LEARNING PDE TANGENT OPERATORS FROM STOCHASTIC WALKS

Anonymous authors
Paper under double-blind review

ABSTRACT

Gradient-based optimization for PDE-constrained inverse problems and inverse design requires accurate solution sensitivities. These sensitivities can be approximated by differentiating a learned solution surrogate, but accurate solution predictions do not guarantee accurate derivatives. Learning sensitivities directly offers an alternative, although obtaining high-accuracy derivative supervision can be costly. We introduce the Context-Aware Linearity-preserving Tangent Operator (CALTO), which learns reusable tangent operators from sparse, noisy directional observations generated with few stochastic paths. Its nonlinear context encoder modulates both sides of linear kernels, with a bias-free readout. This enforces superposition consistent with shared-path observations, jointly constraining one linear operator per base problem. Across four 2D elliptic benchmarks and one 3D CAD diffusion–reaction benchmark, CALTO achieves 0.71–8.88% context-averaged relative L_2 error on held-out in-distribution instances, reducing error by 50.7–74.8% against the strongest learned baseline per benchmark. Comparisons with a nonlinear variant support directional linearity under sparse, noisy supervision; controlled coefficient identification further shows that CALTO tangents improve parameter recovery and reduce final residual ratios by 98.9–99.5% when correcting learned primal surrogates.

1 INTRODUCTION

PDE-constrained inverse problems and inverse design seek model parameters ϑ whose associated solution field $u = \mathcal{G}(\vartheta)$ matches observations or prescribed physical targets. Applications include parameter identification (O'Leary-Roseberry et al., 2024), control of physical systems (Holl et al., 2020), and material optimization (Xue et al., 2023). A common approach for seeking ϑ is to minimize a mismatch loss of u using gradient-based optimization (Giles & Pierce, 2000). The required parameter gradients depend on the tangent operator $J_\vartheta = D\mathcal{G}(\vartheta)$ of the differentiable solution map, whose action $J_\vartheta v$ gives the first-order change in the solution field induced by a parameter perturbation v.

Neural operators such as FNO and DeepONet approximate $\mathcal{G}$ and can obtain tangent responses as Jacobian–vector products (JVPs) through automatic differentiation (Li et al., 2021; Lu et al., 2021). Accurate solution prediction, however, does not guarantee accurate parameter sensitivities (Behroozi et al., 2026). Derivative-informed variants mitigate this discrepancy by additionally supervising Jacobians or selected derivative information (O'Leary-Roseberry et al., 2024; Qiu et al., 2024; Behroozi et al., 2025; Yao et al., 2026). Their tangent responses remain derivatives of a learned state map, coupling sensitivity approximation to state reconstruction. We instead learn tangent responses directly, separating sensitivity approximation from state-map fitting.

We formulate this task as learning the tangent responses $(\vartheta, v) \mapsto J_\vartheta v$. Directional observations across reference configurations and perturbations supervise selected Jacobian actions without requiring the entire matrix (Qiu et al., 2024). Generating these observations with finite element methods typically involves global state and linearized PDE solves; repeating these computations across many configurations can make large-scale supervision costly. Stochastic differentiation offers an alternative by estimating solution derivatives through Monte Carlo sampling. Differential Walk on Spheres (DWoS), for example, uses random walks to estimate derivatives with respect to physical parameters and domain geometry at queried locations without a global volumetric solve (Miller et al., 2024).

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.

Small path budgets trade sampling accuracy for lower cost, enabling broader coverage of configurations and directions within a fixed budget. The challenge is to learn an accurate, reusable tangent operator from these inexpensive, noisy observations.

We address this challenge by encoding the tangent operator's linear dependence on v while learning its generally nonlinear dependence on ϑ. Inspired by Neural Green's Operators, which retain linear action on inhomogeneity fields while learning their dependence on PDE coefficients (Melchers et al., 2026), we separate nonlinear operator conditioning from linear response propagation. We introduce the Context-Aware Linearity-preserving Tangent Operator (CALTO), which maps physical perturbation features $r = \mathcal{C}_{c,\mathcal{P}} v$ to tangent responses, conditioned on the base PDE context $c = c(\vartheta)$ and query/support point configuration $\mathcal{P}$. Its nonlinear context branch generates attention weights and multiplicative gates solely from base-PDE and geometric features. These gates modulate both sides of linear kernels for perturbation aggregation and response propagation, followed by a shared bias-free linear readout at interior and boundary queries. For fixed context, point configuration, and query grouping, all directional observations jointly constrain a single learned linear operator. This structural prior focuses the noisy directional observations on learning the context-dependent tangent operator. Our shared-path labels preserve the same directional relations, making this constraint compatible with the stochastic supervision.

We evaluate CALTO across four two-dimensional elliptic families and a three-dimensional CAD diffusion–reaction benchmark, reducing in-distribution tangent-prediction error by 50.7–74.8% relative to the strongest learned baseline per benchmark. Further studies show that directional linearity improves robustness to sparse, noisy supervision and enables recovery of full-parameter Jacobians and VJPs from limited directional observations. Learned tangents also substantially improve coefficient identification, both when replacing surrogate-derived derivatives and when correcting the local derivatives of primal surrogates demonstrating the utility of learned tangent operators for PDE-constrained inverse problems.

Our contributions are threefold. **(i)** We formulate direct learning of the tangent responses from few-path stochastic directional observations, focusing supervision on solution sensitivities. **(ii)** We introduce CALTO, which combines nonlinear PDE conditioning with a linear perturbation pathway and context-dependent modulation on both sides of propagation kernels, preserving directional linearity by construction. **(iii)** We evaluate tangent-response accuracy and generalization on four two-dimensional PDE families and one three-dimensional CAD benchmark, with controlled two-dimensional studies of supervision, architecture, derivative recovery, and coefficient identification.

2 Related Work

Stochastic PDE supervision. Grid-free Monte Carlo solvers provide local PDE estimates without volumetric discretization (Sawhney & Crane, 2020). Walk on Stars handles mixed boundary conditions (Sawhney et al., 2023), while DWoS estimates parameter and shape sensitivities (Miller et al., 2024). A related differential Monte Carlo method uses a boundary-integral formulation to estimate parameter and shape derivatives for Dirichlet Poisson problems (Yu et al., 2024). Grid-free derivative estimators have also been used for inverse PDE problems, including reverse-mode differentiation of branching random walks (Yilmazer et al., 2024). Neural Walk-on-Spheres and WoS-NO further use stochastic estimates as supervision for learned PDE solvers (Nam et al., 2024; Viswanath et al., 2026). These two methods primarily amortize state solutions. CALTO instead learns conditional tangent actions from fixed offline directional observations, with shared random realizations coupling multiple directions through the same random linear operator.

