RAL-MoE-ROM: HIERARCHICAL MIXTURE-OF-EXPERTS FOR REGIME-ADAPTIVE REDUCED-ORDER MODEL

Anonymous authors
Paper under double-blind review

ABSTRACT

Reduced-order models (ROMs) approximate high-dimensional dynamical systems using a small number of state variables. However, their predictive accuracy can deteriorate when parameter changes lead to different dynamical regimes. Different regimes may require different reduced representations and dynamics, making it difficult for a single model to remain accurate across the full parameter range. We introduce RAL-MoE-ROM, a hierarchical mixture-of-experts framework with two levels: regime-local specialists and cross-regime routing with fusion. At the first level, each local model, called a specialist, uses a structured sparse mixture of experts to improve its reduced dynamics and is trained through multi-step autonomous rollouts. At the second level, a parameter-only outer router provides preferences within prescribed adjacent specialist pairs, supporting Top-1 selection or history-conditioned specialist fusion. When two specialists are selected, a learned gate uses the initial physical history to combine their reconstructed fields. Across three flow benchmarks spanning steady, Hopf-onset, and periodic regimes, the local specialists provide accurate autonomous predictions, while cross-regime routing further improves prediction in the evaluated regime overlaps.

1 INTRODUCTION

High-fidelity simulations of high-dimensional dynamical systems can be expensive when repeated predictions are required across many parameter settings. Reduced-order models (ROMs) provide a cost-effective surrogate by representing the physical state with a small number of state variables. The problem becomes more challenging when changing a parameter also changes the type of regimes. In incompressible flows, for example, varying the Reynolds number can lead from relaxation toward a steady state to oscillatory growth and sustained periodic motion.

Such regime variation introduces two closely related challenges for ROMs. The first concerns representation: spatial structures that are important in one regime may differ from those required in another, making a single representation less effective. The second is dynamical: the reduced evolution law must accommodate qualitatively different behaviors, including stable relaxation, instability growth, and nonlinear periodic oscillations. Increasing model capacity alone cannot compensate for an inadequate reduced representation, while improving the representation does not by itself guarantee accurate autonomous rollout. These observations motivate adapting the local representation and the local dynamics separately, while providing a consistent way to assemble their predictions.

Projection-based ROMs provide an explicit reduced representation and reduced dynamics, commonly through proper orthogonal decomposition (POD) and Galerkin projection (Sirovich, 1987; Berkooz et al., 1993; Benner et al., 2015). However, projection-based ROMs are inherently affected by truncation, which neglects the influence of discarded modes on the retained coordinates, leading to closure errors that degrade long-horizon prediction (Amsallem & Farhat, 2012). Data-driven ROMs offer greater flexibility in approximating nonlinear dynamics, but low one-step error does not guarantee accurate autonomous rollouts, and they often lack physical interpretability (Duraisamy et al., 2019; Brunton et al., 2020). Hybrid ROMs combine projection-based ROMs with learned corrections or learned reduced representations, retaining physical structure while increasing approx-

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.

imation flexibility (Dar et al., 2023; Lee & Carlberg, 2020; Fresca & Manzoni, 2022; Geelen et al., 2023). These developments improve the modeling of reduced dynamics, but they do not by themselves resolve the representational difficulty that arises when qualitatively different regimes must be described over a broad parameter range.

A complementary strategy localizes the reduced representation, using multiple subspaces, clusters, or manifolds for different portions of the solution set (Kaiser et al., 2014; Li et al., 2021a; Diaz et al., 2024). These directions address complementary aspects of the broad-parameter problem: hybrid modeling improves the accuracy of the reduced dynamics, whereas localization adapts the reduced representation to regime-dependent structures. Localization, however, creates a new assembly problem. Independently constructed local ROMs generally have different reduced state variables and operators, so their reduced coordinates are chart-specific. Direct interpolation or averaging in reduced-coordinate space therefore has no natural cross-chart interpretation. Mixture-of-experts (MoE) architectures provide a natural framework for conditional specialization (Jacobs et al., 1991; Jordan & Jacobs, 1994; Shazeer et al., 2017; Fedus et al., 2022), but standard MoE formulations typically route specialized predictors within a shared representation or common feature space. Here, in contrast, the experts are complete local ROMs whose coordinates are not shared. The problem is therefore hierarchical: beyond routing among local ROMs, one must assemble or fuse their outputs across representations, while adapting the reduced dynamics within each local representation. This raises a distinct question that standard within-representation MoE routing does not address: how can independently constructed local dynamical models be routed and assembled without forcing them into a common latent coordinate system?

To address this gap, we introduce the Regime-Adaptive Localized Mixture-of-Experts Reduced-Order Model (RAL-MoE-ROM), a two-level hierarchical framework that separates regime-local reduced dynamics from cross-regime routing and fusion. The parameter domain is covered by prescribed, overlapping regime-local regions, each equipped with an independent autonomous specialist. At the first level, each specialist uses an explicit projection-based ROM and augments it with sparsely structured MOEs. Rather than using conventional MLP-only experts, each activated expert decomposes its output into three additive components: a nonlinear neural branch, together with explicit linear and low-rank quadratic branches derived from the projection-based ROM. At the second level, a parameter-only outer router provides preferences over regime-local specialists. Within a prescribed adjacent specialists pair, these preferences support either Top-1 specialist or neighboring specialists . When two specialists are selected, a learned fusion gate combines their reconstructed physical fields conditioned on the initial physical history. The overall architecture is illustrated in Figure 1.

Our main contributions are as follows:

- **Structured regime-local specialists for autonomous reduced dynamics.** Each specialist combines a projection-based ROM with a structured sparse MoE closure and is trained under closed-loop multi-step rollouts.
- **Cross-regime routing and fusion.** An outer router selects a specialist or a pair of neighboring specialists. When two specialists are selected, a learned gate uses the initial physical history to combine their reconstructed fields.
- **Multi-regime evaluation across flow geometries.** We evaluate the framework on three flow benchmarks spanning steady, Hopf-onset, and periodic regimes. Complementary comparisons and ablations examine the contribution of learned local corrections and history-conditioned cross-regime fusion, while routing and cost analyses characterize expert utilization and practical computational trade-offs. Together, these studies identify the benefits of the framework across the evaluated settings.

2 Related Work

Projection, learned dynamics, and autonomous rollout. POD–Galerkin methods project the governing equations onto a reduced basis and retain an explicit reduced dynamical system (Sirovich, 1987; Berkooz et al., 1993; Benner et al., 2015). Their long-horizon accuracy can deteriorate when truncated interactions are not adequately represented, motivating closure and stabilization strategies (Amsallem & Farhat, 2012). Non-intrusive and data-driven ROMs instead infer reduced evolution

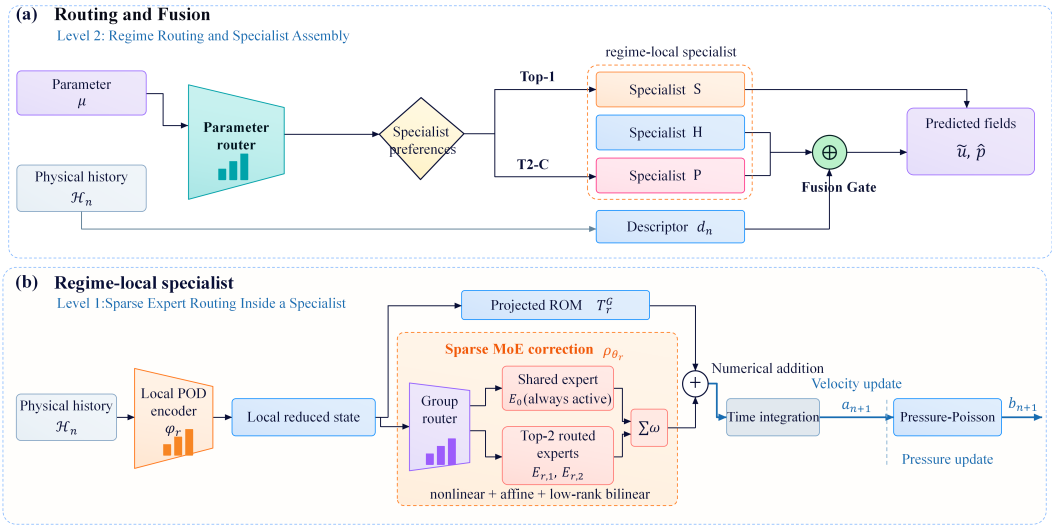


Figure 1: Framework of RAL-MoE-ROM, a two-level hierarchical framework. (a) Level 2: a parameter-only outer router provides trajectory-level preferences over regime-local specialists. For a neighbouring specialists selection, a history-conditioned gate fuses their reconstructed physical fields. (b) Level 1: each specialist augments a projected Galerkin ROM with a structured sparse MoE correction, using internal routing to select correction experts within its local reduced chart.

laws directly from simulation data, ranging from projection-based operator inference to neural and latent-state dynamics (Peherstorfer & Willcox, 2016; Hesthaven & Ubbiali, 2018; Champion et al., 2019; Lee & Carlberg, 2020; Pan & Xiao, 2024; Park et al., 2024). Operator-inference methods provide a particularly direct connection between data-driven reduced dynamics and classical projection-based ROMs (Peherstorfer & Willcox, 2016). Neural-operator approaches learn mappings between function spaces rather than explicit low-dimensional state evolution (Lu et al., 2021; Kovachki et al., 2023). Hybrid ROMs retain projected or physics-based structure while learning missing or unresolved reduced dynamics (Duraisamy et al., 2019; Fresca & Manzoni, 2022; Dar et al., 2023). RAL-MoE-ROM follows this hybrid principle, but places a structured sparse-MoE correction inside each regime-local ROM and trains the resulting specialist under multi-step autonomous rollout.

Localized representations and local reduced dynamics. Localization has been explored through parameter-dependent reduced bases, localized operators and nonlinear manifolds (Amsallem & Farhat, 2008; Peherstorfer et al., 2014; Kaiser et al., 2014; Li et al., 2021a; Diaz et al., 2024; Bhat et al., 2025). These approaches all reduce the burden on a single global representation, differing in whether localization is performed along parameter, state, or spatial coordinates. Recent work has further emphasized explicitly local reduced dynamics. More recently, ql-ROM partitions the solution manifold into clusters, constructs an intrusive ROM within each cluster, and connects the local models through assignment and change-of-basis operations (Colanera & Magri, 2025). Therefore, when complete local ROMs are constructed independently, their reduced coordinates and operators need not be directly compatible, so switching or combining them requires an explicit coordination mechanism. RAL-MoE-ROM takes a different approach: instead of coordinating local models in reduced-coordinate space, it lets local specialists evolve independently in their native coordinates and combines neighboring predictions only after reconstruction in the common physical space.

Mixture-of-experts and cross-chart assembly. Classical hierarchical mixtures learn conditional expert assignments (Jacobs et al., 1991; Jordan & Jacobs, 1994), while sparse MoE architectures activate only a subset of experts for each input (Shazeer et al., 2017; Fedus et al., 2022). MoE-based conditional specialization has also been introduced into scientific machine learning, including physics-informed constraints, neural-operator pretraining, and PDE forecasting (Chalapathi et al., 2024; Ben-Shabat et al., 2024; Wang et al., 2025; Li et al., 2026). These methods demonstrate the value of conditional specialization, but do not address routing among complete autonomous ROMs.

RAL-MoE-ROM therefore applies routing at two levels: sparse experts model unresolved dynamics within each local ROM, while an outer router acts on the complete regime-local specialists. The outer level selects a single specialist or an applicable neighboring pair, with the latter combined through a history-conditioned gate after physical reconstruction.

3 RAL-MoE-ROM

RAL-MoE-ROM decomposes reduced modeling into three complementary components: regime-local ROMs, autonomous local specialists, and cross-regime routing and fusion. These components are organized by a two-level routing hierarchy: internal sparse expert routing within each local specialist, and outer parameter-only regime routing across specialists. We describe these components in detail below.

3.1 Regime-local ROMs

We consider parameterized incompressible flows governed by the Navier–Stokes equations,

$$\partial_t \boldsymbol{u} + (\boldsymbol{u}\cdot\nabla)\boldsymbol{u} = -\nabla p + \nu(\mu)\Delta\boldsymbol{u} + \boldsymbol{f}(\mu), \qquad \nabla\cdot\boldsymbol{u} = 0, \tag{1}$$

with the initial and boundary conditions of each benchmark. Here $\boldsymbol{u}$ and p denote velocity and pressure, μ is the physical parameter, ν is the viscosity, and $\boldsymbol{f}$ denotes a prescribed forcing term. In our benchmarks, parameters $\mu = Re$ and $\nu(\mu) = Re^{-1}$. Finite-volume spatial discretization gives the semi-discrete momentum equation and incompressibility constraint,

$$\dot{\boldsymbol{u}} = \boldsymbol{f}_h(\mu) + \mathcal{L}_h(\mu)\boldsymbol{u} - \mathcal{N}_h(\boldsymbol{u},\boldsymbol{u};\mu) - \nabla_h(\mu)p, \qquad \mathrm{div}_h(\mu)\boldsymbol{u} = 0. \tag{2}$$

while $\boldsymbol{u} \in \mathbb{R}^{N_u}$ and $p \in \mathbb{R}^{N_p}$ denotes the discrete field vectors, using the same symbols as their continuous counterparts. The subscript h denote spatial discretization: $\mathcal{L}_h$ and $\mathcal{N}_h$ represent the viscous and convective terms, ∇_h and div_h the gradient and divergence, and $\boldsymbol{f}_h$ the discrete forcing.

