-configs; and non-GA baselines: GVP Jing et al. (2021), VN Deng et al. (2021) (both scalarization equivariant methods), Equivariant MLP (EMLP) Finzi et al. (2021), and standard MLP. The number of parameters for non-GA models is aligned with our best-performing and largest configs respectively. Models are trained on 256, 1024, and 4096 samples. The results are in Figure 3 (A) and Table 6. Triple-Γ-config achieves the lowest observed mean MSE across our runs among all homogeneous configs and competitors.

[9] See Appendix F for a discussion of alternative nonlinearities, including GA-SiLU.

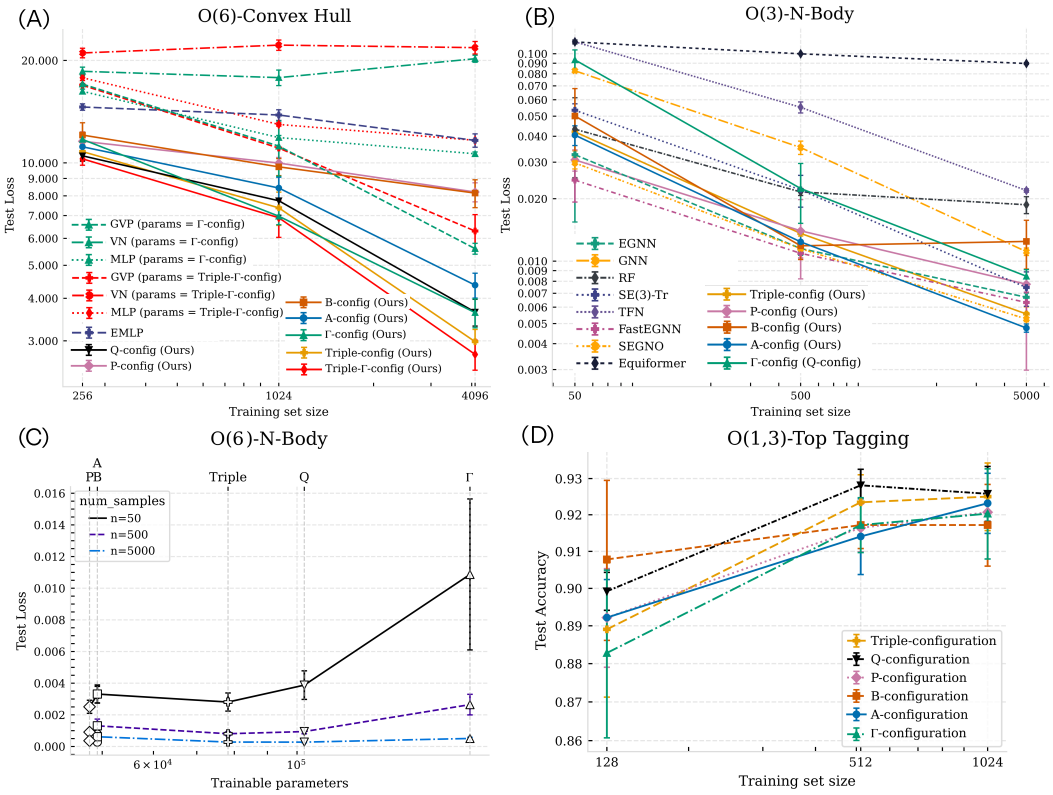


Figure 3: Test loss vs. (i) plots A, B, D: training set size; (ii) plot C: model size (x-axis; in terms of trainable parameters) and training set size (line type). Loss is MSE (↓) for A-C; Accuracy (↑) for D.

Almost all our configs achieve lower observed mean MSE than non-GA models across our runs, especially when the train set is small.

$\mathrm{O}(8)$-Regression We design a synthetic regression task to test the ability of our models to learn a non-linear function of geometric quantities. Each data sample has two channels x_i, $i = 1, 2$, each represented by one scalar ($c_i \in \mathbb{R}$), one vector ($\mathbf{v}_i = (a_{1i}, \ldots, a_{8i}) \in \mathbb{R}^8$, for example, coordinates of a point or vector), and one pseudoscalar ($p_i \in \mathbb{R}$, for example, oriented volume), all sampled from a standard Gaussian distribution. The regression target is $c_1 \sin(\|\mathbf{v}_1\|) - \frac{1}{2}|\mathbf{v}_2|^3 + c_1 c_2 \frac{(\mathbf{v}_1^{\mathrm{T}} \mathbf{v}_2)}{\|\mathbf{v}_1\| \|\mathbf{v}_2\|} + \arctan(|p_1|)$. The data x_i, $i = 1, 2$, can be embedded into $C\ell_8$ as $x_i = c_i e + a_{1i} e_1 + \cdots + a_{8i} e_8 + p_i e_{1\ldots8}$. The results are in Figure 8 and Table 7. Medium-sized configs (Q, Q-Triple, Triple) achieve the lowest test MSE across all train set sizes. On the smallest dataset, the parameter-light A-config has the best performance, underscoring the regularizing effect of larger groups when data is scarce.

$\mathrm{O}(3)$- and $\mathrm{O}(6)$-N-Body Dynamics Simulation The N-body is a standard benchmark for assessing the performance of equivariant (graph) neural networks in simulating physical systems. The system is defined by the masses, initial positions, and initial velocities of the bodies, which evolve under Newtonian gravity forces. The objective is to predict their final positions after 1000 steps of Euler integration. We construct a graph network with message and update networks chosen from our configs. This experiment is considered, for example, in Kipf et al. (2018); Ruhe et al. (2023a); Brehmer et al. (2023); Filimoshina & Shirokov (2025) in $\mathrm{O}(3)$- and $\mathrm{O}(5)$-settings. We consider 2 tasks: simulation the dynamics of $N = 5$ charged particles (bodies) in $\mathbb{R}^{3,0}$ (standard setting, for fair comparison with irreps-based equivariant methods, since they are tailored specially for $\mathrm{O}(3)$-equivariance) and $\mathbb{R}^{6,0}$ (to compare our configs). Performance is evaluated across train set sizes of 50, 500, and 5000 samples. Results are in Figure 3 (B-C) and Tables 8-9. In $\mathrm{O}(3)$-setting, we compare our P-, A-, B-, Triple-, Γ- (coincides with Q in this low-dimensional case) configs with: a) Scalarization equivariant methods: Radial Field (RF) Köhler et al. (2020), EGNN Satorras et al. (2021), FastEGNN Zhang et al. (2024); b) Irreps/tensor products equivariant methods: SE(3)-Transformers Fuchs et al. (2020), TFN Thomas et al. (2018), Equiformer Liao & Smidt (2023); c) Method based on Graph Neural ODE: SEGNO Liu et al. (2024b); d) Non-equivariant counterpart of EGNN: GNN Satorras et al. (2021). The architectures and numbers of parameters for non-GA models are taken from their code repositories (best-reported models). In general, our configs per-

form as non-GA state-of-the-art models SEGNO and FastEGNN. In $\mathrm{O}(6)$-setting, we compare A-, P-, B-, Triple-; baseline Q- and Γ-; and A-P-, A-Triple-, and A-Q-configs. On the smallest dataset, the parameter-efficient configs achieve much lower MSE than Q- and Γ-configs. A-Triple-config achieves among the lowest observed mean MSE across all train set sizes. Γ-config advantage over more parameter-efficient configs is minimal compared to the 20x parameter increase.

$\mathrm{O}(1,3)$-Top Tagging Top quark jet tagging benchmark (Ruhe et al., 2023a; Kasieczka et al., 2019) is a large-scale particle physics task requiring equivariance under Lorentz boosts (transformations preserving the Minkowski metric; see details on $C\ell_{1,3}$ in Appendix E.1). The input consists of the energy-momentum 4-vectors for up to 200 constituent particles per jet, which must be processed in a way that respects the $\mathrm{O}(1,3)$ symmetry of spacetime. The task is to identify and categorize high-energy jets produced in particle collisions, such as those measured by the ATLAS experiment at CERN (Butterworth et al., 2008), by identifying those originating from top quarks, the heaviest particles in the Standard Model (ATLAS Collaboration, 2017; Incandela et al., 2009). We compare A-, P-, B-, Triple, Q-, and Γ-configs on 128, 512, and 1024 training samples, see Figure 3 (D) and Table 10. All configs achieve remarkably similar accuracy, however the number of parameters in the models varies a lot (from 251k for P- to 321k for Γ-config). Note that we were able to get accuracy 0.945 for Q- and 0.941 for Γ-config when training on 1024 samples, while CGENN paper reports accuracies on the full dataset (1.2M samples): 0.942 CGENN, 0.936 ResNeXt Xie et al. (2016), 0.93 P-CNN CMS Collaboration (2017), 0.932 PFN Komiske et al. (2019), 0.94 ParticleNet Qu & Gouskos (2020), 0.922 EGNN Satorras et al. (2021), 0.929 LGN Bogatskiy et al. (2020), 0.942 LorentzNet Gong et al. (2022).

$\mathrm{O}(3)$-Atom Position Prediction We evaluate our models in molecular domain on MD17 benchmark Chmiela et al. (2017). The task is to predict atom positions in the next ten time steps using ten time frames for each molecule (we took aspirin, benzene, and malonaldehyde). Following the standard setup, we use 5000 train and 2000 test samples. We take all baselines from CSMPN (coincides with Γ-config) and EqMotion Xu et al. (2023). For our configs, we use a similar architecture to CSMPN (matches competitors in number of parameters, ≈300k) with 5 message passing layers with GA-layers as message/update functions. Our models have 136k (A), 207k (Triple), and 369k (Γ) parameters. In Table 2, we report ADE (RMSE averaged over time steps) and FDE (RMSE at final time step). Notably, even with fewer parameters, A- and Triple-configs at 10^4 steps achieve comparable observed mean ADE/FDE to many 10^5-step baselines. We note that our Γ-config is trained with 10^4 steps due to computational constraints (see limitations in Appendix J), whereas the CSMPN uses 10^5 steps and is therefore not directly comparable.

Table 2: ADE/FDE ($\times 10^{-2}$, ↓) on the MD17 test set.

MODEL	# TRAIN STEPS	ASPIRIN	BENZENE	MALONALDEHYDE
A	10^4	7.51/10.35	1.12/1.74	5.17/7.42
Triple	10^4	7.13/10.13	1.31/2.01	4.89/7.13
Γ	10^4	6.30/9.33	1.33/1.85	5.04/7.38
Γ (CSMPN)	10^5	3.82/5.75	1.03/1.60	3.88/5.94
EMPSN Liu et al. (2024a)	10^5	9.53/12.63	1.03/1.12	7.83/10.85
EqMotion Xu et al. (2023)	10^5	5.95/8.38	1.18/1.73	5.85/9.02
GroupNet Xu et al. (2022)	10^5	10.62/14.00	2.02/2.95	7.99/12.49
NMMP Hu et al. (2020)	10^5	10.41/14.67	2.21/3.33	9.50/14.89
NRI Kipf et al. (2018)	10^5	12.60/18.50	1.89/2.58	12.79/19.86
EGNN Satorras et al. (2021)	10^5	14.61/20.65	7.50/12.16	15.21/24.00
SE(3)-Tr. Fuchs et al. (2020)	10^5	15.70/22.39	7.62/12.50	15.44/24.47
TFN Thomas et al. (2018)	10^5	15.02/21.35	7.55/12.30	15.21/24.32
RF Köhler et al. (2020)	10^5	17.98/26.20	7.73/12.47	16.53/25.10

7 KEY TAKEAWAYS AND CONCLUSIONS

This work establishes that orthogonal $\mathrm{O}(V,\mathfrak{q})$-equivariance can be guaranteed by enforcing equivariance with respect to larger groups. For this purpose, we introduce and consider the classes of groups P, A, B, Q in GAs and prove that equivariance with respect to them implies the Clifford group Γ-equivariance, which is equivalent to $\mathrm{O}(V,\mathfrak{q})$-equivariance. The larger is the chosen equivariance group, the more restrictions we put on our model, imposing stronger symmetry constraints and forcing it to be equivariant with respect to some other transformations besides orthogonal. With such an approach, we reduce expressiveness of the model, but improve parameter efficiency. Our framework provides a spectrum of models configs, from highly efficient (large-group-equivariant) to highly expressive (small-group-equivariant), with possibility of combining them in heterogeneous configs, which can achieve balance across tasks. Further, it would be interesting to extend this principle to other symmetry groups and apply the framework to other large-scale scientific problems.

AI USE STATEMENT

In this work, we used generative AI tools only for editing and polishing the manuscript to improve readability, grammar, and style. We have not used generative AI tools for any tasks requiring disclosure, including generating synthetic datasets, developing theoretical models or conceptual frameworks, formulating mathematical claims, providing critical ingredients for proving mathematical claims, assisting in the writing of proofs, proposing or refining hypotheses, designing or providing feedback on research methodology or experiments, implementing methods, cleaning or reformatting datasets, supporting qualitative or thematic data analysis, or interpreting results. We also did not use generative AI tools for other recommended-disclosure tasks such as creating or modifying scientific figures or images, suggesting experimental parameters, creating or editing software code, summarizing or analysing existing literature, discovering research topics or identifying gaps, brainstorming, sourcing or searching for information, identifying relevant literature, formatting references, suggesting a structure for a research paper, or proposing a title or keywords for a research paper. We have reviewed all AI-assisted work. We take responsibility for the final content of this work, including text, claims, and artifacts.

ETHICS STATEMENT

This work presents fundamental research on equivariant neural networks based on geometric algebras. Our experiments are conducted on public benchmarking tasks. We are not aware of any immediate negative ethical impacts arising directly from our methodology. However, as with any advancement in machine learning, there is a potential for future downstream applications to have societal consequences. We encourage responsible development and deployment of such technologies.

REPRODUCIBILITY STATEMENT

To facilitate reproducibility and further research, we provide an anonymous code repository as part of the supplementary materials. It contains implementations of all proposed equivariant layers and model configurations, the exact experimental configurations, hyperparameter settings, data generation and preprocessing scripts, and training and evaluation scripts for all benchmarks reported in this work.

For the theoretical results, complete proofs of all theorems, lemmas, and corollaries are provided in Appendix E.5 (groups P, A, B, Q) and Appendix E.6 (equivariance of mappings), together with the required definitions, assumptions, and auxiliary statements in Appendices E.1–E.4.

For the experimental results, we describe all datasets, data splits, embedding procedures, model architectures, layer configurations, optimizers, batch sizes, learning rates, and numbers of training steps in code repository. The synthetic benchmarks ($\mathrm{O}(6)$-Convex Hull, $\mathrm{O}(8)$-Regression, and $\mathrm{O}(3)$- and $\mathrm{O}(6)$-N-Body) are generated by scripts included in the supplementary materials, which fully specify the data generation distributions. For the public datasets ($\mathrm{O}(1,3)$-Top Tagging and MD17), we reference the original sources.

All results are reported as mean $\pm$ standard deviation over multiple runs with varying random seeds, as specified in the captions of the corresponding tables. Computational resources used for the experiments are described in Appendix G.6, and wall-clock times are reported in Appendix H.

REFERENCES

R. Ablamowicz. Structure of spin groups associated with degenerate Clifford algebras. *Journal of Mathematical Physics*, 27(1):1–6, 1986.

Riccardo Ali, Paulina Kulytė, Haitz Sáez de Ocáriz Borde, and Pietro Lio. Metric learning for Clifford group equivariant neural networks. In *ICML 2024 Workshop on Geometry-grounded Representation Learning and Generative Modeling*, 2024. URL https://openreview.net/forum?id=4tw1U41t6Q.

M. Atiyah, R. Bott, and A. Shapiro. Clifford modules. *Topology*, 3:3–38, 1964.

ATLAS Collaboration. Jet energy scale measurements and their systematic uncertainties in proton-proton collisions at $\sqrt{s} = 13$ tev with the ATLAS detector. *arXiv preprint*, 2017.

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):2453, 2022. doi: 10.1038/s41467-022-29939-5.

E. Bayro-Corrochano. *Geometric Algebra Applications, vol. I, Computer Vision, Graphics and Neurocomputing*. Springer, Cham, 2019.

Eduardo Bayro-Corrochano and Sven Buchholz. Geometric neural networks. In Gerald Sommer and Jan J. Koenderink (eds.), *Algebraic Frames for the Perception-Action Cycle*, pp. 379–394, Berlin, Heidelberg, 1997. Springer Berlin Heidelberg.

Eduardo Bayro-Corrochano and Gerik Scheuermann. *Geometric Algebra Computing in Engineering and Computer Science*. Springer-Verlag, London, 2010.

I. Benn and R. Tucker. *An Introduction to Spinors and Geometry with Applications in Physics*. Adam Hilger, Bristol, 1987.

Alexander Bogatskiy, Brandon Anderson, Jan Offermann, Marwah Roussi, David Miller, and Risi Kondor. Lorentz group equivariant neural network for particle physics. In Hal Daumé III and Aarti Singh (eds.), *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pp. 992–1002, 13–18 Jul 2020.

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

J. Brehmer, P. De Haan, S. Behrends, and T. Cohen. Geometric algebra transformer. In *Advances in Neural Information Processing Systems*, volume 37, 2023.

Michael Bronstein, Joan Bruna, Taco Cohen, and Petar Veličković. *Geometric Deep Learning: Grids, Groups, Graphs, Geodesics, and Gauges*. 2021.

J. Brooke. *Clifford Algebras, Spin Groups and Galilei Invariance—New Perspectives*. PhD thesis, University of Alberta, 1980.

Sven Buchholz and Gerald Sommer. On Clifford neurons and Clifford multi-layer perceptrons. *Neural Networks*, 21(7):925–935, 2008.

J. M. Butterworth, J. Thion, U. Bratzler, P. N. Ratoff, R. B. Nickerson, J. M. Seixas, I. Grabowska-Bold, F. Meisel, S. Lokwitz, et al. The ATLAS experiment at the CERN large hadron collider. *Journal of Instrumentation*, 3:S08003, 2008.

Stefan Chmiela, Alexandre Tkatchenko, Huziel E. Sauceda, Igor Poltavsky, Kristof T. Schütt, and Klaus-Robert Müller. Machine learning of accurate energy-conserving molecular force fields. *Science Advances*, 3(5), 2017. doi: 10.1126/sciadv.1603015. URL `https://www.science.org/doi/abs/10.1126/sciadv.1603015`.

CMS Collaboration. Boosted jet identification using particle candidates and deep neural networks. Detector Performance Figures CMS-DP-17-049, CERN, 2017.

T. Cohen and M. Welling. Group equivariant convolutional networks. In *Proceedings of The 33rd International Conference on Machine Learning*, volume 48 of *PMLR*, pp. 2990–2999, 2016.

A. Crumeyrolle. *Orthogonal and Symplectic Clifford Algebras*. Springer, Netherlands, 1st edition, 1990.

C. Deng, O. Litany, Y. Duan, A. Poulenard, A. Tagliasacchi, and L. J. Guibas. Vector neurons: A general framework for SO(3)-equivariant networks. In *IEEE International Conference on Computer Vision (ICCV)*, pp. 12180–12189, Montreal, QC, Canada, 2021. IEEE.

Paul A. M. Dirac. The quantum theory of the electron. *Proceedings of the Royal Society of London. Series A*, 117(778):610–624, 1928.

C. Doran and A. Lasenby. *Geometric Algebra for Physicists*. Cambridge University Press, Cambridge, UK, 2003.

L. Dorst, C. Doran, and J. Lasenby. *Applications of Geometric Algebra in Computer Science and Engineering*. Birkhäuser, Boston, MA, 2002.

Leo Dorst and Steven De Keninck. A guided tour to the plane-based geometric algebra PGA, 2022. URL `http://bivector.net/PGA4CS.html`. Available at `http://bivector.net/PGA4CS.html`.

Leo Dorst, Daniel Fontijne, and Stephen Mann. *Geometric Algebra for Computer Science: An Object-Oriented Approach to Geometry*. Morgan Kaufmann, San Francisco, CA, 2007.

Alexandre Duval, Simon V. Mathis, Chaitanya K. Joshi, Victor Schmidt, Santiago Miret, Fragkiskos D. Malliaros, Taco Cohen, Pietro Lio, Yoshua Bengio, and Michael Bronstein. A hitchhiker's guide to geometric GNNs for 3D atomic systems. *arXiv preprint*, December 2023.

E. Filimoshina and D. Shirokov. On some Lie groups in degenerate Clifford geometric algebras. *Advances in Applied Clifford Algebras*, 33(44), 2023.