Derivative-informed operator learning. Sobolev training augments function-value fitting with supervision of input derivatives (Czarnecki et al., 2017). Derivative-informed neural operators augment state fitting with sensitivity supervision, using reduced derivative representations or sampled Jacobian information (O'Leary-Roseberry et al., 2024; Qiu et al., 2024; Behroozi et al., 2025; 2026; Yao et al., 2026). Extensions also address shape sensitivities and varying geometries (Gong et al., 2026). Their primary learned object remains a state operator whose derivatives are supervised. Theoretical work also studies recovery of certain compact operators from forward actions without adjoint queries (Boullé et al., 2024). Relatedly, Neural Green's operators condition linear solution operators on PDE parameters or geometry (Melchers et al., 2026; Yoo et al., 2025); CALTO applies this condition–action separation to the local tangent operator.

3 PROBLEM FORMULATION

PDE and tangent response. At physical parameters $\vartheta_0 \in \mathbb{R}^d$, let $u_0 = \mathcal{G}(\vartheta_0)$ solve $\mathcal{F}(u_0, \vartheta_0) = 0$ on a base domain Ω; $\mathcal{F}$ includes the interior equation and boundary conditions. The context c collects the base geometry, coefficients, source, boundary types and data, and any supplied base-state approximation $\widetilde{u}_0$. For a physical parameter or shape direction $v \in \mathbb{R}^d$, $\delta u = D\mathcal{G}(\vartheta_0)[v]$ is the first-order solution variation. With all derivatives evaluated at the exact base state (u_0, ϑ_0), define $L_c = D_u\mathcal{F}$ and $\mathcal{B}_c = -D_\vartheta\mathcal{F}$. Assuming Fréchet differentiability and a bounded inverse for L_c,

$$L_c\delta u = \mathcal{B}_c v, \qquad J_{c,X}v = P_X\delta u = P_X L_c^{-1}\mathcal{B}_c v, \tag{1}$$

where $X = (x_1, \ldots, x_Q)$ contains interior or boundary queries and $P_X w = (w(x_1), \ldots, w(x_Q))^\top$ is assumed well defined. Thus $J_{c,X} : \mathbb{R}^d \to \mathbb{R}^Q$ is linear in v.

Physical perturbations. A parameterized PDE input field $h(x; \vartheta)$ induces $\delta h(x) = D_\vartheta h(x; \vartheta_0)[v]$. For screened Poisson, $(-\Delta + \lambda)u_0 = f$, where f is the source and λ the screening coefficient, differentiation gives

$$\begin{aligned} (-\Delta + \lambda)\delta u &= \delta f - u_0\delta\lambda \quad \text{in } \Omega, \\ \delta u &= \delta g_D \quad \text{on } \Gamma_D, \\ \partial_n\delta u &= \delta g_N \quad \text{on } \Gamma_N. \end{aligned} \tag{2}$$

Here Γ_D, Γ_N form the fixed boundary partition, n is the outward normal, and g_D, g_N are the prescribed base Dirichlet and Neumann data. For shape perturbations of a homogeneous Dirichlet boundary, the Eulerian response on the base domain instead has trace $\delta u = \delta g_D - V_n\partial_n u_0$, with normal velocity V_n linear in v. Let $\mathcal{P} = (X, X_\Omega, X_\partial)$ collect query, interior-support, and boundary-support points. Evaluating these physical variations on $\mathcal{P}$ gives the linear input representation $r = \mathcal{C}_{c,\mathcal{P}}v$; $\mathcal{B}_c v$ denotes the induced tangent-equation data. PDE-specific equations and input channels are given in Appendices A and C.

Supervision and learning objective. For each context, $V = [v_1, \ldots, v_R]$ collects the observed directions and $Y = \widehat{J}_W V \in \mathbb{R}^{Q\times R}$ their stochastic responses, where $\widehat{J}_W$ averages W complete random realizations shared across directions. Finite paths introduce sampling noise; approximate base fields and numerical stopping can also bias the labels. Only $\mathrm{span}(V)$ is directly observed. We seek $F_\theta(c, v; \mathcal{P}) \approx J_{c,X}v$, with network weights θ, that is linear in v for fixed c, point sets and query grouping (queries evaluated jointly). We abbreviate $J_{c,X}$ as J_c, and $F_\theta(c, v; \mathcal{P})$ as $F_\theta(c, v; X)$, when the omitted configuration is fixed. The benchmarks comprise four 2D families—Dirichlet and mixed-boundary screened Poisson (SP-D/SP-DN), variable-coefficient diffusion–reaction (VC-DR), and semilinear reaction–diffusion (SL-RD)—plus 3D variable-coefficient diffusion–reaction on ABC CAD geometries (CAD-DR) (Koch et al., 2019). Numerical targets and bias sources are detailed in Appendix A.3.

4 CALTO: LEARNING CONDITIONAL TANGENT OPERATORS

CALTO implements the predictor $F_\theta(c, v; \mathcal{P})$ of Section 3 through two branches: a nonlinear context branch generates attention weights and multiplicative gates, while a linear response trunk maps physical perturbation features to query responses. This separation allows the learned operator to adapt to the base PDE while preserving its linear action on v. Figure 1 summarizes the architecture. We train a separate model for each PDE family, using the same architecture with family-specific input channels in 2D and 3D.

4.1 CONTEXT-CONDITIONED LINEAR RESPONSE

Input representation. Using the point sets $\mathcal{P}$ and representation $\mathcal{C}_{c,\mathcal{P}}$ defined in Section 3, we write $r = (r_\mathrm{q}, r_\Omega, r_\partial) = \mathcal{C}_{c,\mathcal{P}}v$. These physical perturbation features accompany the corresponding pointwise contexts $(c_\mathrm{q}, c_\Omega, c_\partial)$. We apply no nonlinear transformation or magnitude normalization to r; shape velocities are evaluated on the fixed base boundary. For SL-RD, the context also includes a frozen coarse state $\widetilde{u}_0$ and the linearized potential $\lambda + 3\gamma\widetilde{u}_0^2$, with reaction coefficient γ. Appendix C lists the family-specific channels.