As μ varies, the flow can exhibit different dynamical regimes. Let $r \in \{S, H, P\}$ index steady, Hopf-onset, and periodic regimes, with corresponding overlapping parameter regions $\mathcal{M}_r$. For each region, we construct separate velocity and pressure basis by applying area-weighted proper orthogonal decomposition (POD) to high-fidelity snapshots. The velocity basis $\Phi_u^{(r)}$ satisfies ${\Phi_u^{(r)}}^\top M_u \Phi_u^{(r)} = I$, where M_u is the quadrature matrix. It defines the local encoding and reconstruction

$$\boldsymbol{a}^{(r)} = {\Phi_u^{(r)}}^\top M_u(\boldsymbol{u} - \bar{\boldsymbol{u}}^{(r)}), \qquad \widehat{\boldsymbol{u}}^{(r)} = \bar{\boldsymbol{u}}^{(r)} + \Phi_u^{(r)}\boldsymbol{a}^{(r)}, \tag{3}$$

where $\bar{\boldsymbol{u}}^{(r)}$ is the local training mean and $\boldsymbol{a}^{(r)} \in \mathbb{R}^{d_u^{(r)}}$ denotes the local reduced velocity coordinates for regime r, with $d_u^{(r)}$ the number of retained velocity POD modes. Pressure is first mapped to a common weighted zero-mean gauge and then encoded analogously using $\Phi_p^{(r)}$, yielding coefficients $\boldsymbol{b}^{(r)}$. Appendix A provides details on the problem setting and weighted POD construction.

Substituting the local velocity and pressure reconstructions into Eq. (2) and projecting its momentum equation onto $\Phi_u^{(r)}$ in the M_u inner product gives the Galerkin equation:

$$\dot{\boldsymbol{a}}^{(r)} = \boldsymbol{c}_r(\mu) + \boldsymbol{A}_r(\mu)\boldsymbol{a}^{(r)} + \boldsymbol{H}_r(\mu) : \left(\boldsymbol{a}^{(r)} \otimes \boldsymbol{a}^{(r)}\right) + \boldsymbol{C}_r^p(\mu)\boldsymbol{b}^{(r)} = \mathcal{F}_r^G(\boldsymbol{a}^{(r)}, \boldsymbol{b}^{(r)}; \mu), \tag{4}$$

Appendix B provides the definitions of these projected operators and introduces the projected pressure–Poisson equation(PPE).

3.2 Regime-local hybrid specialists

A hybrid specialist augments the chart-local projected Galerkin equation with a learned correction,

$$\dot{\boldsymbol{a}}^{(r)} = \mathcal{F}_r^G(\boldsymbol{a}^{(r)}, \boldsymbol{b}^{(r)}; \mu) + \boldsymbol{\rho}_{\theta_r}^u. \tag{5}$$

Local features and two computation paths. Let $\boldsymbol{g}_n^{(r)} = \mathcal{F}_r^G(\boldsymbol{a}_n^{(r)}, \boldsymbol{b}_n^{(r)}; \mu)$. The chart-specific feature map constructs a standardized vector

$$\boldsymbol{\xi}_n^{(r)} = \mathrm{Feat}_r\Big((\boldsymbol{a}_{n-\ell}^{(r)}, \boldsymbol{b}_{n-\ell}^{(r)}, \boldsymbol{g}_{n-\ell}^{(r)})_{\ell=0}^2, \mu\Big), \qquad \boldsymbol{z}_n^{(r)} = \mathrm{Enc}_r(\boldsymbol{\xi}_n^{(r)}). \tag{6}$$

Features include current states, projected Galerkin equation, and parameter features. The encoded features feed two paths: expert responses use $[\boldsymbol{z}; \boldsymbol{s}^c]$, whereas routing uses $\boldsymbol{\chi} = [\boldsymbol{z}; \boldsymbol{\xi}]$. $c \in \{u, p\}$ denotes the velocity or pressure output channel, and

$$\boldsymbol{s}^u = \widetilde{\boldsymbol{a}}^{(r)}, \qquad \boldsymbol{s}^p = [\widetilde{\boldsymbol{a}}^{(r)}; \widetilde{\boldsymbol{b}}^{(r)}] \tag{7}$$

are the current-state slices of the standardized feature vector.

Expert-response path. For expert e in group m, a nonlinear residual feed-forward network $N^c_{e,m}$ first projects $[\boldsymbol{z}; \boldsymbol{s}^c]$ to its hidden representation, applies expanded residual blocks with GELU activations, and maps the result to d_c outputs, with $d_c = d_u^{(r)}$ or $d_p^{(r)}$. This response is augmented by explicit linear and low-rank quadratic functions of the current state:

$$[E^c_{e,m}(\boldsymbol{z}, \boldsymbol{s}^c)]_j = [N^c_{e,m}([\boldsymbol{z}; \boldsymbol{s}^c])]_j + [L^c_{e,m}\boldsymbol{s}^c]_j + \kappa \sum_{\ell=1}^{q_b} \big(\boldsymbol{v}^{c,(1)}_{e,m,j\ell}{}^\top \boldsymbol{s}^c\big)\big(\boldsymbol{v}^{c,(2)}_{e,m,j\ell}{}^\top \boldsymbol{s}^c\big). \tag{8}$$

The linear map has no bias, and the two quadratic factors are independent for each output component j. The verified configurations use $q_b = 4$ and $\kappa = 0.05$. Shared and routed experts have the same architecture but separate parameters. Thus the nonlinear branch uses history-conditioned features, while the explicit polynomial branches act directly on local state coordinates.

Routing-and-weighting path. The group router receives $\boldsymbol{\chi} = [\boldsymbol{z}; \boldsymbol{\xi}]$ and computes

$$\boldsymbol{\ell}^{\mathrm{grp}} = R^{\mathrm{grp}}_r(\boldsymbol{\chi}), \qquad \boldsymbol{p}^{\mathrm{grp}} = \mathrm{softmax}(\boldsymbol{\ell}^{\mathrm{grp}}/T_g), \qquad m^\star = \arg\max_m p^{\mathrm{grp}}_m, \tag{9}$$

where $T_g > 0$ is the group temperature. The router is a layer-normalized MLP with a SiLU hidden activation and dropout. Selection is hard Top-1 for each sample. During training, a straight-through gate $\widetilde{\boldsymbol{q}} = \mathrm{onehot}(m^\star) - \mathrm{stopgrad}(\boldsymbol{p}^{\mathrm{grp}}) + \boldsymbol{p}^{\mathrm{grp}}$ retains hard selection in the forward pass while providing a softmax gradient path. Within the selected group, independent velocity and pressure routers with the same MLP form score six routed experts:

$$\boldsymbol{\ell}^c = R^c_{r,m^\star}(\boldsymbol{\chi}), \qquad \boldsymbol{p}^c = \mathrm{softmax}(\boldsymbol{\ell}^c/T_c), \qquad \mathcal{T}^c_2 = \mathrm{Top2}(\boldsymbol{p}^c). \tag{10}$$

Here Top2 returns the indices of the two largest scores and $T_c > 0$ is the channel temperature. The shared expert $e = 0$ is excluded from this competition and remains active in the selected group. Given configured shared and routed scales $\lambda_0, \lambda_{\mathrm{rt}} > 0$, the weights are

$$\omega^c_0 = \frac{\lambda_0}{\lambda_0 + \lambda_{\mathrm{rt}}}, \qquad \omega^c_e = \frac{\lambda_{\mathrm{rt}}}{\lambda_0 + \lambda_{\mathrm{rt}}} \frac{p^c_e}{\sum_{j \in \mathcal{T}^c_2} p^c_j}, \quad e \in \mathcal{T}^c_2. \tag{11}$$

The two paths meet in the sparse mixture

$$\mathcal{M}^c_{\theta_r} = \omega^c_0 E^c_{0,m^\star}(\boldsymbol{z}, \boldsymbol{s}^c) + \sum_{e \in \mathcal{T}^c_2} \omega^c_e E^c_{e,m^\star}(\boldsymbol{z}, \boldsymbol{s}^c). \tag{12}$$

Expert outputs are combined in standardized target coordinates and then converted using the chart-specific output scalers to the velocity and pressure corrections $\boldsymbol{\rho}^u_{\theta_r}$ and $\boldsymbol{\rho}^p_{\theta_r}$ in the additive instances.

Algebraic pressure reconstruction. In adaptive-gate instances, pressure is reconstructed after the velocity update as

$$\boldsymbol{b}^{(r)}_{n+1} = \boldsymbol{\gamma}^{(r)}_n \odot \mathcal{Q}^{PP}_r(\boldsymbol{a}^{(r)}_{n+1}; \mu) + \boldsymbol{\rho}^{p,(r)}_n, \qquad \boldsymbol{\gamma}^{(r)}_n = \mathrm{sigmoid}(\boldsymbol{\eta}^p_r(\boldsymbol{z}^{(r)}_n)). \tag{13}$$

The independent confidence head $\boldsymbol{\eta}^p_r$ maps the shared embedding through a 64-unit GELU hidden layer to $d_p^{(r)}$ outputs. $\mathcal{Q}^{PP}_r(\boldsymbol{a}^{(r)}_{n+1}; \mu)$ is algebraic pressure equation defined in Appendix B; Both terms are combined in the same unstandardized local pressure coordinates.

3.3 Closed-loop rollout training

Each specialist is initialized by encoding a physical history into its own local coordinates. It then advances autonomously, using its predicted velocity and reconstructed pressure to update the history.

Time integration with fixed context. During each time-integration step, the pressure coefficient $\boldsymbol{b}_n^{(r)}$, past states, and the physical parameter are held fixed and collectively define the context $\mathcal{C}_n^{(r)}$. For an intermediate velocity state $\boldsymbol{a}$, the hybrid right-hand side and the numerical update are

$$F_r(\boldsymbol{a}; \mathcal{C}_n^{(r)}) = \mathcal{F}_r^G(\boldsymbol{a}, \boldsymbol{b}_n^{(r)}; \mu) + \boldsymbol{\rho}_{\theta_r}^u\left(\boldsymbol{\xi}^{(r)}(\boldsymbol{a}; \mathcal{C}_n^{(r)})\right), \tag{14}$$

$$\boldsymbol{a}_{n+1}^{(r)} = \mathrm{Step}_r\left(\boldsymbol{a}_n^{(r)}; F_r(\cdot; \mathcal{C}_n^{(r)}), \Delta t_r\right). \tag{15}$$

Here Δt_r is the time step for specialist r, and Step_r denotes its time-stepping scheme. For continuous-time formulations, each Runge–Kutta stage uses the same fixed context while recomputing the Galerkin term, group selection, channel-wise Top-2 routing weights, and expert responses at the intermediate velocity state. Pressure coefficients are not updated within these stages. After the velocity update, pressure coefficients are reconstructed using Eq. (13) and the local history is updated for the next step. Benchmark-specific time-stepping schemes are provided in Appendix D.

Rollout prediction loss. We unroll the complete specialist update for K steps and minimize the following normalized coefficient prediction loss:

$$\mathcal{L}_r = \frac{1}{K}\sum_{k=1}^{K}\left[\frac{\left\|(\widehat{\boldsymbol{a}}_{n+k}^{(r)} - \boldsymbol{a}_{n+k}^{(r)}) \oslash \boldsymbol{\sigma}_a^{(r)}\right\|_2^2}{d_u^{(r)}} + \frac{\left\|(\widehat{\boldsymbol{b}}_{n+k}^{(r)} - \boldsymbol{b}_{n+k}^{(r)}) \oslash \boldsymbol{\sigma}_b^{(r)}\right\|_2^2}{d_p^{(r)}}\right]. \tag{16}$$

$\widehat{\boldsymbol{a}}_{n+k}^{(r)}$ denote recursively predicted coefficients, whereas $\boldsymbol{a}_{n+k}^{(r)}$ symbols denote reference coefficients. The dimensions $d_u^{(r)}$ and $d_p^{(r)}$ depend on the local chart. The scaling vectors $\boldsymbol{\sigma}_a^{(r)}$ and $\boldsymbol{\sigma}_b^{(r)}$ are estimated from the corresponding training set, and $\oslash$ denotes elementwise division. We progressively increase the rollout horizon during training to account for accumulated prediction errors. Further details on the settings are provided in Appendix D.

3.4 Cross-regime routing and fusion

Unlike the internal routing in Section 3.2, the outer parameter-only regime router, denoted E2 in the experiments, produces a preference over the regime-local specialists:

$$\widetilde{\mu}_R = (\mu - \mu_R)/\sigma_R, \qquad \boldsymbol{\pi}(\mu) = \mathrm{softmax}\big(\mathcal{R}_{E2}(\widetilde{\mu}_R)\big). \tag{17}$$

R_{E2} is a simple three-layer MLP—a 1–24–3 MLP with 123 parameters. It is trained on chart-assigned parameter labels, so $\boldsymbol{\pi}$ is a parameter-conditioned preference. The deterministic parameter-only output is identical for windows with the same μ and remains fixed during each rollout. The E2 normalization and training protocol are specified in Appendix D.

The evaluation protocol supplies an ordered candidate pair $(r, s) = (S, H)$ or (P, H) on prespecified common chart support. For the supplied pair, define

$$\bar{\pi}_r = \frac{\pi_r}{\pi_r + \pi_s}, \qquad \bar{\pi}_s = 1 - \bar{\pi}_r. \tag{18}$$

E2 Top-1 selects r when $\bar{\pi}_r \geq 1/2$ and s otherwise. The pair order means that the first-specialist weight always refers to Steady in the S–H study and Periodic in the H–P study.

