PACE-FNO: PHYSICS-ALIGNED CANONICAL EQUIVARIANCE FOR FOURIER NEURAL OPERATORS

Anonymous authors
Paper under double-blind review

ABSTRACT

Neural operators are often tested on states that differ physically from training data. A distinct failure occurs when the physical dynamics are unchanged but the observed coordinate frame differs from training. PACE-FNO addresses this case by estimating the frame, predicting after pulling the field to a canonical representative, and restoring the requested terminal frame. The default inference path uses one forward prediction; optional test-time adaptation (TTA) updates only the low-dimensional coordinate. On translated and Galilean-shifted Burgers and shallow-water systems, PACE-FNO lowers out-of-distribution (OOD) relative error by up to $12\times$ relative to a data-augmented Fourier Neural Operator (FNO+Aug). A matched-estimator control supports attributing the gain to prediction in the canonical frame rather than added estimator capacity. Our analysis separates the canonical approximation error from the two alignment residuals. We also test regional WeatherBench2 time-series forecasting with the fifth-generation ECMWF reanalysis (ERA5) under natural time OOD, where PACE-FNO and FNO achieve nearly identical performance. Experiments with approximate rotation, other backbones, irregular domains, rollouts, and image data delineate the conditions under which the mechanism is effective. The anonymous code repository is available at `https://anonymous.4open.science/r/pace-fno-C3B4`.

1 INTRODUCTION

Neural operators learn solution maps between function spaces, enabling fast predictions across varying inputs and parameters in applications such as fluid dynamics and weather forecasting (Karniadakis et al., 2021; Li et al., 2021; Kovachki et al., 2023; Pathak et al., 2022). A vortex translated across a periodic domain still follows the same equation, yet an operator trained around its original location may treat the translated field as OOD. Galilean-shifted shallow-water states and rotated fields have the same property: the dynamics are familiar, while the observed coordinate frame is new.

Data augmentation exposes a model to more transformed examples. Equivariant architectures build selected transformations into the predictor (Wang et al., 2020; Helwig et al., 2023). Both approaches improve coverage of selected transformations, but neither explicitly separates frame estimation from field evolution. When the action is known at test time, asking one predictor to infer the frame and evolve the field couples two tasks that fail differently under a novel coordinate frame.

PACE-FNO separates frame estimation from field prediction. It estimates a low-dimensional group coordinate, pulls the observation back to a canonical representative, applies an FNO, and restores the terminal frame. The terminal transformation is prescribed by the known dynamics rather than learned as an independent map. Consequently, the input and terminal transformations may differ. We train the estimator and operator jointly. During inference, one-shot prediction is the default; TTA refines the coordinate while keeping the learned weights fixed.

Contributions. We formulate PACE-FNO as a wrapper around a Fourier Neural Operator (FNO). The wrapper estimates a low-dimensional transformation coordinate, applies the inverse action to map the observed input to a canonical representative, predicts in that canonical frame, and applies the dynamics-prescribed terminal action to the output. Periodic translations are implemented spectrally. Under the exact input–terminal relation, compact canonical support, continuity, and Lipschitz conditions stated in Appendix D.7, we derive a conditional field-space error decomposition.

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

The bound separates canonical approximation error, residual input alignment error, and residual terminal alignment error. For approximate numerical actions, it includes an additional symmetry residual. The result does not guarantee recovery of a unique physical coordinate or provide a distribution-free out-of-distribution guarantee.

We evaluate the wrapper on exact-action partial differential equation benchmarks with periodic translation and Galilean shifts. PACE-FNO reduces out-of-distribution relative error by up to $12\times$ relative to a data-augmented Fourier Neural Operator, while a matched-estimator control retains the estimator and its supervision but removes the pullback and pushforward operations. We further evaluate approximate rotation, alternative predictor backbones, an irregular-domain Poisson problem, a 300-step Kuramoto–Sivashinsky rollout, image transformations, and WeatherBench2 time-series forecasting. These experiments measure portability of the prescribed alignment mechanism and also identify settings in which the supplied geometric structure does not provide a clear advantage.

Figure 1 illustrates this decomposition: an observed field is mapped to a canonical representative, evolved there, and returned to the requested terminal frame.

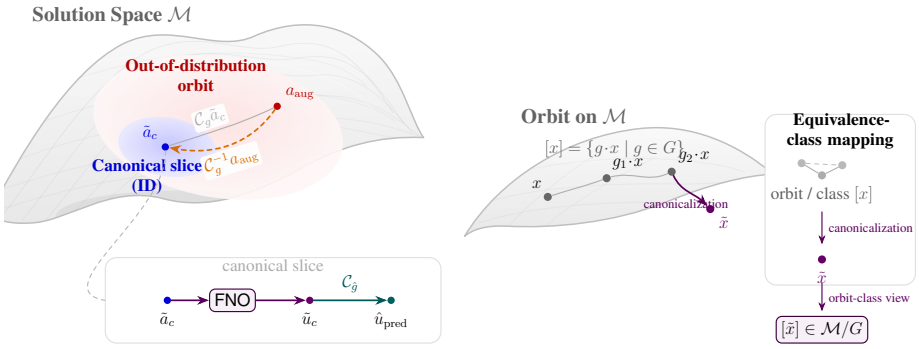


Figure 1: Geometric view of PACE-FNO. (a) The input is pulled from its observed orbit to a canonical slice, evolved there, and pushed forward to the target frame. (b) Points connected by the group action share an orbit $[x] = \{g \cdot x \mid g \in G\}$; canonicalization chooses one representative $\tilde{x}$ from each equivalence class.

2 Related Work

2.1 Neural Operators and Geometric Distribution Shift

DeepONet, Fourier Neural Operator (FNO), and later operator models learn solution maps directly from discretized functions or fields (Lu et al., 2021; Li et al., 2021; Kovachki et al., 2023; Wu et al., 2024; Wang et al., 2025). These methods address how the solution map is represented and evaluated on a discretized domain. They do not, by themselves, specify how an input should be transformed when the observed coordinate frame differs from the training frame. This distinction is relevant for periodic fields: a translated field can satisfy the same governing equation while appearing outside the canonical support used to train the predictor.

2.2 Equivariance and Data Augmentation

Data augmentation exposes a predictor to transformed training examples. Group-equivariant networks instead encode selected transformations in the model structure (Cohen & Welling, 2016; Finzi et al., 2020; Helwig et al., 2023; Wang et al., 2020), while Lie-group augmentation samples transformations during training (Brandstetter et al., 2022). Both approaches can improve performance on transformations represented by the training protocol. They do not require an explicit estimate of the frame of each test input. PACE-FNO separates this estimation step from field prediction and uses the prescribed action to map the input toward a canonical representative before the predictor is applied.

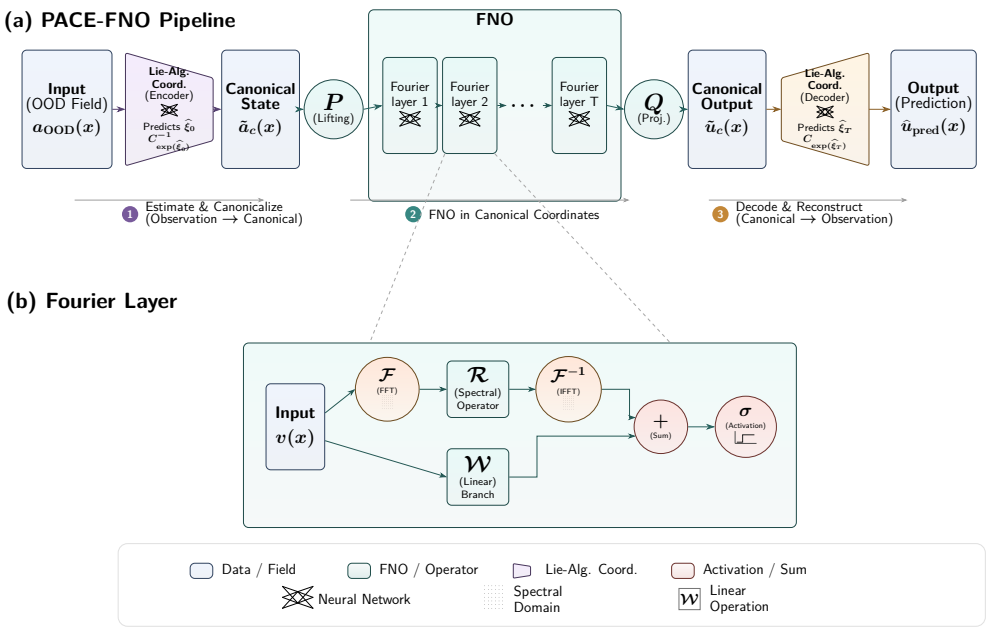


Figure 2: PACE-FNO architecture. (a) The Lie-algebra encoder estimates the input coordinate, the inverse action pulls the observation into a canonical frame, the FNO predicts there, and the terminal action restores the requested output frame. (b) A representative FNO layer retains its standard spectral and linear branches; PACE-FNO changes the surrounding coordinate path rather than the predictor architecture.

2.3 Canonicalization and Lie-group adaptation

Canonicalization methods construct a representative for an equivalence class induced by a transformation group. Frame averaging and learned canonicalization provide architecture-agnostic symmetry mechanisms, although a continuous global representative need not exist (Puny et al., 2022; Kaba et al., 2023; Ma et al., 2024; Dym et al., 2024). LieLAC is the closest prior work in spirit because it uses Lie-algebra canonicalization for neural operators under Lie-group transformations and evaluates PDE solvers (Shumaylov et al., 2025). PACE-FNO differs in jointly learning an amortized coordinate estimator with the operator, using one-shot canonicalization as the default inference path, and allowing the known dynamics to induce different input and terminal transformations.

2.4 Positioning of PACE-FNO

PACE-FNO is also related to geometry-aware and intrinsically equivariant operators. Geo-FNO learns a deformation from an irregular physical domain to a uniform latent space before applying Fourier layers (Li et al., 2023). Clifford Neural Layers encode geometric structure directly in multivector-valued neural layers and Fourier transforms (Brandstetter et al., 2023). The irregular-domain and backbone-transfer experiments in this paper do not reproduce these baselines under matched protocols. They instead evaluate whether the prescribed alignment wrapper can be transferred across predictors and domains.

Figure 2 illustrates the resulting architecture: PACE-FNO wraps a standard FNO with learned coordinate estimation and prescribed input/output actions.

3 PACE-FNO method

3.1 Problem formulation

Let $\mathcal{G}^\dagger : \mathcal{A} \to \mathcal{U}$ be a physical solution operator and let $\mathcal{G}_\theta$ be its learned approximation, trained by

$$\|\mathcal{G}^\dagger - \mathcal{G}_\theta\|^2_{L^2_\mu(\mathcal{A};\mathcal{U})} = \mathbb{E}_{a\sim\mu}\|\mathcal{G}^\dagger(a) - \mathcal{G}_\theta(a)\|^2_{\mathcal{U}}.$$

We take canonical inputs from μ_c and construct geometric OOD inputs as $a_{\text{OOD}} = \mathcal{C}_g(a_c)$ for $a_c \sim \mu_c$ and $g \sim P_G$. In the exact-action experiments, this changes the frame while preserving the viscosity,

forcing, solver, and boundary conditions. We write the action on fields as $[\mathcal{C}_g a](x) = a(g^{-1} \cdot x)$ and parameterize a local group element by $g = \exp(\xi)$, $\xi \in \mathfrak{g} = T_e G$.

When spatial and kinematic transformations are both present, we use the decomposition

$$\mathfrak{g} = \mathfrak{g}_{\mathcal{S}} \oplus \mathfrak{g}_{\mathcal{K}},$$

where $\mathfrak{g}_{\mathcal{S}}$ contains spatial frame variables and $\mathfrak{g}_{\mathcal{K}}$ contains problem-specific kinematic variables. This decomposition determines which coordinates are estimated, which are computed from the observed field, and which are refined by TTA.

3.2 Kinematic reduction and coordinate parameterization

The estimator acts on the geometric part of the frame after removing problem-specific kinematic components that can be obtained from the observed field. For a velocity field $\boldsymbol{v}$, we use its spatial mean as the background velocity,

$$\boldsymbol{v}_{\mathrm{bg}}(a) = \frac{1}{|D|} \int_D \boldsymbol{v}(x)\, \mathrm{d}x, \qquad \mathcal{S}_{\boldsymbol{v}}(a) = a - \boldsymbol{v}_{\mathrm{bg}}(a).$$

The estimator receives $\mathcal{S}_{\boldsymbol{v}}(a)$ and predicts the geometric coordinate. This separates learned spatial alignment from the kinematic quantity used to determine the terminal transport.

Let $k = \dim(\mathfrak{g}_{\mathcal{S}})$. The coordinate network produces $\mathcal{M}_\phi(a) \in \mathbb{R}^{2k}$. For the jth generator basis element B_j, the two corresponding outputs are mapped to the bounded coefficient

$$\tau_j(z) = \tau_{j,\max} \frac{z_{2j}}{\sqrt{z_{2j-1}^2 + z_{2j}^2 + \varepsilon_{\mathrm{ret}}^2}}, \qquad \varepsilon_{\mathrm{ret}} = 10^{-6},$$

and the spatial estimate is $\hat{\xi}_{\mathcal{S}} = \sum_j \tau_j(\mathcal{M}_\phi(\mathcal{S}_{\boldsymbol{v}}(a)))B_j$. The full estimate combines this learned coordinate with the analytic kinematic component. The retraction enforces the support used by the training and OOD protocols; it does not provide a coordinate identifiability guarantee.

3.3 Canonicalization and terminal transport

Given a_{OOD}, the estimator $\mathcal{E}_\phi$ predicts an input coordinate $\hat{\xi}_0$. The inverse action maps the observation toward the canonical training frame,

$$\tilde{a} = \mathcal{C}^{-1}_{\exp(\hat{\xi}_0)}(a_{\mathrm{OOD}}).$$

The predictor operates on $\tilde{a}$ rather than directly on the observed field. Problem-specific kinematic components, when needed, are computed from the observed field. The terminal coordinate is $\hat{\xi}_T = \Gamma_B(\hat{\xi}_0; T)$, giving

$$\hat{u}_{\mathrm{pred}} = \mathcal{C}_{\exp(\hat{\xi}_T)} \circ \mathcal{G}_\theta \circ \mathcal{C}^{-1}_{\exp(\hat{\xi}_0)}(a_{\mathrm{OOD}}).$$

The input and terminal transformations may differ because the terminal coordinate is prescribed by the known dynamics rather than learned as an independent map. Periodic translations use the Fourier shift theorem. The approximate rotation experiment uses a three-shear construction. These choices keep the numerical action explicit and avoid interpolation in the periodic translation operation.

3.4 Transformation-coordinate parameterization

The estimator output is mapped to the task-specific support before it is used to construct a group element. Rotational variables use the two-channel parameterization above, which avoids a singular representation at the origin. The resulting coordinate is a transformation parameter, not a direct prediction of the target field. The task-specific generator sources and support bounds are listed in Appendix A.

In the Burgers and shallow-water settings, the observed background velocity determines the terminal transport over the prediction horizon. The spatial coordinate is estimated from the input, while the corresponding terminal displacement is obtained from the known evolution relation. This construction allows PACE-FNO to use different input and terminal actions while keeping the predictor in the canonical frame.

Algorithm 1 PACE-FNO training and inference

Input: Training pairs (x, y), predictor $\mathcal{G}_\theta$, estimator $\mathcal{E}_\phi$, group action $\mathcal{C}$, and loss weights.
Output: Trained (θ, ϕ) and one-shot or TTA prediction.

1: **for** each training step **do**
2: Sample (x, y) and a bounded training generator Δ_0.
3: Estimate $\hat{\xi}_0 = \mathcal{E}_\phi(x)$ and compute $\hat{\xi}_T = \Gamma_B(\hat{\xi}_0; T)$.
4: Form $x_c = \mathcal{C}^{-1}_{\exp(\hat{\xi}_0)} x$, predict $\hat{u}_c = \mathcal{G}_\theta(x_c)$, and restore $\hat{y} = \mathcal{C}_{\exp(\hat{\xi}_T)} \hat{u}_c$.
5: Evaluate the PDE loss on $(\hat{y}, y)$, form $x^+ = \mathcal{C}_{\exp(\Delta_0)} x$, and estimate $\hat{\xi}_0^+ = \mathcal{E}_\phi(x^+)$.
6: Evaluate the equivariance and anchor losses, then update (θ, ϕ) with the weighted sum of all terms.
7: **end for**
8: **One-shot:** apply the estimator, inverse input action, predictor, and terminal action once.
9: **Optional TTA:** freeze both networks, refine only the spatial coordinate using $\mathcal{J}_{\mathrm{TTA}}$, and recompute $\hat{\xi}_T$.

3.5 Training objective

The estimator and operator are trained jointly. The total loss is

$$\mathcal{L}_{\mathrm{total}} = \mathcal{L}_{\mathrm{PDE}} + \lambda_1 \mathcal{L}_{\mathrm{equiv}} + \lambda_2 \mathcal{L}_{\mathrm{anchor}}.$$

The PDE term supervises the predicted field. The equivariance term compares predictions under the prescribed action, and the anchor term constrains the estimator on canonical inputs. The complete loss definitions and task-specific weights are given in Appendix A. The matched control used in the experiments retains the estimator, its supervision, and the same parameter budget, but removes the input pullback and terminal pushforward during prediction.

For a clean pair (x, y) and an independently perturbed input $x^+ = \mathcal{C}_{\exp(\Delta_0)} x$, the three terms have distinct roles. The PDE term compares the restored prediction with y in field space. The equivariance term matches the relative estimator response to Δ_0, using wrapped coordinate differences for commuting translations and a local group difference for the rotation–translation block. The anchor term penalizes the spatial coordinate predicted for a canonical input. This separation keeps the physical prediction loss on the clean pair while using the perturbed input only for the geometric consistency constraint.

3.6 One-shot inference and test-time adaptation

One-shot inference uses a single estimator pass, one inverse input action, one predictor evaluation, and one terminal action. The learned weights remain fixed. Optional test-time adaptation (TTA) refines only the spatial coordinate by minimizing canonical self-consistency,

$$\mathcal{J}_{\mathrm{TTA}}(\hat{\xi}_{\mathcal{S}}) = \left\| \mathcal{P}_{\mathcal{S}} \circ \mathcal{E}_\phi\Big(\mathcal{C}^{-1}_{\exp(\hat{\xi}_{\mathcal{S}})}(a_{\mathrm{OOD}}) \Big) \right\|^2_{\mathfrak{g}},$$

where $\mathcal{P}_{\mathcal{S}}$ selects spatial generator components. TTA updates a low-dimensional coordinate while keeping both the estimator and predictor weights fixed. After each coordinate update, the terminal coordinate is recomputed from the known dynamics. The batch objective and optimization details are given in Appendix A.

The resulting procedure is therefore: estimate the input coordinate, compute the analytic kinematic component when the task provides one, apply the inverse input action, evaluate the predictor once, and restore the terminal frame. Optional TTA repeats only the coordinate refinement step and reuses the same predictor weights. The task-specific actions, loss coefficients, and timing conventions are given in Appendix B; the complete three-seed results are in Appendix C.

4 Theory

4.1 Conditional field-space error decomposition

Under an exact input–terminal relation, the prediction error can be separated into approximation error on the canonical support, residual input alignment error, and residual terminal alignment error. The output-action Lipschitz constant scales the canonical approximation term. For an approximate physical action, the bound includes an additional symmetry residual.

Theorem 4.1 (Error decomposition, abbreviated)**.** *Assume compact canonical support, a continuous exact input–terminal action, and the Lipschitz conditions stated in Appendix D.2. If the chosen FNO has canonical approximation error at most ϵ, then*

$$\sup_{a\in\mathcal{A}_{\mathrm{OOD}}} \|\mathcal{C}_{\hat{g}_T}\mathcal{G}_\theta\mathcal{C}_{\hat{g}_0}^{-1}(a) - \mathcal{G}^\dagger(a)\|_{\mathcal{U}} \le K_C\epsilon + K_C L_\theta \epsilon_{\mathrm{geo},0} + \epsilon_{\mathrm{geo},T}.$$

Here $\epsilon_{\mathrm{geo},0}$ and $\epsilon_{\mathrm{geo},T}$ are field-space residuals from input and terminal alignment. For an isometric output action $K_C = 1$; an approximate physical action contributes an additional symmetry residual. The complete statement and proof are in Appendix D.7.

The bound is stated in field space. It does not imply that the estimator recovers a unique generating coordinate, because symmetry-equivalent coordinates can produce the same canonical field. It also does not provide a distribution-free guarantee for arbitrary distribution shifts. The empirical target is therefore the reduction of the two field-space alignment residuals while preserving approximation on canonical support.

4.2 Local stability of test-time adaptation

The TTA objective has a local stability property under the conditions stated below.

Proposition 4.2 (Local stability of TTA)**.** *Let $\mathcal{J}_{\mathrm{TTA}}$ be differentiable near its zero set $\Xi^* = \{\xi : \mathcal{J}_{\mathrm{TTA}}(\xi) = 0\}$. If its gradient is L-Lipschitz there and, for some $\alpha > 0$, $\mathcal{J}_{\mathrm{TTA}}(\xi) \ge \alpha\,\mathrm{dist}(\xi,\Xi^*)^2$, and if the initial sublevel set is compactly contained in that neighborhood, then gradient flow $\dot{\xi} = -\nabla\mathcal{J}_{\mathrm{TTA}}(\xi)$ satisfies*

$$\frac{\mathrm{d}}{\mathrm{d}t}\mathcal{J}_{\mathrm{TTA}}(\xi(t)) = -\|\nabla\mathcal{J}_{\mathrm{TTA}}(\xi(t))\|_{\mathfrak{g}}^2 \le 0, \qquad \mathrm{dist}(\xi(t),\Xi^*) \le \alpha^{-1/2}\mathcal{J}_{\mathrm{TTA}}(\xi(t))^{1/2}.$$

The complete hypotheses and proof are given in Appendix D.9.

This result is local. It does not claim global convergence for arbitrary OOD inputs or recovery of a unique physical frame. Symmetry-related minimizers may form a manifold.