Context branch. A shared nonlinear pointwise encoder processes query and interior-support contexts; a separate encoder processes boundary-support contexts. The Attention and Gate Generator

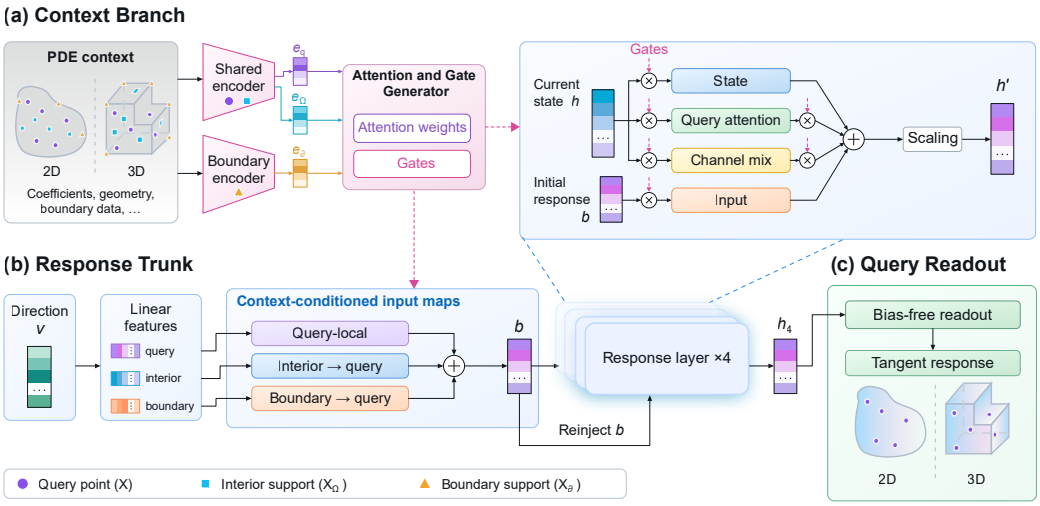

Figure 1: Overview of CALTO. (a) A shared encoder produces encoded query and interior-support contexts e_q, e_Ω; a separate encoder produces boundary-support contexts e_∂. The Attention and Gate Generator produces attention weights and elementwise gates from context alone (dashed arrows). (b) The response trunk combines perturbation features from all three point sets into an initial query response b, then applies four layers that reinject b. Gates act before and after attention and channel mixing. (c) A shared bias-free linear readout predicts the physical tangent response δu.

uses query/key projections and softmax to compute attention weights, and affine projections and sigmoid to generate channelwise gates. Attention combines responses across points, while each gate scales response channels by context-dependent coefficients. These quantities depend only on context and point sets and are reused across directions. The resulting predictor has the form

$$F_\theta(c, v; \mathcal{P}) = \mathcal{A}_\theta(c; \mathcal{P})\, \mathcal{C}_{c,\mathcal{P}} v, \tag{3}$$

where $\mathcal{A}_\theta(c; \mathcal{P})$ is the linear response map implemented by the trunk, readout, and context-derived weights.

Initial query response. Separate bias-free lifts map r_{q}, r_Ω, and r_∂ to 48 channels. Query features are gated locally. Each support branch gates its features, aggregates them at queries through attention, and gates the output using query context. The two support branches have separate attention and gate parameters. Writing these three linear maps as $\mathcal{M}_{\mathrm{q}}$, $\mathcal{M}_\Omega$, and $\mathcal{M}_\partial$, their outputs initialize the query state:

$$b = \mathcal{M}_{\mathrm{q}}(c) r_{\mathrm{q}} + \mathcal{M}_\Omega(c) r_\Omega + \mathcal{M}_\partial(c) r_\partial, \quad h_0 = b. \tag{4}$$

Response propagation and readout. Four response layers update the 48-channel query states. Each layer combines gated state retention, attention among queries in the same group, bias-free pointwise channel mixing, and gated reinjection of b. The four outputs are summed and scaled by $1/2$. Context-derived gates act on both sides of attention and channel mixing, modulating both the incoming and propagated responses. A shared bias-free linear readout maps the final state h_4 to one physical tangent response per query. Appendix C.2 gives the complete layer equations.

4.2 Directional Linearity

For fixed θ, c, $\mathcal{P}$, and query grouping, all attention weights and gates are independent of v. The representation $\mathcal{C}_{c,\mathcal{P}}$ is linear, and the response trunk uses only bias-free linear maps, weighted sums, and fixed gates, with no activation or normalization. Consequently,

$$F_\theta(c, \alpha v + \beta w; \mathcal{P}) = \alpha F_\theta(c, v; \mathcal{P}) + \beta F_\theta(c, w; \mathcal{P}) \tag{5}$$

for all directions v, w and scalars α, β. Proposition C.1 proves this property. This enforces consistent directional recombination; generalization beyond the observed span relies on structure learned across contexts. Changing the query grouping can change predictions, so groups and supports remain fixed across directions (Appendix C.3).

4.3 Stochastic Supervision and Training

We generate the directional labels of Section 3 using few-path stochastic estimators. The 2D constructions use coarse quadratic (P2) finite-element base states; shape responses additionally use differential Walk on Spheres (Miller et al., 2024). At each query, all directions share the complete random realization, including nested estimates, while different context–query pairs are sampled separately. Fixed-geometry Dirichlet queries use their known tangent boundary values. Appendices B.1 and A.3 detail the estimators and their numerical targets. The resulting labels $Y = \widehat{J}_W V$ satisfy

$$Y\alpha = \widehat{J}_W(V\alpha), \tag{6}$$

matching the predictor's directional linearity even at finite path budgets. Shared paths preserve these relations but do not remove sampling noise or numerical bias; observations constrain the operator only on $\mathrm{span}(V)$. Appendix B analyzes these errors and directional coverage.

We train on fixed offline labels using the context-averaged relative L_2 error (L2RE):

$$\mathcal{L}(\theta) = \frac{1}{N}\sum_{i=1}^{N} \frac{\|\mathbf{p}_i(\theta) - \widehat{\mathbf{y}}_i\|_2}{\|\widehat{\mathbf{y}}_i\|_2}. \tag{7}$$

Here N is the number of training contexts. For context c_i, $\mathbf{p}_i(\theta), \widehat{\mathbf{y}}_i \in \mathbb{R}^{Q_i R_i}$ stack predictions and stochastic labels over Q_i active queries and R_i directions. Independent high-accuracy responses select models and provide test references.

