MSc Artificial Intelligence
Master Thesis

A Scalable Geometric Deep Learning Approach to Weather Forecasting

by
Coen van den Elsen
12744956

2025 – 2026
36 ECTS

Supervisor: D. Wessels

Examiner: Dr E. Bekkers

Universiteit van Amsterdam
Instituut voor Informatica

A Scalable Geometric Deep Learning Approach to Weather Forecasting

Abstract

Machine learning weather models increasingly match numerical weather prediction in skill, but not physical fidelity, and few studies ask how a model's treatment of geometry shapes either. We introduce Platonic-Erwin, a transformer architecture combining linearly-scaling hierarchical ball-tree attention with principled weight-sharing, enabling roto-translation group equivariance without sacrificing scalability. We validate Platonic-Erwin on a large-scale molecular-dynamics benchmark, where it outperforms prior equivariant and non-equivariant baselines. We run geometry ablations on global ERA5 weather forecasting, isolating grid topology, vector-valued wind-loss formulations, 2D vs. 3D modelling, positional-encoding design, and equivariance and weight sharing. Every geometric choice measurably shifts deterministic skill, spectral fidelity, or both, and in the equivariance case we find that geometric symmetry is valuable as an inductive bias but not as a hard constraint. Interpreting the learnable symmetry-breaking features across varying equivariance configurations, we find they recover physically recognisable structure. Supplying the corresponding static symmetry-breaking information as input features improves skill, except under strict roto-translation equivariance. Finally, we scale the best approximately equivariant configuration and a matched non-equivariant baseline across model size and fine-tune both for probabilistic forecasting: scale buys deterministic skill, whereas the inductive bias is what buys kinetic-energy spectral fidelity, calibration, and parameter efficiency. Code: github.com/CoenvdE/geometric-weather.

Chapter 1

Introduction

The frequency and intensity of extreme weather events have escalated under climate change, as the warming atmosphere holds more moisture and energy, intensifying floods, droughts, and storms [Masson-Delmotte et al., 2021]. This has rendered accurate atmospheric forecasting a critical pillar of global disaster preparedness and agricultural planning [United Nations, 2022]. Yet the capability to produce such forecasts is unevenly distributed, because traditional Numerical Weather Prediction (NWP) carries a prohibitive infrastructure cost: up to $8 million per radar installation and hundreds of millions for the national supercomputing facilities required to run the models [Ndlovu, 2026]. This concentrates forecasting power in the global north and leaves the regions most exposed to climate volatility least equipped to anticipate it. This inequality is visible across the observational network as well, where the Global South averages roughly 33 weather stations per million inhabitants against 600 in the north [Ndlovu, 2026], a disparity laid bare by the global distribution of land surface stations (Figure 1.1). Hardware-efficient AI models offer a democratising solution to the lack of supercomputing facilities, replacing expensive supercomputing hours with seconds of GPU inference and achieving a 2000–4545× cost reduction relative to conventional infrastructure [Ndlovu, 2026]. This makes the development of specialised, resource-efficient AI weather models an urgent priority.

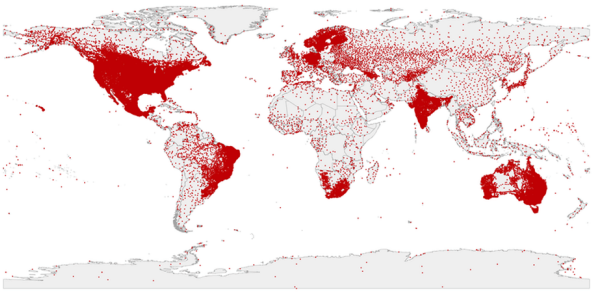


Figure 1.1: Spatial distribution of weather stations in the Global Historical Climatology Network in 2020. Figure from Ortiz-Bobea [2021].

Machine learning weather prediction (MLWP) models, including GraphCast [Lam et al., 2023], Pangu-Weather [Bi et al., 2022], and Aurora [Bodnar et al., 2025], have demonstrated the potential to rival or exceed traditional NWP systems in both accuracy and speed. Their skill, however, comes with characteristic failure modes that trace back to a common root. Traditional NWP encodes the governing physics by construction; MLWP models replace that built-in structure with learning, and how much structure to retain, in the objective, in the architecture, or in the representation of the data, becomes a design choice. Where structure is absent, the learning fills the gap in statistically convenient but physically unfaithful ways. The clearest example is the 'double penalty' effect of the MSE objective: a spatially displaced feature is penalised twice, once for its absence at the true location and once for its presence at the wrong one, so the network minimises error by predicting a blurred conditional mean rather than sharp, physically plausible extremes [Subich et al., 2025]. This spectral blurring systematically suppresses fine-scale structure and erodes the model's effective resolution. Likewise, the absence of structural physical constraints lets models violate global conservation laws, surfacing as drizzle bias and long-term instability in autoregressive rollouts [Sha et al., 2025].

The field has responded to these deficiencies along two main fronts. The first reworks the training objective: probabilistic targets such as the Continuous Ranked Probability Score (CRPS) and flow-matching for ensemble generation [Lang et al., 2026, Alet et al., 2025, Couairon et al., 2024], and physics-aligned losses that constrain global mass and energy budgets [Dunstan et al., 2025, Sha et al., 2025]. The second couples learning to physics directly, as in the differentiable fluid solver of NeuralGCM [Kochkov et al., 2024] or the continuous-time formulation of WeatherODE [Liu et al., 2024]. These advances sharpen forecasts and mitigate conservation drift, but they share a blind spot: each refines the objective or bolts on a solver while treating the model's representation of the data as fixed; the loss is examined, the geometry is not. A second cost compounds the issue: the long training cycles of these models throttle experimental iteration and raise the barrier to entry, slowing progress for the broader community [Diaconu et al., 2026].

This geometric blind spot has physical consequences. By Noether's theorem, every continuous symmetry of physical system's action corresponds to a conserved quantity: the fundamental conservation laws are the mathematical consequences of translation, rotation and time symmetry [Noether, 1918]. A model that misrepresents these symmetries therefore discards the very structure from which conservation follows. Many state-of-the-art models do exactly this: they treat spherical weather fields as pseudo-planar images and vector quantities as unstructured scalars, respecting neither the geometry of the data nor its symmetries [Linander et al., 2025, Ballerin et al., 2025]. As a result, these models do not naturally learn a single, universal set of physical laws. For instance, when the Earth is flipped and passed to NeuralGCM, the model produces nonphysical artefacts such as phantom mountains where none exist [Stier, 2025]. By discarding these geometric inductive biases, a network forfeits the very symmetry constraints that govern atmospheric dynamics and forgoes the efficiency gains that such biases are known to confer [Brehmer et al., 2024]. Recent work has begun to bring geometry into weather modelling [Bonev et al., 2025, Liu et al., 2025b], but these approaches neither *isolate* the effect of their geometric choices (which remain confounded with other architectural changes) nor *interpret* what these geometric choices lead the model to learn. This thesis sets out to do both.

In this thesis, we investigate how geometric inductive biases shape weather-model performance and physical fidelity. Rather than reading these biases off as a by-product of architectural comparisons, we isolate each design choice one at a time. We aim to move toward weather models that are not only accurate

but parameter-efficient and interpretable. Our contributions are four-fold:

- **Platonic-Erwin Architecture.** We propose and validate a transformer architecture that enables group equivariance at scale, combining the linear scalability of hierarchical ball-tree attention [Zhdanov et al., 2025] with the principled group-equivariant framework of Platonic Transformers [Islam et al., 2025]. On a large-scale protein molecular-dynamics benchmark, it outperforms prior equivariant and non-equivariant baselines.
- **Controlled geometry ablations for weather data.** We isolate the effect of grid topology (equiangular vs. equal-area HEALPix), vector-loss formulation, 2D vs. 3D modelling, positional-encoding frequency schemes, and strict and approximate equivariance. Each choice measurably shifts deterministic skill, spectral fidelity, or both, and in the equivariance case we find that geometric symmetry is valuable as an inductive bias, but not as a hard constraint. Together these show that geometry is a design axis in its own right, separable from the architectural changes it is usually bundled with.
- **Interpreting learned symmetry breaking.** We provide a visual interpretation of the symmetry-breaking features across four degrees of equivariance, finding that equivariant models recover structure intrinsic to the Earth: Coriolis-like antisymmetry, insolation banding, orography, and general circulation/jet-stream banding. We show that these learned features improve short-range deterministic skill across all four, but purchase that skill at a cost in spectral fidelity (blurring). Injecting static symmetry-breaking information improves skill, with the exception of strict roto-translation equivariant modelling.
- **Stress-testing approximate equivariance.** We take the winning approximately equivariant configuration and its matched non-equivariant baseline, scale both across model size, and fine-tune both for probabilistic forecasting. We find that scale buys deterministic skill, whereas the inductive bias is what buys kinetic-energy spectral fidelity, calibration, and parameter efficiency.

Chapter 2

Related Work

2.1 Geometric Deep Learning in Weather Prediction

In order to contextualise our framework, we categorise existing literature into two primary branches of Geometric Deep Learning (GDL): geometry and equivariance.[1] The former focuses on the impact of the geometric representation of the data, while the latter examines symmetry-based inductive biases and principled weight sharing (reusing a single set of filter weights across all rotated, reflected, or translated copies of a feature, rather than learning each separately).

2.1.1 Geometry in Weather Prediction

Geometry at the points - vectors: Many methods treat vector data as separate scalars, introducing bias in wind speed and kinetic energy. To quantify the bias in wind magnitude, we apply Jensen's Inequality to the L_2 norm. Given that the Euclidean norm $f(\mathbf{v}) = \sqrt{u^2 + v^2}$ is a convex function, Jensen's Inequality states:

$$\mathbb{E}[\sqrt{u^2 + v^2}] \geq \sqrt{\mathbb{E}[u]^2 + \mathbb{E}[v]^2} \tag{2.1}$$

This implies that a model predicting the conditional mean of the components ($\mathbb{E}[u], \mathbb{E}[v]$), as is typical with Mean Squared Error (MSE) loss, will systematically underestimate the expected wind speed $\mathbb{E}[S]$. FastNet [Dunstan et al., 2025] explicitly addresses this. It identifies that modelling wind as separate u and v scalars results in a "slow bias" where the model minimises component-wise RMSE by reducing predicted magnitudes. However, their framework lacks a systematic ablation of alternative loss formulations, and the specific magnitude weighting factor of 5 appears empirically arbitrary.

Geometry of the space - grid: The representation of global atmospheric states on an equiangular planar grid introduces significant geometric distortions, particularly as one approaches the poles. Standard equirectangular (latitude-longitude) grids suffer from an uneven distribution of information density, since the convergence of meridians over-samples grid points at high latitudes, introducing a latitude-dependent geometric bias (Figure 2.1).

A common technique to manage the unequal area of latitude-longitude cells is area-weighting the loss function, where each grid point's contribution is scaled by the cosine of its latitude [Lam et al., 2023, Lang et al., 2026]. GraphCast [Lam et al., 2023] addresses polar bias architecturally, mapping the input latitude-longitude grid internally to a spatially homogeneous "multi-mesh" representation, ensuring a uniform resolution across the globe. CirT [Liu et al., 2025a] mitigates spherical geometric distortion through Circular Patching, which partitions data latitudinally to maintain uniform information density.

[1] Throughout, "equivariant" refers to the spatial symmetry groups under study, rotations and translations, not to the permutation equivariance inherent to attention.

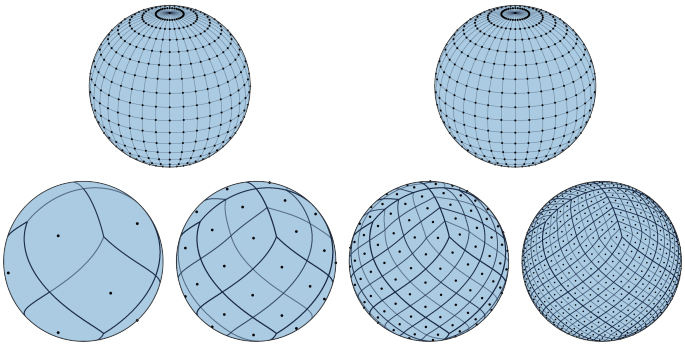


Figure 2.1: Comparison of different spherical grid discretisations. **Top:** the equiangular latitude–longitude grid places the same number of points in every latitude band, so its nodes crowd together towards the poles, over-sampling high latitudes. **Bottom:** the HEALPix grid, shown at increasing resolutions $n_{\text{side}} \in \{1, 2, 4, 8\}$, is equal-area by construction and distributes its nodes uniformly across the sphere at every resolution.

Traditional Numerical Weather Prediction (NWP) systems, such as the European Centre for Medium-Range Weather Forecasts (ECMWF) Integrated Forecasting System (IFS), typically employ reduced Gaussian grids. By decreasing the number of longitudinal points at higher latitudes, these grids mitigate the redundancy and disproportionate compute requirements at the poles found in standard equiangular grids. Modern machine learning models like FourCastNet 3 [Bonev et al., 2025] also leverage Gaussian grids for their latent representations. However, despite these improvements over latitude-longitude representations, Gaussian grids still exhibit non-uniform resolution across latitudes. As highlighted by Karlbauer et al. [2024] and Linander et al. [2025], the HEALPix grid [Gorski et al., 2005] offers a robust alternative by providing a perfectly equal-area discretisation across the entire sphere. This uniform distribution (see Figure 2.1) natively removes unphysical inductive biases, ensuring that convolution kernels or attention windows process consistent physical territories regardless of their latitudinal position. HEALPix's resolution is governed by a single parameter, n_{side}, which sets the total pixel count as $12\, n_{\text{side}}^2$; doubling n_{side} quadruples the pixel count while preserving the equal-area property at every resolution. Because each pixel subdivides into exactly four children at the next resolution, HEALPix grids form a natural hierarchy well suited to the coarsen-and-refine architectures common in weather modelling [Zhdanov et al., 2026, Linander et al., 2025].

2.1.2 Equivariance in Weather Prediction

The integration of geometric inductive biases into weather modelling has evolved from simple CNN-based architectures like U-Cast [Cachay et al., 2026] to sophisticated architectures that respect the underlying spherical topology of the Earth. The Spherical Fourier Neural Operator (SFNO) [Bonev et al., 2023]

generalises the FNO (Fourier Neural Operator) architecture to the sphere by replacing the discrete Fourier transform with the Spherical Harmonic Transform (SHT). This spectral approach achieves $SO(3)$ equivariance and maintains physically plausible dynamics for autoregressive rollouts exceeding one year. By respecting spherical topology, SFNO avoids the spectral artefacts and polar singularities common in standard FFT-based models that incorrectly assume a flat geometry. Building upon these spectral principles, FourCastNet 3 (FCN3) [Bonev et al., 2025] implements a geometric machine learning approach that combines global spectral convolutions with localised spherical group convolutions (convolution filters applied identically across rotated copies of a local neighbourhood, so the same filter recognises a pattern regardless of its orientation on the sphere). The latter is implemented via discrete-continuous (DISCO) convolutions by Ocampo et al. [2022]. This architecture rivals the accuracy of leading diffusion-based models while generating forecasts 8–60$\times$ faster and maintaining excellent probabilistic calibration for lead times up to 60 days [Bonev et al., 2025]. The Equivariant and Invariant Message Passing (EIMP) framework [Liu et al., 2025b] utilises graph neural networks to directly process spherical grid data, circumventing the distortions inherent in planar projections. It addresses the heterogeneity of meteorological data by maintaining $SO(3)$ equivariant embeddings for vector fields and $SO(3)$ invariant embeddings for scalars, which interact through a shared invariant message embedding. Experimental results demonstrate that this method achieves superior geometric consistency, particularly in polar regions.

Despite the theoretical appeal of strict equivariance, real-world climate modelling often involves symmetry-breaking factors such as heterogeneous topography. Recent research by Ashman et al. [2024a,b] on gridded and approximately equivariant Neural Processes (NPs) has addressed this by developing frameworks that can flexibly depart from exact symmetry in a data-driven manner. They demonstrate that while translation equivariance is a powerful inductive bias for stationary spatio-temporal data, strictly equivariant models are often too restrictive when modelling factors like elevation or land use that vary across the Earth's surface. Their findings show that approximately equivariant models can outperform both non-equivariant and strictly equivariant baselines in complex environmental regression tasks, as they can exploit the underlying physical symmetries of atmospheric dynamics while simultaneously learning the specific local factors that violate those symmetries. This balance is critical for large-scale weather prediction, where global conservation laws must coexist with the unique, irregular geography of the Earth.

Existing frameworks often present a forced choice between geometric rigor, expressiveness, and computational scalability. In works like Garg et al. [2026], models are evaluated based on their overall performance. This approach blurs geometric inductive biases (such as equivariance) with underlying backbone architectures (such as convolutions versus attention), preventing a systematic isolation of how geometric choices independently influence model behaviour.

Chapter 3

Background

3.1 Geometric Deep Learning

Geometric Deep Learning (GDL) provides a mathematical framework to incorporate the structure of the data and domain-specific symmetries into modelling methodologies. A core principle is *equivariance*: the property that a transformation of the input (e.g. a rotation or translation) produces a predictable transformation of the output. The convolutional neural network is the canonical example: convolution is translation-equivariant by construction, and the resulting weight sharing across spatial positions is precisely what makes CNNs so data- and parameter-efficient on images [LeCun et al., 1998]. GDL generalises this principle from translations to arbitrary symmetry groups, so that a single set of weights can be shared across rotations, reflections, and translations, including on curved domains such as the sphere. By leveraging these geometric inductive biases, GDL-based models achieve superior compute efficiency, consistently outperforming vanilla architectures at any fixed computational budget [Brehmer et al., 2024].

Group Equivariance

A group $(G, \cdot)$ is a set of transformations equipped with an operation that satisfies the four group axioms: closure, associativity, identity, and invertibility, formally representing the collection of symmetries inherent to a system. Given such a group G, let $\rho_{in}(g), \rho_{out}(g)$ be its representations; linear operators, such as rotation matrices, that describe how a group element g transforms the input and output vector spaces, respectively. A function f is defined as G-equivariant if it satisfies:

$$f(\rho_{in}(g)x) = \rho_{out}(g)f(x), \quad \forall g \in G \tag{3.1}$$

This property ensures that a transformation of the input results in a predictable transformation of the output. Traditional Group Equivariant Convolutional Networks (G-CNNs) [Cohen and Welling, 2016] implement this through weight sharing across group actions, ensuring that the model learns a single, canonical representation of a feature that generalises across all its transformed versions within the group. This higher degree of weight sharing allows the model to exploit global symmetries, increasing its expressive capacity without a corresponding increase in parameters, since a single shared weight is reused across every group element and so counts once toward the trainable parameter budget.

3.2 Rotary Position Embeddings

Rotary Position Embeddings (RoPE) [Su et al., 2024] inject positional information by rotating pairs of query and key features by an angle proportional to their position. For a query at position p_i and a key at p_j, these rotations compose so that the attention logit depends only on the relative offset $p_j - p_i$, through a block-diagonal rotation matrix ρ_Ω:

$$s_{ij} = q_i^\top \rho_\Omega(p_j - p_i)\, k_j. \tag{3.2}$$

Because the score is a function of the relative position alone, translating every position by a constant offset t leaves all attention logits unchanged: RoPE is *translation-equivariant* by construction. This is the same weight-sharing-across-position principle that makes convolutions translation-equivariant, here expressed inside the attention mechanism rather than through a fixed kernel.

By contrast, Absolute Positional Encodings (APE), as used by Vaswani et al. [2017], add a position-dependent vector directly to the token embedding. The representation is then tied to absolute coordinates, so a global shift of the input changes the encoding and the equivariance is lost.

3.3 Platonic Transformers

While the principle of symmetry has given rise to highly data-efficient and robust group equivariant networks [Cohen and Welling, 2016], scaling these symmetry-aware networks has been difficult, as they introduce significant computational overhead compared to standard architectures [Islam et al., 2025]. The Platonic Transformer [Islam et al., 2025] addresses this tension, alongside the lack of geometric inductive biases in vanilla transformers [Vaswani et al., 2017], by incorporating group equivariance without altering the underlying architecture or computational graph. This framework builds upon the translation equivariance of Rotary Position Embeddings (RoPE), where the attention score depends on relative positions via a block-diagonal rotation matrix ρ_Ω. Platonic Transformers generalise this principle by processing features from multiple reference frames $R \in \mathcal{G}$ derived from discrete symmetry groups, such as the Platonic solid groups (tetrahedral, octahedral, or icosahedral; Figure 3.1). This attention is equivalent to a dynamic group convolution, effectively carrying G-CNNs into the transformer domain.

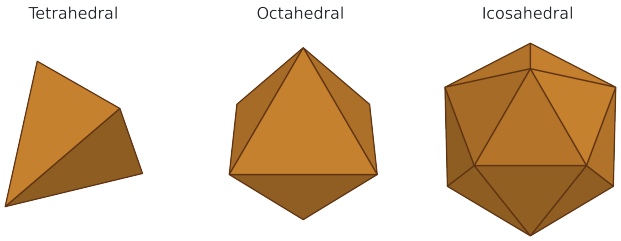


Figure 3.1: Three examples of Platonic solids.

Formally, input features f_i are lifted to the group $\mathcal{G}$ to become functions $f_i(\cdot) : \mathcal{G} \to \mathbb{R}^C$, where $f_i(R)$ represents the signal

from the perspective of frame R. The self-attention mechanism is then applied in parallel across these frames:

$$s_{ij}(R) = q_i(R)^\top \rho_\Omega((p_j - p_i)(R))k_j(R) \tag{3.3}$$

where each reference frame is effectively treated as a distinct attention head. Intuitively, the attention filters are rotated and applied to the data from different reference frames. By sharing weights across frames and constraining linear layers to be group convolutions, the model ensures that the predicted fields transform predictably under group actions. Notably, when keys are held constant, the linear attention mechanism is formally equivalent to a dynamic group convolution with content-adaptive geometric filters, enabling a linear-time $\mathcal{O}(N)$ convolutional variant.

Despite these advantages, scaling the Platonic Transformer to high-resolution data is constrained by memory bottlenecks that differ per attention mechanism. Standard softmax self-attention materialises all pairwise interactions, and its $\mathcal{O}(N^2)$ cost quickly becomes prohibitive for large graphs or dense point clouds. The linear-time convolutional variant avoids this quadratic cost by reordering the computation: instead of comparing every query against every key, it first aggregates key-value products into a single kernel and then applies this kernel to each query. This reordering, however, introduces its own memory limit. In the graph-structured formulation, the key-value outer product must be explicitly materialised for every node before aggregation, yielding an intermediate tensor of size $\mathcal{O}(N \cdot d \cdot d_{\text{head}})$ that dominates memory for wide hidden states or large node counts. (The dense, fixed-grid formulation avoids this by fusing the aggregation into a single batched matrix product, at the cost of requiring padded inputs.) Beyond memory, linear attention also carries an expressivity cost: as shown by Zhong et al. [2025], linear formulations are inherently less expressive than the exponential kernel of softmax attention. Linear attention thus exchanges representational sharpness and retrieval capacity for scalability, a trade-off that matters when modelling complex physical interactions with high fidelity.

3.4 Erwin

Erwin is a hierarchical transformer architecture designed to scale self-attention to large physical systems, handling regular grids and irregular point clouds alike [Zhdanov et al., 2025]. In this sense, Erwin can be viewed as a generalisation of the Swin Transformer's [Liu et al., 2021] hierarchical, windowed attention that is often used in weather modelling [Bodnar et al., 2025, Diaconu et al., 2026]. To overcome the quadratic complexity $\mathcal{O}(N^2)$ of standard attention, Erwin utilises ball tree partitioning to organise computation. A ball tree recursively divides the physical domain into nested, disjoint balls of a fixed size m. This structure allows the model to compute Multi-Head Self-Attention (MHSA) locally within each ball, reducing the total computational cost to $\mathcal{O}(N \cdot m)$, which scales linearly with the number of nodes N for a fixed ball size m [Zhdanov et al., 2025].

To capture long-range interactions and multi-scale coupling common in physical systems, Erwin employs a U-Net-like hierarchy. The architecture features progressive coarsening, where leaf nodes are pooled to the centres of their containing balls to form coarser representations, and refinement, which distributes information back to finer scales. Each block follows a standard pre-norm [Touvron et al., 2023] structure with attention and a SwiGLU [Shazeer, 2020] feed-forward network:

$$x \leftarrow x + \text{Attention}(\text{Norm}(x))$$
$$x \leftarrow x + \text{FFN}(\text{Norm}(x), z)$$

Furthermore, Erwin mitigates the boundary artefacts inherent in static partitioning by implementing a cross-ball interaction mechanism. Inspired by the shifted-window approach of the Swin Transformer, Erwin alternates between original and rotated ball tree configurations (rotating trees), ensuring that information can propagate between nodes that would otherwise remain in separate partitions. This combination of hierarchical scaling and localised attention enables state-of-the-art performance in domains such as cosmology and fluid dynamics. However, while Erwin scales efficiently to large irregular grids, it does not inherently respect the geometric symmetries often present in physical data.

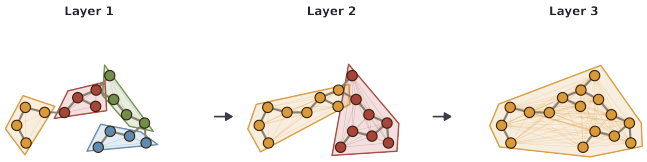
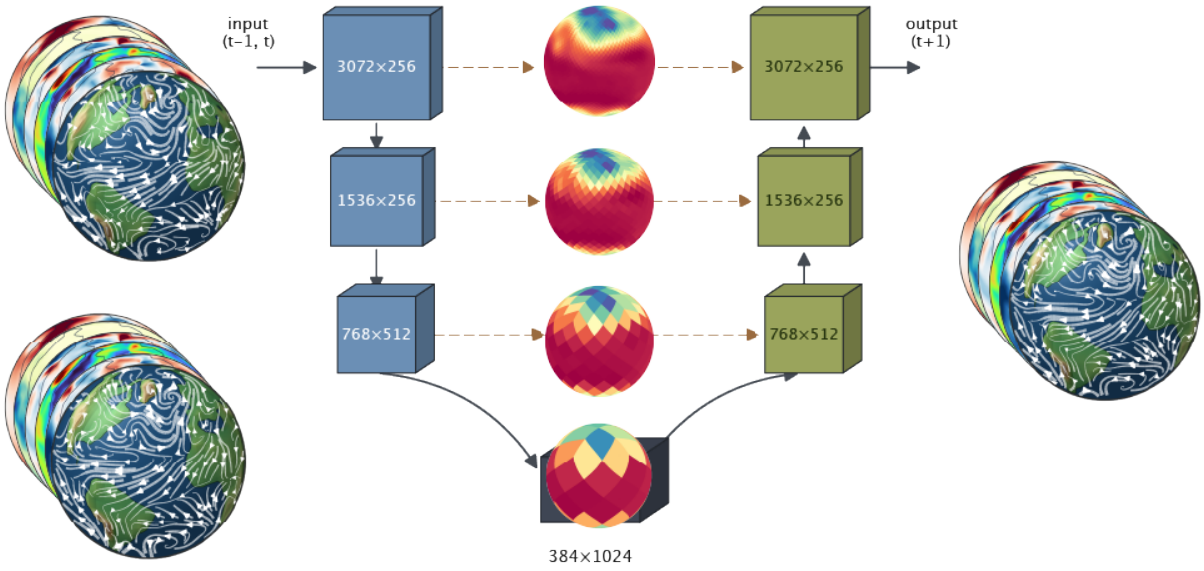


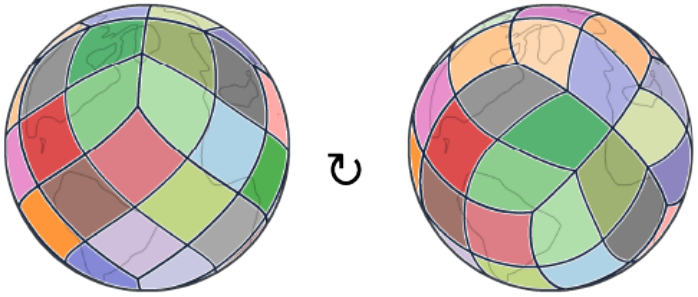
Figure 3.2: Erwin ball tree attention illustrated on a small molecule. The ball tree partitions the nodes into nested, disjoint balls; Multi-Head Self-Attention is computed locally within each ball (faint within-ball connections). Successive layers progressively coarsen the tree ($4 \to 2 \to 1$ balls), growing the receptive field from local neighbourhoods to the full molecule.

Chapter 4

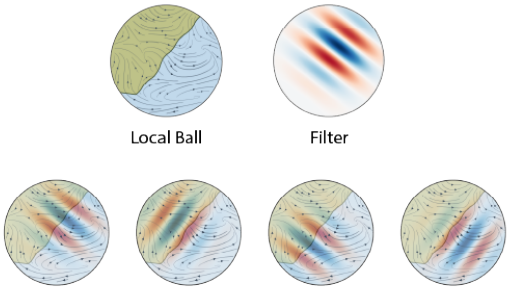
The Platonic-Erwin Architecture



(a) Multi-scale U-Net architecture on the HEALPix grid.



(b) Ball-tree partition (left) and its rotated counterpart (right).



(c) Local processing of a ball.

Figure 4.2: Illustration of Platonic-Erwin. **(a)** the multi-scale U-Net architecture processing the atmospheric state on the HEALPix grid, **(b)** the ball-tree partitions between which attention alternates, and **(c)** local processing of a ball: the Platonic mechanism applies the rotated copies of each filter (bottom row) to the local ball.

4.1 Platonic-Erwin

The Platonic-Erwin architecture (Figure 4.2) is designed to bridge the gap between rigorous physical inductive biases and the computational demands of large-scale weather modelling. While Platonic Transformers offer high expressivity and group equivariance, their quadratic complexity $\mathcal{O}(N^2)$ renders them inapplicable to the million-node grids required for high-resolution weather or protein modelling. Conversely, the Erwin framework provides the necessary linear scalability through hierarchical partitioning but lacks inherent mechanisms to respect global symmetries.

Platonic-Erwin resolves this tension by combining hierarchical linear scaling with a principled group-equivariant attention mechanism. Our design philosophy is rooted in two fundamental principles: **Noether's Theorem**, which dictates that conservation laws are the mathematical consequence of symmetries [Noether, 1918], and **Tobler's First Law of Geography**, which states that "nearby things are more related than distant things" [Tobler, 1970]. By restricting equivariant Platonic interactions to local ball partitions, we circumvent the significant computational overhead typically associated with global equivariance (Chapter 3) while maintaining physical fidelity.

Design refinements for scalable equivariance. To stabilise the training dynamics when processing high-dimensional group-equivariant features, we implement **QK-Normalisation**, following recent advancements in Vision Transformers [Wang et al., 2026, Henry et al., 2020]; queries and keys are normalised prior to the computation of attention logits. Computational efficiency

is further enhanced by replacing standard LayerNorm with **RMSNorm** [Zhang and Sennrich, 2019]. Lastly, we introduce dedicated **Register Tokens** to each ball partition [Darcet et al., 2023]. The specific methodology for this is given in Section A.2.

Equivariant pooling and unpooling. Platonic-Erwin coarsens and refines the ball-tree hierarchy with learned pooling maps, constructed so that both directions preserve equivariance. Let $\mathcal{S}_i = \{j_1, \ldots, j_s\}$ be the s children of coarse node i, with relative position $\Delta_j = \mathbf{x}_j - \bar{\mathbf{x}}_i$ from the group centroid $\bar{\mathbf{x}}_i$. Each node j carries Platonic-frame-stacked features f_j, where $f_j(R)$ is the feature vector at frame $R \in \mathcal{G}$.

For each node j, the scalar distance norm is concatenated to the features at every frame R:

$$\tilde{f}_j(R) = \left[f_j(R),\ |\Delta_j|\right]$$

Let

$$h_j = \left[\tilde{f}_j(R_1), \ldots, \tilde{f}_j(R_{|\mathcal{G}|})\right]$$

denote the features of node j, stacked across all reference frames $R \in \mathcal{G}$. Let

$$H_i = \left[h_{j_1}, \ldots, h_{j_s}\right]$$

denote the concatenated features of the children of node i.

Pooling ($\downarrow$):

$$f_i^{\text{coarse}} = \text{RMSNorm}\left(W_{\downarrow} H_i\right)$$

For unpooling, the scalar distance norm is concatenated again to the coarse features at every frame R:

$$\tilde{f}_i^{\text{coarse}}(R) = \left[f_i^{\text{coarse}}(R),\ |\Delta_i|\right]$$

Unpooling($\uparrow$):

$$f_j^{\text{fine}} = f_j^{\text{skip}} + W_{\uparrow}\, \tilde{f}_{i \to j}^{\text{coarse}}, \quad j \in \mathcal{S}_i$$

Equivariance follows because $W_{\downarrow}$ and $W_{\uparrow}$ are each a Platonic-linear layer [Islam et al., 2025], and $|\Delta_j|$ is rotation-invariant ($|R\Delta_j| = |\Delta_j|$ for any $R \in \mathcal{G}$).[1] Together with the equivariant ball attention, this makes the full coarsen-refine hierarchy equivariant by construction.

Set-axis ball tree construction. The standard ball tree algorithm splits nodes along axis-aligned directions of the (x, y, z) frame, so the resulting hierarchy is not invariant to global rotations of the input and therefore breaks the equivariance required by Platonic interactions. We resolve this by applying a *global PCA-based canonicalisation* prior to tree construction (Figure 4.3; details in Section A.1). During training we deliberately use the *standard* ball tree on un-canonicalised positions, so that data augmentations (e.g. random SO(3) rotations in the Protein MD experiments) induce a different hierarchical partition at every step; this acts as a strong data-dependent regulariser. At inference time we instead use the PCA-aligned variant that first canonicalises each batch and then builds the tree on the canonical frame, which, under the assumptions in Section A.1, yields a tree structure that is invariant to rotations. This decoupling keeps the practical benefits of stochastic tree sampling at train time, while still guaranteeing a strictly group equivariant prediction whenever the model is deployed. This is analogous to the random-rotation augmentation used when training G-CNNs.

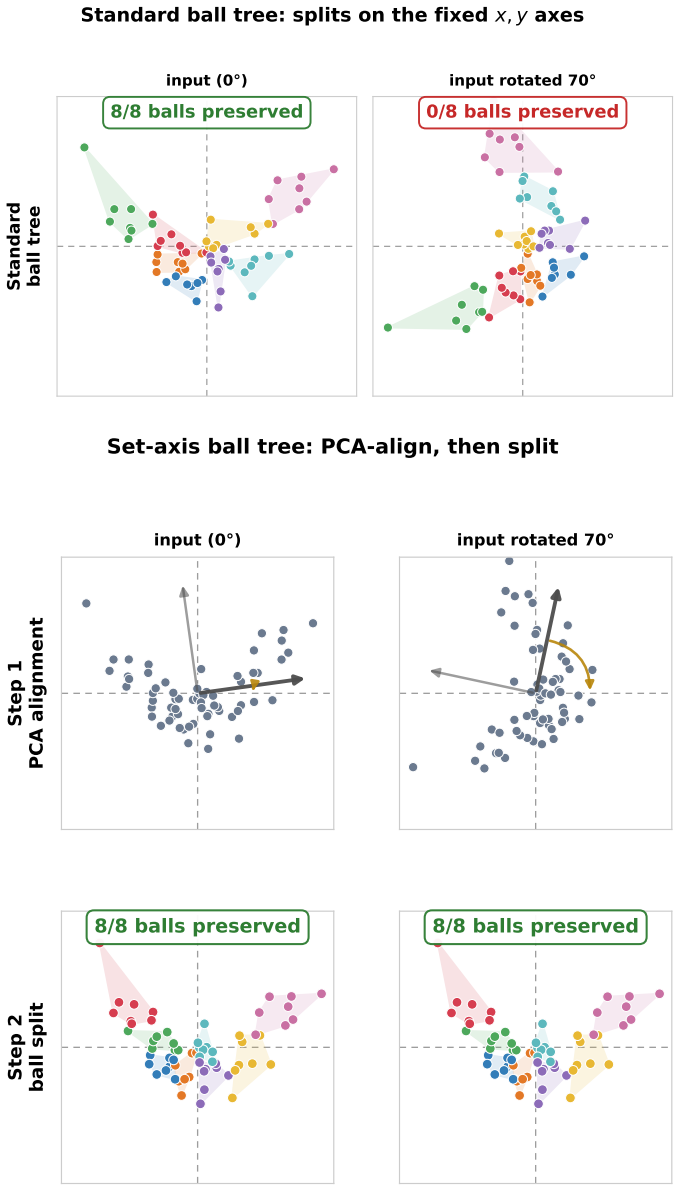


Figure 4.3: Set-axis versus standard ball tree construction under a global input rotation, on a 2D point cloud.