Both specialists encode the same physical initialization history in their own charts and advance independently. We refer to their history-conditioned physical-space fusion as T2-C (history-conditioned Top-2 fusion). Here, Top-2 denotes the two specialists in the prescribed adjacent pair, rather than the internally routed correction experts. T2-C adjusts the parameter-only pair preference using a chart-independent descriptor of the initial physical history:

$$\widehat{\boldsymbol{q}}_{\mathrm{T2C}}(t) = \alpha\,\widehat{\boldsymbol{q}}^{(r)}(t) + (1-\alpha)\widehat{\boldsymbol{q}}^{(s)}(t), \qquad \alpha = \mathrm{sigmoid}\left(\mathrm{logit}(\bar{\pi}_r^{\mathrm{clip}}) + c_\psi([\widetilde{\mu}_G; \boldsymbol{d}_n])\right). \tag{19}$$

Here $\bar{\pi}_r^{\mathrm{clip}}$ is $\bar{\pi}_r$ clipped to $[10^{-6}, 1-10^{-6}]$, and $\boldsymbol{q}$ collects reconstructed velocity and gauge-corrected pressure on the common physical mesh. The eight descriptor channels $\boldsymbol{d}_n$ are computed from the initial three physical states, without future reference states. The gate input $[\widetilde{\mu}_G; \boldsymbol{d}_n]$ uses training-window normalization, distinct from the trajectory-label normalization $\widetilde{\mu}_R$ used by E2. The correction network is a 9–64–64–1 MLP with SiLU hidden activations and a zero-initialized final layer.

4 EXPERIMENTS

We organize the experiments into four parts: the accuracy and reliability of local specialists prediction, the benefit of cross-regime routing and fusion, the behavior of internal routing and constraints and invariants , and the computational cost. Three test-case configurations: fluidic pinball, flow past a cylinder and flow past a square, are summarized in Appendix D.

4.1 LOCAL SPECIALISTS PREDICTION ACCURACY

Autonomous Rollout Error Comparison We first compare local specialists with classical POD–Galerkin over $K = 32$–56 steps on the fluidic-pinball benchmark. Table 1 shows local specialists achieve low physical-field errors across all four evaluation settings. Compared with POD–Galerkin, the learned specialists consistently reduce both velocity and pressure errors, with the largest differences appearing in pressure. For example, the Hopf joint error decreases from 48.94% to 2.0850%, while the standard Periodic error decreases from 43.20% to 0.8415%. The longer-horizon Periodic-core evaluation shows the same trend, with E_{joint} reduced from 41.91% to 1.672%. Together, these results show that the learned specialists maintain low physical-field error over the evaluated autonomous horizons. In particular, compared with POD-Galerkin, by incorporating a sparse mixture-of-experts correction into the projected Galerkin equation, the local specialists substantially reduce the long-horizon error.

Table 1: Autonomous local-specialist prediction on the fluidic-pinball benchmark.

Evaluation setting	Model	E_u	E_p	E_{joint}
Steady ($K = 56$)	POD–Galerkin	–	–	–
	Local-Specialist	0.2247	0.7593	0.9840
Hopf ($K = 56$)	POD–Galerkin	0.4378	48.50	48.94
	Local-Specialist	0.0995	1.9860	2.0850
Periodic ($K = 32$)	POD–Galerkin	1.6410	41.56	43.20
	Local-Specialist	0.2897	0.5518	0.8415
Periodic ($K = 56$)	POD–Galerkin	2.5550	39.36	41.91
	Local-Specialist	0.6222	1.050	1.6720

Next, we compare the accuracy of the local specialists for flow past a circular cylinder in the Hopf and periodic regimes over shared held-out windows at $K = 24$ and $K = 48$, as an architecture ablation study. Dense model matches the proposed model's active neural parameter budget, while Structured model retains explicit response branches but removes input-dependent expert routing, global specialist predicts Hopf and Periodic dynamics in a single global POD space. We also include a validation-selected operator inference (OpInf) fit as an external ROM comparison.

Table 2 presents an ablation comparison under joint error metric. In the Periodic regime, the local specialist achieves the lowest errors at both horizons, substantially outperforming Dense and Structured, both of which remove input-dependent expert routing. The external OpInf baseline, by contrast, yields errors as high as about 114% and nearly fails. This indicates that in periodic flows, the input-dependent sparse routing mechanism plays a decisive role in reducing long-horizon prediction errors. In the Hopf regime, the local specialist outperforming Dense and Structured model, and remaining at a comparable accuracy level to OpInf. To assess the value of regime-local representations relative to a single global ROM, we additionally evaluate Global specialist. The proposed local specialist consistently outperforms the global specialist in both regimes, pointing to the need for regime-local representations to achieve the reported accuracy; a single global representation remains less accurate than the corresponding local specialist in both regimes. Nevertheless, the proposed method maintains the lowest or near-lowest errors at both horizons, confirming the effectiveness and robustness of its local prediction. Overall, this ablation study supports the effectiveness of the local specialist accuracy : removing the input-dependent sparse routing leads to a marked degradation in accuracy, demonstrating that this component is a key source of performance improvement. Evaluation windows, component errors, implementation details, and more comprehensive comparisons against POD–LSTM, POD–GRU, QL-ROM, and FNO are provided in Appendix E.1, Table 6.

Table 2: Ablation study for flow past a circular cylinder.

Regime	Model	$K = 24$	$K = 48$
Hopf	Local-Specialist	0.0433 ± 0.0238	0.0504 ± 0.0344
	Dense	0.1541 ± 0.1636	0.1548 ± 0.1553
	Structured	0.1254 ± 0.0461	0.1272 ± 0.0430
	OpInf	0.0387	0.0534
	Global	0.1314	0.1299
Periodic	Proposed	$\mathbf{1.3251 \pm 0.2268}$	$\mathbf{2.3998 \pm 0.6599}$
	Dense	2.8791 ± 1.7269	4.9443 ± 3.1576
	Structured	2.8666 ± 1.4342	5.2810 ± 2.5679
	OpInf	114.0941	114.9897
	Global	2.7135	3.8990

4.2 CROSS-REGIME ROUTING AND PHYSICAL-SPACE FUSION.

Comparison for the cross-regime routing We evaluate whether T2-C improves prediction over selecting a single local predictor or combining two predictors with simpler weighting rules. On the centered-square benchmark, we fix the local specialist and the E2 router and consider the prescribed Steady–Hopf (S–H) and Hopf–Periodic (H–P) specialist pairs in their respective overlap regions. Each overlap contains 16 held-out windows with $K = 24$ prediction steps. The specialist pairs and their applicability regions are specified in advance. This comparison evaluates selection and fusion within these pairs, not automatic determination of the candidate set or the number of active specialists.

We compare seven assembly rules. E2 Top-1 selects the specialist with the larger E2 preference, whereas the E2-probability blend directly uses the normalized pair preferences as fusion weights. The equal blend assigns weight $1/2$ to each specialist. Train-fixed α fits one constant fusion weight per overlap using training windows. Train-fixed logit offset instead fits a constant additive offset to the E2 pair log-odds before applying the sigmoid; its resulting weight may vary with the parameter but not with the physical history at a fixed parameter. The validation-fixed specialist always uses the predictor with the lower aggregate validation loss. T2-C uses the initial-history descriptor together with the parameter and E2 preference to determine the fusion weight. All fusion rules combine independently predicted fields after reconstruction in physical space.

Table 3: Centered-square routing comparison at $K = 24$

Assembly rule	S–H	H–P
E2 Top-1	1.5786	1.6424
E2-probability blend	1.8145	1.4489
Equal blend	2.3849	1.3930
Train-fixed α	2.0591	0.9767
Train-fixed logit offset	2.0149	0.9821
Validation-fixed specialist	1.5786	1.0068
T2-C	**1.0954**	**0.8123**

Table 3 shows that T2-C achieves the lowest full-window joint error among the tested rules in both overlaps. Compared with E2 Top-1, it reduces the error from 1.5786% to 1.0954% in S–H and from 1.6424% to 0.8123% in H–P. Neither direct E2-probability weighting nor equal weighting consistently improves on selecting one specialist. T2-C also outperforms the training-fitted constant weight, constant logit offset, and validation-selected fixed specialist. These results show that the tested constant-mixture and calibration controls do not recover the accuracy of T2-C for these fixed specialist sets. To isolate the contribution of history information more directly, we next compare T2-C with an architecture-matched gate whose history inputs are removed.

Contribution of History-conditioned. We compare T2-C with an architecture-matched history-ablated control that retains the parameter and E2-preference inputs. With the local predictors and E2

fixed, T2-C reduces the mean full-window joint error by 39.64% on S–H and 16.85% on H–P over three paired gate-training seeds at $K = 24$. These results support the contribution of history information beyond parameter-dependent calibration on the evaluated windows. Appendix F.1 provides more details about the ablation protocol.

Physical-field diagnostics of fusion. To assess fusion beyond prediction error, we compare the Square specialists and T2-C using offline residuals and kinetic-energy error. T2-C has a lower pressure residual at both interfaces, but hopf specialist has a lower H–P momentum residual and a smaller S–H energy error as shown in F.4 Table 9. Figure 5 isolates the additional effect of blending: the mean energy deficit relative to weighted specialist energies is small, whereas nonlinear residual defects remain nonzero and vary over the rollout. Definitions and results are given in Appendices F.3 and F.4.

4.3 Internal expert routing and utilization

We analyze expert selection and weight allocation within the frozen original Circular H/P specialists, separately from E2 routing between regime-local specialists. Across the evaluated windows, Periodic specialist activates 18/18 velocity experts and 14/18 pressure experts, with 33 and 16 distinct group–pair combinations, respectively. The evaluated Hopf specialist uses a single expert group and selects a fixed Top-2 pair in each channel. Nevertheless, its mixing weights are nonuniform, and pressure-channel allocation varies across Reynolds numbers.

These observations reveal two forms of internal routing: broad expert utilization in Periodic specialist and fixed-pair, parameter-dependent mixing in Hopf specialist. Thus, adaptive weighting can remain active even when expert selection is unchanged. Appendix G.1 provides the sampling protocol and detailed selection and weight statistics.

4.4 Computational cost and timing boundaries

We distinguish active-parameter fractions, local inference latency, and a historical CFD timing reference. On the circular-cylinder benchmark, the Hopf and Periodic specialists activate 44.53% and 17.09% of their trainable parameters per prediction-path decision, respectively. For the Periodic specialist at batch size one and $K = 48$, median prediction times are 2.1848 s for the local specialist, 0.6579 s for Dense model, and 0.8652 s for Structured model. Despite skipping unselected experts, the sparse implementation executes multiple separate expert networks at each integration stage and incurs additional routing overhead. As a result, it takes 3.32× as long as Dense. A separate measurement including physical-history encoding takes 2.18747 s, compared with 1996 s for a historical FOM run over the same physical interval. The resulting ratio of approximately 912.5 is a historical FOM/warm-specialist comparison . Appendix H details the counting conventions, timing protocols, and comparison boundaries.

5 Conclusion

We introduce RAL-MoE-ROM, a hierarchical mixture-of-experts framework with two levels: regime-local specialists and cross-regime routing with fusion. Within each specialist, a projected Galerkin model is augmented with a structured sparse mixture-of-experts correction combining nonlinear neural, explicit linear, and low-rank quadratic branches, and trained through closed-loop multi-step rollouts. Across specialists, parameter-conditioned routing and initial-history-conditioned physical-space fusion allow specialists to evolve in their native coordinates without requiring a shared latent representation. Experiments on circular-cylinder, centered-square, and fluidic-pinball flows demonstrate accurate autonomous predictions across steady, Hopf-onset, and periodic regimes. Architectural ablations support the contribution of input-dependent local routing, while controlled overlap studies show that physical-history information improves fusion beyond parameter-only calibration and fixed weighting. Taken together, these results support a modular approach to regime-adaptive reduced-order modeling that couples locally specialized dynamics with history-informed prediction assembly. Future work will investigate learned regime partitions, longer-horizon reliability, and more efficient sparse execution.

Reproducibility statement

The model formulation and training procedure are described in Section 3. Appendices A–D provide the reduced-order formulation, routing details, dataset configurations, and training settings, while Appendices E–H describe the evaluation protocols, additional comparisons, and computational-cost measurements. We plan to publicly release the simulation, training, and evaluation code, together with the relevant datasets and configuration files, upon acceptance to facilitate independent verification and further research.

AI use statement

Generative AI tools, including ChatGPT and Codex, were used to assist with language polishing and LaTeX formatting. The authors take full responsibility for reviewing and verifying all scientific content, mathematical formulations, experimental configurations, numerical results, and conclusions, and for the final manuscript and all reported results.

References

David Amsallem and Charbel Farhat. Interpolation method for adapting reduced-order models and application to aeroelasticity. *AIAA Journal*, 46(7):1803–1813, 2008. doi: 10.2514/1.35374.

David Amsallem and Charbel Farhat. Stabilization of projection-based reduced-order models. *International Journal for Numerical Methods in Engineering*, 91(4):358–377, 2012. doi: 10.1002/nme.4274.

Yizhak Ben-Shabat, Chamin Hewa Koneputugodage, Sameera Ramasinghe, and Stephen Gould. Neural experts: Mixture of experts for implicit neural representations. In A. Globerson, L. Mackey, D. Belgrave, A. Fan, U. Paquet, J. Tomczak, and C. Zhang (eds.), *Advances in Neural Information Processing Systems*, volume 37, pp. 101641–101670. Curran Associates, Inc., 2024. doi: 10.52202/079017-3224. URL https://proceedings.neurips.cc/paper_files/paper/2024/file/b83fae17d73b079b1b98ab200276db9f-Paper-Conference.pdf.

Peter Benner, Serkan Gugercin, and Karen Willcox. A survey of projection-based model reduction methods for parametric dynamical systems. *SIAM Review*, 57(4):483–531, 2015. doi: 10.1137/130932715.