5 Experiments

Benchmarks. We learn $F_\theta(c, v; X) \approx J_{c,X} v$ from limited directional observations with few-path estimates, using separately trained models for four two-dimensional elliptic PDE families and **CAD-DR**, a three-dimensional variable-coefficient diffusion–reaction benchmark on CAD domains. **SP-D** and **SP-DN** solve the screened Poisson equation, $(-\Delta + \lambda)u = f$, under Dirichlet and mixed Dirichlet/Neumann (D/N) boundary conditions, respectively. Besides, SP-D uses prescribed perturbations of the outer boundary shape, and SP-DN varies the physical data while holding the D/N partition fixed during differentiation. **VC-DR** solves the variable-coefficient diffusion–reaction equation, $-\nabla \cdot (\kappa \nabla u) + qu = f$. **SL-RD** solves the semilinear reaction–diffusion equation, $-\Delta u + \lambda u + \gamma u^3 = f$, whose linearization depends on the state. Both impose zero Dirichlet boundary conditions. Across all benchmarks, the targets are solution variations, including those evaluated at Neumann queries.

Generalization. In-distribution (ID) evaluation uses unseen contexts drawn from the training distribution. Out-of-distribution (OOD) evaluation changes either the base PDE problem or the perturbation direction. Base-problem shifts change coefficients, forcing, boundary data or partitions, and geometry. Direction shifts include four-sparse and balanced Rademacher draws, amplitude changes, and equivalent reparameterizations. Appendices D.1 and D.2 provides the details of PDE definitions, splits, sampling ranges, and reference construction.

Supervision and baselines. For the two-dimensional benchmarks, CALTO uses fixed, offline weak JVP targets with $R = 8$ directions per training context and W4 sampling, defined as four stochastic walks per query. For each query, all directions share the complete random realization. We adapt all six learned baselines to the same weak-JVP prediction setting, while retaining their respective mechanisms for producing tangent responses. SC-FNO (Behroozi et al., 2025), DE-DeepONet (Qiu et al., 2024), and DINO (O'Leary-Roseberry et al., 2024) produce state-surrogate outputs, from which JVPs are obtained by directional differentiation; training supervises only these resulting JVPs with the weak targets, without primal-value supervision. GINO (Li et al., 2023), Transolver (Wu et al., 2024), and Transolver++ (Luo et al., 2025) are instead adapted to map the PDE context and perturbation direction directly to tangent responses. Thus, all six methods are trained solely from the same weak JVP targets. For the state-surrogate-based methods, directional differentiation is itself linear in the perturbation direction. FNO+AD (Li et al., 2021) is trained only on primal solution values and obtains JVPs by automatic differentiation with respect to the physical inputs. This baseline measures tangent prediction under the chosen primal-only training procedure. Raw W4 estimates serve as an unlearned reference. We report context-averaged L2RE over query–direction pairs and

split-pooled R^2. For CAD-DR, the point-set comparisons use W10 weak supervision and two physical directions. Appendices D.2 and D.3 give the CAD-DR protocol, learning settings, baseline details, and metric definitions.

5.1 Tangent Prediction and Generalization

Table 1 reports context-averaged L2RE and R^2 for 2D test-ID contexts. CALTO has the lowest L2RE in all four families. Direct tangent estimates averaged over four outer paths per stochastic query (Raw W4) have errors 9.2–21.4 times those of CALTO, showing the benefit of learning from fixed weak observations rather than using these estimates directly. On SP-D and SP-DN, CALTO achieves 0.71% and 3.03% L2RE, reducing error by 50.7% and 52.7% relative to the strongest learned baselines, Transolver and Transolver++, respectively.

Table 1: Two-dimensional test-ID tangent prediction.

Method	Supervision type	SP-D		SP-DN		VC-DR		SL-RD	
		L2RE (%)	R^2	L2RE (%)	R^2	L2RE (%)	R^2	L2RE (%)	R^2
CALTO	JVP-only	**0.71**	**1.000**	**3.03**	**0.999**	**8.88**	**0.993**	**8.30**	**0.994**
SC-FNO		5.08	0.998	14.52	0.982	30.96	0.914	26.27	0.933
DE-DeepONet		2.03	1.000	8.19	0.994	28.86	0.926	23.02	0.952
DINO		5.19	0.998	14.29	0.983	38.82	0.868	51.67	0.756
GINO		7.96	0.994	19.97	0.960	26.11	0.932	21.66	0.955
Transolver		1.44	1.000	6.46	0.994	25.33	0.944	22.61	0.954
Transolver++		1.88	1.000	6.41	0.995	26.23	0.940	23.26	0.952
FNO+AD	Primal-only	36.10	0.848	38.61	0.867	62.36	0.648	92.26	0.143
Raw W4	–	15.18	0.982	44.21	0.816	87.66	0.338	76.38	0.498

CALTO also improves prediction for coefficient and state-dependent tangents. In VC-DR, diffusivity perturbations enter through $\nabla\cdot(\delta\kappa\nabla u_0)$; L2RE is 8.88%, versus 25.33% for Transolver. In SL-RD, both the linearized potential and tangent forcing depend on the base state. With the coarse state supplied as context, CALTO achieves 8.30%, versus 21.66% for GINO. These reductions of 65.0% and 61.7% support direct tangent learning in these families. FNO+AD, trained instead on solution values, yields tangent L2REs of 36.10–92.26%, which shows this primal-only baseline does not produce accurate tangents under the tested settings. CALTO-3D uses W10 weak JVP targets on 1,000 ID CAD-DR contexts for two physical directions: source/boundary and coefficient; the latter also changes source and boundary data. CALTO-3D achieves 2.37% L2RE; GINO, the best of four baselines, has 9.39% (Table 2). Figure 2 evaluates out-of-distribution contexts and direc-

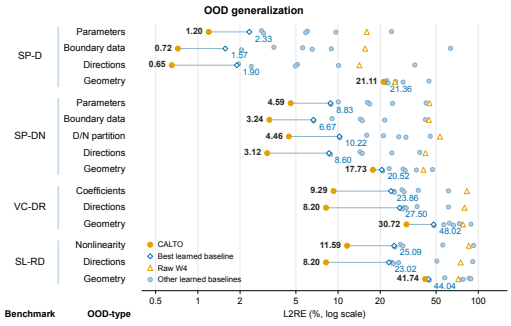

Figure 2: Two-dimensional OOD tangent prediction. Points compare CALTO, learned baselines, and Raw W4 using context-averaged L2RE (%).