E. Filimoshina and D. Shirokov. On generalization of Lipschitz groups and spin groups. *Mathematical Methods in the Applied Sciences*, 47(3):1375–1400, 2024.

E. Filimoshina and D. Shirokov. GLGENN: A novel parameter-light equivariant neural networks architecture based on Clifford geometric algebras. In *Proceedings of the 42nd International Conference on Machine Learning (Vancouver, Canada, 2025)*, 2025.

M. Finzi, M. Welling, and A. G. Wilson. Practical method for constructing equivariant multilayer perceptrons for arbitrary matrix groups. volume 139 of *PMLR*, pp. 3318–3328, 2021.

Fabian Fuchs, Daniel Worrall, Volker Fischer, and Max Welling. SE(3)-transformers: 3D roto-translation equivariant attention networks. In *Advances in Neural Information Processing Systems*, volume 33, pp. 1970–1981, 2020.

Shiqi Gong, Qi Meng, Jue Zhang, Huilin Qu, Congqiao Li, Sitian Qian, Weitao Du, Zhi-Ming Ma, and Tie-Yan Liu. An efficient lorentz equivariant graph neural network for jet tagging. *Journal of High Energy Physics*, 2022.

J. Helmstetter and A. Micali. *Quadratic Mappings and Clifford Algebras*. Birkhäuser, Basel, 2008.

D. Hestenes and G. Sobczyk. *Clifford Algebra to Geometric Calculus – A Unified Language for Mathematical Physics*. Reidel Publishing Company, Dordrecht, 1984.

Dietmar Hildenbrand. *Introduction to Geometric Algebra Computing*. Chapman and Hall/CRC, New York, 1st edition, 2018.

Yue Hu, Siheng Chen, Ya Zhang, and Xiao Gu. Collaborative motion prediction via neural motion message passing. In *Computer Vision and Pattern Recognition (CVPR)*, pp. 6318–6327, 2020.

Lei Huang, Jie Qin, Yi Zhou, Fan Zhu, Li Liu, and Ling Shao. Normalization techniques in training dnns: Methodology, analysis and application. *IEEE transactions on pattern analysis and machine intelligence*, 45(8):10173–10196, 2023.

Wenbing Huang, Jiaqi Han, Yu Rong, Tingyang Xu, Fuchun Sun, and Junzhou Huang. Equivariant graph mechanics networks with constraints. In *International Conference on Learning Representations (ICLR)*, 2022.

Joseph R. Incandela, Arnulf Quadt, Wolfgang Wagner, and Daniel Wicke. Status and prospects of top-quark physics. *Progress in Particle and Nuclear Physics*, 63(2):239–292, 2009.

B. Jing, S. Eismann, P. Suriana, R. Townshend, and R. Dror. Learning from protein structure with geometric vector perceptrons. In *ICLR*, 2021.

Gregor Kasieczka, Tilman Plehn, Jennifer Thompson, and Michael Russel. Top quark tagging reference dataset. https://doi.org/10.5281/zenodo.2603256, March 2019.

Thomas Kipf, Ethan Fetaya, Kuan-Chieh Wang, Max Welling, and Richard Zemel. Neural relational inference for interacting systems. In *International Conference on Machine Learning*, PMLR, pp. 2688–2697, 2018.

M.-A. Knus. *Quadratic and Hermitian forms over rings*. Springer, Berlin, 1991.

Jonas Köhler, Leon Klein, and Frank Noé. Equivariant flows: exact likelihood generative learning for symmetric densities. In *Proceedings of the 37th International Conference on Machine Learning*, ICML'20, 2020.

Patrick T. Komiske, Eric M. Metodiev, and Jesse Thaler. Energy flow networks: deep sets for particle jets. *Journal of High Energy Physics*, 2019.

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

C. Liu, D. Ruhe, F. Eijkelboom, and P. Forré. Clifford group equivariant simplicial message passing networks. In *ICLR*, 2024a.

Yang Liu, Jiashun Cheng, Haihong Zhao, Tingyang Xu, Peilin Zhao, Fugee Tsung, Jia Li, and Yu Rong. Improving generalization in equivariant graph neural networks with physical inductive biases. In *International Conference on Learning Representations*, 2024b. URL https://openreview.net/forum?id=Hkx1qkrKPr](https://openreview.net/forum?id=3oTPsORaDH.

P. Lounesto. *Clifford Algebras and Spinors*. Cambridge University Press, Cambridge, 1997.

N. Marchuk and D. Shirokov. *Theory of Clifford algebras and spinors (in Russian)*. Krasand (URSS), ISBN 978-5-396-01014-7, Moscow, 2020.

J.K. Pearson and D.L. Bisset. Neural networks in the Clifford domain. In *Proceedings of 1994 IEEE International Conference on Neural Networks (ICNN'94)*, volume 3, pp. 1465–1469 vol.3, 1994.

A. Pepe, S. Buchholz, and J. Lasenby. Clifford group equivariant neural network layers for protein structure prediction. In *Northern Lights Deep Learning Conference*, 2024.

Christian Perwass. *Geometric Algebra with Applications in Engineering*. Springer, Berlin, Heidelberg, 2009.

I. Porteous. *Clifford Algebras and the Classical Groups*. Cambridge University Press, Cambridge, 1995.

Huilin Qu and Loukas Gouskos. Jet tagging via particle clouds. *Physical Review D*, 2020.

David Ruhe, Johannes Brandstetter, and Patrick Forré. Clifford group equivariant neural networks. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023a.

David Ruhe, Jayesh K. Gupta, S. De Keninck, Max Welling, and Johannes Brandstetter. Geometric Clifford algebra networks. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pp. 29306–29337, 2023b. arXiv:2302.06594.

V. G. Satorras, E. Hoogeboom, and M. Welling. E(n)-equivariant graph neural networks. In *ICML*, 2021.

D. Shirokov. On inner automorphisms preserving fixed subspaces of Clifford algebras. *Advances in Applied Clifford Algebras*, 31:30, 2021.

D. S. Shirokov. Classification of elements of Clifford algebras according to quaternionic types. *Doklady Mathematics*, 80(1):610–612, 2009.

D. S. Shirokov. Quaternion typification of Clifford algebra elements. *Advances in Applied Clifford Algebras*, 22(1):243–256, 2012a.

D. S. Shirokov. Development of the method of quaternion typification of Clifford algebra elements. *Advances in Applied Clifford Algebras*, 22(2):483–497, 2012b.

N. Thomas, T. E. Smidt, S. Kearnes, L. Yang, L. Li, K. Kohlhoff, and P. Riley. Tensor field networks: Rotation- and translation-equivariant neural networks for 3D point clouds. *CoRR*, 2018.

T. Walpuski. Differential geometry IV, Lecture notes, 2022. `https://walpu.ski/Teaching/SS22/DifferentialGeometry4/`.

Ziming Wang and Rebecka Jörnsten. SE(3)-bi-equivariant transformers for point cloud assembly. In *NeurIPS 2024*, 2023.

Daniel Worrall, Stephan Garbin, Daniyar Turmukhambetov, and Gabriel J. Brostow. Harmonic networks: Deep translation and rotation equivariance. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 5028–5037, 2017.

Saining Xie, Ross Girshick, Piotr Dollár, Zhuowen Tu, and Kaiming He. Aggregated residual transformations for deep neural networks. *arXiv preprint arXiv:1611.05431*, 2016.

Chenxin Xu, Maosen Li, Zhenyang Ni, Ya Zhang, and Siheng Chen. Groupnet: Multiscale hypergraph neural networks for trajectory prediction with relational reasoning. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 6498–6507, 2022.

Chenxin Xu, Robby T Tan, Yuhong Tan, Siheng Chen, Yu Guang Wang, Xinchao Wang, and Yanfeng Wang. Eqmotion: Equivariant multi-agent motion prediction with invariant interaction reasoning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 1410–1420, 2023.

Yuelin Zhang, Jiacheng Cen, Jiaqi Han, Zhiqiang Zhang, Jun Zhou, and Wenbing Huang. Improving equivariant graph neural networks on large geometric graphs via virtual nodes learning. In Ruslan Salakhutdinov, Zico Kolter, Katherine Heller, Adrian Weller, Nuria Oliver, Jonathan Scarlett, and Felix Berkenkamp (eds.), *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 58662–58679. PMLR, 21–27 Jul 2024. URL `https://proceedings.mlr.press/v235/zhang24f.html`.

M. Zhdanov, D. Ruhe, M. Weiler, A. Lucic, J. Brandstetter, and P. Forré. Clifford-steerable convolutional neural networks. volume 235 of *PMLR*, 2024.

Maksim Zhdanov. Flash clifford: Hardware-efficient implementation of Clifford algebra neural networks, 2025. URL `https://github.com/maxxxzdn/flash-clifford`.

APPENDIX

A Contents of appendix **16**

B Summary of notation **16**

C Roadmap of main definitions **16**

D Advantages of the GA approach **16**

E Geometric (Clifford) algebras **19**

E.1 Definition and examples 19

E.2 Subspaces and projections 21

E.3 Adjoint representations 22

E.4 Clifford groups, Lipschitz groups, and spin groups 23

E.5 Groups P, A, B, and Q 25

E.6 New equivariant mappings 27

F Equivariant layers **28**

F.1 GA-linear layers 29

F.2 GA-geometric product layers 29

F.3 GA-normalization layers 31

F.4 Number of parameters 31

G Experimental results and discussion **32**

G.1 $\mathrm{O}(6)$-Convex Hull 32

G.2 $\mathrm{O}(8)$-Regression 34

G.3 $\mathrm{O}(3)$- and $\mathrm{O}(6)$-N-Body Simulation 35

G.4 $\mathrm{O}(1,3)$-Top Tagging 38

G.5 $\mathrm{O}(3)$-Atom Position Prediction 39

G.6 Experiments compute resources 39

H Computational cost trade-off **40**

I Regularization alternatives **40**

J Limitations **42**

A CONTENTS OF APPENDIX

In Appendix B, we provide a comprehensive summary of notation used throughout the paper.

In Appendix C, we present a roadmap of the main definitions for different goals.

In Appendix D, we discuss the advantages of the GA approach over non-GA methods for constructing equivariant networks.

In Appendix E.1, we provide the formal definition of geometric (Clifford) algebras and discuss important examples.

Appendix E.2 discusses the fundamental subspace decompositions of geometric algebras, including grade subspaces, even/odd subspaces, and quaternion type subspaces determined by the grade involution and reversion operations. We prove key properties of projection operations onto these subspaces.

Appendix E.3 studies the adjoint and twisted adjoint representations in geometric algebras, establishing their fundamental properties and characterizing their kernels. These representations play a crucial role in defining equivariant mappings.

Appendix E.4 considers Lipschitz groups, Clifford groups, and spin groups in arbitrary degenerate and non-degenerate geometric algebras, proving several key properties of these representations.

In Appendix E.5, we provide a detailed study of the groups P, A, B, and Q, including the complete proof of Theorem 3.4, which establishes that these groups preserve fundamental subspaces of the geometric algebra under the adjoint action.

Appendix E.6 contains the complete proof of Theorem 3.5, establishing the equivariance properties of projection mappings with respect to the various groups introduced in the paper.

In Appendix F, we provide detailed descriptions of the equivariant layers used in our framework, including implementation specifics and parameter counts for each configuration, and recommendations on the choice of configurations.

In Appendix G, we discuss the experimental results for the experiments from Section 6 and additional results not included in the main text due to pages constraints.

Appendix H compares the computational cost of our equivariant configurations.

In Appendix I, we answer a natural question whether the benefit of using wider equivariance groups can be matched by simply adding traditional regularizers to a less constrained model.

In Appendix J, we provide limitations of our work.

B SUMMARY OF NOTATION

In Table 3, we provide an overview of notation used throughout the paper. We write out the notation, its description, and the place where it is mentioned for the first time in this paper.

C ROADMAP OF MAIN DEFINITIONS

To facilitate understanding of the dense mathematical details underlying our proposed approach, we present a roadmap of decompositions and operations for different goals in Figure 4.

D ADVANTAGES OF THE GA APPROACH

We see several advantages of the GA approach over non-GA methods for constructing equivariant networks.

First, GA extends classical instruments for equivariant layer construction. GA admits the use of new layer operations naturally defined for inputs as multivectors (elements of GA) and not defined

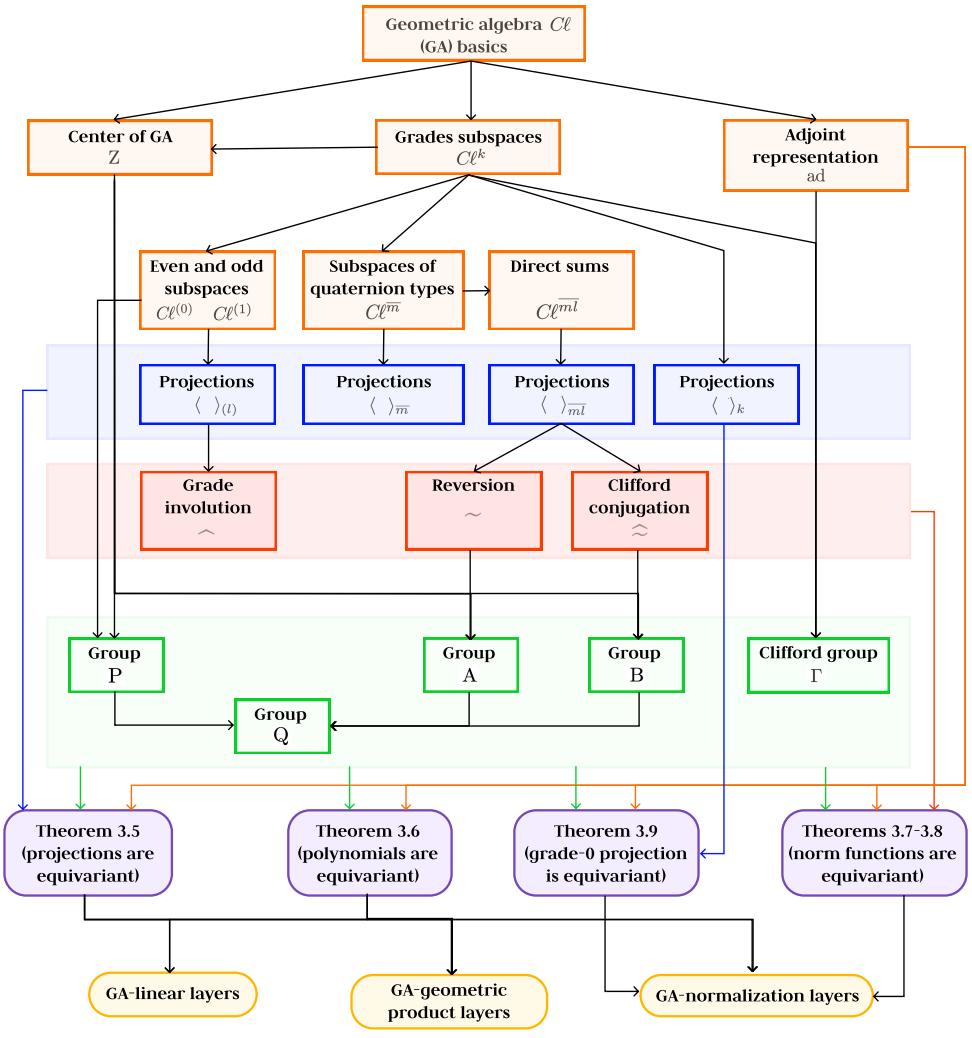


Figure 4: The roadmap of decompositions and operations for different goals.

Table 3: Summary of main notation used throughout the paper

NOTATION	DESCRIPTION	FIRST MENTION
$C\ell = C\ell_{p,q,r}$	Geometric (Clifford) algebra of signature (p, q, r)	Sec. 2
$V = \mathbb{R}^{p,q,r}$	Underlying vector space with quadratic form $\mathfrak{q}$	Sec. 2
$\mathfrak{q}$	Quadratic form on V	Sec. 2
η	Matrix of $\mathfrak{q}$ in normal form	Sec. 2
$\mathrm{O}(V, \mathfrak{q}), \mathrm{SO}(V, \mathfrak{q})$	(Special) pseudo-orthogonal group	Sec. 2
$e_1, \ldots, e_n$	Orthogonal basis vectors (generators) of $C\ell$	Sec. 2
$e_{a_1 \ldots a_k}$	Basis k-vector: $e_{a_1} \cdots e_{a_k}$ for $a_1 < \cdots < a_k$	Sec. 2
$C\ell^k$	Grade-k subspace (k-vectors)	Sec. 2
$C\ell^{(0)}, C\ell^{(1)}$	Even and odd subspaces	Sec. 2
$C\ell^{\overline{m}}$	Quaternion type subspace ($m = 0, 1, 2, 3$)	Sec. 2
$C\ell^{\overline{ml}}$	Direct sum of quaternion type subspaces	Sec. 2
$\langle U \rangle_k$	Projection onto grade-k subspace	Eq. (2)
$\langle U \rangle_{(s)}$	Projection onto even/odd subspace ($s = 0, 1$)	Eq. (2)
$\langle U \rangle_{\overline{m}}$	Projection onto quaternion type subspace	Eq. (2)
$\widehat{U}$	Grade involution	Eq. (3)
$\widetilde{U}$	Reversion	Eq. (3)
$\widehat{\widetilde{U}}$	Clifford conjugation	Eq. (3)
Γ	Clifford group	Eq. (4)
P	Group preserving even/odd structure	Eq. (9)
A	Group defined by reversion norm	Eq. (9)
B	Group defined by Clifford conjugation norm	Eq. (9)
Q	Intersection $\mathrm{A} \cap \mathrm{B}$	Eq. (9)
$\mathrm{ad}_T(U) = TUT^{-1}$	Adjoint action	Eq. (4)
$\tilde{\mathrm{ad}}_T(U)$	Twisted adjoint action	Eq. (36)
Z	Center of the algebra $C\ell$	Eq. (5)
Λ	Grassmann (exterior) subalgebra	Sec. 2
W^k	Family of groups preserving grade subspaces	Eq. (62)
$\mathrm{W}^{\mathrm{rev}}, \mathrm{W}^{\mathrm{conj}}$	Groups defined by reversion/Clifford conjugation	Eq. (63)
S	Set of subspace indices for a given group	Eq. (10)
GALinear	GA-linear layer	Eq. (11)
GAProduct	GA-geometric product layer	Eq. (12)
GANormalization	GA-normalization layer	Eq. (13)
config	Model configuration (set of equivariant layers)	Sec. 5

for matrices, for example, grade involution, reversion, projections onto grade subspaces and their direct sums.

Second, GA allows us to naturally embed input data representing distinct geometric or physical objects and work with them in one algebra, as opposed to scalarization approaches. Data can be embedded into GA as a linear combination of scalars, vectors (for example, coordinates, velocities), bivectors (representing areas with orientations), trivectors (volumes with orientations), pseudoscalars (in physics problems), and so on (see details in the paper). In GA, we can sum them up, multiply via the geometric product, and get geometrically meaningful results. For example, the product of two orthogonal vectors given as input to a neural network is forced to be an oriented plane by applying a geometric product layer. Another point is that GA-based neural networks directly transform data in a vector basis (as scalarization methods) instead of operating on alternative basis representations such as spherical harmonics used in irreps approaches.

Third, the GA approach is the same for any pseudo-orthogonal group $\mathrm{O}(p, q)$. This advantage comes from the well-known fact that spin groups (two-sheeted coverings of $\mathrm{O}(p, q)$) are defined in GA for any p, q in a unified way Atiyah et al. (1964). Non-GA approaches (for example, based on irreps or tensor products) often require case-by-case construction for each p, q (for example, using Clebsch-Gordan coefficients, which are not trivial to obtain).

E GEOMETRIC (CLIFFORD) ALGEBRAS

E.1 DEFINITION AND EXAMPLES

There exist several (equivalent) definition of geometric (Clifford) algebra (for example, Lounesto (1997) provides five different definitions). Let us consider the definition of Clifford algebra that is the closest one to the one given by William Clifford.

Let E be a vector space over the field $\mathbb{F}$ of real numbers $\mathbb{R}$ or complex numbers $\mathbb{C}$. Let n be a natural number, and $\dim E = 2^n$. Suppose E has a basis:

$$e, e_a, e_{a_1 a_2}, \ldots, e_{1\ldots n}, \quad \text{where } a_1 < a_2 < \cdots \quad (\text{total of } 2^n \text{ elements}),$$

indexed by ordered multi-indices of length from 0 to n. Indices $a, a_1, a_2, \ldots$ range from 1 to n.