4.2 Platonic-Erwin - Weather

HEALPix integration via equivariant cross-attention. The native ERA5 data is defined on an equiangular latitude-longitude grid, whose cells subtend a fixed angular step in latitude and longitude. Because the meridians converge toward the poles (Figure 2.1), the physical area covered by each cell varies strongly with latitude: grid points are densely oversampled near the poles and sparsely sampled near the equator. This spatial inhomogeneity means that a fixed local operator (a ball-tree neighbourhood,

[1] Scalar distance concatenation is the default. The model also supports an equivariant alternative that lifts the directional offset $\Delta_j = \mathbf{x}_j - \bar{\mathbf{x}}$ into the frames — feeding the $|\mathcal{G}|$ rotated copies $\{R\Delta_j\}_{R \in \mathcal{G}}$ to the linear map instead of the single invariant $|\Delta_j|$. This is used in the final weather runs.

an attention window, or a convolutional filter) spans very different physical scales depending on where it is applied, so it effectively sees a geometry that stretches with latitude. To eliminate spatial inhomogeneity, the architecture can utilise a hierarchical HEALPix grid as the internal latent representation. HEALPix is equal-area by construction [Gorski et al., 2005]. Its nested, hierarchical pixel ordering aligns naturally with Platonic-Erwin's coarsening and refinement, and because the grid is static the ball-tree partitions and pooling hierarchy can be precomputed once rather than rebuilt at every forward pass. This contiguous layout additionally suits block-based computation: loading a spatially local block of pixels is a single coalesced memory read rather than the scattered accesses required on a latitude-longitude grid [Zhdanov et al., 2026]. Transitions between the input equiangular grid and the HEALPix mesh are done with group-equivariant cross-attention, which performs k-nearest neighbour interpolation similar to Wessels et al. [2024]: each destination node attends over its k nearest source nodes, with relative geometry injected through (Platonic) RoPE, yielding a learned, geometry-aware interpolation. The full formulation is given in Section B.1.

2D modelling with periodic boundaries. To adapt to the 2D latitude-longitude grid, the model enforces longitudinal periodicity. While the model operates on native 2D coordinates, the ball-tree construction algorithm and k-NN queries utilise a temporary 3D Cartesian projection to ensure local neighbourhoods remain physically contiguous across the dateline. To prevent coordinate discontinuities, ball centroids are calculated with a circular mean, and relative positions use the shortest signed longitude difference. This relative distance is used with RoPE in the ball-attention blocks; because it is always bounded below $180°$, it never approaches the periodic boundary and needs no further correction. Cross-attention, by contrast, operates on absolute node positions that can lie anywhere on the globe. For experiments on wrap-around inductive RoPE/APE biases, longitudinal frequencies are additionally quantised to exact harmonics of the globe's circumference to ensure the embeddings are perfectly $360°$-periodic (detailed in Appendix D). Together, this allows the model to maintain a 2D grid while respecting the global topology of the sphere.

3D Cartesian modelling with vector transformations. For 3D modelling experiments we utilise a full 3D Cartesian pipeline. In this mode, wind components are lifted from their local 2D tangent planes into global 3D Cartesian vectors. This is further detailed in Appendix C. The transformation also enables the application of 3D Platonic groups.

Symmetry group selection for weight sharing. The Platonic-Erwin architecture utilises the Platonic groups to implement group-equivariant weight sharing for weather modelling. For the 3D setting, the natural first candidate for weight sharing is a longitudinal cyclic group $\mathcal{C}_{n,z}$ (detailed in Section B.4), since rotation about Earth's axis is an exact global symmetry. That symmetry, however, is only meaningful for a global operator. Platonic-Erwin's attention is local, and a rotation about the global z-axis rotates a local filter out of the tangent plane in which the physical field lives (Figure 4.4), producing filters that are unphysical for modelling the atmosphere. We therefore utilise the tetrahedron group. While this group enables better weight sharing with *in-plane* rotations, it still introduces the undesirable rotations of the filters.

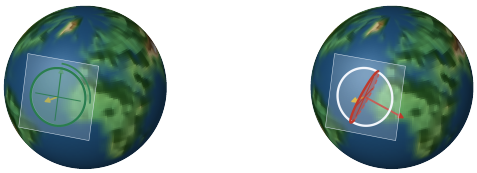


Figure 4.4: The tilted-filter problem. **(Left)** In-plane rotation. The filter stays in the tangent plane. **(Right)** In-axis rotation. The filter rotates out of the tangent plane.

For 2D, we use the 2D Cyclic-4 group. As we operate on the 2D manifold, the local filters can be seen as operating on a local tangent-plane approximation. This group therefore models in-plane rotations on the tangent plane, capturing the relevant meteorological symmetries without the unphysical geometric distortions inherent to the 3D group.

Time embeddings. To provide the model with temporal information, we utilise a sinusoidal time embedding layer initialised with random Fourier features, similar to [Bodnar et al., 2025]. A more detailed description of this can be found in Section B.2.

Chapter 5

Architectural Validation on Protein Molecular Dynamics

Setup

To validate our architecture on large-scale inputs, we evaluate on a molecular-dynamics benchmark that models the equilibrium-time evolution of protein structures, where atom interactions depend on both local and global-range geometry. Following Han et al. [2022], we use the AdK equilibrium MD trajectory of Seyler and Beckstein [2017], processed with MDAnalysis, yielding 4,186 protein structures with trajectories. Since we specifically care about our model performing on samples with many data points, we only evaluate on the all-atom (3341 atoms) variant. Due to computational constraints, we run our best configuration ($\sim$14.4M) only twice (with different seeds). Because the all-atom graphs are large, we use a lazy, per-graph on-disk cache that keeps per-worker memory independent of dataset size; the implementation is detailed in Section E.2. The reported configuration is the one selected by our integrative ablation study (Section E.4.4); hyperparameters are listed in Table E.1.

Results

We compare Platonic-Erwin against the state-of-the-art (SOTA) methods for protein molecular dynamics. For completeness, we also include Erwin. As can be seen in Table 5.1, Platonic-Erwin achieves SOTA performance compared to all baselines, proving its capability. Notably, it significantly outperforms the non-equivariant Erwin model, validating that equivariance provides a useful inductive bias for this task. It should be noted that the results of the Platonic Transformer were not reproducible using their code and hyperparameters.

Table 5.1: Final ProteinMD results (*results from Islam et al. [2025]). Best scores in **bold**, second best underlined.

Method	Force MSE ↓
EGNN*	2.72 ± 0.003
G-Transformer*	3.67 ± 0.640
G-Hyena*	2.49 ± 0.037
Platonic Trans.*	$\underline{2.36 \pm 0.011}$
P. T. Reproduction	2.49088
Erwin	2.84331
Platonic-Erwin	$\mathbf{2.339 \pm 0.019}$

In Table 5.2 it can be seen that the tree-building algorithm `Setaxis` (PCA canonicalisation) yielded inferior results compared to the `normal` ball-tree configuration. This is in line with the hypothesis that the standard axis-aligned splitting rule provides a beneficial form of implicit data augmentation during training.

Table 5.2: Effect of training with different tree partitioning. Platonic-Erwin-Init refers to a baseline version *without* SOTA-specific optimisations and isolates the impact of `Setaxis` vs. `Normal` construction. Best scores in **bold**.

	Force MSE ↓	
Model	**Normal**	**Setaxis**
Erwin	**2.843**	2.865
Platonic-Erwin-Init	**2.518**	2.765

As demonstrated in Table 5.3, the difference between `Normal` and `Setaxis` testing is negligible, suggesting the model has successfully learned a robust equivariant representation that is indifferent to the specific tree partitioning scheme. This means we have found a way to augment data by relaxing equivariance during training to create a more efficient and generalisable learning strategy, which can be changed back to strict equivariance during inference. The full ablation study is detailed in Section E.4.

Table 5.3: Platonic-Erwin seed stability and inference-time tree sensitivity. Both rows are the same best configuration, trained with `Normal` tree-building; the two MSE columns reflect inference with `Normal` vs. `Setaxis` tree partitioning.

	Force MSE ↓	
Seed	**Normal**	**Setaxis**
3	2.32498	2.32497
13	2.35255	2.35256

Summary

Taken together, these results validate Platonic-Erwin as the backbone for the remainder of this thesis. Two findings carry forward. First, the architecture is both *scalable* and *effective*: it handles the large all-atom protein graphs (3341 atoms) that motivated its design, matches or exceeds the state of the art (Table 5.1), and beats the non-equivariant Erwin baseline. This confirms that group equivariance is a useful inductive bias, not merely an added cost. Second, the tree-partitioning ablation shows that equivariance can be *relaxed during training* where axis-aligned ball-tree splitting acts as implicit data augmentation, and *restored at inference*. The model proved indifferent to the partitioning scheme it is tested on.

Having established that the backbone scales and that its geometric inductive biases pay off on a controlled molecular-dynamics benchmark, we now carry this validated architecture and its best configuration into the setting this thesis is really about: geometric weather forecasting, where the data lives on the sphere and the relevant symmetries are those of the rotating Earth.

Chapter 6

Ablations: Weather Forecasting & Geometry

6.1 Experimental Setup

We structured our experiments to systematically examine the impact of geometric inductive biases on weather forecasting. Each experiment is designed to isolate specific geometric choices to determine their contribution to predictive skill. The training setup is detailed in Appendix G.

6.1.1 Dataset

We use the 5.6° resolution variant of the WeatherBench 2 ERA5 dataset [Rasp et al., 2024]; comprehensive specifications are given in Appendix F. This coarse resolution keeps the full ablation study within our compute budget and, more importantly, enables many controlled, like-for-like comparisons under identical conditions rather than a handful of expensive high-resolution runs. For the same reason, the ablation models are trained on deterministic single-step prediction only: rollout fine-tuning and a probabilistic (CRPS) objective each multiply the per-run training cost several-fold. The architecture predicts the next normalised atmospheric state $\hat{\mathbf{y}}_{t+1}$ directly, conditioned on the two preceding states $\{\mathbf{y}_t, \mathbf{y}_{t-1}\}$. We expect the geometric findings to carry over to higher-resolution, rollout-fine-tuned, and CRPS-trained settings, though confirming this is left to future work. The years 1979–2018 are used for training, 2019 for validation, and 2020 for held-out testing.

6.1.2 Default Configuration

We reuse the base architecture configuration validated on ProteinMD. Ablations showed no gains from register tokens, or cross-attention pooling; they introduce computational overhead and are therefore not used (Section G.5). Unless otherwise stated, all geometry experiments share the same baseline, using a deterministic MSE loss:

- **HEALPix.** Cross-attention re-projection onto a HEALPix grid (n_{side}=16, k=16).
- **Vector loss.** The vector loss is calculated with an ANGLE+MAGNITUDE objective (instead of MSE on the individual components (u, v)); scalars use GraphCast-style per-variable and per-pressure-level weighted MSE [Lam et al., 2023]. We omit this variable weighting for vector fields when using the ANGLE+MAGNITUDE objective, as the GraphCast weights are not calibrated for this loss objective.
- **2D vs. 3D.** 3D-Cartesian modelling, with tangent-plane projection of the vector loss.[1]
- **Freq. init.** Spiral RoPE initialisation with learnable frequencies, and random APE initialisation with learnable frequencies.

This set encodes our *a priori* best setting. We use the standard ball tree algorithm here instead of the set-axis canonicalisation, as the ERA5 data itself is already canonicalised. The default configuration is a ∼20M-parameter model. Because its embedding width is smaller than the per-node input dimension, the encoder mildly compresses the data, which Zhdanov et al. [2026] showed to be suboptimal. We consider this acceptable for the controlled geometry ablations, where relative comparisons matter more than absolute skill. Any deviations from this default setup are listed per experiment in Section G.1.

6.1.3 Evaluation Metrics

Each method's performance is evaluated following the protocols established in the WeatherBench 2 benchmark [Rasp et al., 2024]. We validate on their headline variables: 500hPa geopotential (Z500), 850hPa temperature (T850), 700hPa specific humidity (Q700), 850hPa wind vector (WV850), 2m temperature (T2M), 10m wind speed (WS10), and mean sea level pressure (MSLP).[2] Wind speed is defined as $\text{WS} = \sqrt{u^2 + v^2}$, the scalar magnitude of the wind, which scores only how fast the air moves and is blind to direction. The wind vector scores the full horizontal wind arrow (u, v) jointly, $\text{MSE}_{\text{WV}} = \langle(\hat{u}-u)^2+(\hat{v}-v)^2\rangle = \text{MSE}(u)+\text{MSE}(v)$, so a forecast with the right speed but the wrong direction is penalised by the wind-vector MSE. Variable definitions and the evaluation protocol follow WeatherBench 2 [Rasp et al., 2024]; the specific metrics we report are defined below. We perform and evaluate 10-day autoregressive rollouts on the test year 2020. However, because our 5.6° geometry-experiment models are trained exclusively on deterministic single-step prediction, autoregressive errors compound rapidly over the rollout. We therefore report the full 10-day rollout for every experiment, but weight the two horizons differently: rankings and conclusions are drawn primarily from lead times up to 72 hours, similar to Diaconu et al. [2026], where compounding is limited and the single-step training regime is most representative. Behaviour beyond that horizon (crossovers, instabilities, drift) is read as a qualitative diagnostic of rollout stability rather than as a skill ranking. For brevity, the per-variable deterministic and spectral figures in each experiment below show a representative subset of these headline variables rather than the full set of seven; unless stated otherwise, the omitted variables and pressure levels show the same patterns.

[1] We vary only the loss applied to the tangent-plane (u, v) wind components. In 3D-Cartesian mode the model predicts a 3D vector, so before computing the objective we project both prediction and target onto the local tangent basis $(\hat{e}_\lambda, \hat{e}_\varphi)$ and discard the radial component. ERA5 winds are tangent to the sphere $(\mathbf{v} \cdot \mathbf{p} = 0)$, so the radial direction carries no observed signal: scoring it would reward the trivial task of zeroing an unobserved component, and a raw 3D MSE would average over three components instead of two, lowering the loss by a skill-independent factor and making it incomparable with the 2D setting.

[2] We exclude 24h precipitation (TP24hr) from this evaluation, as our model does not predict this variable.

Deterministic Skill

Following the WeatherBench 2 protocol [Rasp et al., 2024], all metrics are computed per grid point and then latitude-weighted to account for the varying grid-cell area of the equiangular grid, and averaged per variable and pressure level. We report the latitude-weighted Mean Squared Error (MSE), the anomaly correlation coefficient (ACC), and the Bias. ACC isolates spatial pattern skill (higher is better), scoring structural positioning independently of amplitude (full definition in Section G.3). The Bias, $\frac{1}{N}\sum_i(\hat{y}_i - y_i)$, isolates systematic, non-zero-mean drift under autoregressive rollout. We additionally report an aggregated skill score (higher is better), averaged over the headline variables introduced above, following the methodology of Couairon et al. [2024] and Diaconu et al. [2026] (full definition in Section G.3). Unlike those works, we don't score against a fixed external forecast: each ablation nominates one of its own arms as that experiment's reference, and every other arm's skill is reported relative to it.

Spectral Fidelity

A model can post good deterministic metrics, while systematically blurring away fine-scale variability (the "double-penalty" problem [Subich et al., 2025]). Such scores alone therefore reveal little about whether a forecast retains the right amount of physical structure. To assess this, we compute spatial power spectra and kinetic-energy (KE) spectra, which decompose each field into spatial wavenumbers k, displayed against the corresponding equatorial wavelength $\lambda \propto 1/k$, and show how much structure (for KE, how much kinetic energy) the forecast carries at each spatial scale (from large planetary waves at long wavelength / low k down to fine local features at short wavelength / high k). A faithful forecast matches the reference spectrum at every scale. Because spectra span several orders of magnitude, small deviations between models are hard to judge from the raw curves. Therefore, we also show the (KE) spectral retention ratio $R(k) = \frac{S_{\text{pred}}(k)}{S_{\text{target}}(k)}$, the fraction of the true atmosphere's power the model reproduces at each scale. $R(k) = 1$ is perfect; $R(k) < 1$ means the forecast is too smooth at that scale (lost structure), while $R(k) > 1$ means it has injected spurious small-scale noise.

6.2 Experiment 1: Grid Topology

We first investigate the influence of grid-induced biases by contrasting the performance of the architecture on a biased and unbiased grid. We compare two otherwise-identical configurations that differ only in the internal grid they operate on: LAT-LON operates on the 64×32 equirectangular grid (2048 pixels), while HEALPIX inserts a cross-attention step that re-projects the input onto an n_{side}=16 HEALPix mesh (3072 equal-area pixels). Karlbauer et al. [2024] do a similar experiment, but they do internal grid compression, which is lossy [Zhdanov et al., 2026]. We do this experiment with *grid expansion*.

Results

Deterministic skill. The aggregated deterministic skill picture is shown in Figure 6.1. Figure 6.2 reports geopotential as a representative headline variable. On lead times up to 72 h, modelling with the HEALPIX grid results in higher aggregated skill ($\sim$12%) and better MSE and ACC on all headline variables. While it continues to lead on ACC for the full 10-day window, a crossover happens at $\sim$6 days in MSE and aggregated skill, where modelling with the LAT-LON grid has a small advantage ($\sim$5%).

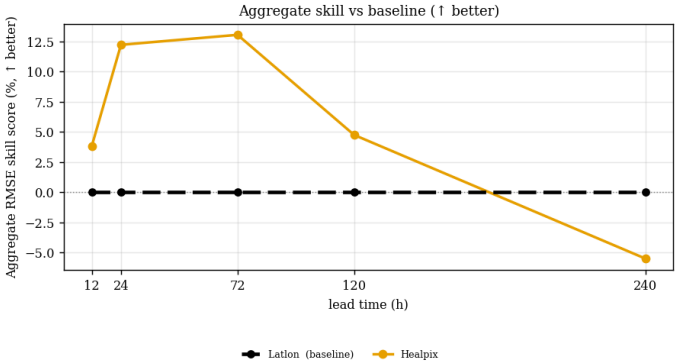


Figure 6.1: Aggregated deterministic skill of the grid ablation over the 10-day rollout, relative to the reference model (using the LAT-LON grid).

Spectral Fidelity. The spectral picture is mixed. The kinetic-energy retention ratio figures (Figure 6.2, middle column) show that at 12 h both grids track ERA5 ($R \approx 1$) across the resolved band and differ only at the smallest scales, where the LAT-LON grid retains modestly more energy while HEALPIX blurs. At regional to global scales, however, HEALPIX maintains the higher fidelity. By 72 h the two separate: HEALPIX stays faithful ($R \approx 1$) across the larger scales that carry the bulk of the energy, whereas LAT-LON is visibly damped ($R \approx 0.7$–0.85). The curves cross only at the finest scales, but there LAT-LON overshoots the reference ($R \approx 1.3$): spurious small-scale energy rather than resolved structure. Figure 6.2 (right column) shows the zonal spectra at 72 h for geopotential and temperature (mean sea-level pressure and specific humidity given in Figure H.1). HEALPIX preserves higher spectral fidelity on geopotential and mean sea-level pressure, while LAT-LON yields sharper spectra on specific humidity and temperature.

Insights

Within the 72 hour scope, modelling on the equal-area HEALPIX grid improves deterministic skill across all variables, while no grid has a clear spectral advantage. The contrast between ACC and MSE at longer lead is interesting in itself: HEALPIX leads on ACC at every lead, yet its MSE advantage erodes over the rollout and turns into a small deficit against LAT-LON at the longest lead times. We read this as a difference in blurring rate rather than a reversal of which grid is better: HEALPix continues to reproduce the spatial structure (pattern) of the anomalies more faithfully, which is exactly what the (amplitude-blind) ACC rewards, while

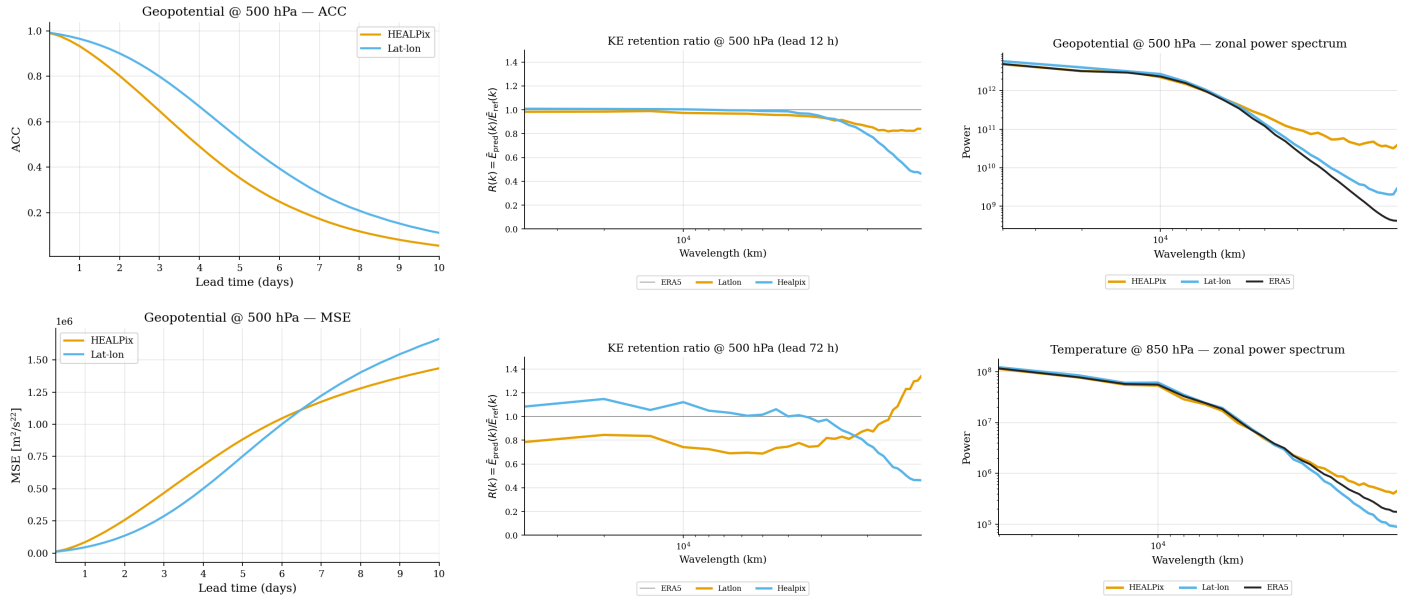


Figure 6.2: Grid ablation summary at 500 hPa (WB2 headline level unless noted). **Left column:** deterministic skill of geopotential (ACC top, MSE bottom). **Middle column:** kinetic-energy retention ratio (12 h lead top, 72 h lead bottom), $R{=}1$ matches ERA5. **Right column:** zonal power spectra at 72 h lead (geopotential top, temperature at 850 hPa bottom), ERA5 (black) is the reference. HEALPix (blue) and LAT-LON (orange).

lat-lon's forecast collapses toward the climatological mean once genuine skill has decayed. Squared error rewards rather than punishes that collapse once specific placement is more often wrong than right (the same conditional-mean-seeking behaviour behind the double-penalty effect, Chapter 1).

We attribute HEALPix's advantage to two mechanisms: mainly (i) the uniform-area discretisation, which eliminates the polar oversampling of the equirectangular grid; (ii) an implicit register-token effect from the larger HEALPix latent grid, which we hypothesise supplies additional capacity for global aggregation. HEALPix's fixed neighbour structure is also more compute-efficient for caching, pooling, and memory layout across rollout steps (Section 4.2).

6.3 Experiment 2: Vector Modeling and Loss Formulation

To address the systematic underestimation of wind magnitude inherent in component-wise MSE loss, we evaluate several vector-loss objectives. By comparing against the standard Mean Squared Error (MSE), we isolate the benefit of modelling wind as a geometric entity rather than independent scalars. To guarantee a fair comparison, GraphCast vector weighting is disabled across all evaluated vector losses (including plain MSE). Dunstan et al. [2025] run a similar experiment with their FastNet model but do not motivate their specific choices. They decompose the wind into a unit-vector direction and a scalar magnitude. One of FastNet's contributions is the particular weighting (the FASTNET and FASTNET×5 variants in Table 6.1): they adopt a $5\times$ magnitude factor without an ablation justifying it. We provide a more rigorous study on the vector loss and question if their loss formulation is optimal for our purposes or if it can be simplified.

MSE	$\|\hat{\mathbf{v}} - \mathbf{v}\|_2^2$
ANGLE + MAG	$(1 - \cos\theta) + (\hat{s} - s)^2$
MSE + MAG	$\|\hat{\mathbf{v}} - \mathbf{v}\|_2^2 + (\hat{s} - s)^2$
MSE + MAG + ANGLE	$\|\hat{\mathbf{v}} - \mathbf{v}\|_2^2 + (\hat{s} - s)^2 + (1 - \cos\theta)$
FASTNET	$\|\hat{\mathbf{v}}/\hat{s} - \mathbf{v}/s\|_2^2 + (\hat{s} - s)^2$
FASTNET×5	$\|\hat{\mathbf{v}}/\hat{s} - \mathbf{v}/s\|_2^2 + 5\,(\hat{s} - s)^2$

Table 6.1: Loss objectives evaluated in the vector-loss ablation, applied per wind-vector node and variable. $\mathbf{v} = (u, v)$ is the target, $\hat{\mathbf{v}}$ the prediction, $s = \|\mathbf{v}\|$ the target speed, $\hat{s} = \|\hat{\mathbf{v}}\|$ the predicted speed, and θ the angle between prediction and target.

The six objectives, each computed per node and variable, are shown in Table 6.1. FASTNET normalises the vector components before calculating the loss, making the directional term scale-invariant; FASTNET×5 further up-weights the magnitude term. ANGLE + MAG provides a simpler counterpoint where direction is measured by angle alone; note that the FASTNET directional term $\|\hat{\mathbf{v}}/\hat{s} - \mathbf{v}/s\|^2$ reduces to $2(1 - \cos\theta)$, so both objectives share the same directional geometry (and magnitude component); as implemented (averaging over the two tangent components) the angular penalties coincide exactly (derivation in Section H.2). The augmented MSE variants (MSE + MAG, MSE + MAG + ANGLE) double-penalise speed (once implicitly through the component MSE, which underestimates magnitude (Section 2.1.1), and once explicitly through the magnitude term) and we include them to test whether naively adding geometric terms to MSE helps. We report per-variable skill for wind fields only, since the vector-loss objective acts directly on them; the aggregated skill score here is computed across wind metrics only (WS500, WS700, WS850, WV500, WV700, WV850, WS10 and WV10).

Results

Deterministic skill. The wind-aggregate deterministic skill is shown in Figure 6.3. Figure 6.4 reports ACC and MSE for the 10 m surface wind (columns 1-2). The runs separate mainly at longer lead. The ranking of objectives is consistent across deterministic metrics: component-wise MSE performs best, followed in most cases by MSE + Mag + Angle and MSE + Mag. The objectives that omit the component-wise term (the FastNet-style decoupled loss (with and without the $5\times$ directional weighting) and Angle + Mag) rank last, with FastNet$\times 5$ performing the worst. Adding the angle term (MSE + Mag + Angle vs. MSE + Mag) gives a marginally better ($\sim 2\%$) aggregated skill at every lead, as can be seen in Figure 6.3.

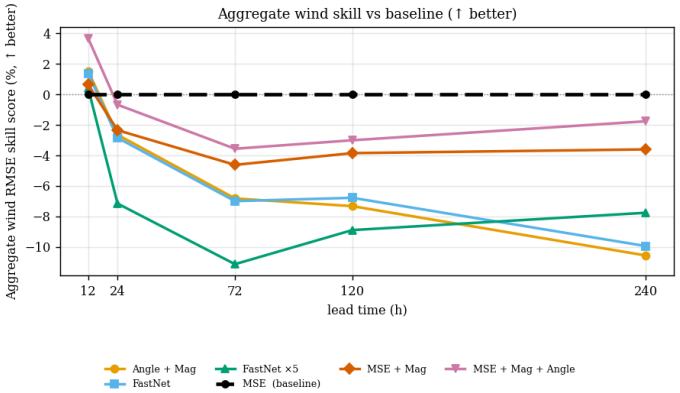


Figure 6.3: Wind-aggregate RMSE skill of the vector-loss ablation over the 10-day rollout, relative to the reference model trained with MSE loss.

The signed bias of 10 m wind speed and u_{10} is given in Figure 6.5. A consistent sign pattern emerges across the wind variables: component-wise MSE produces a *negative* wind speed bias early, while every objective with an explicit magnitude term causes the bias to be *positive*, the over-correction varying in size across variables. The FastNet, FastNet$\times 5$ and angle+mag variants tend to have a significant positive bias, while MSE + Mag and MSE + Mag + Angle are the least biased.

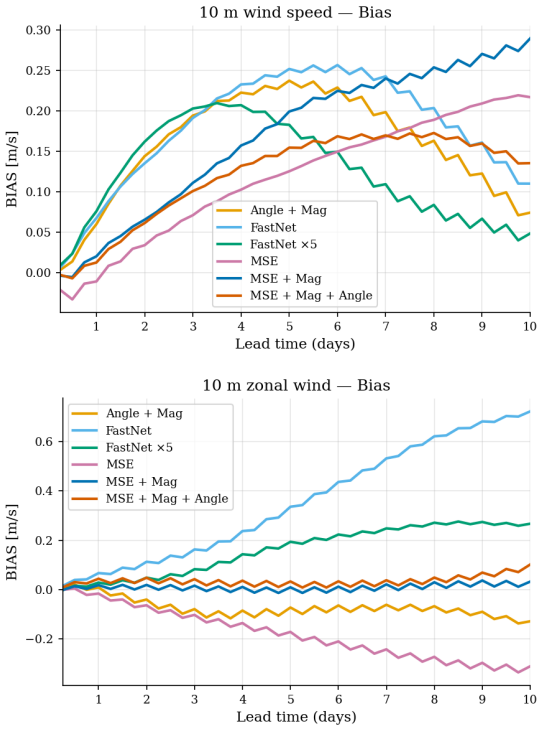


Figure 6.5: Signed bias for the 10 m surface wind on the vector-loss ablation. **Top:** wind speed. **Bottom:** zonal component u_{10}.

Spectral Fidelity. Figure 6.4 (column 3) reports the KE retention ratio at 12 h and 72 h lead at 500 hPa (WB2 headline level). At 12 h the objectives differ little: FastNet$\times 5$ marginally retains the most kinetic energy, MSE the least, the rest in between, and the spectra, though hard to discriminate at this short lead, are consistent with the retention picture. By 72 h, FastNet$\times 5$ generates a marked excess of energy at small scales while MSE under-produces there, the remaining objectives in between. The objectives MSE + Mag, Angle + Mag, FastNet and MSE + Mag + Angle are sitting in the middle band. The zonal spectra were hard to distinguish (Section H.2, Figure H.2).

Insights

No single wind-loss objective wins on every metric. There is a trade-off between deterministic skill and spectral fidelity. Component-wise MSE is best on ACC and MSE (its own training target) but blurs the most (lowest small-scale KE retention) and is the only objective with a systematically negative wind-speed bias, the Jensen's-inequality-driven underestimation described in Section 2.1.1. Every objective that has an explicit magnitude term flips this bias positive: the FastNet-style and Angle + Mag variants over-correct substantially, while MSE + Mag and MSE + Mag + Angle remain closest to unbiased.

The best compromise is to add a magnitude term on top of MSE rather than in place of the component-wise term. MSE + Mag and MSE + Mag + Angle track MSE's skill, retain small-scale energy stably, and stay near-unbiased, because the magnitude term counters the underestimation while the retained component term anchors the loss to the well-behaved MSE optimum. The FastNet-style and Angle + Mag objectives drop that component term and lose skill as a result. Adding the an-

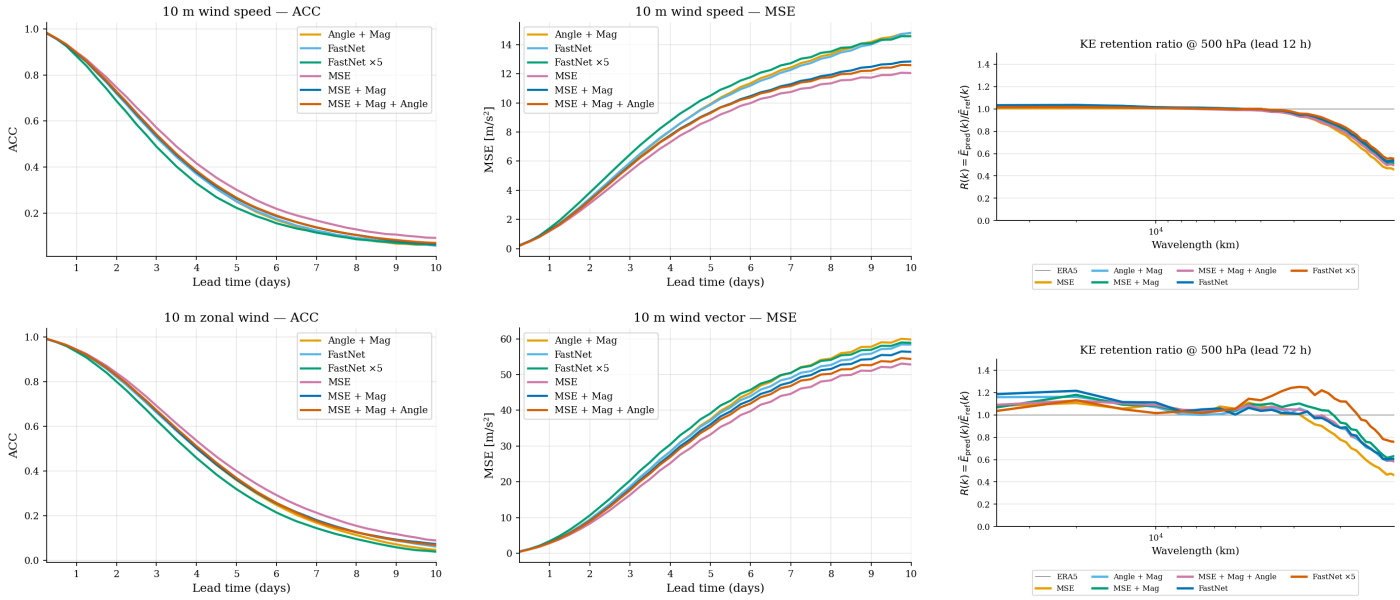


Figure 6.4: Deterministic skill and spectral fidelity for the 10 m surface wind on the vector-loss ablation. **Column 1:** ACC (wind speed top, u_{10} bottom). **Column 2:** MSE (wind speed top, wind vector bottom). **Column 3:** kinetic-energy retention ratio at 500 hPa (WB2 headline level, 12 h lead top, 72 h lead bottom), $R(k) = 1$ matches ERA5.

gle term on top (MSE + Mag + Angle) buys a further marginal gain in deterministic skill at similar spectral fidelity, making it the best objective overall. More broadly, treating wind as a geometric vector quantity, with explicit magnitude and direction, does help, but only as an augmentation of the component-wise representation, not a replacement for it.