Table 2: CAD-DR ID and OOD tangent prediction. L2RE is reported in percent.

Method	Metric	ID	Geometry OOD	Parameter OOD	Joint OOD
CALTO-3D	L2RE	**2.37**	**2.36**	**3.90**	**4.50**
	R^2	**0.999**	**0.999**	**0.998**	**0.997**
GINO	L2RE	9.39	10.12	9.72	10.53
	R^2	0.992	0.990	0.991	0.989
Transolver	L2RE	19.96	21.46	23.37	25.52
	R^2	0.963	0.958	0.947	0.936
Transolver++	L2RE	11.40	11.87	13.22	15.08
	R^2	0.987	0.985	0.981	0.975
FNO+AD	L2RE	35.96	38.25	37.89	38.96
	R^2	0.880	0.867	0.865	0.862

tions. On SP-D direction OOD, CALTO achieves 0.65% L2RE, versus 1.90% for Transolver, the strongest learned baseline. Across the four direction suites, CALTO reduces L2RE by 63.7–70.2% relative to the strongest learned baselines. Geometry shifts are harder: SP-D L2RE is 21.11%, close to DE-DeepONet at 21.36%; SL-RD reaches 41.74%, narrowing the improvement to 5.2%. On CAD-DR geometry, parameter, and joint OOD splits, CALTO has the lowest error in all comparisons, with L2REs of 2.36–4.50% and reductions of 57.2–76.7% relative to the strongest learned

baseline (Table 2). Together, the results support reuse across the tested physical and directional changes, while identifying geometry and distribution-specific limitations. Figure 3 shows representative test-ID tangent-response fields and pointwise errors. Each OOD type is visualized in Figures 6–8. Parameter-matched structural ablations across the four two-dimensional families further evaluate the architecture design, with results reported in Appendix D.7. Detailed OOD results of two-dimensional benchmarks are reported in Appendix D.4.

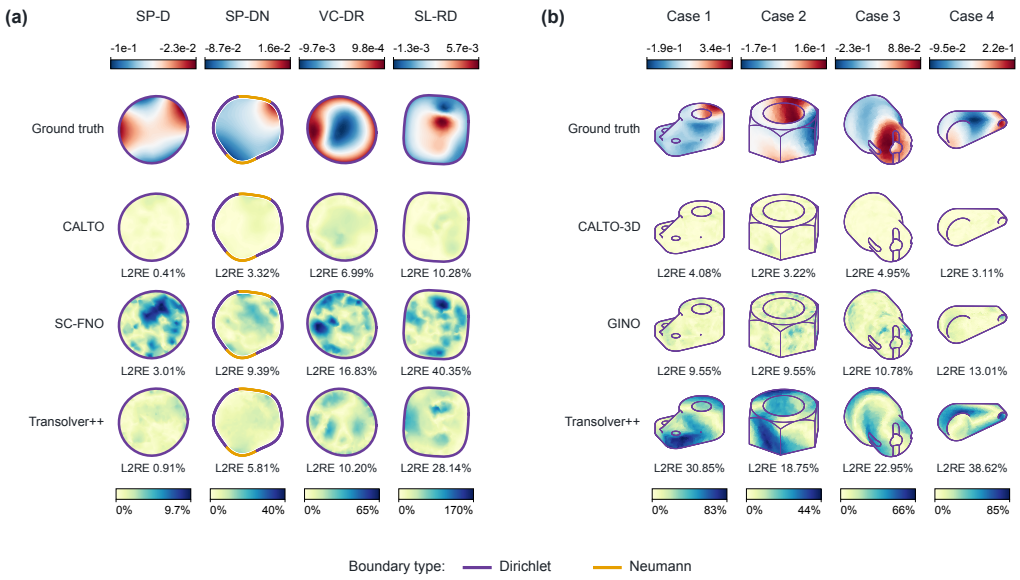

Figure 3: Representative test-ID tangent-response fields and pointwise errors. (a) Shows a representative 2D perturbation direction, and (b) The 3D coefficient perturbation. Ground-truth rows show the reference tangent responses, and error maps show the spatial distribution of prediction errors, reported in L2RE. Field and error color scales are set separately for each case.

5.2 Effects of Walk Count and Training Data

To examine how stochastic weak-supervision settings affect tangent learning, we use SP-DN as a testbed and vary the number of walks per stochastic query W, training contexts N, and perturbation directions per context R (Figure 4). We also train the same CALTO architecture on numerical reference tangent responses with $N = 4000$ and $R = 8$, providing a strong-supervision reference. CALTO achieves substantially lower errors than the raw weak labels across all tested walk counts. Increasing W from 4 to 16 reduces ID/direction-OOD L2RE from 3.03%/3.12% to 2.31%/2.38%, approaching the strong-supervision reference of 1.89%/1.97%. Errors also decrease as N or R increases (Appendix D.6). Notably, two configurations with the same $NR = 8000$ context–direction pairs yield substantially different results. At fixed NR, using 4000 contexts with two directions each yields ID and direction-OOD L2REs of 4.06% and 4.29%, respectively, compared with 4.83% and 5.07% for 1000 contexts with eight directions each, corresponding to relative reductions of approximately 16% and 15%. These results favor broader coverage of PDE contexts over denser directional sampling within each context.