Gal Berkooz, Philip Holmes, and John L. Lumley. The proper orthogonal decomposition in the analysis of turbulent flows. *Annual Review of Fluid Mechanics*, 25:539–575, 1993. doi: 10.1146/annurev.fl.25.010193.002543.

Sourabh P. Bhat, Nicolas Barral, and Mario Ricchiuto. Error-based efficient parameter space partitioning for mesh adaptation and local reduced order models. *Computer Methods in Applied Mechanics and Engineering*, 435:117649, 2025. doi: 10.1016/j.cma.2024.117649.

Steven L. Brunton, Bernd R. Noack, and Petros Koumoutsakos. Machine learning for fluid mechanics. *Annual Review of Fluid Mechanics*, 52:477–508, 2020. doi: 10.1146/annurev-fluid-010719-060214.

Nithin Chalapathi, Yiheng Du, and Aditi Krishnapriyan. Scaling physics-informed hard constraints with mixture-of-experts. In B. Kim, Y. Yue, S. Chaudhuri, K. Fragkiadaki, M. Khan, and Y. Sun (eds.), *International Conference on Learning Representations*, volume 2024, pp. 36015–36033, 2024. URL https://proceedings.iclr.cc/paper_files/paper/2024/file/9aeda582add763c41c7b39691ce19ab0-Paper-Conference.pdf.

Kathleen Champion, Bethany Lusch, J. Nathan Kutz, and Steven L. Brunton. Data-driven discovery of coordinates and governing equations. *Proceedings of the National Academy of Sciences*, 116 (45):22445–22451, 2019. doi: 10.1073/pnas.1906995116.

Kyunghyun Cho, Bart van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. Learning phrase representations using RNN encoder–decoder

for statistical machine translation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 1724–1734. Association for Computational Linguistics, 2014. doi: 10.3115/v1/D14-1179.

Antonio Colanera and Luca Magri. Quantized local reduced-order modeling in time (ql-rom). *Computer Methods in Applied Mechanics and Engineering*, 447:118393, 2025. doi: 10.1016/j.cma.2025.118393.

Zulkeefal Dar, Joan Baiges, and Ramon Codina. Artificial neural network based correction for reduced order models in computational fluid mechanics. *Computer Methods in Applied Mechanics and Engineering*, 415:116232, 2023. doi: 10.1016/j.cma.2023.116232.

Alejandro N. Diaz, Youngsoo Choi, and Matthias Heinkenschloss. A fast and accurate domain decomposition nonlinear manifold reduced order model. *Computer Methods in Applied Mechanics and Engineering*, 425:116943, 2024. doi: 10.1016/j.cma.2024.116943.

Karthik Duraisamy, Gianluca Iaccarino, and Heng Xiao. Turbulence modeling in the age of data. *Annual Review of Fluid Mechanics*, 51:357–377, 2019. doi: 10.1146/annurev-fluid-010518-040547.

William Fedus, Barret Zoph, and Noam Shazeer. Switch transformers: Scaling to trillion parameter models with simple and efficient sparsity. *Journal of Machine Learning Research*, 23(120):1–39, 2022. URL https://jmlr.org/papers/v23/21-0998.html.

Stefania Fresca and Andrea Manzoni. POD-DL-ROM: Enhancing deep learning-based reduced order models for nonlinear parametrized PDEs by proper orthogonal decomposition. *Computer Methods in Applied Mechanics and Engineering*, 388:114181, 2022. doi: 10.1016/j.cma.2021.114181.

Rudy Geelen, Stephen J. Wright, and Karen E. Willcox. Operator inference for non-intrusive model reduction with quadratic manifolds. *Computer Methods in Applied Mechanics and Engineering*, 403:115717, 2023. doi: 10.1016/j.cma.2022.115717.

Jan S. Hesthaven and Stefano Ubbiali. Non-intrusive reduced order modeling of nonlinear problems using neural networks. *Journal of Computational Physics*, 363:55–78, 2018. doi: 10.1016/j.jcp.2018.02.037.

Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural Computation*, 9(8):1735–1780, 1997. doi: 10.1162/neco.1997.9.8.1735.

Robert A. Jacobs, Michael I. Jordan, Steven J. Nowlan, and Geoffrey E. Hinton. Adaptive mixtures of local experts. *Neural Computation*, 3(1):79–87, 1991. doi: 10.1162/neco.1991.3.1.79.

Michael I. Jordan and Robert A. Jacobs. Hierarchical mixtures of experts and the EM algorithm. *Neural Computation*, 6(2):181–214, 1994. doi: 10.1162/neco.1994.6.2.181.

Eurika Kaiser, Bernd R. Noack, Laurent Cordier, Andreas Spohn, Marc Segond, Markus Abel, Guillaume Daviller, Jan Östh, Siniša Krajnović, and Robert K. Niven. Cluster-based reduced-order modelling of a mixing layer. *Journal of Fluid Mechanics*, 754:365–414, 2014. doi: 10.1017/jfm.2014.355.

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. URL https://jmlr.org/papers/v24/21-1524.html.

Kookjin Lee and Kevin T. Carlberg. Model reduction of dynamical systems on nonlinear manifolds using deep convolutional autoencoders. *Journal of Computational Physics*, 404:108973, 2020. doi: 10.1016/j.jcp.2019.108973.

Hao Li, Daniel Fernex, Richard Semaan, Jianguo Tan, Marek Morzyński, and Bernd R. Noack. Cluster-based network model. *Journal of Fluid Mechanics*, 906:A21, 2021a. doi: 10.1017/jfm.2020.785.

Siyang Li, Yize Chen, Yan Guo, Ming Huang, and Hui Xiong. Towards generalizable pde dynamics forecasting via physics-guided invariant learning. In C. Vondrick, B. Hariharan, C. Raffel, L. Pinto, D. Yang, and A. Faust (eds.), *International Conference on Learning Representations*, volume 2026, pp. 72050–72079, 2026. URL https://proceedings.iclr.cc/paper_files/paper/2026/file/754612bde73a8b65ad8743f1f6d8ddf6-Paper-Conference.pdf.

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*, 2021b. URL https://arxiv.org/abs/2010.08895.

Lu Lu, Pengzhan Jin, Guofei Pang, Zhongqiang Zhang, and George Em Karniadakis. Learning nonlinear operators via DeepONet based on the universal approximation theorem of operators. *Nature Machine Intelligence*, 3:218–229, 2021. doi: 10.1038/s42256-021-00302-5.

Xinyu Pan and Dunhui Xiao. Domain decomposition for physics-data combined neural network based parametric reduced order modelling. *Journal of Computational Physics*, 519:113452, 2024.

Jun Sur Richard Park, Siu Wun Cheung, Youngsoo Choi, and Yeonjong Shin. tLaSDI: Thermodynamics-informed latent space dynamics identification. *Computer Methods in Applied Mechanics and Engineering*, 429:117144, 2024. doi: 10.1016/j.cma.2024.117144.

Benjamin Peherstorfer and Karen Willcox. Data-driven operator inference for nonintrusive projection-based model reduction. *Computer Methods in Applied Mechanics and Engineering*, 306:196–215, 2016. doi: 10.1016/j.cma.2016.03.025.

Benjamin Peherstorfer, Daniel Butnaru, Karen Willcox, and Hans-Joachim Bungartz. Localized discrete empirical interpolation method. *SIAM Journal on Scientific Computing*, 36(1):A168–A192, 2014. doi: 10.1137/130924408.

Sk. Mashfiqur Rahman, Suraj Pawar, Omer San, Adil Rasheed, and Traian Iliescu. Nonintrusive reduced order modeling framework for quasigeostrophic turbulence. *Physical Review E*, 100(5): 053306, 2019. doi: 10.1103/PhysRevE.100.053306.

Noam Shazeer, Azalia Mirhoseini, Krzysztof Maziarz, Andy Davis, Quoc Le, Geoffrey Hinton, and Jeff Dean. Outrageously large neural networks: The sparsely-gated mixture-of-experts layer. In *International Conference on Learning Representations*, 2017. URL https://openreview.net/forum?id=B1ckMDqlg.

Lawrence Sirovich. Turbulence and the dynamics of coherent structures. part i: Coherent structures. *Quarterly of Applied Mathematics*, 45(3):561–571, 1987. doi: 10.1090/qam/910462.

Hong Wang, Haiyang Xin, Jie Wang, Xuanze Yang, Fei Zha, huanshuo dong, and Yan Jiang. Mixture-of-experts operator transformer for large-scale pde pre-training. In D. Belgrave, C. Zhang, H. Lin, R. Pascanu, P. Koniusz, M. Ghassemi, and N. Chen (eds.), *Advances in Neural Information Processing Systems*, volume 38, Main Conference, pp. 31498–31527. Curran Associates, Inc., 2025. doi: 10.52202/085713-1057. URL https://proceedings.neurips.cc/paper_files/paper/2025/file/2d23a9991a6f64482bf395628e279f5f-Paper-Conference.pdf.

A Full problem setup and weighted POD construction

A.1 Parameterized full-order model, gauge, and regime coverage

Finite-volume spatial discretization of the Navier–Stokes equations in Eq. (1) yields a semi-discrete system, as in Eq. (2), governing the velocity $\boldsymbol{u}(t;\mu) \in \mathbb{R}^{N_u}$ and pressure $p(t;\mu) \in \mathbb{R}^{N_p}$:

$$\dot{\boldsymbol{u}} = \boldsymbol{f}_h(\mu) + \mathcal{L}_h(\mu)\boldsymbol{u} - \mathcal{N}_h(\boldsymbol{u},\boldsymbol{u};\mu) - \nabla_h(\mu)p, \qquad \mathrm{div}_h(\mu)\boldsymbol{u} = 0. \tag{20}$$

Here $\mathcal{L}_h$ and $\mathcal{N}_h$ denote the spatially discretized viscous and convective operators, while ∇_h and div_h denote the pressure gradient and velocity divergence operators. The dot denotes a continuous-time derivative; time advancement of the reduced model is specified in Section 3.3. Pressure is

defined only up to an additive constant. With positive finite-volume weights $\boldsymbol{w}$, each pressure snapshot is mapped to the weighted zero-mean gauge

$$\Gamma_p(q) = q - \frac{\boldsymbol{w}^\top q}{\boldsymbol{w}^\top \mathbf{1}}\mathbf{1}, \qquad p^\circ = \Gamma_p(p), \qquad \boldsymbol{w}^\top p^\circ = 0. \tag{21}$$

The parameter domain is covered by overlapping regime-local parameter regions,

$$\mathcal{M} \subseteq \bigcup_{r\in\{S,H,P\}} \mathcal{M}_r, \qquad \mathcal{O}_{SH} = \mathcal{M}_S \cap \mathcal{M}_H, \qquad \mathcal{O}_{HP} = \mathcal{M}_H \cap \mathcal{M}_P, \tag{22}$$

where S, H, and P denote the steady, Hopf-onset, and periodic regimes, respectively. Only adjacent overlaps are used; no direct S–P overlap is considered.

A.2 Area-weighted local POD

The finite-volume quadrature induces the weighted inner products

$$\langle \boldsymbol{v}, \boldsymbol{z}\rangle_{M_u} = \boldsymbol{v}^\top M_u \boldsymbol{z}, \qquad \langle q, s\rangle_{M_p} = q^\top M_p s, \qquad M_p = \mathrm{diag}(\boldsymbol{w}), \qquad M_u = \mathrm{diag}(M_p, M_p). \tag{23}$$

For chart r, let $d_u^{(r)}$ and $d_p^{(r)}$ denote the retained velocity and pressure dimensions. Let $X_u^{(r)}$ and $X_p^{(r)}$ collect the centered velocity and gauge-corrected pressure snapshots from the training trajectories assigned to that chart. The corresponding chart-specific means are denoted by $\bar{\boldsymbol{u}}^{(r)}$ and $\bar{p}^{(r),\circ}$. The local POD bases solve

$$\Phi_u^{(r)} \in \underset{\Psi^\top M_u \Psi = I_{d_u^{(r)}}}{\arg\min} \left\| X_u^{(r)} - \Psi\Psi^\top M_u X_u^{(r)} \right\|_{M_u,F}^2, \tag{24}$$

$$\Phi_p^{(r)} \in \underset{\Psi^\top M_p \Psi = I_{d_p^{(r)}}}{\arg\min} \left\| X_p^{(r)} - \Psi\Psi^\top M_p X_p^{(r)} \right\|_{M_p,F}^2, \tag{25}$$

where

$$\|Y\|_{M,F}^2 = \mathrm{tr}(Y^\top M Y). \tag{26}$$

Equivalently, consider the weighted singular value decompositions

$$M_u^{1/2} X_u^{(r)} = U_u^{(r)} \Sigma_u^{(r)} {V_u^{(r)}}^\top, \qquad M_p^{1/2} X_p^{(r)} = U_p^{(r)} \Sigma_p^{(r)} {V_p^{(r)}}^\top. \tag{27}$$

The retained bases are

$$\Phi_u^{(r)} = M_u^{-1/2} U_u^{(r)}(:, 1{:}d_u^{(r)}), \qquad \Phi_p^{(r)} = M_p^{-1/2} U_p^{(r)}(:, 1{:}d_p^{(r)}), \tag{28}$$

and satisfy

$${\Phi_u^{(r)}}^\top M_u \Phi_u^{(r)} = I, \qquad {\Phi_p^{(r)}}^\top M_p \Phi_p^{(r)} = I. \tag{29}$$

Because the pressure snapshots and their chart mean satisfy the same weighted gauge condition, the pressure basis also satisfies

$$\boldsymbol{w}^\top \Phi_p^{(r)} = \mathbf{0}^\top. \tag{30}$$

Encoding and decoding are performed independently within each chart:

$$\boldsymbol{a}^{(r)} = {\Phi_u^{(r)}}^\top M_u\big(\boldsymbol{u} - \bar{\boldsymbol{u}}^{(r)}\big), \qquad \mathcal{D}_r^u(\boldsymbol{a}^{(r)}) = \bar{\boldsymbol{u}}^{(r)} + \Phi_u^{(r)} \boldsymbol{a}^{(r)}, \tag{31}$$

$$\boldsymbol{b}^{(r)} = {\Phi_p^{(r)}}^\top M_p\big(\Gamma_p(p) - \bar{p}^{(r),\circ}\big), \qquad \mathcal{D}_r^p(\boldsymbol{b}^{(r)}) = \bar{p}^{(r),\circ} + \Phi_p^{(r)} \boldsymbol{b}^{(r)}. \tag{32}$$

The reduced coefficients are standardized using chart-local statistics estimated from the corresponding training trajectories:

$$\widetilde{\boldsymbol{a}}^{(r)} = \big(\boldsymbol{a}^{(r)} - \boldsymbol{\mu}_a^{(r)}\big) \oslash \boldsymbol{\sigma}_a^{(r)}, \qquad \widetilde{\boldsymbol{b}}^{(r)} = \big(\boldsymbol{b}^{(r)} - \boldsymbol{\mu}_b^{(r)}\big) \oslash \boldsymbol{\sigma}_b^{(r)}. \tag{33}$$

Additional specialist features and their normalization are specified in Section 3.2.