Let p, q, and r be non-negative integers such that $p + q + r = n$. Define an $n \times n$ diagonal matrix:

$$\eta = (\eta_{ab}) = \operatorname{diag}(\underbrace{1, \ldots, 1}_{p \text{ times}}, \underbrace{-1, \ldots, -1}_{q \text{ times}}, \underbrace{0, \ldots, 0}_{r \text{ times}}).$$

Define *Clifford (or geometric) multiplication* UT on E by:

1. Distributivity: For all $U, T, W \in E$ and $\alpha, \beta \in \mathbb{F}$:
$$U(\alpha T + \beta W) = \alpha UT + \beta UW, \quad (\alpha U + \beta T)W = \alpha UW + \beta TW.$$
2. Associativity: For all $U, T, W \in E$:
$$(UT)W = U(TW).$$
3. Unitality: For all $U \in E$:
$$Ue = eU = U.$$
4. Fundamental relation: For all $a, b = 1, \ldots, n$:
$$e_a e_b + e_b e_a = 2\eta_{ab} e. \tag{14}$$
5. Basis multiplication: For all $1 \le a_1 < \cdots < a_k \le n$:
$$e_{a_1} \cdots e_{a_k} = e_{a_1 \ldots a_k}.$$

The algebra defined this way is called *Clifford algebra* and denoted as $C\ell(V) = C\ell(\mathbb{R}^{p,q,r})$ in the real case $\mathbb{F} = \mathbb{R}$ and $C\ell(V) = C\ell(\mathbb{C}^{p+q,0,r})$ in the complex case $\mathbb{F} = \mathbb{C}$. When considering both cases, we denote it by $C\ell_{p,q,r} = C\ell$. By conditions 1)–4), we obtain an associative, non-commutative, unital algebra with defining relations (14). The elements e_a are called the *generators* of the Clifford algebra, and the element e is called the *identity element* of the Clifford algebra. The numbers (p, q, r) are called the *signature* of the Clifford algebra $C\ell_{p,q,r}$.

Let us consider several examples of Clifford algebras of small dimensions.

We have the following algebra isomorphisms:

$$C\ell(\mathbb{R}^{1,0,0}) \simeq \mathbb{R} \oplus \mathbb{R}, \qquad C\ell(\mathbb{R}^{0,1,0}) \simeq \mathbb{C}.$$

For elements of the Clifford algebra $C\ell(\mathbb{R}^{2,0,0})$, we can construct a matrix representation by associating the basis elements with the following matrices:

$$e \mapsto \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}, \quad e_1 \mapsto \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}, \quad e_2 \mapsto \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}, \quad e_{12} \mapsto \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix}.$$

We have the following isomorphisms:

$$C\ell(\mathbb{R}^{2,0,0}) \simeq C\ell(\mathbb{R}^{1,1,0}) \simeq \operatorname{Mat}(2, \mathbb{R}),$$

where $\operatorname{Mat}(k, \mathbb{F})$ denotes the algebra of $k \times k$ matrices over the field $\mathbb{F}$.

Let us consider the Clifford algebra $C\ell(\mathbb{R}^{0,2,0})$. For the basis elements we have:

$$(e_1)^2 = -e, \quad (e_2)^2 = -e, \quad (e_{12})^2 = -e,$$
$$e_1e_2 = -e_2e_1 = e_{12}, \quad e_{12}e_1 = -e_1e_{12} = e_2, \quad e_2e_{12} = -e_{12}e_2 = e_1.$$

Thus, by defining the correspondence:

$$e_1 \mapsto i, \quad e_2 \mapsto j, \quad e_{12} \mapsto k,$$

we find that the Clifford algebra $C\ell(\mathbb{R}^{0,2,0})$ is isomorphic to the algebra (division ring) of quaternions $\mathbb{H}$. The quaternion algebra is an associative algebra with unit 1. Elements of $\mathbb{H}$ (quaternions) are written as:

$$q = a + bi + cj + dk,$$

where $a, b, c, d \in \mathbb{R}$ and the imaginary units i, j, k satisfy:

$$i^2 = j^2 = k^2 = -1,$$
$$ij = -ji = k, \quad jk = -kj = i, \quad ki = -ik = j.$$

The considered examples show that Clifford algebras can be viewed as generalizations of the quaternions and complex numbers.

Note that in physics and mechanics, the Clifford algebras $C\ell_{1,3}$ and $C\ell_3$ find the greatest application. In the first case, the generator e_1 corresponds to time, while the generators e_2, e_3, e_4 correspond to the three spatial coordinates. In the Clifford algebra $C\ell_3$, all three generators correspond to spatial coordinates.

For $C\ell(\mathbb{R}^{3,0,0})$, the matrix representation is usually given using the Pauli matrices:

$$\sigma_1 = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}, \quad \sigma_2 = \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix}, \quad \sigma_3 = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}$$

as follows:

$$e \mapsto \sigma_0 = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}, \quad e_1 \mapsto \sigma_1, \quad e_2 \mapsto \sigma_2, \quad e_3 \mapsto \sigma_3.$$

The product of generators maps to the matrix obtained by multiplying the matrices corresponding to these generators. Due to the relations:

$$\sigma_1\sigma_2 = i\sigma_3, \quad \sigma_2\sigma_3 = i\sigma_1, \quad \sigma_3\sigma_1 = i\sigma_2,$$

we find that the basis elements of the Clifford algebra $C\ell(\mathbb{R}^{3,0,0})$ are assigned the following matrices:

$$\sigma_0, \sigma_1, \sigma_2, \sigma_3, i\sigma_1, i\sigma_2, i\sigma_3, i\sigma_0,$$

which form a basis in $\mathrm{Mat}(2, \mathbb{C})$. We have the isomorphism:

$$C\ell(\mathbb{R}^{3,0,0}) \simeq \mathrm{Mat}(2, \mathbb{C}).$$

Recall that the Pauli matrices $\sigma_1, \sigma_2, \sigma_3$ form a basis in the space of all 2×2 Hermitian matrices with zero trace. They were introduced by Wolfgang Pauli to describe electron spin in quantum mechanics. The following relations hold:

$$(\sigma_i)^\dagger = \sigma_i, \qquad \mathrm{tr}(\sigma_i) = 0, \qquad (\sigma_i)^2 = \sigma_0, \qquad i = 1, 2, 3,$$
$$\sigma_i\sigma_j = -\sigma_j\sigma_i, \qquad i \neq j, \qquad i, j = 1, 2, 3,$$

where † denotes the Hermitian conjugation operation, which is the composition of transposition and complex conjugation of all matrix elements.

A matrix representation for $C\ell(\mathbb{C}^{1,3,0})$ can be obtained using the 4×4 Dirac matrices:

$$\gamma^0 = \begin{pmatrix} I_2 & 0 \\ 0 & -I_2 \end{pmatrix}, \quad \gamma^i = \begin{pmatrix} 0 & \sigma_i \\ -\sigma_i & 0 \end{pmatrix}, \quad i = 1, 2, 3,$$

where σ_i are the Pauli matrices and I_2 is the 2×2 identity matrix. The representation is given as follows:

$$e \mapsto I_4, \quad e_i \mapsto \gamma^{i-1}, \quad i = 1, 2, 3,$$

where I_4 is the 4×4 identity matrix. The Dirac matrices, also known as gamma-matrices, were used by Dirac in formulating the equation for the electron. They satisfy the following relations:

$$\begin{aligned}
&(\gamma^0)^2 = I_4, \qquad (\gamma^i)^2 = -I_4, && i = 1,2,3,\\
&\mathrm{tr}(\gamma^0) = \mathrm{tr}(\gamma^i) = 0, && i = 1,2,3,\\
&\{\gamma^i, \gamma^j\} = \gamma^i\gamma^j + \gamma^i\gamma^j = 2\eta_{ij}1_4, && i,j = 0,1,2,3,\\
&(\gamma^i)^\dagger = \gamma^0\gamma^i\gamma^0, && i = 0,1,2,3,
\end{aligned}$$

where η_{ij} are the elements of the diagonal matrix $\eta = \mathrm{diag}(1,-1,-1,-1)$, which is also called *Minkowski metric*. We have the isomorphism:

$$C\ell(\mathbb{C}^{1,3,0}) \simeq \mathrm{Mat}(4,\mathbb{C}).$$

We use this particular Clifford algebra in the top-tagging experiment (see Section 6).

E.2 Subspaces and projections

Let us consider the fixed *grade subspaces* $C\ell^k$, $k = 0,1,\ldots,n$, which elements are linear combinations of basis elements $e_{a_1\ldots a_k}$ with ordered multi-indices of length k. We also consider the even $C\ell^{(0)}$ and odd $C\ell^{(1)}$ subspaces:

$$C\ell^{(0)} := C\ell^0 \oplus C\ell^2 \oplus C\ell^4 \oplus \cdots, \quad C\ell^{(1)} := C\ell^1 \oplus C\ell^3 \oplus C\ell^5 \oplus \cdots \tag{15}$$

Note that

$$C\ell^{(k)}C\ell^{(l)} \subset C\ell^{(k+l) \bmod 2}, \qquad k,l = 0,1. \tag{16}$$

We also consider the *quaternion types subspaces* $C\ell^{\overline{m}}$ (Shirokov, 2009; 2012a;b):

$$C\ell^{\overline{0}} := C\ell^0 \oplus C\ell^4 \oplus C\ell^8 \oplus \cdots, \quad C\ell^{\overline{1}} := C\ell^1 \oplus C\ell^5 \oplus C\ell^9 \oplus \cdots, \tag{17}$$

$$C\ell^{\overline{2}} := C\ell^2 \oplus C\ell^6 \oplus C\ell^1 0 \oplus \cdots, \quad C\ell^{\overline{3}} := C\ell^3 \oplus C\ell^7 \oplus C\ell^1 1 \oplus \cdots \tag{18}$$

and their direct sums $C\ell^{\overline{01}} := C\ell^{\overline{0}} \oplus C\ell^{\overline{1}}$, $C\ell^{\overline{12}} := C\ell^{\overline{1}} \oplus C\ell^{\overline{2}}$, $C\ell^{\overline{23}} := C\ell^{\overline{2}} \oplus C\ell^{\overline{3}}$, and $C\ell^{\overline{03}} := C\ell^{\overline{0}} \oplus C\ell^{\overline{3}}$. Note that $C\ell^{\overline{02}} = C\ell^{(0)}$ and $C\ell^{\overline{13}} = C\ell^{(1)}$.

Let us consider the projection operation $U \mapsto \langle U\rangle_k \in C\ell^k$, defined for any $U \in C\ell$. For example, $\langle U\rangle_0 = \langle ue + u_1e_1 + \cdots + u_{12}e_{12} + \cdots + u_{1\ldots n}e_{1\ldots n}\rangle_0 = ue$ for any $U \in C\ell$, $u,\ldots,u_{1\ldots n} \in \mathbb{F}$. We also use projections $\langle\cdot\rangle_{(0)}$, $\langle\cdot\rangle_{(1)}$ onto the even $C\ell^{(0)}$ and odd $C\ell^{(1)}$ subspaces; projections $\langle\cdot\rangle_{\overline{m}}$ onto the quaternion types subspaces $C\ell^{\overline{m}}$, $m = 0,1,2,3$; projections $\langle\cdot\rangle_{\overline{ml}}$ onto their direct sums $C\ell^{\overline{ml}}$, $m,l = 0,1,2,3$. Any multivector $U \in C\ell$ can be decomposed as:

$$U = \langle U\rangle_0 + \langle U\rangle_1 + \cdots + \langle U\rangle_n, \quad \langle U\rangle_k \in C\ell^k, \quad k = 0,\ldots,n; \tag{19}$$

$$U = \langle U\rangle_{(0)} + \langle U\rangle_{(1)}, \quad \langle U\rangle_{(0)} \in C\ell^{(0)}, \quad \langle U\rangle_{(1)} \in C\ell^{(1)}; \tag{20}$$

$$U = \langle U\rangle_{\overline{0}} + \langle U\rangle_{\overline{1}} + \langle U\rangle_{\overline{2}} + \langle U\rangle_{\overline{3}}, \quad \langle U\rangle_{\overline{m}} \in C\ell^{\overline{m}}, \quad m = 0,1,2,3; \tag{21}$$

$$U = \langle U\rangle_{\overline{01}} + \langle U\rangle_{\overline{23}}, \quad U = \langle U\rangle_{\overline{12}} + \langle U\rangle_{\overline{03}}, \quad \langle U\rangle_{\overline{ml}} \in C\ell^{\overline{ml}}, \quad m,l = 0,1,2,3. \tag{22}$$

The operations of projections satisfy the following properties:

Lemma E.1 *The projection $\langle\cdot\rangle_k$, $k = 0,\ldots,n$, satisfies linearity:*

$$\langle U+V\rangle_k = \langle U\rangle_k + \langle V\rangle_k; \qquad \langle\alpha U\rangle_k = \alpha\langle U\rangle_k, \qquad \alpha \in \mathbb{F}. \tag{23}$$

The projection $\langle\cdot\rangle_0$ satisfies:

$$\langle UV\rangle_0 = \langle VU\rangle_0, \qquad \forall U,V \in C\ell; \tag{24}$$

$$\langle UVW\rangle_0 = \langle WVU\rangle_0, \qquad \forall U,V,W \in C\ell; \tag{25}$$

$$\langle TUT^{-1}\rangle_0 = \langle U\rangle_0, \qquad \forall U \in C\ell, \quad \forall T \in C\ell^\times. \tag{26}$$

Proof. Linearity (23) follows directly from the definition of the projection.

Let us prove (24). Suppose $U = ue + u_1e_1 + \cdots + u_{12}e_{12} + \cdots + u_{1\ldots n}e_{1\ldots n}$ and $V = ve + v_1e_1 + \cdots + v_{12}e_{12} + \cdots + v_{1\ldots n}e_{1\ldots n}$ are two arbitrary multivectors, and $u, \ldots, u_{1\ldots n}, v, \ldots, v_{1\ldots n} \in \mathbb{F}$. Then

$$\langle UV\rangle_0 = uve + u_1e_1v_1e_1 + \cdots + u_{12}e_{12}v_{12}e_{12} + \cdots u_{1\ldots n}e_{1\ldots n}v_{1\ldots n}e_{1\ldots n} = \langle VU\rangle_0. \tag{27}$$

We get (25) as a corollary of (24), and further obtain (26) as a corollary of (25). □

Note that Lemma E.1 implies that all the other projections $\langle\cdot\rangle_{(s)}$, $s = 0, 1$; $\langle\cdot\rangle_{\overline{m}}$, $m = 0, 1, 2, 3$; $\langle\cdot\rangle_{\overline{ml}}$, $m, l = 0, 1, 2, 3$ satisfy linearity as well by definition.

The subspaces $C\ell^{(s)}$, $s = 0, 1$; $C\ell^{\overline{m}}$, $m = 0, 1, 2, 3$; $C\ell^{\overline{ml}}$, $m, l = 0, 1, 2, 3$, can be equivalently defined using the following conjugation operations in $C\ell$. We consider the *grade involution* $\widehat{\ }$, the reversion $\widetilde{\ }$, and the *Clifford conjugation* $\widehat{\widetilde{\ }}$:

$$\widehat{U} := \sum_{k=0}^{n}(-1)^k\langle U\rangle_k = \langle U\rangle_{(0)} - \langle U\rangle_{(1)}, \tag{28}$$

$$\widetilde{U} := \sum_{k=0}^{n}(-1)^{\frac{k(k-1)}{2}}\langle U\rangle_k = \langle U\rangle_{\overline{01}} - \langle U\rangle_{\overline{23}}, \tag{29}$$

$$\widehat{\widetilde{U}} := \sum_{k=0}^{n}(-1)^{\frac{k(k+1)}{2}}\langle U\rangle_k = \langle U\rangle_{\overline{03}} - \langle U\rangle_{\overline{12}}. \tag{30}$$

Note that $\widehat{UV} = \widehat{U}\widehat{V}$ and $\widetilde{UV} = \widetilde{V}\widetilde{U}$ for any $U, V \in C\ell$.

Equivalently, the even and odd subspaces can be defined via grade involution:

$$C\ell^{(s)} = \{U \in C\ell : \quad \widehat{U} = (-1)^sU\}, \qquad s = 0, 1;$$

subspaces of quaternion types can be defined via grade involution and reversion:

$$C\ell^{\overline{m}} = \{U \in C\ell : \widehat{U} = (-1)^mU,\ \widetilde{U} = (-1)^{\frac{m(m-1)}{2}}U\}, \quad m = 0, 1, 2, 3;$$

and their direct sums can be defined either by reversion:

$$C\ell^{\overline{01}} = \{U \in C\ell : \widetilde{U} = U\}, \qquad C\ell^{\overline{23}} = \{U \in C\ell : \widetilde{U} = -U\},$$

or by Clifford conjugation:

$$C\ell^{\overline{03}} = \{U \in C\ell : \widehat{\widetilde{U}} = U\}, \qquad C\ell^{\overline{12}} = \{U \in C\ell : \widehat{\widetilde{U}} = -U\}.$$

In GA, there are no involutions or anti-involutions of the form of linear combinations of grade projections other than $\widehat{\ }$, $\widetilde{\ }$, and $\widehat{\widetilde{\ }}$ (we have $\widehat{\widehat{U}} = \widehat{U}\widehat{V}$, $\widetilde{\widetilde{U}} = \widetilde{V}\widetilde{U}$, and $\widehat{\widetilde{\widehat{U}}} = \widehat{\widetilde{V}}\widehat{\widetilde{U}}$ for any $U, V \in C\ell$).

