Full Equivariant Feature Extraction for Multivectors in Clifford Group Equivariant Neural Networks

1093084

Abstract

We propose a general, learnable framework for multivector feature extraction in Clifford Group Equivariant Neural Networks (CGENNs), which addresses a key limitation in existing models. Our method—termed Full Equivariant Feature Extraction for Multivectors (FEFE)—leverages grade-wise projections and geometric products to unify and extend previous ad-hoc approaches. We demonstrate improved performance across three benchmark tasks, including scalar and vector predictions.

1 Introduction

Geometric deep learning is grounded on the idea that neural networks should respect the inherent symmetries of a given problem [1]. These constraints provide a powerful *inductive bias* within tasks across engineering and science [2] [3] [4] [5] [6]. Symmetries can be modelled theoretically using group theory and geometric deep learning is concerned with neural networks that are either *equivariant* or *invariant*—respectively meaning that output transforms in the same way as input or not at all—under a group action [1]. When a problem involves physical data, in fields ranging from molecular dynamics [2] and crystallography [7], to robotics [8] and particle-physics [9], it is particularly common to deal with symmetries of rotation, reflection and translation, technically referred to as the *Euclidean group* $\mathrm{E}(3)$. Depending on the problem, it might also be natural to consider any of the subgroups $\mathrm{SE}(3)$ (translations and rotations), $\mathrm{O}(3)$ (rotations and reflections) or $\mathrm{SO}(3)$ (rotations). For dealing with such symmetry groups, the preferred architecture is *Geometric Graph Neural Networks* (Geometric GNNs)—a graph structure that respects the symmetries of 3D space [10].

A related framework, which is particularly promising due to its potential for unifying and advancing the field of Geometric GNNs, is a class of architectures based on the theory of *Clifford algebras* [10] [11] [12]. This construction offers a natural language for representing geometric quantities and provides a rich family of equivariant models. In particular [11] introduces *Clifford Group Equivariant Neural Networks* (CGENNs), a general class of networks that are equivariant under orthogonal transformations (i.e. rotations and reflections). This reduces to O(3)-equivariance in the context of physical data, but can also be extended to handle $\mathrm{E}(3)$. While specialised architectures for dealing with $\mathrm{E}(3)$ exists [8] [13], we choose to consider CGENNs here due to their generality and subsequent potential for unification.

The elements of a Clifford algebra are referred to as multivectors, and naturally encode a hierarchy of geometric information ranging from points, to areas, volumes and beyond [11]. The different scales within this hierarchy is formally referred to as the *grades* of the multivector. While [11] demonstrates the effectiveness of Clifford algebras for geometric and physical problems, it does not provide a principled approach for extracting task-specific information from multivectors. Instead, it relies on ad-hoc methods to transform multivector representations into output features. Nevertheless, feature extraction from multivectors is a central step in producing the output of a CGENN, so calls for a principled treatment. We propose a unified, *multivector feature extraction* framework for CGENNs and show that it improves performance across three benchmark tasks, including both scalar and vector prediction.

2 Background

2.1 Theoretical Background

We begin by providing a theoretical background of Clifford algebras and present results that are relevant for equivariant transformations and feature extraction from multivectors. The following exposition is adapted from [11] and [14].

Definition 1. *Let V be a vector space over a field $\mathbb{F}$, and let $\mathfrak{q} : V \to \mathbb{F}$ be a quadratic form on V.* The Clifford algebra $\mathrm{Cl}(V, \mathfrak{q})$ over V with quadratic form $\mathfrak{q}$ *is the universal algebra generated by V, under the constraint that $v^2 = \mathfrak{q}(v)$ for all $v \in V$. The elements of* $\mathrm{Cl}(V, \mathfrak{q})$ *are called* multivectors *and the product of this algebra is referred to as the* geometric product.

For the remainder of the paper we shall assume that $\dim V = n \in \mathbb{N}$. We associate with V a bilinear form $\mathfrak{b} : V \times V \to \mathbb{F}$, given by

$$\mathfrak{b}(v, w) = \frac{1}{2}[\mathfrak{q}(v + w) - \mathfrak{q}(v) - \mathfrak{q}(w)]$$

for each $v, w \in V$. The following result is known as *the fundamental Clifford identity*.