B Projected Galerkin and pressure operators

B.1 Chart-local Galerkin momentum dynamics

For each chart $r \in \{S, H, P\}$, substituting the chart-local affine reconstructions from Appendix A into the semi-discrete momentum Eq. (20) and testing against the local velocity basis in the M_u inner product yields

$$\dot{\boldsymbol{a}}^{(r)} = \boldsymbol{c}_r(\mu) + A_r(\mu)\boldsymbol{a}^{(r)} + H_r(\mu) : \big(\boldsymbol{a}^{(r)} \otimes \boldsymbol{a}^{(r)}\big) + C_r^p(\mu)\boldsymbol{b}^{(r)}. \tag{34}$$

Here ":" denotes the double contraction. For velocity modes $\boldsymbol{\phi}_i^{u,(r)}$, $i = 1, \dots, d_u^{(r)}$, and pressure modes $\phi_\ell^{p,(r)}$, $\ell = 1, \dots, d_p^{(r)}$, the projected operators are

$$[\boldsymbol{c}_r(\mu)]_i = \Big\langle \boldsymbol{\phi}_i^{u,(r)}, \boldsymbol{f}_h(\mu) + \mathcal{L}_h(\mu)\bar{\boldsymbol{u}}^{(r)} - \mathcal{N}_h\big(\bar{\boldsymbol{u}}^{(r)}, \bar{\boldsymbol{u}}^{(r)}; \mu\big) - \nabla_h(\mu)\bar{p}^{(r),\circ} \Big\rangle_{M_u}, \tag{35}$$

$$[A_r(\mu)]_{ij} = \Big\langle \boldsymbol{\phi}_i^{u,(r)}, \mathcal{L}_h(\mu)\boldsymbol{\phi}_j^{u,(r)} - \mathcal{N}_h\big(\bar{\boldsymbol{u}}^{(r)}, \boldsymbol{\phi}_j^{u,(r)}; \mu\big) - \mathcal{N}_h\big(\boldsymbol{\phi}_j^{u,(r)}, \bar{\boldsymbol{u}}^{(r)}; \mu\big) \Big\rangle_{M_u}, \tag{36}$$

$$[H_r(\mu)]_{ijk} = -\Big\langle \boldsymbol{\phi}_i^{u,(r)}, \mathcal{N}_h\big(\boldsymbol{\phi}_j^{u,(r)}, \boldsymbol{\phi}_k^{u,(r)}; \mu\big) \Big\rangle_{M_u}, \tag{37}$$

$$[C_r^p(\mu)]_{i\ell} = -\Big\langle \boldsymbol{\phi}_i^{u,(r)}, \nabla_h(\mu)\phi_\ell^{p,(r)} \Big\rangle_{M_u}. \tag{38}$$

We denote the resulting chart-local projected momentum vector field by

$$\mathcal{F}_r^G\big(\boldsymbol{a}^{(r)}, \boldsymbol{b}^{(r)}; \mu\big) = \boldsymbol{c}_r(\mu) + A_r(\mu)\boldsymbol{a}^{(r)} + H_r(\mu) : \big(\boldsymbol{a}^{(r)} \otimes \boldsymbol{a}^{(r)}\big) + C_r^p(\mu)\boldsymbol{b}^{(r)}. \tag{39}$$

This is the projected Galerkin equation summarized in Eq. (4); the additive hybrid specialist augments it with the learned closure in Eq. (5).

B.2 Reduced pressure–Poisson equation

Taking the divergence of the incompressible momentum equation with spatially constant viscosity gives

$$-\Delta p^\circ = \nabla \cdot \big[(\boldsymbol{u} \cdot \nabla)\boldsymbol{u} - \boldsymbol{f}\big], \qquad \int_\Omega p^\circ \, \mathrm{d}\Omega = 0, \tag{40}$$

where p° denotes gauge-corrected pressure. Substituting Eq. (32) into Eq. (40), and applying local pressure-mode testing, integration by parts, and Appendix A quadrature for the volume integrals, yields a reduced pressure system.

$$K_r^p(\mu)\boldsymbol{b}^{PP,(r)} = \boldsymbol{d}_r(\mu) + B_r(\mu)\boldsymbol{a}^{(r)} + Q_r(\mu) : \big(\boldsymbol{a}^{(r)} \otimes \boldsymbol{a}^{(r)}\big), \tag{41}$$

where

$$[K_r^p]_{\ell m} = -\big\langle \nabla_h\phi_\ell^{p,(r)}, \nabla_h\phi_m^{p,(r)} \big\rangle_{M_u}, \tag{42}$$

$$[\boldsymbol{d}_r]_\ell = \big\langle \nabla_h\phi_\ell^{p,(r)}, \mathcal{N}_h(\bar{\boldsymbol{u}}^{(r)}, \bar{\boldsymbol{u}}^{(r)}; \mu) - \boldsymbol{f}_h(\mu) + \nabla_h\bar{p}^{(r),\circ} \big\rangle_{M_u}, \tag{43}$$

$$[B_r]_{\ell j} = \big\langle \nabla_h\phi_\ell^{p,(r)}, \mathcal{N}_h(\bar{\boldsymbol{u}}^{(r)}, \phi_j^{u,(r)}; \mu) + \mathcal{N}_h(\phi_j^{u,(r)}, \bar{\boldsymbol{u}}^{(r)}; \mu) \big\rangle_{M_u}, \tag{44}$$

$$[Q_r]_{\ell jk} = \big\langle \nabla_h\phi_\ell^{p,(r)}, \mathcal{N}_h(\phi_j^{u,(r)}, \phi_k^{u,(r)}; \mu) \big\rangle_{M_u}. \tag{45}$$

The resulting algebraic pressure map is

$$\mathcal{Q}_r^{PP}(\boldsymbol{a}^{(r)}; \mu) = \big(K_r^p(\mu)\big)^\dagger \Big[\boldsymbol{d}_r(\mu) + B_r(\mu)\boldsymbol{a}^{(r)} + Q_r(\mu) : \big(\boldsymbol{a}^{(r)} \otimes \boldsymbol{a}^{(r)}\big)\Big], \tag{46}$$

where † denotes the Moore–Penrose pseudoinverse and coincides with the inverse when K_r^p is nonsingular. The pressure field is then reconstructed as $\mathcal{Q}_r^{PP}(\boldsymbol{a}^{(r)}; \mu)$. Thus, pressure is recovered algebraically from the local velocity state rather than advanced as an independent dynamical variable.

C OUTER ROUTING AND T2-C DETAILS

Chart-independent physical-history descriptor. The implementation in Section 3.4 uses eight descriptor channels and one parameter input. These features are computed from the initial three physical states at $t_{n-2} < t_{n-1} < t_n$, with pressure mapped to the common zero-mean gauge. They are independent of the specialists' POD coordinates. All quantities below use the nondimensional data conventions of the experiment, and $\epsilon = 10^{-12}$ is the numerical floor.

Define $A_\Omega = \sum_i w_i$, $U_j = \|\boldsymbol{u}_j\|^2_{M_u}$, $P_j = \|p_j^\circ\|^2_{M_p}$, and

$$h_j = \max(t_j - t_{j-1}, \epsilon), \quad s_j^u = \sqrt{\max(U_j, \epsilon)}, \qquad s_j^p = \sqrt{\max(P_j, \epsilon)}, \tag{47}$$

$$v_j = (\boldsymbol{u}_j - \boldsymbol{u}_{j-1})/h_j, \qquad g_j = \frac{\log\max(U_j, \epsilon) - \log\max(U_{j-1}, \epsilon)}{h_j}. \tag{48}$$

The ordered raw descriptor is

$$\boldsymbol{s}_n = \mathrm{col}\begin{pmatrix} \frac{1}{2}\log\max(U_n/A_\Omega, \epsilon), \\ \frac{1}{2}\log\max(P_n/A_\Omega, \epsilon), \\ \log(1 + \|v_n\|_{M_u}/(s_n^u + \epsilon)), \\ \log(1 + \|v_{n-1}\|_{M_u}/(s_{n-1}^u + \epsilon)), \\ \tanh(g_n), \\ \tanh(g_n - g_{n-1}), \\ \log(1 + \|p_n^\circ - p_{n-1}^\circ\|_{M_p}/[h_n(s_n^p + \epsilon)]), \\ \log(1 + \|v_n - v_{n-1}\|_{M_u}/(s_n^u + \epsilon)) \end{pmatrix}. \tag{49}$$

The last channel measures a difference of velocity rates: it is not divided again by a time interval and is not the second-derivative descriptor in an acceleration discretization. The log and hyperbolic-tangent transformations above are part of the actual feature construction. For the gate, the nine raw features $[\mu; \boldsymbol{s}_n]$ are standardized using the means and population standard deviations of training windows only. Any feature standard deviation below 10^{-8} is replaced by one. We denote the resulting vector by $[\widetilde{\mu}_G; \boldsymbol{d}_n]$; its parameter normalization is fitted for the gate and is distinct from the E2 training-row normalization $\widetilde{\mu}_R$.

History-conditioned gate and output fusion. The logit-correction network receives $[\widetilde{\mu}_G; \boldsymbol{d}_n] \in \mathbb{R}^9$ and uses two hidden layers of width 64 with SiLU activations followed by a scalar linear output (9–64–64–1; 4,865 trainable parameters). The final layer is initialized at zero, so the initial correction vanishes and the fusion weight reduces to the E2 pair preference. Specifically,

$$\Delta_\psi = c_\psi\left([\widetilde{\mu}_G; \boldsymbol{d}_n]\right), \qquad \alpha = \mathrm{sigmoid}\left(\ell^{E2} + \Delta_\psi\right). \tag{50}$$

The two selected specialists encode the same physical history in their respective local charts and then evolve independently. The scalar α is fixed throughout the prediction window and is shared by velocity and pressure. After reconstruction on the common physical mesh, T2-C forms

$$\widehat{\boldsymbol{q}}_\alpha^c(t) = \alpha\,\widehat{\boldsymbol{q}}_1^c(t) + (1-\alpha)\,\widehat{\boldsymbol{q}}_2^c(t), \qquad c \in \{u, p\}, \tag{51}$$

where the pressure field is gauge corrected. The fused field is an output only and is never projected back into either specialist.

D REPRODUCIBILITY AND IMPLEMENTATION DETAILS

Physical-field error metrics. We evaluate reconstructed velocity and gauge-corrected pressure using the relative weighted field errors

$$E_u(n) = 100\frac{\|\widehat{\boldsymbol{u}}_n - \boldsymbol{u}_n\|_{M_u}}{\|\boldsymbol{u}_n\|_{M_u}}, \qquad E_p(n) = 100\frac{\|\widehat{p}_n^\circ - p_n^\circ\|_{M_p}}{\|p_n^\circ\|_{M_p}}. \tag{52}$$

Here $\|\boldsymbol{v}\|_M = (\boldsymbol{v}^\top M \boldsymbol{v})^{1/2}$ uses the finite-volume quadrature introduced in Appendix A; the velocity norm includes both components, and p° denotes pressure in the common zero-mean gauge. Unless stated otherwise, benchmark-specific temporal and trajectory aggregation is reported with the corresponding experiment. Whenever a joint score is reported, $E_{\mathrm{joint}} = E_u + E_p$.

Trajectory-level partitioning. Table 4 summarizes the resulting chart-wise parameter coverage, reduced dimensions, trajectory splits, and evaluation horizons. All splits are defined at the level of complete parameter trajectories rather than individual snapshots or rollout windows. Local bases, normalization statistics, and model parameters are constructed from training trajectories only.

Table 4: Chart-wise parameter coverage, reduced dimensions, trajectory splits, and evaluation horizons. K denotes the number of autonomous rollout steps used for the corresponding local evaluation.