4.3 Scope of the theoretical statements

The exact input–terminal relation is used for the periodic translation and Galilean experiments. The arbitrary-angle Navier–Stokes rotation is treated as an approximate numerical action and therefore carries a separate symmetry residual. The theory does not assume that the transformation support used for OOD evaluation is itself a subgroup.

5 Experiments

5.1 Experimental protocol

We evaluate periodic 1-D and 2-D Burgers, shallow-water (SWE), and Navier–Stokes (NS) systems under transformation ranges excluded from training. We report in-distribution (ID) and out-of-distribution (OOD) results. Fourier Neural Operator (FNO), data-augmented FNO (FNO+Aug), and group-equivariant Fourier neural operator (GFNO) follow the reported protocols (Li et al., 2021; Takamoto et al., 2022; Li et al., 2022; Wang et al., 2020; Helwig et al., 2023). Each learned result averages seeds $\{42, 1234, 3407\}$ with identical data splits per method.

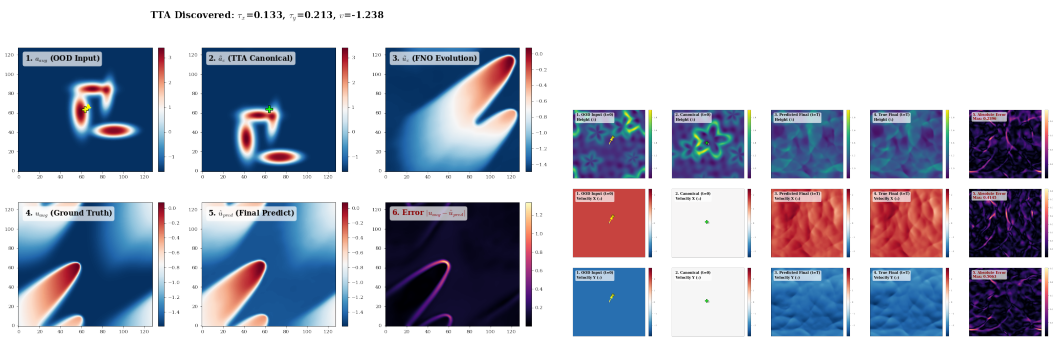
The suite covers exact-action PDE shifts, an approximate rotation diagnostic, backbone and domain transfer, mechanism controls, vision shifts, and WeatherBench2 time OOD. The protocols and measurement conventions are in Appendix B; complete tables and diagnostic figures are in Appendix C.

5.2 Exact-action PDE benchmarks

The periodic tests use cross-resolution OOD evaluation ($1024 \to 2048$ in 1-D Burgers and $64^2 \to 128^2$ otherwise); full equations, transformation ranges, hyperparameters, runtimes, and uncertainty tables

Table 1: Primary geometric-OOD results (mean ± standard deviation over three seeds). One-shot PACE-FNO is the default operating point. The NS rotation and translation row includes an approximate numerical rotation; all other rows use the stated exact continuum actions.

Problem and action	FNO	FNO+Aug	PACE-FNO (one-shot)	PACE-FNO + TTA
1-D Burgers, translation + Galilean	1.0352 ± 0.0001	0.3222 ± 0.0077	0.0267 ± 0.0014	$\mathbf{0.0259 \pm 0.0007}$
2-D Burgers, translation + Galilean	0.6137 ± 0.0362	0.2035 ± 0.0423	0.0755 ± 0.0139	$\mathbf{0.0640 \pm 0.0012}$
2-D SWE, Galilean	0.7260 ± 0.0003	0.5136 ± 0.0010	$\mathbf{0.1069 \pm 0.0168}$	0.1071 ± 0.0168
NS, translation + approximate rotation	0.6422 ± 0.0302	0.2840 ± 0.2001	0.2766 ± 0.0353	$\mathbf{0.2593 \pm 0.0522}$



(a) 2-D Burgers under translation and Galilean drift. (b) SWE under Galilean perturbation.

Figure 3: Representative exact-action PDE predictions. Full method comparisons and generation protocols are retained in Appendices B and C.

are in Appendices B and C. Table 1 collects the primary evidence. Its first three rows use exact periodic translation and Galilean actions. The NS rotation–translation row uses an approximate arbitrary-angle numerical rotation, so it is a diagnostic evaluation rather than an exact-equivariance guarantee.

Unless stated otherwise, the primary metric is relative field error,

$$\mathrm{RelErr}(\hat{u}, u) = \frac{\|\hat{u} - u\|_{L^2}}{\|u\|_{L^2}},$$

which measures the error after returning the prediction to the requested output frame.

On the exact-action benchmarks, one-shot PACE-FNO reduces OOD error from 0.3222 to 0.0267 on 1-D Burgers, from 0.2035 to 0.0755 on 2-D Burgers, and from 0.5136 to 0.1069 on SWE relative to FNO+Aug. TTA provides a smaller Burgers improvement at higher inference cost and leaves the SWE value unchanged at the reported precision. The NS change from 0.2840 ± 0.2001 to 0.2766 ± 0.0353 is small relative to the baseline spread, so we treat it as a diagnostic rather than a headline improvement.

On the same 1-D Burgers split, a dedicated LieLAC+FNO run gives 0.0342 ± 0.0119 OOD error and 8.73 s inference, versus 0.0267 ± 0.0014 and 0.53 s for one-shot PACE-FNO (Table 9). The comparison contrasts amortized one-shot inference with per-instance iterative optimization; timing conventions are documented in the table caption.

The OOD reduction is accompanied by higher ID error on Burgers: 0.0623 versus 0.0293 on 1-D Burgers and 0.0393 versus 0.0261 on 2-D Burgers. On SWE and NS, one-shot PACE-FNO has lower ID error than FNO+Aug. The complete ID/OOD tables are in Appendix C.

5.3 Controls and Adaptation Diagnostics

The matched control isolates the mechanism. It uses the same FNO, the same $297{,}156$-parameter estimator, and the same frame supervision as PACE-FNO, but removes the pullback and pushforward. On standard 2-D Burgers OOD it attains 0.66021 ± 0.03644 relative error, compared with 0.05863 ± 0.00287 for PACE-FNO. The comparison tests whether the pullback and pushforward, rather than added estimator capacity, create the canonical support used by the predictor.

Figure 11 illustrates the low-dimensional TTA objective when one-shot alignment leaves a residual frame error. The objective has symmetry-induced low-energy basins. Its non-convexity also explains why the refinement has a latency cost and can be sensitive to initialization.

Noise and sparse observations test the same mechanism without changing the predictor. With $\alpha = 0.10$ observation noise, PACE-FNO reports 0.05867 ± 0.00291, while the matched auxiliary-estimator control reports 0.66016 ± 0.03650. At 50% randomly retained observations, their errors are 0.2397 ± 0.0031 and 0.6622 ± 0.0413. Structured missingness raises PACE-FNO to 0.3244 ± 0.0016, still below FNO+Aug at 0.3560 ± 0.0119. These checks do not claim that canonicalization resolves arbitrary sensing loss.

Figures 11 and 14 in the appendix show the symmetry-induced TTA basins and the nearly unchanged SWE response over the tested learning rates and step budgets.

5.4 Backbone and domain transfer

We retain the alignment procedure and replace only the PDE predictor. On 2-D Burgers, PACE-MambaGalerkin reduces OOD error from 0.32505 ± 0.11714 to 0.04246 ± 0.00048; on SWE, PACE-Transolver reduces it from 0.78036 ± 0.06026 to 0.33906 ± 0.00456. These replacements use the Mamba Neural Operator, Transolver, and Continuous Vision Transformer (CViT) as the underlying predictor families (Cheng et al., 2026; Wu et al., 2024; Wang et al., 2025); CViT improves on average, although its seed variance prevents a precise ranking. The Mamba comparison includes a data-augmented baseline, whereas CViT+Aug and Transolver+Aug were not run; those two rows therefore measure portability relative to the listed base models, not a fully augmentation-matched wrapper effect.

The appendix also reports runtimes for these one-shot transfer settings. PACE-MambaGalerkin uses 22.876 ms per OOD sample and adds 11.4% training time relative to Mamba. PACE-Transolver uses 18.061 ms per sample, versus 20.064 ms for Transolver, while increasing training time by 53.4%. The corresponding parameter increases are 6.12% and 2.65%, respectively (Table 13).

We also evaluate the wrapper on an irregular unit-ball Poisson task, where PACE-Transolver reduces 90° rotation OOD error from 0.3000 ± 0.0470 to 0.1815 ± 0.0178. In a 300-step closed-loop Kuramoto–Sivashinsky rollout, PACE-FNO reaches 0.35919 ± 0.07196 versus 0.46978 ± 0.02451 for FNO+Aug. Table 2 collects these results, which support portability of the prescribed wrapper rather than discovery of an unknown symmetry.

Table 2: Additional scope checks under their fixed protocols (three seeds). These rows test portability and protocol-specific limitations, not the primary exact-action claim. Resource cells report Reference/PACE-FNO in that order; peak memory is allocated CUDA memory in MiB from the same RTX 4080 runs.

Setting	Reference predictor	Reference OOD	PACE-FNO OOD	Total Params (Ref./PACE)	Peak Train MiB (Ref./PACE)	Peak Infer MiB (Ref./PACE)
2-D Burgers	MambaGalerkin+Aug	0.32505 ± 0.11714	$\mathbf{0.04246 \pm 0.00048}$	4,854,177/5,151,333	3108.2/3124.8	3270.4/3221.7
2-D SWE	Transolver	0.78036 ± 0.06026	$\mathbf{0.33906 \pm 0.00456}$	11,229,251/11,526,983	967.4 ± 44.4/1034.8 ± 46.3	1489.6/1501.4
3-D Poisson-ball (90°)	Transolver	0.3000 ± 0.0470	$\mathbf{0.1815 \pm 0.0178}$	2,808,129/3,068,676	644.8/701.8	94.6/96.7
1-D KS (300 steps)	FNO+Aug	0.46978 ± 0.02451	$\mathbf{0.35919 \pm 0.07196}$	21,334,273/21,359,363	667.4 ± 215.3/532.3 ± 94.1	134.53/135.12

For the Navier–Stokes rotation diagnostic, the action itself is approximate. The corresponding residual is

$$\epsilon_{\mathrm{sym}} = \sup_{a \in \mathcal{A}_{\mathrm{OOD}}} \left\| \mathcal{G}^\dagger(a) - \mathcal{C}_{g_T(a)} \mathcal{G}^\dagger(a_c(a)) \right\|_{\mathcal{U}},$$

which adds to the exact-action bound. The residual keeps numerical action error separate from estimator and predictor error.

Figure 4 visualizes this approximate-action diagnostic; the residual is a property of the numerical transformation, not an unreported change to the PDE task.

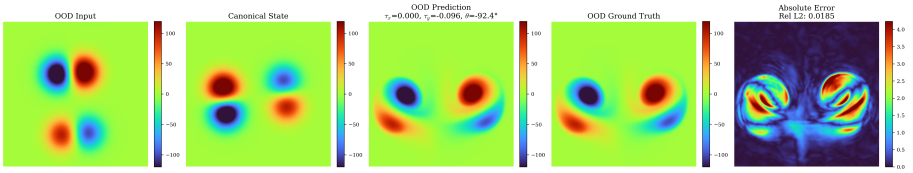
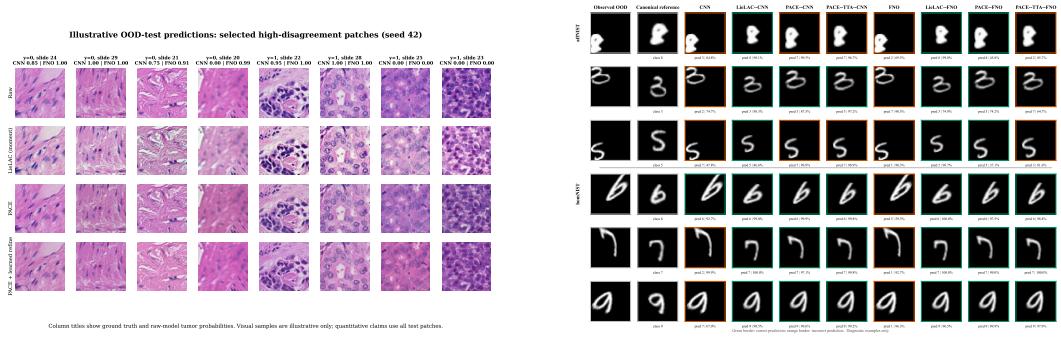


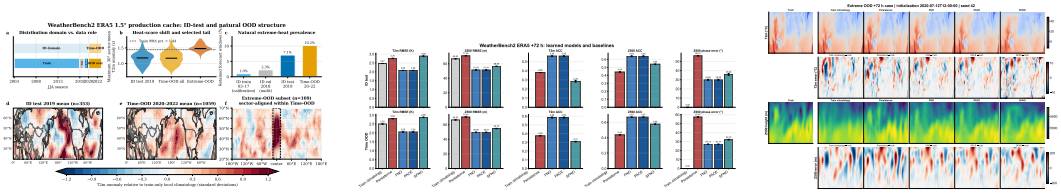
Figure 4: Navier-Stokes rotation and translation diagnostic. Translation is exact; arbitrary-angle rotation is the stated approximate three-shear numerical action and therefore carries a symmetry residual.



(a) Camelyon17 hospital-2 patches with high method disagreement.

(b) Selected affNIST and homNIST examples.

Figure 5: Qualitative vision diagnostics. Quantitative claims use the complete test sets.



(a) Data roles and natural-OOD audit.

(b) +72-h ID and time-OOD metrics.

(c) Fixed extreme-OOD fields.

Figure 6: WeatherBench2 diagnostics. The complete metrics and efficiency measurements are reported in Appendix C.

5.5 Image and WeatherBench2 evaluations

5.5.1 Image transformations

Camelyon17 uses a six-dimensional H&E stain/color transformation. affNIST and homNIST use affine and projective spatial warps. The figures are fixed diagnostics; quantitative claims use the complete test sets.

On Camelyon17 hospital 2, learned TTA reaches 83.31% OOD accuracy, 12.36 points above raw FNO and 2.68 points above learned LieLAC, at 0.597 rather than 21.854 ms per patch. The seed deviations preclude a significance claim for these point differences. One-shot PACE-FNO reaches 89.953% on affNIST and 97.180% on homNIST, while LieLAC reaches 95.503% and 98.433% at much higher latency. Complete results are in Tables 16 and 17.

5.5.2 WeatherBench2 boundary case

WeatherBench2 tests natural time OOD rather than a prescribed geometric action. The audit contains 353 ID-test, 1,059 time-OOD, and 109 pre-defined extreme-OOD windows; Figure 6a summarizes these roles.

At direct $+72$-h lead, FNO and PACE-FNO differ by less than 0.003 on the aggregate metrics. PACE-FNO processes 600.8 samples/s with 1.59 GiB peak GPU memory, versus 669.0 samples/s and 1.39 GiB for FNO; SFNO reaches 919.1 samples/s with 0.62 GiB. The full measurements are in Tables 18 and 19. These results do not support a universal claim for temporal distribution shift.

The fixed extreme-OOD case shows similar large-scale T2m and Z500 patterns, with local residual differences (Figure 6c).

6 Conclusion

PACE-FNO canonicalizes the input, predicts in the canonical frame, and restores the requested output frame through known dynamics. The theory and experiments support this design when the transformation action is explicit and evaluable in field space.

Reproducibility statement

The main text states the evaluation protocol and claim boundaries. The appendix provides the governing equations, transformation supports, dataset splits, three-seed convention, hyperparameters, hardware and runtime details, complete tables, and proofs. The anonymous supplementary material contains the associated implementation and diagnostic artifacts needed to inspect the reported results.

AI use statement

We used generative AI tools for editorial assistance, including revising the manuscript structure and prose, figure selection, layout, and captions, clarifying the scope of existing results, checking LaTeX consistency, and reviewing the mathematical arguments. The tools did not run experiments or create experimental data, numerical results, or citations. The authors independently reviewed the final text, claims, equations, proofs, figures, tables, citations, and cross-references and take responsibility for the submission.

References

Ronald N. Bracewell. *The Fourier Transform and Its Applications*. McGraw-Hill, New York, 1986.

Johannes Brandstetter, Max Welling, and Daniel E. Worrall. Lie point symmetry data augmentation for neural PDE solvers. In *International Conference on Machine Learning*, pp. 2241–2256. PMLR, 2022.

Johannes Brandstetter, Rianne van den Berg, Max Welling, and Jayesh K. Gupta. Clifford neural layers for PDE modeling. In *International Conference on Learning Representations*, 2023.

Chun-Wun Cheng, Jiahao Huang, Yi Zhang, Guang Yang, Carola-Bibiane Schönlieb, and Angelica I. Avilés-Rivero. Mamba neural operator: Who wins? transformers vs. state-space models for PDEs. *Journal of Computational Physics*, 548:114567, 2026. doi: 10.1016/j.jcp.2025.114567.

Taco Cohen and Max Welling. Group equivariant convolutional networks. In *International Conference on Machine Learning*, pp. 2990–2999. PMLR, 2016.

Nadav Dym, Hannah Lawrence, and Jonathan W. Siegel. Equivariant frames and the impossibility of continuous canonicalization. *arXiv preprint arXiv:2402.16077*, 2024.

Luca Falorsi, Pim de Haan, Tim R. Davidson, Nicola De Cao, Maurice Weiler, Patrick Forré, and Taco S. Cohen. Explorations in homeomorphic variational auto-encoding. *arXiv preprint arXiv:1807.04689*, 2018.

Marc Finzi, Samuel Stanton, Pavel Izmailov, and Andrew Gordon Wilson. Generalizing convolutional neural networks for equivariance to Lie groups on arbitrary continuous data. In *International Conference on Machine Learning*, pp. 3165–3176. PMLR, 2020.

Jacob Helwig, Xuan Zhang, Cong Fu, Jerry Kurtin, Stephan Wojtowytsch, and Shuiwang Ji. Group equivariant Fourier neural operators for partial differential equations. In *International Conference on Machine Learning*, pp. 13376–13411. PMLR, 2023.

Sékou-Oumar Kaba, Arnab Kumar Mondal, Yan Zhang, Yoshua Bengio, and Siamak Ravanbakhsh. Equivariance with learned canonicalization functions. In *International Conference on Machine Learning*, pp. 15546–15566. PMLR, 2023.

Hamed Karimi, Julie Nutini, and Mark Schmidt. Linear convergence of gradient and proximal-gradient methods under the Polyak-Łojasiewicz condition. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pp. 795–811. Springer, 2016.

George Em Karniadakis, Ioannis G. Kevrekidis, Lu Lu, Paris Perdikaris, Sifan Wang, and Liu Yang. Physics-informed machine learning. *Nature Reviews Physics*, 3(6):422–440, 2021.

Nikola Kovachki, Zongyi Li, Burigede Liu, Kamyar Azizzadenesheli, Kaushik Bhattacharya, Andrew Stuart, and Anima Anandkumar. Neural operator: Learning maps between function spaces with applications to PDEs. *Journal of Machine Learning Research*, 24(89):1–97, 2023.

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.

Zongyi Li, Nikola Kovachki, Kamyar Azizzadenesheli, Burigede Liu, Kaushik Bhattacharya, Andrew Stuart, and Anima Anandkumar. Fourier neural operator for parametric partial differential equations. In *International Conference on Learning Representations*, 2021.

Zongyi Li, Miguel Liu-Schiaffini, Nikola Kovachki, Kamyar Azizzadenesheli, Burigede Liu, Kaushik Bhattacharya, Andrew Stuart, and Anima Anandkumar. Learning chaotic dynamics in dissipative systems. In *Advances in Neural Information Processing Systems*, volume 35, pp. 16768–16781, 2022.

Zongyi Li, Daniel Zhengyu Huang, Burigede Liu, and Anima Anandkumar. Fourier neural operator with learned deformations for PDEs on general geometries. *Journal of Machine Learning Research*, 24:1–26, 2023.

Lu Lu, Pengzhan Jin, and George Em Karniadakis. DeepONet: Learning nonlinear operators for identifying differential equations based on the universal approximation theorem of operators. *Nature Machine Intelligence*, 3(3):218–229, 2021. doi: 10.1038/s42256-021-00302-5. URL https://www.nature.com/articles/s42256-021-00302-5.

George Ma, Yifei Wang, Derek Lim, Stefanie Jegelka, and Yisen Wang. A canonicalization perspective on invariant and equivariant learning. In *Advances in Neural Information Processing Systems*, volume 37, 2024.

Alan W. Paeth. A fast algorithm for general raster rotation. In *Graphics Gems*, pp. 179–195. Academic Press, 1990.

Jaideep Pathak, Shashank Subramanian, Peter Harrington, Sanjeev Raja, Ashesh Chattopadhyay, Morteza Mardani, Thorsten Kurth, David Hall, Zongyi Li, Kamyar Azizzadenesheli, Pedram Hassanzadeh, Karthik Kashinath, and Anima Anandkumar. FourCastNet: A global data-driven high-resolution weather model using adaptive Fourier neural operators. *arXiv preprint arXiv:2202.11214*, 2022.

Stephen B. Pope. *Turbulent Flows*. Cambridge University Press, 2000.

Omri Puny, Matan Atzmon, Heli Ben-Hamu, Ishan Misra, Aditya Grover, Edward J. Smith, and Yaron Lipman. Frame averaging for invariant and equivariant network design. In *International Conference on Learning Representations*, 2022.

Zakhar Shumaylov, Peter Zaika, James Rowbottom, Ferdia Sherry, Melanie Weber, and Carola-Bibiane Schönlieb. Lie algebra canonicalization: Equivariant neural operators under arbitrary Lie groups. In *International Conference on Learning Representations*, 2025.

Arnold Sommerfeld. *Mechanics: Lectures on Theoretical Physics*, volume 1. Academic Press, 1952.

Yu Sun, Xiaolong Wang, Zhuang Liu, John Miller, Alexei A. Efros, and Moritz Hardt. Test-time training with self-supervision for generalization under distribution shifts. In *International Conference on Machine Learning*, pp. 9229–9248. PMLR, 2020.

Makoto Takamoto, Timothy Praditia, Raphael Leiteritz, Dan MacKinlay, Francesco Alesiani, Dirk Pflüger, and Mathias Niepert. PDEBench: An extensive benchmark for scientific machine learning. In *Advances in Neural Information Processing Systems*, volume 35, 2022.