E.3 Adjoint representations

Let us consider the *adjoint representation* ad is the acting on the group of all invertible elements $\mathrm{ad} : C\ell^\times \to \mathrm{Aut}(C\ell)$ as $T \mapsto \mathrm{ad}_T : C\ell \to C\ell$, where

$$\mathrm{ad}_T(U) := TUT^{-1} \tag{31}$$

acts as a similarity transformation for any $U \in C\ell$. It has the following important properties:

Lemma E.2 *For any $T \in C\ell^\times$, the adjoint representation satisfies:*

1. *linearity:* $\mathrm{ad}_T(U + V) = \mathrm{ad}_T(U) + \mathrm{ad}_T(V)$ *and* $\mathrm{ad}_T(\alpha U) = \alpha\mathrm{ad}_T(U)$ *for any $U, V \in C\ell$, $\alpha \in \mathbb{F}$;*
2. *scalar invariance:* $\mathrm{ad}_T(\alpha) = \alpha$ *for any $\alpha \in \mathbb{F}$;*
3. *multiplicativity:* $\mathrm{ad}_T(UV) = \mathrm{ad}_TU\mathrm{ad}_T(V)$ *for any $U, V \in C\ell$.*

Proof. Suppose $U, V \in C\ell$, $\alpha \in \mathbb{F}$, and $T \in C\ell^{\times}$, then

$$\mathrm{ad}_T(U+V) = T(U+V)T^{-1} = TUT^{-1} + TVT^{-1} = \mathrm{ad}_T(U) + \mathrm{ad}_T(V), \tag{32}$$

$$\mathrm{ad}_T(\alpha U) = T(\alpha U)T^{-1} = \alpha TUT^{-1} = \alpha \mathrm{ad}_T(U), \tag{33}$$

$$\mathrm{ad}_T(\alpha) = T\alpha T^{-1} = \alpha, \tag{34}$$

$$\mathrm{ad}_T(UV) = T(UV)T^{-1} = TUT^{-1}TVT^{-1} = \mathrm{ad}_T(U)\mathrm{ad}_T(V), \tag{35}$$

and the proof is completed. □

We also need the *twisted adjoint representation* $\tilde{\mathrm{ad}}$ (see, for example, Helmstetter & Micali (2008); Knus (1991); Walpuski (2022)), which is the operation acting on the group of all invertible grade-m elements $\tilde{\mathrm{ad}} : C\ell^{m\times} \to \mathrm{Aut}(C\ell)$ as $T \mapsto \tilde{\mathrm{ad}}_T$ with $\tilde{\mathrm{ad}}_T : C\ell \to C\ell$ defined as:

$$\tilde{\mathrm{ad}}_T(U) = (-1)^{km} TUT^{-1}, \qquad U \in C\ell^k, \qquad T \in C\ell^{m\times}. \tag{36}$$

The operation $\tilde{\mathrm{ad}}_T$ can be extended by linearity $\tilde{\mathrm{ad}}_T(U+V) = \tilde{\mathrm{ad}}_T(U) + \tilde{\mathrm{ad}}_T(V)$, and we get

$$\tilde{\mathrm{ad}}_T(U) = T\langle U\rangle_{(0)}T^{-1} + \widehat{T}\langle U\rangle_{(1)}T^{-1}, \qquad \forall U \in C\ell, \qquad T \in C\ell^{\times}. \tag{37}$$

Consider the *kernels of the adjoint and the twisted adjoint representations*:

$$\ker(\mathrm{ad}) = \{T \in C\ell^{\times} : \quad TUT^{-1} = U, \quad \forall U \in C\ell\},$$

$$\ker(\tilde{\mathrm{ad}}) = \{T \in C\ell^{\times} : \ T\langle U\rangle_{(0)}T^{-1} + \widehat{T}\langle U\rangle_{(1)}T^{-1} = U, \qquad \forall U \in C\ell\}.$$

Lemma E.3 *(Filimoshina & Shirokov, 2025; Ruhe et al., 2023a; Filimoshina & Shirokov, 2023) The kernels of* ad *and* $\tilde{\mathrm{ad}}$ *have the form:*

$$\ker(\mathrm{ad}) = \mathrm{Z}^{\times} = \begin{cases} (\Lambda^{(0)} \oplus C\ell^n)^{\times} & \text{if } n \text{ is odd}, \\ \Lambda^{(0)\times} & \text{if } n \text{ is even}; \end{cases} \tag{38}$$

$$\ker(\tilde{\mathrm{ad}}) = \Lambda^{\times}. \tag{39}$$

We also use the following lemma:

Lemma E.4 (Lemma D.3 (Filimoshina & Shirokov, 2025)) *We have*

$$\{X \in C\ell : \quad XV = VX, \quad \forall V \in C\ell^{(0)}\} = \begin{cases} \Lambda \oplus C\ell^n, & r \neq n, \\ \Lambda, & r = n, \end{cases}$$

$$\{X \in C\ell : \quad \widehat{X}V = VX, \quad \forall V \in C\ell^{(0)}\} = \begin{cases} \Lambda^{(0)}, & n \text{ is odd}; \ r = n \text{ is even}, \\ \Lambda^{(0)} \oplus C\ell^n, & n \text{ is even}, \ r \neq n. \end{cases}$$

E.4 Clifford groups, Lipschitz groups, and spin groups

Let us consider the *Clifford group* Γ and the Lipschitz group $\tilde{\Gamma}$ (see, for example, Lounesto (1997); Crumeyrolle (1990)), which are the groups of all invertible elements preserving the subspace $C\ell^1$ of vectors under the adjoint ad and twisted adjoint $\tilde{\mathrm{ad}}$ representations respectively:

$$\Gamma := \{T \in C\ell^{\times} : \quad \mathrm{ad}_T(C\ell^1) := TC\ell^1T^{-1} = C\ell^1\}, \tag{40}$$

$$\tilde{\Gamma} := \{T \in C\ell^{\times} : \quad \tilde{\mathrm{ad}}_T(C\ell^1) := \widehat{T}C\ell^1T^{-1} = C\ell^1\}. \tag{41}$$

The well-known spin groups (see, for example, Lounesto (1997); Crumeyrolle (1990); Ablamowicz (1986); Brooke (1980)) are defined as normalized subgroups of the Lipschitz group $\tilde{\Gamma}$ and its even subgroup $\tilde{\Gamma} \cap C\ell^{(0)\times}$. For example, in the non-degenerate geometric algebras $C\ell_{p,q}$, they are defined in the following way:

$$\mathrm{Pin}_{p,q} := \{T \in \tilde{\Gamma} : \quad \widetilde{T}T = \pm e\} = \{T \in \tilde{\Gamma}^1_{p,q} : \quad \widehat{\widetilde{T}}T = \pm e\}, \tag{42}$$

$$\mathrm{Pin}^{\mathrm{rev}}_{p,q} := \{T \in \tilde{\Gamma} : \quad \widetilde{T}T = +e\}, \tag{43}$$

$$\mathrm{Pin}^{\mathrm{conj}}_{p,q} := \{T \in \tilde{\Gamma} : \quad \widehat{\widetilde{T}}T = +e\}, \tag{44}$$

$$\mathrm{Spin}_{p,q} := \{T \in \tilde{\Gamma} \cap C\ell^{(0)\times} : \quad \widetilde{T}T = \pm e\} = \{T \in \tilde{\Gamma} \cap C\ell^{(0)\times}_{p,q} : \quad \widehat{\widetilde{T}}T = \pm e\}, \tag{45}$$

$$\mathrm{Spin}^{+}_{p,q} := \{T \in \tilde{\Gamma} \cap C\ell^{(0)\times} : \quad \widetilde{T}T = +e\} = \{T \in \tilde{\Gamma} \cap C\ell^{(0)\times} : \quad \widehat{\widetilde{T}}T = +e\}. \tag{46}$$

Note that the Clifford group Γ and the Lipschitz group $\tilde{\Gamma}$ can be equivalently defined (see, for example, Crumeyrolle (1990)):

$$\begin{aligned}\tilde{\Gamma} &= \{v_1\cdots v_s\gamma_{1,1}\cdots\gamma_{r,p+q}\lambda, \quad v_l\in C\ell^{1\times}_{p,q}, \quad \gamma_{i,j}=e+m_{i,j}e_je_{i+p+q} &(47)\\ &\quad m_{i,j}\in\mathbb{R}, \quad i=1,\dots,r, \quad j=1,\dots,p+q, \quad \lambda\in\Lambda^{\times}_r\}, &(48)\\ \Gamma &= \{v_1\cdots v_s\gamma_{1,1}\cdots\gamma_{r,p+q}z, \quad v_l\in C\ell^{1\times}_{p,q}, \quad \gamma_{i,j}=e+m_{i,j}e_je_{i+p+q} &(49)\\ &\quad m_{i,j}\in\mathbb{R}, \quad i=1,\dots,r, \quad j=1,\dots,p+q, \quad z\in \mathrm{Z}^{\times}_{p,q,r}\}. &(50)\end{aligned}$$

In other words, any element of the Lipschitz group $\tilde{\Gamma}$ can be represented as a product of the invertible vectors $v_1,\dots,v_s$ from the non-degenerate subalgebra $C\ell_{p,q}=C\ell_{p,q,0}\subseteq C\ell_{p,q,r}$; elements of the form $e+m_{i,j}e_je_{i+p+q}$, where the bivector part consists of one generator that squares to either $+e$, or $-e$ and one generator that squares to 0; and one invertible element λ from the Grassmann subalgebra. We get the similar decomposition for the elements of the Clifford group Γ, but we add one additional term z that is in the center $\mathrm{Z}_{p,q,r}$ of $C\ell_{p,q,r}$.

Note that in the case of the non-degenerate geometric algebra $C\ell_{p,q}$, the equivalent definitions (47) and (49) have much simpler form:

$$\begin{aligned}\tilde{\Gamma}^1_{p,q} &= \{v_1\cdots v_s, \qquad v_1,\dots,v_s\in C\ell^{1\times}_{p,q}\}, &(51)\\ \Gamma^1_{p,q} &= \{v_1\cdots v_sz, \qquad v_1,\dots,v_s\in C\ell^{1\times}_{p,q}, \qquad z\in \mathrm{Z}^{\times}_{p,q}\}. &(52)\end{aligned}$$

We can consider the following analogues Γ^k of the Clifford group Γ, which preserve the subspaces $C\ell^k$, $k=0,\dots,n$, under the adjoint representation ad:

$$\Gamma^k:=\{T\in C\ell^{\times}: \quad \mathrm{ad}_T(C\ell^k):=TC\ell^kT^{-1}=C\ell^k\}, \qquad k=0,\dots,n. \tag{53}$$

We have $\Gamma^1=\Gamma$. The Clifford group Γ is related to the groups Γ^k in the following way:

Theorem E.5 *In any degenerate and non-degenerate $C\ell$, the Clifford group Γ is a subgroup of all the groups Γ^k:*

$$\Gamma\subseteq\Gamma^k, \qquad k=0,\dots,n. \tag{54}$$

Consequently, $\Gamma=\bigcap_{k=0}^n\Gamma^k=\{T\in C\ell^{\times}: \quad TC\ell^kT^{-1}=C\ell^k, \quad \forall k=0,\dots,n\}$.

Proof. Suppose $T=v_1\cdots v_s\gamma_{1,1}\cdots\gamma_{r,p+q}z\in\Gamma^1$, where v_l, $\gamma_{i,j}$, and z are of the form as in the equivalent definition (49)–(50) of the Clifford group. Suppose $U_k\in C\ell^k$ for some fixed $k=0,\dots,n$; we need to prove $\mathrm{ad}_T(U_k)=C\ell^k$. We have

$$\begin{aligned}\mathrm{ad}_T(U_k) &= TU_kT^{-1} &(55)\\ &= v_1\cdots v_s\gamma_{1,1}\cdots\gamma_{r,p+q}zU_kz^{-1}\gamma^{-1}_{r,p+q}\cdots\gamma^{-1}_{1,1}v_s^{-1}\cdots v_1^{-1}. &(56)\end{aligned}$$

Since $z\in\mathrm{Z}^{\times}$, we have $zU_kz^{-1}=U_k$. Since $\widehat{v_l}=-v_l$ for any $l=1,\dots,s$, we have $\widehat{v_1}\cdots\widehat{v_s}=\pm v_1\cdots v_s$. Since all $\gamma_{i,j}\in(C\ell^0\oplus C\ell^2)^{\times}$ are even, we get $\widehat{\gamma_{i,j}}=\gamma_{i,j}$. Moreover, note that for any $\gamma_{i,j}=e+m_{i,j}e_je_{i+p+q}$, we have $\gamma^{-1}_{i,j}=\widetilde{\gamma_{i,j}}=e-m_{i,j}e_je_{i+p+q}$. For any $v_l\in C\ell^{1\times}_{p,q}$, we have $\widetilde{v_l}=v_l$ and $v_l^{-1}=\alpha v_l$ for some $\alpha\neq0$ using $v_l^2\in C\ell^{0\times}$. Using these facts, we obtain for any $l=1,\dots,s$, $i=1,\dots,r$, and $j=1,\dots,p+q$:

$$(v_lU_kv_l^{-1})\tilde{}=\widetilde{v_l^{-1}}\widetilde{U_k}\widetilde{v_l}=v_l\widetilde{U_k}v_l^{-1}, \quad (v_lU_kv_l^{-1})\hat{}=\widehat{v_l}\widehat{U_k}\widehat{v_l^{-1}}=v_l\widehat{U_k}v_l^{-1}; \tag{57}$$

$$(\gamma_{i,j}U_k\gamma^{-1}_{i,j})\tilde{}=\widetilde{\gamma^{-1}_{i,j}}\widetilde{U_k}\widetilde{\gamma_{i,j}}=\gamma_{i,j}\widetilde{U_k}\gamma^{-1}_{i,j}, \quad (\gamma_{i,j}U_k\gamma^{-1}_{i,j})\hat{}=\widehat{\gamma_{i,j}}\widehat{U_k}\widehat{\gamma^{-1}_{i,j}}=\gamma_{i,j}\widehat{U_k}\gamma^{-1}_{i,j}. \tag{58}$$

From (57) and (58), we get that the elements U_k, $v_lU_kv_l^{-1}$, and $\gamma_{i,j}U_k\gamma^{-1}_{i,j}$ are in the same subspace $C\ell^{\overline{w}}_{p,q,r}$, where $w=k \bmod 4$. Note that any $\gamma_{i,j}$ can be represented as the product of two vectors $\gamma_{i,j}=e_j(e_j^{-1}-m_{i,j}e_{i+p+q})$, where each of the two multipliers lies in $C\ell^1$. The grades of the elements $v_lU_kv_l^{-1}$, $e_jU_ke_j^{-1}$, and $(e_j^{-1}-m_{i,j}e_{i+p+q})U_k(e_j^{-1}-m_{i,j}e_{i+p+q})^{-1}$ take values between $k-2$ and $k+2$, since the left and right multipliers in each of them are vectors. Thus, the grades are equal to k, and we get

$$v_lU_kv_l^{-1}\in C\ell^k_{p,q,r}, \quad e_jU_ke_j^{-1}\in C\ell^k, \tag{59}$$

$$(e_j^{-1}-m_{i,j}e_{i+p+q})U_k(e_j^{-1}-m_{i,j}e_{i+p+q})^{-1}\in C\ell^k. \tag{60}$$

Therefore, substituting this to (56), we finally get $\mathrm{ad}_T(U_k)=C\ell^k$, and the proof is completed. □

Lemma E.6 *The Clifford group is a subgroup of any group defined by the preservation of a direct sum of grade subspaces, where $k_i \in \{0, 1, \dots, n\}$ is any selection of grades:*

$$\Gamma \subseteq \{T \in C\ell^\times : \quad T(\oplus_i C\ell^{k_i})T^{-1} = \oplus_i C\ell^{k_i}\}. \tag{61}$$

Proof. If $T \in \Gamma$, then by Theorem E.5, T preserves each grade-k subspace: $TC\ell^k T^{-1} = C\ell^k$ for all $k = 0, \dots, n$. Thus, T preserves any direct sum of these subspaces. □

E.5 Groups P, A, B, and Q

We are interested in groups containing the Clifford group Γ as a subgroup, since equivariance of a mapping with respect to these groups automatically implies its pseudo-orthogonal groups equivariance (Theorem 3.2). Guided by Lemma E.6, we introduce a family of groups W^k preserving the direct sum $C\ell^1 \oplus C\ell^{1+k} \oplus C\ell^{1+2k} \oplus \cdots$, $k \in \mathbb{N}$, of the grade subspaces with step size k under the adjoint representation ad:

$$\mathrm{W}^k := \{T \in C\ell^\times: \quad T(\bigoplus_{j=1 \bmod k} C\ell^j)T^{-1} = \bigoplus_{j=1 \bmod k} C\ell^j\}. \tag{62}$$

This family yields several important special cases:

$$\begin{aligned}
\mathrm{W}^1 &= \{T \in C\ell^\times : \quad TC\ell^0 T^{-1} = C\ell^0\} = C\ell^\times, \\
\mathrm{W}^2 &= \{T \in C\ell^\times : \quad TC\ell^{(1)} T^{-1} = C\ell^{(1)}\}, \\
\mathrm{W}^4 &= \{T \in C\ell^\times : \quad TC\ell^{\overline{1}} T^{-1} = C\ell^{\overline{1}}\}, \\
\mathrm{W}^n &= \{T \in C\ell^\times : \quad TC\ell^1 T^{-1} = C\ell^1\} = \Gamma.
\end{aligned}$$

Note that $\mathrm{W}^k = \mathrm{W}^n$ for $k \geq n+1$, since $C\ell^k = \varnothing$ in this case. A smaller k corresponds to a larger and less constrained group W^k, which lies 'further' from Γ. This relationship between the index k and the size of the group is central to our method, since it results in the trade-off between parameter efficiency and expressiveness in the resulting W^k-equivariant networks.

In addition to the W^k family, we propose a second approach for constructing groups that contain the Clifford group. We consider the groups preserving the subspaces $C\ell^{\overline{01}}$ and $C\ell^{\overline{12}}$ determined by the fundamental operations in GAs (reversion and Clifford conjugation):

$$\mathrm{W}^{\mathrm{rev}} := \{T \in C\ell^\times : \quad TC\ell^{\overline{01}} T^{-1} = C\ell^{\overline{01}}\}, \tag{63}$$

$$\mathrm{W}^{\mathrm{conj}} := \{T \in C\ell^\times : \quad TC\ell^{\overline{12}} T^{-1} = C\ell^{\overline{12}}\}. \tag{64}$$

The groups W^k, $\mathrm{W}^{\mathrm{rev}}$, and $\mathrm{W}^{\mathrm{conj}}$ can be viewed as natural generalizations of the Clifford group Γ. By Lemma E.6, Γ is a subgroup of each W^k, $\mathrm{W}^{\mathrm{rev}}$, and $\mathrm{W}^{\mathrm{conj}}$:

Lemma E.7 $\Gamma \subseteq \mathrm{W}^{\mathrm{rev}}$, $\Gamma \subseteq \mathrm{W}^{\mathrm{conj}}$, $\Gamma \subseteq \mathrm{W}^k$ *for any* $k = 1, \dots, n-1$.

To facilitate the construction of mappings equivariant to these groups, we now establish equivalent definitions for W^k, $\mathrm{W}^{\mathrm{rev}}$, and $\mathrm{W}^{\mathrm{conj}}$ that avoid similarity transformations. Instead, these new definitions are based on the concepts of norm functions and the center Z (5) of $C\ell$, which significantly simplifies the derivation of equivariant mappings. We begin by introducing the group P, defined as:

$$\mathrm{P} := \mathrm{P}_{p,q,r} := (C\ell^{(0)\times} \cup C\ell^{(1)\times})\mathrm{Z}^\times, \tag{65}$$

which elements have the form UV, where $U \in (C\ell^{(0)\times} \cup C\ell^{(1)\times})$ is even or odd, and $V \in \mathrm{Z}^\times$ is in the center (5). Also we introduce the groups defined using the norm functions of GA elements $\widetilde{T}T$ and $\widehat{\widetilde{T}}T$, for $T \in C\ell$, which are used in the theory of spin groups (Lounesto, 1997; Benn & Tucker, 1987; Crumeyrolle, 1990):

$$\mathrm{A} := \mathrm{A}_{p,q,r} := \{T \in C\ell^\times : \ \widetilde{T}T \in \mathrm{Z}^\times\}, \tag{66}$$

$$\mathrm{B} := \mathrm{B}_{p,q,r} := \{T \in C\ell^\times : \ \widehat{\widetilde{T}}T \in \mathrm{Z}^\times\}. \tag{67}$$

These normalization conditions generalize those used in spin groups definitions (42)–(46). Finally,

$$\mathrm{Q} := \mathrm{Q}_{p,q,r} := \{T \in C\ell^\times : \ \widetilde{T}T \in \mathrm{Z}^\times, \ \widehat{\widetilde{T}}T \in \mathrm{Z}^\times\}, \tag{68}$$

$$\mathrm{Q} = \mathrm{A} \cap \mathrm{B} = \mathrm{P} \cap \mathrm{A} = \mathrm{P} \cap \mathrm{B}. \tag{69}$$

Lemma E.8 *The group* Q *is a subgroup of the group* P*:* $\mathrm{Q} \subseteq \mathrm{P}$.

Proof. Suppose $T \in \mathrm{Q}$; then $\widetilde{T}T = W \in \mathrm{Z}^\times$ and $\widehat{\widetilde{T}}T = U \in \mathrm{Z}^\times$. Hence, $T = \widetilde{T^{-1}}W$ and $\widehat{T^{-1}} = \widehat{U^{-1}}\widetilde{T}$. We obtain $\widehat{T^{-1}}T = (\widehat{U^{-1}}\widetilde{T})(\widetilde{T^{-1}}W) = \widehat{U^{-1}}W \in \mathrm{Z}^\times = \ker(\mathrm{ad})$ (Lemma E.3). Therefore, $T \in \mathrm{P}$ by (76). $\square$ Therefore, $\mathrm{Q} = \mathrm{P} \cap \mathrm{A} = \mathrm{P} \cap \mathrm{B}$ (see the Euler diagram in Figure 1).