Lemma 1. *For any $v, w \in V$ we have that*

$$vw + wv = 2\mathfrak{b}(v, w).$$

In particular, if v and w are orthogonal, then $vw = -wv$.

Every element $x \in \mathrm{Cl}(V, \mathfrak{q})$ can be written as a linear combination of formal and non-commutative products:

$$x = \sum_{i \in I} x_i v_{i1}..v_{ik_i}, \quad x_i \in \mathbb{F}, \quad v_{ij} \in V, \quad |I| < \infty$$

modulo the constraint mentioned in Definition 1. Using this presentation we can define the *main anti-involution* β of $\mathrm{Cl}(V, \mathfrak{q})$, by $\beta(x) = \sum_{i \in I} x_i v_{ik_i}...v_{i1}$. Given a basis $\{e_1, ..., e_n\}$ of V and $k \in \mathbb{N}$ we can also define the *grade-k part* $\mathrm{Cl}^{(k)}(V, \mathfrak{q})$ of $\mathrm{Cl}(V, \mathfrak{q})$ as

$$\mathrm{Cl}^{(k)}(V, \mathfrak{q}) := \text{span}\left\{e_A : A \subset [n],\ |A| = k\right\},$$

where we define $e_A := \prod_{i \in A}^{<} e_i$ with the product taken in increasing order. A theoretical result tells us that this subset is intrinsic, in the sense that it is invariant of the choice of basis. We now present a result of central importance to the construction of CGENNs.

Theorem 1. *The Clifford algebra* $\mathrm{Cl}(V, \mathfrak{q})$ *decomposes as a direct sum of grade parts:*

$$\mathrm{Cl}(V, \mathfrak{q}) = \bigoplus_{k=1}^{n} \mathrm{Cl}^{(k)}(V, \mathfrak{q}),$$

i.e. each $x \in \mathrm{Cl}(V, \mathfrak{q})$ can be written uniquely as $x = x^{(0)} + ... + x^{(n)}$, where $x^{(k)} \in \mathrm{Cl}^{(k)}(V, \mathfrak{q})$.

As a bonus, Theorem 1 enables us to extend the bilinear and quadratic forms of V to $\mathrm{Cl}(V, \mathfrak{q})$ via

$$\bar{\mathfrak{b}}(x, y) := (\beta(x)y)^{(0)}, \quad \bar{\mathfrak{q}}(x) := \bar{\mathfrak{b}}(x, x)$$

for all $x, y \in \mathrm{Cl}(V, \mathfrak{q})$. The previous direct-sum decomposition is then an orthogonal-sum decomposition with respect to $\bar{\mathfrak{b}}$. Furthermore, if $\{e_i\}_{i=1}^n$ is an orthonormal basis of V with respect to $\mathfrak{b}$ then $\{e_A\}_{A \subset [n]}$ is an orthonormal basis of $\mathrm{Cl}(V, \mathfrak{q})$ with respect to $\bar{\mathfrak{b}}$. The next result will also be helpful.

Lemma 2. *For any $x \in \mathrm{Cl}(V, \mathfrak{q})$ and $i, j \in [n]$ we have that*

$$x^{(i)} x^{(j)} = \left(x^{(i)} x^{(j)}\right)^{(|i-j|)} + \left(x^{(i)} x^{(j)}\right)^{(|i-j|+2)} + ... + \left(x^{(i)} x^{(j)}\right)^{(i+j)}$$

i.e. $x^{(i)} x^{(j)}$ has contributions only from $\mathrm{Cl}^{(k)}(V, \mathfrak{q})$*, where $k = |i - j|, |i - j| + 2, ..., i + j$.*

With all the relevant notation and geometric operations in place we are now ready to relate to the orthogonal group $\mathrm{O}(V, \mathfrak{q})$ and present the result that will enable us to produce equivariant and invariant models.