Michael Unser, Philippe Thevenaz, and Leonid Yaroslavsky. Convolution-based interpolation for fast, high-quality rotation of images. *IEEE Transactions on Image Processing*, 4(10):1371–1381, 1995.

Dequan Wang, Evan Shelhamer, Shaoteng Liu, Bruno Olshausen, and Trevor Darrell. Tent: Fully test-time adaptation by entropy minimization. In *International Conference on Learning Representations*, 2021.

Rui Wang, Robin Walters, and Rose Yu. Incorporating symmetry into deep dynamics models for improved generalization. *arXiv preprint arXiv:2002.03061*, 2020.

Sifan Wang, Jacob H. Seidman, Shyam Sankaran, Hanwen Wang, George J. Pappas, and Paris Perdikaris. CViT: Continuous vision transformer for operator learning. In *International Conference on Learning Representations*, 2025.

Haixu Wu, Huakun Luo, Haowen Wang, Jianmin Wang, and Mingsheng Long. Transolver: A fast transformer solver for PDEs on general geometries. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 53681–53705, 2024.

A Additional Method Details

This appendix defines the quantities used in the main text and lists the training details.

Let G denote the problem-dependent ambient transformation group. Training and evaluation draw generators from bounded supports $S_{\text{train}}, S_{\text{test}} \subset G$; these supports are not assumed to be subgroups. When both spatial and kinematic components are present, the corresponding Lie algebra admits the decomposition

$$\mathfrak{g} = \mathfrak{g}_{\mathcal{S}} \oplus \mathfrak{g}_{\mathcal{K}}.$$

Kinematic reduction. Let $\boldsymbol{v} \in L^1(D; \mathbb{R}^d)$ denote the velocity field. The kinematic generator $\boldsymbol{v}_{\text{bg}} \in \mathfrak{g}_{\mathcal{K}}$ and the centering operator $\mathcal{S}_{\boldsymbol{v}} : \mathcal{A} \to \mathcal{A}$ are defined by

$$\boldsymbol{v}_{\text{bg}} = \mathcal{P}_{\text{phys}}(a) := \frac{1}{|D|} \int_D \boldsymbol{v}(x)\, \mathrm{d}x, \qquad \mathcal{S}_{\boldsymbol{v}}(a) = a - \boldsymbol{v}_{\text{bg}}.$$

This co-moving decomposition matches the usual interpretation of background transport and momentum removal (Pope, 2000; Sommerfeld, 1952).

Neural parameterization. To parameterize the geometric subalgebra $\mathfrak{g}_{\mathcal{S}}$ without singularities, let $k = \dim(\mathfrak{g}_{\mathcal{S}})$ and use a neural mapping

$$\mathcal{M}_\phi : \mathcal{A} \to \mathbb{R}^{2k},$$

followed by the smooth bounded retraction

$$\mathrm{Ret}(z) = \sum_{j=1}^{k} \tau_j(z) B_j, \qquad \tau_j(z) = \tau_{j,\max} \frac{z_{2j}}{\sqrt{z_{2j-1}^2 + z_{2j}^2 + \varepsilon_{\text{ret}}^2}}, \qquad \varepsilon_{\text{ret}} = 10^{-6},$$

where $\{B_j\}_{j=1}^k$ is the fixed Lie-algebra basis, $z = \mathcal{M}_\phi(a) \in \mathbb{R}^{2k}$, and $\tau_{j,\max}$ is the task-specific support bound. The retraction constrains each coordinate to its task-specific range. The positive regularizer ε_{ret} prevents a singularity at $z = 0$. We use this parameterization for rotational variables (Falorsi et al., 2018; Shumaylov et al., 2025). The spatial generator is

$$\xi_{\mathcal{S}} = \mathrm{Ret}(\mathcal{M}_\phi(\mathcal{S}_{\boldsymbol{v}}(a)))\,.$$

The full Lie-algebra coordinate estimate is then

$$\mathcal{E}_\phi(a) = \xi_{\mathcal{S}} \oplus \boldsymbol{v}_{\text{bg}}.$$

Spectral transformation implementation. Given the full Lie algebra coordinate $\hat{\xi} \in \mathfrak{g}$, we obtain the corresponding transformation parameter through $\hat{g} = \exp(\hat{\xi})$. To avoid dissipation induced by interpolation, the transformation is implemented spectrally. For translations, the continuous Fourier shift theorem gives (Bracewell, 1986)

$$\mathcal{F}(\mathcal{T}_{\boldsymbol{\tau}_{\text{trans}}} \circ a)(\mathbf{k}) = \exp\big(-2\pi i \langle \mathbf{k}, \boldsymbol{\tau}_{\text{trans}} \rangle\big) \hat{a}(\mathbf{k}).$$

Periodic translations implement an exact torus action up to floating-point and spectral truncation error. The three-shear map at an arbitrary angle is instead a numerical rotated-field transformation: except for lattice-compatible angles, it is not an exact automorphism of the square torus and is treated as an approximate action in the Navier–Stokes experiment. For rotations, we use a three-shear factorization in the spectral domain (Paeth, 1990; Unser et al., 1995). With $\mathcal{H}_x(a)(x, y) = (x + ay, y)$ and $\mathcal{H}_y(b)(x, y) = (x, y + bx)$,

$$R_\theta = H_x(\alpha) H_y(\beta) H_x(\alpha), \qquad \alpha = -\tan(\theta/2), \qquad \beta = \sin\theta,$$

and the transformed spectrum applies the corresponding three shear operators,

$$\mathcal{F}(\mathcal{R}_\theta \circ a) = \Big(\widetilde{\mathcal{H}}_x(\alpha) \circ \widetilde{\mathcal{H}}_y(\beta) \circ \widetilde{\mathcal{H}}_x(\alpha)\Big)\, \mathcal{F}(a).$$

Training-time perturbations. We expose the learned operator to controlled geometric shifts by sampling bounded perturbation generators ξ_0 from a restricted training support $\Omega_{\text{train}} \subsetneq \Omega_{\text{OOD}}$ and generating

$$a_{\text{aug}} = \mathcal{C}_{\exp(\xi_0)}(a),$$

for canonical samples $a \sim \mu_c$, following the logic of restricted Lie-group augmentation (Brandstetter et al., 2022). The same sampled generator defines the perturbed training target through the kinematic map $\xi_T = \Gamma_B(\xi_0; T)$. These sampled values are used only to construct training pairs and alignment losses; PACE-FNO does not receive them during in-distribution (ID) or out-of-distribution (OOD) evaluation.

Throughout this appendix, a parameter range attached to a "training perturbation" or Ω_{train} describes only augmented training pairs. The OOD range is the independently generated support Ω_{OOD} used for the reported evaluation tables; it is never reused as a training range. For SWE both supports are given numerically below. For the Burgers tasks, the displayed uniform boxes are the training supports ($\delta_x \in [-0.1, 0.1]$, $v_{\text{bg}} \in [-0.2, 0.2]$ for 1-D; $t_x, t_y \in [-0.1, 0.1]$, $U_{\text{bg}} \in [-0.5, 0.5]$ for 2-D), while the OOD sampler is a separate held-out distribution at the evaluation resolution and is not to be inferred from those training boxes.

Source of input and terminal generators. At inference, PACE-FNO receives only the observed field. The estimator predicts the spatial alignment coordinate, analytic reductions compute kinematic backgrounds from the physical channels, and the stated PDE relation determines the terminal transformation. The coordinate need not identify a unique physical frame. Table 3 lists the source of each quantity.

Table 3: Source of the generators used by PACE-FNO. Learned quantities come from $\mathcal{E}_\phi$ at inference; analytic quantities come from the observed field or the symmetry-induced transport rule. Sampled generators are used only for training perturbations and alignment losses.

Test case	Learned at inference	Analytic at inference	Training-only sampled generator
1-D Burgers	Input translation phase $\hat{\delta}_x$	Background offset $\hat{v}_{\text{bg}}$ from the field mean; terminal shift $\hat{\delta}_{x,T} = \hat{\delta}_x + \hat{v}_{\text{bg}} T$	δ_x, v_{bg} construct consistency perturbations and alignment losses
2-D Burgers	Input translation $\hat{\mathbf{t}} = (\hat{t}_x, \hat{t}_y)$	Scalar background $\hat{U}_{\text{bg}}$ from the field mean; terminal translation $\hat{\mathbf{t}}_T = \hat{\mathbf{t}} + (\hat{U}_{\text{bg}} T, \hat{U}_{\text{bg}} T)$	$\mathbf{t}$, U_{bg} construct consistency perturbations and alignment losses
2-D SWE	Input translation $\hat{\Delta\mathbf{x}}$	Background velocity $\hat{\mathbf{V}}_{\text{bg}}$ from the spatial mean of the stored primitive velocity channels; terminal translation $\hat{\Delta\mathbf{x}}_T = \hat{\Delta\mathbf{x}} + \hat{\mathbf{V}}_{\text{bg}} T$	$\Delta\mathbf{x}$, Fr, and direction angle construct velocity-boost consistency perturbations
(2+1)-D NS	Input rotation–translation parameters $(\hat{\theta}, \hat{t}_x, \hat{t}_y)$	The same numerical spatial transformation is restored, so $\hat{\xi}_T = \hat{\xi}_0$	(θ, t_x, t_y) rotate/translate training trajectories and supervise alignment losses

Detailed losses. The total training objective is

$$\mathcal{L}_{\text{total}}(\theta, \phi) = \mathcal{L}_{\text{PDE}} + \lambda_1 \mathcal{L}_{\text{equiv}} + \lambda_2 \mathcal{L}_{\text{anchor}}.$$

The implementation separates the supervised dynamics path from the geometric-consistency path. The physical term is evaluated on the clean training pair (x, y) after one-shot canonicalization and frame restoration:

$$\mathcal{L}_{\text{PDE}} = \mathbb{E}\left[\frac{\|\hat{u}_{\text{pred}}(x) - y\|_{L^2}}{\|y\|_{L^2}} + \gamma \frac{\||\mathbf{K}| \odot (\mathcal{F}(\hat{u}_{\text{pred}}(x)) - \mathcal{F}(y))\|_{L^2}}{\||\mathbf{K}| \odot \mathcal{F}(y)\|_{L^2}} \right].$$

Independently, a bounded generator Δ_0 produces a perturbed input x^+. The equivariance regularizer matches the relative estimator response to that known perturbation:

$$\mathcal{L}_{\text{equiv}} = \mathbb{E}\big[\|\mathbf{M}(\hat{\xi}_0^+ \ominus \hat{\xi}_0) - \mathbf{M}(\Delta_0)\|_F^2\big],$$

where $\mathbf{M}$ is the fixed coordinate embedding that scales each Lie-algebra component to its training support and returns a matrix representation for matrix-valued generators. For commuting periodic

translations, $\ominus$ is componentwise wrapped subtraction on $\mathbb{T}^d$. For a non-commutative component such as the NS rotation–translation action, it means the local group difference

$$\hat{\xi}_0^+ \ominus_G \hat{\xi}_0 := \mathrm{Log}\Big(\exp(\hat{\xi}_0^+)\exp(-\hat{\xi}_0)\Big),$$

whose Baker–Campbell–Hausdorff terms retain the rotation–translation ordering; we use $\ominus = \ominus_G$ for that block. The clean data are sampled near the canonical slice, so the anchor loss penalizes the spatial estimator output on the clean input:

$$\mathcal{L}_{\mathrm{anchor}} = \mathbb{E}\big[\|P_{\mathcal{S}}\hat{\xi}_0\|_{\mathfrak{g}}^2\big].$$

Here $\mathbf{K}$ is the grid wavenumber array: for a d-dimensional periodic grid with side lengths $L_1, \ldots, L_d$, $\mathbf{K}(\mathbf{n}) = (2\pi n_1/L_1, \ldots, 2\pi n_d/L_d)$ and $|\mathbf{K}| = (\sum_j K_j^2)^{1/2}$; the zero mode is therefore unweighted. The task-specific channel weights, invariant-height term for SWE, and coefficients are listed in Table 5. The anchor coefficient is fixed for the Burgers and SWE runs; only the Navier–Stokes configuration uses the epoch-dependent decay shown in that table.

B Additional Implementation Details

This section reports the problem construction and additional results. We train on canonical low-resolution data and evaluate on geometrically shifted OOD data at inference. The equations follow standard operator-learning benchmarks (Li et al., 2021; Kovachki et al., 2023; Takamoto et al., 2022); the symmetry-induced OOD splits are specific to this study.

B.1 Implementation and reproducibility details

All reported numbers use three random seeds, $\{42, 1234, 3407\}$. The Python, NumPy, and PyTorch random number generators are re-seeded before each data loader is constructed. Tables report means and standard deviations across seeds. Runtime entries are wall-clock time per training epoch and per full OOD evaluation pass, including TTA updates when enabled. All experiments ran on one NVIDIA GeForce RTX 4080 GPU with driver 580.105.08 and CUDA 13.0. The listed epochs, per-epoch time, and three seeds determine the total training compute. FNO+Aug uses the same FNO architecture and optimizer settings as FNO; only the training and evaluation data construction changes.

The implementation and reproduction scripts are provided in the anonymized supplementary material.

Runtime convention and limitation. For the 1-D runtime entries, the OOD tensors and data loader are constructed before the timed evaluation loop. The timed interval includes synchronized device transfers, model forward passes, metric computation, and prediction caching, but not OOD data construction or checkpoint loading. No historical runtime logs are available. We therefore treat the 1-D runtime values as descriptive full-loop measurements and do not interpret the atypical FNO+Aug value as evidence that augmentation intrinsically increases inference latency.

Tables 4 and 5 record the shared training configuration and the PACE-FNO-specific losses and TTA settings.

Table 4: Core training hyperparameters used by the experimental scripts. "Batch" denotes train/OOD-evaluation batch size. GFNO uses the square-grid $p4$ symmetry group (integer translations and rotations by multiples of 90°), which does not cover the continuous sub-pixel translations or Galilean boosts in these OOD tests; for GFNO on NS we use time modes 10 in addition to spatial modes 10. For SWE, the FNO-model weight decay is 10^{-4} for FNO+Aug/PACE-FNO and 10^{-3} for FNO without augmentation and GFNO, matching the experiment configuration. This unequal decay is a protocol limitation and means those SWE rows are not a perfectly regularization-matched architectural comparison.

Test case	Models	Train/Test	ID Grid	OOD Grid	Batch	Epochs	Optimizer, Scheduler, Architecture
1-D Burgers	FNO, FNO+Aug, GFNO, PACE-FNO	1000/200	1024	2048	20/1	500	Adam, lr 10^{-3}, wd 10^{-4}; StepLR $(50, 0.5)$; modes 20, width 64
2-D Burgers	FNO, FNO+Aug, GFNO, PACE-FNO	1000/200	64^2	128^2	20/10	500	Adam, lr 10^{-3}, wd 10^{-4}; StepLR $(100, 0.5)$; modes 12, width 32
2-D SWE	FNO, FNO+Aug, GFNO, PACE-FNO	1000/200	64^2	128^2	20/10	500	Adam, lr 10^{-3}; StepLR $(100, 0.2)$; modes 16, width 32
(2+1)-D NS	FNO, FNO+Aug, GFNO, PACE-FNO	1000/200	64^2	128^2	10/10	100	Adam, lr 2.5×10^{-3}, wd 10^{-4}; StepLR $(100, 0.5)$; spatial modes 10, width 20

For the 1-D Burgers split, the exponential moving average in the loss label "EMA H^1 weight" tracks the normalized batch ratio

$$r_e = \frac{\mathcal{L}_{H^1,e}}{\mathcal{L}_{2,e} + \mathcal{L}_{H^1,e} + 10^{-8}}, \qquad \bar{r}_e = 0.9\,\bar{r}_{e-1} + 0.1\,r_e, \qquad \lambda_{H^1,e} = \mathrm{clip}(\bar{r}_e, 0.05, 0.95),$$

Table 5: PACE-FNO specific losses and TTA settings. One-shot PACE-FNO uses the same trained checkpoint and sets the number of TTA steps to zero. The Lie-algebra coordinate estimator parameter group uses the same learning rate as the FNO predictor and no additional weight decay unless explicitly listed.

Test case	PACE-FNO Training Loss	TTA Variables and Optimizer	TTA Steps / LR
1-D Burgers	Split L^2/H^1 loss with EMA H^1 weight ($\alpha = 0.9$) plus $10\,\mathcal{L}_{\text{equiv}}$	Translation phase, Adam	20 / 0.05
2-D Burgers	$\mathcal{L}_2 + 5\mathcal{L}_{H^1} + 1\mathcal{L}_{\text{equiv}} + 5\mathcal{L}_{\text{anchor}}$	(t_x, t_y), Adam	50 / 0.01
2-D SWE	Channel-weighted PDE loss plus scheduled H^1, $10\mathcal{L}_{\text{equiv}}$, $5\mathcal{L}_{\text{anchor}}$, and 0.5 invariant-height loss	(t_x, t_y), AdamW with wd 10^{-4}	80 / 0.05
(2+1)-D NS	$\mathcal{L}_2 + 5\mathcal{L}_{H^1} + 1\mathcal{L}_{\text{equiv}} + \lambda_{\text{anc}}(e)\mathcal{L}_{\text{anchor}}$, $\lambda_{\text{anc}}(e) = 20(0.85)^e + 0.1$	(θ, t_x, t_y), Adam	5 / 0.05

with $\bar{r}_0 = r_0$; the split loss is $(1 - \lambda_{H^1,e})\mathcal{L}_2 + \lambda_{H^1,e}\mathcal{L}_{H^1}$. Thus the EMA smooths the relative contribution of the two normalized physical terms, not model weights or predictions.

B.2 1-D Burgers equation

The benchmark solves

$$\partial_t u + \tfrac{1}{2}\partial_x(u^2) = \nu\partial_{xx}u, \qquad u(x,0) = u_0(x), \qquad x \in \mathbb{T},$$

with periodic boundary conditions. Canonical ID initial fields are sampled from a Gaussian random field, while OOD samples are generated by

$$u_{0,\text{OOD}}(x) = u_0(x - \Delta x) + c,$$

which combines spatial translation Δx and a Galilean velocity offset c. This follows the restricted Lie-group augmentation framework (Brandstetter et al., 2022).

Data augmentation. We sample measure-preserving perturbations from the Galilean group $G = \mathbb{T}^1 \rtimes \text{Gal}$. For a canonical initial field $u_0(x)$, the augmented state at $t = 0$ is generated by the pushforward action of $g_0 = \exp(\xi_0)$:

$$u_{\text{aug}}(x,0) = u_0(x - \delta_x) + v_{\text{bg}}$$

where the initial Lie algebra generator $\xi_0 \in \mathfrak{g}$ is defined by the superposition of the spatial translation and velocity boost differential operators:

$$\xi_0 = -\delta_x \frac{\partial}{\partial x} + v_{\text{bg}}\mathcal{I}_u$$

Here, $\delta_x \sim \mathcal{U}(-0.1, 0.1)$ parameterizes the spatial translation generator within the geometric subalgebra $\mathfrak{g}_{\mathcal{S}}$, $\mathcal{I}_u$ denotes the identity shift operator on the codomain, and $v_{\text{bg}} \sim \mathcal{U}(-0.2, 0.2)$ represents the Galilean kinematic velocity boost within $\mathfrak{g}_{\mathcal{K}}$.

To maintain Galilean invariance during physical evolution, the target state at time T is shifted by the induced kinematic convection. The target group element evolves analytically as $g_T = \exp(\xi_0 + \Delta x \cdot \frac{\partial}{\partial x})$, where the spatial translation increment is $\Delta x = v_{\text{bg}}T$.

Self-consistency TTA. PACE-FNO keeps all learned weights fixed and adapts only the coordinate, unlike parameter-updating test-time training (TTT) and test-time entropy minimization (TENT) (Sun et al., 2020; Wang et al., 2021). After canonicalization, the trained estimator $\mathcal{E}_\phi$ should produce a near-zero residual. We define the residual phase shift as $\epsilon_{\text{res}} = \mathcal{P}_{\mathcal{S}} \circ \mathcal{E}_\phi\big(\mathcal{C}^{-1}_{\exp(\hat{\xi}_{\mathcal{S}})}(a_{\text{OOD}})\big)$, where $\mathcal{P}_{\mathcal{S}}$ projects onto the spatial translation subspace. The TTA loss is the batch mean squared residual:

$$\mathcal{J}_{\text{TTA}}(\hat{\xi}_{\mathcal{S}}) = \frac{1}{B}\sum_{i=1}^{B}\left(\epsilon^{(i)}_{\text{res}}\right)^2 = \frac{1}{B}\sum_{i=1}^{B}\left(\mathcal{P}_{\mathcal{S}} \circ \mathcal{E}_\phi\Big(\mathcal{C}^{-1}_{\exp(\hat{\xi}^{(i)}_{\mathcal{S}})}(a^{(i)}_{\text{OOD}})\Big)\right)^2$$

where B denotes the batch size.

Figure 7 shows the pullback on a representative translated and boosted 1-D Burgers state.

Figure 8 compares the resulting OOD predictions under the same shifted input.