Benchmark	Chart	Re range	(r_u, r_p)	Train	Val.	Test	K
Circular cylinder	S	20.0–46.4	$(32, 32)$	14	2	4	56
	H	45.5–59.2	$(32, 32)$	29	2	3	56
	P	57.3–200	$(32, 32)$	53	6	4	48
Centered square	S	50–95.4	$(5, 4)$	60	5	4	16
	H	94–102	$(11, 11)$	23	6	5	48
	P	98.5–150	$(28, 26)$	27	6	4	24
Fluidic pinball	S	10–19	$(4, 3)$	37	8	7	56
	H	16.6–25	$(5, 5)$	39	8	7	56
	P	21.25–60	$(17, 16)$	47	9	9	32

Benchmark discretization and sampling. The centered-square dataset contains 120 complete trajectories over $Re \in [50, 150]$ on a 9,400-cell finite-volume mesh. The CFD time step is $\delta t_{\mathrm{CFD}} = 0.02$, and snapshots are stored every 200 CFD steps, giving $\Delta t_{\mathrm{snap}} = 4$. The fluidic-pinball dataset contains 131 complete trajectories over $Re \in [10, 60]$ on a 70,194-cell mesh, with $\delta t_{\mathrm{CFD}} = 0.005$ and a snapshot interval of 0.25 physical time units. The circular-cylinder dataset contains 100 Reynolds-parameterized trajectories spanning $Re \in [20, 200]$ on a common finite-volume mesh represented by 97,368 spatial points. Trajectories contain different numbers of snapshots, yielding 14,167 snapshots in total. The unsteady simulations employ adaptive CFD time stepping subject to $Co_{\max} = 0.8$ and $\delta t_{\max} = 0.2$, while the snapshot interval is adjusted according to the estimated vortex-shedding period, with eight snapshots retained per shedding cycle. The chart-wise parameter ranges, reduced dimensions, splits, and evaluation horizons are summarized in Table 4.

Specialist optimization. Table 5 summarizes the reported training configurations, which use AdamW with chart-specific learning rates, weight decay, and rollout curricula.

Table 5: Specialist training configurations. "Selected" denotes the validation-selected checkpoint.

Specialist	Initial lr	Weight decay	Rollout curriculum	Budget	Selected
Circular S	1.55×10^{-5}	1.0×10^{-4}	4,8,16	3600 steps	1200 steps
Circular H	1.0×10^{-3}	1.0×10^{-4}	1,2,4,8	8000 steps	7200 steps
Circular P	5.5×10^{-4}	1.5×10^{-4}	4,8,12,16	240 epochs	85 epochs
Square S	1.55×10^{-5}	1.0×10^{-4}	1,4,8,16	4000 steps	3200 steps
Square H	1.0×10^{-3}	1.0×10^{-4}	1,2,4,8	8000 steps	7200 steps
Square P	5.5×10^{-4}	1.5×10^{-4}	4,8,12,16	720 epochs	490 epochs
Pinball S	1.0×10^{-3}	1.0×10^{-4}	4,8,16	8000 steps	400 steps
Pinball H	1.0×10^{-3}	1.0×10^{-4}	4,8,16,32,56	8000 steps	8000 steps
Pinball P	1.0×10^{-3}	1.0×10^{-4}	4,8,16,24,32	8000 steps	8000 steps

Outer E2 router configuration. The parameter-only router is an MLP $1 \to 24 \to 3$ with a Tanh hidden activation and 123 trainable parameters. The scalar Reynolds number is standardized using training-row mean and population standard deviation; a standard deviation below 10^{-7} is replaced by one. Within each chart and split, each unique Reynolds number supplies one row with that chart label. A parameter occurring in more than one chart can therefore produce multiple labeled rows, rather than an assumed unique bifurcation label. The loss is class-weighted cross entropy

plus $0.05\ \mathrm{mean}(\pi_S \pi_P)$, with class weight $N/(3N_c)$. Training uses AdamW (learning rate 3×10^{-4}, weight decay 10^{-4}), a cosine schedule over 8,000 steps ending at 1.5×10^{-5}, sampling with replacement in batches of $\min(256, N)$, and gradient-norm clipping at one. Selection maximizes validation balanced accuracy plus macro-F1 minus 10^{-3} times the current training-batch loss, rather than validation cross entropy alone.

Specialist architectures. For sparse MoE, $D = 13+5(2d_u+d_p)$ is feature width. Features include parameter terms, current coefficients and projected Galerkin equation, 11 derived norm/energy/ratio quantities, and two historical blocks. The time step is an integrator input, not a network feature. Each encoder has three linear layers of width h with LayerNorm and SiLU; the first two include dropout 0.04. Two refinement residual blocks follow. Velocity and pressure share this encoder, but not their router or expert parameters.

Each group contains six routed experts and one shared expert per channel, with separate velocity/pressure Top-2 supports. Routers receive $[\boldsymbol{z}; \boldsymbol{\xi}]$ and use LayerNorm, a width-h SiLU hidden layer, dropout 0.04, and a linear logit head. Each expert projects $[\boldsymbol{z}; \boldsymbol{s}^c]$ to width h, applies B expanded residual feed-forward blocks with GELU, and maps to the channel output. Residual scales are learned scalars initialized to 0.1. The explicit linear branch has no bias; the two quadratic factors each have shape $d_c \times 4 \times \dim(\boldsymbol{s}^c)$, with $\kappa = 0.05$ and no additional output projection. The branches are added within each expert before sparse mixture weighting. Outputs are converted from standardized target coordinates before combination with physical-model terms.

E Local prediction: protocols and supplementary comparisons

E.1 Circular-cylinder local prediction: protocol and baseline comparisons

Evaluation protocol. Hopf comprises 42 windows at $Re = 47.1, 49.0, 51.8$, while Periodic comprises 32 windows at $Re = 70.3, 100.4, 149.1, 189.9$. Each window contains 48 prediction steps; the $K = 24$ evaluation uses the first 24 steps of the same rollout. Table 5 summarizes the archived original training configurations.

Dense model matches the active neural parameter count, rather than total stored parameters or execution time. Structured model retains explicit response branches but removes input-dependent expert routing. Table 6 supplements the main-text joint-error comparison in Table 2 with periodic velocity and pressure errors and additional baselines.

Table 6: Periodic regime velocity and pressure errors on flow past a cylinder case(%).

	$K = 24$		$K = 48$	
Model	E_u	E_p	E_u	E_p
Local Specialist	$\mathbf{0.2446 \pm 0.0323}$	$\mathbf{1.0805 \pm 0.1946}$	$\mathbf{0.4790 \pm 0.1208}$	$\mathbf{1.9208 \pm 0.5391}$
Dense	0.5731 ± 0.3638	2.3060 ± 1.3681	1.0107 ± 0.6513	3.9337 ± 2.5070
Structured	0.5178 ± 0.2667	2.3489 ± 1.1688	1.0405 ± 0.5015	4.2406 ± 2.0664
POD–LSTM	0.5259 ± 0.0512	2.3124 ± 0.1686	0.6879 ± 0.0712	2.8872 ± 0.2677
POD–GRU	0.5467 ± 0.0550	2.4169 ± 0.1755	0.7619 ± 0.0855	3.1817 ± 0.3186
ql-ROM	2.1471 ± 0.0217	20.7538 ± 0.0817	4.0164 ± 0.1313	24.6756 ± 0.2531
FNO	1.5428 ± 0.5461	7.6426 ± 2.2317	2.5059 ± 1.0956	11.5579 ± 4.4028
OpInf	15.8676	98.2264	17.6639	97.3258

Comparison with baseline models. Table 6 shows that the local specialist achieves the lowest mean velocity and pressure errors at both horizons. At $K = 48$, it reduces these errors by 52.6% and 51.2% relative to the Dense model, showing that the improvement is not explained by active parameter count alone. Lower errors than structured model support the benefit of input-dependent routing under the evaluated configuration.

The comparison with recurrent baselines provides complementary evidence. At $K = 48$, the local specialist reduces velocity and pressure errors by approximately 30.4% and 33.5%, respectively, relative to POD–LSTM, the more accurate recurrent baseline. Thus, explicitly modeling temporal dependence with a recurrent network does not recover the same accuracy in these experiments.

The larger errors of the implemented ql-ROM, FNO and OpInf baselines, particularly for pressure, further highlight the effectiveness of the proposed hybrid velocity–pressure predictor under the evaluated configurations. Together, these results support improved finite-horizon prediction beyond both the architectural controls and the tested alternative modeling approaches.

Baseline implementations. We compare against ql-ROM, OpInf, POD–LSTM/GRU, and FNO. Our ql-ROM adaptation (Colanera & Magri, 2025) uses eight centroid-centered local Galerkin charts with 16 modes each within a training-only 32-dimensional POD space, state-dependent switching, and projected-PPE pressure recovery. OpInf (Peherstorfer & Willcox, 2016) fits ridge-regularized quadratic and parameter-dependent dynamics with a separate pressure regression; both methods use RK4 with four substeps. POD–LSTM/GRU (Rahman et al., 2019; Hochreiter & Schmidhuber, 1997; Cho et al., 2014) predict increments of 32 velocity and 32 pressure coefficients from three-frame histories using 128 hidden units. FNO (Li et al., 2021b) predicts field increments using four Fourier layers of width 48 with 12 modes per direction on a 1536×1024 grid. The recurrent and Fourier baselines use validation-selected checkpoints, three training seeds, and 4000 optimization steps with closed-loop rollouts.

F Routing and fusion: protocols and supplementary results

F.1 Supplementary analysis for accuracy, routing, and history ablation on centered-square case

This subsection complements the full-window assembly comparison in Section 4.2. We first examine accuracy at the final prediction step, then resolve the results by Reynolds number and gate-training seed, and finally present representative reconstructed fields. The numerical tables use the same Square test windows and specialist predictors as the main-text study, rather than an additional independent test set.

Comparison for the cross-regime routing. Table 7 reports velocity, pressure, and joint errors at $K = 24$, averaged over the 16 held-out windows in each overlap. T2-C has the lowest terminal velocity and the more accurate individual specialist differs between overlaps: Hopf is preferable in S–H, whereas Periodic is preferable in H–P. T2-C improves on both of these single-specialist references, indicating that its benefit is not limited to correcting the E2 Top-1 choice.

Table 7: Centered-square terminal-step errors at $K = 24$ (%). The convex oracle provides an a posteriori reference using the true fields. Its weight minimizes a full-window objective and is evaluated here at the terminal step. T2-C μ-only uses the same gate architecture as T2-C but removes the standardized history descriptor.

Overlap	Predictor	E_u	E_p	E_{joint}
S–H	Steady specialist	2.5976	2.2450	4.8426
	Hopf specialist	0.4942	1.2356	1.7298
	E2 Top-1	0.4942	1.2356	1.7298
	Equal blend	1.3644	1.6677	3.0321
	E2-probability blend	0.7898	1.3949	2.1847
	T2-C μ-only	0.7898	1.3948	2.1846
	T2-C	**0.4615**	**0.7343**	**1.1958**
	Convex oracle	0.4659	0.7183	1.1842
H–P	Periodic specialist	0.6420	0.5659	1.2079
	Hopf specialist	1.0034	1.8684	2.8718
	E2 Top-1	0.7422	1.1052	1.8473
	Equal blend	0.6077	1.0069	1.6146
	E2-probability blend	0.6154	1.0464	1.6618
	T2-C μ-only	0.5875	0.5629	1.1504
	T2-C	**0.4509**	**0.3956**	**0.8465**
	Convex oracle	0.4479	0.3905	0.8384

Table 8 separates comparison by Reynolds number. T2-C reduces joint error relative to E2 Top-1 at all four tested parameter values, so the aggregate improvement is not confined to a single parameter.

In S–H, E2 selects Hopf at both Reynolds numbers, while T2-C assigns approximately 0.097 mean weight to Steady and retains a predominantly Hopf prediction. In H–P, E2 switches from Hopf to Periodic as the parameter changes; the mean Periodic fusion weight increases from 0.5870 to 0.7094.

Table 8: Parameter-wise centered-square routing and T2-C results at $K = 24$.

Overlap	Re	E2 choice	$\bar{\alpha}$	E2 E_{joint}	T2-C E_{joint}	Reduction
S–H	95.1	Hopf	0.0967	1.7140	1.1827	31.00%
S–H	95.3	Hopf	0.0971	1.7456	1.2090	30.74%
H–P	100.5	Hopf	0.5870	2.6485	0.8571	67.64%
H–P	102.0	Periodic	0.7094	1.0462	0.8360	20.09%

History-conditioned ablation. To isolate the contribution of physical-history information, we compare full T2-C with an architecture-matched history-ablated control. The control retains the parameter and E2-preference inputs but sets all standardized history-descriptor channels to zero. Figure 2 summarizes the full-window joint error over all $K = 24$ prediction steps and the 16 held-out windows at each interface. On S–H, full T2-C achieves $1.0954 \pm 0.0000\%$, compared with $1.8147 \pm 0.0002\%$ for the history-ablated control. On H–P, the corresponding errors are $0.8123 \pm 0.0002\%$ and $0.9769 \pm 0.0002\%$. Full T2-C has lower error in every paired-seed comparison, reducing the mean joint error by approximately 39.64% and 16.85% on S–H and H–P, respectively. These results support the contribution of history information on the evaluated overlap windows beyond the information supplied by the parameter and E2 preferences.

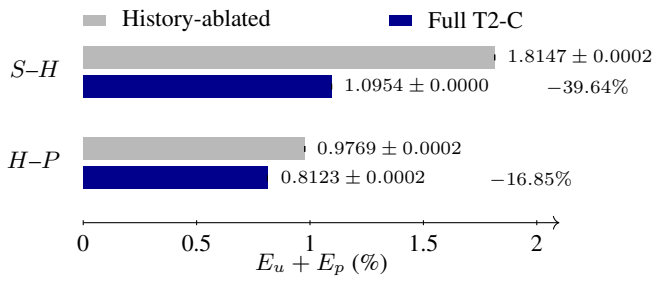


Figure 2: Effect of history conditioning on Square fusion.