We need to prove Theorem E.9 for the proof of equivariance of projections onto different subspaces of GAs with respect to the groups P, A, B, and Q. For this purpose, we need to introduce several auxiliary groups:

$$\mathrm{P}^\Lambda := \mathrm{P}\Lambda^\times = (C\ell^{(0)\times} \cup C\ell^{(1)\times})\mathrm{Z}^\times\Lambda^\times, \tag{70}$$

$$\mathrm{A}' := \{T \in C\ell^\times : \quad \widetilde{T}T \in \mathrm{Z}^{\overline{23}\times}\}, \tag{71}$$

$$\mathrm{B}' := \{T \in C\ell^\times : \quad \widehat{\widetilde{T}}T \in \mathrm{Z}^{\overline{3}\times}\}, \tag{72}$$

where

$$\mathrm{Z}^{\overline{kl}} := \{T \in C\ell : \quad TU = UT, \quad \forall U \in C\ell^{\overline{kl}}\}, \quad k,l = 0,1,2,3; \tag{73}$$

$$\mathrm{Z}^{\overline{k}} := \{T \in C\ell : \quad TU = UT, \quad \forall U \in C\ell^{\overline{k}}\}, \quad k = 0,1,2,3. \tag{74}$$

Note that

$$\mathrm{P} \subseteq \mathrm{P}^\Lambda, \quad \mathrm{A} \subseteq \mathrm{A}', \quad \mathrm{B} \subseteq \mathrm{B}'. \tag{75}$$

We use the following equivalent definitions of P and P^Λ proved in Theorems 4.7 and 4.8 (Filimoshina & Shirokov, 2023):

$$\mathrm{P} = \{T \in C\ell^\times : \quad \widehat{T^{-1}}T \in \mathrm{Z}^\times\}, \tag{76}$$

$$\mathrm{P}^\Lambda = \begin{cases} \{T \in C\ell^\times : \quad \widehat{T^{-1}}T \in (\Lambda \oplus C\ell^n)^\times\}, & r \neq n, \\ \Lambda^\times, & r = n. \end{cases} \tag{77}$$

Theorem E.9 (Theorem 3.4 is contained in this theorem) *The groups* W^2, W^4 *(62),* $\mathrm{W}^{\mathrm{rev}}$, *and* $\mathrm{W}^{\mathrm{conj}}$ *(63) coincide with the groups* P, Q, A, *and* B *respectively. Furthermore, the following inclusions hold:*

$$\mathrm{W}^2 = \{T \in C\ell^\times : \, TC\ell^{(1)}T^{-1} = C\ell^{(1)}\} = \mathrm{P} \subseteq \{T \in C\ell^\times : \, TC\ell^{(0)}T^{-1} = C\ell^{(0)}\}, \tag{78}$$

$$\mathrm{W}^4 = \{T \in C\ell^\times : \, TC\ell^{\overline{1}}T^{-1} = C\ell^{\overline{1}}\} = \mathrm{Q} \subseteq \{T \in C\ell^\times : \, TC\ell^{\overline{m}}T^{-1} = C\ell^{\overline{m}}\}, \, m = 0,2,3, \tag{79}$$

$$\mathrm{W}^{\mathrm{rev}} = \{T \in C\ell^\times : \, TC\ell^{\overline{01}}T^{-1} = C\ell^{\overline{01}}\} = \mathrm{A} \subseteq \{T \in C\ell^\times : \, TC\ell^{\overline{23}}T^{-1} = C\ell^{\overline{23}}\}, \tag{80}$$

$$\mathrm{W}^{\mathrm{conj}} = \{T \in C\ell^\times : \, TC\ell^{\overline{12}}T^{-1} = C\ell^{\overline{12}}\} = \mathrm{B} \subseteq \{T \in C\ell^\times : \, TC\ell^{\overline{03}}T^{-1} = C\ell^{\overline{03}}\}. \tag{81}$$

Proof. Firstly, we prove (78). Let us prove $\mathrm{W}^2 \subseteq \mathrm{P}$. Suppose $T \in C\ell^\times$ satisfies $TC\ell^{(1)}T^{-1} \subseteq C\ell^{(1)}$. We get $TUT^{-1} = -(TUT^{-1})\hat{} = \widehat{T}U\widehat{T^{-1}}$ for any $U \in C\ell^{(1)}$. Multiplying both sides of this equation on the left by $\widehat{T^{-1}}$, on the right by T, we get

$$(\widehat{T^{-1}}T)U = U(\widehat{T^{-1}}T), \qquad \forall U \in C\ell^{(1)}. \tag{82}$$

In particular, (82) is true for any generator $U = e_a \in C\ell^{(1)}$, $a = 1, \ldots, n$. Since the identity element $U = e \in C\ell^0$ satisfies (82) as well, we obtain $\mathrm{ad}_{\widehat{T^{-1}}T}(U) = U$ for any $U \in C\ell$. Therefore, $\widehat{T^{-1}}T \in \ker(\mathrm{ad}) = \mathrm{Z}^\times$ (Lemma E.3). Thus, $T \in \mathrm{P}$ by (76).

Let us prove $\mathrm{P} \subseteq \mathrm{W}^2$. Suppose $T = XW \in \mathrm{P}$ (65), where $X \in C\ell^{(0)\times} \cup C\ell^{(1)\times}$ and $W \in \mathrm{Z}^\times$. Then we get $TC\ell^{(1)}T^{-1} = XWC\ell^{(1)}W^{-1}X^{-1} = XC\ell^{(1)}WW^{-1}X^{-1} = XC\ell^{(1)}X^{-1} \subseteq C\ell^{(1)}$, where we use (16). Thus, $T \in \mathrm{W}^2$.

Now let us show that $\mathrm{W}^2 \subseteq \{T \in C\ell^\times : \, TC\ell^{(0)}T^{-1} = C\ell^{(0)}\}$. We will now show that $\{T \in C\ell^\times : \, TC\ell^{(0)}T^{-1} = C\ell^{(0)}\} = \mathrm{P}^\Lambda$ (70) and then automatically get the statement, since $\mathrm{P} \subseteq \mathrm{P}^\Lambda$. So, let us show that $\{T \in C\ell^\times : \, TC\ell^{(0)}T^{-1} = C\ell^{(0)}\} \subseteq \mathrm{P}^\Lambda$. Suppose $T \in C\ell^\times$ satisfies $TC\ell^{(0)}T^{-1} \subseteq C\ell^{(0)}$. Then we get $TUT^{-1} = (TUT^{-1})\hat{} = \widehat{T}U\widehat{T^{-1}}$ for any $U \in C\ell^{(0)}$. Multiplying both sides of this equation on the left by $\widehat{T^{-1}}$, on the right by T, we obtain $(\widehat{T^{-1}}T)U = U(\widehat{T^{-1}}T)$ for any $U \in C\ell^{(0)}$. Using Lemma E.4, we have $\widehat{T^{-1}}T \in \Lambda \oplus C\ell^n$ if $r \neq n$ and $\widehat{T^{-1}}T \in \Lambda$, which means

$T \in \Lambda^\times$, if $r = n$. Thus, $T \in \mathrm{P}^\Lambda$ by (77). Let us prove $\mathrm{P}^\Lambda \subseteq \{T \in C\ell^\times : \ TC\ell^{(0)}T^{-1} = C\ell^{(0)}\}$. Suppose $T = XW \in \mathrm{P}^\Lambda$ (70), where $X \in C\ell^{(0)\times} \cup C\ell^{(1)\times}$, $W \in (\Lambda \oplus C\ell^n)^\times$ in the case of odd n and $r \neq n$ and $W \in \Lambda^\times$ in the case of even n or $r = n$. We obtain $TC\ell^{(0)}T^{-1} = XWC\ell^{(0)}W^{-1}X^{-1} = XC\ell^{(0)}WW^{-1}X^{-1} = XC\ell^{(0)}X^{-1} \subseteq C\ell^{(0)}$, where we use the property (16) and that $WC\ell^{(0)} = C\ell^{(0)}W$ by Lemma E.4. Thus, $T \in \{T \in C\ell^\times : \ TC\ell^{(0)}T^{-1} = C\ell^{(0)}\}$. This completes the proof of (78).

Now let us prove (80) and (81). We will first prove that $\mathrm{W}^{\mathrm{rev}} = \mathrm{A}$, $\mathrm{W}^{\mathrm{conj}} = \mathrm{B}$, $\{T \in C\ell^\times : \ TC\ell^{\overline{23}}T^{-1} = C\ell^{\overline{23}}\} = \mathrm{A}'$, and $\{T \in C\ell^\times : \quad TC\ell^{\overline{03}}T^{-1} = C\ell^{\overline{03}}\} = \mathrm{B}'$, and then automatically get the inclusions in (80) and (81) using (75). We have

$$\{T \in C\ell^\times : \ TC\ell^{\overline{kl}}T^{-1} \subseteq C\ell^{\overline{kl}}\} = \{T \in C\ell^\times : \ TC\ell^{\overline{k}}T^{-1} \subseteq C\ell^{\overline{kl}}, \quad TC\ell^{\overline{l}}T^{-1} \subseteq C\ell^{\overline{kl}}\}. \quad (83)$$

Consider the case $(k, l) = (0, 1)$ or $(2, 3)$. By Lemma D.5 (Filimoshina & Shirokov, 2025), we get

$$\{T \in C\ell^\times : \ TC\ell^{\overline{kl}}T^{-1} \subseteq C\ell^{\overline{kl}}\} = \{T \in C\ell^\times : \widetilde{T}T \in \mathrm{Z}^{\overline{k}\times}, \widetilde{T}T \in \mathrm{Z}^{\overline{l}\times}\} = \{T \in C\ell^\times : \widetilde{T}T \in \mathrm{Z}^{\overline{kl}\times}\}.$$

Thus,

$$\mathrm{W}^{\mathrm{rev}} = \{T \in C\ell^\times : \ \widetilde{T}T \in \mathrm{Z}^{\overline{01}\times} = \mathrm{Z}^\times\} = \mathrm{A},$$
$$\{T \in C\ell^\times : \ TC\ell^{\overline{23}}T^{-1} \subseteq C\ell^{\overline{23}}\} = \{T \in C\ell^\times : \ \widetilde{T}T \in \mathrm{Z}^{\overline{23}\times}\} = \mathrm{A}'.$$

It remains to consider the case $(k, l) = (1, 2)$ or $(0, 3)$. By Lemma D.5 (Filimoshina & Shirokov, 2025), $\{T \in C\ell^\times : \quad TC\ell^{\overline{k}}T^{-1} \subseteq C\ell^{\overline{kl}}\} = \{T \in C\ell^\times : \quad \widehat{\widetilde{T}}T \in \mathrm{Z}^{\overline{k}}\}$, so, from the equality (83), we get $\{T \in C\ell^\times : \quad TC\ell^{\overline{kl}}T^{-1} \subseteq C\ell^{\overline{kl}}\} = \{T \in C\ell^\times : \ \widehat{\widetilde{T}}T \in \mathrm{Z}^{\overline{kl}}\}$. Thus,

$$\mathrm{W}^{\mathrm{conj}} = \{T \in C\ell^\times : \quad \widehat{\widetilde{T}}T \in \mathrm{Z}^{\overline{12}\times} = \mathrm{Z}^\times\} = \mathrm{B},$$
$$\{T \in C\ell^\times : \quad TC\ell^{\overline{03}}T^{-1} \subseteq C\ell^{\overline{03}}\} = \{T \in C\ell^\times : \quad \widehat{\widetilde{T}}T \in \mathrm{Z}^{\overline{03}\times}\} = \mathrm{B}',$$

and the proof of (80) and (81) is completed.

Now let us prove (79). Note that the statement $\mathrm{Q} \subseteq \{T \in C\ell^\times : \ TC\ell^{\overline{m}}T^{-1} = C\ell^{\overline{m}}\}$, $m = 0, 1, 2, 3$, follows directly from (80) and (81), since Q is a subgroup of both A and B, so $TC\ell^{\overline{m}}T^{-1} = C\ell^{\overline{m}}$, for any $T \in \mathrm{Q}$, $m = 0, 1, 2, 3$. We now only need to prove $\mathrm{W}^4 \subseteq \mathrm{Q}$. We get

$$\begin{aligned}
\mathrm{W}^4 &= \{T \in C\ell^\times : \quad TC\ell^{\overline{1}}T^{-1} \subseteq (C\ell^{\overline{01}} \cap C\ell^{\overline{12}})\} && (84)\\
&= \{T \in C\ell^\times : \quad TC\ell^{\overline{1}}T^{-1} \subseteq C\ell^{\overline{01}}, \quad TC\ell^{\overline{1}}T^{-1} \subseteq C\ell^{\overline{12}}\} && (85)\\
&= \{T \in C\ell^\times : \quad \widetilde{T}T \in \mathrm{Z}^{\overline{1}\times} = \mathrm{Z}^\times, \quad \widehat{\widetilde{T}}T \in \mathrm{Z}^{\overline{1}\times} = \mathrm{Z}^\times\} = \mathrm{Q}, && (86)
\end{aligned}$$

where we use Lemma D.5 (Filimoshina & Shirokov, 2025) in the first equality (86) and $\mathrm{Z}^{\overline{1}} = \mathrm{Z}$ by Theorem D.4 (Filimoshina & Shirokov, 2025) in the second equality (86). This completes the proof of the theorem. □

E.6 New equivariant mappings

Theorem E.10 *Projection mappings are equivariant with respect to the following groups:*

$$\mathrm{ad}_T(\langle U\rangle_{(s)}) = \langle \mathrm{ad}_T(U)\rangle_{(s)}, \qquad s = 0, 1, \qquad \forall T \in \mathrm{P}, \quad \forall U \in C\ell; \quad (87)$$
$$\mathrm{ad}_T(\langle U\rangle_{\overline{01}}) = \langle \mathrm{ad}_T(U)\rangle_{\overline{01}}, \quad \mathrm{ad}_T(\langle U\rangle_{\overline{23}}) = \langle \mathrm{ad}_T(U)\rangle_{\overline{23}}, \qquad \forall T \in \mathrm{A}, \quad \forall U \in C\ell; \quad (88)$$
$$\mathrm{ad}_T(\langle U\rangle_{\overline{12}}) = \langle \mathrm{ad}_T(U)\rangle_{\overline{12}}, \quad \mathrm{ad}_T(\langle U\rangle_{\overline{03}}) = \langle \mathrm{ad}_T(U)\rangle_{\overline{03}}, \qquad \forall T \in \mathrm{B}, \quad \forall U \in C\ell; \quad (89)$$
$$\mathrm{ad}_T(\langle U\rangle_{\overline{m}}) = \langle \mathrm{ad}_T(U)\rangle_{\overline{m}}, \qquad \forall m = 0, 1, 2, 3, \quad \forall T \in \mathrm{Q}, \quad \forall U \in C\ell. \quad (90)$$

Proof. Let us prove (87). Suppose $T \in \mathrm{P}$ (9) and $U \in C\ell$. Then, for $s = 0, 1$, we have

$$\langle \mathrm{ad}_T(U)\rangle_{(s)} = \langle T\langle U\rangle_{(0)}T^{-1}\rangle_{(s)} + \langle T\langle U\rangle_{(1)}T^{-1}\rangle_{(s)} = \langle T\langle U\rangle_{(s)}T^{-1}\rangle_{(s)} = \mathrm{ad}_T(\langle U\rangle_{(s)}), \quad (91)$$

where in the first equality, we use (20) and apply linearity of the projection (Lemma E.1); and in the next two equalities, we apply $TC\ell^{(s)}T^{-1} \subseteq C\ell^{(s)}$ by statement (9). This completes the proof.

We similarly prove (88). Suppose $T \in \mathrm{A}$ and $U \in C\ell$. Then for $(m, l) = (0, 1)$ or $(2, 3)$:

$$\langle \mathrm{ad}_T(U)\rangle_{\overline{ml}} = \langle TUT^{-1}\rangle_{\overline{ml}} = \langle T\langle U\rangle_{\overline{01}}T^{-1}\rangle_{\overline{ml}} + \langle T\langle U\rangle_{\overline{23}}T^{-1}\rangle_{\overline{ml}} \tag{92}$$

$$= \langle T\langle U\rangle_{\overline{ml}}T^{-1}\rangle_{\overline{ml}} = T\langle U\rangle_{\overline{ml}}T^{-1} = \mathrm{ad}_T(\langle U\rangle_{\overline{ml}}), \tag{93}$$

where in the second equality of (92), we use (22) and linearity of the projection (Lemma E.1); and in the first two equalities of (93), we apply $TC\ell^{\overline{01}}T^{-1} \subseteq C\ell^{\overline{01}}$ and $TC\ell^{\overline{23}}T^{-1} \subseteq C\ell^{\overline{23}}$ by Theorem 3.4.

Let us prove (89). For $T \in \mathrm{B}$, $U \in C\ell$, $(m, l) = (1, 2)$ or $(0, 3)$, we have:

$$\langle \mathrm{ad}_T(U)\rangle_{\overline{ml}} = \langle T\langle U\rangle_{\overline{12}}T^{-1}\rangle_{\overline{ml}} + \langle T\langle U\rangle_{\overline{03}}T^{-1}\rangle_{\overline{ml}} = \langle T\langle U\rangle_{\overline{ml}}T^{-1}\rangle_{\overline{ml}} = \mathrm{ad}_T(\langle U\rangle_{\overline{ml}}), \tag{94}$$

where in the first equality, we use (22) and linearity of the projection (Lemma E.1); and then we apply $TC\ell^{\overline{12}}T^{-1} \subseteq C\ell^{\overline{12}}$ and $TC\ell^{\overline{03}}T^{-1} \subseteq C\ell^{\overline{03}}$ by Theorem 3.4.

Now we show (90). Suppose $T \in \mathrm{Q}$ and $U \in C\ell$. Then, for $m = 0, 1, 2, 3$, we have

$$\langle \mathrm{ad}_T(U)\rangle_{\overline{m}} = \sum_{i=0}^{3} \langle T\langle U\rangle_{\bar{i}}T^{-1}\rangle_{\overline{m}} = \langle T\langle U\rangle_{\overline{m}}T^{-1}\rangle_{\overline{m}} = \mathrm{ad}_T(\langle U\rangle_{\overline{m}}), \tag{95}$$

where in the first equality, we apply (21) and linearity of the projection; and then we apply $TC\ell^{\bar{i}}T^{-1} \subseteq C\ell^{\bar{i}}$, $i = 0, 1, 2, 3$, by Theorem 3.4. □

Theorem E.11 *Polynomial functions are $C\ell^\times$-equivariant. Specifically, for any polynomial $F \in \mathbb{F}[U_1, \ldots, U_l]$ of l multivectors $U_1, \ldots, U_l \in C\ell$ over the field $\mathbb{F}$: $\mathrm{ad}_T(F(U_1, \ldots, U_l)) = F(\mathrm{ad}_T(U_1), \ldots, \mathrm{ad}_T(U_l))$ for any $T \in C\ell^\times$ and $U_1, \ldots, U_l \in C\ell$.*

Proof. This follows directly from the multiplicativity, linearity, and scalar invariance of the adjoint action ad_T for any $T \in C\ell^\times$ by Lemma E.2. □

Theorem E.12 *Grade involution is P-equivariant: $\mathrm{ad}_{T_1}(\widehat{U}) = (\mathrm{ad}_{T_1}(U))\widehat{\ }$ for any $T_1 \in \mathrm{P}$, $U \in C\ell$. Reversion is A-equivariant: $\mathrm{ad}_{T_2}(\widetilde{U}) = (\mathrm{ad}_{T_2}(U))\widetilde{\ }$ for any $T_2 \in \mathrm{A}$ and $U \in C\ell$. Clifford conjugation is B-equivariant: $\mathrm{ad}_{T_3}(\widehat{\widetilde{U}}) = (\mathrm{ad}_{T_3}(U))\widehat{\widetilde{\ }}$ for any $T_3 \in \mathrm{B}$ and $U \in C\ell$.*

Proof. Grade involution, reversion, and Clifford conjugation can be expressed as linear combinations of projections onto the subspaces $C\ell^{(0)}$ and $C\ell^{(1)}$; $C\ell^{\overline{01}}$ and $C\ell^{\overline{23}}$; $C\ell^{\overline{12}}$ and $C\ell^{\overline{03}}$ respectively (see (3)). The result then follows from Theorems E.10 and E.11. □

Theorem E.13 *Norm functions are equivariant: $\mathrm{ad}_{T_1}(\widetilde{U}U) = (\mathrm{ad}_{T_1}(U))\widetilde{\ }\mathrm{ad}_{T_1}(U)$, $\mathrm{ad}_{T_2}(\widehat{\widetilde{U}}U) = (\mathrm{ad}_{T_2}(U))\widehat{\widetilde{\ }}\mathrm{ad}_{T_2}(U)$, $\mathrm{ad}_{T_3}(\widehat{U}U) = (\mathrm{ad}_{T_3}(U))\widehat{\ }\mathrm{ad}_{T_3}(U)$ for any $T_1 \in \mathrm{A}$, $T_2 \in \mathrm{B}$, $T_3 \in \mathrm{P}$, $U \in C\ell$.*

Proof. The result follows from equivariance of polynomials with respect to $C\ell^\times$ (Theorem E.11) and conjugation operations $\widetilde{\ }$, $\widehat{\widetilde{\ }}$, and $\widehat{\ }$ with respect to A, B, and P respectively (Theorem E.12). □

Theorem E.14 *The projection onto the grade-0 subspace $C\ell^0$ is $C\ell^\times$-equivariant: $\mathrm{ad}_T(\langle U\rangle_0) = \langle \mathrm{ad}_T(U)\rangle_0$ for any $T \in C\ell^\times$ and $U \in C\ell$.*

Proof. For any $T \in C\ell^\times$ and $U \in C\ell$, we have $\mathrm{ad}_T(\langle U\rangle_0) = \langle U\rangle_0 = \langle TUT^{-1}\rangle_0 = \langle \mathrm{ad}_T(U)\rangle_0$, where we use the property of projection onto $C\ell^0$ (Lemma E.1). □

F EQUIVARIANT LAYERS

This section describes in detail our framework for constructing equivariant layers using geometric algebras (GAs): GA-linear layers, GA-geometric product layers, and GA-normalization layers. In this and the following sections, we consider the real GAs $C\ell(\mathbb{R}^{p,q,r})$. The inputs and outputs of our neural networks are represented as GA elements (multivectors), and all layer operations are

multivector-valued. Almost any geometric-based data can be represented as multivectors, see example in Figure 2 and Subsection 6. Since optimizable parameters (weights) are represented as real numbers, their optimization is still possible with common optimization methods and tools.