6.4 Experiment 3: 2D vs. 3D Modeling

Earth and its weather are three-dimensional, yet almost every machine-learned forecast model treats them in two. To determine whether the model should operate in 2D or 3D for global weather forecasting, we compare modelling on a 2D latitude-longitude manifold versus a full 3D Cartesian representation. The distinction in dimensionality primarily affects coordinate representation and vector dimensionality. Note that 2D RoPE already approximates translation equivariance over the sphere: in lat-lon coordinates, the sphere is parameterised as a rectangle, periodic in longitude, a shift here corresponds exactly to a rotation of the globe about the Earth's axis. In latitude, translation equivariance is only approximate: grid cells narrow towards the poles, so shifting a pattern in latitude changes its physical shape. 3D RoPE, by contrast, is equivariant to translations in 3D, none of which map the sphere onto itself. This asymmetry partially confounds a direct 2D-3D comparison under RoPE, which is why we also include the APE configurations.

Results

Deterministic skill. Figure 6.6 compares the 2D latitude-longitude configurations against the 3D Cartesian ones for all four positional-encoding variants with aggregated skill. Figure 6.7 reports geopotential as a representative headline variable across the four positional-encoding variants. The ordering emerges as 2D APE > 3D APE and 2D RoPE ≥ 3D RoPE on all metrics, with the 2D and 3D configurations differing by less than 1% on day one and the gap widening at longer lead. 2D APE is the most skilful (highest ACC, lowest MSE) and 3D RoPE the least.

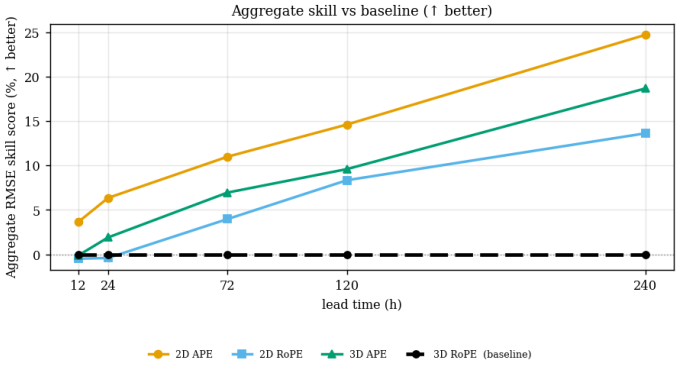


Figure 6.6: Aggregated deterministic skill of the dimensionality ablation over the 10-day rollout, relative to the reference model which utilises 3D RoPE.

Spectral Fidelity. The spectral picture is mixed and hard to distinguish (Figure 6.7, middle and right columns, show representative levels and variables). No dimensionality has a clear advantage.

Insights

A model's representation of space, coordinates and vectors alike, should match the manifold it operates on, not the space the data is embedded in. Within each family, 2D modelling beats the 3D

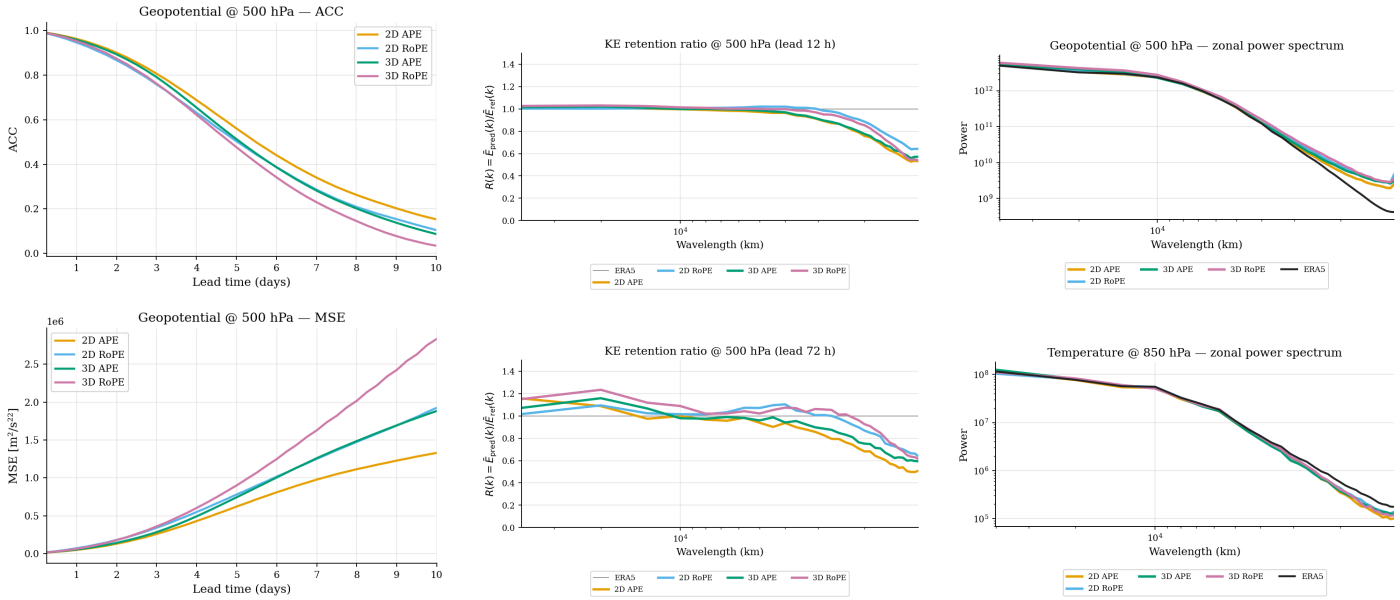


Figure 6.7: Dimensionality ablation summary at 500 hPa (WB2 headline level unless noted), four positional-encoding variants (2D/3D × APE/RoPE). **Left column:** deterministic skill of geopotential (ACC top, MSE bottom). **Middle column:** kinetic-energy retention ratio (12 h lead top, 72 h lead bottom), $R{=}1$ matches ERA5. **Right column:** zonal power spectra at 72 h lead (geopotential top, temperature at 850 hPa bottom), ERA5 (black) is the reference.

representation on deterministic skill (2D APE $>$ 3D APE and 2D RoPE $\geq$ 3D RoPE, the gap opening beyond $\sim$day 3). The dimensionality does not seem to have an impact on spectral fidelity. Because the ordering holds for the non-equivariant APE models, the effect is attributable to dimensionality rather than to equivariance alone.

We attribute this to three complementary mechanisms: (i) a simplicity bias in favour of the lower-dimensional 2D parameterisation, which aligns with the spatial locality of the tangent-plane view; (ii) the 2D representation implicitly rotates the kernel around the sphere whereas the 3D one does not; and (iii) an unsupervised third dimension introduced by the 3D vector lifting, which carries no independent observational signal (ERA5 wind is tangent to the sphere, so the radial component receives no direct supervision from the loss) and therefore could act as a noisy variable the model must learn to suppress, rather than a genuine source of expressiveness.

6.5 Experiment 4: Positional Encoding Frequency Schemes

We evaluate the role of geometric inductive bias in APE and RoPE for the Earth's geometry by comparing a standard spiral initialisation against a 2D log-space (lat-lon) initialisation, as detailed in Appendix D.[3] We also examine the effect of encoding longitudinal wrap-around for both schemes; only the non-periodic initialisation supports learnable frequencies.

[3]The initial standard deviation of APE and RoPE frequencies are fixed and not tuned here due to computational constraints

Results

Deterministic skill. The aggregated deterministic skill picture is shown in Figure 6.8. Figure 6.9 (left and middle columns) reports deterministic metrics on representative variable geopotential for APE and RoPE, each comparing the six frequency-initialisation schemes.

For RoPE the curves largely overlap on geopotential and are relatively close on aggregated skill. Across all variables a consistent ordering nonetheless emerges for lead times up to 72 h: the plain fixed (non-learnable) schemes rank worst, with log-spaced the weakest of all and spiral fixed ahead of it. Making the log-spaced and spiral (fixed) frequencies wrap results in a small but reliable gain, though still short of the learnable schemes. The learnable frequencies are best, with the log-spaced initialisation holding the overall edge.

For APE the picture is dominated by a single failure: log-spaced learnable frequencies collapse (last by a wide margin). Log-spaced fixed is significantly better but still only fifth of six, and making them wrap helps and makes the run cluster with the top performing ones. For the spiral layout the wrap trick does not help (aggregate skill goes down by $\sim 4\%$). The spiral fixed is the best scheme overall, with the baseline (spiral learnable) and log-spaced wrap close behind, and spiral wrap trailing a few points further back. All four are still well clear of log-spaced fixed and log-spaced learnable.

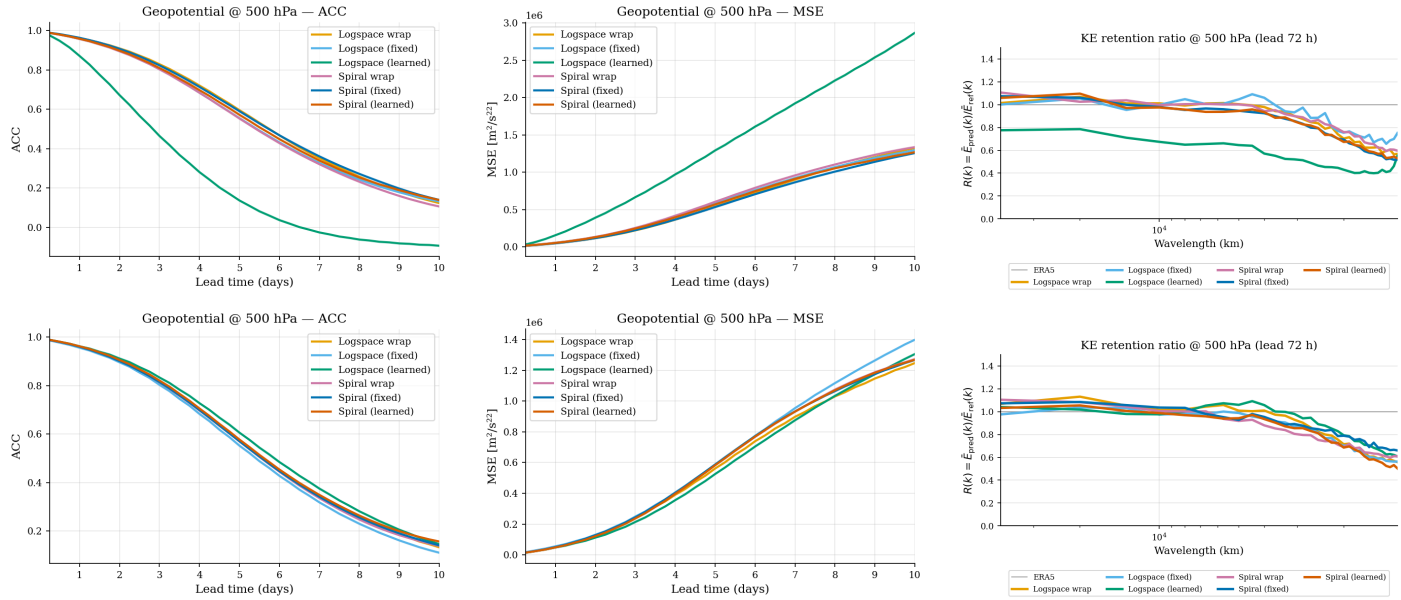
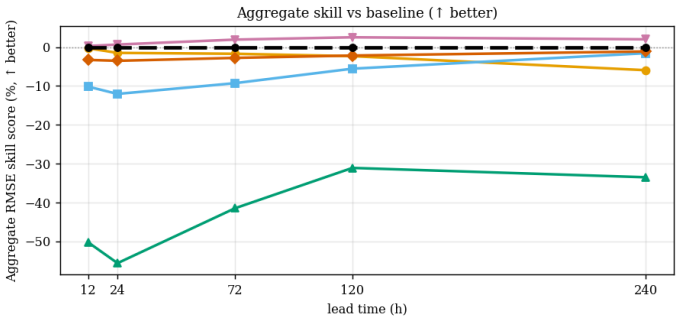
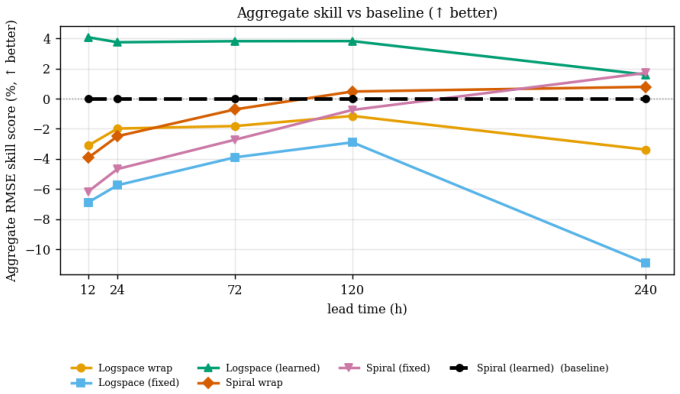


Figure 6.9: Positional-encoding frequency scheme ablation summary at 500 hPa (WB2 headline level), **top row** APE and **bottom row** RoPE. **Left column:** ACC of geopotential. **Middle column:** MSE of geopotential. **Right column:** kinetic-energy retention ratio at 72 h lead, R=1 matches ERA5.



(a) APE.



(b) RoPE.

Figure 6.8: Aggregated deterministic skill of the positional-encoding frequency scheme ablation for the full 10-day rollout, relative to the reference model with SPIRAL LEARNED initialisation.

Spectral Fidelity. The kinetic-energy retention ratio (Figure 6.9, right column) confirms the global picture: for RoPE, all six schemes sit close together. For APE, the five working schemes are clustered, while log-spaced learnable is problematic. The zonal spectra were hard to distinguish (Section H.3: Figure H.3, Figure H.4).

Insights

RoPE encodes the *relative* position between data points, so frequencies benefit from being learnable (they can adapt to the data during training) and from an initialisation that already reflects the actual distances in the data. A data-driven, log-spaced initialisation (for latitude and longitude) gives RoPE's learnable frequencies a starting point that results in higher skill at the end of training than the standard spiral initialisation. However, this advantage holds only when frequencies are learnable. With frequencies held fixed, spiral instead outperforms log-spaced (if the frequencies are not wrapped). We hypothesise that this is because spiral's broad coverage of joint lat/lon directions is a structural property of its construction. Whereas log-spaced's narrower, correlated coverage is a limitation that training can subsequently correct by shifting the learnable frequencies away from their initial values. No such correction is possible once frequencies are frozen.

APE encodes *absolute* position, so its role is closer to memorising a distinct code per grid location. Adaptivity does not help with this: shifting an already narrow, correlated log-spaced code-book away from its initial values only makes neighbouring grid locations harder to tell apart, which is consistent with the log-spaced learnable scheme collapsing specifically under APE. The (fixed) spiral initialisation instead gives the most evenly-covered coordinate space to memorise against, which is why it is the strongest scheme for APE regardless of whether frequencies

are learnable.

The choice of frequencies also interacts with the symmetry group. Without a rotation group, the best scheme per encoding is learnable log-spaced for RoPE and spiral fixed for APE. Under a rotation group (Cyclic-4), RoPE should switch to spiral learnable, because a rotation swaps the per-axis (lat/lon) frequencies, so the anisotropic log-spaced layout is not rotation-consistent whereas the isotropic spiral is (Appendix D). The small log-spaced edge for RoPE thus appears only in the rotation-free case.

6.6 Experiment 5: Equivariance and Weight Sharing

We evaluate the effect of strict and approximate equivariance realised through weight sharing. We model approximate equivariance to determine whether a flexible departure from exact symmetry improves performance. To break symmetry, we introduce symmetry-breaking Absolute Positional Encodings (APE), a trivial readout, and learnable local scalars and vectors, the latter following an approach similar to Ashman et al. [2024b] (notably, ECMWF's AIFS utilises comparable learnable parameters within a standard, non-equivariant APE-only framework [Lang et al., 2024]). These learnable local features are per-grid-point parameters, concatenated to the input scalar and vector fields before the group lift. Because they are initialised to zero, the model remains exactly equivariant at the start of training and can subsequently learn to break symmetry during optimisation.[4] We explore the properties of these learned features further in Chapter 7.

Modelling the Earth requires different symmetry groups depending on the chosen dimensionality of the space: As mentioned in Experiment 3, translation equivariance on a 2D latitude-longitude grid natively captures movement across the planetary surface, whereas in 3D Cartesian space, linear translation moves points off the spherical manifold, necessitating 3D rotational equivariance to represent valid surface shifts. To isolate the impact of these geometric inductive biases from sheer parameter scale, we evaluate configurations across two capacity tiers in both dimensionalities. Because the weight-sharing models share parameters across the group, maintaining a minimum attention-head dimension of 32 requires a large total parameter count (∼80M in 2D, ∼178M in 3D), even though only ∼15-20M of those parameters are trainable. A non-equivariant model at the same width shares nothing, so all its parameters are trainable, opening a large trainable-parameter gap between the two. To isolate geometry from raw capacity, we therefore add "small" non-equivariant baselines whose trainable-parameter count (∼20M) is matched to the equivariant models. All configurations are listed in Table 6.2.

[4] Since these parameters are unconstrained, the model is not forced to learn exclusively symmetry-breaking patterns; it may also capture other features that generally improve overall performance.

Results

2D Symmetry: Cyclic-4 Group

Deterministic skill. The aggregated deterministic skill relative to the reference model is shown in Figure 6.10. Deterministic metrics of representative variable geopotential are shown in Figure 6.11. Across all configurations, the ACC, MSE and aggregated skill ordering is led by the large trivial run (with or without APE) and is broadly consistent across variables and levels. A clear second band (large APE-ONLY, CYCLIC-4 TRIVIAL-READOUT and its APE variant) tracks closely; notably CYCLIC-4 TRIVIAL-READOUT reaches nearly the same skill as large APE-ONLY, with the small remaining gap narrowing further over the full 10-day rollout (∼10% to ∼5% on aggregated skill). In the lower band, the other CYCLIC-4 variants (plain, APE, or symmetry-breaking) outperform the small trivial models at short lead and lose their advantage after 72 h. The strict CYCLIC-4 models destabilise at long lead, more so with APE and most of all with the symmetry-breaking features; the trivial-readout variants are unaffected. We further explore this in Section 7.2.

Table 6.2: Overview of the equivariance regimes evaluated in the 2D and 3D experiments. The translation group T is implemented via Rotary Position Embedding (RoPE). Symmetries are modelled using the cyclic-4 group (C_4) for 2D and the tetrahedral group ($\mathcal{T}$) for 3D. In the Group column, the notation $T \rtimes C_4$, $\mathcal{T}$ compactly indicates the roto-translation groups used for the 2D and 3D configurations, respectively.

			Params (M)	
Regime	**Group**	**Symmetry-breaking**	**2D**	**3D**
Vanilla - Non-equivariant	$\{e\}$	APE	79.4	178.0
Vanilla - Non-equivariant (small)	$\{e\}$	APE	19.9	19.9
Translation-only	T	None	79.4	178.0
Translation-only (small)	T	None	19.9	19.9
Approximate Translation-only	T	APE	79.4	178.0
Approximate Translation-only (small)	T	APE	19.9	19.9
Strict equivariant	$T \rtimes C_4$, $\mathcal{T}$	None	19.9	15.0
Approximate equivariant	$T \rtimes C_4$, $\mathcal{T}$	APE	19.9	15.0
Approximate equivariant	$T \rtimes C_4$, $\mathcal{T}$	Non-equivariant readout	20.1	15.8
Approximate equivariant	$T \rtimes C_4$, $\mathcal{T}$	APE + Non-equivariant readout	20.1	15.8
Approximate equivariant	$T \rtimes C_4$, $\mathcal{T}$	Learnable local features	19.9	15.0

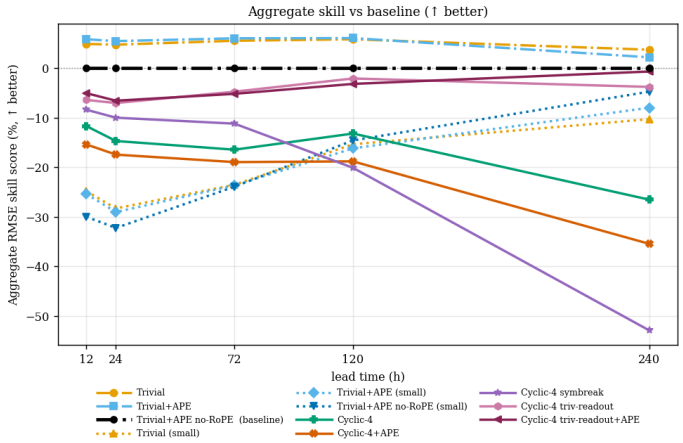


Figure 6.10: Aggregated deterministic skill of the 2D equivariance and weight-sharing ablation over the 10-day rollout, relative to the reference model configuration with APE ONLY.

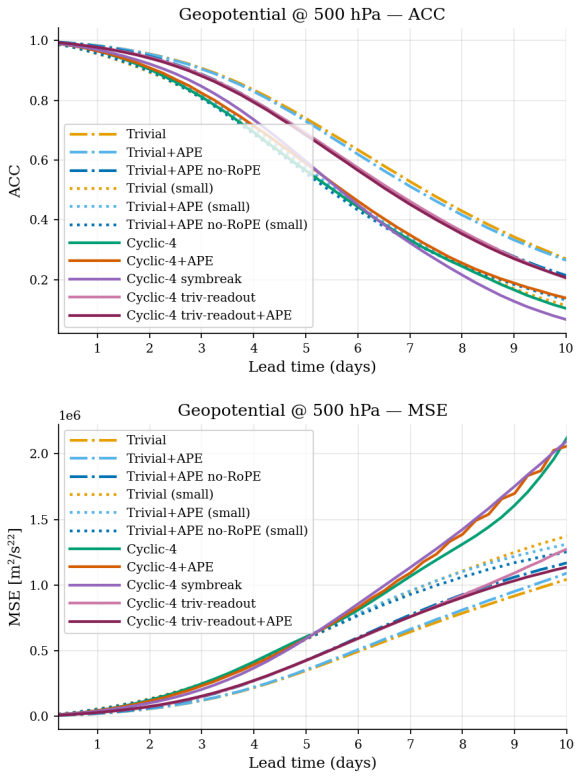


Figure 6.11: Deterministic skill of representative variable geopotential at 500 hPa (WB2 headline level) for the 2D equivariance and weight-sharing ablation: ACC **(top)** and MSE **(bottom)**.

Spectral Fidelity. On KE spectra (Figure 6.12, top row), at 12 h, the runs are hard to separate. The CYCLIC-4 TRIVIAL-READOUT models (with or without APE) are marginally the most faithful, closely followed by the large models; the SYMMETRY-BREAKING and APE-ONLY runs are the worst within each capacity class, with the remainder of the runs sitting in between (Figure 6.12, top row). At 72 h the same ordering holds, but is more pronounced: the small APE-ONLY run injects spurious small-scale energy (a high-wavenumber noise peak), and among the trivial backbones APE+RoPE is more faithful than RoPE, which in turn beats APE-ONLY. The CYCLIC-4 TRIVIAL-READOUT (+APE) models remain most faithful. The zonal spectra separate the models only weakly, at 72 h the curves are close and mostly inseparable (representative variables in Figure 6.12, bottom row).

3D Symmetry: Tetrahedral Group

The 3D tetrahedral group ($\mathcal{T}$, G=12) follows the same capacity-dependent pattern as the 2D CYCLIC-4 case, but with a weaker payoff (Section H.4). At matched (small) capacity the group-equivariant TETRA models are better on deterministic metrics than the non-equivariant baselines, though the advantage is weaker and less consistent than in 2D; kinetic-energy and zonal-spectra diagnostics tell the same story but separate the models only weakly.

2D vs. 3D symmetry

Lastly the group-equivariant families are compared directly against one another: the 2D CYCLIC-4 and the 3D TETRAHEDRAL variants, matched configuration for configuration (Section H.5). The 2D CYCLIC-4 variants are strictly better than their counterparts at short-to-medium lead: every CYCLIC-4 variant beats its tetrahedral counterpart (e.g. CYCLIC-4 + APE vs. TETRA + APE) across the headline fields. At the longest lead times the two families perform comparably.

Insights

Geometric symmetry is useful as an inductive bias, but not as a hard constraint. Two results support this. First, adding translation equivariance (RoPE) helps unconditionally: RoPE-based configurations improve over the APE-only baseline, pointing to a real translational symmetry in weather, while the presence of APE matters little. Second, relaxing the group beats enforcing it: the trivial-readout (approximately equivariant) variants outperform the strict CYCLIC-4 models, which destabilise at long lead. Weight sharing across the four rotations buys effective capacity: CYCLIC-4-TRIVIAL-READOUT (+APE) reaches nearly the skill of the large APE model at roughly a quarter of the trainable parameters, and is the most KE-faithful configuration of the family. Finally, the 2D CYCLIC-4 variants beat their 3D TETRAHEDRAL counterparts, and the TETRAHEDRAL payoff over the non-equivariant baselines is weaker, consistent with Experiment 3: the lower-dimensional surface is the easier manifold to learn on.

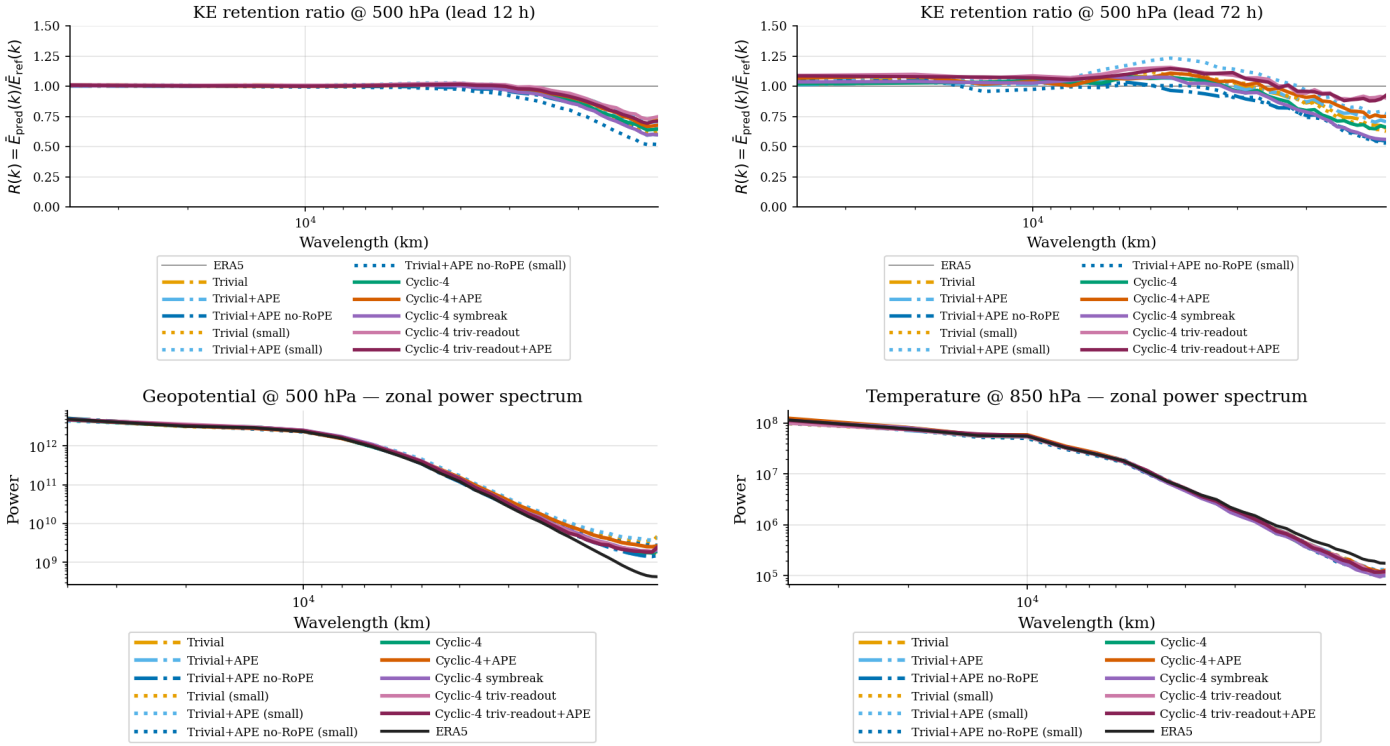


Figure 6.12: Spectral fidelity of the 2D equivariance and weight-sharing ablation at 500 hPa (WB2 headline level unless noted). **Top row:** kinetic-energy retention ratio (12 h lead left, 72 h lead right), $R{=}1$ matches ERA5. **Bottom row:** zonal power spectra at 72 h lead (geopotential left, temperature at 850 hPa right), ERA5 (black) is the reference.

Chapter 7

Symmetry-Breaking Features

The fundamental equations of fluid flow have no preferred origin or orientation, and by Noether's theorem these symmetries are exactly what produce the conserved quantities of the dynamics (linear and angular momentum); this makes roto-translation equivariance a physically natural inductive bias. Earth's boundary conditions, however, break these symmetries: its geography is fixed to absolute locations, and the Coriolis force changes sign between the hemispheres. A strictly equivariant model cannot represent these effects on its own. This raises two questions:

> *Question 1: What structural and physical properties does the model learn to inject through these symmetry-breaking features (Experiment 6)?*
>
> *Question 2: Does learning these features improve deterministic forecast skill, and does this depend on the type of equivariance enforced (Experiment 7)?*

Like in Section 6.6, we add a small set of learnable, per-position bias features, zero-initialised so the model starts out exactly equivariant and only breaks the symmetry where the data demands it.

7.1 Experiment 6: Interpreting the Learned Symmetry-Breaking Features

We investigate what features the model learns to break symmetry when strict equivariance is relaxed with learnable features, as explained in Section 6.6. We restrict this analysis to 2D, visualising the learned bias channels for 8 different configurations (Table 7.1). By comparing across four degrees of equivariance, we interpret what each model has learned to break symmetry. Each configuration reuses the backbone from Experiment 5 (2D), augmented with six learnable scalars and one learnable vector prepended to the model input. We treat this as exploratory interpretability: rather than claiming these patterns *are* what the model learns or what *drives* its skill, we look for correlations between known meteorological structure and what the learned channels appear to encode. The inspection is qualitative: many channels are noisy or possibly meaningless, and the attribution to physical templates is partly subjective. Finally, there is no reason each scalar channel should individually encode a single physical field: because the channels are free learned features, a given signal may be split across several of them or carried by the set as a whole.

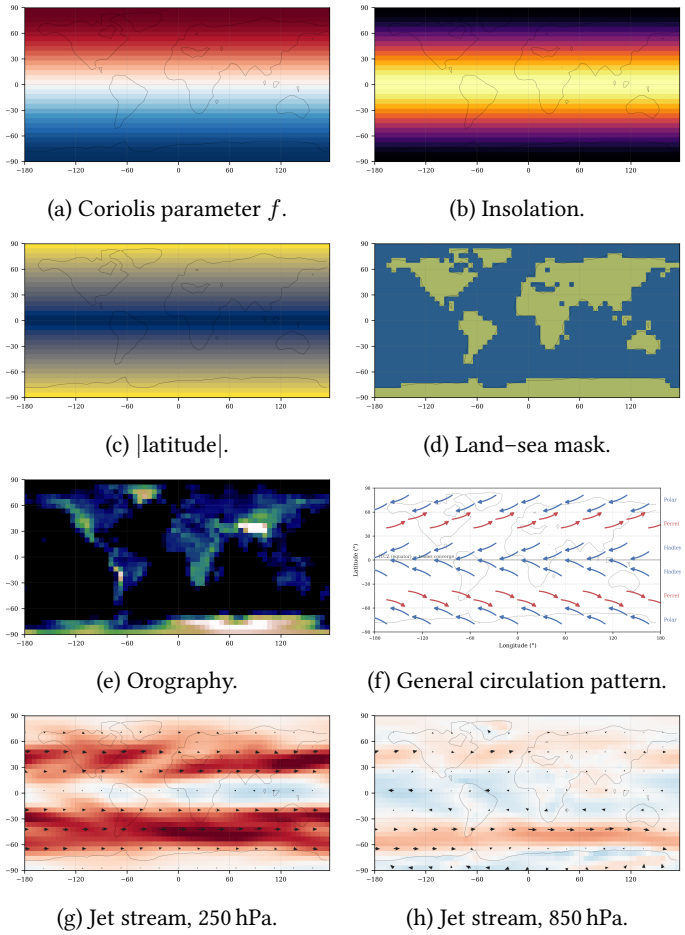
Symmetry-breaking references

Before reading the learned channels we hypothesise the physical fields that symmetry-breaking features could plausibly encode. Figure 7.1 shows the reference templates against which the learned channels are compared: the Coriolis parameter $f = 2\Omega \sin \varphi$, insolation, |latitude|, the land-sea mask, and orography.[1] Three further references come from the atmosphere's general circulation. Because the equator receives more solar energy than the poles, the atmosphere overturns to transport heat poleward; on a rotating planet this overturning organises into three cells per hemisphere (Hadley, Ferrel, Polar). Coriolis deflection of the surface branches of these cells produces quasi-stationary surface wind belts: trade easterlies, mid-latitude westerlies, and polar easterlies (Figure 7.1f). Aloft, the equator-to-pole temperature gradient concentrated at the cell boundaries implies, through thermal-wind balance, that the westerlies strengthen with height, peaking as narrow jet-stream cores near the tropopause at 200–300 hPa; Figure 7.1g and Figure 7.1h show these jets directly in ERA5 (referred to as jet / zonal-circulation banding in the remainder of this chapter). All of these structures are pinned to absolute latitude and the rotation axis.

Table 7.1: Design of the symmetry-breaking analysis.

		Static inputs	
Symmetry level	**Group**	**With**	**Without**
Translation-only	T	✓	✓
Roto-translation (strict)	$T \rtimes C_4$	✓	✓
Roto-translation (trivial readout)	$T \rtimes C_4$	✓	✓
Non-equivariant (APE)	$\{e\}$	✓	✓

[1] Note that f, insolation and |latitude| are all functions of latitude alone and hence mutually correlated. We therefore leave latitude out of the comparison.

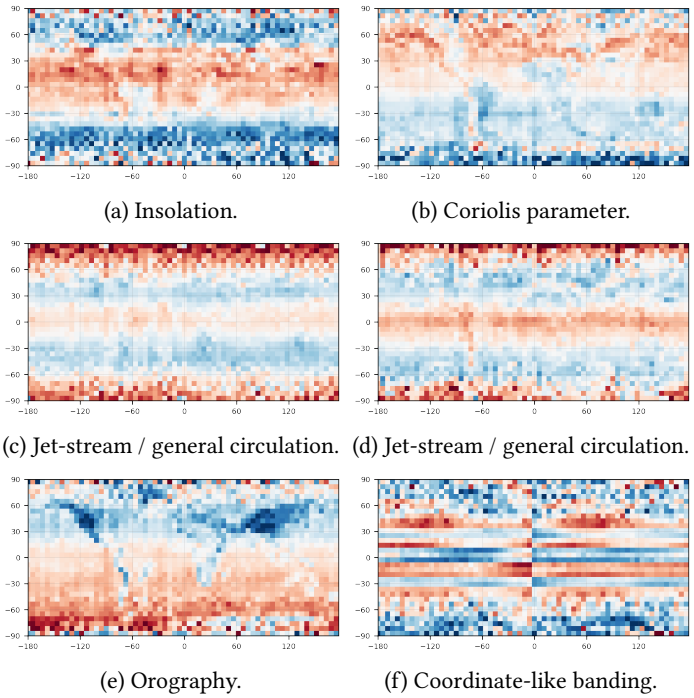


(a) Coriolis parameter f. (b) Insolation. (c) |latitude|. (d) Land–sea mask. (e) Orography. (f) General circulation pattern. (g) Jet stream, 250 hPa. (h) Jet stream, 850 hPa.