B.3 2-D Burgers equation

The two-dimensional benchmark solves

$$\partial_t u + \tfrac{1}{2}\partial_x(u^2) + \tfrac{1}{2}\partial_y(u^2) = \nu(\partial_{xx}u + \partial_{yy}u), \qquad (x,y) \in \mathbb{T}^2.$$

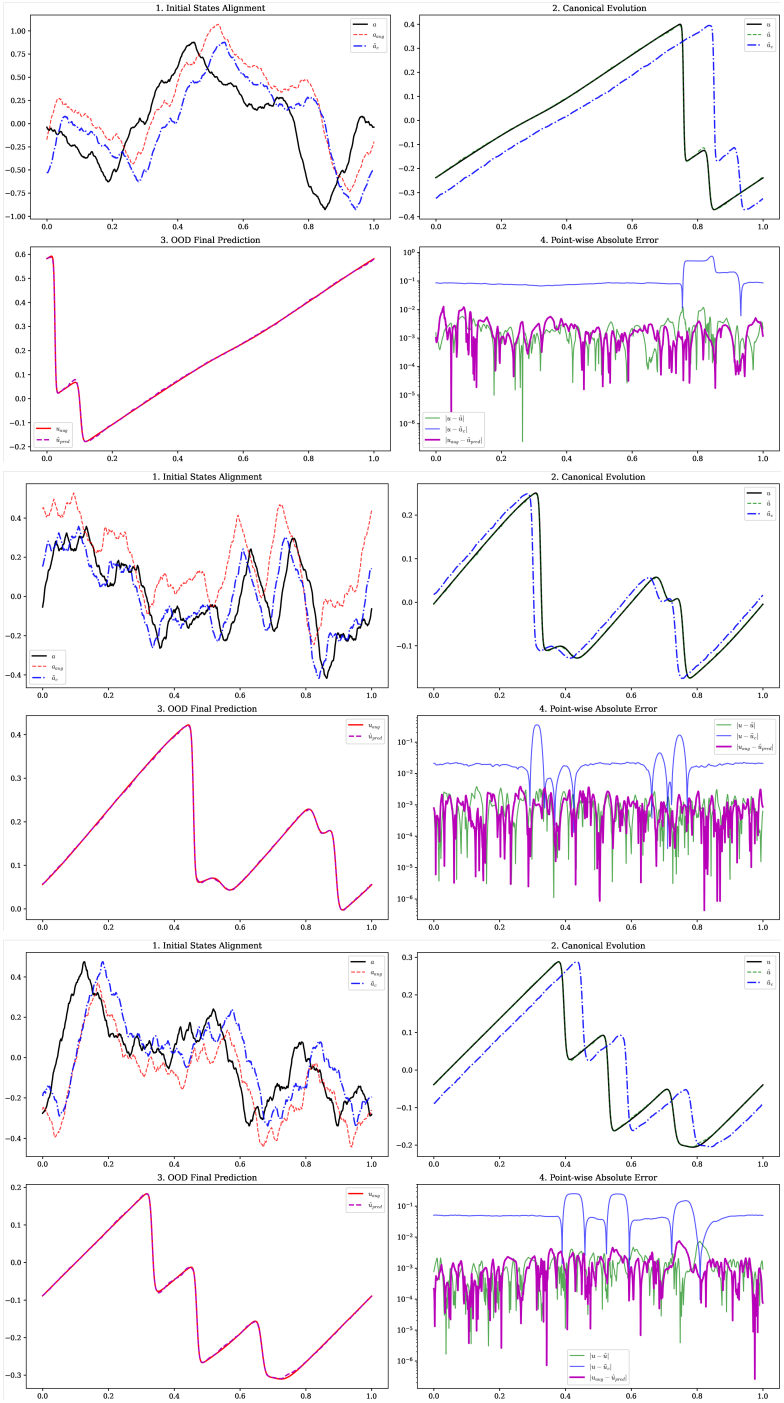
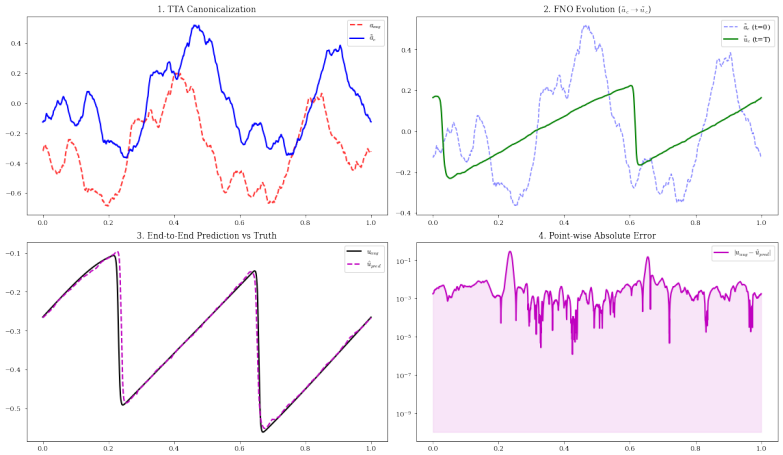
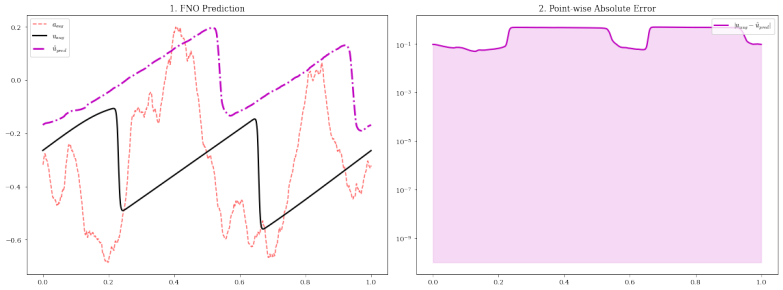


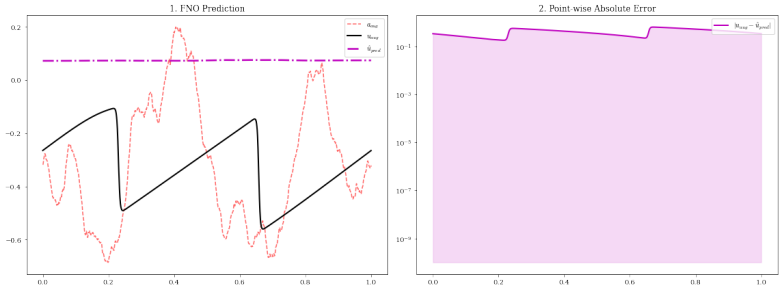
Figure 7: Canonicalization on 1-D Burgers. The physical panels show the translated and boosted input being pulled back toward the canonical frame; the latent panels show that the estimator residual is small after alignment.



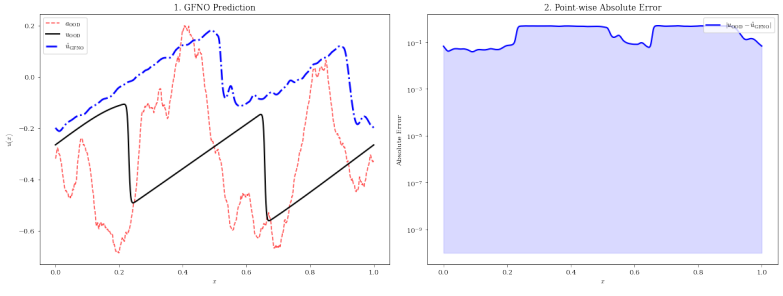
(a) PACE-FNO.



(b) FNO.



(c) FNO+Aug.



(d) GFNO.

Figure 8: OOD prediction on 1-D Burgers under translation and Galilean drift. PACE-FNO aligns the wave location and amplitude with the target, while the baselines show frame mismatch or phase error under the same shifted input.

We use the scalar 2-D Burgers variant rather than the coupled vector-valued system to isolate the effect of geometric canonicalization from vector-valued coupling. Canonical states are centered and aligned, while OOD samples follow

$$u_{0,\mathrm{OOD}}(x, y) = u_0(x - t_x, y - t_y) + U_{\mathrm{bg}},$$

with translations (t_x, t_y) and background flow U_{bg}.

Data augmentation. For the 2-D setting, we sample measure-preserving perturbations from the Galilean group $G = \mathbb{T}^2 \rtimes \mathrm{Gal}$. The corresponding initial Lie algebra generator $\xi_0 \in \mathfrak{g}$ is defined by the superposition of the two-dimensional spatial translation and velocity boost differential operators:

$$\xi_0 = -t_x \frac{\partial}{\partial x} - t_y \frac{\partial}{\partial y} + U_{\mathrm{bg}} \mathcal{I}_u$$

Here, the translation vector $\mathbf{t} = (t_x, t_y) \sim \mathcal{U}(-0.1, 0.1)^2$ parameterizes the spatial generators within the geometric subalgebra $\mathfrak{g}_{\mathcal{S}}$, $\mathcal{I}_u$ denotes the identity shift operator on the codomain, and $U_{\mathrm{bg}} \sim \mathcal{U}(-0.5, 0.5)$ represents the Galilean velocity boost within $\mathfrak{g}_{\mathcal{K}}$. These shifts violate the mass and amplitude constraints of the in-distribution dataset.

To preserve Galilean invariance during spatiotemporal evolution, the target state at time T undergoes kinematic convection. The target group element evolves analytically as $g_T = \exp\left(\xi_0 + \Delta\mathbf{x} \cdot \nabla\right)$, where the translation increment vector is driven by the background flow such that $\Delta\mathbf{x} = (U_{\mathrm{bg}} T, U_{\mathrm{bg}} T)$.

Self-consistency TTA. The estimator applied to the canonicalized state should return near-zero residual shifts when alignment succeeds. We define the residual two-dimensional phase shift vector as $\boldsymbol{\epsilon}_{\mathrm{res}} = (\epsilon_{\mathrm{res},x}, \epsilon_{\mathrm{res},y})$. The adaptation objective during evaluation is the empirical mean squared error of these two-dimensional phase residuals over a given batch:

$$\mathcal{J}_{\mathrm{TTA}}(\hat{\xi}_{\mathcal{S}}) = \frac{1}{B} \sum_{i=1}^{B} \left((\epsilon_{\mathrm{res},x}^{(i)})^2 + (\epsilon_{\mathrm{res},y}^{(i)})^2 \right)$$

where B denotes the batch size.

Figure 9 shows the learned pullback before the 2-D Burgers rollout.

Figure 10 gives the corresponding baseline OOD predictions.

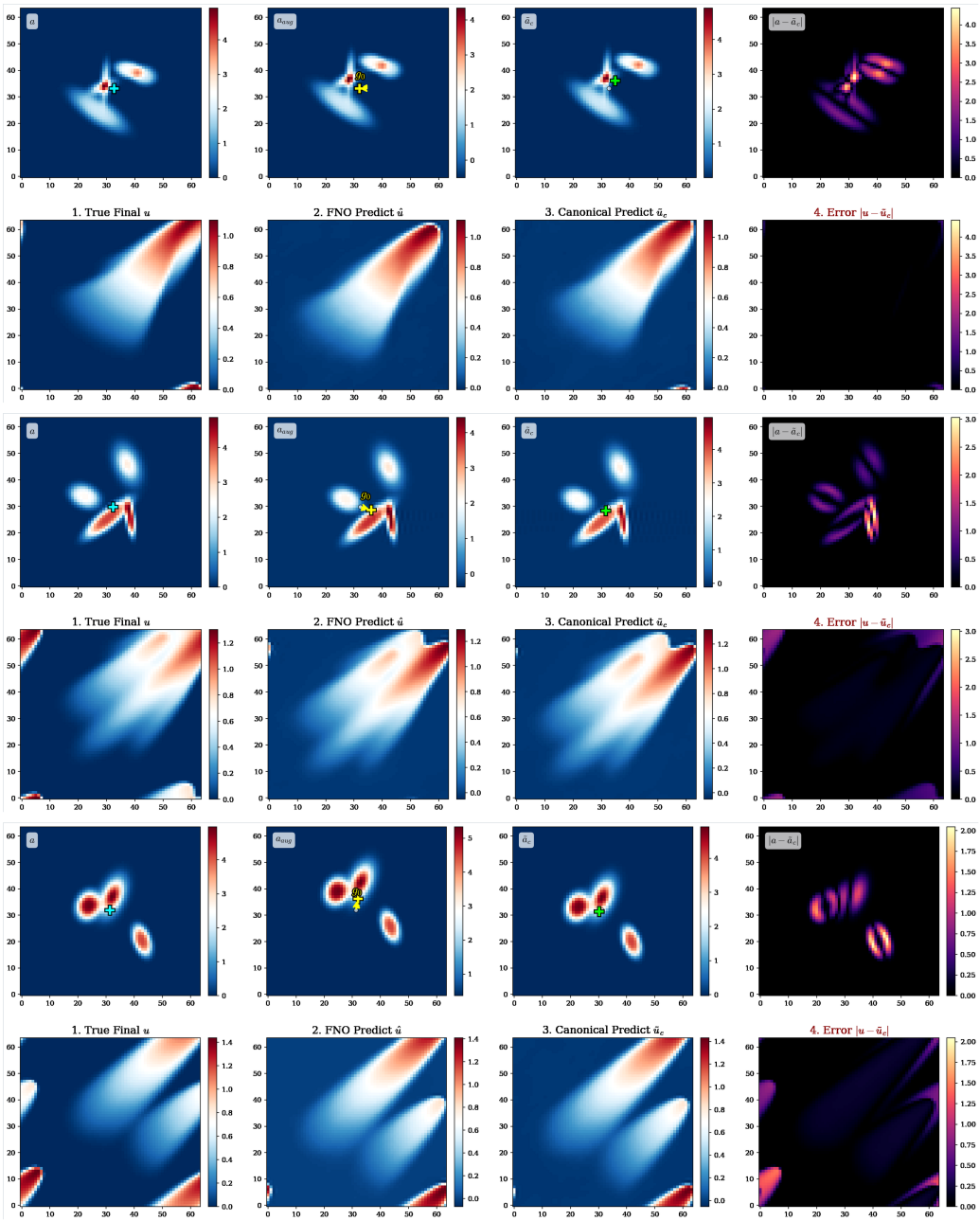
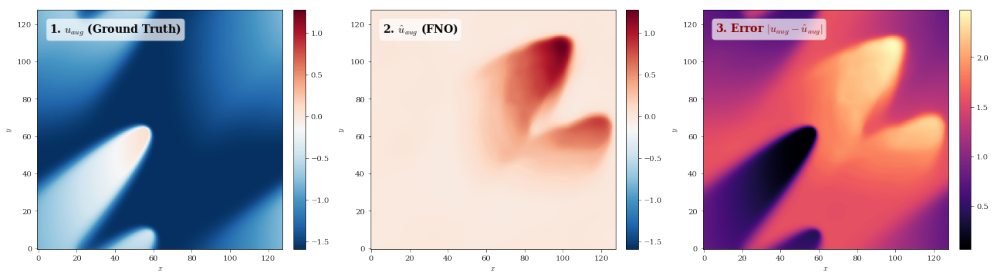
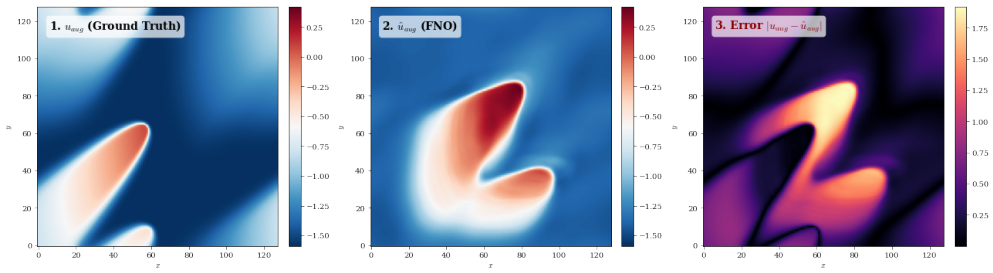


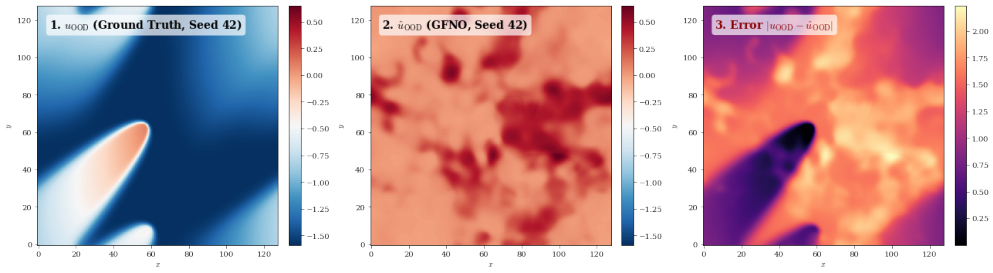
Figure 9: Canonicalization on 2-D Burgers. The learned pullback removes the dominant spatial shift and background-flow component before the FNO rollout, leaving a canonical field whose latent residual is close to the ID frame.



(a) FNO.



(b) FNO+Aug.



(c) GFNO.

Figure 10: Baseline OOD predictions on 2-D Burgers under translation and Galilean drift. The corresponding PACE-FNO prediction is shown in the main paper; these additional panels expose the frame errors retained by FNO, FNO+Aug, and GFNO under the same shifted input.

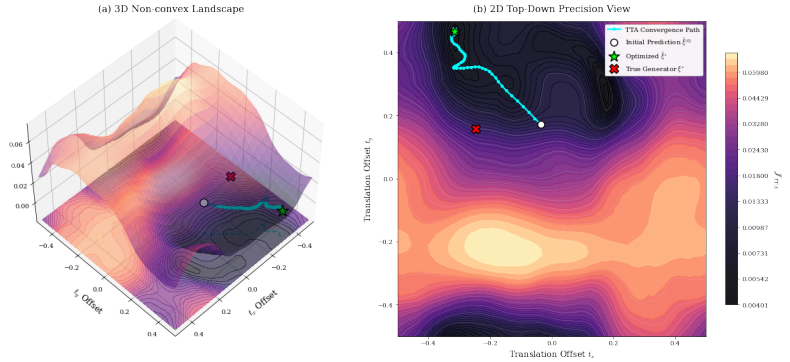


Figure 11: The 2-D Burgers TTA objective has symmetry-induced low-energy basins. TTA changes only the estimated coordinate; it does not update the FNO or estimator weights.

B.4 2-D SHALLOW-WATER EQUATIONS

The numerical solver evolves the conservative variables $\mathbf{U} = (h, \mathbf{m})$ according to

$$\partial_t h + \nabla \cdot \mathbf{m} = \nu \nabla^2 h, \qquad \partial_t \mathbf{m} + \nabla \cdot \left(\frac{\mathbf{m} \otimes \mathbf{m}}{h} \right) + \nabla \left(\tfrac{1}{2} g h^2 \right) = \nu \nabla^2 \mathbf{m}$$

on $\mathbb{T}^2$. The released operator-learning arrays, however, store the primitive representation $(h, \mathbf{v})$ with $\mathbf{v} = \mathbf{m}/h$; the solver converts back to velocity before saving every frame. Canonical conditions

enforce zero macroscopic background velocity and centered mass distribution. OOD states are generated by

$$\mathbf{U}_{0,\mathrm{OOD}}(\mathbf{x}) = \mathcal{B}_{\mathbf{V}_{\mathrm{bg}}}\mathbf{U}_0(\mathbf{x} - \Delta\mathbf{x}),$$

which combines sub-pixel translation and a Galilean background flow. In the stored primitive representation, the boost is $\mathcal{B}_{\mathbf{V}}[h, \mathbf{v}]^\top = [h, \mathbf{v} + \mathbf{V}]^\top$; equivalently, in the solver's conservative variables, $\mathcal{B}_{\mathbf{V}}[h, \mathbf{m}]^\top = [h, \mathbf{m} + h\mathbf{V}]^\top$.

Data augmentation. For shallow-water dynamics, we use the same Galilean group $G = \mathbb{T}^2 \rtimes \mathrm{Gal}$ to perturb the initial vector field. The pushforward action of $g_0 = \exp(\xi_0)$ is the boost-and-translation map above, with initial Lie algebra generator

$$\xi_0 = -t_x \frac{\partial}{\partial x} - t_y \frac{\partial}{\partial y} + \mathbf{V}_{\mathrm{bg}}\mathcal{I}_{\mathbf{v}}$$

Here, the translation vector $\Delta\mathbf{x} = (t_x, t_y) \sim \mathcal{U}(-0.15, 0.15)^2$ parameterizes the spatial generators within the geometric subalgebra $\mathfrak{g}_S$. To reduce interpolation-induced dissipation of high-frequency surface ripples, these continuous sub-pixel translations are executed analytically in the Fourier domain. The boost direction $\mathcal{I}_{\mathbf{v}}$ acts additively on the stored velocity channels. For each clean sample, let A denote its inferred surface-height amplitude and define the reference wave speed

$$c_{\mathrm{ref}} = \sqrt{g(1 + A/2)}.$$

Training perturbations sample $\mathrm{Fr} \sim \mathcal{U}(0, 0.2)$ and $\alpha \sim \mathcal{U}(0, 2\pi)$, then set

$$\mathbf{V}_{\mathrm{bg}} = c_{\mathrm{ref}}\,\mathrm{Fr}\,(\cos\alpha, \sin\alpha).$$

Thus the implemented training support is radial in Froude number rather than a component-wise uniform box in $(u_{\mathrm{bg}}, v_{\mathrm{bg}})$. The independently solved OOD set uses $\mathrm{Fr} \sim \mathcal{U}(0.3, 0.8)$ and samples the wave center uniformly over the periodic grid, giving $t_x, t_y \in [-\frac{1}{2} + \frac{1}{2s}, \frac{1}{2} - \frac{1}{2s}]$ at the generation resolution $s = 256$. These shifts lie outside the training support and violate the mass-centering and amplitude constraints of the in-distribution dataset.

To maintain Galilean invariance during spatiotemporal evolution, the target state at time T undergoes kinematic convection. The target group element evolves analytically as $g_T = \exp\left(\xi_0 + \Delta\mathbf{x}_T \cdot \nabla\right)$, where the translation increment vector is dynamically driven by the background flow such that $\Delta\mathbf{x}_T = \mathbf{V}_{\mathrm{bg}}T$.

Self-consistency TTA. After canonicalization, the estimator should produce near-zero residual shifts. We define the residual two-dimensional phase shift vector as $\boldsymbol{\epsilon}_{\mathrm{res}} = (\epsilon_{\mathrm{res},x}, \epsilon_{\mathrm{res},y})$. The TTA objective is the empirical mean squared error over a batch:

$$\mathcal{J}_{\mathrm{TTA}}(\hat{\xi}_{\mathcal{S}}) = \frac{1}{B}\sum_{i=1}^{B}\left((\epsilon_{\mathrm{res},x}^{(i)})^2 + (\epsilon_{\mathrm{res},y}^{(i)})^2\right)$$

where B denotes the batch size.

Figure 12 shows the aligned shallow-water states before prediction.

Figure 13 shows the baseline predictions for the same shallow-water shift.