Suppose each input data object has l channels denoted by $x_1, \dots, x_l \in C\ell$. For example, in the task of estimating the volume of a tetrahedron defined by four points in $\mathbb{R}^3$, each data object (tetrahedron) would have 4 input channels, one for each point. Note that we can represent any $x_{c_{in}}, c_{in} = 1, \dots, l$, as sums of projections (2):

$$x_{c_{in}} = \langle x_{c_{in}} \rangle_0 + \cdots + \langle x_{c_{in}} \rangle_n = \langle x_{c_{in}} \rangle_{\overline{0}} + \langle x_{c_{in}} \rangle_{\overline{1}} + \langle x_{c_{in}} \rangle_{\overline{2}} + \langle x_{c_{in}} \rangle_{\overline{3}} \tag{96}$$

$$= \langle x_{c_{in}} \rangle_{(0)} + \langle x_{c_{in}} \rangle_{(1)} = \langle x_{c_{in}} \rangle_{\overline{01}} + \langle x_{c_{in}} \rangle_{\overline{23}} = \langle x_{c_{in}} \rangle_{\overline{12}} + \langle x_{c_{in}} \rangle_{\overline{03}}. \tag{97}$$

For a unified description of the layers, we introduce an auxiliary notation for the set of subspace indices associated with each group:

$$\mathrm{S} := \begin{cases} \{(0), (1)\}, & \text{for P;} \\ \{\overline{01}, \overline{23}\}, & \text{for A;} \\ \{\overline{12}, \overline{03}\}, & \text{for B;} \\ \{\overline{0}, \overline{1}, \overline{2}, \overline{3}\}, & \text{for Q (GLGENN (Filimoshina \& Shirokov, 2025));} \\ \{0, 1, \dots, n\}, & \text{for } \Gamma \text{ (CGENN (Ruhe et al., 2023a)).} \end{cases} \tag{98}$$

F.1 GA-LINEAR LAYERS

We now introduce GA-linear layers, where we parameterize linear combinations of projections of input multivectors onto specific $C\ell$ subspaces. We define three types of linear transformations:

$$y^{\mathrm{P}}_{c_{out}} := \sum_{c_{in}=1}^{l} \Big(\phi_{c_{out}c_{in}(0)} \langle x_{c_{in}} \rangle_{(0)} + \phi_{c_{out}c_{in}(1)} \langle x_{c_{in}} \rangle_{(1)} \Big), \tag{99}$$

$$y^{\mathrm{A}}_{c_{out}} := \sum_{c_{in}=1}^{l} \Big(\phi_{c_{out}c_{in}\overline{01}} \langle x_{c_{in}} \rangle_{\overline{01}} + \phi_{c_{out}c_{in}\overline{23}} \langle x_{c_{in}} \rangle_{\overline{23}} \Big), \tag{100}$$

$$y^{\mathrm{B}}_{c_{out}} := \sum_{c_{in}=1}^{l} \Big(\phi_{c_{out}c_{in}\overline{12}} \langle x_{c_{in}} \rangle_{\overline{12}} + \phi_{c_{out}c_{in}\overline{03}} \langle x_{c_{in}} \rangle_{\overline{03}} \Big), \tag{101}$$

where $\phi_{c_{out}c_{in}(0)}, \dots, \phi_{c_{out}c_{in}\overline{03}} \in \mathbb{R}$ are optimizable parameters (weights). The difference between these three mappings is that they are equivariant with respect to the groups P, A, and B respectively. This holds since these mappings are compositions of polynomials and projections, which are equivariant by Table 1 (see Theorems E.10 and E.11 in Section 3.2 for details). A more general formulation of a GA-linear layer is given by:

$$\mathrm{GALinear}(x_1, \dots, x_l, c_{out}, \mathrm{G}) \quad := \sum_{c_{in}=1}^{l} \sum_{s \in \mathrm{S}} \phi_{c_{out}c_{in}s} \langle x_{c_{in}} \rangle_s, \tag{102}$$

where G specifies the desired equivariance group, S (98) is the set of subspace indices associated with that group, and $\phi_{c_{out}c_{in}s} \in \mathbb{R}$ are optimizable parameters. For example, $\mathrm{GALinear}(x_1, \dots, x_l, c_{out}, \mathrm{P}) = y^{\mathrm{P}}_{c_{out}}$, $\mathrm{GALinear}(x_1, \dots, x_l, c_{out}, \mathrm{A}) = y^{\mathrm{A}}_{c_{out}}$, and $\mathrm{GALinear}(x_1, \dots, x_l, c_{out}, \mathrm{B}) = y^{\mathrm{B}}_{c_{out}}$. Note that $\mathrm{GALinear}(x_1, \dots, x_l, c_{out}, \mathrm{Q})$ is a GLGENN linear layer (Filimoshina & Shirokov, 2025) and $\mathrm{GALinear}(x_1, \dots, x_l, c_{out}, \Gamma)$ is a CGENN linear layer (Ruhe et al., 2023a).

F.2 GA-GEOMETRIC PRODUCT LAYERS

To introduce non-linearities consistent with GA, we propose a framework for parameterizing geometric products that are equivariant with respect to different groups.

First, for the input channels $x_1, \dots, x_l$, we calculate their linear combinations using GA-linear layer as $z_1 := \mathrm{GALinear}(x_1, \dots, x_l, c_1, \mathrm{G}); \dots; z_l := \mathrm{GALinear}(x_1, \dots, x_l, c_l, \mathrm{G})$. We then define a parameterized geometric product for each pair $(x_{c_{in}}, z_{c_{in}})$, $c_{in} = 1, \dots, l$, as

$$y^{G}_{c_{out}} := \mathrm{GAProduct}(x_{c_{in}}, z_{c_{in}}, \mathrm{G}) := \sum_{s_1, s_2, s_3 \in \mathrm{S}} \phi_{s_1 s_2 s_3} \langle \langle x_{c_{in}} \rangle_{s_1} \langle z_{c_{in}} \rangle_{s_2} \rangle_{s_3}, \tag{103}$$

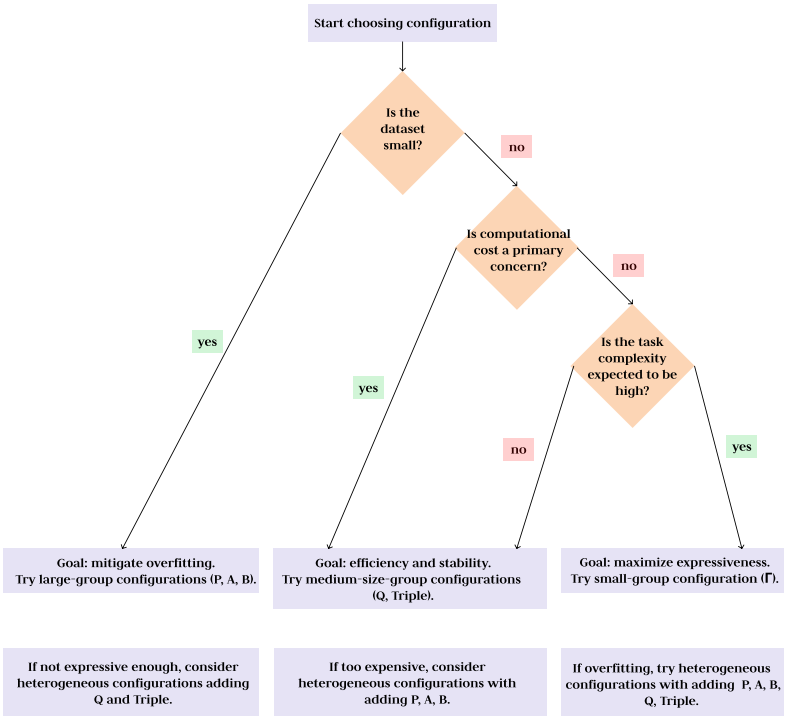


Figure 5: A decision flowchart for model configuration selection (Section 5), guided by task requirements.

where G specifies the equivariance group, S (98) is the corresponding set of indices, and $\phi_{s_1 s_2 s_3} \in \mathbb{R}$ are optimizable weights. Note that $y^{\mathrm{Q}}_{c_{out}}$ and $y^{\Gamma}_{c_{out}}$ are GLGENN and CGENN geometric product layers respectively (Filimoshina & Shirokov, 2025; Ruhe et al., 2023a). The mappings (103) are equivariant with respect to the corresponding group G = P, A, B, Q, Γ, since they are compositions of polynomials and projections, which are equivariant by Table 1 (see Theorems E.10 and E.11 in Section 3.2). The layer defined in (103) preserves the number of input and output channels (l). For greater expressiveness at increased computational cost, we can define fully connected geometric product layer, where the number of output channels may vary:

$$\begin{aligned}&\mathrm{GAFullyConnectedProduct}(x_1, \dots, x_l, z_1, \dots, z_l, c_{out}, \mathrm{G}) \\ &:= \mathrm{GALinear}\Big(\mathrm{GAProduct}(x_1, z_1, \mathrm{G}), \dots, \mathrm{GAProduct}(x_l, z_l, \mathrm{G}), c_{out}, \mathrm{G}\Big).\end{aligned} \tag{104}$$

In (104), we additionally parameterize linear combinations of the outputs from the ordinary geometric product layer using a GA-linear layer.

Note that the geometric product can be implemented efficiently in Python, and this approach is applied to CGENN (Γ-config) in Zhdanov (2025). We implemented a similar acceleration of the geometric product used in the P, A, B, Triple, Q geometric product layers, and likewise obtained reduction in computational cost by a factor of 2^n-times (where $n = \dim V$) in FLOPs across all configurations. This also results in faster performance.

F.3 GA-NORMALIZATION LAYERS

To enhance numerical stability, we introduce a normalization framework applied to layer inputs prior to the GA-geometric product layer. For each input channel $x_{c_{in}}$, $c_{in} = 1, \dots, l$, we compute:

$$\text{GANormalization}(x_{c_{in}}, \text{G}) := \sum_{s \in \text{S}} \frac{\langle x_{c_{in}} \rangle_s}{\sigma(\phi_{c_{in}s})(\text{Norm}(\langle x_{c_{in}} \rangle_s) - 1) + 1}, \tag{105}$$

where G specifies the desired equivariance group, S (98) is the corresponding set of subspace indices, $\phi_{c_{in}s} \in \mathbb{R}$ are optimizable weights, σ is the logistic sigmoid function, and Norm is a norm function that depends on group G: Norm(x) $= \sqrt{|\langle \psi(x) \rangle_0|} = \sqrt{|\langle \widetilde{x} x \rangle_0|}$ can be used in the cases of groups A, Q, Γ; Norm(x) $= \sqrt{|\langle \chi(x) \rangle_0|} = \sqrt{|\langle \widehat{\widetilde{x}} x \rangle_0|}$ can be applied in the cases B, Q, Γ; and Norm(x) $= \sqrt{|\langle \phi(x) \rangle_0|} = \sqrt{|\langle \widehat{x} x \rangle_0|}$ can be applied in the cases P, Q, Γ. The mapping (105) with the correct choice of Norm is equivariant with respect to the corresponding group G = P, A, B, Q, Γ, since it is a composition of polynomials, norm function and projections (including projection onto $C\ell^0$), which are equivariant by Table 1 (see Theorems E.10, E.11, E.13, and E.14 in Section 3.2). The main idea of this normalization (105) is to smoothly vary denominator between 1 and Norm$(\langle x_{c_{in}} \rangle_s) \in C\ell^0 \equiv \mathbb{R}$. This approach generalizes the normalization schemes for Γ and Q presented in Ruhe et al. (2023a); Filimoshina & Shirokov (2025), although other normalization strategies Huang et al. (2023) can be introduced, which we do not explore in this work.

Note that non-linearities can be applied to scalar parts of multivectors equivariantly. However, applying them only to scalars does not allow interactions between grades (for example, vectors and bivectors), while nonlinearities in GA-product layers mix all grades, creating interactions. Moreover, sequential GA-product layers produce higher-order interactions (for example, vectors combine to bivectors, then to trivectors). Also note that other non-linearities can be introduced via our parameterization. For example, we implement GA-SiLU that computes a group-equivariant norm per each projection of input multivector $x_{c_{in}}$ onto fixed subspaces (as in GA-normalization), applies learnable scalars a_s, b_s, and gates via $\sigma(a_s \text{Norm}(\langle x_{c_{in}} \rangle_s) + b_s)\langle x_{c_{in}} \rangle_s$ for each $s \in$ S (10).

We also parameterize linear combinations and geometric products of projections onto the direct sums of grade subspaces with step size 3 and call them *Triple-linear layers* and *Triple-geometric product layers*. They are equivariant with respect to Γ but contain less parameters than CGENN. Triple-layers occupy an intermediate position between layers for G = Q and G = A or B.

F.4 NUMBER OF PARAMETERS

In Table 4, we calculate the overall number of optimizable parameters required for the GA-linear, GA-geometric product, and GA-normalization layers introduced in Section 4.

In GA-linear and GA-normalization layers, we do not parameterize geometric products; therefore, the number of weights depends only on the number of subspaces onto which we project (which in turn depends on the equivariance group) and coincides with the cardinality of the corresponding set S. Specifically, for Γ, we project onto $n+1$ grade subspaces $C\ell^k$, $k = 0, \dots, n$. For Q, we use projections onto the 4 quaternion-type subspaces $C\ell^{\overline{m}}$, $m = 0, 1, 2, 3$. For Triple, we project onto 3 subspaces. For A, B, and P, we project onto 2 subspaces: $C\ell^{\overline{01}}$ and $C\ell^{\overline{23}}$; $C\ell^{\overline{12}}$ and $C\ell^{\overline{03}}$; and $C\ell^{(0)}$ and $C\ell^{(1)}$, respectively. Note that for small n, not all the subspaces mentioned above exist, so the number of parameters is smaller in these cases.

Recall that within the GA-geometric product layers, we first apply the corresponding GA-linear layer to the input, which explains the first terms in the 'GAProduct' row of Table 4. The second terms are obtained by calculating the number of parameters required to multiply the inputs and the outputs of this GALinear layer. Notably, when implementing the projection of products of projections $\langle \langle x_{c_{in}} \rangle_{s_1} \langle z_{c_{in}} \rangle_{s_2} \rangle_{s_3}$ (103), we take into account that some of the products $\langle x_{c_{in}} \rangle_{s_1} \langle z_{c_{in}} \rangle_{s_2}$ inside the projections may be zero by definition, and we do not assign any weight there.

For example, to calculate the number of parameters for the P-configuration of GAProduct, we use property (16). Thus, we employ the fact that the projection of two even elements onto the odd subspace is always 0, and similarly for others. We have the following matrices:

$$\begin{bmatrix} \text{True} & \text{False} \\ \text{False} & \text{True} \end{bmatrix}, \qquad \begin{bmatrix} \text{False} & \text{True} \\ \text{True} & \text{False} \end{bmatrix}. \tag{106}$$

Table 4: Number of trainable parameters on GA-linear, GA-geometric product, and GA-normalization layers, where l is the number of input channels, m is the number of output channels, $n = \dim V$.

LAYER	CONFIGURATION	NUMBER OF PARAMETERS
GALinear	P	$2lm$
	A	$2lm$ if $n \geq 2$
	B	$2lm$
	Triple	$3lm$ if $n \geq 2$
	Q	$4lm$ if $n \geq 3$
	Γ	$(n+1)lm$
GAProduct	P	$2l^2 + 4l$ if $n \geq 2$
	A	$2l^2 + 8l$ if $n \geq 2$
	B	$2l^2 + 8l$ if $n \geq 2$
	Triple	$3l^2 + 23l$ if $n \geq 2$
	Q	$4l^2 + 29l$ if $n \geq 3$
GANormalization	P	$2l$
	A	$2l$ if $n \geq 2$
	B	$2l$
	Triple	$3l$ if $n \geq 2$
	Q	$4l$ if $n \geq 3$
	Γ	$(n+1)l$

The first row and column correspond to $C\ell^{(0)}$, and the second row and column correspond to $C\ell^{(1)}$. The first and second matrices contain the answers to the question of whether we can obtain an element from $C\ell^{(0)}$ and $C\ell^{(1)}$, respectively, as a product of elements from the subspaces indicated by the rows and columns. In these matrices, we have 4 True values, which is why we have $4l$ as the second term. We proceed similarly for other subspaces.

To provide practical guidance for selecting a configuration, we present a decision flowchart in Figure 5. We also present Table 5 as empirical heuristics derived from our benchmarks. These are not guaranteed to transfer universally, but they offer reasonable starting points.

G Experimental results and discussion

In this section, we discuss the experimental results for the experiments from Section 6.

To control overfitting, the data is split into train, validation, and test datasets, following the methodology originally proposed in CGENN (Ruhe et al., 2023a). The results presented in the manuscript are evaluated using the test dataset, while validation datasets are used for monitoring. We report metrics representing the mean $\pm$ standard deviation (with ddof=1) of the results over 20 runs for $\mathrm{O}(8)$-Regression, $\mathrm{O}(6)$-Convex Hull, and N-Body experiments, and over 5 runs for $\mathrm{O}(1,3)$-Top Tagging, with varying random seeds.

G.1 $\mathrm{O}(6)$-Convex Hull

In this task, we again vary training dataset sizes: 256, 1024, and 4096. Points are sampled from a standard Gaussian distribution. Note that some points may lie inside the hull and do not contribute to the volume. The volume is an $\mathrm{O}(6)$-invariant quantity, which enables baseline methods to approximate its value.

As in the regression experiment, all networks adopt the same five-layer structure: one GA-linear layer followed by four GA-geometric product layers with internal GA-normalization. We tested the following homogeneous layer-wise configurations: P-P-P-P-P, B-B-B-B-B, A-A-A-A-A, Triple-Triple-Triple-Triple-Triple, Q-Q-Q-Q-Q (GLGENN (Filimoshina & Shirokov, 2025)), and

Table 5: Practical guidelines for config selection (empirical, based on our experiments).

IF YOUR PRIORITY IS...	RECOMMENDED CONFIG	JUSTIFICATION (BASED ON OUR EXPERIMENTS)
Strong regularization (small train set, < 500 samples)	P, A, or B	Fewest parameters; lowest observed mean MSE on N-body with 50 samples (Table 8)
Minimal computational budget / prototyping	P	The most sparse parameterization; usually 3–10x fewer parameters than Q or Γ (Table 4)
Exploratory / default starting point	Triple or A-Triple	Strong overall trade-off; works well without tuning. Consistently top-tier across all tasks (Tables 6, 8)
Fast initial training	Γ	Trains fastest initially (Figure 7), but beware of overfitting on small datasets
Maximum expressiveness (and you have a large train set, $>$ 4000 samples)	Q or Γ	Lowest observed mean MSE on regression with the largest training sets (Tables 7)
Exact $\mathrm{O}(V, \mathfrak{q})$ or $\mathrm{SO}(V, \mathfrak{q})$ equivariance	Γ	Theorem 3.1; only Γ yields orthogonal group action
Lorentz-equivariant tasks (for example, particle physics)	A or Q	Highest accuracy on top tagging (Table 10)
Invariant scalar prediction (for example, volume, energy)	Triple-Γ	Lowest observed mean MSE on convex hull volume estimation (Table 6)

Γ-Γ-Γ-Γ-Γ (CGENN (Ruhe et al., 2023a)), and heterogeneous configurations: A-A-Q-Triple-Q, Triple$-\Gamma-$Triple$-\Gamma-$Triple, and Triple-Q-Triple-Q-Triple.