Figure 7.1: Symmetry-breaking reference fields. **Rows 1–2:** Coriolis parameter, insolation, |latitude|, and land–sea mask. **Row 3:** orography and general-circulation pattern, i.e. the Coriolis-deflected surface wind belts (arrow colour = zonal sign of the deflection). **Row 4:** the ERA5 climatological (2020) jet stream at 250 hPa and 850 hPa, the upper-level branch of the same general circulation.

Table 7.2: Learned symmetry-breaking scalars (`s0`–`s5`) mapped to physical templates, by symmetry level and static-features on/off (static). Entries are the scalar-channel indices that align with each template; "–" means no channel. † means tentative.

Regime	Static	Coriolis	Insolation	Jet/circ.	Orography	Uninterp.
Strict ($T \rtimes C_4$)	w	`s2`	`s0`	`s1,s5`	–	`s3,s4`
	w/o	–	–	`s5`	`s0,s3`	`s1,s2,s4`
Broken ($T \rtimes C_4$)	w	`s1`	`s3,s5`	`s0`	–	`s2,s4`
(triv.)	w/o	–	`s0`	`s1,s2,s5`	`s0,s1,s3,s4`	**none**
Transl. (T)	w	`s0`	–	`s2,s4`	–	`s1,s3,s5`
	w/o	–	–	`s0,s5`†	`s2`	`s1,s3,s4`
APE	w/wo	*coordinate-like banding*				



(a) Insolation. (b) Coriolis parameter. (c) Jet-stream / general circulation. (d) Jet-stream / general circulation. (e) Orography. (f) Coordinate-like banding.

Figure 7.2: Examples of learned symmetry-breaking scalars. **(a,b)** Insolation and the Coriolis parameter, and **(c,d)** jet-stream / general-circulation banding, all from the roto-translation equivariant (Cyclic-4, with static features) run. **(e)** Orography, learned by the symmetry-broken trivial-readout equivariant run without static inputs. **(f)** The non-equivariant APE-only baseline (without static inputs) reproduces coordinate-like bands rather than a physical field.

Results

Table 7.2 lists, per run, which of the six learned scalar channels align with each physical template and how many are uninterpretable. We were unable to interpret the learned *vector* channel in all cases, so it is excluded from the analysis below and an example is shown only for completeness in Appendix I, together with the full overview of the learned features. Examples of learned features can be seen in Figure 7.2. Across every (approximately) equivariant configuration the same handful of physical signals keep re-appearing in the learned bias: a Coriolis-like channel, latitude-banded insolation, and jet-stream / general-circulation structure. When the static geographic inputs are withheld, every (approximately) equivariant configuration compensates by learning geography. The belt structure, by contrast, emerges in every (approximately) equivariant configuration regardless of static inputs. The non-equivariant APE baseline follows neither pattern: its channels reproduce coordinate bands rather than physical fields.

Insights

Answering Question 1: the learned biases are partly interpretable and consistently recover known symmetry-breaking fields: a Coriolis-like channel, latitude-banded insolation, and jet-stream / general-circulation banding appear in nearly every equivariant configuration. The non-equivariant APE-only baseline is the sole exception, tying these structures directly to its absolute positional encoding rather than learning them as channels. This is the expected behaviour: in the equivariant configurations the learned bias is the only pathway through which absolute position can enter the network, so training fills it with the position-dependent

physical fields the dynamics require, whereas the APE-ONLY baseline already receives absolute position directly through its absolute positional encoding, leaving the bias channels redundant. It re-expresses the coordinate code it already relies on. Withholding the static geographic inputs does not remove this behaviour but redirects it: the model instead learns to encode geography explicitly, using it as a stand-in source of absolute position. Together these results hint at which inductive structures (e.g. an explicit orography or Coriolis channel) should be built into future models directly, rather than left for the network to discover implicitly.

7.2 Experiment 7: Do Symmetry-Breaking Features Improve Skill?

Experiment 6 characterises *what* the model learns to inject. Here, we ask whether that information is actually *useful*, and whether the answer depends on how strictly equivariance is enforced. We supply symmetry-breaking signal in two ways: (i) appending the **static geographic features** directly as inputs, and (ii) adding the **learned symmetry-breaking features** of Experiment 6. We evaluate the same runs as in Table 7.1.

Our prior hypothesis is that the static features should help the equivariant models. The underlying physical laws are symmetric; the asymmetry of the atmosphere comes from its boundary conditions (orography, the land-sea contrast, insolation). A strictly equivariant model without access to these fields must either ignore them or distort its shared weights to approximate them. Supplied as inputs, however, the geography is no longer something the model has to explain, only something it has to respond to, and that response is governed by the same physics everywhere on the globe. What remains to be learned is therefore symmetric again, exactly the function class an equivariant model represents.

Results

Learned features.

Figure 7.3 (top) shows that when adding the learned symmetry-breaking features (with static features) improves the deterministic skill on short to mid range leads uniformly. On the strict CYCLIC-4 model at late lead, the error compounds the static blow-up, collapsing to -23.8% at day 10. Without static features (Figure 7.3, bottom), the picture is more consistently positive: the learned bias helps every family at short lead, and only turns negative for Cyclic-4 (strict) and APE at day 10; the trivial-readout Cyclic-4 model dips negative at mid-range leads but rebounds to a large gain ($+6.1\%$) by day 10. When looking at spectral fidelity, the features degrade performance (representative example shown in Figure 7.4). On all configurations, adding the learned symmetry-breaking features consistently degrades spectral performance.

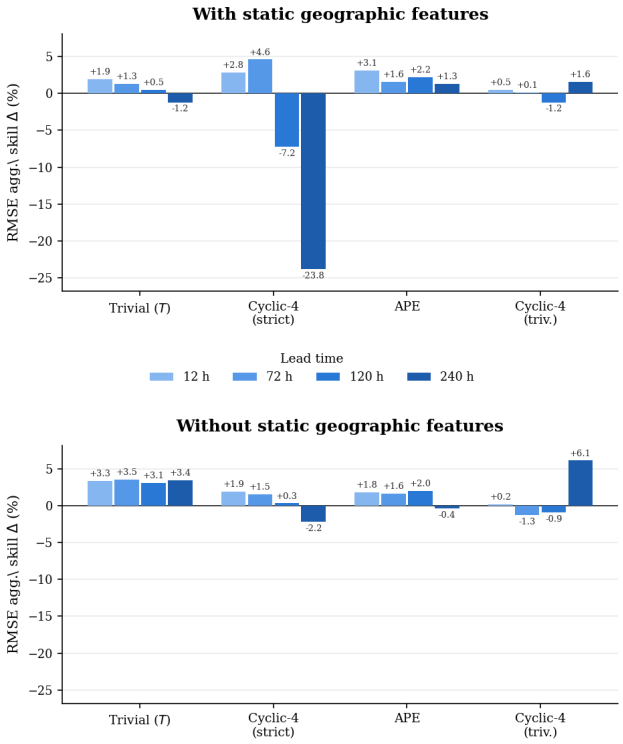
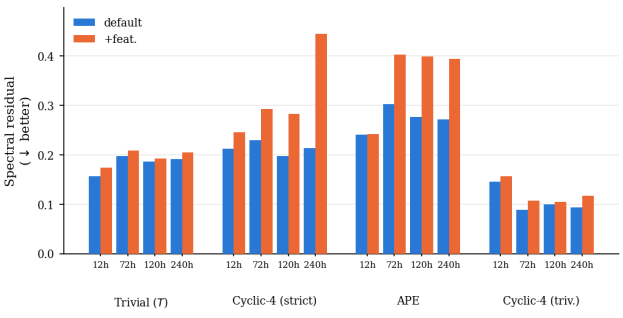
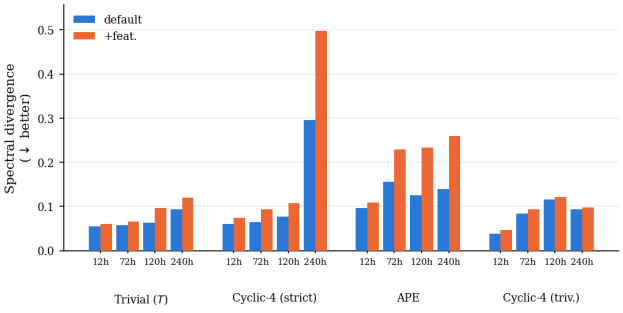


Figure 7.3: **Effect of adding the learned symmetry-breaking features** on aggregated deterministic RMSE skill, by symmetry family and lead time. Values are % change relative to the matching $-$`symbreak` baseline (↑ better). **Top:** with static geographic features. **Bottom:** without static geographic features.



(a) Spectral residual.



(b) Spectral divergence.

Figure 7.4: **The symmetry-breaking effect on spectral fidelity of KE at 500 hPa** (static inputs on) for the four symmetry families (↓ better for both scores).

Static geographic features

Concatenating the static geographic features to the input helps the translation-only and CYCLIC-4 TRIVIAL-READOUT models (Figure 7.5): removing static features costs aggregate skill at every lead, and the gap widens at longer leads. The non-equivariant APE-ONLY baseline is largely indifferent: the sign of the effect flips across leads, but the magnitude stays small. The strict CYCLIC-4 model shows the opposite pattern: it is better without static features, and the gap in its favour widens over time.[2] The MSE results on representative variable mean-sea-level-pressure makes this static-input effect on the roto-equivariant models concrete (Figure 7.6). For the strict CYCLIC-4 model (top), by day 10 the no-static run sits near 1×10^6 Pa^2, the static run is roughly $3\times$ higher ($\sim3.3\times10^6$ Pa^2), and adding the learned symmetry-breaking bias on top blows the model up to $\sim5\times10^6$ Pa^2. Breaking the symmetry with the trivial readout (bottom) removes the blow-up entirely: all four variants collapse onto $\sim1\times10^6$ Pa^2. Note that the y-axis now tops out at 1×10^6 rather than 5×10^6 Pa^2.

Where removing static symmetry-breaking features costs aggregate RMSE skill, the effect on spectral fidelity is neutral at 500 hPa (Figure 7.7, top row), the two arms split the columns evenly and no family shows a consistent direction. At 850 hPa (Figure 7.7, bottom row), static symmetry-breaking features instead help spectral fidelity: the `default` (no-static) arm wins 25 of the 32 columns. The strict roto-equivariant run is the exception, degrading in spectral fidelity.

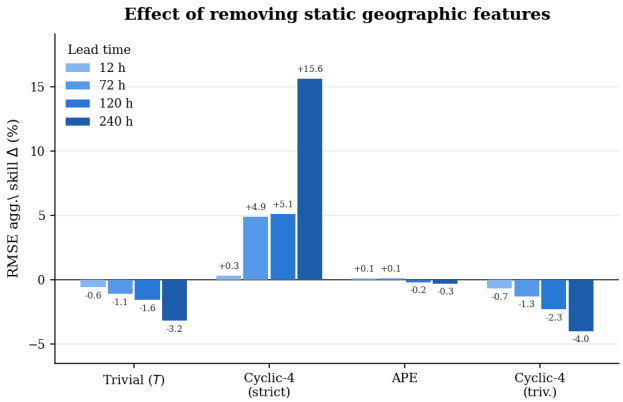
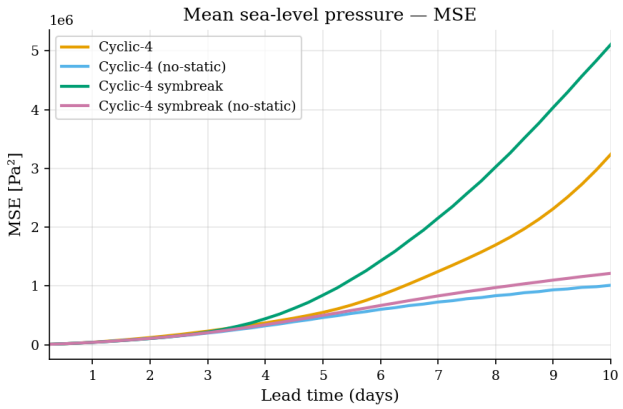
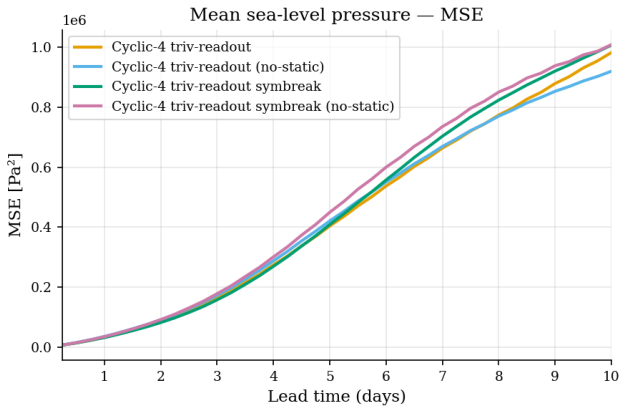


Figure 7.5: **Effect of removing the static geographic features** (with the symmetry-breaking bias off) on aggregated deterministic RMSE skill, relative to the `+static −symbreak` baseline, averaged over the six WB2 headline fields, by symmetry family and lead time. Values are % change (↑ better).



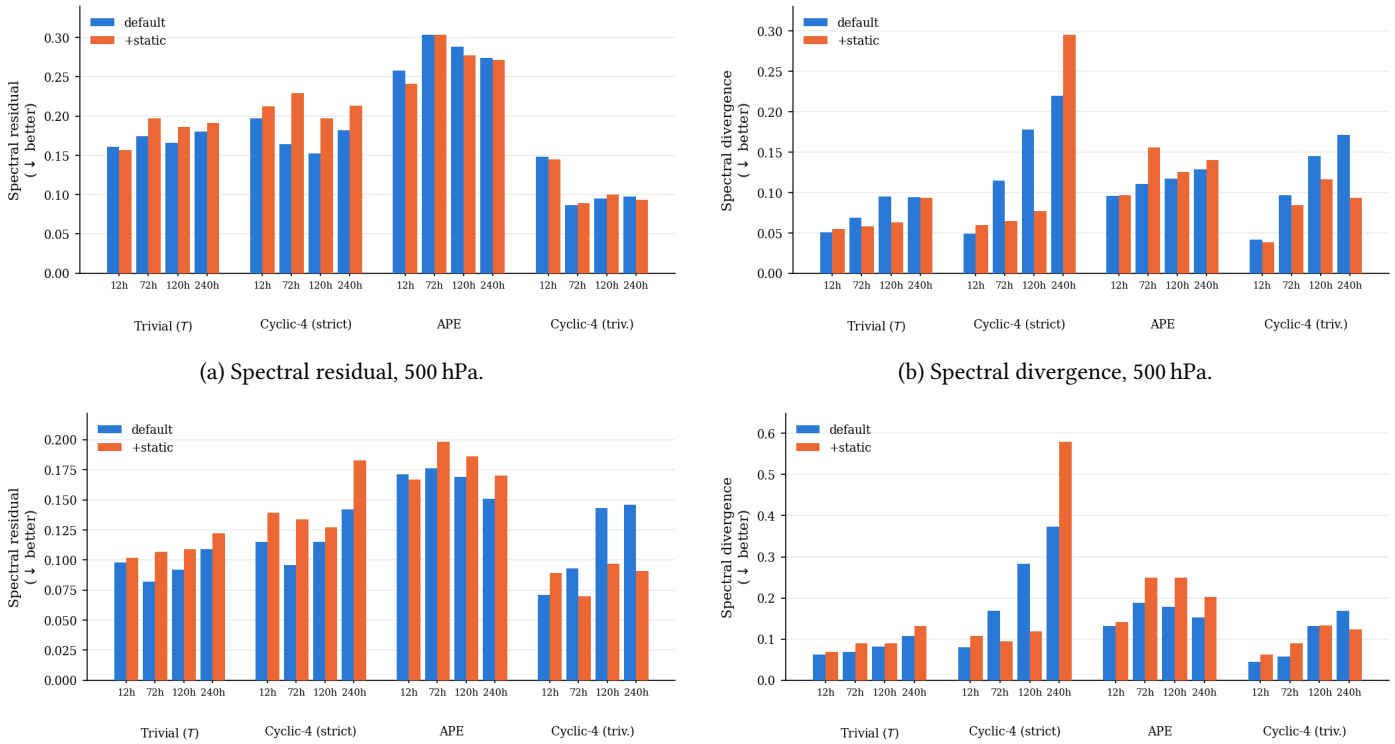
(a) Strict roto-equivariant Cyclic-4.



(b) Cyclic-4 with trivial readout.

Figure 7.6: Mean-sea-level-pressure MSE (WB2 headline level) over the 10-day rollout, each model with/without static inputs and with/without the learned symmetry-breaking bias.

[2] The implications of the equivariance and weight sharing experiment (Section 6.6) are unchanged; see the full with/without-static comparison in Section I.2 (Figure I.13).



(a) Spectral residual, 500 hPa.

(b) Spectral divergence, 500 hPa.

(c) Spectral residual, 850 hPa.

(d) Spectral divergence, 850 hPa.

Figure 7.7: **The static-input effect on spectral fidelity of KE** (learned symmetry-breaking bias off) for the four symmetry families (↓ better for both scores), at 500 hPa **(top row)** and 850 hPa **(bottom row)**.

Insights

Under the deterministic setting, the learned symmetry-breaking features mostly help deterministic skill at short lead across every family, and stay close to neutral for most configurations further out. The one exception is the strict Cyclic-4 model: stacked on top of static geography it already cannot use well, the learned bias compounds that existing blow-up, collapsing skill severely by day 10. However, it buys its skill at the cost of spectral fidelity. It seems to smooth toward the conditional mean: it suppresses fine-scale structure, which the deterministic scores reward while the spectra penalise. In other words, the learned bias improves headline skill by *blurring* rather than by resolving genuinely sharper structure. Because this gain comes from blurring, it is likely tied to deterministic MSE training and would probably not survive a probabilistic (CRPS) objective, which rewards calibrated sharpness rather than smoothing toward the conditional mean. We therefore view this specifically as a tool to discover symmetry-breaking features rather than a modelling design choice like ECMWF's AIFS [Lang et al., 2024].

Concatenating the static geographic inputs improves skill in every configuration except the strict roto-translation equivariant one, where it instead hurts, with the damage growing over lead time. Once the strict symmetry is broken (the trivial readout), the same input becomes beneficial; the purely translation-equivariant model also benefits, and the non-equivariant APE baseline, which already encodes absolute position, is nearly indifferent. The static symmetry-breaking features thus do improve performance, but only under specific circumstances. This rejects our hypothesis: rather than restoring the broken symmetry, the static fields destabilise the strict roto-translation equivariant model. *Answering Question 2*: the learned features buy short-range deterministic skill, but through blurring, and static inputs help precisely when strict roto-translation equivariance is not enforced.

Chapter 8

Stress-Testing the Inductive Bias of Approximate Equivariance

8.1 Experimental Setup

The ablations in experiment 5 established that geometric symmetry is valuable as an inductive bias, but not as a hard constraint. This chapter stress-tests that inductive bias. It compares a NON-EQUIVARIANT baseline against the best-performing APPROXIMATELY EQUIVARIANT model from experiment 5. We ask two questions of these final configurations, each probing an axis the ablations held fixed. *Question 1: How does the gap between approximate equivariance and no equivariance scale with capacity?* (Section 8.3): we fit parameter scaling curves on next timestep prediction across three widths to test whether the gap persists, narrows, or grows. *Question 2: To what extent does the inductive bias of approximate equivariance help probabilistic forecasting?* (Section 8.4): we convert both configurations into ensemble forecasters and compare probabilistic skill and calibration.

Both final configurations share the same backbone, synthesising the best-performing settings from the earlier ablations in Chapter 6: HEALPix mesh topology (Experiment 1), the MSE-magnitude-angle vector loss (Experiment 2), modelling in 2D (Experiment 3), and spiral positional-encoding frequencies, learnable for RoPE and fixed for APE (Experiment 4).[1] On top of this shared backbone, we compare two configurations: (i) an approximately equivariant model, CYCLIC-4 TRIV-READOUT + APE, and (ii) a matched non-equivariant APE-ONLY model. The full hyperparameter and architecture details are given in Section G.4.2. As in the ablation study, neither experiment uses rollout fine-tuning, due to computational constraints; we expect the findings to carry over and leave confirmation to future work.

Ensemble forecasters & Uncertainty

The models of the ablations each produce a single next state and are trained deterministically; they approximate the conditional mean over future states, which is smoother than any individual realisation and therefore systematically suppresses fine-scale spectral power [Alet et al., 2025]. Since Question 2 is evaluated in the probabilistic setting, we convert two configurations into ensemble forecasters. This is done by fine-tuning the models on CRPS, as is common in the weather forecasting domain [Diaconu et al., 2026].

Learned functional perturbations. We follow the setup of Zhdanov et al. [2026]: rather than perturbing the input state, we inject a single global latent $\mathbf{z}$ into the network weights, so that one draw perturbs the entire forecast in a globally consistent way. Concretely, $\mathbf{z} \in \mathbb{R}^{32}$ is sampled once per ensemble member per forward pass (i.e. per rollout step) from $\mathcal{N}(\mathbf{0}, \mathbf{I})$, reshaped by a near-zero-initialised linear "mixer", and threaded to every transformer block as an additive bias inside the SwiGLU gate:

$$\widehat{\mathrm{SwiGLU}}(\mathbf{x}, \mathbf{z}) = \big(\, \sigma(\mathbf{x}W_g + \mathbf{z}) \odot (\mathbf{x}W_v) \,\big)\, W_{\text{out}}, \tag{8.1}$$

where σ is the Swish activation.

Keeping perturbations equivariant. The only departure from Zhdanov et al. [2026] is required by the group structure: a perturbation that differed across the G group slots of our features would break equivariance. We therefore copy-lift $\mathbf{z}$ identically across the G slots and project it with an equivariant PLATONIC linear layer before entering each block. The full lifting construction and invariance argument are given in Section B.3.1.

CRPS fine-tuning. Both models are warm-started weights-only from their deterministic checkpoints; only the fresh noise head is randomly initialised. We then fine-tune with the latitude- and variable-weighted *fair* (unbiased) CRPS objective of Zamo and Naveau [2018], following Alet et al. [2025].

$$\mathrm{CRPS}(\mathbf{x}^{1:N}, y) = \frac{1}{N}\sum_{n} |x^n - y| - \frac{1}{2N(N-1)} \sum_{n,n'} |x^n - x^{n'}|, \tag{8.2}$$

with $N{=}2$ members per sample during training. Further details are deferred to Section G.4.3.

8.2 Evaluation

Deterministic models. Runs are scored with aggregated RMSE for deterministic skill. We additionally report two diagnostics that probe the physical realism (KE) of the forecasts directly, following Kasteleyn et al. [2026]: the spectral residual and the spectral divergence, both computed on the 500 hPa kinetic-energy spectrum $E(k)$ and both zero for a perfect match (lower is better). The spectral residual asks whether the model gets the energy right at each scale. The spectral divergence instead asks whether it gets the shape of the cascade right, independent of overall energy level: a mismatch here means energy is misallocated across scales, for example excess smoothing versus spurious gridpoint noise. Full definitions are given in Section G.3.

Probabilistic models. Each ensemble ($M{=}10$ members) is characterised along three axes, all computed per grid point, latitude-weighted, globally averaged, and reported as a function of lead time: *CRPS* (probabilistic skill; lower is better), *ensemble-mean RMSE* (the deterministic skill of the mean, lower is better), and the *spread-to-skill ratio* (SSR; calibration, with $\mathrm{SSR} \approx 1$ well-calibrated and $\mathrm{SSR} < 1$ over-confident). Formal definitions are given in Section G.3.3. Beyond these ensemble scores, we also

[1] Spiral rather than the log-spaced scheme that led in Experiment 4's group-free setting, since the geometry configuration below adds a Cyclic-4 rotation group, under which log-spaced's anisotropic frequencies are no longer rotation-consistent.

evaluate each CRPS-finetuned configuration on deterministic skill and spectral fidelity, using two estimators: the ensemble mean and a single ensemble member.

Why no external baselines. Throughout this chapter we compare only between our models, not against baselines such as GraphCast or GenCast. Those systems produce native $0.25°$ to $1.5°$ forecasts, conservatively regridded down to our 64×32 ($5.6°$) grid. Regridding both preserves their accurate large scales *and* averages away the hard-to-predict small-scale error, a free skill reduction that our natively-$5.6°$ model never receives. A head-to-head would therefore measure the resolution gap, not the modelling approach.

8.3 Parameter Scaling

To answer Question 1, we fit parameter scaling curves (skill vs. model size), comparing the NON-EQUIVARIANT model against the APPROXIMATELY EQUIVARIANT one across three widths at matched total parameter count, ∼20M, 80M and 178M total parameters; per-width architecture and both the total and trainable parameter counts are given in Table G.5 (Section G.4.2). Because the APPROXIMATELY EQUIVARIANT model shares weights across the group, the same three widths correspond to far fewer trainable parameters (5M, 20M and 45M); we therefore report every metric against both trainable and total parameters.

Results

Deterministic skill. Figure 8.1 (left column) shows the aggregate RMSE skill at 72 h against total (top) and trainable (bottom) parameters. On total parameters, at the smallest width, the NON-EQUIVARIANT configuration is clearly ahead: a ∼24-point gap at 20M. This gap closes rapidly with scale: it narrows to ∼11 points at 80M and to ∼8 points at 178M. On trainable parameters, the APPROXIMATELY EQUIVARIANT model is always ahead: it reaches a given skill level with fewer trainable parameters (e.g. its 45M-trainable point already matches NON-EQUIVARIANT's 80M-trainable point).

KE spectral fidelity. Figure 8.1 (middle and right columns) reports the two single-number spectral metrics at 72 h against total (top row) and trainable (bottom row) parameters. On total parameters, the APPROXIMATELY EQUIVARIANT configurations have a better spectral fidelity than the NON-EQUIVARIANT ones at every width tested, and unlike the deterministic gap, this gap does not continue to close with scale: the spectral-residual gap goes from ∼0.20 at the smallest width to ∼0.10 at the middle width and back up to ∼0.11 at the largest; the spectral-divergence gap barely changes (from ∼0.11 to ∼0.10 to ∼0.09). On trainable parameters, the APPROXIMATE EQUIVARIANT model is always ahead, having better spectral fidelity per trainable parameter.

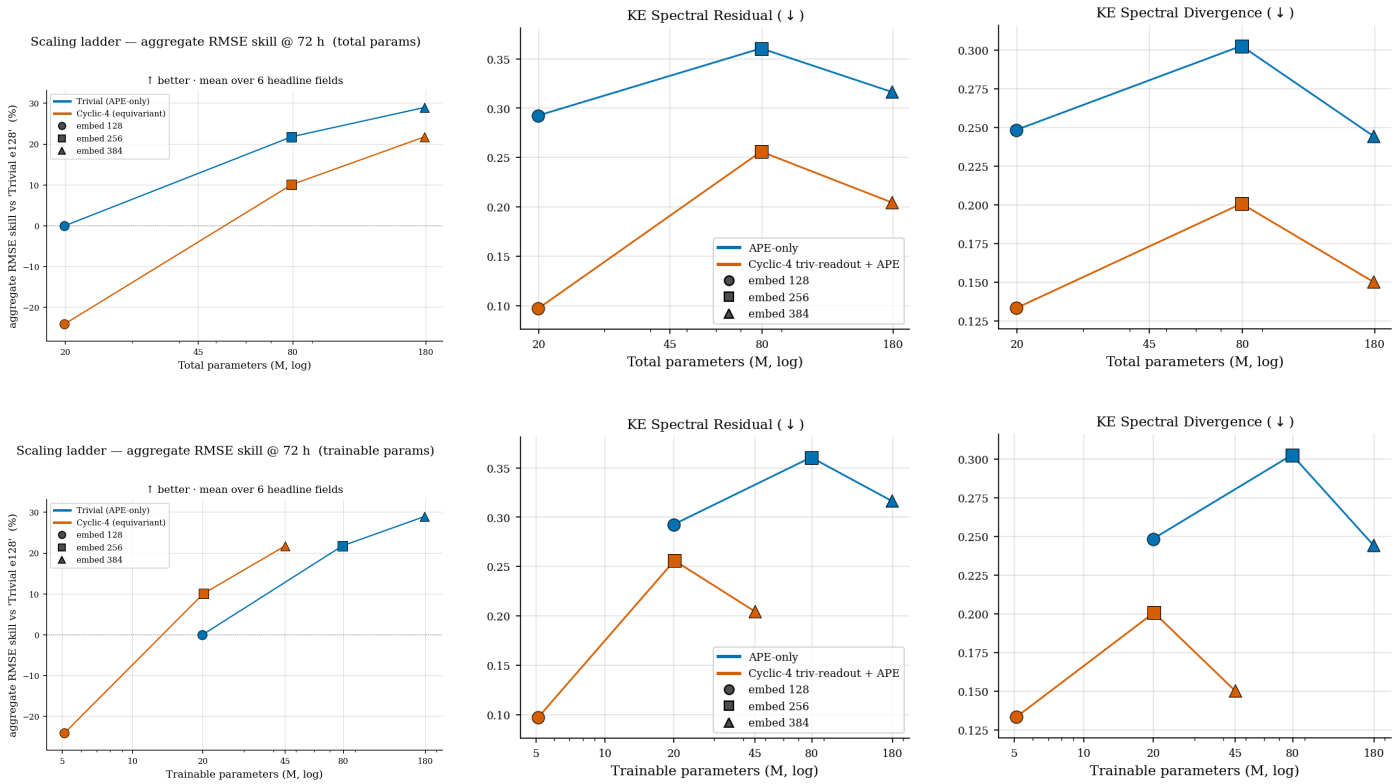


Figure 8.1: Scaling ladder at 72 h lead, APE-ONLY (blue) vs. CYCLIC-4 TRIV-READOUT + APE (orange). **Top row:** against *total* parameter count. **Bottom row:** against *trainable* parameter count. **Left column:** aggregate RMSE skill, % improvement relative to the smallest APE-ONLY model. **Middle column:** KE spectral residual. **Right column:** KE spectral divergence (↓ better for both spectral scores).

Insights

Answering Question 1: scale buys deterministic skill, but not physical consistency. The two families of metrics behave differently at total parameters. On deterministic skill, the NON-EQUIVARIANT-APPROXIMATELY EQUIVARIANT gap seems to be a function of capacity. The spectral gap, by contrast, does not close over the tested range, indicating that the spectral fidelity advantage is a consequence of the inductive bias itself rather than something the NON-EQUIVARIANT model converges to with more parameters. Against trainable parameters, APPROXIMATE EQUIVARIANCE acts as a roughly scale-independent efficiency multiplier: each stored weight is reused across the group, and the model reaches a given skill level with a lower number of learned parameters.

8.4 Ensembles and Uncertainty

To answer Question 2, we convert the two largest configurations, the NON-EQUIVARIANT model at 178M trainable parameters and the APPROXIMATELY EQUIVARIANT model at the matched width (45M trainable), into ensemble forecasters.

Results

Probabilistic metrics. Figure 8.2 shows results for the representative variables. The APPROXIMATELY EQUIVARIANT model tracks the NON-EQUIVARIANT model in CRPS. The two models separate sharply on the spread-to-skill ratio. The NON-EQUIVARIANT model is severely under-dispersed (over-confident): its SSR collapses to ~ 0.5 across the forecast on geopotential. The APPROXIMATELY EQUIVARIANT model stays much closer to the calibrated regime of SSR≈ 1. On temperature, where both under-disperse, APPROXIMATE EQUIVARIANCE's lowest SSR sits at ~ 0.77 versus NON-EQUIVARIANT's ~ 0.55. On ensemble-mean RMSE, NON-EQUIVARIANT is marginally better (~ 0.5 K on temperature, at most $\sim 30\,\mathrm{m}^2/\mathrm{s}^2$ on geopotential; the gap closes at long lead).

Deterministic skill and spectral fidelity. Figure 8.3 extends the comparison beyond the probabilistic scores, scoring each CRPS-finetuned model both as a single ensemble member and as the ensemble mean. On the deterministic metrics the picture of Section 8.3 is unchanged: NON-EQUIVARIANT remains ahead within both estimator classes, and for both models the ensemble mean clearly beats the single member. Notably, the APPROXIMATELY EQUIVARIANT ensemble mean recovers nearly the entire single-member deficit on the representative variable: The run sits close to or on the NON-EQUIVARIANT single-member baseline (representative variable in Figure 8.3, top).

On spectral fidelity, the two estimators disagree about which model is better. On single members, the APPROXIMATELY EQUIVARIANT model has the better spectral fidelity. The kinetic-energy retention (Figure 8.3) is nearly indistinguishable at 12 h, but at 72 h the single APPROXIMATELY EQUIVARIANT member retains the most small-scale energy of all four lines. Under the ensemble mean the ranking inverts: the NON-EQUIVARIANT ensemble mean has the better spectral fidelity.

Insights

Answering Question 2: APPROXIMATE EQUIVARIANCE pays a modest price on the deterministic skill, but it improves significantly on calibration, mirroring the deterministic-vs-spectral split of Section 8.3. Matched CRPS with substantially better spread-to-skill (at a quarter of the trainable parameters) suggests the equivariant inductive bias yields uncertainty estimates that are better matched to the model's true error.

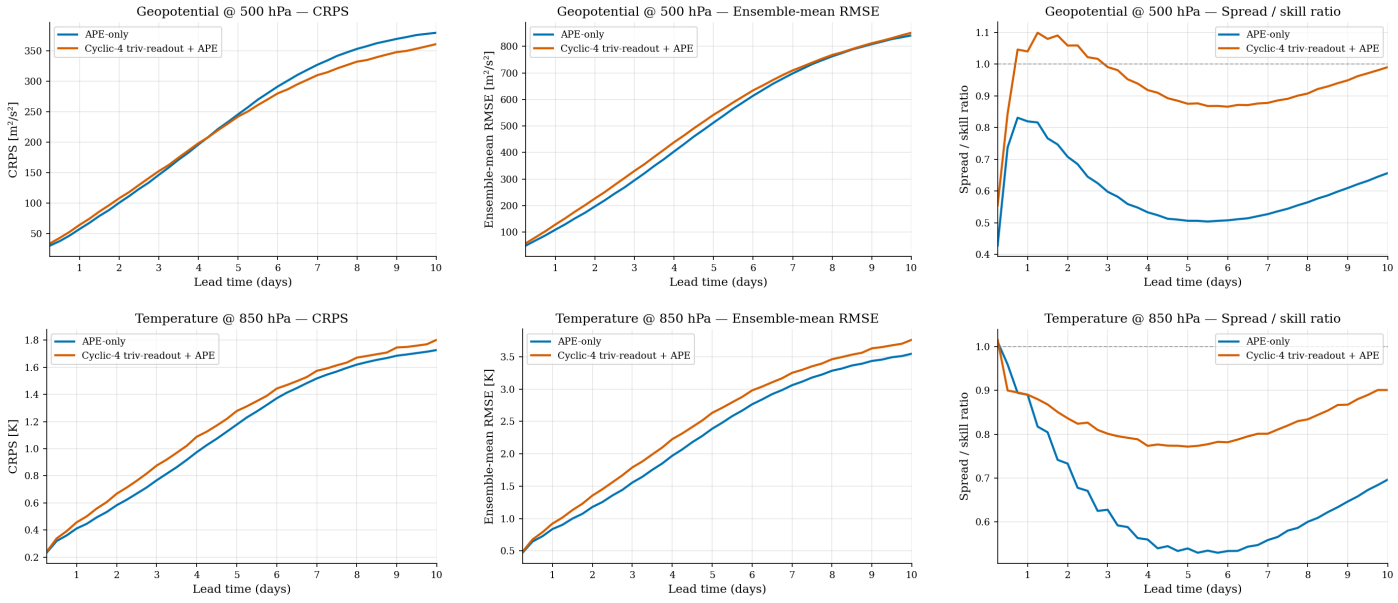


Figure 8.2: Probabilistic evaluation as a function of lead time for the weight-sharing comparison. **Top row:** geopotential at 500 hPa. **Bottom row:** temperature at 850 hPa. **Left column:** CRPS (lower is better). **Middle column:** ensemble-mean RMSE (lower is better). **Right column:** spread-to-skill ratio (calibrated at 1; below 1 is over-confident). APE-ONLY (blue) vs. CYCLIC-4 TRIV-READOUT + APE (orange).