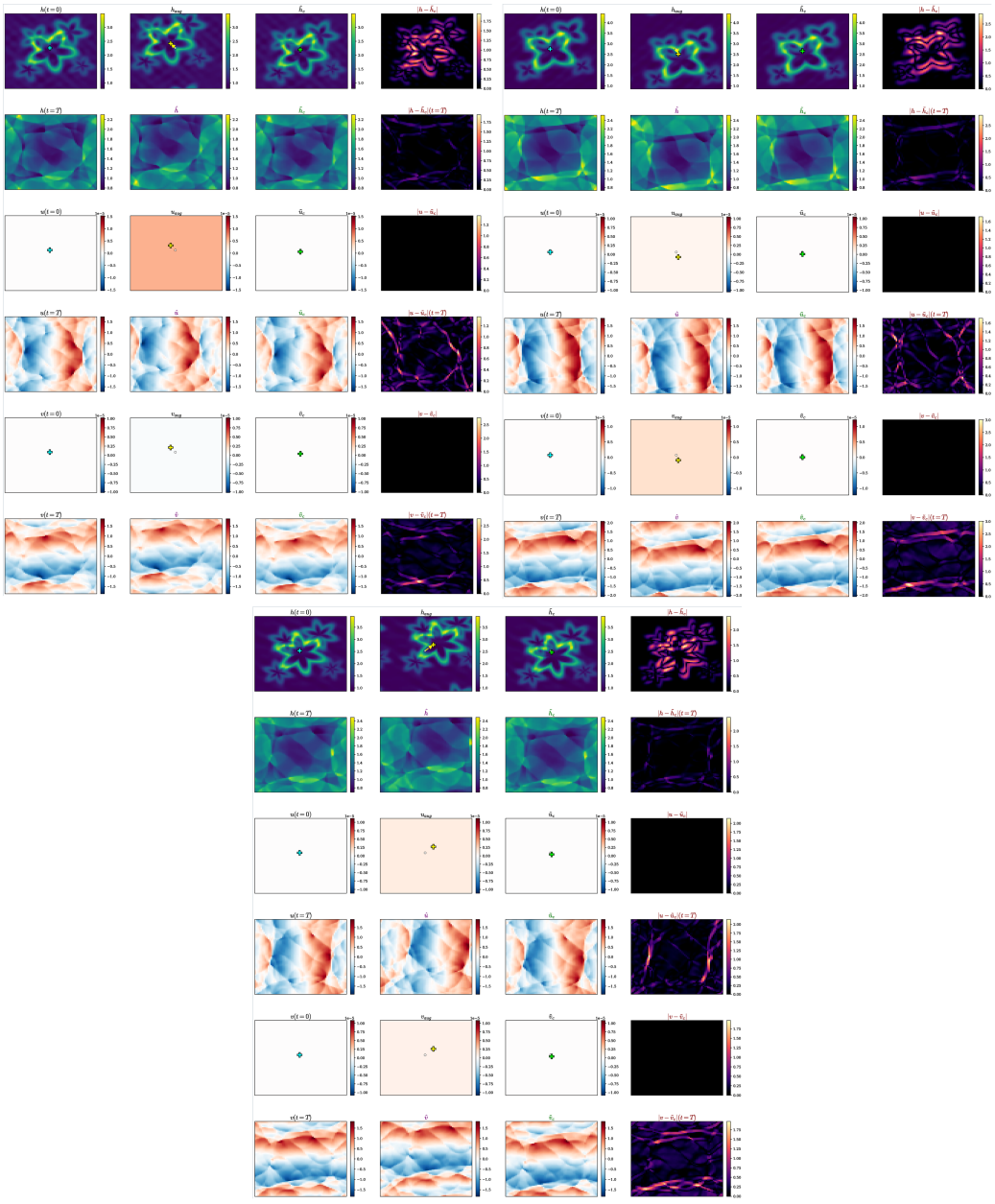
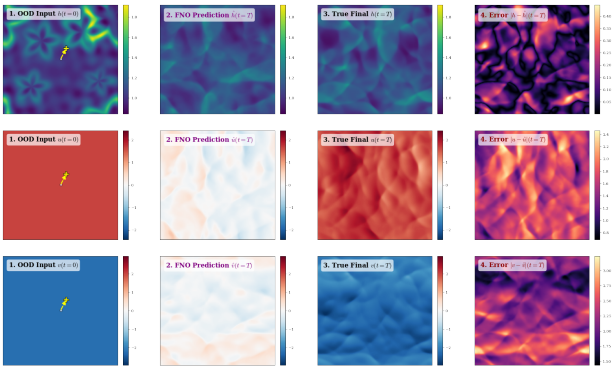
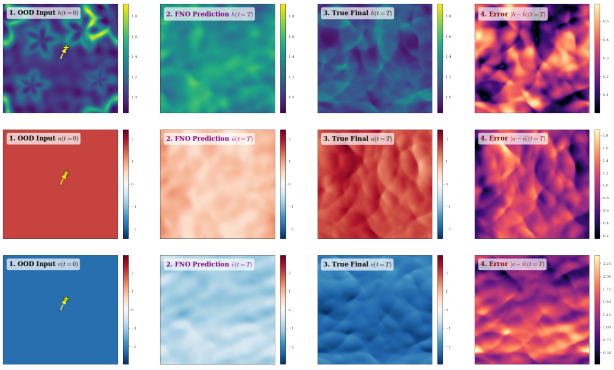


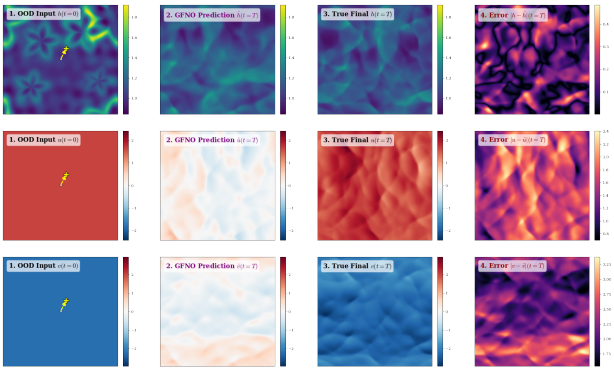
Figure 12: Canonicalization behavior of PACE-FNO under Galilean perturbations for the 2-D shallow-water system. The aligned states remove the dominant translation and background transport before prediction, which explains the smaller OOD error in Table 11.



(a) FNO.



(b) FNO+Aug.



(c) GFNO.

Figure 13: Baseline OOD predictions on the 2-D shallow-water system under Galilean perturbation. The corresponding PACE-FNO prediction is shown in the main paper; the additional baseline panels show the larger displacement and amplitude errors.

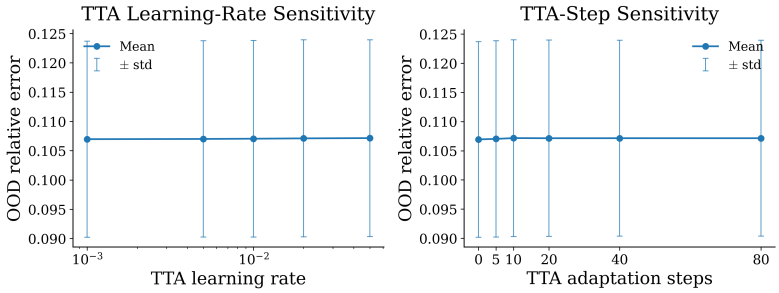


Figure 14: SWE TTA sensitivity to the tested learning rates and adaptation-step budgets. The final OOD relative error changes little over the reported ranges.

B.5 (2+1)-D NAVIER-STOKES EQUATIONS

The vorticity benchmark solves

$$\partial_t \omega + \mathbf{u}\cdot\nabla\omega = \nu\nabla^2\omega, \qquad \mathbf{u} = (\partial_y\psi, -\partial_x\psi), \qquad \nabla^2\psi = -\omega$$

on $\mathbb{T}^2$. It evaluates spatiotemporal rollouts under a controlled coupled translation–rotation transformation. Models observe $T_{\text{in}} = 10$ steps and predict $T = 20$ steps. Each OOD trajectory is solved from an initial condition constructed with one parameter tuple; at $t = 0$ this gives

$$\omega_{0,\text{OOD}}(\mathbf{x}) = \omega_0\big(\mathbf{R}_{-\theta}(\mathbf{x} - \Delta\mathbf{x})\big),$$

which applies a translation and a spatial rotation to the canonical initial condition.

Rotation–translation augmentation. For the Navier–Stokes task, we parameterize the planar transformation by an ambient $\mathfrak{se}(2)$ generator, formed by superposing two-dimensional translation and rigid rotation differential operators:

$$\xi_0 = -t_x\frac{\partial}{\partial x} - t_y\frac{\partial}{\partial y} - \theta\left(x\frac{\partial}{\partial y} - y\frac{\partial}{\partial x}\right)$$

Here, $\Delta\mathbf{x} = (t_x, t_y) \sim \mathcal{U}(-0.05, 0.05)^2$ is an exact periodic translation, while $\theta \sim \mathcal{U}(-\pi/4, \pi/4)$ controls the three-shear rotated-field map. The ambient $\mathfrak{se}(2)$ notation parameterizes this implementation; it does not make the arbitrary-angle map a representation of the full $SE(2)$ group on the square torus. Unlike kinematic background flows, these spatial perturbations do not accumulate a macroscopic advection increment over time.

Self-consistency TTA. After canonicalization, the estimator should return little residual translation or rotation. We write the residual translation as $\mathbf{t}_{\text{res}} = (t_{x,\text{res}}, t_{y,\text{res}})$ and the residual rotation as θ_{res}. To avoid phase-wrapping discontinuities and singular gradients near the periodic parameter identifications, the TTA loss uses a cosine distance rather than mean squared error:

$$\mathcal{J}_{\text{TTA}}(\hat{\xi}_{\mathcal{S}}) = \frac{1}{B}\sum_{k=1}^{B}\Big(\big(1-\cos(\theta_{\text{res}}^{(k)})\big) + \big(1-\cos(2\pi t_{x,\text{res}}^{(k)})\big) + \big(1-\cos(2\pi t_{y,\text{res}}^{(k)})\big)\Big)$$

This formulation removes discontinuities induced by the periodic identification of angles and translations.

Figure 15 shows the frame correction on the Navier–Stokes rollout input.

Figure 16 compares the baseline rollouts under the controlled diagnostic evaluation.

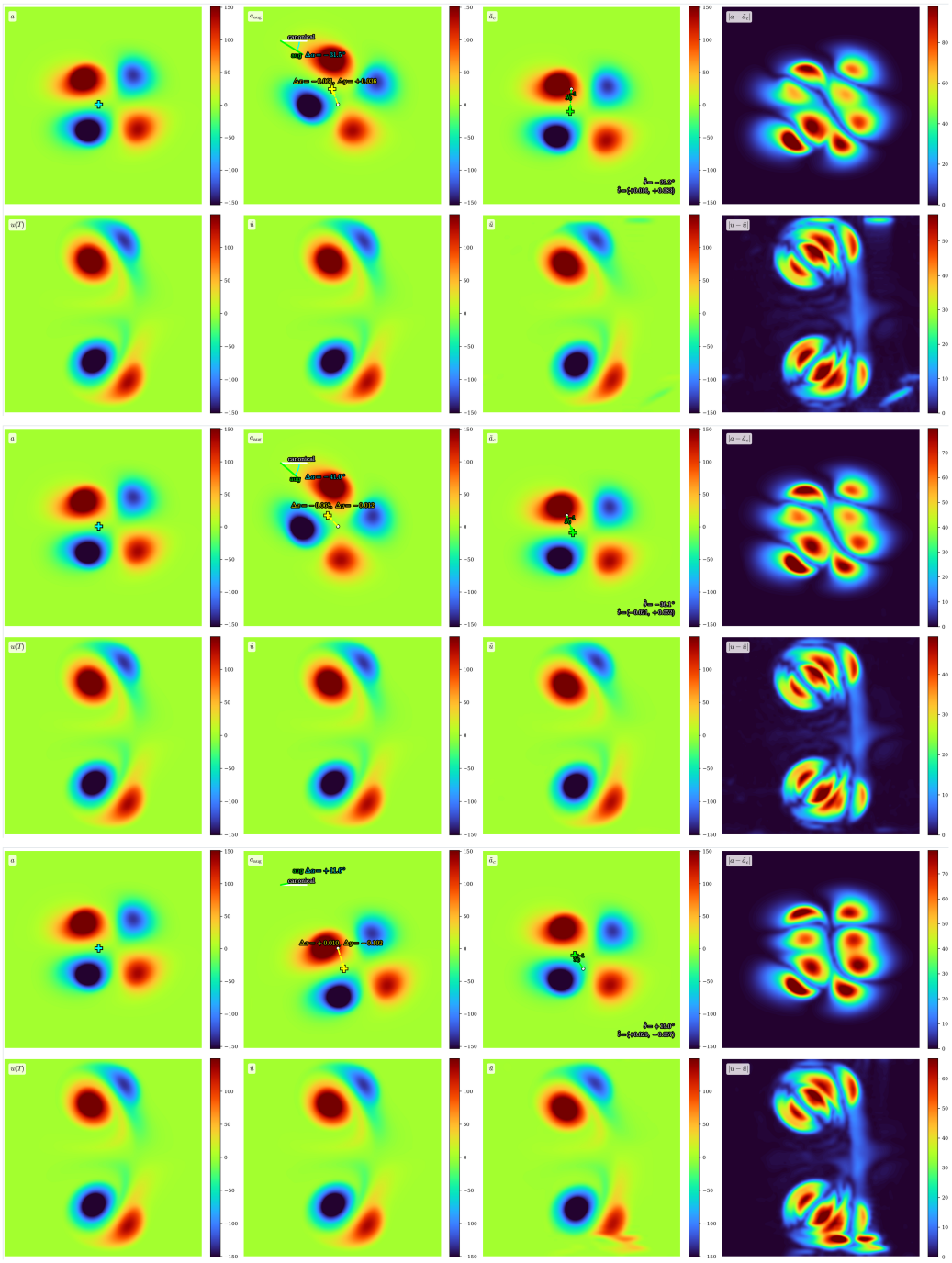
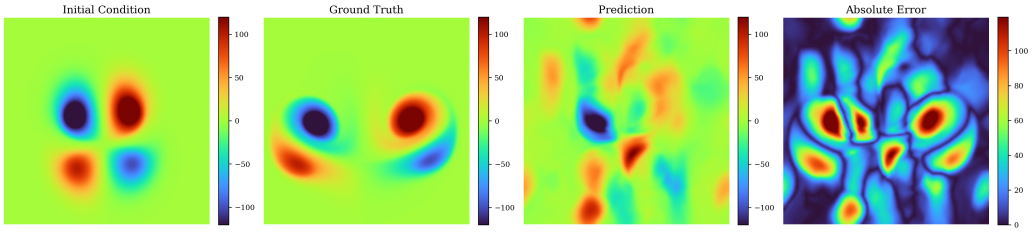
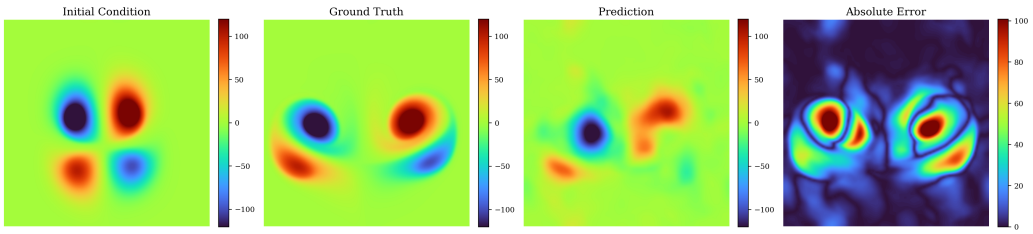


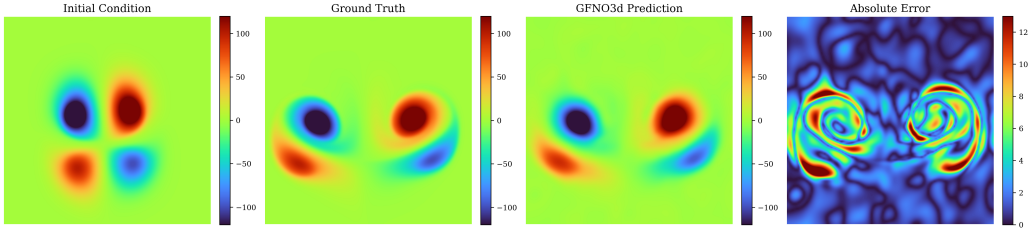
Figure 15: Canonicalization on the $(2+1)$-dimensional Navier-Stokes setting. The learned frame correction is applied to the imposed translation-rotation shift before rollout.



(a) FNO.



(b) FNO+Aug.



(c) GFNO.

Figure 16: Baseline OOD predictions under the controlled rotation–translation diagnostic for the $(2+1)$-dimensional Navier–Stokes equations. The corresponding PACE-FNO prediction is shown in the main paper; these panels isolate the baseline failures in vortex placement and rollout geometry.

B.6 MECHANISM DIAGNOSTICS

Table 6 collects the ablation and observation-diagnostic results used to separate the wrapper from auxiliary estimator capacity.

Table 6: Unified ablation study of PACE-FNO. The *Without canonicalization* variant disables learned pullback/pushforward during inference. For SWE, *Without the full Lie generator* is numerically unstable (unclipped error reported). Dashes indicate entries with no independent training cost or evaluation.

Dataset	Model	Relative Error		Runtime (s)	
		ID relative error ↓	OOD relative error ↓	Training (s/epoch)	OOD evaluation (s)
1D Burgers	PACE-FNO + TTA	0.0621 ± 0.0028	0.0260 ± 0.0012	0.4864 ± 0.0037	5.9590 ± 0.0963
	Without the full Lie generator	0.0471 ± 0.0027	0.0193 ± 0.0007	0.4902 ± 0.0034	6.0546 ± 0.2355
	PACE-FNO (one-shot)	–	0.0267 ± 0.0014	–	0.5287 ± 0.0171
	Without canonicalization	–	1.0384 ± 0.0003	–	0.4016 ± 0.0058
2-D Burgers	PACE-FNO + TTA	0.0393 ± 0.0012	0.0640 ± 0.0012	1.5770 ± 0.0066	4.4345 ± 0.0409
	Without the structured spatial generator	0.0392 ± 0.0008	0.0666 ± 0.0076	1.0761 ± 0.0268	4.3969 ± 0.0336
	Without the full Lie generator	0.0303 ± 0.0018	0.5478 ± 0.0303	1.1030 ± 0.0083	4.3551 ± 0.0131
	PACE-FNO (one-shot)	–	0.0755 ± 0.0139	–	0.1399 ± 0.0023
	Without canonicalization	–	1.0085 ± 0.0035	–	0.1417 ± 0.0311
2-D SWE	PACE-FNO + TTA	0.0491 ± 0.0002	0.1071 ± 0.0168	2.6139 ± 0.0046	8.1388 ± 0.0855
	Without the structured spatial generator	0.0501 ± 0.0011	0.1008 ± 0.0142	2.5280 ± 0.0059	7.8222 ± 0.0129
	Without the full Lie generator	0.0586 ± 0.0058	$2.11 \times 10^4 \pm 2.28 \times 10^4$	2.0817 ± 0.0222	7.5391 ± 0.0296
	PACE-FNO (one-shot)	–	0.1069 ± 0.0168	–	0.2952 ± 0.0349
	Without canonicalization	–	0.7232 ± 0.0008	–	0.1426 ± 0.0260
(2+1)D NS	PACE-FNO + TTA	0.0129 ± 0.0006	0.2593 ± 0.0522	16.5571 ± 0.3631	10.0376 ± 0.0418
	Without canonicalization	0.0133 ± 0.0006	0.6280 ± 0.0076	10.6182 ± 0.0660	2.6558 ± 0.0342
	Without the full Lie generator	0.0133 ± 0.0006	0.6280 ± 0.0076	11.0809 ± 0.0649	2.7494 ± 0.0169
	PACE-FNO (one-shot)	–	0.2766 ± 0.0353	–	3.3728 ± 0.1381

The ablation above separates the one-shot canonicalizer from iterative TTA. Figure 14 visualizes the tested SWE sensitivity range alongside the complete ablation table.

B.7 VISION TRANSFER PROTOCOLS

This appendix lists the data partitions, inference budgets, measurement conventions, and diagnostic-figure rules for the vision experiments in Section C.5. The reported model selection remains unchanged.

B.8 PACE-FNO–LIELAC COMPARISON SCOPE

LieLAC (Shumaylov et al., 2025) is a neural-operator method evaluated on PDE solvers. We include a faithful LieLAC+FNO comparison on the 1-D Burgers OOD split in Table 9, and use Tables 16 and 17 for the additional image comparisons. The remaining PDE backbone experiments use different field actions, data construction, and error objectives; they test wrapper transfer across predictors and are not LieLAC reproductions.

B.9 CAMELYON17-WILDS PROTOCOL AND LEAKAGE AUDIT

We use the official Camelyon17-WILDS v1.0 split. Training and ID validation contain hospitals $\{0, 3, 4\}$, OOD validation contains hospital 1, and OOD test contains hospital 2. The implementation audits duplicate sample identifiers and slide overlap: training and ID validation share the official source-slide pool but have disjoint patches, while each OOD hospital is slide-disjoint from every source split and from the other OOD split. Hyperparameters and checkpoint selection use ID validation only; neither OOD validation nor OOD test is used for fitting or selection.

The Camelyon canonicalizer acts in log-optical-density color space: its six parameters are three channel scales and three offsets, so the hematoxylin and eosin (H&E)-correlated synthetic perturbation is a stain/color transformation and does not spatially move the tissue. It uses a six-dimensional Huber inverse-parameter loss and an ℓ_1 reconstruction term with coefficient 2. For the TTA labels, "moment" initializes the coordinate from low-order source-image color statistics, whereas "learned" initializes it from the amortized encoder; both then optimize only the six-dimensional coordinate with frozen network weights. The convolutional neural network (CNN) and FNO columns in Table 16 are two classifier-backbone variants of the same canonicalizer, and are reported separately to avoid implying

that one method has two simultaneous architectures. Full LieLAC uses 30 gradient steps and 8 hypotheses per patch; PACE-FNO TTA uses the PACE-FNO estimate as its only initialization and performs 5 steps. The reported runtime measures only canonicalizer execution time per patch and excludes data loading and backbone inference, so it measures alignment cost rather than end-to-end clinical throughput.

Table 7: Camelyon17 TTA ablation on hospital-2 OOD test (mean ± standard deviation across three seeds). Additional time is relative to one-shot PACE-FNO.