Theorem 2. *The orthogonal group $\mathrm{O}(V, \mathfrak{q})$ acts in a well-defined way on the whole of $\mathrm{Cl}(V, \mathfrak{q})$, and for $\Phi \in \mathrm{O}(V, \mathfrak{q})$, $x, y \in \mathrm{Cl}(V, \mathfrak{q})$, $c \in \mathbb{F}$, $k \in [n]$ the following properties are satisfied:*

$$\begin{aligned}
\textit{Additivity:}\quad & \Phi(x+y) = \Phi x + \Phi y \\
\textit{Multiplicativity:}\quad & \Phi(xy) = (\Phi x)(\Phi y) \text{ and } \Phi c = c \\
\textit{Orthogonality:}\quad & \bar{\mathfrak{b}}(\Phi x, \Phi y) = \bar{\mathfrak{b}}(x, y) \\
\textit{Grade-respecting:}\quad & \Phi\left(x^{(k)}\right) = (\Phi x)^{(k)}
\end{aligned}$$

2.2 Further Theoretical Results

We now present our own theoretical results, which will be relevant for the development of our method. In the following, let $x, y \in \mathrm{Cl}(V, \mathfrak{q})$ be arbitrary multivectors and suppose $i, j, k \in [n]$, $A, B \subset [n]$ and that $\{e_i\}_{i=1}^n$ is an orthonormal basis for V. With the notation introduced above, we shall also write

$$x = \sum_{A \subset [n]} x_A e_A, \quad y = \sum_{A \subset [n]} y_A e_A,$$

where $x_A, y_A \in \mathbb{F}$ for all $A \subset [n]$. For the purpose of proving upcoming results, let us introduce two helpful notations. First, we denote by $\#k$ the number of transposition needed to reverse a list of k elements. Secondly, for $k \notin A$ we let $\mathrm{idx}(k; A)$ be the index position of k in the sorted list of elements in $A \cup \{k\}$ (starting at index zero). We then have that

$$\beta(x^{(i)})y^{(j)} = \sum_{|A|=i} \sum_{|B|=j} x_A y_B \beta(e_A) e_B = (-1)^{\#i} \sum_{|A|=i} \sum_{|B|=j} x_A y_B e_A e_B = \pm x^{(i)} y^{(j)}, \tag{1}$$

which follows directly from Lemma 1, and the definition of β. The next result highlights another symmetry of the geometric product.

Lemma 3. *For all $i, j \in [n]$ and $x, y \in \mathrm{Cl}(V, \mathfrak{q})$, we have that $\left(x^{(i)} y^{(j)}\right)^{(1)} = \pm \left(y^{(j)} x^{(i)}\right)^{(1)}$.*

Proof. By Lemma 2 we have that $\left(x^{(i)} y^{(j)}\right)^{(1)} = 0$ if $|i - j| \neq 1$, in which case the assertion is clear. Without loss of generality, we may thus assume that $i = j + 1$. By Lemma 1, and the fact that $\{e_A\}_{A \subset [n]}$ is orthonormal with respect to $\bar{\mathfrak{b}}$, we have that

$$\left(x^{(i)} y^{(j)}\right)^{(1)} = \sum_{|A|=i} \sum_{k \notin A} x_{A \cup \{k\}} y_A e_{A \cup \{k\}} e_A = (-1)^{\#i} \sum_{|A|=i} \sum_{k \notin A} (-1)^{\mathrm{idx}(k;A)} x_{A \cup \{k\}} y_A e_k.$$

Similarly, we have that

$$\left(y^{(j)} x^{(i)}\right)^{(1)} = \sum_{|A|=i} \sum_{k \notin A} x_{A \cup \{k\}} y_A e_A e_{A \cup \{k\}} = (-1)^{\#i} \sum_{|A|=i} \sum_{k \notin A} (-1)^{i - \mathrm{idx}(k;A)} x_{A \cup \{k\}} y_A e_k.$$

Using these two expressions, we may conclude that $(-1)^i \left(x^{(i)} y^{(j)}\right)^{(1)} = \left(y^{(j)} x^{(i)}\right)^{(1)}$ and the assertion follows. $\square$

2.3 Architecture

We now show how [11] leverages Theorem 2 to produce a rich and expressive family of models that exhibit equivariance with respect to $\mathrm{O}(V, \mathfrak{q})$. In the following, we let l_{in} and l_{out} be some number of layer-wise input- and output-channels. As we have seen, (elementwise) orthogonal transformations $\mathrm{Cl}(V, \mathfrak{q})^{l_{\text{in}}} \to \mathrm{Cl}(V, \mathfrak{q})^{l_{\text{out}}}$ are additive and respect grade projections. With this in mind [11] introduces an equivariant *multivector-linear layer* given by

$$T_\phi^{\text{lin}}(x)_i := \sum_{j=1}^{l_{\text{in}}} \sum_{k=1}^{n} \phi_{ijk} x^{(k)}, \quad i = 1, ..., l_{\text{out}}$$