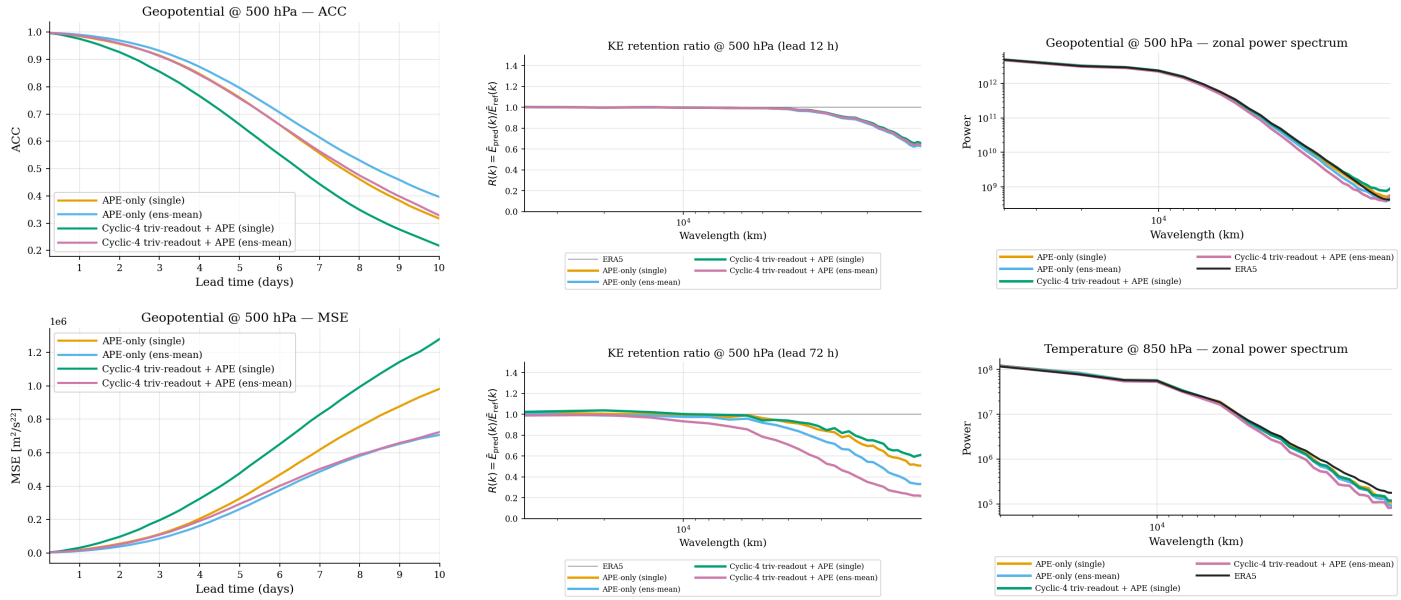


Figure 8.3: CRPS-finetuned models scored as a single member versus the ensemble mean (10 members), four lines per panel: {APE-only, Cyclic-4 triv-readout + APE} × {single, ens-mean}. **Left column:** deterministic skill of geopotential at 500 hPa (ACC top, MSE bottom). **Middle column:** kinetic-energy retention ratio at 500 hPa (12 h lead top, 72 h lead bottom), $R{=}1$ matches ERA5. **Right column:** zonal power spectra at 72 h lead (geopotential at 500 hPa top, temperature at 850 hPa bottom), ERA5 (black) is the reference.

The single-member vs. ensemble-mean contrast shows what this difference in spread does to the deterministic and spectral metrics. Averaging helps the squared-error metrics of both models by construction: the ensemble mean minimises squared error, while a single member carries the small-scale variance that squared-error metrics penalise. This is why ten members at 45M trainable parameters buy back the deterministic gap to a single forecast of the four-times-larger model. The same averaging destroys spectral fidelity: diverged members place small-scale features at different locations, so averaging cancels their small-scale power, and the amount cancelled grows with the spread of the ensemble. This explains why the non-equivariant ensemble mean has the better spectral fidelity even though its single members do not: its under-dispersed members stay close together and their small scales survive averaging, whereas the well-calibrated approximately equivariant members diverge as fast as the true error grows and their small scales cancel.

This yields a practical rule for evaluating ensemble forecasters: the estimator must match the metric. Squared-error skill is a property of the ensemble mean, while spectral and physical fidelity are properties of the individual members. Ranking models by ensemble-mean spectra inverts the true ordering and rewards under-dispersion, penalising exactly the calibration that the probabilistic scores are designed to measure.

Chapter 9

Discussion

This thesis set out to isolate the effect of geometric inductive biases on weather-model skill and physical fidelity, and to interpret what models learn across varying levels of equivariance when given the freedom to break symmetry. This chapter draws the individual findings together, states the boundaries of the claims, and pairs each boundary with the research direction it motivates.

What geometry does

The grid experiment (Experiment 1) showed that a model's internal processor grid should operate on cells of equal physical area. The native equiangular lat-lon grid does not: its cells shrink towards the poles, so a local operation covers a different physical area at every latitude. Re-projecting onto the equal-area HEALPix grid removes this distortion and improves every headline variable within the 72-hour scope.

The vector-loss experiment (Experiment 2) showed that no single objective wins on every axis: there is a trade-off between deterministic skill and spectral fidelity. Component-wise MSE is best on its own deterministic target but blurs the most and is the only objective that systematically underestimates wind speed; objectives that drop the component-wise term entirely, including the FastNet formulation, lose deterministic skill outright. Scoring wind as a vector *on top of* the component-wise term (`mse+mag+angle`) emerges as the best compromise: near-MSE deterministic skill, stable small-scale energy retention, and the most balanced speed bias. Since probabilistic weather models are typically fine-tuned from a deterministic pretraining stage, a deterministic wind objective that does not bake in a systematic speed bias is of direct practical value.

The experiment on 2D vs. 3D modelling (Experiment 3) showed that a model's representation of space, coordinates and vectors alike, should match the manifold it operates on, not the space the data is embedded in. Our model acts locally, and since modelling in 2D beats 3D, those local regions are best represented as tangent planes to the sphere. This is a choice many methods make implicitly but which, to our knowledge, had not been tested in isolation. The result runs against the physically motivated intuition that a 3D representation should be the more faithful one. We read it as a genuine negative finding rather than proof that 3D geometry cannot work: wind is physically a tangent vector on a sphere, and global 3D rotational equivariance is in principle the more faithful symmetry group. In practice, however, global operations are computationally prohibitive at the $0.25°$ resolution of an operational forecasting grid (roughly one million nodes), which is precisely why we restrict ourselves to 2D local tangent-plane modelling. A globally- or gauge-equivariant 3D formulation could plausibly close this gap; we leave this for future work.

The positional-encoding frequency scheme experiment (Experiment 4) showed that the best scheme depends on what the encoding is asked to do. RoPE encodes relative distances in the data and therefore profits from data-driven adaptation: learnable frequencies with a data-informed initialisation reach the highest skill. APE encodes absolute position, a task closer to memorising a code per grid location, and is best served by fixed frequencies that cover the space evenly. The choice further interacts with the symmetry group: under a Cyclic-4 rotation group the anisotropic log-spaced layout is no longer rotation-consistent, so its edge appears only in the group-free setting.

Experiment 5 showed that geometric symmetry is valuable as an inductive bias, but not as a hard constraint. Two results back this. First, adding translation equivariance helps unconditionally: the RoPE-based configurations, with or without symmetry breaking, improve over the APE-only baseline. Second, building in the group structure while allowing the model to depart from it beats enforcing it strictly: the trivial-readout variants outperform their strictly equivariant counterparts. Used in this soft form, group-equivariant weight sharing under the planar Cyclic-4 group buys effective capacity: the Cyclic-4 trivial-readout configuration (with or without APE) reaches nearly the skill of the non-equivariant APE-only baseline while using roughly a quarter of the trainable parameters. It is the most KE-spectrally faithful configuration of the family. The 2D Cyclic-4 family also beats its 3D tetrahedral counterpart configuration-for-configuration, consistent with Experiment 3.

Symmetry-breaking features: what they learn, and whether they help

The interpretability experiment (Experiment 6) showed that the (approximately) equivariant models, when handed free channels to break symmetry, fill it with recognisable physics, whereas the non-equivariant baseline does not. The learned channels recovered physical structure in almost every equivariant configuration tested: a hemisphere-antisymmetric channel resembling the Coriolis parameter, latitude-banded structure consistent with insolation, mid-latitude banding we read as jet-stream or general-circulation structure, and, whenever static geographic inputs were withheld, geography itself (e.g. orography) learned directly in the channels. Only the non-equivariant APE baseline showed none of this, instead reproducing its coordinate grid. This contrast is informative in itself: recognisable physics emerges only where a model makes the learned bias the sole pathway for absolute position, so the channels are interpretable precisely because the rest of the model cannot see where it is. This is, to our knowledge, the first qualitative account of which symmetry-breaking signals a weather model injects when given a controlled channel to do so.

Under deterministic training, the learned features improved short-range skill in most configurations, but the gain came from smoothing forecasts toward the conditional mean: spectral fidelity degraded consistently wherever deterministic skill improved. Static symmetry-breaking inputs improved skill when-

ever the model was non-strictly roto-translation-equivariant.

Two questions remain for future work. The first is evaluating these features under a probabilistic objective, where the incentive to smooth disappears. The second is why static features destabilise the strictly roto-translation-equivariant models. One hypothesis is that the Cyclic-4 group action exchanges the latitude and longitude axes of the shared filters. As Experiment 6 showed, the symmetry-breaking structure the atmosphere actually carries is strongly latitude-banded (Coriolis, insolation, the jets), varying with latitude but nearly uniform in longitude. Such patterns are maximally asymmetric under a latitude-longitude swap, so weight sharing across the rotated frames mixes them, something translation equivariance does not do.

The value and limits of approximate equivariance

The stress test of inductive bias showed that scale buys deterministic skill, whereas the inductive bias of approximate equivariance is what buys physical consistency, calibration, and parameter efficiency. Three results back this. On deterministic skill the non-equivariant baseline leads at matched total parameters, but its lead shrinks with capacity, marking it a capacity effect. The spectral gap, by contrast, never closes: the approximately equivariant model is more spectrally faithful at every size on both parameter axes, so fidelity is a consequence of the inductive bias rather than a deficit scale repairs. The same bias also acts as a parameter-efficiency multiplier at every scale, reaching a given skill with a fraction of the trainable parameters. This split survives CRPS fine-tuning: the approximately equivariant model nearly matches the baseline's probabilistic skill at a fraction of the trainable parameters while being markedly better calibrated.

Two limitations bound these claims. First, the scaling ladder spans only three widths: it supports the qualitative claim that the deterministic gap narrows with total parameters while the spectral gap does not within the tested range, but cannot support extrapolation, and a crossover at larger capacity can be neither claimed nor excluded. Second, the probabilistic comparison is made at matched total but not matched trainable parameters, so the calibration and efficiency advantages cannot be cleanly separated from the difference in trainable capacity. Extending both the width ladder and a trainable-matched comparison is left for future work.

Limitations

First, all geometry ablations were run at $5.6°$ resolution with deterministic, single-step MSE training and no rollout fine-tuning. This was a deliberate trade: many like-for-like comparisons under identical conditions rather than a handful of expensive high-resolution runs. This means, however, that the results are established in a coarse data-regime where autoregressive errors compound quickly, which is why the primary analysis is restricted to 72-hour lead times. The final evaluation partially relaxes this: the CRPS fine-tuning experiment shows the central equivariance findings survive a probabilistic objective, but rollout fine-tuning remains untested. The transfer of every individual geometric finding to operational resolutions and objectives therefore remains an assumption, not a result. The corresponding directions follow in order of importance. The most immediate is resolution: confirming the headline findings at $1.5°$, and ultimately at the $0.25°$ operational scale. Alongside resolution, rollout fine-tuning would replace the transfer assumption above with a result. Resolution is also what gates external comparison: this thesis compares only within its own model cohort because, as argued in Chapter 8, regridding models such as GraphCast or GenCast to $5.6°$ would measure the resolution gap rather than the modelling choices under study. Until the architecture is validated at operational resolution, no claim about weather in this thesis is a claim about the operational state of the art.

Second, the experiments cover six headline variables and treat the atmosphere as a stack of horizontal fields. Future iterations should broaden the variable set, both for predictive capability and to include explicitly static symmetry-breaking predictors such as incident solar radiation, which would provide a sharper stress test of the equivariance and weight-sharing findings.

Finally, Noether's theorem motivates *temporal* equivariance as the natural sequel to the spatial symmetries studied here: enforcing (approximate) equivariance across time steps would make conservation of quantities such as energy and momentum a structural property of the network rather than a learned approximation. This is most relevant for climate-length rollouts, extremes, and precipitation, where accumulated conservation violations are most damaging. A promising avenue is variational flow matching as proposed by Eijkelboom et al. [2024], where we can put constraint on the models prediction, while the objective stays probabilistic. Examples of these constraints would be the non-negativity of precipitation (via ReLU-based activations) or use the physics-aligned components as in Sha et al. [2025]. This would be particularly relevant for long-range climate rollouts, predicting extremes and modelling precipitation, where even minor violations of conservation laws lead to catastrophic divergence.

Chapter 10

Conclusion

This thesis asked whether the geometry a weather model imposes on its data measurably affects forecast skill and physical fidelity, independently of the backbone architecture those choices are usually bundled with. To make that question answerable, we proposed Platonic-Erwin, an architecture combining the linear-scaling hierarchical ball-tree attention of Erwin with the group-equivariant weight sharing of Platonic Transformers, and validated it on a large-scale protein molecular-dynamics benchmark, where it outperformed prior equivariant and non-equivariant methods.

With this backbone fixed, we ran controlled geometric ablations on $5.6°$ ERA5 forecasting, isolating grid topology, vector-loss formulation, 2D-versus-3D representation, and positional-encoding design one at a time against otherwise identical models. Four lessons follow. A model's internal processor grid should divide the sphere into cells of equal physical area, so that a local operation covers the same amount of atmosphere at every latitude; the equiangular lat-lon grid does not, and re-projecting onto HEALPix improved every headline variable. No single wind objective wins on every axis: deterministic skill and spectral fidelity trade off against each other, and scoring wind as a vector *on top of* the component-wise term (`mse+mag+angle`) is the best compromise. A model's representation of space should match the manifold it operates on, not the space that data is embedded in: local *2D* modelling beats the *3D* representation in every configuration family. Lastly, the best positional-frequency scheme depends on what the encoding is asked to do: RoPE encodes relative distance in the data and profits from data-driven adaptation, whereas APE encodes absolute position, a task closer to memorising a code per grid location, and is best served by fixed frequencies that cover the space evenly.

Ablations on weight sharing and equivariance showed that geometric symmetry is valuable as an inductive bias, but not as a hard constraint. A flexible departure from exact symmetry beat enforcing it strictly, and in this approximate group-equivariant weight sharing acts as a parameter-efficiency multiplier: the 2D Cyclic-4 trivial-readout configuration reached nearly the skill of a non-equivariant baseline while using roughly a quarter of the trainable parameters, and remained the most faithful configuration of its family on the kinetic-energy spectra.

Beyond the controlled comparisons, we asked what a model injects when handed a free channel to break symmetry. Approximately (equivariant) models fill it with recognisable physics, whereas the non-equivariant baseline does not: the learned channels recovered Coriolis-like hemispheric antisymmetry, insolation banding, jet-stream banding, and, wherever static geographic inputs were withheld, geography itself. Only the non-equivariant baseline showed none of this, reproducing its coordinate grid instead. Under deterministic training the learned features improved short-range skill in most configurations, but the gain came from smoothing forecasts toward the conditional mean. Static symmetry-breaking inputs helped whenever the model was non-strictly roto-translation-equivariant.

The stress test of inductive bias separated two kinds of advantage between the non-equivariant and approximately equivariant models. Scale buys deterministic skill: the baseline's lead there behaves like a capacity effect and narrows with parameter count. It does not buy the rest. Spectral fidelity, calibration, and parameter efficiency are consequences of the inductive bias itself rather than deficits that scale repairs.

This answers the problem that motivated the thesis. The failure modes we set out from, blurred extremes and conservation drift, are failures of physical fidelity, and physical fidelity is exactly what scale did not buy: across the sizes we tested the spectral gap never closed. Physical consistency is something a model can be designed to have rather than scaled into, and that is what makes it democratising: the faithful configuration is not necessarily the largest one, so trustworthy forecasting doesn't need to depend on the compute budgets concentrated in the global north. By isolating each geometric choice from the architecture it is usually bundled with, we hope to spare the community from re-litigating confounded comparisons, and to lower the bar for others to build on these findings rather than rediscover them. In that spirit, we hope this encourages a habit of making weather models smarter before making them bigger, and that the mathematics of symmetry finds a lasting place in machine-learned weather prediction.

Acknowledgments

First and foremost, I would like to thank David Wessels for his excellent supervision throughout this project. I thank Erik Bekkers for his valuable input and the insightful discussions. I am further grateful to Maxim Zhdanov for his advice on efficiency and experimental design, and to Wessel Bruinsma for his guidance on weather modelling. Finally, I thank Daan Heijke for being a great sparring partner throughout, and for his help in shaping the narrative of this thesis.

Bibliography

Ferran Alet, Ilan Price, Andrew El-Kadi, Dominic Masters, Stratis Markou, Tom R Andersson, Jacklynn Stott, Remi Lam, Matthew Willson, Alvaro Sanchez-Gonzalez, et al. Skillful joint probabilistic weather forecasting from marginals. *arXiv preprint arXiv:2506.10772*, 2025.

Matthew Ashman, Cristiana Diaconu, Eric Langezaal, Adrian Weller, and Richard E Turner. Gridded transformer neural processes for large unstructured spatio-temporal data. *arXiv preprint arXiv:2410.06731*, 2024a.

Matthew Ashman, Cristiana Diaconu, Adrian Weller, Wessel Bruinsma, and Richard E Turner. Approximately equivariant neural processes. *Advances in neural information processing systems*, 37:97088–97123, 2024b.

Francesco Ballerin, Nello Blaser, and Erlend Grong. So (3)-equivariant neural networks for learning vector fields on spheres. *arXiv preprint arXiv:2503.09456*, 2025.

Kaifeng Bi, Lingxi Xie, Hengheng Zhang, Xin Chen, Xiaotao Gu, and Qi Tian. Pangu-weather: A 3d high-resolution model for fast and accurate global weather forecast. *arXiv preprint arXiv:2211.02556*, 2022.

Cristian Bodnar, Wessel P Bruinsma, Ana Lucic, Megan Stanley, Anna Allen, Johannes Brandstetter, Patrick Garvan, Maik Riechert, Jonathan A Weyn, Haiyu Dong, et al. A foundation model for the earth system. *Nature*, pages 1–8, 2025.

Boris Bonev, Thorsten Kurth, Christian Hundt, Jaideep Pathak, Maximilian Baust, Karthik Kashinath, and Anima Anandkumar. Spherical fourier neural operators: Learning stable dynamics on the sphere. In *International conference on machine learning*, pages 2806–2823. PMLR, 2023.

Boris Bonev, Thorsten Kurth, Ankur Mahesh, Mauro Bisson, Jean Kossaifi, Karthik Kashinath, Anima Anandkumar, William D Collins, Michael S Pritchard, and Alexander Keller. Fourcastnet 3: A geometric approach to probabilistic machine-learning weather forecasting at scale. *arXiv preprint arXiv:2507.12144*, 2025.

Johann Brehmer, Sönke Behrends, Pim De Haan, and Taco Cohen. Does equivariance matter at scale? *arXiv preprint arXiv:2410.23179*, 2024.

Salva Rühling Cachay, Duncan Watson-Parris, and Rose Yu. U-cast: A surprisingly simple and efficient frontier probabilistic ai weather forecaster. *arXiv preprint arXiv:2604.09041*, 2026.

Taco Cohen and Max Welling. Group equivariant convolutional networks. In *International conference on machine learning*, pages 2990–2999. PMLR, 2016.

Guillaume Couairon, Renu Singh, Anastase Charantonis, Christian Lessig, and Claire Monteleoni. Archesweather & archesweathergen: a deterministic and generative model for efficient ml weather forecasting. *arXiv preprint arXiv:2412.12971*, 2024.

Timothée Darcet, Maxime Oquab, Julien Mairal, and Piotr Bojanowski. Vision transformers need registers. *arXiv preprint arXiv:2309.16588*, 2023.

Cristiana Diaconu, Jonas Scholz, Aliaksandra Shysheya, Stratis Markou, Payel Mukhopadhyay, Miles Cranmer, and Richard E Turner. Otter weather: Skillful and computationally efficient medium-range weather forecasting. *arXiv preprint arXiv:2606.26421*, 2026.

Tom Dunstan, Oliver Strickson, Thusal Bennett, Jack Bowyer, Matthew Burnand, James Chappell, Alejandro Coca-Castro, Kirstine Ida Dale, Eric G Daub, Noushin Eftekhari, et al. Fastnet: Improving the physical consistency of machine-learning weather prediction models through loss function design. *arXiv preprint arXiv:2509.17601*, 2025.

Floor Eijkelboom, Grigory Bartosh, Christian A Naesseth, Max Welling, and Jan-Willem van de Meent. Variational flow matching for graph generation. *Advances in Neural Information Processing Systems*, 37:11735–11764, 2024.

Piyush Garg, Diana R Gergel, Andrew E Shao, and Galen J Yacalis. The recipe matters more than the kitchen: Mathematical foundations of the ai weather prediction pipeline. *arXiv preprint arXiv:2604.01215*, 2026.

Krzysztof M Gorski, Eric Hivon, Anthony J Banday, Benjamin D Wandelt, Frode K Hansen, Mstvos Reinecke, and Matthia Bartelmann. Healpix: A framework for high-resolution discretization and fast analysis of data distributed on the sphere. *The Astrophysical Journal*, 622(2):759–771, 2005.

Jiaqi Han, Wenbing Huang, Tingyang Xu, and Yu Rong. Equivariant graph hierarchy-based neural networks. *Advances in Neural Information Processing Systems*, 35:9176–9187, 2022.

Alex Henry, Prudhvi Raj Dachapally, Shubham Shantaram Pawar, and Yuxuan Chen. Query-key normalization for transformers. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 4246–4253, 2020.

Hans Hersbach, Bill Bell, Paul Berrisford, Shoji Hirahara, András Horányi, Joaquín Muñoz-Sabater, Julien Nicolas, Carole Peubey, Raluca Radu, Dinand Schepers, et al. The era5 global reanalysis. *Quarterly journal of the royal meteorological society*, 146(730):1999–2049, 2020.

Mohammad Mohaiminul Islam, Rishabh Anand, David R Wessels, Friso de Kruiff, Thijs P Kuipers, Rex Ying, Clara I Sánchez, Sharvaree Vadgama, Georg Bökman, and Erik J Bekkers. Platonic transformers: A solid choice for equivariance. *arXiv preprint arXiv:2510.03511*, 2025.

Matthias Karlbauer, Danielle C Maddix, Abdul Fatir Ansari, Boran Han, Gaurav Gupta, Yuyang Wang, Andrew Stuart, and Michael W Mahoney. Comparing and contrasting deep learning weather prediction backbones on navier-stokes and atmospheric dynamics. *arXiv preprint arXiv:2407.14129*, 2024.

Emma Kasteleyn, Timo Maier, Axel Lauer, Veronika Eyring, Pierre Gentine, and Ana Lucic. Physmetrics. weather: An evaluation framework for physical consistency in ml weather models. *arXiv preprint arXiv:2606.10642*, 2026.

Dmitrii Kochkov, Janni Yuval, Ian Langmore, Peter Norgaard, Jamie Smith, Griffin Mooers, Milan Klöwer, James Lottes, Stephan Rasp, Peter Düben, et al. Neural general circulation models for weather and climate. *Nature*, 632(8027):1060–1066, 2024.

Remi Lam, Alvaro Sanchez-Gonzalez, Matthew Willson, Peter Wirnsberger, Meire Fortunato, Ferran Alet, Suman Ravuri, Timo Ewalds, Zach Eaton-Rosen, Weihua Hu, et al. Learning skillful medium-range global weather forecasting. *Science*, 382 (6677):1416–1421, 2023.

Simon Lang, Mihai Alexe, Matthew Chantry, Jesper Dramsch, Florian Pinault, Baudouin Raoult, Mariana CA Clare, Christian Lessig, Michael Maier-Gerber, Linus Magnusson, et al. Aifs–ecmwf's data-driven forecasting system. *arXiv preprint arXiv:2406.01465*, 2024.

Simon Lang, Mihai Alexe, Mariana CA Clare, Christopher Roberts, Rilwan Adewoyin, Zied Ben Bouallègue, Matthew Chantry, Jesper Dramsch, Peter D Dueben, Sara Hahner, et al. Aifs-crps: ensemble forecasting using a model trained with a loss function based on the continuous ranked probability score. *npj Artificial Intelligence*, 2(1):18, 2026.

Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

Hampus Linander, Christoffer Petersson, Daniel Persson, and Jan E Gerken. Pear: Equal area weather forecasting on the sphere. *arXiv preprint arXiv:2505.17720*, 2025.

Peiyuan Liu, Tian Zhou, Liang Sun, and Rong Jin. Mitigating time discretization challenges with weatherode: A sandwich physics-driven neural ode for weather forecasting. *arXiv preprint arXiv:2410.06560*, 2024.

Yang Liu, Zinan Zheng, Jiashun Cheng, Fugee Tsung, Deli Zhao, Yu Rong, and Jia Li. Cirt: Global subseasonal-to-seasonal forecasting with geometry-inspired transformer. *arXiv preprint arXiv:2502.19750*, 2025a.

Yang Liu, Zinan Zheng, Yu Rong, Deli Zhao, Hong Cheng, and Jia Li. Equivariant and invariant message passing for global subseasonal-to-seasonal forecasting. In *Proceedings of the 31st ACM SIGKDD Conference on Knowledge Discovery and Data Mining V. 2*, pages 1879–1890, 2025b.

Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 10012–10022, 2021.

V. Masson-Delmotte, P. Zhai, A. Pirani, et al., editors. *Climate Change 2021: The Physical Science Basis. Contribution of Working Group I to the Sixth Assessment Report of the Intergovernmental Panel on Climate Change*. Cambridge University Press, Cambridge, United Kingdom and New York, NY, USA, 2021. doi: 10.1017/9781009157896.

Qness Ndlovu. Closing africa's early warning gap: Ai weather forecasting for disaster prevention. *arXiv preprint arXiv:2602.17726*, 2026.

E Noether. Invariante variationsprobleme, nachrichten von der gesellschaft der wissenschaften zu göttingen, mathematisch-physikalische klasse 1918 (1918) 235–257. *URL http://eudml.org/doc/59024*, 1918.

Harry Nyquist. Certain topics in telegraph transmission theory. *Transactions of the American Institute of Electrical Engineers*, 47(2):617–644, 1928.

Jeremy Ocampo, Matthew A Price, and Jason D McEwen. Scalable and equivariant spherical cnns by discrete-continuous (disco) convolutions. *arXiv preprint arXiv:2209.13603*, 2022.

Ariel Ortiz-Bobea. Climate, agriculture and food. *arXiv preprint arXiv:2105.12044*, 2021. doi: 10.48550/arXiv.2105.12044. Chapter prepared for the Handbook of Agricultural Economics.

Stephan Rasp, Stephan Hoyer, Alexander Merose, Ian Langmore, Peter Battaglia, Tyler Russell, Alvaro Sanchez-Gonzalez, Vivian Yang, Rob Carver, Shreya Agrawal, et al. Weatherbench 2: A benchmark for the next generation of data-driven global weather models. *Journal of Advances in Modeling Earth Systems*, 16(6):e2023MS004019, 2024.

Sean Seyler and Oliver Beckstein. Molecular dynamics trajectory for benchmarking mdanalysis. *URL: https://figshare. com/articles/Molecular dynamics trajectory for benchmarking MDAnalysis/5108170, doi*, 10(m9):7, 2017.

Yingkai Sha, John S Schreck, William Chapman, and David John Gagne. Improving ai weather prediction models using global mass and energy conservation schemes. *Journal of Advances in Modeling Earth Systems*, 17(11):e2025MS005138, 2025.

Noam Shazeer. Glu variants improve transformer. *arXiv preprint arXiv:2002.05202*, 2020.

Philip Stier. Probabilistic ai for environmental science: Prof philip stier, 2025. URL https://www.youtube.com/watch?v=GQpCBVV48F8&t=1121s. Available at https://www.youtube.com/watch?v=GQpCBVV48F8, timestamp 18:41.

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

Christopher Subich, Syed Zahid Husain, Leo Separovic, and Jing Yang. Fixing the double penalty in data-driven weather forecasting through a modified spherical harmonic loss function. *arXiv preprint arXiv:2501.19374*, 2025.

Waldo R Tobler. A computer movie simulating urban growth in the detroit region. *Economic geography*, 46(sup1):234–240, 1970.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*, 2023.

United Nations. Early warnings for all, 2022. URL https://earlywarningsforall.org/site/early-warnings-all. Accessed: 2026-03-30.

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

Feng Wang, Sucheng Ren, Tiezheng Zhang, Predrag Neskovic, Anand Bhattad, Cihang Xie, and Alan Yuille. Vit-5: Vision transformers for the mid-2020s. *arXiv preprint arXiv:2602.08071*, 2026.

David R Wessels, David M Knigge, Samuele Papa, Riccardo Valperga, Sharvaree Vadgama, Efstratios Gavves, and Erik J Bekkers. Grounding continuous representations in geometry: Equivariant neural fields. *arXiv preprint arXiv:2406.05753*, 2024.

Michaël Zamo and Philippe Naveau. Estimation of the continuous ranked probability score with limited information and applications to ensemble weather forecasts. *Mathematical Geosciences*, 50(2):209–234, 2018.

Biao Zhang and Rico Sennrich. Root mean square layer normalization. *Advances in neural information processing systems*, 32, 2019.

Maksim Zhdanov, Max Welling, and Jan-Willem van de Meent. Erwin: A tree-based hierarchical transformer for large-scale physical systems. *arXiv preprint arXiv:2502.17019*, 2025.

Maksim Zhdanov, Ana Lucic, Max Welling, and Jan-Willem van de Meent. (sparse) attention to the details: Preserving spectral fidelity in ml-based weather forecasting models. *arXiv preprint arXiv:2604.16429*, 2026.

Shu Zhong, Mingyu Xu, Tenglong Ao, and Guang Shi. Understanding transformer from the perspective of associative memory. *arXiv preprint arXiv:2505.19488*, 2025.

Appendix A

Architecture implementation

A.1 Set-Axis Ball-Tree Construction via PCA Canonicalization

A.1.1 Algorithm

Let $P = (p_1, \dots, p_N) \in \mathbb{R}^{d\times N}$ denote the points of one batch, with centroid $\bar{p} = \frac{1}{N}\sum_i p_i$ and centered covariance

$$\Sigma(P) \;=\; \sum_{i=1}^{N}(p_i - \bar{p})(p_i - \bar{p})^\top \;\in\; \mathbb{R}^{d\times d}. \tag{A.1}$$

The ball tree is built by the following pipeline, applied independently per batch and in parallel across batches:

1. **Principal direction.** Compute the leading eigenvector $e_1(P) \in \mathbb{R}^d$ of $\Sigma(P)$. For $d = 2$ we use the closed-form solution; for $d = 3$ we run power iteration initialized with the column of $\Sigma(P)$ of largest norm.

2. **Sign disambiguation.** An eigenvector is only defined up to sign, so we fix it via a data-dependent rule: replace $e_1 \leftarrow \mathrm{sign}\big(s(P, e_1)\big) \cdot e_1$, where

$$s(P, d) \;=\; \sum_i \big\langle p_i - \bar{p},\, d\big\rangle^3 \tag{A.2}$$

   is the (unnormalized) directional skewness along d. If the skewness vanishes, the 5th central moment $\sum_i \langle p_i - \bar{p}, d\rangle^5$ is used instead.

3. **Frame completion.** For $d = 2$ the frame is already complete: requiring $R \in \mathrm{SO}(2)$ fixes $e_2 = (-e_{1,y},\, e_{1,x})$, the 90° counterclockwise rotation of e_1, with no additional sign choice. For $d = 3$ a residual rotation about e_1 remains, which we fix using the point furthest from the principal axis: choose $e_2(P)$ as the normalized component orthogonal to e_1 of the offset $p_i - \bar{p}$ that maximizes $\|p_i - \bar{p} - \langle p_i - \bar{p},\, e_1\rangle e_1\|$. We then sign-disambiguate e_2 with the same rule (A.2) and set $e_3 = e_1 \times e_2$, closing a right-handed orthonormal frame.

4. **Rotation matrix.** Stack the orthonormal frame into

$$R(P) \;=\; \begin{bmatrix} e_1(P)^\top \\ e_2(P)^\top \\ e_3(P)^\top \end{bmatrix} \;\in\; \mathrm{SO}(d), \tag{A.3}$$

   which maps the data-dependent frame onto the standard basis: $e_1 \mapsto x$, $e_2 \mapsto y$ and, for $d = 3$, $e_3 \mapsto z$.

5. **Canonicalize and build.** Form $\tilde{P} := R(P)\, P$ (i.e., $\tilde{p}_i = R(P)\, p_i$). On $\tilde{P}$, run the standard complete-binary ball tree: recursively (a) pick the coordinate axis of maximal spread, $\arg\max_j \big(\max_i \tilde{p}_{i,j} - \min_i \tilde{p}_{i,j}\big)$, and (b) partition at the median along that axis. Leaves hold 1 or 2 points, with a mask marking duplicated singletons.

6. **Return.** The tree structure $\mathcal{T}(\tilde{P})$ (index hierarchy), the duplicate mask, and the rotation $R(P)$. The rotation is reused to align the leaves before constructing the rotated (shifted-window) trees, and at inference time to map any vector-valued outputs back to the original frame.

A.1.2 Assumptions for rotation-invariant tree building

Since every step of the pipeline is built from rotation-equivariant quantities, invariance follows directly. This argument is valid under the following assumptions. Intuitively, each one rules out a point cloud that is "too symmetric" to single out a unique canonical frame.

(A1) The leading eigenvalue of $\Sigma(P)$ is simple, i.e. $\lambda_1 > \lambda_2$. In words: the cloud must have one clearly longest direction. A cigar-shaped cloud satisfies this; a perfect sphere, cube, or disc does not, because every direction (or every in-plane direction) is equally long and "the" principal axis is no longer well defined.

(A2) The directional skewness (A.2) is non-zero (or, as a fallback, the 5th moment is non-zero) along e_1 and, for $d = 3$, along e_2. In words: the cloud must be at least slightly lopsided along its axes, so that "front" and "back" can be told apart. A pear shape has an unambiguous heavy end; a perfectly mirror-symmetric shape (e.g. an ellipsoid) looks identical from both ends, and no moment-based rule can pick a sign.

(A3) For $d = 3$, the point furthest from the principal axis is unique. This point anchors the second axis e_2; if two points are exactly tied for furthest (as in any rotationally symmetric cloud, e.g. points on a ring around the axis), the choice between them is arbitrary.

(A4) The partition medians used by the tree builder are attained at unique points, i.e. no two points share the exact same coordinate value at a split. Ties occur naturally on regular lattices, where many points lie at identical coordinates.

The weather grids used in this thesis are worth noting explicitly, since regular grids are precisely the highly symmetric point clouds the assumptions rule out. In practice these caveats are immaterial for the weather experiments:the canonical frame on these grids is decided by tie-breaks and rounding noise, but the point set and its storage order never change, so the resulting tree is deterministic and is precomputed once, identical at training and inference. This is why the weather experiments use the standard axis-aligned construction, the grid itself already provides a fixed canonical frame (subsection 6.1.2). The assumptions only become relevant for dynamic point clouds (e.g. ProteinMD), where the positions change every step and no fixed frame is

available; there we deliberately use the standard axis-aligned tree during training, as a form of implicit data augmentation, and reserve the canonicalized variant for inference (subsection 4.1).