F.2 Representative field reconstructions

The following snapshots complement the aggregate routing and fusion results with spatially resolved comparisons. All reference fields are taken from the true high-fidelity CFD solutions.

Pressure reconstruction at the H–P interface. Figure 3 compares E2 Top-1 and T2-C on the same frozen specialist trajectories. In both displayed windows, E2 selects the Hopf specialist, whereas T2-C combines the Hopf and Periodic reconstructions using a window-fixed weight. The full-domain, quadrature-weighted relative pressure error at the displayed terminal step decreases from 9.9697% to 0.5382% for Square and from 0.3444% to 0.0924% for Pinball.

Velocity reconstructions across geometries. Figure 4 provides a complementary overview of velocity reconstruction in all three geometries. The predictions reproduce the principal low-speed wakes, while the vector-error maps reveal spatial discrepancies that are less apparent from speed alone.

F.3 Output-space fusion: conditional algebraic identities

We first derive how fixed-weight fusion affects constraints, residuals, and energy, then compare the specialists and their blend using common diagnostic operators in Appendix F.4. The identities separate inherited specialist properties from additional fusion terms; their empirical consequences must

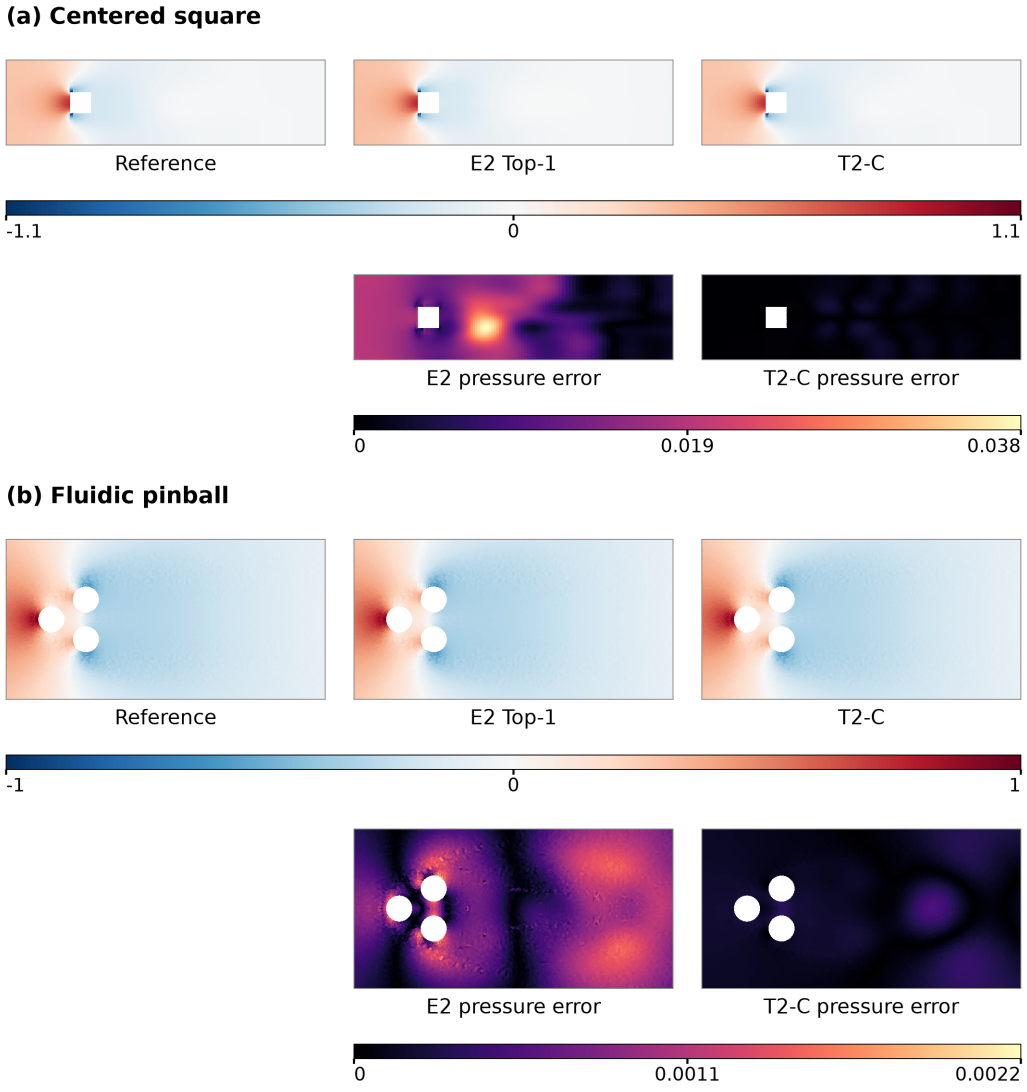


Figure 3: Representative H–P pressure reconstructions for (a) centered square and (b) fluidic pinball. In each case, the upper row shows, from left to right, the high-fidelity CFD reference, E2 Top-1 prediction, and T2-C prediction; the lower row shows the corresponding E2 and T2-C pointwise absolute pressure errors in the middle and right columns. The Square case is set $\mathrm{Re} = 100$, $K = 24$, and $t = 104$ (rollout duration 96), while Pinball at $\mathrm{Re} = 22$, $K = 56$, and $t = 50$ (duration 14). The Periodic specialist weights are 0.9791 and 0.9377, respectively.

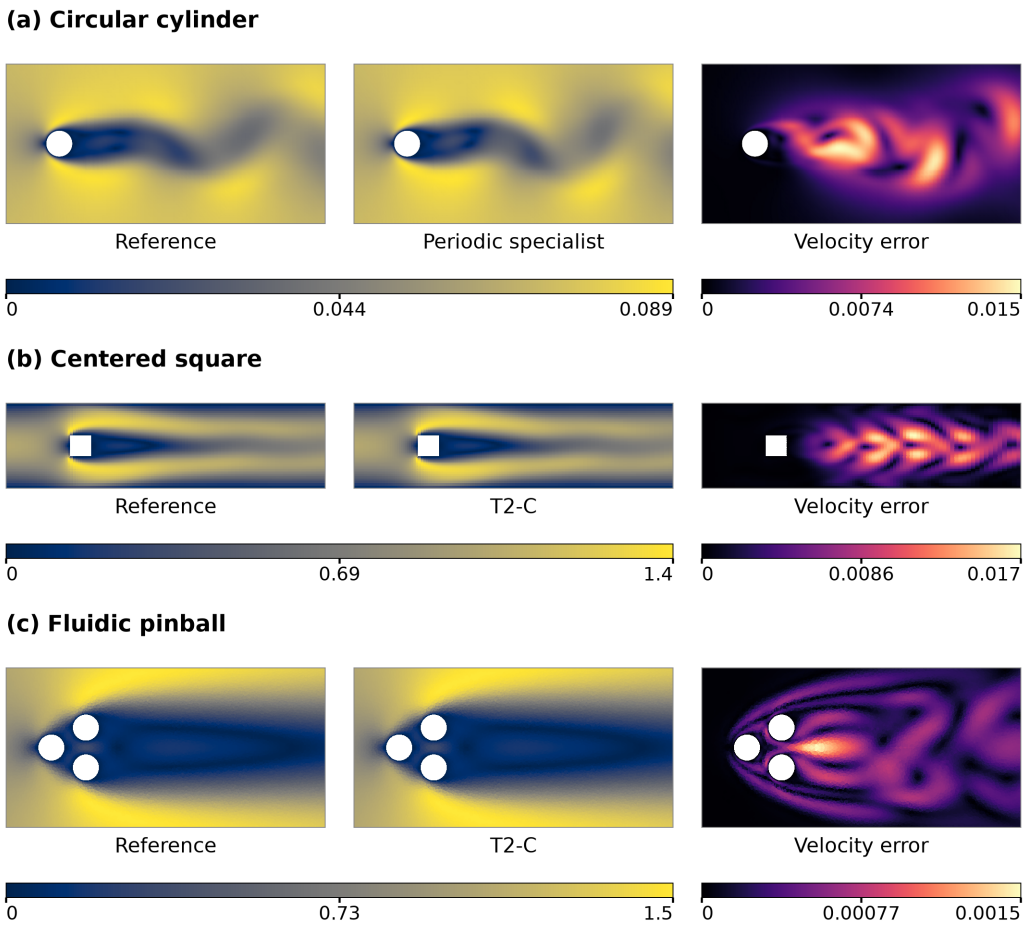


Figure 4: Representative terminal velocity reconstructions for (a) circular cylinder, (b) centered square, and (c) fluidic pinball. Columns show, from left to right, the high-fidelity CFD reference speed $\|\mathbf{u}_{\mathrm{ref}}\|_2$, predicted speed $\|\widehat{\mathbf{u}}\|_2$, and pointwise velocity-vector error $\|\widehat{\mathbf{u}} - \mathbf{u}_{\mathrm{ref}}\|_2$. Circular case is set at $\mathrm{Re} = 70.3$, $K = 48$, and $t = 1983$ (rollout duration 598). Square and Pinball use the same T2-C configurations and terminal snapshots as Figure 3.

be evaluated rather than assumed. Let $\boldsymbol{u}_\alpha = \alpha \boldsymbol{u}_1 + \beta \boldsymbol{u}_2$ and $p_\alpha = \alpha p_1 + \beta p_2$, where $\beta = 1 - \alpha$, $\alpha \in [0, 1]$ is constant in space and time within each prediction window. Here, $\boldsymbol{u}_1, \boldsymbol{u}_2$ and p_1, p_2 are the velocity and pressure fields obtained from the rollout predictions of the two adjacent specialists. For a common linear operator D_h, $D_h \boldsymbol{u}_\alpha = \alpha D_h \boldsymbol{u}_1 + \beta D_h \boldsymbol{u}_2$. Consequently, zero discrete divergence, a zero-mean pressure gauge, and identical affine boundary constraints are preserved if both specialists satisfy them.

Nonlinear residuals. For a fixed bilinear discretization N_h, define $\delta \boldsymbol{u} = \boldsymbol{u}_1 - \boldsymbol{u}_2$. Then

$$N_h(\boldsymbol{u}_\alpha, \boldsymbol{u}_\alpha) = \alpha N_h(\boldsymbol{u}_1, \boldsymbol{u}_1) + \beta N_h(\boldsymbol{u}_2, \boldsymbol{u}_2) - \alpha\beta N_h(\delta \boldsymbol{u}, \delta \boldsymbol{u}). \tag{53}$$

Using the sign convention $R_u(\boldsymbol{u}, p) = \partial_t \boldsymbol{u} - \boldsymbol{f}_h - L_h \boldsymbol{u} + N_h(\boldsymbol{u}, \boldsymbol{u}) + G_h p$, this yields

$$R_u(\boldsymbol{u}_\alpha, p_\alpha) = \alpha R_u(\boldsymbol{u}_1, p_1) + \beta R_u(\boldsymbol{u}_2, p_2) - \alpha\beta N_h(\delta \boldsymbol{u}, \delta \boldsymbol{u}). \tag{54}$$

Similarly, for $R_p(\boldsymbol{u}, p) = A_h p - [\boldsymbol{s}_0 + S_1 \boldsymbol{u} + B_p(\boldsymbol{u}, \boldsymbol{u})]$ with fixed bilinear B_p,

$$R_p(\boldsymbol{u}_\alpha, p_\alpha) = \alpha R_p(\boldsymbol{u}_1, p_1) + \beta R_p(\boldsymbol{u}_2, p_2) + \alpha\beta B_p(\delta \boldsymbol{u}, \delta \boldsymbol{u}). \tag{55}$$

Kinetic energy. For $E(\boldsymbol{u}) = \frac{1}{2}\|\boldsymbol{u}\|_M^2$ with a common positive quadrature matrix M,

$$E(\boldsymbol{u}_\alpha) = \alpha E(\boldsymbol{u}_1) + \beta E(\boldsymbol{u}_2) - \frac{\alpha\beta}{2}\|\boldsymbol{u}_1 - \boldsymbol{u}_2\|_M^2. \tag{56}$$

Thus, disagreement among the predicted velocities of different specialists reduces the blend's energy relative to the weighted specialist energies, potentially through phase cancellation. It does not imply lower energy than both specialists or improved agreement with the reference.

F.4 Physical-field diagnostics on flow past a square

We evaluate specialists, their frozen T2-C blend, and a common high-fidelity reference on the same 16 held-out windows per Square interface used for fusion evaluation ($K = 24$). The specialists are Steady/Hopf for S–H and Periodic/Hopf for H–P. Each interface contains two parameter values with eight overlapping windows each.

On the common 9,400-cell grid, quadratic local least-squares fits define D_x, D_y, and the Laplacian L. The initial stencil contains 24 nearest cells with independent coordinate scaling; it is expanded only if the polynomial matrix is rank-deficient or its condition number exceeds 10^6. A second version starts from 36 neighbors to test sensitivity; actual stencils contain at most 72 cells. Both versions differentiate a manufactured quadratic polynomial with maximum absolute error below 1.4×10^{-10}. We compute

$$\widehat{R}_u = \delta_t \boldsymbol{u} + N_D(\boldsymbol{u}, \boldsymbol{u}) + \nabla_D p - \mathrm{Re}^{-1} L \boldsymbol{u}, \tag{57}$$

$$\widehat{R}_p = L p + D_x N_{D,x} + D_y N_{D,y}, \tag{58}$$

where $N_D(\boldsymbol{u}, \boldsymbol{u}) = u D_x \boldsymbol{u} + v D_y \boldsymbol{u}$ and $N_{D,x}, N_{D,y}$ are its two components. Centered temporal differences and summary statistics use output steps 2–23.