for $x \in \mathrm{Cl}(V, \mathfrak{q})$, where $\phi = (\phi_{ijk})$ is some tensor of parameters. Additionally, we can also learn a scalar bias term. A big part of the expressivity in these models comes from the fact that they can incorporate geometric products, due to the multiplicativity of orthogonal transformations in the Clifford algebra. We can thus introduce an equivariant parametrized geometric product:

$$P_\phi(x, y) := \sum_{i=1}^{n} \sum_{j=1}^{n} \sum_{k=1}^{n} \phi_{ijk} \left(x^{(i)} y^{(j)} \right)^{(k)}$$

for $x, y \in \mathrm{Cl}(V, \mathfrak{q})$. Given an input with l_{in} channels, taking all combinations of products would require l_{in}^2 applications of P_ϕ, which can quickly lead to high computational costs. For this reason [11] argues to first apply a linear layer and then produce the channel-wise geometric products between the original multivector and the linear output. We can interpret this as the linear layer learning the most important geometric products. This reasoning prompts us to introduce the following *fully connected geometric product layer*:

$$T_\phi^{\text{prod}}(x)_i = \sum_{j=1}^{l_{\text{in}}} P_{\phi_{ij}} \left(x_j, (T_{\phi_0}^{\text{lin}} x)_j \right), \quad i = 1, ..., l_{\text{out}},$$

which is equivariant to $\mathrm{O}(V, \mathfrak{q})$. We may also consider a less expensive, yet less expressive *element-wise geometric product layer* by setting $l_{\text{in}} = l_{\text{out}}$ and $\phi_{ij} = 0$ for $i \neq j$. Lastly, it will be useful to introduces an equivariant *gated non-linearity* via the mapping $x^{(k)} \mapsto \sigma(\bar{\mathfrak{q}}(x^{(k)}))x^{(k)}$, where σ is some non-linear scalar-function.

3 Limitations and motivation

As discussed in the introduction, the main objective of equivariant networks—such as CGENNs—is to model physical data in a symmetry-aware way. In practice, this often involves predicting scalar or vector quantities. Within the framework of Clifford algebras, this corresponds to the grade-zero and grade-one components of multivectors. In practice CGENNs thus work by first embedding scalar or vector data into a corresponding Clifford algebra $\mathrm{Cl}(V, \mathfrak{q})$ before passing this as input to any combination of the layers introduced in Section 2.3.

While higher-grade components are vital for the model's expressivity, the final output must ultimately be converted into task-specific features (typically scalars or vectors). This process, which we refer to as *multivector feature extraction*, is both essential and non-trivial.

In [11], this step is handled using task-specific and unexplained heuristics, differing across experiments. This inhibits benchmarking, and can harm performance (as we will show). A general approach to multivector feature extraction, that generalises existing methods and performs well across tasks, would lead to a more universal CGENN architecture and enable principled benchmarking of task-specific variants.

Our contributions are as follows:

- We introduce a unified feature extraction framework for multivectors in CGENNs, that generalizes the ad-hoc methods from [11] in a learnable way.
- We demonstrate improved performance on three benchmark tasks spanning scalar and vector prediction.

4 Method

4.1 Framework

Our goal is to provide a unified, learnable, and interpretable approach to extracting task-relevant features from multivector representations in CGENNs. Recall that the output of an CGENN-layer is a multivector x, which can be decomposed into grade parts: $x = x^{(0)} + ... + x^{(n)}$. To give some background, we will first present the approaches to scalar feature extraction that were used in [11].

One straightforward method from the original paper is to simply project onto the grade-zero component, i.e. to extract $x^{(0)}$. However, this overlooks geometric information from higher grades, which can be valuable, depending on what we want to predict. Hence, in another experiment, more focused towards volume prediction, the full norm $\sqrt{\bar{\mathfrak{q}}(x)}$ is used as a scalar feature. If we want a slightly more informative feature extraction, we could split the previous approaches into grades, to extract the collection $\{x^{(0)}, \bar{\mathfrak{q}}(x^{(1)}), ..., \bar{\mathfrak{q}}(x^{(n)})\}$, which is actually the most general method used in [11]. Recall from Theorem 2 that all of the aforementioned quantities are indeed invariant. Looking back at the definition of $\bar{\mathfrak{q}}$ and taking equation (1) into account, we are really interested in scalars of the form $\left(x^{(i)}x^{(j)}\right)^{(0)}$ for $i, j \in [n]$. With this perspective, we are no longer restricted to extracting scalars: by simply changing the grade-projection we can produce equivariant data of any grade. This line of reasoning prompts us to make the following central definition.