A.2 Register Tokens

Register tokens are additional, learnable tokens appended to the input sequence that do not correspond to any specific spatial patch or physical location. Instead, they provide a dedicated, persistent subspace for the model to store, process, and dump information throughout the network layers Darcet et al. [2023]. In our implementation, register tokens are instantiated as global, learnable embeddings (PERSISTENT REGISTERS. During the forward pass, these global tokens are dynamically broadcast to every spatial ball partition within the Multi-Head Self-Attention (MSA) blocks. Crucially, to preserve the strict group equivariance of the Platonic-Erwin architecture, register tokens are assigned a constant relative position of zero, corresponding to the center of the local ball partition. Following the local attention computation within each ball, the updated register embeddings are aggregated back into a unified global representation via a scatter-mean operation across the batch. This synchronized pooling ensures that the persistent global context is seamlessly updated and propagated through the network while rigorously maintaining its structural symmetries.

Appendix B

Weather-specific architecture implementation

B.1 Equivariant Cross-Attention Interpolation.

Data is transferred between the ERA5 latitude-longitude grid and the HEALPix mesh via group-equivariant cross-attention [Wessels et al., 2024, Zhdanov et al., 2026]. This can be seen as a continuous form of patchification. Unlike a convolutional or pooling encoder, which downsamples onto a fixed regular grid using a fixed-shape local window, our cross-attention transfers features between two arbitrary point sets (the equiangular lat–lon grid and the HEALPix mesh) via a learned, geometry-aware interpolation: each destination node attends over its k nearest source nodes, with relative position injected through (Platonic) RoPE. For each destination node i, the k nearest source nodes $\mathcal{N}_i$ are identified by cosine similarity on the unit sphere. Queries are initialised from the mean-pooled neighbourhood features; keys and values are projected from source features via `PlatonicLinear`:

$$\mathbf{q}_i = W_q\,\bar{\mathbf{x}}_{\mathcal{N}_i}, \quad \mathbf{k}_j = W_k\,\mathbf{x}_j, \quad \mathbf{v}_j = W_v\,\mathbf{x}_j, \qquad j \in \mathcal{N}_i, \tag{B.1}$$

$$\mathbf{o}_i = W_o \sum_{j\in\mathcal{N}_i} \mathrm{softmax}_j\left(\frac{\mathbf{q}_i^\top \mathbf{k}_j}{\sqrt{d}}\right)\mathbf{v}_j. \tag{B.2}$$

where d is the per-head feature dimension. (Platonic) RoPE is applied to both queries and keys using the absolute positions of the destination and source nodes respectively, making the attention score sensitive to their relative geometry while remaining equivariant under the destination stage's group action. The neighbourhood size k is chosen to ensure full spatial coverage; at the poles the HEALPix pixel density is significantly lower than the ERA5 grid, requiring a larger k to avoid gaps (see Figure B.1a). The same layer is used in both directions, grid $\to$ mesh at the encoder and mesh $\to$ grid at the decoder, with the roles of source and destination exchanged, so the re-projection introduces no non-equivariant operations into the pipeline.

B.2 Temporal embeddings

The layer encodes the elapsed time (in hours) using a bank of sine and cosine functions with learnable frequencies initialized on a log-spaced grid, spanning wavelengths from 1 hour to approximately one year (8766 hours). Writing the F frequencies as $\{\omega_f\}_{f=1}^{F}$ and the scalar time as τ, the raw features

$$\phi(\tau) = \big[\sin(\omega_1\tau), \ldots, \sin(\omega_F\tau),\ \cos(\omega_1\tau), \ldots, \cos(\omega_F\tau)\big] \in \mathbb{R}^{2F} \tag{B.3}$$

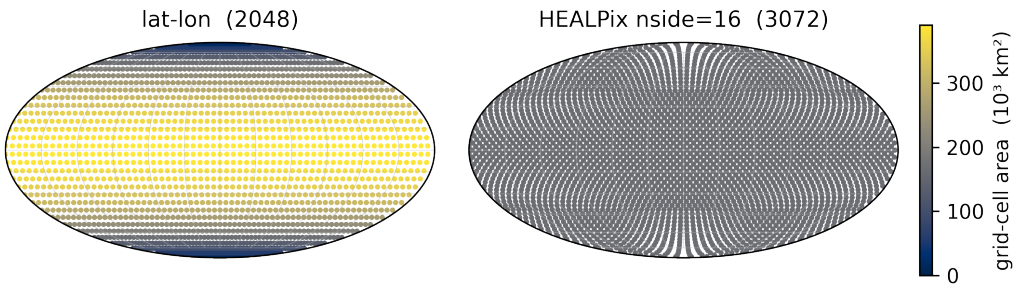
are linearly projected to a per-slot embedding $\mathbf{t} = W\phi(\tau) + b \in \mathbb{R}^{d/|G|}$. To preserve group equivariance, $\mathbf{t}$ is replicated identically across all $|G|$ group slots before being added to the feature tensor:

$$\mathbf{h} \leftarrow \mathbf{h} + \mathbf{1}_{|G|} \otimes \mathbf{t}. \tag{B.4}$$

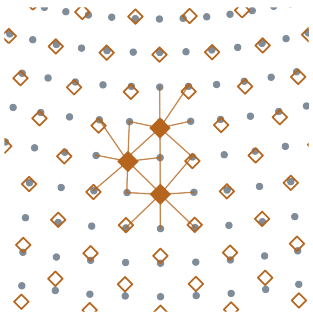
Since $\mathbf{t}$ is independent of the group-slot index, this addition commutes with the group action and leaves the equivariant structure intact.

B.3 Positional Information Across Call Sites (Weather)

ERA5 Positional information enters the Platonic-Erwin-Weather pipeline at several distinct call sites, and the way it is treated depends on two settings: the coordinate dimensionality (`spatial_dims` $\in \{2, 3\}$) and, in the 2D case, whether longitudinal wrap-around is enabled in the model setup. In 2D, the input position is the lat–lon pair $\mathbf{p} = [\phi, \lambda]^\top$ in degrees (with the WeatherBench longitude convention $\lambda \in [0, 360)$); in 3D, it is the Cartesian unit vector $\mathbf{p} = [x, y, z]^\top$ on the sphere (Appendix C). Table B.1 summarizes, for every call site, which position it consumes (absolute, relative-to-ball-centre, pairwise, or frequency-level) and how it behaves in each of the three regimes.



(a) Grid density comparison.



(b) Cross-attention neighborhood.

Figure B.1: Node density and cross-attention mechanics. **(a)** Node density of the regular latitude-longitude grid versus the HEALPix grid; near the poles the lat-lon grid concentrates far more nodes in a small area than HEALPix. **(b)** Each lat-lon query (light) attends over its k nearest HEALPix keys (dark).

Table B.1: How positional information is consumed at each call site of the Platonic-Erwin ERA5 pipeline, across the three coordinate regimes. "abs." denotes absolute position, "rel." position relative to a ball/group centre, "pair" pairwise distances, and "freq" the RoPE frequency vectors.

Call site	**Pos.**	**2D, no wrap**	**2D, wrap**	**3D (xyz)**
Absolute position embedding (APE)	abs.	sinusoid of raw (ϕ, λ); no seam fix	same (APE is not wrap-corrected)	sinusoid of (x, y, z)
Ball-tree + k-NN graph	abs.	Euclidean on (ϕ, λ) (seam at $0/360°$)	project to xyz, then build (contiguous)	raw xyz (already correct)
Embedding k-NN self-attention	abs.	flat lat–lon k-NN	xyz k-NN	xyz k-NN
Ball-attention RoPE	rel.	centre = mean, offset = subtract	centre = circular mean, offset = shortest-arc $\Delta\lambda$	centre = mean, offset = subtract
Ball-attention distance bias (opt.)	pair	`cdist` on raw (ϕ, λ)	xyz chordal `cdist`	xyz chordal `cdist`
Pooling / unpooling	rel.	centre = mean, offset = subtract	centre = circular mean, offset = shortest-arc $\Delta\lambda$	centre = mean, offset = subtract
Cross-attention (equiangular ↔ HEALPix)	abs.	flat lat–lon k-NN; RoPE$(Q \mid \mathbf{p}_q)$, RoPE$(K \mid \mathbf{p}_c)$	xyz k-NN; same RoPE	xyz k-NN; same RoPE
Inductive wrap RoPE	freq	off	longitude freqs snapped to $360°$ harmonics	n/a (no longitude axis)

B.3.1 Equivariant functional perturbations

Every hidden channel of the backbone is a stack of G group "slots" whose permutation under a group element is what makes the network equivariant; a perturbation that differed across slots would break this symmetry. We therefore treat the latent $\mathbf{z}$ as a trivial-representation (scalar) latent with no group axis, and before it enters a block we copy-lift it identically across the G slots, $\mathbb{R}^{d_z} \to \mathbb{R}^{G \times d_z} \to \mathbb{R}^{G \cdot d_z}$ with $d_z = 32$, then project it with an equivariant Platonic linear layer. Since the lifted input is constant across slots (an invariant), the gate bias it produces is also constant across slots: an invariant in maps to an invariant out, so the perturbed gate of Equation 8.1 remains group-equivariant.

B.4 The $\mathcal{C}_{n,z}$ Cyclic Group

For global weather modeling, we propose the $\mathcal{C}_{n,z}$ group, which consist of n discrete rotations around the z-axis. A group element $g \in \mathcal{C}_{n,z}$ is represented by the matrix:

$$\rho(g) = \begin{bmatrix} \cos(\theta) & -\sin(\theta) & 0 \\ \sin(\theta) & \cos(\theta) & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{B.5}$$

where $\theta = \frac{2\pi k}{n}$ for $k \in \{0, 1, \ldots, n-1\}$.

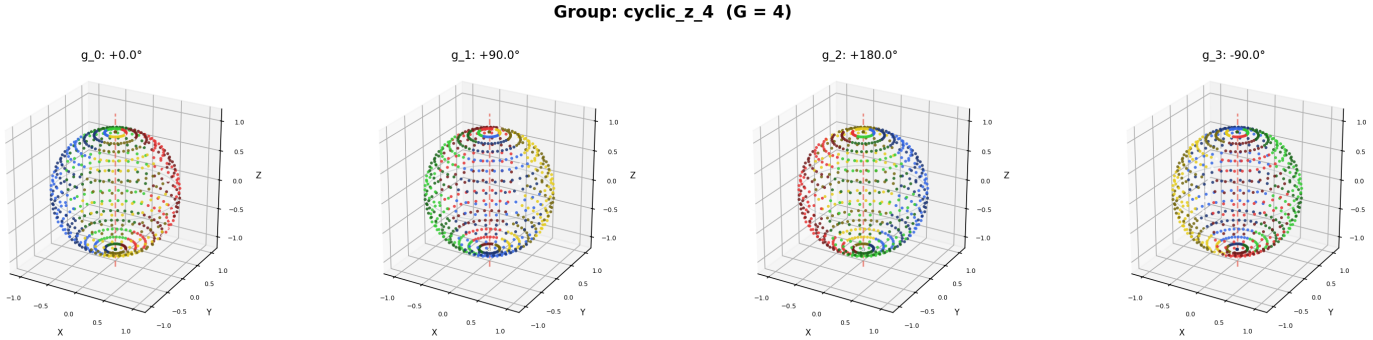


Figure B.2: Visualization of the longitudinal cyclic group $\mathcal{C}_{4,z}$: the four group elements correspond to successive $90°$ rotations about the vertical (z) axis, mapping the sphere onto itself.

Appendix C

Modeling weather data in 3D

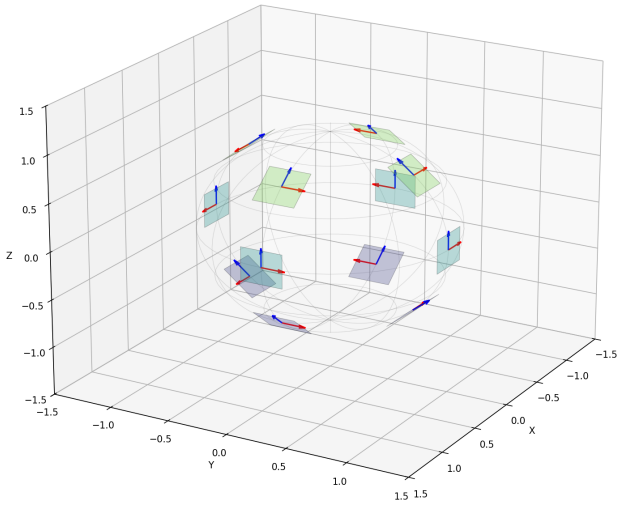


Figure C.1: Visualization of local tangent planes across the sphere. Each plane represents the local Euclidean space where eastward and northward wind components are defined.

Coordinate Transformation

To bypass the boundary artifacts and periodicity issues of 2D grids, we project the data onto a 3D Cartesian coordinate system. We convert the 2D latitude-longitude coordinates to 3D unit vectors on the sphere ($R = 1$) using:

$$\begin{aligned} x &= \cos(\phi)\cos(\lambda) \\ y &= \cos(\phi)\sin(\lambda) \\ z &= \sin(\phi) \end{aligned} \tag{C.1}$$

By moving to a 3D Cartesian representation, we treat the Earth not as a bounded rectangle, but as a sphere. In this space, the query-key similarity in RoPE-based attention implicitly utilizes the chordal distance (the straight-line Euclidean distance through the sphere's interior) rather than the great-circle distance (the distance along the surface). Because chordal distance is a strictly increasing function of the great-circle distance and defines a valid metric on the sphere, it preserves local neighborhood structure while avoiding the coordinate singularities of lat–lon grids at the poles. This helps the model represent spherical interactions without polar artifacts, and the model is theoretically able to learn the mathematical relation between these metrics.

3D Vector Transformation

While coordinates are easily transformed, wind vectors (u, v) are local to the tangent plane at (ϕ, λ). To process them within a 3D Platonic Transformer, we must lift these 2D vectors into 3D Cartesian space, this is visualized in Figure C.1. We do this with tangent planes, defined as:

$$\begin{aligned} \hat{\mathbf{e}}_{east} &= [-\sin(\lambda), \cos(\lambda), 0]^\top \\ \hat{\mathbf{e}}_{north} &= [-\sin(\phi)\cos(\lambda), -\sin(\phi)\sin(\lambda), \cos(\phi)]^\top \end{aligned} \tag{C.2}$$

The 3D wind velocity vector $\mathbf{v}_{3D}$ is then computed as a linear combination in the tangent plane:

$$\mathbf{v}_{3D} = u \cdot \hat{\mathbf{e}}_{east} + v \cdot \hat{\mathbf{e}}_{north} \tag{C.3}$$

This transformation preserves the vector magnitude ($\|\mathbf{v}_{3D}\| = \sqrt{u^2 + v^2}$) and ensures the vectors remain perfectly tangent to the sphere ($\mathbf{v}_{3D} \cdot \mathbf{p} = 0$), an example is shown in Figure C.2. The model operates entirely in 3D. For both the vector loss (Section 6.1.2) and for visualization, we project the predicted 3D velocity vectors $\mathbf{v}_{3D}$ back onto the local 2D basis $(\hat{\mathbf{e}}_{east}, \hat{\mathbf{e}}_{north})$ using:

$$u = \mathbf{v}_{3D} \cdot \hat{\mathbf{e}}_{east}, \quad v = \mathbf{v}_{3D} \cdot \hat{\mathbf{e}}_{north} \tag{C.4}$$

This projection ensures that the model's 3D latent predictions remain physically consistent with the original 2D eastward and northward wind components.

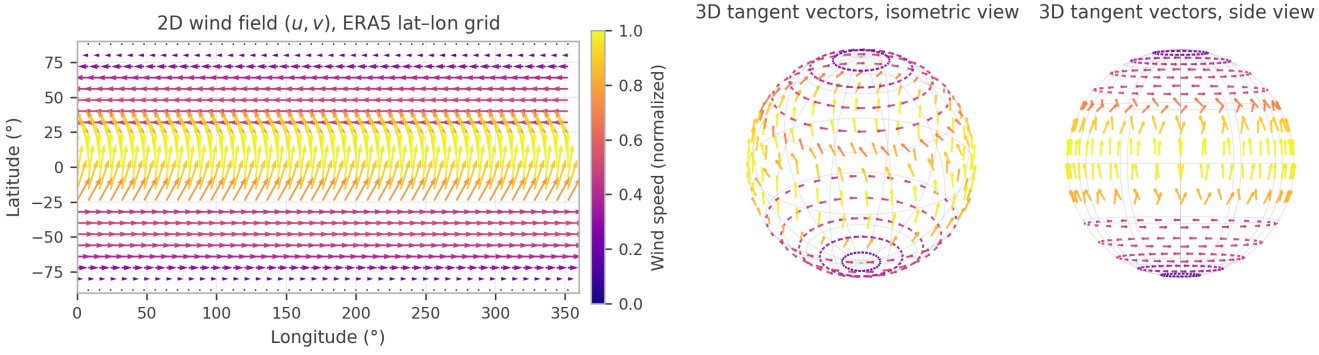


Figure C.2: Transformation of a global Hadley Cell circulation pattern from 2D components into 3D Cartesian vectors tangent to the sphere.

Appendix D

Inductive biases in RoPE for weather data

Modeling Weather Data in 2D with the Latitude-Longitude Grid

RoPE is defined as a series of 2D rotations applied to pairs of feature components, where the rotation angle is the dot product of a coordinate vector and a frequency vector. The most common representation for global weather data, such as ERA5, is a regular latitude-longitude grid. In this 2D setting, the input position is defined as $\mathbf{p} = [\phi, \lambda]^\top$, where ϕ is latitude and λ is longitude. Input wind vectors can be left in their native form as (u, v) components, which are defined in units of m/s.

Log-spaced initialization

To capture multi-scale features, we propose a RoPE initialization with frequencies following a log-space distribution. Following the approach of Bodnar et al. [2025], we sample wavelengths log-uniformly from $0.1°$ (fine-scale detail) up to $360°$ for longitude and $180°$ for latitude (global-scale patterns).[1] This can be seen in Figure D.1. We set the minimum RoPE wavelength to twice the grid resolution ($2 \times 5.625°$ for the ERA5 runs), the Nyquist limit below which spatial frequencies cannot be resolved by the sampling grid and would alias [Nyquist, 1928].

The Longitude Wrap-Around Condition

Because the Earth is a sphere, the dateline ($\lambda = -180°$ and $\lambda = +180°$) represents the same physical point. For the encoding to be periodic in longitude with a period of $360°$, the RoPE angle transformation must satisfy:

$$\cos(f_{lon} \times 360°) = 1 \implies 360° \cdot f_{lon} = 2\pi n, \quad n \in \mathbb{Z} \quad \text{(D.1)}$$

This yields quantized longitude frequencies of the form $f_{lon} = \frac{\pi n}{180}$. Note that the wrap-corrected longitude frequencies must be held non-learnable: they are snapped to integer harmonics of the $360°$ period to guarantee exact periodicity, and allowing them to be learned would let gradient descent move them off this harmonic grid, breaking the wrap-exactness. Latitude frequencies (and all non-wrap runs) remain learnable.

In the our method, we enforce this periodicity by first sampling wavelengths λ in log-space and then "snapping" them to the nearest valid harmonic. Specifically, we compute the harmonic number $n = \text{round}(360°/\lambda)$, and redefine the frequency as $f_{lon} = 2\pi n/360$. This ensures that every frequency component in the longitude dimension completes an integer number of cycles around the globe, preventing "seams" at the dateline during standard self-attention.

Because this method still operates on a 2D projection, it fails to account for the convergence of meridians toward the poles. This results in a "distance distortion" where points at high latitudes are physically closer than their 2D grid coordinates suggest, as illustrated in Figure D.2. Furthermore, unlike the original RoPE spiral intiialization, which adds a deterministic phase offset to frequencies for each head, we distribute the unique sampled frequencies across the available heads. We avoid per-head phase shifts because they can violate the longitude wrap-around, which would introduce phase-discontinuities at the dateline. By instead partitioning the frequency spectrum across heads, we effectively make each head attend to a distinct range of periodic spatial patterns while maintaining global continuity, as shown in Figure D.3.

Symmetry Breaking Under Rotation

While this wrap-around condition works in the trivial case (identity), it is destroyed by the group action in the Platonic Transformer. For a cyclic group C_4 ($90°$ rotations), the rotated frequency $f_{rotated} = R^\top f$ swaps the latitude and longitude com-

[1] Absolute positions are optionally rescaled before entering RoPE (identity for `deg`; $\times \pi/180$ for `rad`.

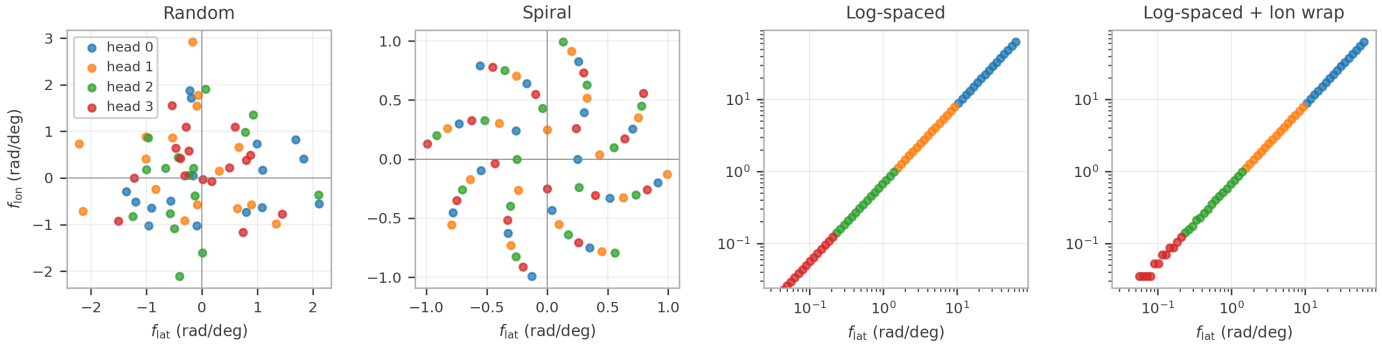


Figure D.1: Comparison of RoPE frequency initialization strategies.

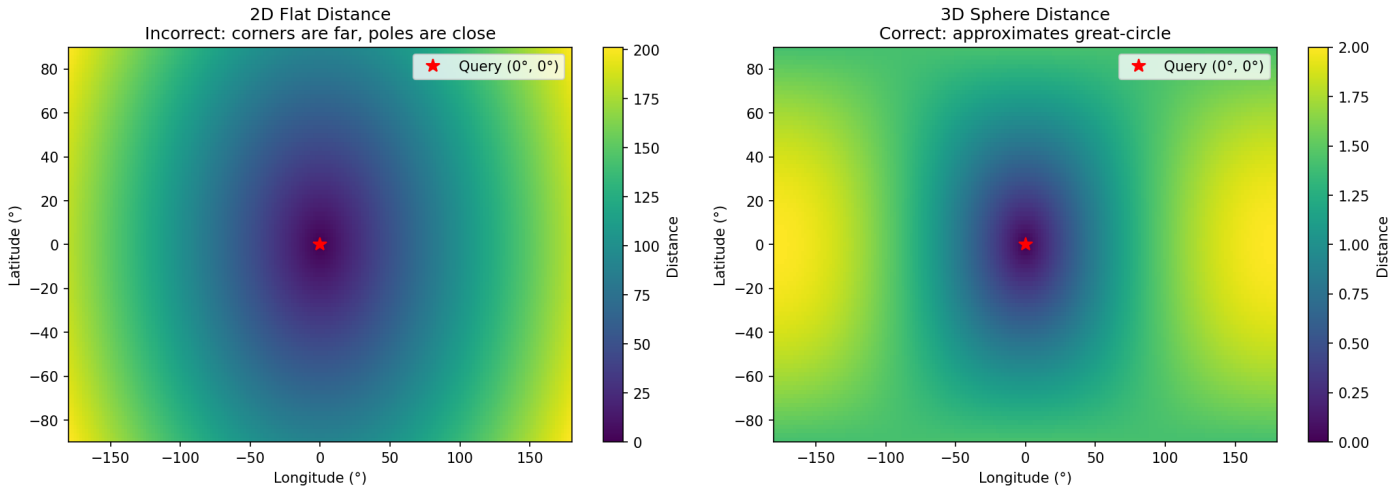


Figure D.2: Spatial distance metric comparison. The **left** panel shows the distorted Euclidean distance on a 2D flat grid, where longitude points near the poles seem further away then they should. The **right** panel shows the 3D chordal distance, which closely approximates the true great-circle distance.

ponents:

$$R_{90^\circ}^{\top} \begin{bmatrix} f_{lat} \\ f_{lon} \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix} \begin{bmatrix} f_{lat} \\ f_{lon} \end{bmatrix} = \begin{bmatrix} f_{lon} \\ -f_{lat} \end{bmatrix} \tag{D.2}$$

After rotation, the unquantized latitude frequency becomes the effective longitude frequency, breaking the periodicity at the dateline. Enforcing periodicity in both dimensions fix this issue, but would effectively model the Earth as a torus, which is unusable for weather modeling.

For a C_8 group (45° rotations), the mixing is even more severe:

$$(R_{45^\circ}^{\top} f)_1 = \frac{1}{\sqrt{2}}(f_{lon} - f_{lat}) \tag{D.3}$$

Because $\frac{1}{\sqrt{2}}$ is irrational, no quantization of f_{lat} can preserve the wrap-around condition. This can be seen in Figure D.4

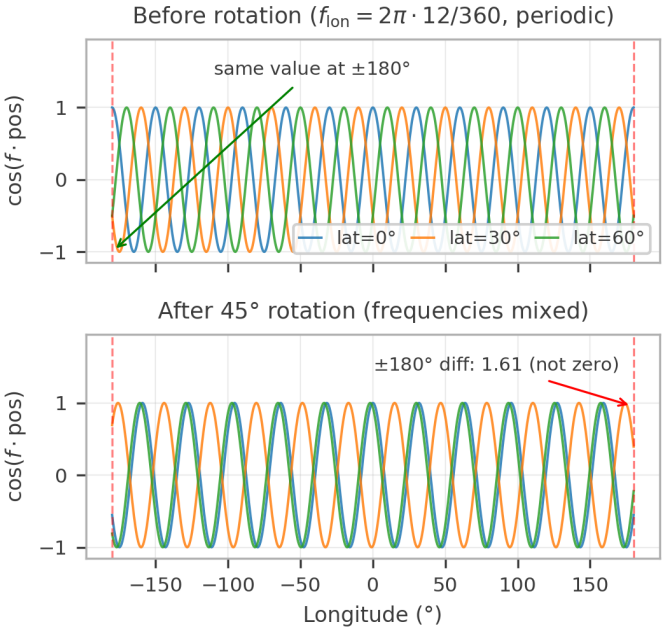


Figure D.4: Visualization of frequency mixing under a 45° rotation (C_8 group). While the signal is periodic at $\pm 180°$ before rotation **(top)**, the rotation causes a discontinuity after rotation **(bottom)**.

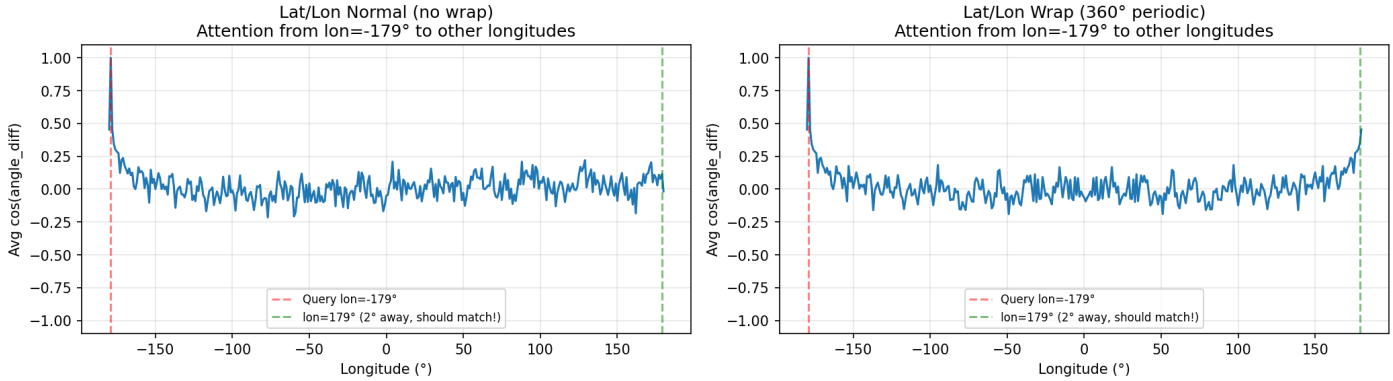


Figure D.3: Attention score distribution across the dateline. The "Wrap" configuration **(right)** shows continuous attention scores from $\lambda = -179°$ to $+179°$, eliminating the artificial "seam" seen in the normal configuration **(left)**.

Appendix E

Details of experiments on Protein MD

E.1 Description of the Dataset

The ADK equilibrium trajectory dataset, comprising 4,171 molecular graphs, is partitioned into training (2,481), validation (827), and test (863) sets, representing a 59.5% / 19.8% / 20.7% split ratio. The model objective is to minimize the Mean Squared Error (MSE) by predicting the displacement to the target Cartesian coordinates at $t + 15$ frames.

E.2 Data Implementation

The all-atom variant is comparatively heavy on a per-sample basis: every graph carries 3,341 nodes, and each of the 2,481/827/863 train/validation/test frames is materialised as a separate PyTorch Geometric `Data` object. The original loader processed an entire partition up front and held the result as a single in-memory Python list, indexing into that list inside `__getitem__`. This design is convenient but does not scale well to multi-worker data loading: because we use the `spawn` multiprocessing context (required for CUDA-safe workers), each `DataLoader` worker receives its own full copy of the list rather than sharing it copy-on-write. Memory therefore grows roughly linearly in the number of workers, which both caps the achievable worker count and makes `persistent_workers` expensive, as those copies are retained across epochs.

We address this with a *lazy* dataset variant. The expensive MDAnalysis processing is run exactly once and the result is written to a per-graph on-disk cache — one pickle file per frame (`graph_000000.pkl`, `graph_000001.pkl`, ...) rather than one monolithic pickle per partition. The dataset object then holds only lightweight metadata: the cache-directory path and the number of graphs. Each `__getitem__(i)` simply reads the single corresponding pickle on demand. As a result, per-worker memory is independent of the dataset size (only the handful of graphs currently in flight (bounded by the batch size and `prefetch_factor`) are ever resident) so workers can be scaled freely, `persistent_workers` and prefetching are essentially free, and start-up no longer pays the reprocessing cost once the cache exists. In practice this removed the memory ceiling that previously limited the configuration to few workers, allowing data loading to keep the GPU saturated at low RAM usage.

E.3 Training Details

We evaluate two primary architectures for the Protein MD experiments: Platonic-Erwin and the Platonic Transformer (hyperparameters in Table E.1. Standard training utilizes the AdamW optimizer with specific parameter groups: weight decay is applied to linear and kernel weights, while biases, RoPE frequencies, layer-scale parameters, and normalization parameters use zero weight decay.

Hyperparameter	Value
Architecture & Capacity: Platonic-Erwin	
Embedding Dimension (`embed_dim`)	96
Hidden Dimensions (`c_hidden`)	[96, 192, 384, 768, 1536]
Encoder Heads (`enc_num_heads`)	[12, 24, 24, 48, 48]
Decoder Heads (`dec_num_heads`)	[12, 24, 24, 48]
Encoder Depths (`enc_depths`)	[2, 2, 2, 6, 2]
Decoder Depths (`dec_depths`)	[2, 2, 2, 2]
Ball Sizes (`ball_sizes`)	[128, 128, 128, 64, 32]
Strides (`strides`)	[2, 2, 2, 2]
MLP Ratio (`mlp_ratio`)	4.0
Architecture & Capacity: Platonic Transformer	
Hidden Dimension (`hidden_dim`)	1152
Layers (`layers`)	7
Heads (`heads`)	72
Dropout (`dropout`)	0
Drop Path Rate (`drop_path_rate`)	0
Attention (`attention`)	False
Use Key (`use_key`)	False
Geometry & Symmetries	
Platonic Group (`solid_name`)	Tetrahedron (T, order 12)
RoPE Initialization (`freq_init`)	`spiral`
RoPE Sigma (`freq_sigma` / `rope_sigma`)	1.0
FFN Readout (`ffn_readout`)	True
Distance Bias (`use_distance_bias`)	True (Platonic-Erwin only)
APE Sigma (`ape_sigma`)	None
Optimization Setup	
Learning Rate (`learning_rate`)	5×10^{-4}
Weight Decay (Platonic-Erwin / Platonic Trans.)	0.05 / 10^{-8}
Scheduler (`scheduler`)	Linear warmup (300 steps) + Cosine decay
Minimum Learning Rate (`eta_min`)	1.0×10^{-6}
Epochs (`epochs`)	200
Batch Size (`batch_size`)	32
Precision (Ablations / Final)	`bf16-mixed` / `fp32`
Gradient Clipping (`grad_clip`)	1.0
Early Stop Metric (`early_stop_metric`)	`val/MSE`
Patience (`patience`)	50
Data Processing	
Delta Frame (`delta_frame`)	15
Cutoff Rate (`cutoff_rate`)	0.25
Load Cached (`load_cached`)	True
Use Lazy Loading (`use_lazy`)	True
Scalar Features (`scalar_features`)	`[]`
Vector Features (`vector_features`)	`["velocity"]`

Table E.1: Key hyperparameters and training details for the Protein MD experiments.

E.4 Ablations

We conduct a systematic ablation study using the Adenylate Kinase (ADK) protein molecular dynamics task to validate specific architectural design choices. The investigation begins with the v1 baseline, which represents the initial integration of Platonic equivariant attention within the hierarchical Erwin ball-tree framework. This baseline serves as the foundation for isolating the performance impacts of various architectural designs.

E.4.1 Weight Decay and Feed-Forward Architecture

We evaluate the impact of regularization and alternative feed-forward network (FFN) designs on the Platonic–Erwin model performance, utilizing the `normal` ball-tree configuration as the control. The results of these ablations are summarized in Table

E.2.

Table E.2: Ablation study: weight decay and FFN design

Variant	test MSE ↓	val MSE ↓
Baseline ($wd = 0.05$)	**2.51828**	**2.29054**
Reduced Weight Decay ($wd = 10^{-8}$)	2.53016	2.31216
FFN with LayerScale	2.63359	2.42732

The baseline configuration employs a weight decay of 0.05 and a SwiGLU-based feed-forward network. Reducing the weight decay to 10^{-8}, like platonic transformers, results in a higher test and validation MSE, confirming that lower regularization allows the model to overfit the training set at the expense of generalizability. Additionally, we investigated replacing the standard SwiGLU block with an FFN utilizing LayerScale, a design choice intended to improve training stability in deep architectures as shown by Wang et al. [2026]. Although this variant reduces total parameter count and maintains a stable training curve, it yields significantly higher error rates than the baseline. This suggests that the complex gating and increased expressivity of the SwiGLU mechanism are essential for capturing the intricate molecular dynamics of the ADK protein.

E.4.2 Cross-Attention Pooling

We explore the transition from static pooling to an equivariant cross-attention mechanism for hierarchical coarsening, where coarse-level nodes act as queries to aggregate information from fine-level nodes. The results are presented in Table E.3.

Table E.3: Ablation study: cross-attention pooling and V2 register tokens

Variant	test MSE ↓
Baseline	**2.51828**
Cross-Attention Pooling	2.53102

While cross-attention pooling is theoretically more expressive, the results indicate performance comparable to the baseline with a slight increase in test error. This suggests that the attention mechanism may require a larger data regime to fully converge compared to the static projections used in the baseline.