5.3 Directional Linearity and Derivative Recovery

We compare direction-linear and nonlinear response models under sparse, noisy supervision, then assess recovery of the derivative matrix and its transpose actions. On SP-DN, we compare CALTO-Linear with CALTO-Nonlinear, which adds nonlinearities to the response layers and query readout while retaining the same context encoder, conditioning, and support points. Table 3 varies the number of supervised directions R and the walk count W, and also includes strong-reference supervision (Appendix D.6). CALTO-Linear has lower errors in the reported ID and direction-OOD compar-

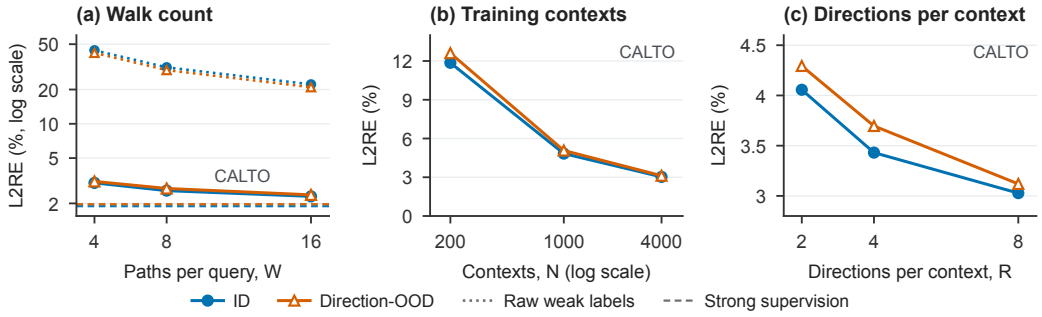

Figure 4: Effects of walk count and training data on SP-DN. ID and direction-OOD L2RE versus (a) Walk count W, (b) Training contexts N, and (c) Perturbation directions per context R, varying one factor at a time from $(N, R, W) = (4000, 8, 4)$. Solid, dotted, and dashed lines in (a) denote CALTO, raw weak labels, and the strong-supervision reference, respectively.

isons, with the largest gap at $W = 4$ and $R = 2$. This suggests that exact directional linearity is particularly useful when supervision contains only a small number of noisy directional observations.

Table 3: Direction prediction on SP-DN with different linearity configurations. Each entry reports linear/nonlinear variant with L2RE (%).

Supervision	ID	Direction OOD
W4, R2	**4.06** / 4.38	**4.29** / 4.66
W4, R4	**3.43** / 3.63	**3.70** / 3.84
W4, R8	**3.03** / 3.17	**3.12** / 3.25
W4, Full directions	**2.76** / 2.79	- / -
W8, R8	**2.59** / 2.67	**2.70** / 2.78
W16, R8	**2.31** / 2.37	**2.38** / 2.47
Strong, R8	**1.89** / 1.97	**1.97** / 2.07

Table 4: Full-parameter derivative recovery under W4R2 supervision. Entries are Jacobian / VJP relative error (%).

Family	Linear	Nonlinear	Raw W4
SP-D	**0.80 / 0.74**	0.93 / 0.89	13.18 / 9.89
SP-DN	**3.49 / 3.21**	3.97 / 3.80	40.35 / 29.22
VC-DR	**11.37** / 11.71	11.43 / **11.51**	83.31 / 60.25
SL-RD	**9.74 / 10.17**	10.03 / 10.59	73.32 / 58.65

We next evaluate derivative recovery at $W = 4$, $R = 2$ on the first 128 test-ID contexts of each 2D family, with all 1,024 queries. Let S_c contain the prescribed perturbation magnitudes (Appendix D.1) on its diagonal and e_j be the jth standard basis vector in $\mathbb{R}^{d_c}$. Stacking these responses gives the full-parameter derivative matrix:

$$\widehat{K}_c = [F_\theta(c, S_c e_j; X)]_{j=1}^{d_c}, \qquad K_c^\star = J_{c,X} S_c, \qquad \widehat{K}_c, K_c^\star \in \mathbb{R}^{1024 \times d_c}. \tag{8}$$

Once the Jacobian is reconstructed, VJPs require no additional network evaluation: for a query-space weight vector $w \in \mathbb{R}^{1024}$,

$$\widehat{g}_c(w) = \widehat{K}_c^\top w, \qquad g_c^\star(w) = (K_c^\star)^\top w. \tag{9}$$

Each VJP has d_c entries, one per parameter. The dimensions are listed in Appendix D.8. Table 4 reports the context-averaged relative Frobenius error of the reconstructed Jacobian and the L2RE of the corresponding VJPs. For each context, we evaluate VJPs using four Rademacher and four smooth localized query-space weight vectors, each normalized to unit Euclidean norm. VJP errors are averaged over these eight weights and then over contexts. Appendix D.8 specifies the weight construction and evaluation details. Under W4R2 supervision, CALTO-Linear has lower Jacobian error in all four families and lower VJP error in SP-D, SP-DN, and SL-RD. The differences are largest on SP-DN, where the Jacobian error decreases from 3.97% to 3.49% and the VJP error from 3.80% to 3.21%. VC-DR is effectively tied: CALTO-Linear has slightly lower Jacobian error (11.37% versus 11.43%), whereas CALTO-Nonlinear has slightly lower VJP error (11.51% versus 11.71%). These results support direction-linear models for predicting tangent responses and assembling reusable derivative matrices from limited directional supervision.

5.4 Inverse Problems with Learned Tangent Operators

We finally test whether learned tangent information improves coefficient identification while retaining FEM for forward evaluation and step acceptance. For VC-DR, we recover the six coefficients ξ in $\log \kappa = \sum_{j=1}^{6} \xi_j \phi_j$ with preset spatial basis functions ϕ_j (Appendix D.1), holding the domain, source, reaction coefficient, and homogeneous Dirichlet data fixed. Observations and forward evaluations use P2 FEM with $h/D = 0.01$ (mesh-size parameter h, domain diameter D). Results are averaged over 16 problems with two initializations each, using 128 noiseless observations for optimization and 384 other interior points for reconstruction. Let $\mathcal{F}(\xi)$ denote the FEM sensor response and y the target. The residual ratio is $E_N = \frac{\|\mathcal{F}(\xi_N)-y\|_2}{\|\mathcal{F}(\xi_0)-y\|_2}$, where ξ_N is the last accepted estimate within N FEM forward evaluations.

Jacobian substitution. The first experiment varies the Jacobian source while FEM supplies the residuals and evaluates candidate steps (Table 5a). All methods share damped Gauss–Newton and trust-region settings (Appendix D.10). CALTO reduces mean E_8 by 95.6% and 98.0% relative to SC-FNO AD and DE-DeepONet AD, respectively. Both surrogates receive weak primal and JVP supervision in this experiment. CALTO-Hybrid jointly predicts primal and tangent responses: replacing AD of its primal output with its direct JVPs reduces mean E_{16} from 1.33×10^{-3} to 1.72×10^{-4}, an 87.1% reduction. FEM tangents give the lowest final residuals.