Definition 2. *The* grade-k equivariants *of a multivector* $x \in \mathrm{Cl}(V, \mathfrak{q})$ *is defined as the set:*

$$\mathrm{EQ}_k(x) := \left\{x^{(k)}\right\} \cup \left\{\left(x^{(i)}x^{(j)}\right)^{(k)} : i, j \in [n]\right\}.$$

Note that equivariance is equivalent to invariance for scalar output (see Theorem 2), so this terminology makes sense. The grade-k equivariants is essentially the primitive collection of grade-k data that can be derived from a multivector, through grade-projections and geometric products. We view $\mathrm{EQ}_k(x)$ as a general feature set—a collection of grade-specific quantities derived from geometric interactions between components of x, from which task-relevant predictions can be learned. We base our feature extraction framework on this definition, and refer to it as *Full Equivariant Feature Extraction for Multivectors* (FEFE).

4.2 Practical Considerations and Implementation

In practice, we are typically interested only in scalar or vector data. For these purposes, the general presentation in Definition 2 may be a bit misleading. First of all, because the information in $(x^{(0)})^2$ is already contained in $x^{(0)}$ we exclude this term from the grade-zero equivariants. Second of all, Lemma 3 implies that half of the grade-one equivariants are redundant. Third of all, according to Lemma 2, the geometric product is sparse in the sense that only certain grades contribute to certain other grades, which is especially true for grades zero and one. For the purpose of efficiency, we take this into account when implementing our framework. In practice, we thus use

$$\widetilde{\mathrm{EQ}}_0(x) := \left\{x^{(0)}\right\} \cup \left\{\left(x^{(i)}x^{(i)}\right)^{(0)} : 1 \leq i \leq n\right\}, \tag{2}$$

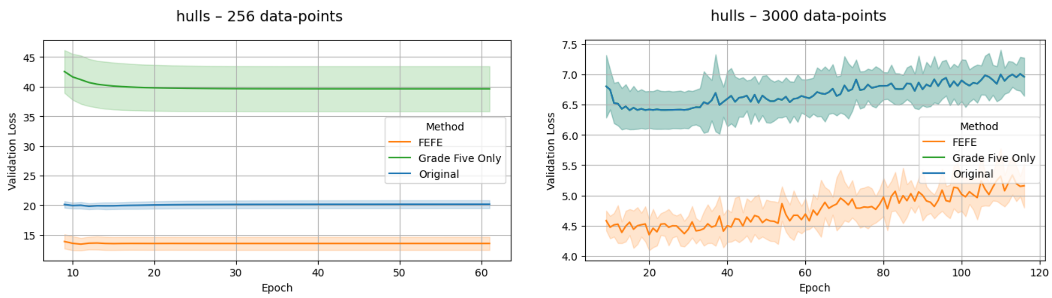
and

$$\widetilde{\mathrm{EQ}}_1(x) := \left\{x^{(1)}\right\} \cup \left\{\left(x^{(i-1)}x^{(i)}\right)^{(1)} : 1 \leq i \leq n\right\}.$$

For brevity, we shall refer to these also as the grade-zero and grade-one equivariants, respectively. Our next consideration regards how to combine the extracted data into a single final output. Recall first that scalar values are in fact invariant (while vectors and higher grades are equivariant). A consequence of this is that, after extracting grade-zero features, we are free to combine the results in any way without losing invariance. In the scalar case we thus propose to take the channel-wise grade-zero equivariants $\widetilde{\mathrm{EQ}}_0$ and then pass the result through a multi-layer perceptron (MLP) $\mathbb{R}^{l \times n} \to \mathbb{R}$. The non-scalar case is more complicated. Because the equivariants of higher grades are exactly that—equivariant and not invariant—we must combine them in an equivariant way. The best approach we found was to first collapse the multivector output into one channel through an equivariant multivector-linear layer, before taking the equivariants, and combining these linearly into one output of the relevant grade. We also consider applying an equivariant gated non-linearity aggregated data (before a final scaling). Our reasoning was that a non-linearity could make the model more expressive and provide an equivariant analogue of the MLP in the scalar-valued case. As we shall see below, this seems to improve performance slightly. We stress that the aggregation mechanism has learned parameters, giving an automatic and adequate conversion from general to task-specific features through training. In fact, all the manual extraction techniques used in [11], can be recovered through learned parameters using the above framework. For example, the scalar projection $x^{(0)}$,