E.4.3 Normalization

We investigate the impact of different normalization strategies. QK-Normalization is applied to queries and keys prior to the attention calculation to bound scores and stabilize training dynamics. Additionally, RMSNorm is utilized as a computationally efficient alternative to standard LayerNorm.

Table E.4: Ablation study: normalization schemes

Variant	test MSE ↓
Baseline	2.51828
RMSNorm	2.49520
QK-Norm	2.45658
QK-Norm + RMSNorm	**2.44100**

The results in Table E.4 indicate that both methods individually outperform the baseline. QK-Normalization provides substantial stability and accelerates the learning process. While RMSNorm is primarily adopted for its reduced parameter count and speed, it also resulted in improved learning performance.

E.4.4 Integrative Ablations: Normalization, Registers tokens, and Ball Scaling

This phase of the study investigates the compound effects of combining optimized normalization (QK-norm and RMSNorm) with register token implementations (Section A.2) and increased spatial partitions (ball sizes). The results are summarized in Table E.5. During this phase, we also experimented with adding sink tokens solely during the attention computation phase (Transient Sinks and throwing them away immediately afterward, rather than propagating their hidden states to subsequent layers. This approach did not perform as well as fully persistent register tokens.

Table E.5: Integrative ablation study on normalization, registers, and scaling. All variants build on the QK-Norm + RMSNorm base.

Registers	Init	Other	test MSE ↓	val MSE ↓
–	–	–	2.44100	2.18656
Transient Sinks	Zero Init	–	2.50276	2.28117
Persistent Registers	Zero Init	–	2.38138	**2.16154**
Persistent Registers	Trunc. Normal	–	2.40530	2.20803
Persistent Registers	Zero Init	2× Ball Size	**2.37949**	2.22556

A clear synergy emerges between the Persistent Registers implementation and the QK-norm/RMSNorm configuration: together they substantially outperform the Transient Sinks variants, with the zero-initialized persistent registers attaining the lowest test MSE in this series. We do not have a definitive structural explanation, but the result is consistent with the registers acting as a persistent *storage* rather than as transient attention sinks: because their state is propagated across layers, they can accumulate and redistribute global context, which the discarded sinks cannot.

E.4.5 Message passing pre-attention

We evaluate alternative local interaction layers and optimized attention kernels to determine their impact on computational efficiency and predictive accuracy. The original Erwin architecture precedes its ball-tree attention with a message-passing (MPNN) layer to inject local geometric structure; we evaluate replacing it with k-Nearest Neighbour (KNN) attention message

passing prior to the ball-tree attention. We adopt k-NN attention because it fills a similar role as the MPNN while being more efficient, remaining equivariant, and scaling better: an equivariant MPNN must form and transform per-edge messages for every group element, whereas k-NN attention reuses shared equivariant projections across neighbours and composes with optimized attention kernels.

Table E.6: Ablation study: message passing. All configurations include QK-normalization and RMSNorm.

Variant	**test MSE** $\downarrow$
Baseline	**2.51828**
KNN Message Passing + Ball-Tree Attention	2.55398

The inclusion of a KNN-based self-attention message-passing layer before the primary ball-tree attention resulted in increased validation instability.

Appendix F

ERA5 Weather Data

F.1 Description of the Dataset

We train and evaluate on ERA5 [Hersbach et al., 2020], the fifth-generation atmospheric reanalysis produced by the European Centre for Medium-Range Weather Forecasts (ECMWF). ERA5 combines a numerical weather prediction model with historical observations via data assimilation to produce a physically consistent four-dimensional record of the global atmosphere from 1940 to the present, at a native resolution of approximately $0.25°$ (≈ 28 km). All data is accessed through WeatherBench 2 [Rasp et al., 2024], a standardised benchmark for data-driven global weather prediction that provides pre-processed ERA5 fields as Zarr archives on Google Cloud Storage, conservatively remapped to coarser equiangular grids and subsampled to 6-hourly intervals. We use $5.6°$ (64×32 grid points). The set of atmospheric variables used as model inputs and targets is listed in Table F.2.

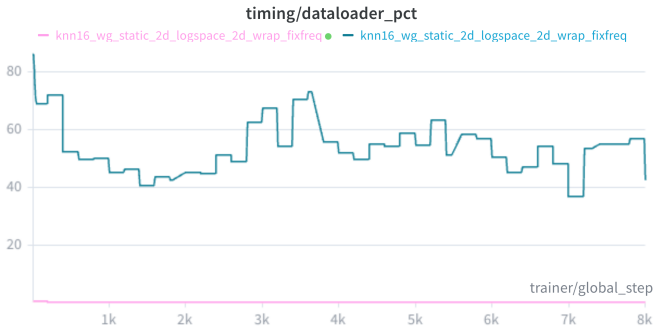


Figure F.1: Comparison of dataloader overhead as a percentage of total training step time. Staging data to NVM minimizes the I/O bottleneck, allowing for near-maximal GPU utilization. Blue is training with local storage, Pink is training with NVM storage

F.2 Storage and Data Pipeline

Storage Optimization with Zarr v3

To eliminate the overhead associated with traditional formats and Dask-Zarr implementations for threading, we utilize **Zarr v3**. Data is re-chunked to 2-5 MB chunk or to chunks of size $(1, n_{lat}, n_{lon})$ or $(1, n_{lat}, n_{lon}, n_{level})$. By doing so, we ensure that the I/O pattern aligns with the training requirements as well as optimal inode usage on HPC servers.

Non-Volatile Memory (NVM) Staging

A significant bottleneck in training on high-resolution weather datasets is the I/O throughput required to feed the model. Initial profiling using a custom `TimingBottleneckCallback` revealed that when utilizing standard local storage, the dataloader wait time accounted for approximately 65% of the total training step wall-clock time. This indicated that the GPU was frequently idle while waiting for batch retrieval.

To resolve this, we implemented a data-staging protocol where the required dataset is copied to local Non-Volatile Memory (NVM) storage prior to the commencement of long-duration training runs. Although the initial staging carries time cost, the operational benefits are substantial: the dataloader overhead dropped to 0.1% of the training step (as shown in Figure F.1), effectively shifting the training regime from being I/O-bound to entirely compute-bound.

HPC Efficiency via SquashFS

The used supercomputer resources, exhibited performance degradation when handling the millions of inodes generated by large Zarr datasets. To mitigate metadata overhead:

1. The Zarr directory is compressed into a single **SquashFS** (.sqsh) image using `mksquashfs`.
2. This image is mounted as a read-only loopback device at runtime.

As a result, per-chunk metadata lookups are resolved locally from the mounted image rather than by the file system's metadata server, and reads stream sequentially from a single contiguous file.

F.3 variables and statistics

An overview of the data used during training is given in Table F.2. Pressure data is provided at 13 levels: {50, 100, 150, 200, 250, 300, 400, 500, 600, 700, 850, 925, 1000} hPa.

The model ingests the atmospheric state as a per-node feature vector built from 4 surface-level and 6 pressure-level dynamic variables, the latter resolved at the 13 pressure levels listed above. How the horizontal wind is counted depends on whether it is treated as two independent scalars or as a single geometric vector. In the *component* view of Table F.2, the zonal and meridional wind (u, v) are tallied as separate channels, giving $4 + 6 \times 13 = 82$ dynamic channels per timestep. In the geometric pipeline the wind is instead carried as a single vector field, so the state comprises $2 + 4 \times 13 = 54$ scalar channels and $1 + 1 \times 13 = 14$ wind-vector fields.

Treating wind as a vector makes the input dimensionality depend on the modelling topology. In the 2D latitude–longitude

Variable	Type	Unit	Channels
Surface-level (Dynamic)			
2-meter temperature	Surface	K	1
10-meter U-wind	Surface	m/s	1
10-meter V-wind	Surface	m/s	1
Mean sea level pressure	Surface	Pa	1
Pressure-level (Dynamic)			
Geopotential	Pressure	m^2/s^2	13
Specific humidity	Pressure	kg/kg	13
Temperature	Pressure	K	13
U-component of wind	Pressure	m/s	13
V-component of wind	Pressure	m/s	13
Vertical velocity	Pressure	Pa/s	13
Static variables			
Surface geopotential	Static	m^2/s^2	1
Land-sea mask	Static	0–1	1
Soil type	Static	Categorical	1
Total dynamic channels per timestep			**82**

Table F.2: Variables used, including dynamic surface-level, pressure-level, and static fields.

setting each wind field carries its two tangent-plane components (u, v); in the 3D Cartesian setting each field is lifted to three Cartesian components (v_x, v_y, v_z) that remain tangent to the sphere (Appendix C). Every one of the 14 wind fields therefore costs *one extra channel* in 3D, raising the dynamic channel count from 82 to $54 + 14 \times 3 = 96$; the position descriptor likewise grows from (λ, φ) in 2D to (x, y, z) in 3D. Table F.1 summarises this. On top of the dynamic block the model also receives the 3 static fields and a learned sinusoidal time embedding (Section B.2); the dynamic fields are stacked over the input timesteps before the encoder lift.

Per-node input	**2D** (λ, φ)	**3D** (x, y, z)
Components per wind field, d	2 (u, v)	3 (v_x, v_y, v_z)
Scalar channels	54	54
Wind-vector channels ($14\,d$)	28	42
Dynamic channels	**82**	**96**
Static channels	3	3
Position descriptor	2	3

Table F.1: Per-node input composition in the 2D and 3D pipelines. Because each of the 14 wind fields is a geometric vector, the 3D lift adds one component per field (+14 channels), so the dynamic channel count rises from 82 to 96. The dynamic block is additionally stacked over `input_timesteps`=2 input states before entering the encoder.

where μ and σ are the global mean and standard deviation, respectively, stored in a JSON manifest for on-the-fly normalization by the data loader. We want per variable normalization since the ranges of individual variables can differ significantly, making batch-norm unsuitable.

Two static fields are not natively continuous. The *land–sea mask* is binary at the native $0.25°$ resolution; conservative regridding to the coarser grids turns it into a continuous field on $[0, 1]$, which is well defined and physically meaningful (it is simply the land fraction of each cell), so feeding it as a continuous channel is the natural choice. The *soil type*, by contrast, is a categorical field (ERA5 slt: integer classes 0–7, where 1–5 are the soil-texture classes *Coarse* through *Very fine*, 6–7 are organic classes, and 0 marks non-land points). Because classes 1–5 are ordered by texture from coarse to fine, a conservatively-averaged value is roughly monotone in soil fineness rather than pure noise. Two caveats remain: the scale is ordinal, not interval (the numeric gaps between classes are not metrically meaningful, which the subsequent z-scoring ignores) and the ordering is broken by the off-continuum classes (organic, and the ocean sentinel), where averaging across a boundary yields a value belonging to an unrelated category. We therefore treat the coarsened, standardised index as an approximate continuous proxy and flag it as a simplification. Treating the fields in this manner is done in related works like [Zhdanov et al., 2026]

Data Standardization

Prior to training, global statistics are calculated across the entire temporal extent per variable. Input variables are normalized to zero mean and unit variance using:

$$x_{norm} = \frac{x - \mu}{\sigma} \tag{F.1}$$

Appendix G

Details of experiments on ERA5

G.1 Deviations from default setup per experiment

Deviations from default setup for Experiment 1: Grid Topology Loss: pure MSE on (u, v) components (no angle–magnitude objective for vectors). We use full GraphCast variable weighting here (including on vector components, as the weights are defined for all components in the MSE case and all runs in this experiment use this loss so it is a good cross comparison).

Deviations from default setup for Experiment 2: Vector Modeling and Loss Formulation GraphCast variable weighting is disabled for the vector channels (including for the pure MSE objective) for a fair cross-loss comparison.

Deviations from default setup for Experiment 5: Equivariance and Weight Sharing To ensure comparability, all models were trained with an effective batch size of 8, constrained by the memory limits of the largest configuration on a single GPU.

G.2 Objective

At a given step it splits into a scalar term $\mathcal{L}_{\mathrm{s}}$ and a wind-vector term $\mathcal{L}_{\mathrm{v}}$,

$$\mathcal{L} = \mathcal{L}_{\mathrm{s}} + \mathcal{L}_{\mathrm{v}}. \tag{G.1}$$

Both follow the GraphCast weighting scheme [Lam et al., 2023], in which the per-node error is scaled by three independent factors: a latitude (cell-area) weight w^{lat}, a per-pressure-level weight w^{lev}, and a per-variable weight w^{var}. Each factor can be toggled independently on the scalar and vector paths.

G.2.1 Variable loss weights

Two channel-wise factors act on top of the latitude weight, both taken from GraphCast. The *per-level* factor up-weights the higher-mass (lower-altitude) pressure levels: a variable at pressure level p (in hPa) receives

$$w^{\mathrm{lev}}(p) = \frac{p}{\bar{p}}, \qquad \bar{p} = \frac{1}{13}\sum_{\ell=1}^{13} p_\ell, \tag{G.2}$$

with $\bar{p}$ the mean over the 13 levels; surface (single-level) variables take $w^{\mathrm{lev}}{=}1$. The *per-variable* factor w^{var} balances the relative contribution of each physical field and is listed in Table G.1.

Variable	w^{var}
2 m temperature	1.0
Mean sea-level pressure	0.1
10 m wind (u, v)	0.1
Geopotential	1.0
Specific humidity	1.0
Temperature	1.0
Vertical velocity	1.0
Wind $(u, v$, pressure levels)	1.0

Table G.1: Per-variable loss weights w^{var} (GraphCast convention [Lam et al., 2023]). The near-surface diagnostics (10 m wind, mean sea-level pressure) are down-weighted relative to the prognostic atmospheric fields. In the vector-valued wind losses, the speed and angle terms carry no separate coefficients; they are summed with the component MSE and weighted jointly by the factors above.

G.2.2 Latitude loss weights

The equiangular grid over-samples high latitudes, so each node n is weighted by its cell area. With latitude φ_n (recovered as $\arcsin z_n$ in the 3D Cartesian pipeline),

$$w_n^{\mathrm{lat}} = \frac{\cos\varphi_n}{\frac{1}{N}\sum_{m=1}^{N}\cos\varphi_m}, \tag{G.3}$$

i.e. $\cos\varphi$ normalised to unit mean over the N nodes.

G.2.3 Loss function

Writing $\hat{y}$ for the prediction and y for the target, the scalar term is a weighted mean squared error over the N nodes and C_{s} scalar channels,

$$\mathcal{L}_{\mathrm{s}} = \frac{1}{N\,C_{\mathrm{s}}}\sum_{n=1}^{N}\sum_{c=1}^{C_{\mathrm{s}}} w_n^{\mathrm{lat}}\, w_c^{\mathrm{lev}}\, w_c^{\mathrm{var}}\big(\hat{y}_{n,c} - y_{n,c}\big)^2. \tag{G.4}$$

The vector term scores each of the K wind fields as a geometric entity. In the 3D pipeline the predicted and target vectors are first projected onto the local tangent basis $(\hat{e}_\lambda, \hat{e}_\varphi)$, discarding the unobserved radial component, so the objective is identical and comparable across the 2D and 3D settings (Section 6.3). For node n and wind field k with prediction $\hat{\mathbf{v}}$ and target $\mathbf{v}$, the default ANGLE+MAGNITUDE per-element loss is

$$\ell_{n,k} = \underbrace{\big(1 - \cos(\hat{\mathbf{v}}, \mathbf{v})\big)}_{\text{direction}} + \underbrace{\big(\|\hat{\mathbf{v}}\| - \|\mathbf{v}\|\big)^2}_{\text{magnitude}}, \tag{G.5}$$

the plain-MSE variant instead uses $\|\hat{\mathbf{v}} - \mathbf{v}\|_2^2$. The full set of vector objectives we compare is given in Table 6.1. The vector loss is the (optionally weighted) area-mean

$$\mathcal{L}_{\mathrm{v}} = \frac{1}{N\,K}\sum_{n=1}^{N}\sum_{k=1}^{K} w_n^{\mathrm{lat}}\, w_k^{\mathrm{lev}}\, w_k^{\mathrm{var}}\, \ell_{n,k}, \tag{G.6}$$

G.3 Evaluation methodology

Following the WeatherBench 2 protocol [Rasp et al., 2024], forecasts are initialised twice daily, at 00:00 and 12:00 UTC, on every day of the held-out test year (2020). Before scoring, both the forecasts and the ERA5 ground truth are first-order conservatively regridded to $1.5°$. All metrics are computed per grid point and then latitude-weighted by $\cos\varphi$ (normalised to unit mean, Section G.2) to account for the varying cell area of the equiangular grid, and finally averaged per variable and pressure level.

G.3.1 Deterministic evaluation metrics

ACC. The Anomaly Correlation Coefficient (ACC) measures the correlation between the forecast and analysis anomalies relative to climatology:

$$\mathrm{ACC} = \frac{\sum_{i=1}^{N}(f_i - c_i)(a_i - c_i)}{\sqrt{\sum_{i=1}^{N}(f_i - c_i)^2 \sum_{i=1}^{N}(a_i - c_i)^2}} \tag{G.7}$$

where f_i is the forecast (predicted) value, a_i the analysis (truth) value, c_i the climatological mean, and N the number of spatial grid points.

Aggregated skill. Following Couairon et al. [2024], Diaconu et al. [2026], we additionally report an aggregated skill score, defined as the mean relative RMSE improvement over a reference model across the headline variable set V (500 hPa geopotential Z500, 850 hPa temperature T850, 700 hPa specific humidity Q700, 850 hPa wind vector WV850, 2 m temperature T2M, 10 m wind speed WS10, and mean sea-level pressure MSLP):

$$\text{Aggregated Skill} = \frac{1}{|V|}\sum_{v\in V}\left(1 - \frac{\mathrm{RMSE}_v(\text{model})}{\mathrm{RMSE}_v(\text{reference})}\right) \times 100 \tag{G.8}$$

Because each experiment in this chapter is a controlled ablation rather than a comparison against an external forecast, the reference model is chosen per experiment. Every other arm in that experiment reports skill relative to this control, so 0% reproduces the reference exactly, positive values are RMSE improvements over it, and negative values are regressions.

G.3.2 Spectral evaluation metrics

Spectral Consistency. Following Kasteleyn et al. [2026], we additionally report two spectral-consistency diagnostics on the 500 hPa kinetic-energy spectrum $E(k)$, obtained via a spherical-harmonic transform of the wind field. The spectral residual is the RMSE between the log-transformed predicted and reference spectra:

$$\mathrm{SpecRes} = \sqrt{\frac{1}{K+1}\sum_{k=0}^{K}\left(\log E_{\text{pred}}(k) - \log E_{\text{ref}}(k)\right)^2} \tag{G.9}$$

The spectral divergence normalises the spectrum into a distribution $P(k) = E(k)/\sum_{j=0}^{K} E(j)$ and computes the 1-Wasserstein distance between the predicted and reference distributions:

$$\mathrm{SpecDiv} = W_1(P_{\text{ref}}, P_{\text{pred}}) = \sum_{k=0}^{K}\left|\sum_{j=0}^{k} P_{\text{ref}}(j) - \sum_{j=0}^{k} P_{\text{pred}}(j)\right| \tag{G.10}$$

G.3.3 Probabilistic evaluation metrics

CRPS. The fair (unbiased) CRPS [Zamo and Naveau, 2018] for an N-member ensemble $\mathbf{x}^{1:N}$ and observation y is given as:

$$\mathrm{CRPS}(\mathbf{x}^{1:N}, y) = \frac{1}{N}\sum_{n}|x^n - y| - \frac{1}{2N(N-1)}\sum_{n,n'}|x^n - x^{n'}| \tag{G.11}$$

This is a proper scoring rule for a univariate forecast distribution: it generalises the absolute error to ensembles and is minimised only when the predicted distribution matches the truth. It decomposes into a *reliability* term (first sum, how far members sit from the observation) and a *sharpness* term (second sum, ensemble spread, credited only where uncertainty is warranted). Lower is better; it jointly rewards accuracy and honest spread. The $\frac{1}{N(N-1)}$ normalisation makes the spread term unbiased at small N, which matters during training where $N{=}2$.

Ensemble-mean RMSE. The latitude-weighted RMSE of the ensemble mean, $\sqrt{(\bar{\mathbf{x}} - y)^2}$. This is the same number as the deterministic skill of Section 8.3, now applied to the averaged ensemble, and lets us check that adding stochasticity did not cost point accuracy. Because averaging smooths, the ensemble mean is expected to score slightly better than any single member at long lead.

Spread-to-skill ratio. The fair spread-to-skill ratio

$$\mathrm{SSR} = \sqrt{\frac{M+1}{M}}\,\frac{\text{spread}}{\text{skill}}, \qquad \text{spread} = \sqrt{\overline{\mathrm{Var}}}, \qquad \text{skill} = \mathrm{RMSE}(\bar{\mathbf{x}}), \tag{G.12}$$

compares the ensemble's own predicted uncertainty (root mean ensemble variance) to its actual error (RMSE of the ensemble mean). The $\sqrt{(M+1)/M}$ factor [Alet et al., 2025] makes the calibrated target exactly 1 for a finite M-member ensemble. $\mathrm{SSR} \approx 1$ is well-calibrated, $\mathrm{SSR} < 1$ is *under-dispersed* (overconfident: the ensemble is too narrow for its own errors), and $\mathrm{SSR} > 1$ is over-dispersed.

G.4 Training Details

G.4.1 Hyperparameters

The Platonic-Erwin architecture and training scheme for the geometry experiments utilize the hyperparameters as shown in Table G.2. These configurations enable the model to capture the 3D vector geometries and respect spherical topologies efficiently. The different shapes for the equivariance and weight-sharing

Hyperparameter	Default value
	Backbone architecture (default)
Embedding dim (`embed_dim`)	128
Stage hidden dims (`c_hidden`)	$[128, 128, 256, 512]$
Encoder depths / heads	$[2, 2, 6, 2]$ / $[8, 8, 16, 32]$
Decoder depths / heads	$[2, 2, 2]$ / $[8, 8, 16]$
Ball sizes / strides	$[128, 128, 64, 64]$ / $[2, 2, 2]$
MLP ratio	4.0
Normalization	RMSNorm + QK-norm, $\varepsilon = 10^{-5}$
FFN activation	SwiGLU
FFN readout / message-passing steps	true / 0
Register tokens	0
Distance bias	true
Dropout / drop-path	none
	Geometry, cross-attention and positional encoding
Symmetry group (`solid_name`)	`trivial` (3D, $G = 1$)
Spatial dims / output vector dims	3 / 3 (3D-Cartesian; tangent projection in loss)
Coordinate system	`xyz`
HEALPix coarse grid (`healpix_nside`)	16 (nested ordering: false; 0–360 lon)
Cross-attention k-NN (`cross_attn_knn`)	16
Static k-NN cross-attention	true (bipartite neighbours cached after first forward)
RoPE init / learnable freqs	`spiral` / true
RoPE freq σ (`freq_sigma`)	1.0
APE σ (`ape_sigma`)	null (APE off)
Time embedding	32 freqs, periods $[1, 8766]$ h, learnable
	Loss
Vector loss (`vector_loss_type`)	`tangent_2d_angle_magnitude` (3D) / `angle_magnitude` (2D)
Loss weighting (GraphCast-style)	latitude + scalar level + scalar var + vector level; vector var off
	Optimization and trainer
Optimizer	AdamW (β_1=0.9, β_2=0.999, ε=10^{-8}; decoupled WD on all parameters)
Learning rate	5×10^{-4}
Weight decay	0.05
Scheduler	WSD: linear warmup + constant + linear cooldown (`decay_fraction`: 0.1, $\eta_{\min} = 10^{-6}$)
Warmup steps / total steps	3000 / 100 000
Precision / grad clip	bf16-mixed / 1.0 (global norm)
EMA	none
Validation / checkpointing	every 500 steps; best `val/loss` + last kept
`seed_everything`	13 (single seed per run)
Batch size	eff. 32 (experiments 1–4); eff. 8 for the equivariance experiments (5–6)

Table G.2: Default hyperparameters shared by all 5.6° geometry runs.

Backbone hyperparameter	**Small** (G=1, default)	**Big** (2D-full / 3D-big)
Embedding dim (`embed_dim`)	128	256 (2D) / 384 (3D)
Stage hidden (`c_hidden`)	$[128, 128, 256, 512]$	$[256, 256, 512, 1024]$ (2D) / $[384, 384, 768, 1536]$ (3D)
Encoder heads (`enc_num_heads`)	$[8, 8, 16, 32]$	$[8, 8, 16, 32]$ (2D) / $[12, 12, 24, 48]$ (3D)
Per-stage head dim	16	32
Encoder depths	$[2, 2, 6, 2]$	$[2, 2, 6, 2]$
Ball sizes / strides	$[128, 128, 64, 64]$ / $[2, 2, 2]$	identical

Table G.3: The two backbone shapes used across the weight-sharing experiments; depth, ball-size and stride schedules are held constant. The equivariant models reuse the **Big** shape unchanged.

ablation are given in Table G.3. Table G.2 lists the *default* configuration shared by every 5.6° run; Table G.3 shows the two backbone shapes used across the weight-sharing ablation. In this cohort the cross-attention uses $k = 16$ nearest HEALPix neighbours per node, the largest neighbourhood that fit in a single GPU's memory under this setup; the final-evaluation cohort later raised this to $k = 24$ (Table G.4).

Ball-tree partitions. Figure G.1 shows the concrete hierarchical ball-tree partition produced by the default weather configuration (Table G.2).

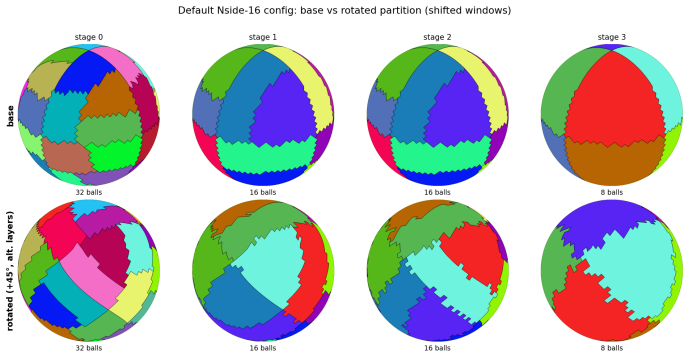


Figure G.1: Hierarchical ball-tree partition actually used by Platonic-Erwin in the default weather configuration (`nside-16` latent mesh, stride-2 pooling, nested HEALPix ordering). The four panels are the successive U-Net levels (level 0–3); colours denote distinct balls. Ball size and ball count per level: level 0 (128, 24 balls) → level 1 (128, 12) → level 2 (64, 12) → level 3 (64, 6).

G.4.2 Final-evaluation configurations

The final evaluation (Section 8.3) trains the two winning configurations from the earlier ablations (non-weight-sharing TRIVIAL (APE-only) and weight-sharing CYCLIC-4 (trivial readout + APE)) at three matched widths, giving the six scaling-ladder runs listed in Table G.5. All six share a single training recipe (Table G.4); the only architectural differences are the symmetry group and.

	Embed 128	Embed 256	Embed 384
c_hidden	[128, 128, 256, 512]	[256, 256, 512, 1024]	[384, 384, 768, 1536]
Encoder heads	[4, 4, 8, 16]	[8, 8, 16, 32]	[12, 12, 24, 48]
Decoder heads	[4, 4, 8]	[8, 8, 16]	[12, 12, 24]
TRIVIAL total / trainable	20M / 20M	80M / 80M	178M / 178M
CYCLIC-4 total / trainable	20M / 5M	80M / 20M	178M / 45M

Table G.5: The three widths of the scaling ladder.

G.4.3 CRPS fine-tuning schedule

Both flagship models of the scaling ladder are converted into ensemble forecasters with an identical fine-tuning recipe, summarised in Table G.6; the two runs differ only in which deterministic checkpoint they warm-start from.

Initialisation and trainable parameters. Each run is warm-started *weights-only* from the final checkpoint of its 100 000-step deterministic base run (Table G.4), loaded non-strictly so that the only freshly initialised parameters are the noise-injection path of Equation 8.1: the per-block noise projections (∼5% of the parameters), whose output "mixer" is near-zero initialised so that fine-tuning starts from the deterministic model's behaviour.

Objective. The deterministic objective is replaced by the latitude- and variable-weighted fair CRPS of Equation 8.2, estimated with $N{=}2$ ensemble members per training sample [Zhdanov et al., 2026]. CRPS is applied per channel for the scalar fields and per component on (u, v) for the wind fields; the deterministic speed and angle terms are dropped.[1] Each member draws an independent latent $\mathbf{z} \in \mathbb{R}^{32}$ from $\mathcal{N}(\mathbf{0}, \mathbf{I})$ per forward pass, with a fixed noise seed for reproducibility.

Schedule. Fine-tuning runs for 30 000 single-step (6 h) prediction steps with AdamW at a peak learning rate of 1×10^{-4} (1/5 of the pretraining peak) and the same WSD shape as pretraining: linear warmup over the first 1 000 steps (from 1% of peak), a constant plateau, and a linear cooldown over the final 10% (3 000 steps) to $\eta_{\min} = 10^{-6}$. The effective batch size of 8 is preserved by halving the per-device batch to 2 and compensating with 2× gradient accumulation on 2 GPUs: the $N{=}2$ member forwards double the activation footprint, keeping memory on par with the deterministic runs.

Hyperparameter	Value
Warm start	deterministic final checkpoint (weights-only, non-strict)
Fresh parameters	noise-SwiGLU head (∼5%, near-zero-init mixer)
Frozen parameters	none (full network fine-tuned)
Objective	fair CRPS (Equation 8.2), $N{=}2$ members per sample
Vector treatment	per-component CRPS on (u, v) (no speed/angle terms)
Loss weighting	as deterministic: latitude + scalar level/var. + vector level
Noise latent	$\mathbf{z} \in \mathbb{R}^{32} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$, per member per forward pass
Optimizer	AdamW ($\beta_1{=}0.9$, $\beta_2{=}0.999$, $\varepsilon{=}10^{-8}$), weight decay 0.05
Peak learning rate	1×10^{-4}
Scheduler	WSD: warmup 1000 steps, constant, cooldown last 10% to 10^{-6}
Fine-tuning steps	30 000
Rollout length during fine-tuning	1 step (6 h; no autoregressive rollout)
Batch size	eff. 8 (per-device 2 × grad. accum. 2 × 2 GPUs, DDP)
Precision / grad clip	bf16-mixed / 1.0 (global norm)
Seeds (`seed_everything` / noise)	13 / 13 (rank-offset under DDP)
Evaluation ensemble	$M{=}10$ members, fresh noise per member per rollout step

Table G.6: CRPS fine-tuning schedule, shared by the TRIVIAL (178M) and CYCLIC-4 (45M trainable) flagships.

G.5 Ablations

Table G.7: Preliminary architectural ablations on the 5.6° ERA5 task. No run scores significantly better.

Variant	Test. Loss
Baseline (linear proj., no reg.)	0.043
+ 4 register tokens	0.043
+ Cross-attention pooling	0.043

[1] These terms compensate for a bias of the deterministic objective itself: a component-wise MSE is minimised by the conditional mean, which by Jensen's inequality ($|\mathbb{E}[\mathbf{v}]| \leq \mathbb{E}[|\mathbf{v}|]$) systematically underestimates wind speed. CRPS instead rewards members that behave like samples from the predictive distribution rather than its mean, so the underestimate vanishes at the optimum and no corrective terms are needed.

Hyperparameter	Value
Backbone architecture (shared skeleton, all three widths)	
Encoder depths / decoder depths	$[2, 2, 6, 2]$ / $[2, 2, 2]$
Ball sizes / strides	$[128, 128, 64, 64]$ / $[2, 2, 2]$
Per-stage head dim	32
MLP ratio / message-passing steps	4.0 / 0
Normalization	RMSNorm + QK-norm, $\varepsilon = 10^{-5}$
FFN activation / readout	SwiGLU / true
Register tokens	0
Geometry, cross-attention and positional encoding	
Spatial dims / output vector dims / coordinate system	2 / 2 / `latlon` (`lon_wrap`: true)
HEALPix coarse grid (`healpix_nside`)	16 (nested ordering: false; 0–360 lon)
Cross-attention k-NN (`cross_attn_knn`)	24, static (pre-norm + QK-norm)
Pooling position feature	lifted relative position, isotropic RMS norm, radians
APE (`ape_sigma` / init / learnable)	1.0 / `spiral` / false (fixed)
RoPE (`freq_sigma` / init / positions)	1.0 / `spiral` / radians
Distance bias	true
Time embedding	32 freqs, periods $[1, 8766]$ h, learnable
Loss	
Vector loss (`vector_loss_type`)	`tangent_2d_mse_speed_angle` (MSE + magnitude + angle; $\varepsilon_{\text{ang}} = 10^{-3}$)
Loss weighting (GraphCast-style)	latitude + scalar level + scalar var. + vector level; vector var. off
Optimization and trainer	
Optimizer / LR / weight decay	AdamW (β_1=0.9, β_2=0.999, ε=10^{-8}) / 5×10^{-4} / 0.05
Scheduler	WSD (warmup 3000, `decay_fraction`: 0.1, $\eta_{\min} = 10^{-6}$), 100 000 steps
Precision / grad clip	bf16-mixed / 1.0 (global norm)
Dropout / drop-path / EMA	none
Validation / checkpointing	every 500 steps; best `val/loss` + last kept
`seed_everything`	13 (single seed per run)
Batch size	eff. 8 (per-device 4 $\times$ 2 GPUs, DDP)

Table G.4: Shared training recipe for the final-evaluation scaling ladder. Every row is identical across all six runs of Table G.5; the two branches differ only on `solid_name` (`trivial_2` vs. `cyclic_4`) and on RoPE, which is off for TRIVIAL and on (spiral init, learnable frequencies) with a trivial readout head for CYCLIC-4.

Appendix H

Additional Results: Weather Forecasting & Geometry Ablations

H.1 Experiment 1: Grid Topology

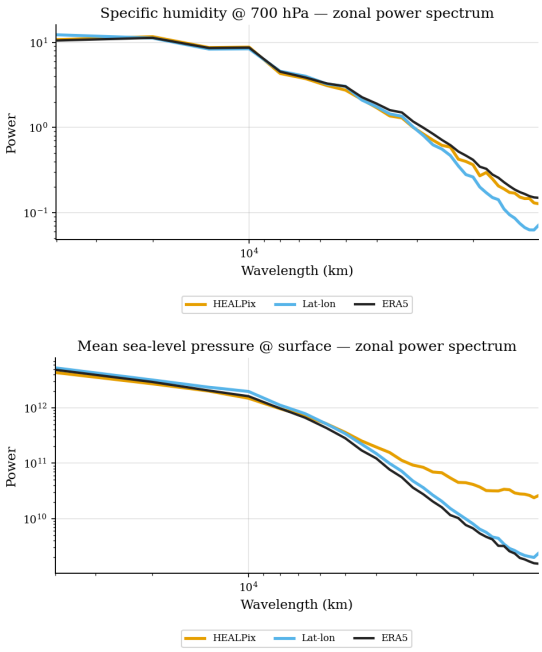


Figure H.1: Zonal power spectra at lead 72 h for the grid comparison, continued from Figure 6.2 (right column): specific humidity at 700 hPa **(top)** and mean sea-level pressure **(bottom)**. ERA5 (black) is the reference.

H.2 Experiment 2: Vector Modeling and Loss Formulation

Unit-Vector Squared Difference as Cosine Distance

The FASTNET directional term uses the squared difference of unit vectors. Expanding:

$$\left\| \frac{\hat{\mathbf{v}}}{\hat{s}} - \frac{\mathbf{v}}{s} \right\|^2 = \left\| \frac{\hat{\mathbf{v}}}{\hat{s}} \right\|^2 + \left\| \frac{\mathbf{v}}{s} \right\|^2 - 2\, \frac{\hat{\mathbf{v}}}{\hat{s}} \cdot \frac{\mathbf{v}}{s} \tag{H.1}$$

$$= 1 + 1 - 2\cos\theta \tag{H.2}$$

$$= 2(1 - \cos\theta), \tag{H.3}$$

where both unit vectors have norm 1 (with $\hat{s} = \|\hat{\mathbf{v}}\|$ and $s = \|\mathbf{v}\|$ clamped by ε at calm-wind pixels) and $\cos\theta = \hat{\mathbf{v}} \cdot \mathbf{v}/(\hat{s}\, s)$. The squared unit-vector difference is therefore $2(1 - \cos\theta)$; since our implementation averages over the two tangent components rather than summing, the FASTNET directional term as computed equals the ANGLE+MAG directional term $(1 - \cos\theta)$ exactly, so the two objectives share the same directional geometry.