Variant	Δ CNN accuracy (pp)	Δ FNO accuracy (pp)	Additional time (ms/patch)
PACE-FNO	0.00	0.00	0.000
PACE-FNO + TTA (moment)	+1.75	+2.33	+0.344
PACE-FNO + TTA (learned)	+3.75	+8.05	+0.575

Figure 17 separates source-hospital ID validation from the two held-out hospitals. The observed accuracy degradation motivates evaluating canonicalization, but hospital 1 is not interchangeable with hospital 2; the primary number throughout the paper remains hospital-2 OOD test. Figure 18 visualizes the three-seed TTA ablation in Table 7. The large CNN standard deviations preclude a statistical-significance claim.

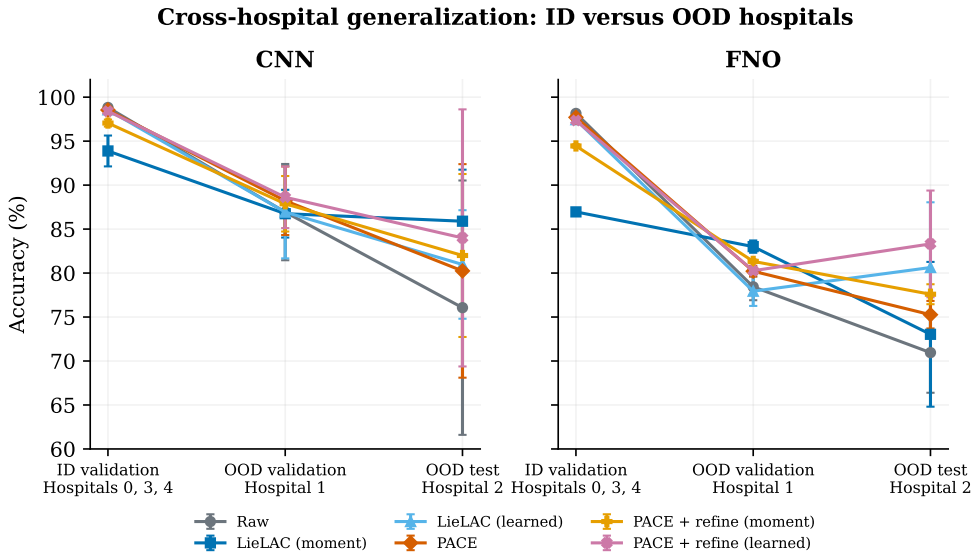


Figure 17: Camelyon17 ID-to-OOD accuracy across source hospitals (0, 3, 4) and held-out hospitals (1, 2).

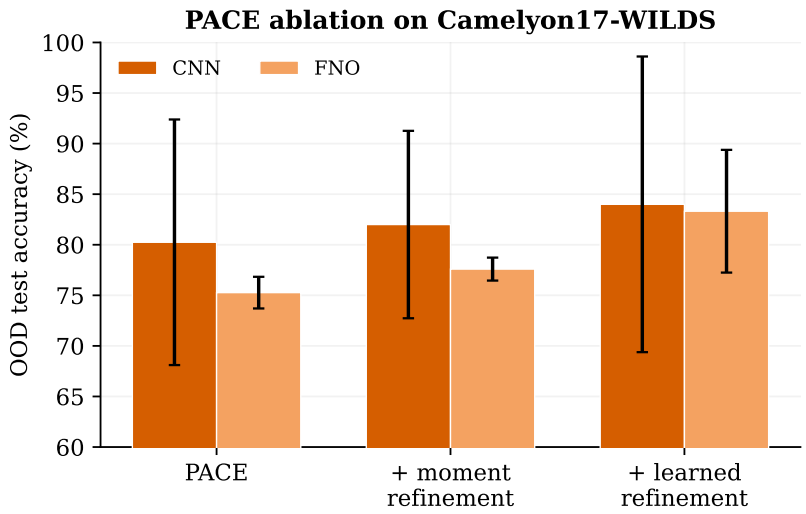


Figure 18: PACE-FNO TTA ablation on Camelyon17 hospital-2 OOD test. Error bars are standard deviations over three seeds.

B.10 Digit protocol, fairness, and timing

The digit experiment retains clean MNIST as an ID control, uses affNIST for affine transfer, and treats homNIST as an evaluation-only projective-shift benchmark. affNIST train and validation sets are partitioned by original digit identity; homNIST is never used for training or model selection. The learned PACE-FNO encoder receives images only at inference. Affine matrices or homographies supervise source-side frame learning when available and are used separately for oracle evaluation, never supplied to the learned test-time encoder.

Training uses ID/source images and finite, label-preserving synthetic nuisance transformations. The synthetic transformations provide alignment supervision; they are not OOD examples and do not cover the full support of an external OOD distribution. Camelyon17 evaluates natural hospital variation, while affNIST and homNIST retain their official independently generated geometric shifts.

The comparison uses 10,000 samples per benchmark with subset seed 42. The digit classifier follows the convolutional MNIST setting (LeCun et al., 1998); affNIST uses an affine action and homNIST uses a projective action. Full LieLAC uses 100 optimization steps and 64 hypotheses. PACE-FNO with TTA starts from its one-shot estimate and takes the listed additional steps. Latency is end-to-end, excludes data loading, and uses synchronized accelerator timing boundaries. PACE-FNO amortizes frame estimation, whereas LieLAC optimizes each sample.

Table 8 records the split roles and inference budgets used in this comparison.

Table 8: Digit evaluation and inference budgets. "Oracle frame" is used only for supervised source-side training targets and separate oracle diagnostics, never as test-time input to PACE-FNO.

Benchmark	Split role	Test-time protocol
MNIST	ID retention control	Image only; identity geometry
affNIST	Affine OOD transfer	Image only; $\mathrm{Aff}(2)$ PACE-FNO; 20-step PACE-FNO with TTA or LieLAC
homNIST	Evaluation-only projective OOD	Image only; $\mathrm{PGL}(3)$ PACE-FNO; 20-step PACE-FNO with TTA or LieLAC

B.11 Diagnostic visualizations

The Camelyon17 montage in Figure 5a selects patches with large cross-method or cross-backbone score disagreement while preferring distinct slides. The digit montage in Figure 5b follows a fixed recovery-oriented rule. These montages are qualitative diagnostics, not random samples, and they do not enter the accuracy calculations. Tables 16 and 17 contain the complete-test-set metrics.

B.12 WeatherBench2 protocol

Table 18 gives the frozen regional WeatherBench2 comparison at a direct $+72$-h lead. It includes the deterministic references, FNO, PACE-FNO, and SFNO. Learned-model entries are means and standard deviations over seeds $\{42, 1234, 3407\}$; model selection uses validation only. The data-role audit, aggregate metrics, and fixed extreme-case fields appear in the main paper.

Table 19 records the selection epoch and runtime. SFNO has the lowest batch-24 memory use and the highest throughput in this benchmark. PACE-FNO is slightly slower and uses more memory than FNO. The memory column comes from the batch-24 CUDA microbenchmark, not from the 30-epoch training logs.

C Additional Empirical Results

This section gives the complete three-seed measurements, runtimes, parameter counts, and diagnostic evaluations behind the main-text comparisons. Benchmark definitions and transformation supports are given in Appendix B.

C.1 Periodic PDE benchmarks

Tables 9–12 give the complete three-seed results for the four core PDE benchmarks.

Table 9: 1-D Burgers under cross-resolution group shift (resolution $1024 \to 2048$). For the FNO and PACE-FNO rows, OOD evaluation time is the timed evaluation loop after the OOD tensors are loaded; it includes device transfers, forward passes, metric computation, and prediction caching, but not data construction or checkpoint loading. It is not a pure backbone-forward latency measurement. The LieLAC+FNO row is the dedicated same-split comparison; its ID error and training time were not recorded in the comparison summary, and its inference time follows that run's timing protocol.

Model	ID relative error ↓	OOD relative error ↓	Training (s/epoch)	OOD evaluation (s)
FNO (no aug.)	0.0662 ± 0.0028	1.0352 ± 0.0001	0.4212 ± 0.0095	0.3943 ± 0.0031
FNO+Aug	0.0293 ± 0.0005	0.3222 ± 0.0077	0.7698 ± 0.0062	1.2826 ± 0.0340
PACE-FNO (one-shot)	0.0623 ± 0.0015	0.0267 ± 0.0014	0.4560 ± 0.0030	0.5287 ± 0.0171
PACE-FNO + TTA	0.0623 ± 0.0015	$\mathbf{0.0259 \pm 0.0007}$	0.4560 ± 0.0030	5.4611 ± 0.1030
GFNO	0.1406 ± 0.0064	1.0244 ± 0.0027	1.0365 ± 0.0411	1.0148 ± 0.0091
LieLAC+FNO	—	0.0342 ± 0.0119	—	8.73

Table 10: 2-D Burgers under cross-resolution group shift (resolution $64^2 \to 128^2$).

Model	ID relative error ↓	OOD relative error ↓	Training (s/epoch)	OOD evaluation (s)
FNO (no aug.)	0.0274 ± 0.0010	0.6137 ± 0.0362	0.6524 ± 0.0157	0.1244 ± 0.0118
FNO+Aug	0.0261 ± 0.0009	0.2035 ± 0.0423	0.7559 ± 0.0177	0.1045 ± 0.0038
PACE-FNO (one-shot)	0.0393 ± 0.0012	0.0755 ± 0.0139	1.5770 ± 0.0066	0.1399 ± 0.0023
PACE-FNO + TTA	0.0393 ± 0.0012	$\mathbf{0.0640 \pm 0.0012}$	1.5770 ± 0.0066	4.4345 ± 0.0409
GFNO	0.0967 ± 0.0054	1.0131 ± 0.0119	4.8613 ± 0.3527	0.7462 ± 0.0080

Table 11: 2-D shallow water under cross-resolution group shift (resolution $64^2 \to 128^2$).

Model	ID relative error ↓	OOD relative error ↓	Training (s/epoch)	OOD evaluation (s)
FNO (no aug.)	0.0827 ± 0.0009	0.7260 ± 0.0003	0.7024 ± 0.0474	0.1501 ± 0.0068
FNO+Aug	0.0769 ± 0.0010	0.5136 ± 0.0010	0.8502 ± 0.0082	0.1503 ± 0.0012
PACE-FNO (one-shot)	0.0491 ± 0.0002	$\mathbf{0.1069 \pm 0.0168}$	2.6139 ± 0.0046	0.2952 ± 0.0349
PACE-FNO + TTA	0.0491 ± 0.0002	0.1071 ± 0.0168	2.6139 ± 0.0046	8.1388 ± 0.0855
GFNO	0.0307 ± 0.0007	0.7593 ± 0.0071	6.7509 ± 0.0355	0.7161 ± 0.0401

Table 12: Spatiotemporal Navier-Stokes under cross-resolution group shift (resolution $64^2 \to 128^2$).

Model	ID relative error ↓	OOD relative error ↓	Training (s/epoch)	OOD evaluation (s)
FNO (no aug.)	0.0377 ± 0.0043	0.6422 ± 0.0302	10.6448 ± 0.0392	1.8308 ± 0.6626
FNO+Aug	0.0266 ± 0.0059	0.2840 ± 0.2001	10.7371 ± 0.0281	1.9761 ± 0.5997
PACE-FNO (one-shot)	0.0129 ± 0.0006	0.2766 ± 0.0353	16.5571 ± 0.3631	3.3728 ± 0.1381
PACE-FNO + TTA	0.0129 ± 0.0006	$\mathbf{0.2593 \pm 0.0522}$	16.5571 ± 0.3631	10.0376 ± 0.0418
GFNO	0.0459 ± 0.0016	0.4091 ± 0.0118	55.9359 ± 0.0496	9.2956 ± 0.0585

C.2 BACKBONE TRANSFER

We apply the same alignment wrapper to the Mamba Neural Operator, Continuous Vision Transformer (CViT), and Transolver predictors (Cheng et al., 2026; Wang et al., 2025; Wu et al., 2024). The "PACE-" names denote our wrapped variants, not new versions of the cited backbones. MambaGalerkin is compared with its base and augmented versions; CViT+Aug and Transolver+Aug are not included, so those comparisons are not causal comparisons with matched data augmentation.

Table 13 reports the matched backbone-transfer comparisons, including runtime measurements. All listed predictors use one-shot input-to-output inference; Burgers is trained at 64^2 and evaluated at 128^2, and SWE uses the same resolution pair. The timed OOD column is milliseconds per sample. Runtime measurements were made on the same RTX 4080 setup described in Appendix B.1.

Table 13: Same-PDE backbone comparisons (three seeds), with one-shot protocol and runtime. Each PACE variant retains the listed predictor and adds only the group-coordinate estimator and prescribed frame actions. Training time is seconds per epoch; OOD latency is milliseconds per sample. Dashes indicate that the corresponding runtime was not reported.

PDE / protocol	Method	ID error ↓	OOD error ↓	Backbone params	Extra params	Total params	Train s/epoch	OOD ms/sample
2-D Burgers, one-shot $64^2 \to 128^2$	MambaGalerkin	0.05006 ± 0.00167	0.73061 ± 0.05598	4,854,177	0	4,854,177	20.757 ± 0.062	22.772
	MambaGalerkin+Aug	$\mathbf{0.03449 \pm 0.00091}$	0.32505 ± 0.11714	4,854,177	0	4,854,177	20.862 ± 0.112	23.494
	PACE-MambaGalerkin	0.04632 ± 0.00203	$\mathbf{0.04246 \pm 0.00048}$	4,854,177	297,156	5,151,333	23.130 ± 0.248	22.876
	CViT	$\mathbf{0.08819 \pm 0.00499}$	0.77986 ± 0.02018	1,834,177	0	1,834,177	—	—
	PACE-CViT	0.18956 ± 0.17901	$\mathbf{0.15340 \pm 0.05810}$	1,834,177	297,156	2,131,333	—	—
2-D SWE, one-shot $64^2 \to 128^2$	Transolver	$\mathbf{0.12479 \pm 0.00302}$	0.78036 ± 0.06026	11,229,251	0	11,229,251	20.991 ± 0.171	20.064 ± 0.575
	PACE-Transolver	0.13924 ± 0.01335	$\mathbf{0.33906 \pm 0.00456}$	11,229,251	297,732	11,526,983	32.194 ± 0.568	18.061 ± 0.007

C.3 IRREGULAR-DOMAIN AND ROLLOUT TESTS

Table 14 reports the irregular-domain and rollout scope tests.

Table 14: Irregular-domain and rollout results (three seeds). Poisson uses relative L_2; KS OOD values are mean-centered relative L_2 over 300 closed-loop steps.

Problem	Method	ID	Parameter/ID OOD	Geometric/shift OOD	Additional diagnostic
3-D Poisson-ball	Transolver	0.0212 ± 0.0007	0.0281 ± 0.0008	0.3000 ± 0.0470 (90°)	0.2344 ± 0.0044 (forcing)
	PACE-Transolver	$\mathbf{0.0180 \pm 0.0002}$	$\mathbf{0.0262 \pm 0.0007}$	$\mathbf{0.1815 \pm 0.0178}$ (90°)	0.2330 ± 0.0044 (forcing)
1-D KS, 300 steps	FNO	0.38783 ± 0.05581	—	0.98605 ± 0.06473	1.37071 ± 0.11634 (final step)
	FNO+Aug	0.45730 ± 0.09516	—	0.46978 ± 0.02451	1.02729 ± 0.07122 (final step)
	PACE-FNO	$\mathbf{0.35860 \pm 0.07267}$	—	$\mathbf{0.35919 \pm 0.07196}$	$\mathbf{0.82025 \pm 0.15761}$ (final step)

C.4 ABLATION AND OBSERVATION DIAGNOSTICS

Table 15 reports the causal control, noise, and sparse-observation checks.

Table 15: Causal control and observation diagnostics on 2-D Burgers (three seeds; standard-resolution OOD, distinct from the cross-resolution results in Table 10). Subsequent rows reuse the top block's checkpoints.

Setting	Metric	FNO	FNO+Aug	FNO+AuxEstimator	PACE-FNO
Standard OOD	Rel. L_2	0.63950 ± 0.03559	0.18080 ± 0.01653	0.66021 ± 0.03644	$\mathbf{0.05863 \pm 0.00287}$
Standard OOD	Centered Rel. L_2	0.97141 ± 0.05360	0.75069 ± 0.03450	0.97814 ± 0.01541	$\mathbf{0.17144 \pm 0.00974}$
Noise $\alpha = 0.10$	Rel. L_2	0.63947 ± 0.03567	0.18090 ± 0.01655	0.66016 ± 0.03650	$\mathbf{0.05867 \pm 0.00291}$
Random sparse, 50%	Rel. L_2	0.6369 ± 0.0344	0.3025 ± 0.0129	0.6622 ± 0.0413	$\mathbf{0.2397 \pm 0.0031}$
Structured missing, 50%	Rel. L_2	0.6808 ± 0.0259	0.3560 ± 0.0119	0.7016 ± 0.0316	$\mathbf{0.3244 \pm 0.0016}$

C.5 Complete Accuracy and Latency Results

C.5.1 Camelyon17: cross-hospital stain shift

Table 16: Camelyon17 held-out hospital-2 OOD accuracy (mean ± standard deviation across three seeds). "CNN accuracy" and "FNO accuracy" denote separate classifier-backbone variants; canonicalization time excludes backbone inference.

Method	CNN accuracy (%)	FNO accuracy (%)	Time (ms/patch)
Raw	76.06 (±) 14.46	70.95 (±) 4.56	0.000
LieLAC (moment)	**85.90** (±) **5.86**	73.04 (±) 8.23	4.823
LieLAC (learned)	80.98 (±) 6.17	80.63 (±) 7.42	21.854
PACE-FNO	80.24 (±) 12.14	75.27 (±) 1.57	**0.022**
PACE-FNO + TTA (learned)	84.00 (±) 14.62	**83.31** (±) **6.07**	0.597

Figure 19 places the Camelyon17 accuracy and canonicalization latency measurements on the same axes.

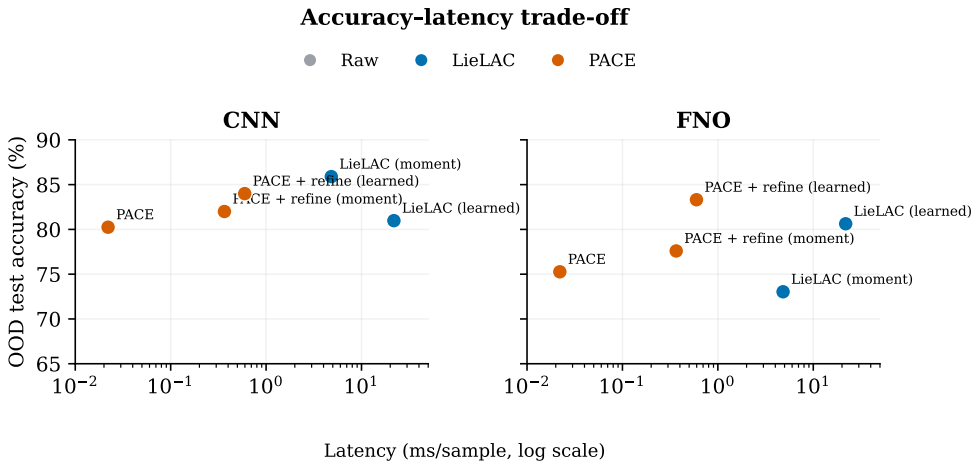


Figure 19: Camelyon17 OOD accuracy–latency trade-off for canonicalization on the held-out hospital.

C.5.2 Digits: affine and projective shifts

Figure 20 gives the corresponding accuracy–latency trade-off for the digit benchmarks.

Table 17: PACE-FNO results on digit benchmarks (mean ± standard deviation across three seeds); latency is end-to-end.

Dataset	Method	Accuracy (%)	Latency (ms/sample)
MNIST	Raw	99.050 (±) 0.062	0.066
	PACE-FNO	99.200 (±) 0.030	0.200
	PACE-FNO + 10 steps	**99.270** (±) **0.017**	3.023
	LieLAC	99.130 (±) 0.040	286.234
affNIST	Raw	37.743 (±) 0.768	0.066
	PACE-FNO	89.953 (±) 0.254	0.202
	PACE-FNO + 20 steps	92.340 (±) 0.300	5.361
	LieLAC	**95.503** (±) **0.370**	288.592
homNIST	Raw	89.400 (±) 1.109	0.066
	PACE-FNO	97.180 (±) 0.141	0.200
	PACE-FNO + 20 steps	97.673 (±) 0.153	5.435
	LieLAC	**98.433** (±) **0.142**	287.818

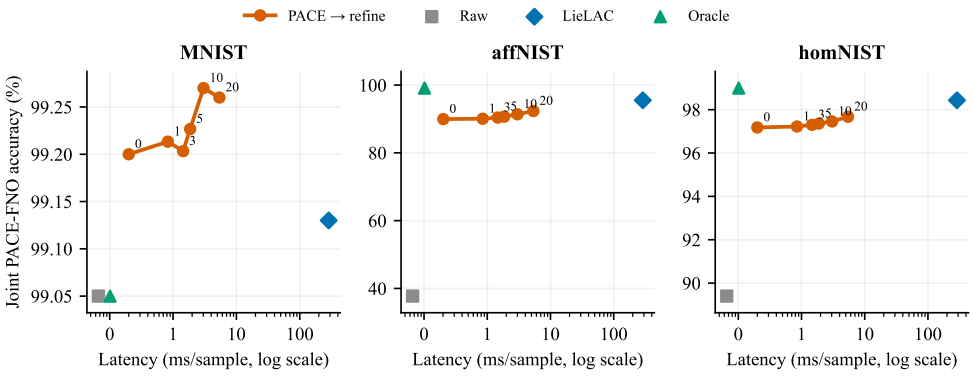


Figure 20: PACE-FNO accuracy–latency trade-off on digit benchmarks.

C.6 WeatherBench2 empirical metrics

Table 18: WeatherBench2 ERA5 +72-h results. Learned models report mean ± standard deviation over three seeds; deterministic baselines have zero seed variance. An em-dash denotes that the anomaly correlation coefficient (ACC) is undefined for the static train-climatology baseline.