Model	256 datapoints	3000 datapoints
Grade Five Only	39.6 ± 3.793	6.28 ± 0.343
Original	19.8 ± 0.562	6.28 ± 0.343
FEFE (Ours)	13.2 ± 1.102	4.21 ± 0.180
Improvement	33.5%	33.1%

Table 1: Average MSE and model improvement (from original model) for convex hulls volume estimation. Bounds correspond to one standard deviation of MSE.



(a) 256 data-points

(b) 3000 data-points

Figure 1: Validation loss histories for convex hulls volume estimation.

used in [11] corresponds to selecting only the first-order term in $\widetilde{\mathrm{EQ}}_0(x)$, while the norm $\sqrt{\bar{\mathfrak{q}}(x)}$ can be recovered via an MLP over the set of grade-zero equivariants in (2). Lastly, the only extraction mechanism for vector data used in [11] was the grade-one projection, which can be recovered through a linear combination of the grade-one invariants (namely just the first-order component). This concludes our discussion on a principled, general, and learnable framework for multivector feature extraction in CGENNs.

5 Experiments

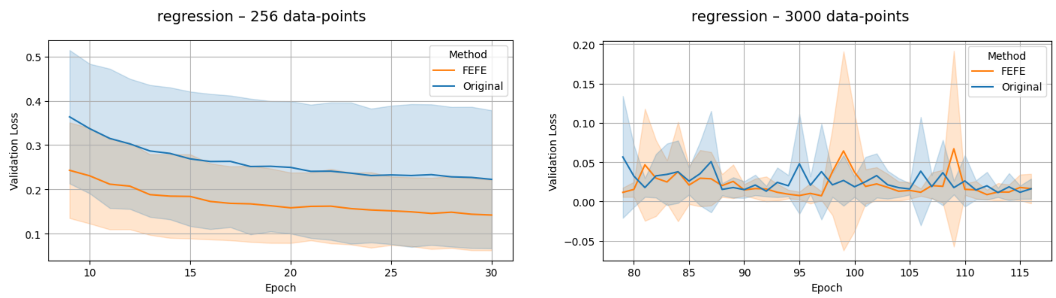
In this section, we present three numerical experiments that were originally examined in [11]. For all of these, we test our framework in the low and high data regime (256 and 3000 data-points respectively) and benchmark against the original models used in [11]. As a metric for performance, we monitor the lowest validation mean squared error (MSE) achieved throughout training. For all experiments, we run our model on 8 separate random seeds and present our results in terms of averages and standard deviations. We additionally plot the tail of the validation loss histories to assess convergence behaviour. Our implementation is heavily based on the code available in [11].

5.1 Experiment 1: Convex Hulls

This experiment deals with scalar prediction from five dimensional euclidean data. The task is to estimate the volume of a convex hull generated by a collection of points in $\mathbb{R}^5$. For this, we use a $\mathrm{O}(5)$-equivariant model, which is elegantly just a special case of the CGENN architecture. The original paper [11] particularly reported issues in the low data regime for this problem. In the original model, the multivector norm is used for scalar extraction. If the output multivector is a pure grade-five element, its norm corresponds to a five-dimensional volume [15]. Motivated by this interpretation, we also evaluated a simplified model—referred to as *Grade Five Only*—which uses only the norm of the grade-five component as the scalar predictor. In the low-data regime, this model is less flexible and performs worse than the original one. However, Figure 1b shows that Grade Five Only converges toward the same solution as the original model as more data becomes available. This is a noteworthy observation, suggesting that CGENNs may naturally learn to encode volume via higher-grade representations. Nevertheless our framework, leads to considerable improvement in both data regimes, which shows that, even when the predicted quantity is highly geometric, scalar values should not be extracted in the most straightforward and interpretable way. Numerical results

Model	**256 datapoints**	**3000 datapoints**
Original	0.219 ± 0.1555	0.00794 ± 0.00504
FEFE (Ours)	0.137 ± 0.0746	0.00552 ± 0.00402
Improvement	37.55%	33.1%

Table 2: Average MSE and model improvement for regression task. Bounds correspond to one standard deviation of MSE.