Table 5: Coefficient identification on 16 VC-DR instances with two initializations each. Mean E_N after at most N FEM forward evaluations (lower is better). (a) Jacobian substitution. (b) Surrogate tangent correction: value correction (V), alone or with a CALTO or FEM derivative. Hybrid denotes CALTO-Hybrid; FEM uses P2 elements.

(a) Jacobian substitution

Jacobian	E_8	E_{16}
SC-FNO (AD)	1.61×10^{-1}	1.19×10^{-1}
DE-DeepONet (AD)	3.53×10^{-1}	3.12×10^{-1}
Hybrid (AD)	1.13×10^{-2}	1.33×10^{-3}
CALTO	7.13×10^{-3}	1.71×10^{-3}
Hybrid (JVP)	6.24×10^{-3}	1.72×10^{-4}
FEM	2.55×10^{-6}	2.55×10^{-6}

(b) Surrogate tangent correction

Surrogate	Correction	E_8	E_{16}
SC-FNO	V	2.05×10^{-1}	1.59×10^{-1}
	V+CALTO	7.25×10^{-3}	1.76×10^{-3}
	V+FEM	3.91×10^{-6}	3.91×10^{-6}
DE-DeepONet	V	3.81×10^{-1}	3.22×10^{-1}
	V+CALTO	7.24×10^{-3}	1.62×10^{-3}
	V+FEM	3.07×10^{-3}	2.52×10^{-6}

Surrogate tangent correction. For a fixed primal surrogate $\widehat{\mathcal{F}}$, this experiment compares two local models of the sensor response at a trial parameter $\xi_k + s$. These models, $m_k^V(s)$ and $m_k^{V+D}(s)$, are used to propose the optimization step s:

$$m_k^V(s) = \widehat{\mathcal{F}}(\xi_k + s) + \mathcal{F}(\xi_k) - \widehat{\mathcal{F}}(\xi_k), \quad m_k^{V+D}(s) = m_k^V(s) + (D_k - \widehat{J}_k)s. \tag{10}$$

Here $\widehat{J}_k = D_\xi \widehat{\mathcal{F}}(\xi_k)$. Value correction (V) adjusts the surrogate's current output using FEM. Tangent correction also adjusts its derivative at ξ_k using D_k, supplied by CALTO (V+CALTO) or FEM (V+FEM), while retaining the surrogate's nonlinear dependence on s. CALTO tangent correction reduces mean E_{16} by 98.9% for SC-FNO and 99.5% for DE-DeepONet (Table 5b). Parameter recovery and reconstruction at unobserved points also improve (Appendix D.10). CALTO thus improves coefficient identification both by supplying the inversion Jacobian and by correcting a primal surrogate's local derivative. Appendix D.11 presents further inverse examples with 2D Dirichlet boundary control and 3D mixed physical control.

6 Conclusion

We introduced CALTO, a reusable PDE tangent operator learned directly from few-path stochastic directional observations. By separating nonlinear context conditioning from a linear response pathway, CALTO preserves directional superposition. Across four 2D elliptic families and a 3D CAD benchmark, it improves tangent prediction and supports Jacobian/VJP recovery and coefficient identification. Together, these results support direct tangent-operator learning as a reusable complement to forward models for sensitivity analysis and PDE-constrained inverse problems.

AI USE STATEMENT

AI tools, including OpenAI Codex, were used to aid and polish the writing, assist with LaTeX formatting and layout, and support the search for relevant literature. They were also used to assist in generating experimental code and checking consistency between the implementation, experimental configurations, and reported results. All AI-assisted content, code, and revisions were reviewed and verified by the authors.

REPRODUCIBILITY STATEMENT

A minimal runnable implementation of the methods and experiments presented in this work is provided in an anonymous repository at `https://anonymous.4open.science/r/calto-55DD`. The repository includes code and example configurations for reproducing the core experimental pipeline.

REFERENCES

Abdolmehdi Behroozi, Chaopeng Shen, and Daniel Kifer. Sensitivity-constrained fourier neural operators for forward and inverse problems in parametric differential equations. In *International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=DPzQ5n3mNm.

Abdolmehdi Behroozi, Chaopeng Shen, Daniel Kifer, and Kathryn Lawson. Sensitivity-constrained neural operators for data-efficient forward and inverse modeling of partial differential equation systems, 2026. URL https://arxiv.org/abs/2608.29888.

Nicolas Boullé, Diana Halikias, Samuel E. Otto, and Alex Townsend. Operator learning without the adjoint. *Journal of Machine Learning Research*, 25(364):1–54, 2024. URL http://jmlr.org/papers/v25/24-0162.html.

Wojciech M. Czarnecki, Simon Osindero, Max Jaderberg, Grzegorz Swirszcz, and Razvan Pascanu. Sobolev training for neural networks. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017. URL https://proceedings.neurips.cc/paper_files/paper/2017/file/758a06618c69880a6cee5314ee42d52f-Paper.pdf.

Michael B Giles and Niles A Pierce. An introduction to the adjoint approach to design. *Flow, turbulence and combustion*, 65(3):393–415, 2000.

Xindi Gong, Dingcheng Luo, Thomas O'Leary-Roseberry, Ruanui Nicholson, and Omar Ghattas. Shape derivative-informed neural operators with application to risk-averse shape optimization, 2026. URL https://arxiv.org/abs/2603.03211.

Philipp Holl, Nils Thuerey, and Vladlen Koltun. Learning to control pdes with differentiable physics. In *International Conference on Learning Representations*, 2020. URL https://openreview.net/forum?id=HyeSin4FPB.

Sebastian Koch, Albert Matveev, Zhongshi Jiang, Francis Williams, Alexey Artemov, Evgeny Burnaev, Marc Alexa, Denis Zorin, and Daniele Panozzo. Abc: A big cad model dataset for geometric deep learning. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019.

Zongyi Li, Nikola Kovachki, Kamyar Azizzadenesheli, Burigede Liu, Kaushik Bhattacharya, Andrew Stuart, and Anima Anandkumar. Fourier neural operator for parametric partial differential equations. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=c8P9NQVtmnO.