Split	Metric	Train climatology	Persistence	FNO	PACE-FNO	SFNO
ID test	T2m RMSE (K)	2.471 ± 0.000	2.761 ± 0.000	2.078 ± 0.005	2.078 ± 0.005	2.886 ± 0.022
ID test	Z500 RMSE (m)	66.83 ± 0.00	71.04 ± 0.00	51.42 ± 0.34	51.41 ± 0.35	56.59 ± 0.08
ID test	T2m ACC	—	0.381 ± 0.000	0.561 ± 0.003	0.561 ± 0.003	0.284 ± 0.009
ID test	Z500 ACC	—	0.443 ± 0.000	0.641 ± 0.005	0.641 ± 0.005	0.541 ± 0.002
ID test	Z500 phase error (deg)	0.00 ± 0.00	56.14 ± 0.00	29.88 ± 0.53	29.88 ± 0.51	35.80 ± 1.07
Time OOD	T2m RMSE (K)	2.504 ± 0.000	2.804 ± 0.000	2.055 ± 0.008	2.055 ± 0.008	2.880 ± 0.017
Time OOD	Z500 RMSE (m)	66.53 ± 0.00	70.53 ± 0.00	49.68 ± 0.06	49.67 ± 0.05	54.75 ± 0.29
Time OOD	T2m ACC	—	0.376 ± 0.000	0.585 ± 0.006	0.586 ± 0.005	0.309 ± 0.003
Time OOD	Z500 ACC	—	0.437 ± 0.000	0.667 ± 0.002	0.667 ± 0.001	0.576 ± 0.003
Time OOD	Z500 phase error (deg)	0.00 ± 0.00	57.47 ± 0.00	27.11 ± 0.18	27.11 ± 0.17	32.18 ± 0.41
Extreme OOD	T2m RMSE (K)	2.468 ± 0.000	2.715 ± 0.000	1.994 ± 0.001	1.993 ± 0.002	2.833 ± 0.016
Extreme OOD	Z500 RMSE (m)	63.57 ± 0.00	68.25 ± 0.00	47.80 ± 0.07	47.77 ± 0.10	52.88 ± 0.45
Extreme OOD	T2m ACC	—	0.403 ± 0.000	0.601 ± 0.002	0.602 ± 0.001	0.326 ± 0.003
Extreme OOD	Z500 ACC	—	0.424 ± 0.000	0.663 ± 0.002	0.664 ± 0.002	0.566 ± 0.009
Extreme OOD	Z500 phase error (deg)	0.00 ± 0.00	55.66 ± 0.00	26.14 ± 0.18	26.06 ± 0.20	30.66 ± 1.37

C.7 WeatherBench2 efficiency

Table 19: Training selection and batch-24 runtime benchmark. Best epoch and wall time are mean ± standard deviation over three seeds. Throughput and memory are from the recorded batch-24 CUDA microbenchmark.

Model	Best epoch	Wall time (min)	Step (s)	Samples/s	Peak GPU (GiB)
FNO	17.7 ± 1.2	58.9 ± 0.7	0.036	669.0	1.39
PACE-FNO	17.7 ± 1.2	62.9 ± 2.7	0.040	600.8	1.59
SFNO	21.7 ± 0.9	62.0 ± 0.1	0.026	919.1	0.62

D Theoretical Details and Proofs

Proof roadmap. We first relate exact compact-subgroup averaging to covering numbers. We then establish the input–terminal relation, decompose prediction error into canonical approximation and field-space alignment terms, and state local TTA stability. Each result is conditional on its displayed assumptions.

D.1 Compact-subgroup averaging and covering numbers

The following result concerns exact compact-subgroup averaging, whereas the experimental supports are bounded sets and need not be groups. A general bounded sampling support therefore has no normalized Haar measure.

Let ρ denote the data distribution over $\mathcal{A} \times \mathcal{U}$, and let $\ell : \mathcal{U} \times \mathcal{U} \to [0, B]$ be a bounded loss function. For a learned operator $F \in \mathcal{F}$, the population risk and empirical risk are

$$\mathcal{R}(F) = \mathbb{E}_{(a,u)\sim\rho}\, \ell(F(a), u), \qquad \widehat{\mathcal{R}}_N(F) = \frac{1}{N}\sum_{i=1}^{N} \ell(F(a_i), u_i).$$

Definition D.1 (Generalization Error)**.** The generalization error of an operator $F \in \mathcal{F}$ with respect to N i.i.d. samples is

$$\mathrm{Gen}_N(F) := \mathcal{R}(F) - \widehat{\mathcal{R}}_N(F).$$

The uniform generalization error over the class is $\sup_{F\in\mathcal{F}} |\mathrm{Gen}_N(F)|$.

Definition D.2 (Covering Number)**.** Let $(\mathcal{F}, d)$ be a metric space. For $\epsilon > 0$, the ϵ-covering number $\mathcal{N}(\mathcal{F}, \epsilon, d)$ is the minimum cardinality of a set $\{F_1, \ldots, F_m\} \subseteq \mathcal{F}$ such that for every $F \in \mathcal{F}$ there exists $j \in [m]$ with $d(F, F_j) \le \epsilon$.

For operator learning, the natural metric is the uniform deviation over the input domain $\mathcal{X}$ supporting the data:

$$d_\infty(F_1, F_2) := \sup_{a\in\mathcal{X}} \|F_1(a) - F_2(a)\|_{\mathcal{U}}.$$

Theorem D.1 (Compact-Subgroup Orbit Averaging)**.** *Let $\mathcal{A}$ and $\mathcal{U}$ be Banach spaces, let $H \leq G$ be a compact subgroup with normalized Haar probability measure ν_H, and let $\mathcal{X}_H \subseteq \mathcal{A}$ be H-invariant. Assume that $\mathcal{C}^{\mathcal{A}} : H \to \mathrm{Iso}(\mathcal{A})$ is a continuous action by bijective isometries and that $\mathcal{C}^{\mathcal{U}} : H \to \mathcal{L}(\mathcal{U})$ is a strongly continuous linear isometric representation. For a class $\mathcal{F}$ of continuous operators $\mathcal{X}_H \to \mathcal{U}$, define*

$$(\Pi_H F)(a) := \int_H \mathcal{C}^{\mathcal{U}}_{h^{-1}} F(\mathcal{C}^{\mathcal{A}}_h a)\, d\nu_H(h), \qquad \mathcal{F}_H := \Pi_H(\mathcal{F}).$$

Then $\Pi_H F$ is H-equivariant and Π_H is non-expansive under d_∞. Consequently, for any $r > 0$,

$$\mathcal{N}(\mathcal{F}_H, r, d_\infty) \leq \mathcal{N}(\mathcal{F}, r, d_\infty).$$

If $\eta_H(F) := \sup_{h\in H,\, a\in\mathcal{X}_H} \|(F \circ \mathcal{C}^{\mathcal{A}}_h)(a) - (\mathcal{C}^{\mathcal{U}}_h \circ F)(a)\|_{\mathcal{U}} \leq \eta$, then $d_\infty(F, \Pi_H F) \leq \eta$. For an L_ℓ-Lipschitz loss evaluated on the same sample,

$$|\mathrm{Gen}_N(F)| \leq |\mathrm{Gen}_N(\Pi_H F)| + 2L_\ell\eta.$$

Lemma D.2 (Covering Number to Generalization Error)**.** *Let $\mathcal{X} \subseteq \mathcal{A}$, let ρ be supported on $\mathcal{X} \times \mathcal{U}$, and let $\mathcal{F}$ be a class of operators from $\mathcal{X}$ to $\mathcal{U}$. Define $d_\infty(F_1, F_2) = \sup_{a\in\mathcal{X}} \|F_1(a) - F_2(a)\|_{\mathcal{U}}$. Assume that the loss ℓ is L_ℓ-Lipschitz in its first argument and bounded in $[0, B]$, and that $\mathcal{N}(\mathcal{F}, \epsilon, d_\infty) < \infty$. For any $\epsilon > 0$ and $\delta \in (0, 1)$, with probability at least $1 - \delta$ over the draw of N i.i.d. samples,*

$$\sup_{F\in\mathcal{F}} \left|\mathcal{R}(F) - \widehat{\mathcal{R}}_N(F)\right| \leq 2L_\ell\epsilon + B\sqrt{\frac{\log\big(2\,\mathcal{N}(\mathcal{F}, \epsilon, d_\infty)/\delta\big)}{2N}}.$$

Proof. Take an ϵ-cover $\{F_1, \ldots, F_m\}$ of $\mathcal{F}$ under d_∞ with $m = \mathcal{N}(\mathcal{F}, \epsilon, d_\infty)$. For any $F \in \mathcal{F}$, let F_j be the cover element with $d_\infty(F, F_j) \leq \epsilon$. The L_ℓ-Lipschitz property of ℓ gives

$$|\ell(F(a), u) - \ell(F_j(a), u)| \leq L_\ell \|F(a) - F_j(a)\|_{\mathcal{U}} \leq L_\ell\epsilon,$$

so both population and empirical risks differ by at most $L_\ell\epsilon$ from those of F_j. Hence

$$|\mathrm{Gen}_N(F)| \leq 2L_\ell\epsilon + |\mathrm{Gen}_N(F_j)|.$$

For a fixed F_j, Hoeffding's inequality for losses bounded in $[0, B]$ yields

$$\Pr\big(|\mathrm{Gen}_N(F_j)| \geq t\big) \leq 2\exp\left(-\frac{2Nt^2}{B^2}\right).$$

A union bound over the m cover elements and the choice $t = B\sqrt{\frac{\log(2m/\delta)}{2N}}$ complete the proof. □

Remark D.1 (Scope of the compact-subgroup bound)**.** Theorem D.1 establishes that the exact compact-subgroup averaging operator Π_H is non-expansive under d_∞, which yields

$$\mathcal{N}(\mathcal{F}_H, \epsilon, d_\infty) \leq \mathcal{N}(\mathcal{F}, \epsilon, d_\infty).$$

Substituting this inequality into Lemma D.2, the resulting *upper bound* for the explicitly projected class $\mathcal{F}_H = \Pi_H(\mathcal{F})$ is no larger than the analogous upper bound for $\mathcal{F}$ at the same sample size. This comparison is between bounds; it does not order the realized generalization gaps of two trained models. If F has compact-subgroup commutation residual $\eta_H(F) \leq \eta$, the proof gives $d_\infty(F, \Pi_H F) \leq \eta$, and the L_ℓ-Lipschitz property of ℓ gives

$$|\mathcal{R}(F) - \mathcal{R}(\Pi_H F)| \leq L_\ell\eta, \qquad |\widehat{\mathcal{R}}_N(F) - \widehat{\mathcal{R}}_N(\Pi_H F)| \leq L_\ell\eta. \tag{1}$$

Combining equation 1 with Lemma D.2 applied to $\Pi_H F \in \mathcal{F}_H$ and using the triangle inequality yields

$$|\mathrm{Gen}_N(F)| \leq \underbrace{2L_\ell\epsilon + B\sqrt{\frac{\log\big(2\,\mathcal{N}(\mathcal{F}_H, \epsilon, d_\infty)/\delta\big)}{2N}}}_{\text{generalization bound for } \mathcal{F}_H} + 2L_\ell\eta.$$

PACE-FNO is not constructed by applying Π_H to an arbitrary operator, and this theorem does not prove that its learned hypothesis class equals $\mathcal{F}_H$. Nor does it identify $\eta_H(F)$ with coordinate-estimation error without an additional argument. For PACE-FNO, the directly applicable result is instead Theorem D.7, whose field-space alignment residuals remain explicit and must be assessed empirically.

D.2 PROOFS

The standard FNO approximation result requires a compact input set and a continuous target operator. We restate the needed assumptions in the notation used here, then prove the two bounds used in the main text.

Induced terminal group element. For an exact transformation of the physical solution operator, an admissible initial element g_0 from the compact support S defined below and a terminal element g_T satisfy the intertwining relation

$$\mathcal{G}^\dagger \circ \mathcal{C}_{g_0} = \mathcal{C}_{g_T} \circ \mathcal{G}^\dagger.$$

For exact periodic translations, $g_T = g_0$. For the Galilean transformations used in the Burgers and SWE settings, the background velocity transports the field by $\mathbf{v}_{\mathrm{bg}}T$ over the time horizon T, inducing $g_T = \exp(\xi_0 + \mathbf{v}_{\mathrm{bg}}T \cdot \nabla)$. Arbitrary-angle rotations on the discretized square torus are excluded from this exact statement. The relation is applied in Theorem D.7 to separate canonical approximation and field-space alignment residuals.

Lemma D.3 (Compactness and Orbit Saturation)**.** *Let $D = \mathbb{T}^d$ be a periodic domain, and let the input and output spaces be $\mathcal{A} = L^2(D;\mathbb{R}^{d_a})$ and $\mathcal{U} = L^2(D;\mathbb{R}^{d_u})$. Assume that the canonical training measure μ is supported on a compact subset $\mathcal{A}_c \subset \mathcal{A}$. Let $S \subset G$ be a compact set of transformation elements used to define the bounded OOD family; S is not assumed to be a subgroup. Define*

$$\mathcal{A}_{\mathrm{OOD}} := \{\mathcal{C}_g(a) : a \in \mathcal{A}_c, g \in S\} \subset \mathcal{A}.$$

If $(g,a) \mapsto \mathcal{C}_g(a)$ is continuous, then $\mathcal{A}_{\mathrm{OOD}}$ is compact as the continuous image of the compact product $S \times \mathcal{A}_c$. For Theorem D.7, also assume that $\mathcal{G}^\dagger$ is continuous on $\mathcal{A}_c$ and satisfies the exact input–terminal intertwining relation for every transformation in S.

Under these assumptions, the standard Fourier Neural Operator approximation result applies on the canonical compact set.

Theorem D.4. *If $\mathcal{G}^\dagger|_{\mathcal{A}_c}$ is continuous, then for any $\epsilon > 0$ there exists a Fourier Neural Operator $\mathcal{G}_\theta$ of sufficient depth or width such that*

$$\sup_{a \in \mathcal{A}_c} \|\mathcal{G}_\theta(a) - \mathcal{G}^\dagger(a)\|_{\mathcal{U}} \le \epsilon.$$

Remark D.2. Theorem D.4 summarizes the standard compact-set approximation setting for neural operators. Its relevance to OOD prediction is limited: when an input field is transformed by an element $g \in S$, the transformed data $a_{\mathrm{OOD}} = \mathcal{C}_g(a)$ may fall outside the original canonical compact set $\mathcal{A}_c$. The theorem then no longer applies directly to that transformed input. PACE-FNO applies an estimated inverse transformation to map the observation toward $\mathcal{A}_c$. The resulting guarantee is conditional on the field-space alignment residual; it neither proves recovery of the generating coordinate nor requires that coordinate to be identifiable.

For every $a \in \mathcal{A}_{\mathrm{OOD}}$, fix an admissible witness $(a_c(a), g_0(a), g_T(a))$ as in Theorem D.7. The witness need not be unique. The input field-space alignment residual is

$$\epsilon_{\mathrm{geo},0} := \sup_{a \in \mathcal{A}_{\mathrm{OOD}}} \|\mathcal{C}^{-1}_{\hat{g}_0(a)}(a) - a_c(a)\|_{\mathcal{A}},$$

and the terminal alignment error is

$$\epsilon_{\mathrm{geo},T} := \sup_{a \in \mathcal{A}_{\mathrm{OOD}}} \|\mathcal{C}_{\hat{g}_T(a)}(\mathcal{G}^\dagger(a_c(a))) - \mathcal{C}_{g_T(a)}(\mathcal{G}^\dagger(a_c(a)))\|_{\mathcal{U}}.$$

Lemma D.5 (Lipschitz Continuity of the Output Action)**.** *Assume that each transformation $\mathcal{C}_g$ used in the exact input–terminal relation is Lipschitz on the output space $\mathcal{U}$. Exact continuum translations are isometries in L^2. More generally, assume that there exists a uniform constant $K_C \ge 1$ such that, for any g arising as an estimated or admissible terminal element and any $u_1, u_2 \in \mathcal{U}$,*

$$\|\mathcal{C}_g(u_1) - \mathcal{C}_g(u_2)\|_{\mathcal{U}} \le K_C \|u_1 - u_2\|_{\mathcal{U}}.$$

Lemma D.6 (Lipschitz Continuity of the Learned Operator)**.** *Assume that the trained Fourier Neural Operator $\mathcal{G}_\theta$ is Lipschitz on a neighborhood of $\mathcal{A}_c$ that contains the pulled-back estimates considered in the proof. That is, for some $L_\theta \ge 1$ and all a_1, a_2 in this neighborhood,*

$$\|\mathcal{G}_\theta(a_1) - \mathcal{G}_\theta(a_2)\|_{\mathcal{U}} \le L_\theta \|a_1 - a_2\|_{\mathcal{A}}.$$

Theorem D.7 (Error Decomposition under Symmetry-Induced Shifts)**.** *Assume the hypotheses of Lemma D.3 (compact canonical support and continuous action), Lemmas D.5–D.6 (isometric/Lipschitz output action and Lipschitz learned operator), and that the* chosen *FNO satisfies* $\sup_{a_c \in \mathcal{A}_c} \|\mathcal{G}_\theta(a_c) - \mathcal{G}^\dagger(a_c)\|_{\mathcal{U}} \le \epsilon$*. The universal-approximation result in Theorem D.4 only motivates the existence of such an FNO; it does not assert that optimization finds it. For each* $a \in \mathcal{A}_{\mathrm{OOD}}$*, fix any admissible witness* $(a_c(a), g_0(a), g_T(a))$ *satisfying*

$$a = \mathcal{C}_{g_0(a)}(a_c(a)), \qquad \mathcal{G}^\dagger(a) = \mathcal{C}_{g_T(a)}(\mathcal{G}^\dagger(a_c(a))).$$

No uniqueness of this witness is assumed. Then the prediction error of PACE-FNO over the bounded OOD family satisfies

$$\sup_{a \in \mathcal{A}_{\mathrm{OOD}}} \left\| \mathcal{C}_{\hat{g}_T} \circ \mathcal{G}_\theta \circ \mathcal{C}_{\hat{g}_0}^{-1}(a) - \mathcal{G}^\dagger(a) \right\|_{\mathcal{U}} \le K_C \epsilon + M_0 \epsilon_{\mathrm{geo},0} + \epsilon_{\mathrm{geo},T},$$

where $\hat{g}_0 = \exp(\hat{\xi}_0)$*,* $\hat{g}_T = \exp(\hat{\xi}_T)$*, and* $M_0 = K_C L_\theta$ *with* K_C *the output-action Lipschitz constant and* L_θ *the FNO Lipschitz constant. Here* ϵ *is the canonical approximation error, while* $\epsilon_{\mathrm{geo},0}$ *and* $\epsilon_{\mathrm{geo},T}$ *are the field-space input and terminal alignment residuals defined above. For an isometric output action,* $K_C = 1$*.*

Proof of Theorem D.7. Let $a \in \mathcal{A}_{\mathrm{OOD}}$ be arbitrary, and use the fixed admissible witness $(a_c(a), g_0(a), g_T(a))$. To simplify notation within this proof, write it as (a_c, g_0, g_T). By definition,

$$a = \mathcal{C}_{g_0}(a_c), \qquad \mathcal{G}^\dagger(a) = \mathcal{C}_{g_T}(\mathcal{G}^\dagger(a_c)).$$

The triangle inequality separates the total prediction error into input-alignment, canonical-approximation, and terminal-alignment terms:

$$\begin{aligned}
\|\mathcal{C}_{\hat{g}_T}(\mathcal{G}_\theta(\mathcal{C}_{\hat{g}_0}^{-1}(a))) - \mathcal{C}_{g_T}(\mathcal{G}^\dagger(a_c))\|_{\mathcal{U}} = \Big\| & \mathcal{C}_{\hat{g}_T}(\mathcal{G}_\theta(\mathcal{C}_{\hat{g}_0}^{-1}(a))) - \mathcal{C}_{\hat{g}_T}(\mathcal{G}_\theta(a_c)) \\
& + \mathcal{C}_{\hat{g}_T}(\mathcal{G}_\theta(a_c)) - \mathcal{C}_{\hat{g}_T}(\mathcal{G}^\dagger(a_c)) \\
& + \mathcal{C}_{\hat{g}_T}(\mathcal{G}^\dagger(a_c)) - \mathcal{C}_{g_T}(\mathcal{G}^\dagger(a_c)) \Big\|_{\mathcal{U}} \\
\le & \underbrace{\|\mathcal{C}_{\hat{g}_T}(\mathcal{G}_\theta(\mathcal{C}_{\hat{g}_0}^{-1}(a))) - \mathcal{C}_{\hat{g}_T}(\mathcal{G}_\theta(a_c))\|_{\mathcal{U}}}_{\text{(I) Initial alignment error}} \\
& + \underbrace{\|\mathcal{C}_{\hat{g}_T}(\mathcal{G}_\theta(a_c)) - \mathcal{C}_{\hat{g}_T}(\mathcal{G}^\dagger(a_c))\|_{\mathcal{U}}}_{\text{(II) Canonical-domain approximation error}} \\
& + \underbrace{\|\mathcal{C}_{\hat{g}_T}(\mathcal{G}^\dagger(a_c)) - \mathcal{C}_{g_T}(\mathcal{G}^\dagger(a_c))\|_{\mathcal{U}}}_{\text{(III) Terminal group-action mismatch}}.
\end{aligned}$$

For the first term, Lemmas D.5 and D.6 give

$$\text{(I)} \le K_C \|\mathcal{G}_\theta(\mathcal{C}_{\hat{g}_0}^{-1}(a)) - \mathcal{G}_\theta(a_c)\|_{\mathcal{U}} \le K_C \cdot L_\theta \|\mathcal{C}_{\hat{g}_0}^{-1}(a) - a_c\|_{\mathcal{A}}.$$

By the definition of $\epsilon_{\mathrm{geo},0}$, this term is bounded by $M_0 \epsilon_{\mathrm{geo},0}$, where $M_0 := K_C L_\theta$. For the canonical-domain approximation term, Lemma D.5 and the theorem's assumed canonical approximation bound give

$$\text{(II)} \le K_C \|\mathcal{G}_\theta(a_c) - \mathcal{G}^\dagger(a_c)\|_{\mathcal{U}} \le K_C \epsilon.$$

The third term is exactly bounded by the definition of the terminal alignment error:

$$\text{(III)} \le \epsilon_{\mathrm{geo},T}.$$

Combining the three bounds and taking the supremum over $a \in \mathcal{A}_{\mathrm{OOD}}$ gives

$$\sup_{a \in \mathcal{A}_{\mathrm{OOD}}} \|\mathcal{C}_{\hat{g}_T} \circ \mathcal{G}_\theta \circ \mathcal{C}_{\hat{g}_0}^{-1}(a) - \mathcal{G}^\dagger(a)\|_{\mathcal{U}} \le M_0 \epsilon_{\mathrm{geo},0} + K_C \epsilon + \epsilon_{\mathrm{geo},T}.$$

If the witness satisfies the intertwining relation only up to a uniform residual ϵ_{sym}, insert $\mathcal{C}_{g_T} \mathcal{G}^\dagger(a_c)$ between the prediction and $\mathcal{G}^\dagger(a)$ and apply the triangle inequality once more. This adds exactly ϵ_{sym} to the right-hand side. □

Proof of Theorem D.1. Fix $F \in \mathcal{F}$ and $a \in \mathcal{X}_H$. Because $\mathcal{X}_H$ is H-invariant, the integrand is defined for every $h \in H$. Strong continuity, continuity of F, compactness of H, and completeness of $\mathcal{U}$ ensure that the displayed orbit average is Bochner integrable.