(a) 256 data-points

(b) 3000 data-points

Figure 2: Validation loss histories for regression task.

are presented in Table 1. Contrary to other tests, we observe an increased variance for our model in the low data regime of this experiment.

5.2 Experiment 2: Regression

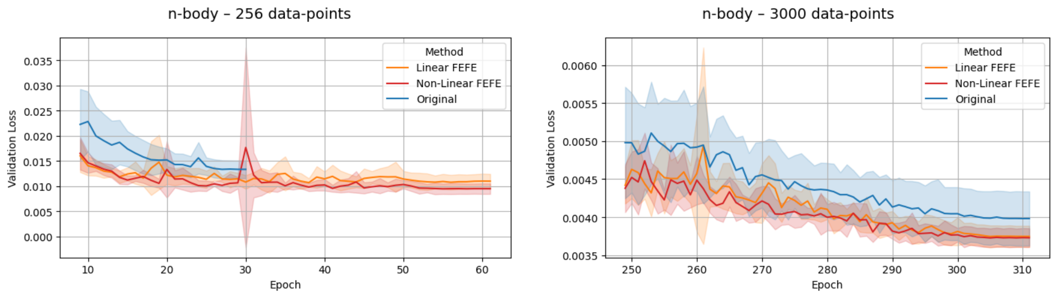
Just as the previous experiment, this test concerns scalar prediction from five dimensional euclidean data. The task is to estimate the function

$$f(x, y) = \sin(||x||) - \frac{1}{2}||y||^3 + \frac{x^T y}{||x|| \, ||y||}$$

where $x, y \in \mathbb{R}^5$ are sampled from a standard normal distribution. The original model used the grade-zero component of output multivectors as a scalar predictor. Once again, we report substantial improvement in both data regimes. Numerical results are presented in Table 2.

5.3 Experiment 3: n-Body System

Unlike the previous experiments, this test involves predicting vector-valued output. The n-body experiment is a common benchmark in geometric deep learning. It evaluates the performance of an equivariant neural network in predicting the dynamics of a complex multi-body system evolving through time. In this instance, we model five charged particles interacting in three-dimensional space and predict the state of this system after 1000 time-steps. For this task [11] uses a slightly different approach from previous experiments. While the other tests used an MLP-like architecture, comprised of stacked linear- and product-layers, a GNN-like structure is used in this case. Furthermore, in order to achieve translation-equivariance, the model operates on mean-subtracted input and predicts only a displacement vector (i.e. an update from the initial state). The result is an $\mathrm{E}(3)$-equivariant model. Both models collapse output multivectors into a single channel before extracting vector-data, but the original model used only the grade-one component of the output multivector as a vector predictor. Numerical results for this experiment are presented in Table 3. Note that [11] reported a mean squared error of 0.0039 for 3000 data-points—the same as our result. While we witness some performance boost in the high data regime, the improvement is less substantial compared to other experiments. We attribute this partly to the flexibility of invariant output, in that we may apply an MLP for combining channels and invariant quantitates. To obtain a more expressive equivariant model, we considered adding a gated non-linearity after aggregating into a single vector output. This non-linearity seems to improve performance slightly. It would thus be interesting to fur-



(a) 256 data-points (b) 3000 data-points

Figure 3: Validation loss histories for n-body experiment.

ther explore how different equivariant transformations may be used for aggregation in the non-scalar case, and in particular how non-linearities can be utilised.

Model	**256 datapoints**	**3000 datapoints**
Original	0.01307 ± 0.001836	0.00390 ± 0.000213
Linear FEFE	0.00973 ± 0.001075	0.00374 ± 0.000141
Non-Linear FEFE	0.00883 ± 0.000679	0.00371 ± 0.000122
Improvement (Linear)	25.6%	4.16%
Improvement (Non-Linear)	32.4%	4.86%

Table 3: Average MSE and model improvement for n-body experiment. Bounds correspond to one standard deviation of MSE.