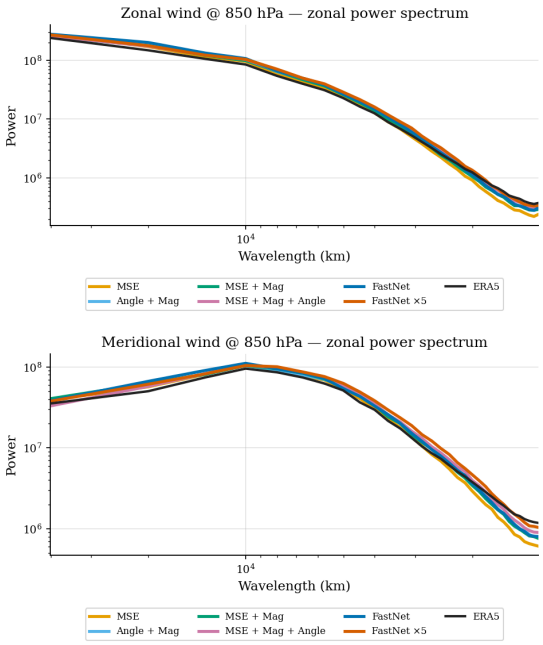


Figure H.2: Power spectra of the zonal wind u (top) and meridional wind v (bottom) at 850 hPa, at 72 h lead, with the ERA5 reference in black.

H.3 Experiment 4: Inductive Biases and Periodicity: RoPE and APE

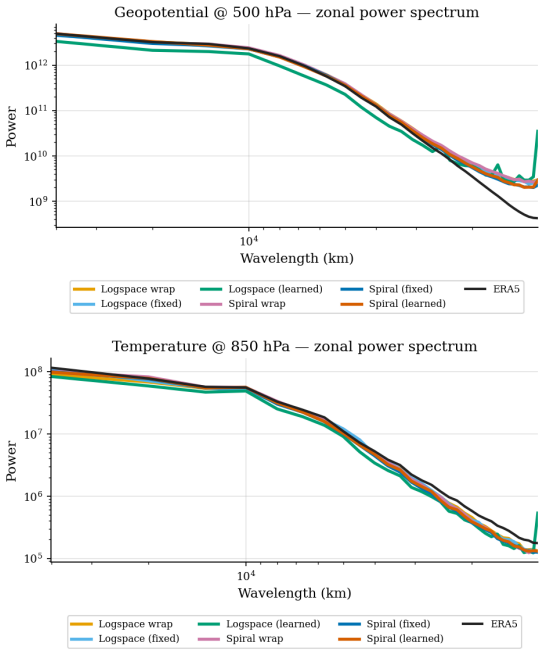


Figure H.3: APE zonal power spectra at lead 72 h for the six frequency-initialisation schemes: geopotential at 500 hPa (top) and temperature at 850 hPa (bottom); ERA5 (black) is the reference.

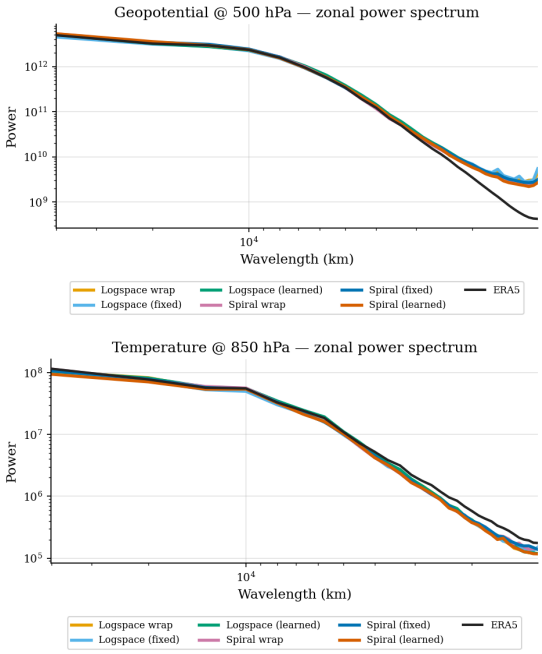


Figure H.4: RoPE zonal power spectra at lead 72 h, same layout as Figure H.3 (geopotential at 500 hPa, top; temperature at 850 hPa, bottom); ERA5 (black) is the reference.

H.4 Experiment 5: Equivariance and Weight Sharing in 3D

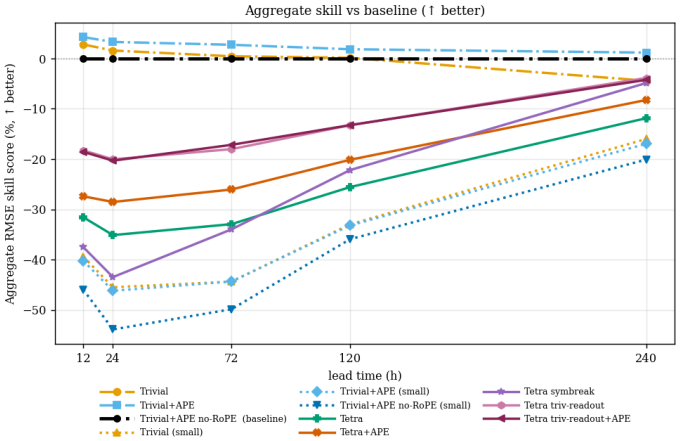


Figure H.5: Aggregated deterministic skill of the 3D symmetry comparison over the 10-day rollout, relative to the reference model configuration with APE only.

H.5 Experiment 5: Equivariance and Weight Sharing, 2D vs. 3D

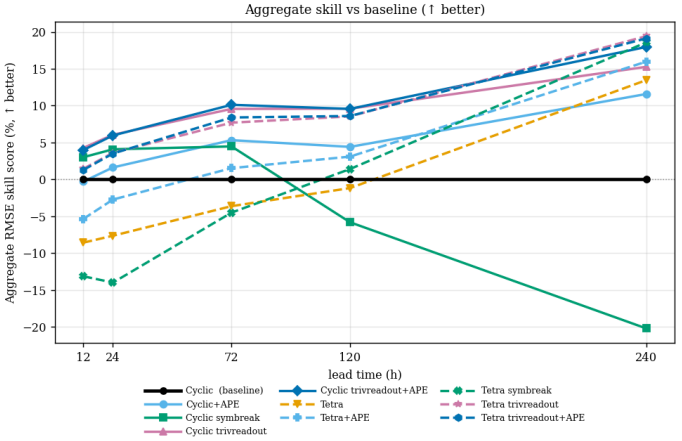


Figure H.7: Aggregated deterministic skill of the 2D Cyclic-4 vs. 3D tetrahedral symmetry over the 10-day rollout, relative to the reference model configuration with APE only.

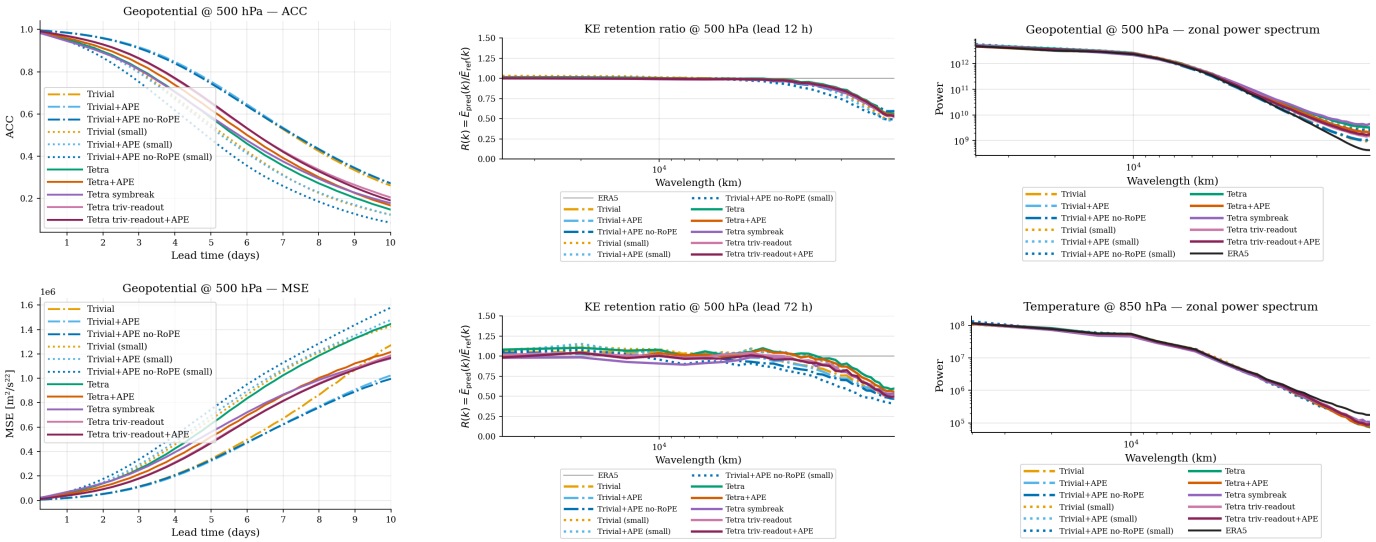


Figure H.6: 3D symmetry comparison summary at 500 hPa (WB2 headline level unless noted). **Left column:** deterministic skill (ACC top, MSE bottom) of geopotential. **Middle column:** kinetic-energy retention ratio (12 h top, 72 h bottom lead), $R{=}1$ matches ERA5. **Right column:** zonal power spectra at lead 72 h (geopotential top, temperature at 850 hPa bottom), ERA5 (black) is the reference.

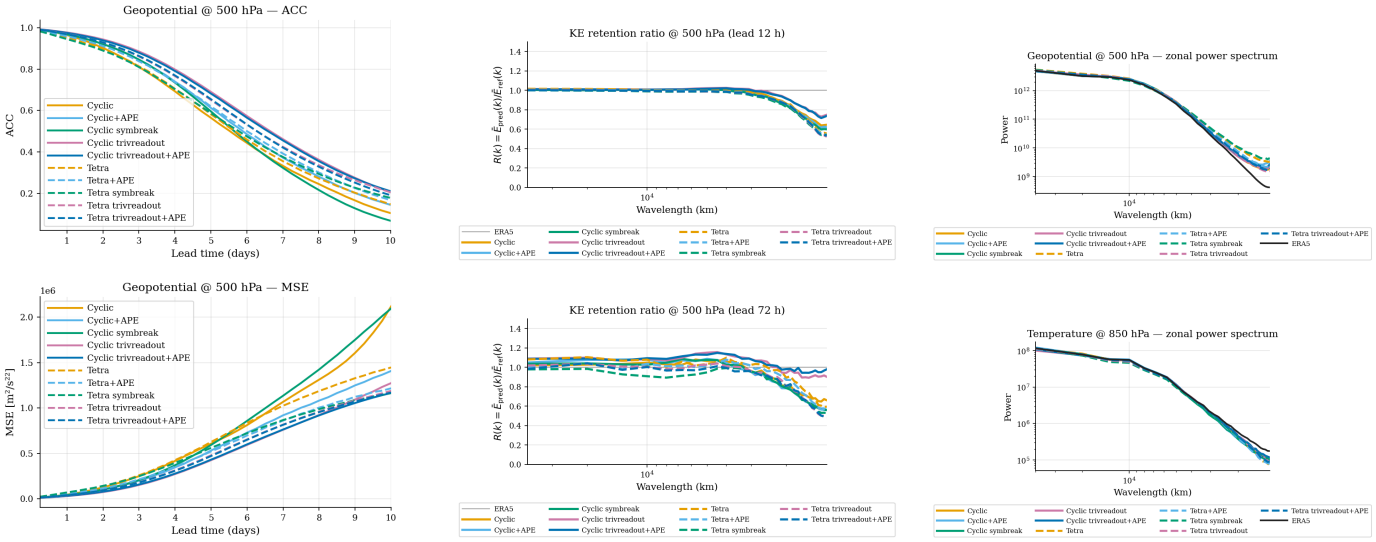


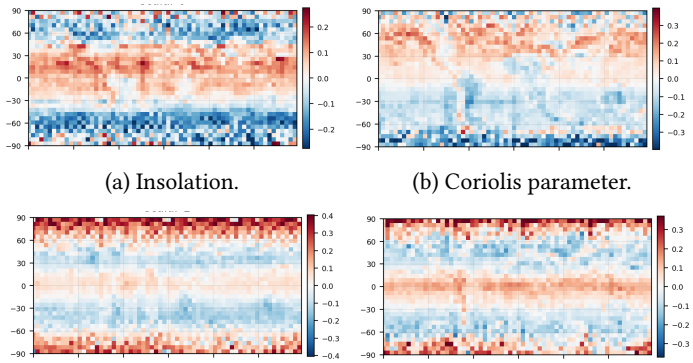
Figure H.8: 2D Cyclic-4 vs. 3D tetrahedral symmetry comparison summary at 500 hPa (WB2 headline level unless noted). **Left column:** deterministic skill (ACC top, MSE bottom) of geopotential. **Middle column:** kinetic-energy retention ratio (12 h top, 72 h bottom lead), $R{=}1$ matches ERA5. **Right column:** zonal power spectra at lead 72 h (geopotential top, temperature at 850 hPa bottom), ERA5 (black) is the reference.

Appendix I

Additional Results: Symmetry-Breaking Features

I.1 Experiment 6: Full Per-Run Breakdown of the Learned Scalar Channels

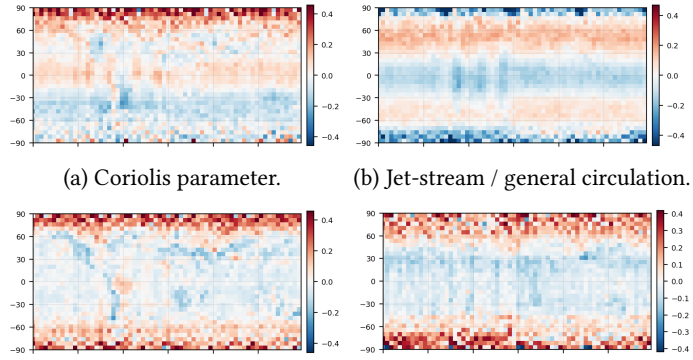
I.1.1 With static information



(a) Insolation. (b) Coriolis parameter.

(c) Jet-stream / general circulation. (d) Jet-stream / general circulation.

Figure I.1: Interpretable learned symmetry-breaking scalars of the **roto-translation equivariant, with static features** run.



(a) Coriolis parameter. (b) Jet-stream / general circulation.

(c) Insolation (with land artifacts). (d) Insolation.

Figure I.2: Interpretable learned symmetry-breaking scalars of the **roto-translation equivariant with trivial readout, with static features** run. The remaining channels (2, 4) are in Figure I.10.

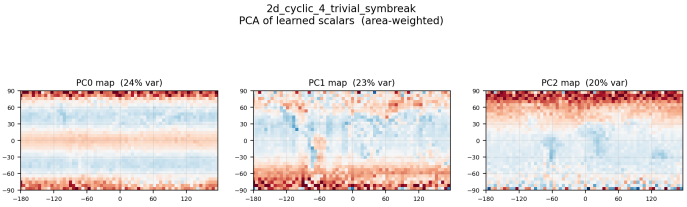
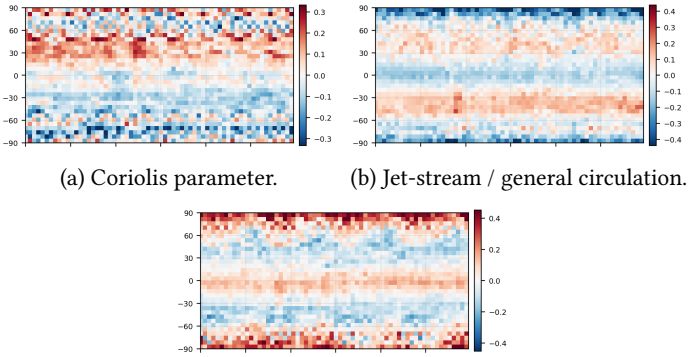


Figure I.3: Leading principal components (PC0–PC2) of the six learned scalar channels of the Cyclic-4 trivial-readout run with static features. PC0 carries the latitude-banded circulation/insolation belts, PC1 and PC2 together show insolation and orographic structure. As the modes carry similar variance (24/23/20%) and the latitude-banded templates are mutually correlated, the physical signals mix across components.

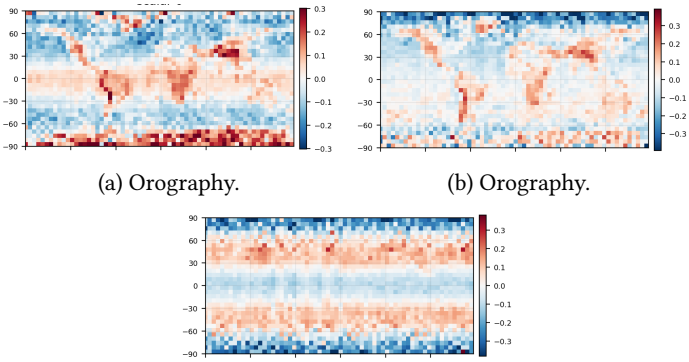


(a) Coriolis parameter. (b) Jet-stream / general circulation.

(c) Jet-stream / general circulation.

Figure I.4: Interpretable learned symmetry-breaking scalars of the **translation-only, with static features** run.

I.1.2 Without static information



(a) Orography. (b) Orography.

(c) Jet-stream / general circulation.

Figure I.5: Interpretable learned symmetry-breaking scalars of the **roto-translation equivariant, without static features** run.

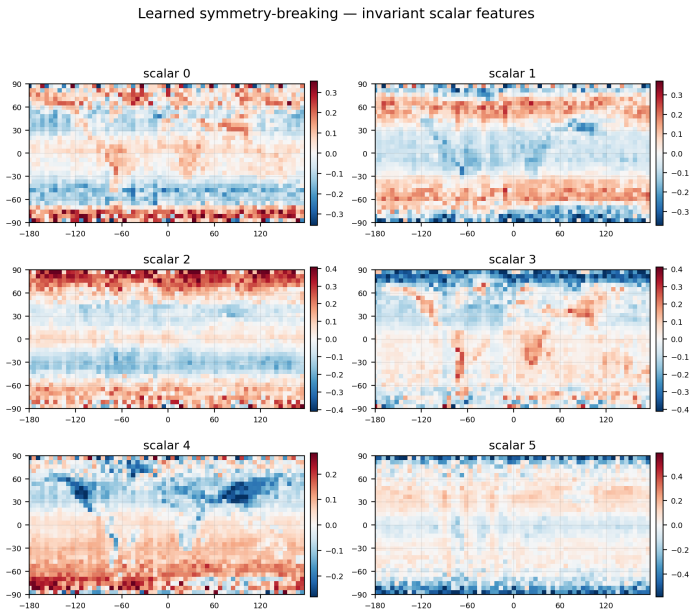


Figure I.6: Learned symmetry-breaking scalars of the **roto-translation equivariant with trivial readout, without static features** run.

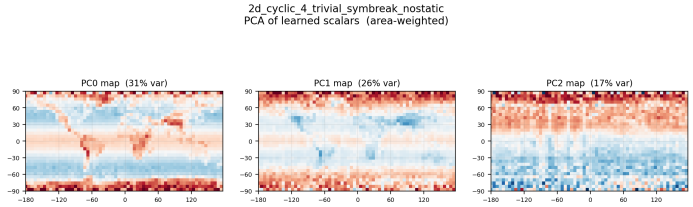
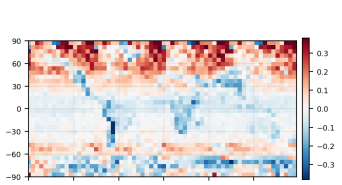
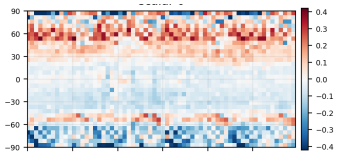


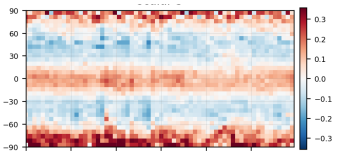
Figure I.7: PCA of the six learned scalar channels of the $T \rtimes C_4$ **trivial-readout, without static features** run. Leading three principal components (PC0–PC2), each with its share of explained variance.



(a) Orography.

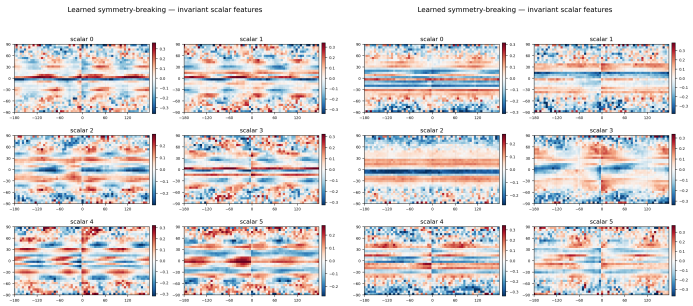


(b) Jet-stream / general circulation (tentative).



(c) Jet-stream / general circulation (tentative).

Figure I.8: Interpretable learned symmetry-breaking scalars of the **translation-only, without static features** run.

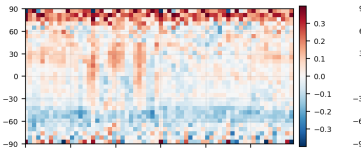


(a) With static features.

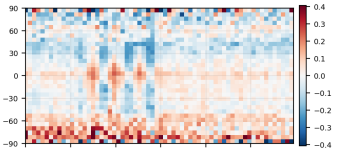
(b) Without static features.

Figure I.9: Learned symmetry-breaking scalars of the **non-equivariant vanilla-APE baseline**, with static features **(left)** and without static features **(right)**.

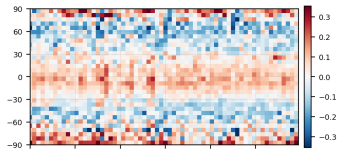
I.1.3 Non-interpretable learned scalar channels



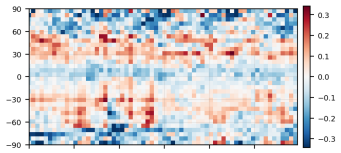
(a) $T \rtimes C_4$ trivial-readout, scalar 2.



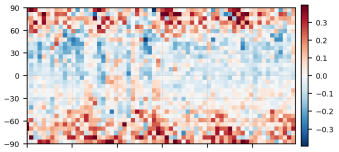
(b) $T \rtimes C_4$ trivial-readout, scalar 4.



(c) Translation-only, scalar 1.



(d) Translation-only, scalar 3.



(e) Translation-only, scalar 5.

Figure I.10: Non-interpretable/noisy scalar channels, both runs with static features. **Top row:** odd-pattern channels (2, 4) of the $T \rtimes C_4$ trivial-readout run. **Bottom two rows:** noisy channels (1, 3, 5) of the translation-only run.

I.1.4 Learned vector channel

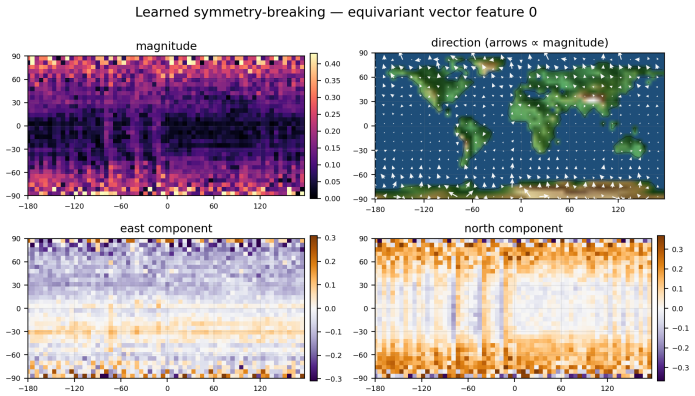


Figure I.11: Learned equivariant symmetry-breaking *vector* of the $T \rtimes C_4$ **trivial-readout** run, shown as magnitude **(top left)**, direction with arrows scaled by magnitude **(top right)**, and the east/north components **(bottom)**.

I.1.5 3D tetrahedral symmetry-breaking bias

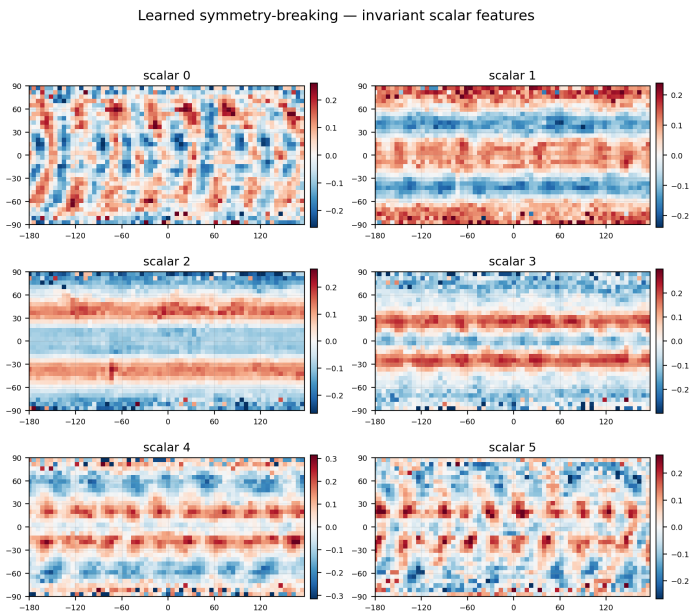


Figure I.12: Learned symmetry-breaking scalar channels of the **3D tetrahedral run**.

I.2 Experiment 7: Skill Without Static Features

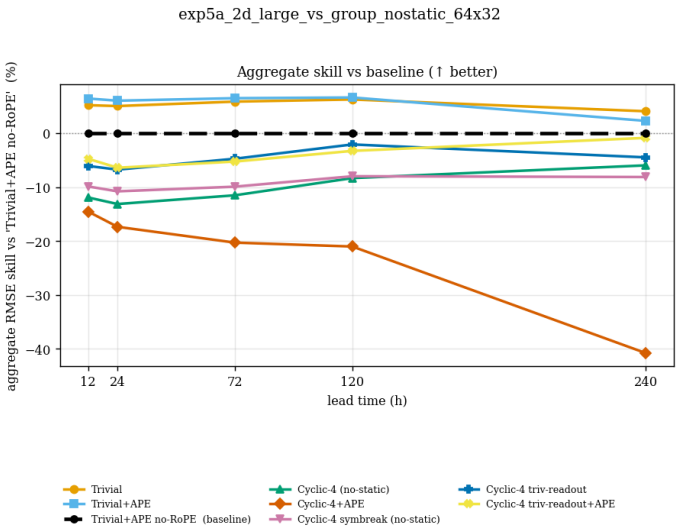


Figure I.13: Aggregated deterministic skill of the 2D symmetry comparison without static features. The ordering does not change from the with-static results in Figure 6.10.

Appendix J

What went wrong during experimentation

J.1 Unnormalized degree coordinates

Initially, positions reached the positional-encoding call sites in raw degrees ($|\text{pos}|$ up to $354°$, because of the $5.6°$ grid), The gradients here are calculated with high numbers, causing noisy updates). This impacted our setup in 3 places: pooling layers, APE and RoPE.

J.1.1 Pooling

We ablated the coordinate mode of the pooling position feature: degrees, radians, and an isotropic RMS-rescaled variant (per-channel z-score standardization is not an option here, as it breaks equivariance; see Section J.1.1). We left degrees in place for the ablation experiments and adopted the isotropic RMS-rescaled variant in the final setup: although raw degrees score marginally higher at long leads (Figure J.1), the RMS variant is the principled, scale-free choice.

The raw-degree positions enter the pooling learned linear layer (as the concatenated positional information). This was not fatal (Figure J.1): the layer can absorb the large input scale with proportionally smaller weights, and AdamW's gradient rescaling together with the learning-rate warmup let it settle there early in training rather than diverging.

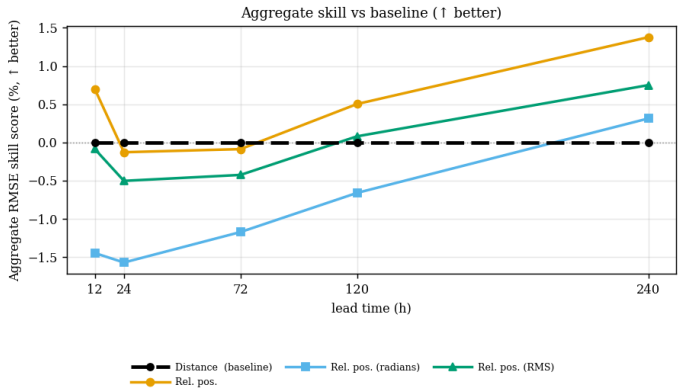


Figure J.1: Aggregated deterministic skill of the pooling coordinate-mode ablation over the 10-day rollout, relative to the distance baseline.

Equivariance-preserving offset normalization

Because the centered pooling offsets $\Delta_j = \mathbf{x}_j - \bar{\mathbf{x}}$ (with $\bar{\mathbf{x}}$ the mean of the s children, so $\sum_j \Delta_j = 0$ within each group) are passed through a learned linear layer, their *scale* matters but their *mean* is already zero, so only their magnitude needs adjusting. When the positions are not already standardized we rescale every offset by a single *detached, isotropic* RMS scalar,

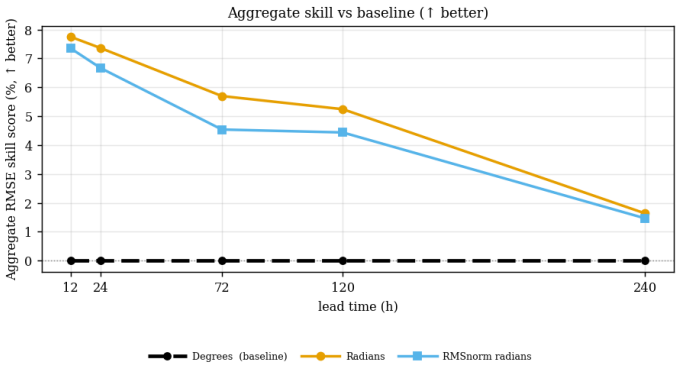
$$\Delta_j \leftarrow \Delta_j / s, \qquad s = \sqrt{\frac{1}{N\,D}\sum_{j,d}\Delta_{j,d}^2}\,, \tag{J.1}$$

where the average runs over all N offsets *and* their D spatial components, yielding one shared scalar applied uniformly to every component.

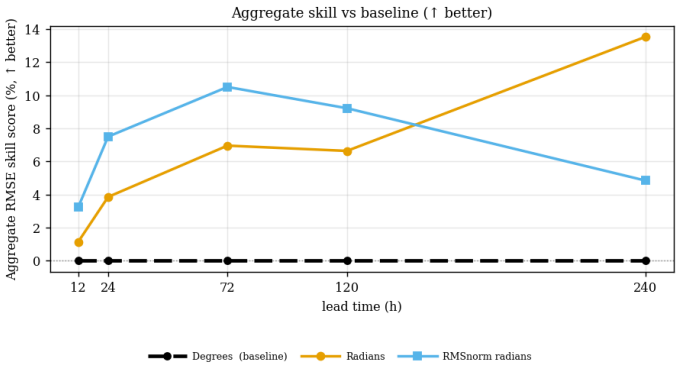
We deliberately avoid the alternative per-channel z-score normalization $\Delta_{j,d} \leftarrow (\Delta_{j,d} - \mu_d)/\sigma_d$, because it breaks equivariance. Dividing each axis by its own standard deviation σ_d is a diagonal map $\mathrm{diag}(1/\sigma_1, \ldots, 1/\sigma_D)$ — a *directional* rescaling that squashes space more along some axes than others. A diagonal matrix does not commute with a rotation $R \in \mathcal{G}$, since a rotation mixes the axes: rotate the geometry first and the per-axis variances are measured along the wrong directions, so "normalize-then-rotate" and "rotate-then-normalize" disagree. The isotropic RMS rescaling, by contrast, is a single positive scalar: it is rotation-invariant because $\|R\Delta_j\| = \|\Delta_j\|$, so it commutes with every frame and leaves the equivariance of the pooling intact while still normalizing the magnitude.

J.1.2 APE and RoPE

We ablated the same coordinate modes: degrees, radians, and an isotropic RMS-rescaled radians variant (z-score standardization again being ruled out by equivariance, Section J.1.1). Unnormalized degrees are far worse for APE than for RoPE: APE consumes the raw absolute coordinates directly, with no relative-difference cancellation to rescue it, whereas RoPE attention scores depend only on *relative* positions (within a ball at self-attention, or between the k nearest neighbours at cross-attention; Section B.1). Figure J.2 shows that RoPE was stable in the degree-based ablation experiments: the degree runs train stably and sit behind radians-based, a calibration gap rather than the noise catastrophe APE suffers. The final runs adopt radians-based RoPE as the scale-appropriate choice (even though marginally behind RMS radians at short lead), the ablation ran in degrees. APE had to be run in radians for both the ablations and the final runs.



(a) APE.



(b) RoPE.

Figure J.2: Aggregated deterministic skill of the coordinate-mode ablation over the 10-day rollout, relative to the degrees baseline, for APE (top) and RoPE (bottom).

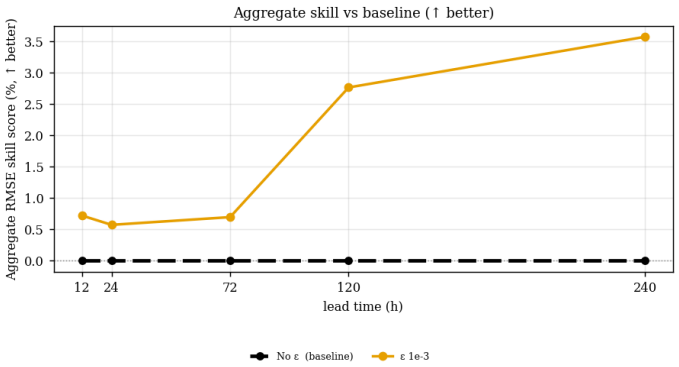


Figure J.3: Aggregated deterministic skill of the raised angular clamp (ε=10^{-3}) over the 10-day rollout, relative to the no-ε baseline, i.e. PyTorch's default ε=10^{-8}.

J.2 Low ε in the vector-loss angular component

The angle term is defined as $1 - \cos(\hat{\mathbf{v}}, \mathbf{v})$, with $\cos(\hat{\mathbf{v}}, \mathbf{v}) = \frac{\hat{\mathbf{v}} \cdot \mathbf{v}}{\|\hat{\mathbf{v}}\| \, \|\mathbf{v}\|}$. This spiked catastrophically on experimentation with another architecture and motivated raising the clamp from PyTorch's default ε=10^{-8}, as this term led to high gradient (and loss) spikes: it diverges at calm-wind pixels ($u \approx v \approx 0$, so $\|\hat{\mathbf{v}}\| \to 0$). The clamp sits in the cosine denominator as $\max(\|\hat{\mathbf{v}}\|, \varepsilon)$. A dedicated ε ablation, comparing PyTorch's default clamp (ε=10^{-8}) against a raised ε=10^{-3} (Figure J.3), was run after the fact. It confirms that ε=10^{-8} was valid for the experimentation phase: training is stable in this architecture, and we did not observe the catastrophic spikes seen in the other architecture. The raised ε=10^{-3} clamp is nonetheless the more performant setting (up to $+3.6\%$ aggregate skill at day 10) and was adopted for the final runs.