We compare our configurations against scalarization equivariant approaches: Geometric Vector Perceptrons (GVP) Jing et al. (2021), Vector Neurons (VN) Deng et al. (2021), and with Equivariant MLP (EMLP) Finzi et al. (2021) and ordinary MLP. The number of parameters for non-GA models is aligned with our best-performing and largest configurations respectively.

The final results are provided in Table 6 and Figure 6. In this task, the number of parameters for different configurations varies even more than in the previous one and follows the order:

$$\text{P-config} < \text{A-config}, \text{B-config} < \text{A-Q-Triple-config} \tag{107}$$

$$< \text{Triple-config} < \text{Triple-Q-config} < \text{Q-Triple-config} < \text{Q-config} < \Gamma\text{-config}. \tag{108}$$

An interesting fact is that Triple-Γ-configuration achieves the lowest observed mean MSE among all homogeneous configurations across all dataset sizes, including on the largest dataset. This demonstrates that strategically mixing layers equivariant with respect to different groups can capture complementary geometric properties that neither group alone provides.

Another interesting insight is presented in Figure 7. It presents test loss in the case of different maximal training step. Γ-configuration trains fastest initially, it eventually overfits, whereas more constrained configurations exhibit better generalization. This highlights the regularization effect of larger symmetry groups.

Table 6: $O(6)$-convex hull volume estimation, MSE (↓) on test dataset. The first column specifies the configuration (Section 5) or the competitor non-GA model, the second column reports the number of the trainable parameters (weights), and the next three columns show the test MSE (mean ± std) in the cases of 256, 1024, and 4096 training samples. We align non-GA model sizes with our (i) best-performing (Triple-Γ, 50k parameters) and (ii) largest (Γ, 98k parameters) GA configs. The results for each setting are averaged over 20 runs. Batch size is 128 in each setting. Maximal number of training steps is 512, 1024, and 2048 for 256, 1024, and 4096 training samples respectively.

CONFIGURATION	**# PARAMS**	**# TRAIN SAMPLES**		
		256	1024	4096
P-config	13185	11.5753 ± 0.7627	9.9877 ± 0.3843	8.2101 ± 0.5285
A-config	13697	11.1509 ± 0.5476	8.4365 ± 0.6376	4.3730 ± 0.3670
B-config	13697	12.0705 ± 1.0472	9.7343 ± 0.5606	8.1517 ± 0.7664
A-Q-Triple-config	21121	**10.4949** ± 0.5659	7.7101 ± 0.7222	3.4206 ± 0.2709
Triple-config	21889	10.7944 ± 0.6043	7.3769 ± 0.8273	**2.9828** ± 0.2571
Triple-Q-config	25345	10.6222 ± 0.5353	7.5950 ± 0.7214	**2.9277** ± 0.1904
Q-config (GLGENN)	28289	**10.4889** ± 0.3718	7.7402 ± 0.7227	3.6442 ± 0.3296
Triple-Γ-config	49537	**10.2771** ± 0.4345	**6.8995** ± 0.8583	**2.7372** ± 0.2779
Γ-config (CGENN)	97761	11.7189 ± 0.4347	**6.9715** ± 0.3910	3.6312 ± 0.3546
GVP	50k	16.9086 ± 0.0744	11.0531 ± 0.7417	6.3064 ± 0.7409
	98k	17.0416 ± 0.0685	11.2007 ± 2.0682	5.5967 ± 0.2060
VN	50k	21.0361 ± 0.6817	22.1814 ± 0.7740	21.8067 ± 0.9222
	98k	18.5774 ± 0.5320	17.8017 ± 0.9326	20.2316 ± 0.4799
MLP	50k	17.7889 ± 0.3293	12.9695 ± 0.1495	11.6500 ± 0.0403
	98k	16.2241 ± 0.2442	11.8670 ± 0.1681	10.6347 ± 0.1308
EMLP	505k	14.5928 ± 0.3150	13.8233 ± 0.4984	11.6414 ± 0.5257

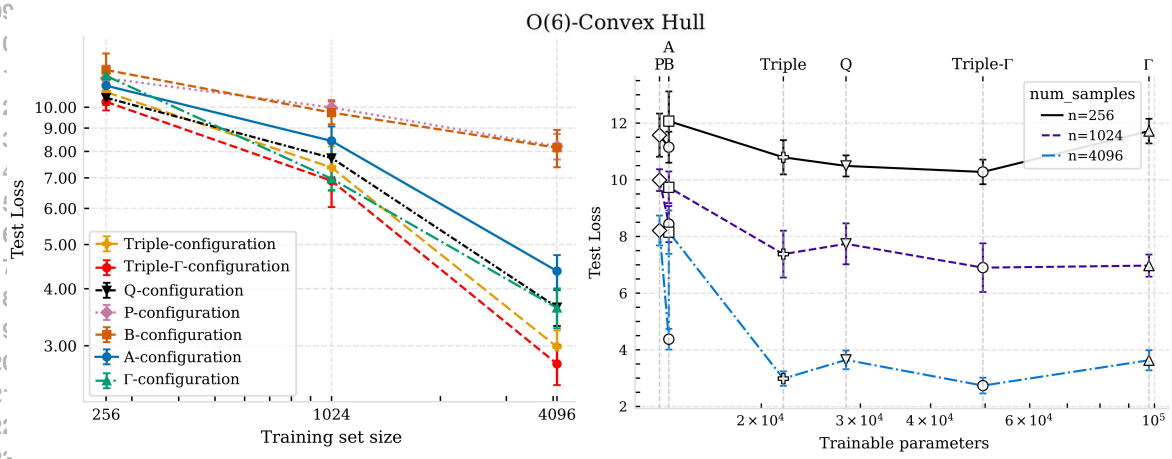


Figure 6: $O(6)$-convex hull volume estimation. Left: test loss (MSE ↓) versus training set size. Right: test loss versus number of parameters (and the corresponding equivariance group), shown for all training set sizes.

G.2 $O(8)$-Regression

We evaluate our models on a synthetic regression task with varying training set sizes: 50, 500, and 5000. We use Mean Squared Error (MSE) as the evaluation metric. Vectors, scalars, and pseudoscalars are sampled from a standard Gaussian distribution to generate the train, test, and validation datasets.

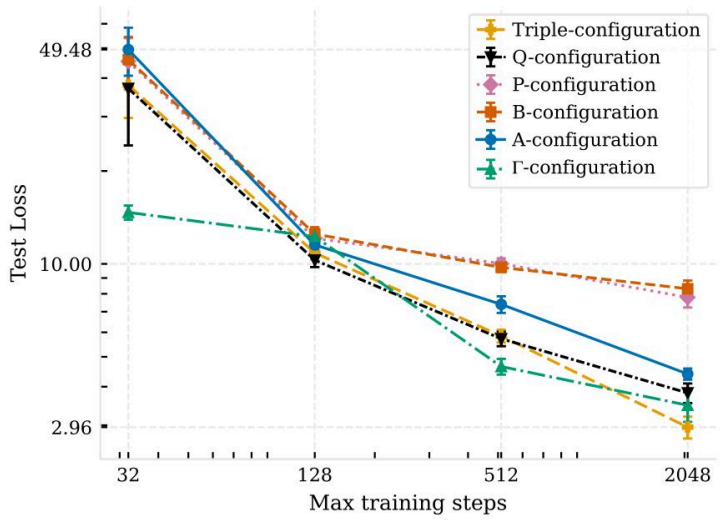


Figure 7: $\mathrm{O}(6)$-convex hull volume estimation. Test MSE (↓) versus maximal training step for 4096 training samples.

All networks share a fixed structure of five layers: one GA-linear layer followed by four GA-geometric product layers, each equipped with internal GA-normalization. We denote the specific configuration of a network by a sequence of five letters, where each letter indicates the equivariance group used in the corresponding layer. For example, the notation 'A-A-A-A-A' denotes a homogeneous network where all five layers are equivariant to the group A, while 'A-Q-A-Q-A' denotes a heterogeneous network with alternating A and Q layers.

We test the following layer-wise configurations (Section 5). Homogeneous configurations include P-P-P-P-P, B-B-B-B-B, A-A-A-A-A, Triple-Triple-Triple-Triple-Triple, Q-Q-Q-Q-Q (GLGENN (Filimoshina & Shirokov, 2025)), and Γ-Γ-Γ-Γ-Γ (CGENN (Ruhe et al., 2023a)). Heterogeneous configurations are selected based on preliminary homogeneous results; they include A-Q-A-Q-A, A-A-Q-Triple-Q, and Q-Triple-Q-Triple-Q. For brevity in the text and tables, we refer to these by their dominant group, for example, 'A-config' for A-A-A-A-A or 'A-Q-config' for A-Q-A-Q-A.

The results presented in Table 7 and in Figure 8 reveal relationship between model size (or chosen equivariance group) and performance. The number of trainable parameters varies significantly across configurations, following the order:

$$\text{P-config} < \text{A-config}, \text{B-config} < \text{A-Q-config} < \text{A-Q-Triple-config} \tag{109}$$

$$< \text{Triple-config} < \text{Q-Triple-config} < \text{Q-config} < \Gamma\text{-config}. \tag{110}$$

Notably, the largest model (Γ-config, CGENN) achieves higher observed mean MSE than medium-sized configs, indicating that its additional expressiveness is unnecessary and may even be harmful for this problem.

In a similar but simpler $\mathrm{O}(5)$-task, Q- and Γ-configs outperform standard MLP, MLP with augmentations, and EMLP as stated in Filimoshina & Shirokov (2025); Ruhe et al. (2023a).

The right part of Figure 8 illustrates the relationship between parameter count and test loss. We observe that medium-sized configurations generally achieve the lowest observed mean MSE on this task, compared to both the smallest (most constrained) and largest (most expressive) models.

G.3 $\mathrm{O}(3)$- AND $\mathrm{O}(6)$-N-BODY SIMULATION

The N-body experiment serves as a standard benchmark for assessing the performance of equivariant (graph) neural networks in simulating physical systems. We consider these tasks in $\mathbb{R}^3$ ($\mathrm{O}(3)$-equivariance) and $\mathbb{R}^6$ ($\mathrm{O}(6)$-equivariance) settings. The first setting is considered for fair

Table 7: $O(8)$-regression task, MSE (↓) on test dataset. The first column specifies the configuration (Section 5), the second column reports the number of the trainable parameters (weights), and the next three columns show the test MSE (mean ± std) in the cases of 50, 500, and 5000 training samples. The results for each setting are averaged over 20 runs. Maximal number of training steps is 1024, batch size is 16 for 50 training samples and 32 for 500 and 5000 training samples.

CONFIGURATION	**# PARAMS**	**# TRAIN SAMPLES**		
		50	500	5000
P-config	753	0.9435 ± 0.5858	0.4085 ± 0.2624	0.3590 ± 0.1667
A-config	881	**0.5989** ± 0.5848	0.1717 ± 0.0505	0.1486 ± 0.0191
B-config	881	1.4578 ± 1.8599	0.3303 ± 0.1604	0.2269 ± 0.1411
A-Q-Triple-config	1441	0.6132 ± 0.5222	0.0751 ± 0.0276	0.0568 ± 0.0100
A-Q-config	1553	0.6840 ± 0.7795	0.0876 ± 0.0423	0.0500 ± 0.0095
Triple-config	1793	0.7471 ± 0.6543	**0.0534** ± 0.0236	0.0353 ± 0.0102
Q-Triple-config	2023	**0.6038** ± 0.4603	0.0661 ± 0.0398	**0.0315** ± 0.0092
Q-config (GLGENN)	2257	0.7484 ± 0.7028	**0.0525** ± 0.0251	**0.0311** ± 0.0112
Γ-config (CGENN)	7905	0.9605 ± 0.7099	0.0652 ± 0.0211	0.0332 ± 0.0069

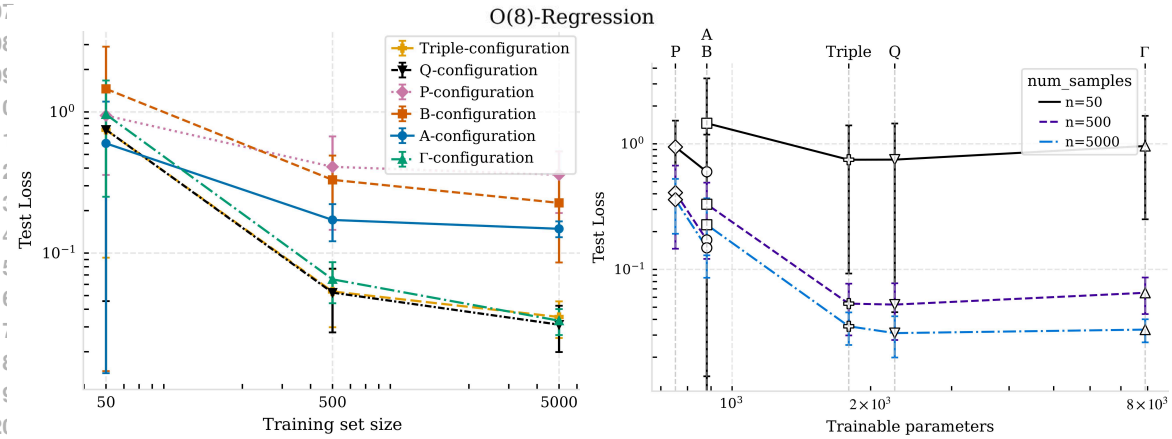


Figure 8: $O(8)$-regression task. Left: test loss (MSE ↓) versus training set size. Right: test loss versus number of parameters (and the corresponding equivariance group), shown for all training set sizes.

comparison of our configs with irreps-based equivariant methods, since they are tailored specially for $O(3)$-equivariance. The second setting is considered for comparison of our configs in the high-dimensional case. In this experiment, we test configurations across 3 train dataset sizes (50, 500, 5000 samples).

In $O(3)$-setting, we compare our homogeneous P-, A-, B-, Triple-, Γ- (coincides with Q in this low-dimensional case) configs with: a) Scalarization equivariant methods: Radial Field (RF) Köhler et al. (2020), EGNN Satorras et al. (2021), FastEGNN Zhang et al. (2024); b) Irreps/tensor products equivariant methods: SE(3)-Transformers Fuchs et al. (2020), Tensor Field Networks (TFN) Thomas et al. (2018), Equiformer Liao & Smidt (2023); c) Method based on Graph Neural ODE: SEGNO Liu et al. (2024b); d) Non-equivariant counterpart of EGNN: GNN Satorras et al. (2021). The architectures and numbers of parameters for non-GA models are taken from their code repositories (best-reported models). The results are presented in Table 9 and Figure 3 (C). In general, several of our configurations perform at the level of non-GA state-of-the-art models such as SEGNO and FastEGNN. Notably, on the 5000-sample setting, our A-config attains the overall best MSE, substantially improving over SEGNO and EGNN.

In $O(6)$-setting, we compare the homogeneous A-, P-, B-, Triple-; baseline Q- (GLGENN) and Γ- (CGENN); and heterogeneous A-P-, A-Triple-, and A-Q-configs. They are ordered by the number

of trainable parameters as follows:

$$\text{P-config} < \text{A-P-config} < \text{A-config, B-config} < \text{A-Triple-config} < \text{A-Q-config} \tag{111}$$

$$< \text{Triple-config} < \text{Q-Triple-config} < \text{Q-config} < \Gamma\text{-config.} \tag{112}$$

The results are in Table 8 and Figure 9. On the smallest dataset, the parameter-efficient configs achieve much lower MSE than Q- and Γ-configs. A-Triple-config consistently delivers top-tier performance across all training dataset sizes, achieving the best results on the largest dataset. Γ-config advantage over more parameter-efficient configs is minimal compared to the 20x parameter increase.

Table 8: $\mathrm{O}(6)$-N-body dynamics simulation, MSE (↓) on test dataset. The first column specifies the configuration (Section 5), the second column reports the number of the trainable parameters (weights), and the next three columns show the test MSE (mean ± std, ddof=1) in the cases of 50, 500, and 5000 training samples. The results for each setting are averaged over 20 runs in the case of 50 and 500 training samples, and over 10 runs in the case of 5000 training samples. Batch size is 16, 64, and 128 in each setting respectively. Maximal number of training steps is 1024 for 50 training samples, and 4096 for 500 and 5000 training samples respectively. Q-config is GLGENN, Γ-config is CGENN.

CONFIG	**# PARAMS**	**# TRAIN SAMPLES**		
		50	500	5000
P	13185	**0.002509** ± 0.000413	0.000910 ± 0.000156	0.000366 ± 0.000054
A-P	48693	0.003163 ± 0.000674	0.000878 ± 0.000140	0.000310 ± 0.000047
A	49141	0.003278 ± 0.000521	**0.000777** ± 0.000119	**0.000280** ± 0.000054
B	49141	0.003314 ± 0.000575	0.001307 ± 0.000426	0.000618 ± 0.000356
A-Triple	62637	**0.002719** ± 0.000488	**0.000770** ± 0.000096	**0.000259** ± 0.000041
A-Q	73557	0.003607 ± 0.000778	0.000800 ± 0.000094	**0.000280** ± 0.000018
Triple	78233	0.002809 ± 0.000573	0.000807 ± 0.000129	0.000281 ± 0.000023
Q	102621	0.003878 ± 0.000903	0.000944 ± 0.000140	**0.000277** ± 0.000043
Γ	185025	0.010871 ± 0.004770	0.002647 ± 0.000650	0.000508 ± 0.000073

Table 9: $\mathrm{O}(3)$-N-body dynamics simulation, MSE (↓) on test dataset. The first column specifies our configuration (Section 5) or competitor model, the second column reports the number of the trainable parameters (weights), and the next three columns show the test MSE (mean ± std, ddof=1) in the cases of 50, 500, and 5000 training samples. The results for each setting are averaged over 3 runs. Batch size is 16, 64, and 128 in each setting respectively. Maximal number of training steps is 1024 for 50 training samples, and 4096 for 500 and 5000 training samples respectively. Q-config is GLGENN, Γ-config is CGENN.

MODEL	**# PARAMS**	**# TRAIN SAMPLES**		
		50	500	5000
P-config	62k	**0.0308** ± 0.0036	0.0140 ± 0.0058	0.0077 ± 0.0048
A-config	63k	0.0405 ± 0.0044	0.0124 ± 0.0012	**0.0048** ± 0.0002
B-config	63k	0.0501 ± 0.0178	**0.0119** ± 0.0017	0.0124 ± 0.0033
Triple-config	96k	0.0420 ± 0.0080	0.0148 ± 0.0010	**0.0056** ± 0.0005
Γ(Q)-config	124k	0.0934 ± 0.0109	0.0224 ± 0.0072	0.0085 ± 0.0007
EGNN	134k	0.0326 ± 0.0172	**0.0115** ± 0.0008	0.0067 ± 0.0003
RF	2k	0.0433 ± 0.0181	0.0216 ± 0.0008	0.0187 ± 0.0018
SE(3)-Tr	176k	0.0534 ± 0.0040	0.0221 ± 0.0039	0.0075 ± 0.0014
TFN	188k	0.1141 ± 0.0005	0.0552 ± 0.0034	0.0219 ± 0.0004
FastEGNN	318k	**0.0246** ± 0.0053	**0.0109** ± 0.0006	0.0063 ± 0.0008
SEGNO	34k	**0.0298** ± 0.0019	**0.0115** ± 0.0005	**0.0053** ± 0.0004
Equiformer	218k	0.1140 ± 0.0024	0.0999 ± 0.0004	0.0897 ± 0.0008
GNN	104k	0.0828 ± 0.0015	0.0353 ± 0.0026	0.0111 ± 0.0005

A-Triple-configuration consistently delivers top-tier performance across all training dataset sizes, achieving the best results on the largest dataset. This suggests that combining the stability of larger-

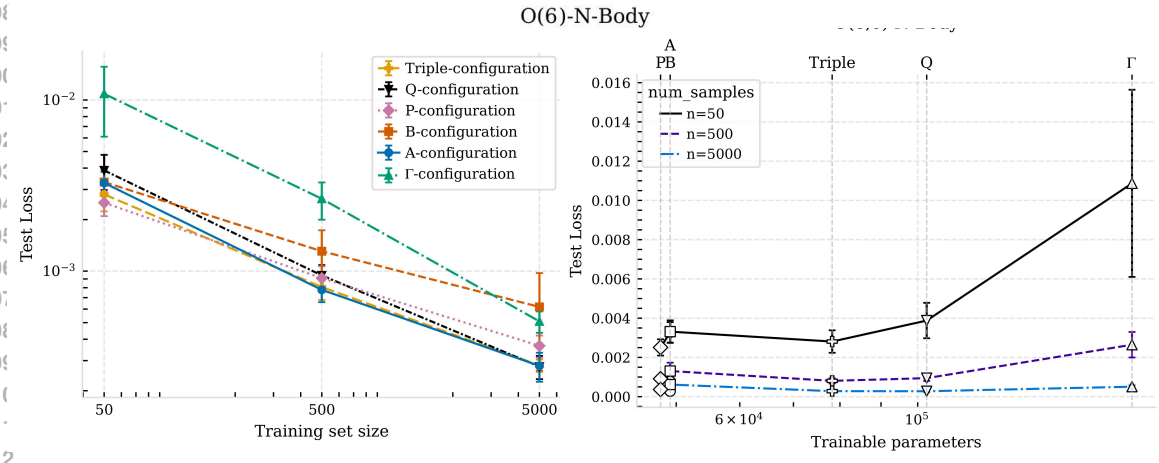


Figure 9: O(6)-N-body dynamics simulation. Left: test loss (MSE ↓) versus training set size. Right: test loss versus number of parameters (and the corresponding equivariance group), shown for all training set sizes.

group layers with the expressiveness of medium-group layers captures the optimal balance for learning physical dynamics.