6 Discussion

In this work, we addressed a conceptual and practical limitation of the CGENN framework: the lack of a general, principled method for extracting grade-specific output from multivectors. Previous methods relied on hand-crafted strategies tailored to specific tasks, lacking both consistency and theoretical justification. We introduced a unified framework for multivector feature extraction, termed FEFE (Full Equivariant Feature Extraction for Multivectors), built entirely from grade projections and geometric products. Notably, our framework generalises previous approaches in a learnable way and applies similarly to both scalar- and vector-valued tasks. We also investigated redundancy in the feature space through theoretical analysis, enabling a more efficient representation. Experimental results showed reduced validation error across three benchmark tasks—including both scalar and vector outputs—with the majority of improvements being substantial. In addition, our method generally led to lower variance across training runs, indicating more stable learning dynamics. Most notably, our approach achieves a 4.5% reduction in error on the n-body task. Future extensions could explore more general equivariant aggregations across channels—especially non-linear transformations—as our preliminary results show promise in this direction. Overall, we believe that principled feature extraction is a key missing component in current Clifford group equivariant architectures. Our results suggest that addressing this gap yields not only practical gains, but also invites deeper theoretical insights within the CGENN framework.

References

[1] Michael M Bronstein, Joan Bruna, Taco Cohen, and Petar Veličković. Geometric deep learning: Grids, groups, graphs, geodesics, and gauges. *arXiv preprint arXiv:2104.13478*, 2021.

[2] John Jumper, Richard Evans, Alexander Pritzel, Tim Green, Michael Figurnov, Olaf Ronneberger, Kathryn Tunyasuvunakool, Russ Bates, Augustin Žídek, Anna Potapenko, et al. Highly accurate protein structure prediction with alphafold. *nature*, 596(7873):583–589, 2021.

[3] Fabian Fuchs, Daniel Worrall, Volker Fischer, and Max Welling. Se (3)-transformers: 3d roto-translation equivariant attention networks. *Advances in neural information processing systems*, 33:1970–1981, 2020.

[4] Vıctor Garcia Satorras, Emiel Hoogeboom, and Max Welling. E (n) equivariant graph neural networks. In *International conference on machine learning*, pages 9323–9332. PMLR, 2021.

[5] Alexander Bogatskiy, Brandon Anderson, Jan Offermann, Marwah Roussi, David Miller, and Risi Kondor. Lorentz group equivariant neural network for particle physics. In *International Conference on Machine Learning*, pages 992–1002. PMLR, 2020.

[6] Haiwei Chen, Shichen Liu, Weikai Chen, Hao Li, and Randall Hill. Equivariant point network for 3d point cloud analysis. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 14514–14523, 2021.

[7] Jonas Köhler, Michele Invernizzi, Pim De Haan, and Frank Noé. Rigid body flows for sampling molecular crystal structures. In *International Conference on Machine Learning*, pages 17301–17326. PMLR, 2023.

[8] Johann Brehmer, Joey Bose, Pim De Haan, and Taco S Cohen. Edgi: Equivariant diffusion for planning with embodied agents. *Advances in Neural Information Processing Systems*, 36:63818–63834, 2023.

[9] Risi Kondor. N-body networks: a covariant hierarchical neural network architecture for learning atomic potentials. *arXiv preprint arXiv:1803.01588*, 2018.

[10] Alexandre Duval, Simon V Mathis, Chaitanya K Joshi, Victor Schmidt, Santiago Miret, Fragkiskos D Malliaros, Taco Cohen, Pietro Liò, Yoshua Bengio, and Michael Bronstein. A hitchhiker's guide to geometric gnns for 3d atomic systems. *arXiv preprint arXiv:2312.07511*, 2023.

[11] David Ruhe, Johannes Brandstetter, and Patrick Forré. Clifford group equivariant neural networks. *Advances in Neural Information Processing Systems*, 36:62922–62990, 2023.

[12] Johann Brehmer, Pim De Haan, Sönke Behrends, and Taco S Cohen. Geometric algebra transformer. *Advances in Neural Information Processing Systems*, 36:35472–35496, 2023.

[13] Christian Hockey, Yuxin Yao, and Joan Lasenby. Simplifying and generalising equivariant geometric algebra networks. 2024.

[14] Chris Doran and Anthony Lasenby. *Geometric algebra for physicists*. Cambridge University Press, 2003.

[15] Alan Macdonald. *Linear and geometric algebra*. Alan Macdonald Nottingham, 2010.