For any $q \in H$, the representation laws, the change of variables $k = hq$, and right invariance of normalized Haar measure give

$$\begin{aligned}(\Pi_H F)(\mathcal{C}_q^{\mathcal{A}} a) &= \int_H \mathcal{C}_{h^{-1}}^{\mathcal{U}} F(\mathcal{C}_{hq}^{\mathcal{A}} a)\, d\nu_H(h) \\ &= \int_H \mathcal{C}_{qk^{-1}}^{\mathcal{U}} F(\mathcal{C}_k^{\mathcal{A}} a)\, d\nu_H(k) \\ &= \mathcal{C}_q^{\mathcal{U}} \int_H \mathcal{C}_{k^{-1}}^{\mathcal{U}} F(\mathcal{C}_k^{\mathcal{A}} a)\, d\nu_H(k) \\ &= \mathcal{C}_q^{\mathcal{U}} (\Pi_H F)(a).\end{aligned}$$

The third equality uses that $\mathcal{C}_q^{\mathcal{U}}$ is bounded and linear, so it commutes with the Bochner integral. Thus $\Pi_H F$ is H-equivariant. Averaging an already equivariant operator returns the same operator, so $\Pi_H^2 = \Pi_H$.

For any $F_1, F_2 \in \mathcal{F}$, linearity and isometry of the output representation imply

$$\begin{aligned}d_\infty(\Pi_H F_1, \Pi_H F_2) &\le \sup_{a \in \mathcal{X}_H} \int_H \left\| \mathcal{C}_{h^{-1}}^{\mathcal{U}} \left(F_1(\mathcal{C}_h^{\mathcal{A}} a) - F_2(\mathcal{C}_h^{\mathcal{A}} a) \right) \right\|_{\mathcal{U}} d\nu_H(h) \\ &= \sup_{a \in \mathcal{X}_H} \int_H \left\| F_1(\mathcal{C}_h^{\mathcal{A}} a) - F_2(\mathcal{C}_h^{\mathcal{A}} a) \right\|_{\mathcal{U}} d\nu_H(h) \\ &\le d_\infty(F_1, F_2),\end{aligned}$$

where the last inequality again uses H-invariance of $\mathcal{X}_H$. Hence Π_H is non-expansive. If $\{F_j\}_{j=1}^m$ is an r-cover of $\mathcal{F}$, then $\{\Pi_H F_j\}_{j=1}^m \subset \mathcal{F}_H$ is an r-cover of $\mathcal{F}_H$. Therefore,

$$\mathcal{N}(\mathcal{F}_H, r, d_\infty) \le \mathcal{N}(\mathcal{F}, r, d_\infty).$$

Lemma D.2 applied on $\mathcal{X}_H$ now gives the stated comparison between the two covering-number upper bounds whenever the relevant covering numbers are finite.

Finally, suppose $\eta_H(F) \le \eta$. For every $h \in H$ and $a \in \mathcal{X}_H$, output isometry gives

$$\left\| \mathcal{C}_{h^{-1}}^{\mathcal{U}} F(\mathcal{C}_h^{\mathcal{A}} a) - F(a) \right\|_{\mathcal{U}} = \left\| F(\mathcal{C}_h^{\mathcal{A}} a) - \mathcal{C}_h^{\mathcal{U}} F(a) \right\|_{\mathcal{U}} \le \eta.$$

The norm inequality for the Bochner integral therefore yields

$$d_\infty(F, \Pi_H F) \le \eta.$$

The L_ℓ-Lipschitz property of the loss implies

$$|\mathcal{R}(F) - \mathcal{R}(\Pi_H F)| \le L_\ell \eta, \qquad |\widehat{\mathcal{R}}_N(F) - \widehat{\mathcal{R}}_N(\Pi_H F)| \le L_\ell \eta.$$

Applying the triangle inequality to $\mathrm{Gen}_N(F) - \mathrm{Gen}_N(\Pi_H F)$ gives

$$|\mathrm{Gen}_N(F)| \le |\mathrm{Gen}_N(\Pi_H F)| + 2L_\ell \eta,$$

which completes the proof. □

Lemma D.8 (Local Smoothness and Quadratic Growth Around the Minima Manifold)**.** *Assume that the zero set Ξ^* is nonempty, and let*

$$\mathcal{J}_{\mathrm{TTA}} : \mathfrak{g} \to \mathbb{R}_{\ge 0}$$

be continuously differentiable on an open neighborhood $\mathcal{N}(\Xi^*) \subset \mathfrak{g}$ of

$$\Xi^* := \{\xi \in \mathfrak{g} \mid \mathcal{J}_{\mathrm{TTA}}(\xi) = 0\}.$$

Assume further that there exists $L > 0$ such that, for all $\hat{\xi}_1, \hat{\xi}_2 \in \mathcal{N}(\Xi^*)$,

$$\|\nabla \mathcal{J}_{\mathrm{TTA}}(\hat{\xi}_1) - \nabla \mathcal{J}_{\mathrm{TTA}}(\hat{\xi}_2)\|_{\mathfrak{g}} \le L \|\hat{\xi}_1 - \hat{\xi}_2\|_{\mathfrak{g}},$$

and that there exists $\alpha > 0$ such that, for all $\hat{\xi} \in \mathcal{N}(\Xi^*)$,

$$\mathcal{J}_{\mathrm{TTA}}(\hat{\xi}) \geq \alpha \operatorname{dist}(\hat{\xi}, \Xi^*)^2,$$

where

$$\operatorname{dist}(\hat{\xi}, \Xi^*) := \inf_{\xi' \in \Xi^*} \|\hat{\xi} - \xi'\|_{\mathfrak{g}}.$$

Since Ξ^* is the zero set of $\mathcal{J}_{\mathrm{TTA}}$, we have

$$\min \mathcal{J}_{\mathrm{TTA}} = 0.$$

The quadratic-growth condition immediately yields the local error bound (Karimi et al., 2016)

$$\operatorname{dist}(\hat{\xi}, \Xi^*) \leq \alpha^{-1/2} \, \mathcal{J}_{\mathrm{TTA}}(\hat{\xi})^{1/2}.$$

Theorem D.9 (Monotone Energy Decay for Test-Time Adaptation Near the Minima Manifold)**.** *Assume the hypotheses of Lemma D.8. Consider a global solution of the continuous-time gradient flow*

$$\frac{\mathrm{d}\hat{\xi}}{\mathrm{d}t} = -\nabla \mathcal{J}_{\mathrm{TTA}}(\hat{\xi}(t)).$$

Let the initial point $\hat{\xi}^{(0)}$ satisfy

$$\widetilde{\mathcal{S}}_0 := \left\{ \xi \in \mathfrak{g} \;\middle|\; \mathcal{J}_{\mathrm{TTA}}(\xi) \leq \mathcal{J}_{\mathrm{TTA}}(\hat{\xi}^{(0)}) \right\}, \qquad \mathcal{S}_0 := \overline{\operatorname{Conn}_{\hat{\xi}^{(0)}}(\widetilde{\mathcal{S}}_0)} \Subset \mathcal{N}(\Xi^*).$$

Here $\operatorname{Conn}_{\hat{\xi}^{(0)}}$ denotes the connected component containing the initial point. This component-wise assumption is needed for periodic coordinates: the full sublevel set in the vector-space chart $\mathfrak{g} \cong \mathbb{R}^k$ can contain infinitely many translated copies and need not be compact. Then, for all $t \geq 0$, the energy is nonincreasing along the flow:

$$\frac{\mathrm{d}}{\mathrm{d}t} \mathcal{J}_{\mathrm{TTA}}(\hat{\xi}(t)) = -\|\nabla \mathcal{J}_{\mathrm{TTA}}(\hat{\xi}(t))\|_{\mathfrak{g}}^2 \leq 0.$$

Consequently,

$$\mathcal{J}_{\mathrm{TTA}}(\hat{\xi}(t)) + \int_0^t \|\nabla \mathcal{J}_{\mathrm{TTA}}(\hat{\xi}(s))\|_{\mathfrak{g}}^2 \, \mathrm{d}s = \mathcal{J}_{\mathrm{TTA}}(\hat{\xi}^{(0)}),$$

and the trajectory remains in the connected component $\mathcal{S}_0 \subset \mathcal{N}(\Xi^*)$. There exists

$$\mathcal{J}_\infty \in [0, \mathcal{J}_{\mathrm{TTA}}(\hat{\xi}^{(0)})]$$

such that

$$\lim_{t \to \infty} \mathcal{J}_{\mathrm{TTA}}(\hat{\xi}(t)) = \mathcal{J}_\infty.$$

In addition,

$$\int_0^\infty \|\nabla \mathcal{J}_{\mathrm{TTA}}(\hat{\xi}(t))\|_{\mathfrak{g}}^2 \, \mathrm{d}t = \mathcal{J}_{\mathrm{TTA}}(\hat{\xi}^{(0)}) - \mathcal{J}_\infty < \infty,$$

and

$$\lim_{t \to \infty} \|\nabla \mathcal{J}_{\mathrm{TTA}}(\hat{\xi}(t))\|_{\mathfrak{g}} = 0.$$

The distance to the minima manifold is controlled by the energy:

$$\operatorname{dist}(\hat{\xi}(t), \Xi^*) \leq \alpha^{-1/2} \, \mathcal{J}_{\mathrm{TTA}}(\hat{\xi}(t))^{1/2} \leq \alpha^{-1/2} \, \mathcal{J}_{\mathrm{TTA}}(\hat{\xi}^{(0)})^{1/2}.$$

If, in addition, $\mathcal{J}_\infty = 0$, then

$$\lim_{t \to \infty} \operatorname{dist}(\hat{\xi}(t), \Xi^*) = 0.$$

The trajectory $\hat{\xi}(t)$ has accumulation points in the compact component $\mathcal{S}_0$, and every trajectory accumulation point is a critical point of $\mathcal{J}_{\mathrm{TTA}}$.

Proof of Theorem D.9. Since

$$\frac{\mathrm{d}\hat{\xi}}{\mathrm{d}t} = -\nabla \mathcal{J}_{\mathrm{TTA}}(\hat{\xi}(t)),$$

the chain rule gives

$$\frac{\mathrm{d}}{\mathrm{d}t}\mathcal{J}_{\mathrm{TTA}}(\hat{\xi}(t)) = \left\langle \nabla \mathcal{J}_{\mathrm{TTA}}(\hat{\xi}(t)), \frac{\mathrm{d}\hat{\xi}}{\mathrm{d}t} \right\rangle_{\mathfrak{g}} = -\|\nabla \mathcal{J}_{\mathrm{TTA}}(\hat{\xi}(t))\|_{\mathfrak{g}}^2 \leq 0.$$

This proves monotonicity, and integrating over $[0, t]$ yields

$$\mathcal{J}_{\mathrm{TTA}}(\hat{\xi}(t)) + \int_0^t \|\nabla \mathcal{J}_{\mathrm{TTA}}(\hat{\xi}(s))\|_{\mathfrak{g}}^2 \,\mathrm{d}s = \mathcal{J}_{\mathrm{TTA}}(\hat{\xi}^{(0)}).$$

Hence the trajectory remains in the initial sublevel set. Since the trajectory is continuous and starts in the component $\mathcal{S}_0$, it cannot move to a disjoint component without leaving the sublevel set; therefore it remains in $\mathcal{S}_0$.

Because $\mathcal{J}_{\mathrm{TTA}}(\hat{\xi}(t))$ is nonincreasing and bounded below by 0, it converges to a limit $\mathcal{J}_\infty \geq 0$. Passing to the limit $t \to \infty$ in the identity above gives

$$\int_0^\infty \|\nabla \mathcal{J}_{\mathrm{TTA}}(\hat{\xi}(s))\|_{\mathfrak{g}}^2 \,\mathrm{d}s = \mathcal{J}_{\mathrm{TTA}}(\hat{\xi}^{(0)}) - \mathcal{J}_\infty < \infty.$$

Because $\mathcal{S}_0$ is compact and $\nabla \mathcal{J}_{\mathrm{TTA}}$ is continuous, there is an $M < \infty$ such that $\|\nabla \mathcal{J}_{\mathrm{TTA}}(\hat{\xi}(t))\|_{\mathfrak{g}} \leq M$ along the trajectory. Hence $\dot{\hat{\xi}} = -\nabla \mathcal{J}_{\mathrm{TTA}}(\hat{\xi})$ implies that $\hat{\xi}(t)$ is M-Lipschitz in time. Composing this trajectory with the L-Lipschitz gradient makes the map

$$t \mapsto \nabla \mathcal{J}_{\mathrm{TTA}}(\hat{\xi}(t))$$

LM-Lipschitz, and therefore uniformly continuous. Together with the square-integrability established above, Barbalat's lemma implies

$$\|\nabla \mathcal{J}_{\mathrm{TTA}}(\hat{\xi}(t))\|_{\mathfrak{g}} \to 0 \qquad \text{as } t \to \infty.$$

The quadratic-growth assumption in Lemma D.8 gives

$$\mathcal{J}_{\mathrm{TTA}}(\hat{\xi}(t)) \geq \alpha \,\mathrm{dist}(\hat{\xi}(t), \Xi^*)^2,$$

and therefore

$$\mathrm{dist}(\hat{\xi}(t), \Xi^*) \leq \alpha^{-1/2}\, \mathcal{J}_{\mathrm{TTA}}(\hat{\xi}(t))^{1/2}.$$

Since the energy is nonincreasing,

$$\mathrm{dist}(\hat{\xi}(t), \Xi^*) \leq \alpha^{-1/2}\, \mathcal{J}_{\mathrm{TTA}}(\hat{\xi}^{(0)})^{1/2}.$$

Finally, if $\mathcal{J}_{\mathrm{TTA}}(\hat{\xi}(t)) \to 0$, the same inequality implies

$$\mathrm{dist}(\hat{\xi}(t), \Xi^*) \to 0.$$

Compactness of $\mathcal{S}_0$ gives accumulation points of the trajectory. Since the gradient vanishes asymptotically and the gradient is continuous, every trajectory accumulation point is a critical point of $\mathcal{J}_{\mathrm{TTA}}$. □

Lemma D.10 (Lipschitz Continuity of Inverse Group Action)**.** *Let G be a finite-dimensional Lie group with Lie algebra $\mathfrak{g}$, fix $a_{\mathrm{OOD}} \in \mathcal{A}_{\mathrm{OOD}}$, and let $V \subset \mathfrak{g}$ be open. Assume that the parameter map $\xi \mapsto \mathcal{C}^{-1}_{\exp(\xi)}(a_{\mathrm{OOD}})$ is continuously differentiable on V. For any compact convex set $K \Subset V$, there exists a constant $L_{a,K} > 0$ such that for all $\hat{\xi}_1, \hat{\xi}_2 \in K$,*

$$\big\|\mathcal{C}^{-1}_{\exp(\hat{\xi}_1)}(a_{\mathrm{OOD}}) - \mathcal{C}^{-1}_{\exp(\hat{\xi}_2)}(a_{\mathrm{OOD}})\big\|_{\mathcal{A}} \leq L_{a,K}\|\hat{\xi}_1 - \hat{\xi}_2\|_{\mathfrak{g}}.$$

This is a local statement on K, not a global Lipschitz claim on an arbitrary exponential chart. For continuum translations, the differentiability assumption is satisfied, for example, on sufficiently regular (e.g., H^1) fields; for the finite-grid implementation it is a property of the bounded resampling map.

Corollary D.11 (Prediction Error Bound Under Symmetry-Induced Equivalent Minima)**.** *Assume the hypotheses of Theorems D.9 and D.7. Let $a_{\mathrm{OOD}} \in \mathcal{A}_{\mathrm{OOD}}$ be an arbitrary out-of-distribution input, and suppose the adaptation trajectory and the zero set Ξ^* lie in a common compact convex coordinate set K_ξ on which the input action is Lipschitz. Assume that the kinematic map $\Gamma_B(\cdot; T)$ is Lipschitz on K_ξ, so that*

$$\|\Gamma_B(\xi; T) - \Gamma_B(\xi'; T)\|_{\mathfrak{g}} \le L_\Gamma \|\xi - \xi'\|_{\mathfrak{g}}.$$

Assume, in addition, that every $\xi' \in \Xi^*$ is calibrated*: its pullback and terminal action are symmetry-equivalent to an admissible witness in Theorem D.7. Let $\hat{\xi}$ denote any finite-time estimator on this trajectory. Then the geometric perturbation argument in Theorem D.7 yields*

$$\|\hat{u}_{\mathrm{pred}} - \mathcal{G}^\dagger(a_{\mathrm{OOD}})\|_{\mathcal{U}} \le K_C \epsilon + C_{\mathrm{geom}} \operatorname{dist}(\hat{\xi}, \Xi^*),$$

for some $C_{\mathrm{geom}} > 0$. Then

$$\|\hat{u}_{\mathrm{pred}} - \mathcal{G}^\dagger(a_{\mathrm{OOD}})\|_{\mathcal{U}} \le K_C \epsilon + C_{\mathrm{geom}}\, \alpha^{-1/2}\, \mathcal{J}_{\mathrm{TTA}}(\hat{\xi})^{1/2}.$$

In particular, along any adaptation trajectory $\hat{\xi}(t)$ such that

$$\mathcal{J}_{\mathrm{TTA}}(\hat{\xi}(t)) \to 0,$$

we have

$$\limsup_{t \to \infty} \|\hat{u}_{\mathrm{pred}} - \mathcal{G}^\dagger(a_{\mathrm{OOD}})\|_{\mathcal{U}} \le K_C \epsilon.$$

Proof of Corollary D.11. Fix an arbitrary estimator $\hat{\xi}$ produced by the test-time adaptation dynamics. Since $\mathfrak{g}$ is finite-dimensional and $\mathcal{J}_{\mathrm{TTA}}$ is continuous, the nonempty zero set

$$\Xi^* = \{\xi \in \mathfrak{g} \mid \mathcal{J}_{\mathrm{TTA}}(\xi) = 0\}$$

is closed. Hence the distance from $\hat{\xi}$ to Ξ^* is attained by some $\xi' \in \Xi^*$, so that

$$\|\hat{\xi} - \xi'\|_{\mathfrak{g}} = \operatorname{dist}(\hat{\xi}, \Xi^*).$$

By Lemma D.10 on the compact coordinate set K_ξ, the inverse group action on the input space is Lipschitz with respect to the Lie algebra coordinates. Hence,

$$\left\| \mathcal{C}^{-1}_{\exp(\hat{\xi})}(a_{\mathrm{OOD}}) - \mathcal{C}^{-1}_{\exp(\xi')}(a_{\mathrm{OOD}}) \right\|_{\mathcal{A}} \le L_a \|\hat{\xi} - \xi'\|_{\mathfrak{g}}.$$

The calibration assumption makes ξ' an admissible symmetry-equivalent witness. The Lipschitz assumption on Γ_B together with the local Lipschitz property of the terminal output action gives, for some $L_u > 0$,

$$\epsilon_{\mathrm{geo},T} \le L_u \|\hat{\xi} - \xi'\|_{\mathfrak{g}}.$$

Applying Theorem D.7 with the symmetrically equivalent minimizer $\xi' \in \Xi^*$ yields

$$\begin{aligned} \|\hat{u}_{\mathrm{pred}} - \mathcal{G}^\dagger(a_{\mathrm{OOD}})\|_{\mathcal{U}} &\le K_C \epsilon + M_0 \left\| \mathcal{C}^{-1}_{\exp(\hat{\xi})}(a_{\mathrm{OOD}}) - \mathcal{C}^{-1}_{\exp(\xi')}(a_{\mathrm{OOD}}) \right\|_{\mathcal{A}} + \epsilon_{\mathrm{geo},T} \\ &\le K_C \epsilon + (M_0 L_a + L_u) \|\hat{\xi} - \xi'\|_{\mathfrak{g}}. \end{aligned}$$

Defining

$$C_{\mathrm{geom}} := M_0 L_a + L_u,$$

we obtain

$$\|\hat{u}_{\mathrm{pred}} - \mathcal{G}^\dagger(a_{\mathrm{OOD}})\|_{\mathcal{U}} \le K_C \epsilon + C_{\mathrm{geom}} \operatorname{dist}(\hat{\xi}, \Xi^*).$$

Finally, applying the quadratic-growth error bound from Lemma D.8, we obtain

$$\operatorname{dist}(\hat{\xi}, \Xi^*) \le \alpha^{-1/2}\, \mathcal{J}_{\mathrm{TTA}}(\hat{\xi})^{1/2},$$

and therefore

$$\|\hat{u}_{\mathrm{pred}} - \mathcal{G}^\dagger(a_{\mathrm{OOD}})\|_{\mathcal{U}} \le K_C \epsilon + C_{\mathrm{geom}}\, \alpha^{-1/2}\, \mathcal{J}_{\mathrm{TTA}}(\hat{\xi})^{1/2}.$$

If $\mathcal{J}_{\mathrm{TTA}}(\hat{\xi}(t)) \to 0$, then the right-hand side has limit $K_C \epsilon$, which yields the stated $\limsup$ bound. □

This is a local stability result for TTA, not a global convergence result for arbitrary OOD inputs. If initialization lies near the symmetry-equivalent minima manifold, reducing the residual energy controls the remaining geometric error. On the experimental tori, shifts by a full spatial period produce the same canonical field, so the bound does not depend on which equivalent minimizer attains the distance.