Residuals are summarized by area-weighted RMS over the interior wake $6 \le x \le 16$, $1 \le y \le 3$ (2,627 cells). For a scalar or vector residual r, this is

$$\|r\|_{\mathrm{RMS},\Omega_w} = \left(\frac{\sum_{i \in \Omega_w} w_i \|r_i\|_2^2}{\sum_{i \in \Omega_w} w_i} \right)^{1/2}. \tag{59}$$

Kinetic energy uses quadrature over the whole domain. We distinguish the energy error from the specialist-relative fusion deficit:

$$\varepsilon_E = 100 \frac{|E(\boldsymbol{u}) - E_{\mathrm{ref}}|}{E_{\mathrm{ref}}}, \tag{60}$$

$$d_E = 100 \frac{\alpha E(\boldsymbol{u}_1) + \beta E(\boldsymbol{u}_2) - E(\boldsymbol{u}_\alpha)}{E_{\mathrm{ref}}}, \tag{61}$$

where $E_{\mathrm{ref}} = E(\boldsymbol{u}_{\mathrm{ref}})$.

Table 9: Square physical-field diagnostics with the 24-neighbor initial stencil. Energy error is evaluated over the whole domain; residuals are interior area-weighted RMS. Entries average output steps 2–23 and the evaluation windows as described above.

Interface	Field	Energy error (%)	$\|\widehat{R}_u\|$	$\|\widehat{R}_p\|$
S–H	Reference	0	0.0046	0.0340
	Steady	2.8110	0.0053	0.0367
	Hopf	0.1430	0.0061	0.0351
	T2-C	0.1764	0.0052	0.0342
H–P	Reference	0	0.1149	0.0374
	Periodic	0.0916	0.1163	0.0379
	Hopf	0.1730	0.1119	0.0405
	T2-C	0.0597	0.1141	0.0363

Measured comparisons and interpretation. Table 9 shows different outcomes for different diagnostics. On S–H, T2-C has smaller momentum and pressure residuals than either specialist, but Hopf has a smaller energy error. On H–P, T2-C has the smallest pressure residual and energy error among the predictions, whereas Hopf has the smallest momentum residual. The pressure-residual ordering also holds with the 36-neighbor initial stencil, although changing the stencil changes residual magnitudes substantially. In particular, a predicted field can have a lower diagnostic residual than the reference without being more accurate.

Additional effects of fusion. Figure 5 isolates fusion-induced changes from the total errors in Table 9. Besides d_E, we report the RMS of the additional residual fields

$$\Delta\widehat{R}_q = \widehat{R}_q(\boldsymbol{u}_\alpha, p_\alpha) - \alpha\widehat{R}_q(\boldsymbol{u}_1, p_1) - \beta\widehat{R}_q(\boldsymbol{u}_2, p_2), \qquad q \in \{u, p\}. \tag{62}$$

The norm is taken after subtracting the residual fields; it is not a difference between residual norms.

The mean normalized energy deficit d_E is 0.0031% for S–H and 0.0018% for H–P over steps 2–23. Thus the average energy reduction relative to the weighted specialist energies is small on these windows; this is not a bound on individual-window deviations or the error against reference energy. The curves show that the additional residuals are nonzero and time-dependent: S–H exhibits an early peak and a secondary rise, whereas H–P shows a later increase. Their magnitudes alone do not determine whether the total fused residual improves, because the additional fields interact with the specialists' existing residuals. These results characterize the finite evaluation windows, not long-time stability or strict conservation.

G Internal expert routing: protocols and supplementary results

G.1 Circular: internal expert utilization

This study characterizes correction-expert selection and weight allocation inside individual regime-local specialists, rather than E2 selection between specialists. We analyze frozen original Circular Hopf and Periodic specialists. Periodic is evaluated on 13 fixed windows at each of four held-out Reynolds numbers, giving 52 windows with $K = 48$. Hopf uses five windows at each of three held-out Reynolds numbers, giving 15 windows with $K = 56$. We record one routing decision per output step at the first integration stage, giving 2496 Periodic and 840 Hopf records per channel, rather than counting every intermediate right-hand-side evaluation. The statistics pool these records; overlapping windows are not treated as independent trajectories. No routing diagnostic is used for checkpoint selection.

Selection frequency and mixing weight. For each channel, let $I_{n,e}$ indicate whether routed expert e is selected at record n, including group identity in the expert index. For N records, define

$$f_e = \frac{1}{2N}\sum_{n=1}^{N} I_{n,e}, \qquad \sum_e f_e = 1, \tag{63}$$

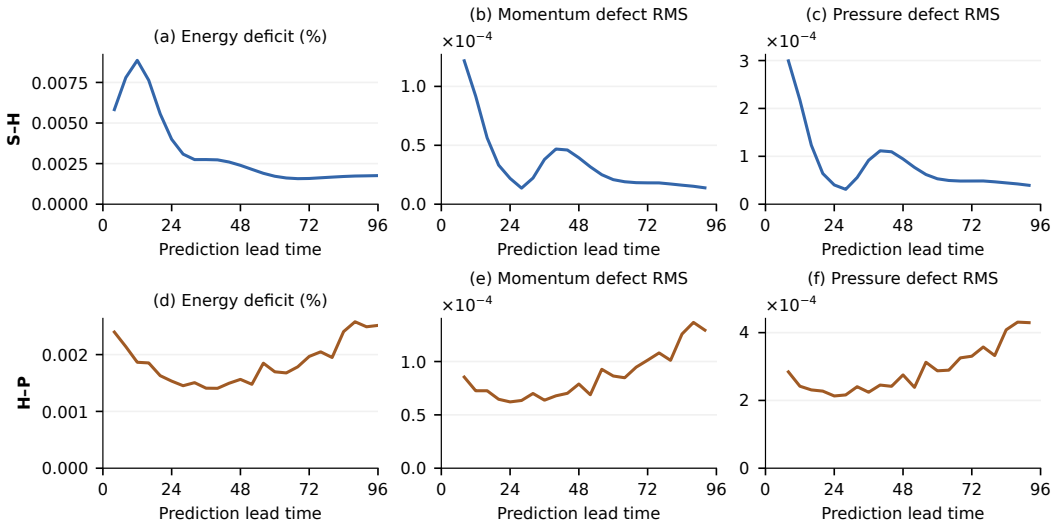


Figure 5: Additional effects of Square T2-C fusion with the 24-neighbor initial stencil. Rows correspond to S–H (top) and H–P (bottom); columns show normalized energy deficit d_E in percent, momentum-defect RMS, and pressure-defect RMS. Each curve averages all 16 windows at each prediction lead time. Energy uses steps 1–24; residuals use steps 2–23. Residual RMS is computed over the interior wake before window averaging.

where the factor two accounts for Top-2 activation at each record. The mean routing mass is

$$m_e = \frac{1}{N} \sum_{n=1}^{N} \omega_{n,e} I_{n,e}, \tag{64}$$

where $\omega_{n,e}$ is the final mixing coefficient, including the scaling of the routed component. Unselected experts retain zero frequency and mass. For these two checkpoints, the shared component has coefficient $4/7$ and the routed Top-2 component has total coefficient $3/7$, so $\sum_e m_e = 3/7$. The shared expert is excluded from the routed-expert counts below. Routing mass quantifies weight allocation. Identifiers such as `g0:e0` are local routed-expert labels and do not denote the shared expert.

Selection diversity. With N_E routed experts available to a channel, define

$$H = -\frac{\sum_{e:f_e>0} f_e \log f_e}{\log N_E}, \qquad 0 \le H \le 1. \tag{65}$$

This entropy measures dispersion of activation frequencies. We also count distinct unordered Top-2 pairs, retaining group identity, and the fraction of records assigned to the most frequent pair. An expert is counted as active if selected at least once in the evaluated records.

Table 10: Post-hoc Circular internal-routing statistics. Dominant-pair share is the fraction of records using the most frequent unordered pair, and H is activation-frequency entropy.

Specialist	Channel	Active experts	Distinct Top-2 pairs	Dominant-pair share (%)	H
Hopf	Velocity	2/6	1	100.00	0.387
	Pressure	2/6	1	100.00	0.387
Periodic	Velocity	18/18	33	16.27	0.929
	Pressure	14/18	16	31.25	0.784

Table 10 shows that Periodic specialist activates all available velocity experts and most pressure experts. Velocity selection is more dispersed than pressure selection, with more distinct pairs, a smaller dominant-pair share, and higher utilization entropy. Figure 6 separates group frequencies

from routed-expert masses across the four held-out Reynolds numbers. The group-selection frequencies vary across parameters, while the channel-specific masses reveal how the routed weight is distributed.

The evaluated Hopf specialist uses a single expert group. Within that group, the observed velocity pair is (e_1, e_3) and the pressure pair is (e_2, e_4) at every recorded step. Velocity expert e_3 receives approximately 99.93% of the routed mass, and pressure expert e_2 receives 84.16%. These percentages are relative to the routed component, not the complete correction including the shared expert. The secondary pressure expert e_4 has a mean mass that increases across the three evaluated Reynolds numbers, so fixed support does not imply constant weight allocation. For a fixed pair, both activation frequencies equal $1/2$ and $H = \log 2/\log 6 \simeq 0.387$, regardless of how uneven their weights are. Thus the entropy in the table does not indicate pair switching or balanced expert contributions.

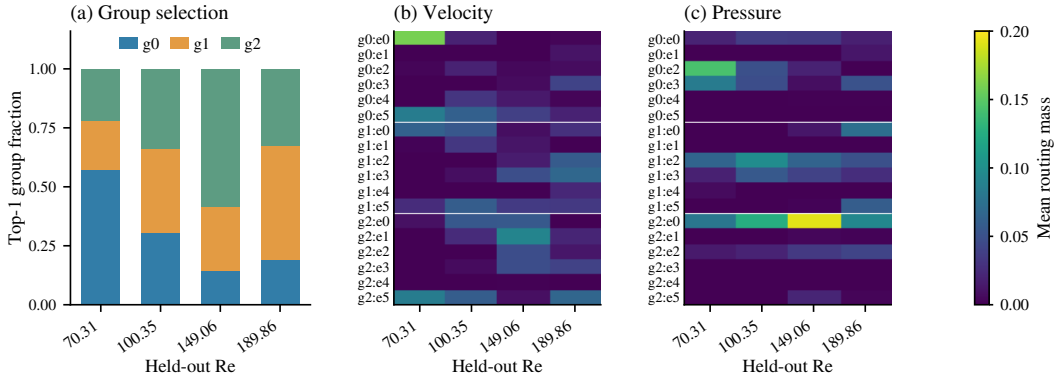


Figure 6: Parameter-dependent internal routing of the frozen Circular Periodic specialist. (a) Top-1 group-selection frequencies. (b,c) Mean routed-expert mass for velocity and pressure, respectively. Records are collected at the first integration stage of each output step over 13 windows per Reynolds number. Expert labels are local to each group and channel; the shared component is not shown.

H Computational cost and timing boundaries

The computational cost of a sparse reduced model depends on both the parameters selected by its routers and the implementation used to advance and reconstruct the flow. We therefore examine parameter utilization and measured prediction time separately. The Circular H/P specialists provide the parameter comparison, while Circular Periodic specialist is used to compare local prediction costs and to place the measured ROM time in the context of a historical full-order simulation.

H.1 Parameter utilization

Table 11 compares the total trainable capacity of each specialist with the parameters active in one prediction-path routing decision. The active count includes the encoder and refinement blocks, executed routers, the selected group's shared experts, channel-specific Top-2 experts, and learned pressure correction. Each trainable parameter is counted once.

Table 11: Total trainable capacity and per-decision active parameters of the Circular specialists.

Specialist	Total trainable	Active prediction	Active fraction
Hopf	31,810,392	14,165,816	44.53%
Periodic	48,617,547	8,308,959	17.09%

The Periodic specialist has more total trainable parameters than Hopf but uses fewer in each analyzed decision. This illustrates the distinction between available model capacity and the capacity selected for an individual prediction. In particular, the Dense control is matched to the active neural parameter

Table 12: Circular Periodic specialist local prediction cost at $K = 48$ on flow past a Circular.

Model	Median time (s)	GPU peak (bytes)	Time / Dense
Proposed	2.1848	204,238,336	3.32
Dense	0.6579	43,111,936	1.00
Structured	0.8652	43,316,736	1.32

budget, not to the full sparse model's parameter count. A smaller active fraction therefore does not, by itself, imply less computation than this control.

H.2 Local prediction time

To test whether sparse parameter use translates into lower latency, we compare the native Circular Periodic specialist implementation with Dense and Structured model, batch size one and $K = 48$ rollout. Table 12 reports median times from ten repetitions following three warm-up runs.

The sparse implementation takes 3.32 times as long as Dense and also has higher peak allocated GPU memory; Structured model is intermediate in latency. Results show that each rollout evaluates the model 192 times. The proposed method genuinely skips forward passes for unselected experts: each evaluation performs forward passes for one shared and two routed experts per channel. Nevertheless, this produces 1,152 expert forward passes per rollout, compared with 384 non-routed expert forward passes for Structured model and 192 joint correction-network forward passes for Dense model.

H.3 Comparison with a full-order simulation

The Dense model comparison measures the computational cost of the local architecture, whereas the CFD comparison assesses reduced prediction relative to full-order simulation. For the Circular Periodic case at $\mathrm{Re} = 70.3$, a $K = 48$ rollout spanning 598.7009 physical time units takes a median of 2.1875 s on an RTX 4090 with a Xeon Platinum 8358P host. This timing includes physical-history encoding and history-dependent right-hand-side construction, extending the scope of Table 12. The corresponding historical OpenFOAM interval requires $1,996$ s over 4,692 solver steps, yielding a wall-time ratio of approximately 912.5. This comparison indicates substantially lower online prediction cost for the RAL-MOE-ROM.