Also note that Γ-configuration advantage over more parameter-efficient configurations is minimal compared to the 20x parameter increase. The right part of Figure 9 shows that this configuration exhibits significant overfitting on small training datasets compared to other models.

G.4 O(1, 3)-Top Tagging

In this Lorentz-equivariant task, we evaluate our models on top quark jet tagging across three dataset sizes (128, 512, and 1024 samples), comparing all proposed homogeneous configurations.

Table 10: O(1, 3)-top tagging, Accuracy (↑) on test dataset. The first column specifies the configuration (Section 5), the second column reports the number of the trainable parameters (weights), and the next four columns show the test MSE (mean ± std) in the cases of 128, 512, and 1024 training samples. Batch size is 32 in all setting. Maximal number of training steps is 1000 for 128 and 512 training samples, and 3000 for 1024 training samples respectively. Q-configuration is GLGENN, Γ-configuration is CGENN.

CONFIGURATION	**# PARAMS**	**# TRAIN SAMPLES**		
		128	512	1024
P-config	250962	0.8922 ± 0.0131	0.9164 ± 0.0065	0.9234 ± 0.0035
A-config	253010	0.8922 ± 0.0102	0.9141 ± 0.0103	**0.9277** ± 0.0023
B-config	253010	**0.9078** ± 0.02168	0.9172 ± 0.0064	0.9172 ± 0.0119
Triple-config	263194	0.8891 ± 0.01782	0.9234 ± 0.0076	0.9250 ± 0.0093
Q-config	268770	0.8992 ± 0.0050	**0.9281** ± 0.0045	0.9258 ± 0.0076
Γ-config	320594	0.8828 ± 0.0221	0.9172 ± 0.0075	0.9209 ± 0.0138

The results are presented in Table 10 and Figure 10. Unlike previous tasks where performance varied significantly, here all configurations achieve remarkably similar accuracy, however the number of parameters in the models varies a lot. The Q-configuration achieves the highest observed mean accuracy on medium-sized datasets, while the A-configuration does so on the largest dataset (1024 samples). The right part of Figure 10 shows that for any considered number of training samples, the best results are achieved in medium-small-sized configurations.

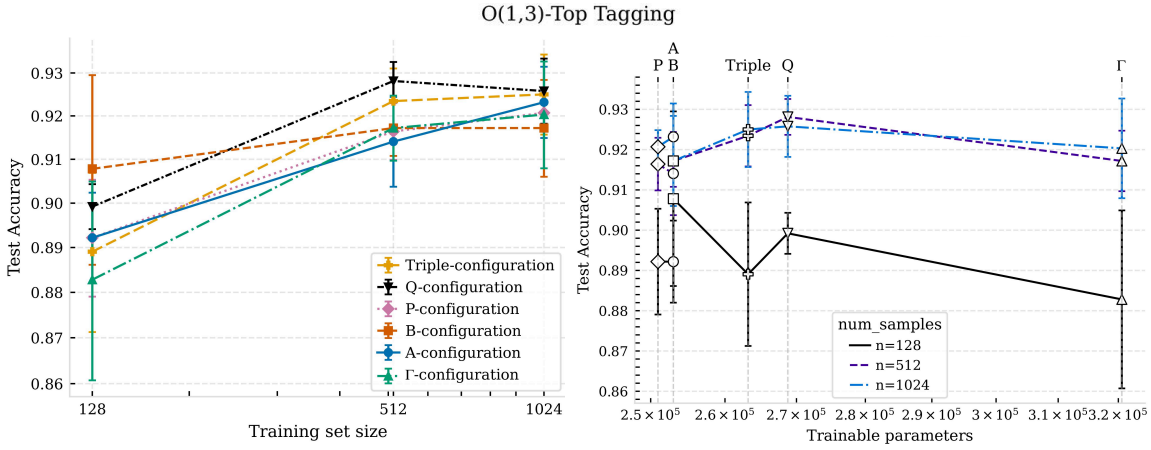


Figure 10: $\mathrm{O}(1,3)$-top tagging. Left: Accuracy (↑) on the test dataset versus training set size. Right: Accuracy (↑) on the test dataset versus number of parameters (and the corresponding equivariance group), shown for all training set sizes.

G.5 $\mathrm{O}(3)$-Atom Position Prediction

The task is to predict atom positions in the next ten time steps using ten time frames for each molecule (we took aspirin, benzene, and malonaldehyde) based on MD17 benchmark Chmiela et al. (2017). We follow CSMPN paper Liu et al. (2024a) and construct fully connected graphs for benzene and malonaldehyde and use $k = 3$-nearest neighbors to construct the regular graphs from the atom positions. Following the standard setup, we use 5000 train and 2000 test samples. We have 1303 validation samples for aspirin and 2000 samples for other molecules.

We take all baselines from CSMPN (coincides with our Γ-config) and EqMotion Xu et al. (2023) and compare against RF Köhler et al. (2020), TFN Thomas et al. (2018), SE(3)-Transformers Fuchs et al. (2020), EGNN Satorras et al. (2021), EMPSN Liu et al. (2024a), NRI Kipf et al. (2018), NMMP Hu et al. (2020), GroupNet Xu et al. (2022), GMN-L Huang et al. (2022), EqMotion, CSMPN (coincides with our Γ-config). For our configs, we use a similar architecture to CSMPN (matches competitors in number of parameters, ≈300k) with 5 message passing layers with GA-layers as message/update functions. Our models have 136k (A-config), 207k (Triple-config), and 369k (Γ-config) parameters. In Table 2, we report ADE (Average Displacement Error, which is RMSE averaged over time steps) and FDE (Final Displacement Error, which is RMSE at final time step).

Γ-config trained with 10^5 steps achieves the lowest observed mean ADE/FDE across all three molecules. Notably, even with fewer parameters, our A- and Triple-configs at 10^4 steps achieve comparable or lower observed mean ADE/FDE than many 10^5-step baselines, demonstrating the parameter efficiency of our approach.

G.6 Experiments compute resources

$\mathrm{O}(6)$-Convex Hull, $\mathrm{O}(3)$- and $\mathrm{O}(6)$-N-Body, $\mathrm{O}(8)$-Regression, and $\mathrm{O}(1,3)$-Top Tagging were conducted on the high-performance cluster. A cluster node provides 768 GB of RAM (per-job allocation is not explicitly documented but was sufficient for all runs) and 40 GB of persistent storage for the project. For $\mathrm{O}(6)$-Convex Hull, $\mathrm{O}(3)$- and $\mathrm{O}(6)$-N-Body, and $\mathrm{O}(8)$-Regression, each job used 1 NVIDIA Tesla V100S 32 GB GPU and 8 CPU cores; for $\mathrm{O}(1,3)$-Top Tagging, 4 GPUs and 8 CPU cores were used. The MD17 experiments were run on a local machine equipped with a single NVIDIA GeForce RTX 3060 GPU, 8 CPU cores, 16 GB of RAM, and approximately 40 GB of storage (though actual usage per experiment was $\leq$4 GB), running CUDA 12.4. Wall-clock times are reported separately in Appendix H.

H COMPUTATIONAL COST TRADE-OFF

We complement the main experimental results with a comparison of the computational cost of our equivariant configurations. All timings were obtained using the `pytorch-lightning` backend. We report wall-clock training time per step and total inference time (mean ± std, in seconds) for the $\mathrm{O}(1,3)$-Top Tagging, $\mathrm{O}(6)$-Convex Hull, and $\mathrm{O}(3)$-N-Body tasks. The trade-off between the number of parameters, training/inference speed, and task performance (MSE or accuracy) is summarized in Figure 11.

For $\mathrm{O}(1,3)$-Top Tagging, Table 11 shows that the A- and B-configs are about 2.8× faster per training step than the Γ-config (CGENN) while using 20% fewer parameters and achieving higher accuracy. Among the larger-group models, B gives the best accuracy with the shortest train time. For $\mathrm{O}(6)$-Convex Hull, Table 12 reports timings on the largest training set (4096 samples). The best performing config (Triple-Γ) is faster than Γ-config but yields the lowest MSE. For $\mathrm{O}(3)$-N-Body, wall-clock timings for all configs and dataset sizes are given in Table 13.

Table 11: $\mathrm{O}(1,3)$-Top Tagging wall-clock time (seconds, mean ± std). 128 training samples, batch size 32. Speed-up relative to Γ-config.

CONFIG	# PARAMS	# TRAIN TIME PER STEP (S)	INFERENCE TIME (S)	ACCURACY	SPEED-UP (TRAIN)
Γ	321k	1.0389 ± 0.1899	993.13 ± 0.1337	0.8828 ± 0.0221	1.00
A	253k	0.3726 ± 0.0005	356.77 ± 0.4729	0.8922 ± 0.0102	2.79
B	253k	0.3714 ± 0.0011	355.70 ± 1.1187	**0.9078** ± 0.0217	2.80
P	251k	0.5704 ± 0.5328	546.33 ± 0.5431	0.8922 ± 0.0131	1.82
Q	269k	0.5970 ± 0.3714	571.78 ± 0.4532	0.8992 ± 0.0050	1.74
Triple	263k	0.5846 ± 1.8581	559.85 ± 1.8182	0.8891 ± 0.0178	1.78

Table 12: $\mathrm{O}(6)$-Convex Hull (4096 train samples, batch size 128). Wall-clock time (seconds) and MSE. Speed-up relative to Γ-config.

CONFIG	# PARAMS	# TRAIN TIME PER STEP (S)	INFERENCE TIME (S)	MSE	SPEED-UP (TRAIN)
Γ	97761	0.1521 ± 0.0018	302.77 ± 3.6279	3.6312 ± 0.3546	1.00
P	13185	0.0843 ± 0.0043	167.60 ± 8.6980	8.2101 ± 0.5285	1.81
A	13697	0.0753 ± 0.0127	149.82 ± 25.3922	4.3730 ± 0.3670	2.02
B	13697	0.0756 ± 0.0124	150.42 ± 24.7848	8.1517 ± 0.7664	2.01
Triple	21889	0.0959 ± 0.0076	190.88 ± 15.1907	**2.9828** ± 0.2571	1.59
Q	28289	0.1013 ± 0.0056	201.36 ± 11.1871	3.6442 ± 0.3296	1.50
A-Q-Triple	21121	0.1213 ± 0.0026	241.48 ± 5.2323	3.4206 ± 0.2709	1.25
Triple-Q	25345	0.1161 ± 0.0075	231.05 ± 15.0216	**2.9277** ± 0.1904	1.31
Triple-Γ	49537	0.1215 ± 0.0059	241.62 ± 11.9445	**2.7372** ± 0.2779	1.25

I REGULARIZATION ALTERNATIVES

A natural question is whether the benefit of using wider equivariance groups can be matched by simply adding traditional regularizers, such as weight decay, to a less constrained model. To investigate this, we compare the best performing configurations from our wider-group family (P for 50 training samples, A-Triple for 500 and 5000 samples) against the narrowest group (Γ, which corresponds to CGENN) trained with various strengths of L2 weight decay.

Table 14 reports the test MSE for the $\mathrm{O}(6)$-N-Body task (mean $\pm$ std). The wider-group models achieve substantially lower MSE on the smallest dataset (50 samples) without any weight decay, while using far fewer parameters (13k for P vs. 185k for Γ). Applying weight decay to Γ does reduce its MSE, but the best decay value (10^{-2}) still yields a higher MSE on 50 samples compared to P. On the larger datasets (500 and 5000 samples), the wider A-Triple configuration (63k parameters) again achieves lower observed mean MSE than all Γ variants, including those with weight decay.

Table 13: O(3)-N-Body wall-clock time (seconds, mean ± std) for our configurations. Batch sizes: 16 (50 samples), 64 (500 samples), 128 (5000 samples).

CONFIG	TRAIN SAMPLES	TRAIN TIME PER STEP (S)	INFERENCE TIME (S)	MSE	# PARAMS
P	50	0.1172±0.0011	116.1926±0.0160	**0.030776**±0.003611	62k
A	50	0.1159±0.0006	115.6753±0.6318	0.040510±0.004423	63k
B	50	0.1160±0.0008	115.7947±0.7489	0.050107±0.017805	63k
Triple	50	0.1350±0.0008	135.0101±0.7488	0.042017±0.007951	96k
Γ(Q)	50	0.1110±0.0002	111.0660±0.2596	0.093442±0.010890	124k
P	500	0.1162±0.0024	461.7475±9.8588	0.014007±0.005793	62k
A	500	0.1175±0.0006	466.8107±2.4427	0.012389±0.001203	63k
B	500	0.1175±0.0004	466.7641±1.6048	**0.011858**±0.001679	63k
Triple	500	0.1389±0.0011	551.3837±4.3870	0.014847±0.000983	96k
Γ(Q)	500	0.1047±0.0004	415.7515±1.6226	0.022394±0.007181	124k
P	5000	0.1580±0.0005	627.0745±1.8492	0.007743±0.004762	62k
A	5000	0.1598±0.0003	634.4831±1.0636	**0.004761**±0.000218	63k
B	5000	0.1593±0.0005	632.4882±1.9462	0.012449±0.003293	63k
Triple	5000	0.1929±0.0005	765.8933±2.1952	**0.005563**±0.000459	96k
Γ(Q)	5000	0.1349±0.0007	535.6175±2.8578	0.008457±0.000651	124k

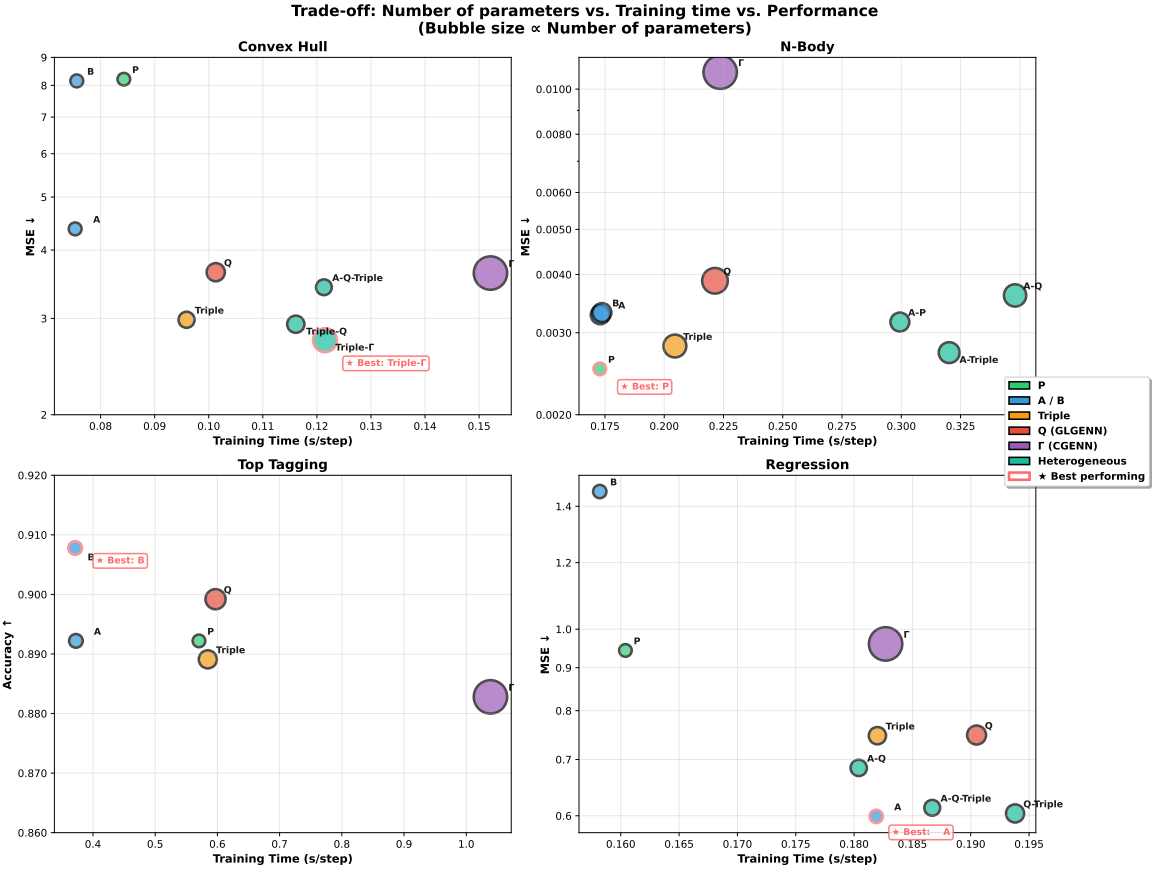


Figure 11: Trade-off: Number of parameters vs. Training time vs. Performance (Bubble size Number of parameters)

These results confirm that the inductive bias provided by widening the equivariance group is a more effective regularizer than simply adding L2 penalties to a highly expressive model. Note that the structural constraint not only reduces overfitting but does so with a fraction of the parameters.

Table 14: Regularization alternatives on $\mathrm{O}(6)$-N-Body (MSE ↓, mean ± std).

GROUP	WEIGHT DECAY	# TRAIN SAMPLES			# PARAMS
		50	500	5000	
P (best)	–	**0.0025**±0.0004	0.0009±0.0002	0.0004±0.0001	13k
A-Triple (best)	–	0.0027±0.0005	**0.0008**±0.0001	**0.0003**±0.00004	63k
Γ	–	0.0109±0.0048	0.0026±0.0007	0.0005±0.0001	185k
Γ	10^{-2}	0.0028±0.0008	0.0020±0.0001	0.0018±0.0005	185k
Γ	10^{-4}	0.0070±0.0015	0.0013±0.0002	0.0004±0.0001	185k
Γ	10^{-5}	0.0123±0.0053	0.0017±0.0003	0.0004±0.0001	185k

J LIMITATIONS

The limitations of the current work are mostly shared with existing GA-based neural networks: a) Without modifications, the current GA networks are suitable for geometric or physical data (for example, point clouds, molecular structures, particle physics data), but further research is needed to extend the framework to other domains (images, texts, etc.). b) The dimension of GA grows as 2^n, making high-dimensional spaces computationally expensive. The framework is best suited for vector spaces of dimension $n \leq 10$ (for $\mathrm{O}(p, q)$ with $p + q = n \leq 10$) common in natural sciences. c) The trade-off between the group (P, A, B, Q, Γ) and task performance is currently guided by heuristics; systematic selection remains an open question. The configuration-selection guidelines in Table 5 and Figure 5 are derived from the five benchmarks studied here and have not been validated as transferring to other task types. d) For the atom motion prediction task (MD17), the proposed models were trained with up to 10^4 steps, whereas several competing baselines used 10^5 steps. This discrepancy is due to limited computational resources.