Zongyi Li, Nikola Kovachki, Chris Choy, Boyi Li, Jean Kossaifi, Shourya Otta, Mohammad Amin Nabian, Maximilian Stadler, Christian Hundt, Kamyar Azizzadenesheli, and Animashree Anandkumar. Geometry-informed neural operator for large-scale 3d PDEs. In *Advances in Neural Information Processing Systems*, volume 36, pp. 35836–35854, 2023. doi: 10.52202/

075280-1556. URL https://proceedings.neurips.cc/paper_files/paper/2023/file/70518ea42831f02afc3a2828993935ad-Paper-Conference.pdf.

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(3):218–229, 2021. doi: 10.1038/s42256-021-00302-5. URL https://doi.org/10.1038/s42256-021-00302-5.

Huakun Luo, Haixu Wu, Hang Zhou, Lanxiang Xing, Yichen Di, Jianmin Wang, and Mingsheng Long. Transolver++: An accurate neural solver for PDEs on million-scale geometries. In Aarti Singh, Maryam Fazel, Daniel Hsu, Simon Lacoste-Julien, Felix Berkenkamp, Tegan Maharaj, Kiri Wagstaff, and Jerry Zhu (eds.), *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 41432–41449. PMLR, 13–19 Jul 2025. URL https://proceedings.mlr.press/v267/luo25o.html.

Hugo Melchers, Joost Prins, and Michael Abdelmalik. Neural green's operators for parametric partial differential equations. *Computer Methods in Applied Mechanics and Engineering*, 455: 118893, 2026. doi: 10.1016/j.cma.2026.118893. URL https://doi.org/10.1016/j.cma.2026.118893.

Bailey Miller, Rohan Sawhney, Keenan Crane, and Ioannis Gkioulekas. Differential walk on spheres. *ACM Transactions on Graphics*, 43(6):1–18, 2024. doi: 10.1145/3687913. URL https://doi.org/10.1145/3687913.

Hong Chul Nam, Julius Berner, and Anima Anandkumar. Solving poisson equations using neural walk-on-spheres. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 37277–37292. PMLR, 2024. URL https://proceedings.mlr.press/v235/nam24a.html.

Thomas O'Leary-Roseberry, Peng Chen, Umberto Villa, and Omar Ghattas. Derivative-informed neural operator: An efficient framework for high-dimensional parametric derivative learning. *Journal of Computational Physics*, 496:112555, 2024. doi: 10.1016/j.jcp.2023.112555. URL https://doi.org/10.1016/j.jcp.2023.112555.

Yuan Qiu, Nolan Bridges, and Peng Chen. Derivative-enhanced deep operator network. In *Advances in Neural Information Processing Systems*, volume 37, pp. 20945–20981, 2024. doi: 10.52202/079017-0660. URL https://proceedings.neurips.cc/paper_files/paper/2024/hash/25297252dcd3f4b11eec7ec7ab06bc80-Abstract-Conference.html.

Rohan Sawhney and Keenan Crane. Monte carlo geometry processing: a grid-free approach to pde-based methods on volumetric domains. *ACM Trans. Graph.*, 39(4), August 2020. ISSN 0730-0301. doi: 10.1145/3386569.3392374. URL https://doi.org/10.1145/3386569.3392374.

Rohan Sawhney, Dario Seyb, Wojciech Jarosz, and Keenan Crane. Grid-free Monte Carlo for PDEs with spatially varying coefficients. *ACM Transactions on Graphics*, 41(4), 2022. doi: 10.1145/3528223.3530134.

Rohan Sawhney, Bailey Miller, Ioannis Gkioulekas, and Keenan Crane. Walk on stars: A grid-free monte carlo method for PDEs with neumann boundary conditions. *ACM Transactions on Graphics*, 42(4):1–20, 2023. doi: 10.1145/3592398. URL https://doi.org/10.1145/3592398.

Hrishikesh Viswanath, Hong Chul Nam, Xi Deng, Julius Berner, Anima Anandkumar, and Aniket Bera. Operator learning using weak supervision from walk-on-spheres, 2026. URL https://arxiv.org/abs/2603.01193.

Haixu Wu, Huakun Luo, Haowen Wang, Jianmin Wang, and Mingsheng Long. Transolver: A fast transformer solver for PDEs on general geometries. In Ruslan Salakhutdinov, Zico Kolter, Katherine Heller, Adrian Weller, Nuria Oliver, Jonathan Scarlett, and Felix Berkenkamp (eds.),

Proceedings of the 41st International Conference on Machine Learning, volume 235 of *Proceedings of Machine Learning Research*, pp. 53681–53705. PMLR, 21–27 Jul 2024. URL https://proceedings.mlr.press/v235/wu24r.html.

Tianju Xue, Shuheng Liao, Zhengtao Gan, Chanwook Park, Xiaoyu Xie, Wing Kam Liu, and Jian Cao. JAX-FEM: A differentiable GPU-accelerated 3D finite element solver for automatic inverse design and mechanistic data science. *Computer Physics Communications*, 291:108802, 2023. doi: 10.1016/j.cpc.2023.108802.

Boyuan Yao, Dingcheng Luo, Lianghao Cao, Nikola Kovachki, Thomas O'Leary-Roseberry, and Omar Ghattas. Derivative-informed fourier neural operator: Universal approximation and applications to PDE-constrained optimization, 2026. URL https://arxiv.org/abs/2512.14086.

Ekrem Fatih Yilmazer, Delio Vicini, and Wenzel Jakob. Solving inverse pde problems using grid-free monte carlo estimators. *ACM Trans. Graph.*, 43(6), November 2024. ISSN 0730-0301. doi: 10.1145/3687990. URL https://doi.org/10.1145/3687990.

Seungwoo Yoo, Kyeongmin Yeo, Jisung Hwang, and Minhyuk Sung. Neural green's functions. In *Advances in Neural Information Processing Systems*, volume 38, pp. 47961–47982, 2025. doi: 10.52202/085713-1599. URL https://proceedings.neurips.cc/paper_files/paper/2025/file/448abd486677165ceedfa790e9a61802-Paper-Conference.pdf.

Zihan Yu, Lifan Wu, Zhiqian Zhou, and Shuang Zhao. A differential monte carlo solver for the poisson equation. In *ACM SIGGRAPH 2024 Conference Papers*, SIGGRAPH '24, New York, NY, USA, 2024. Association for Computing Machinery. ISBN 9798400705250. doi: 10.1145/3641519.3657460. URL https://doi.org/10.1145/3641519.3657460.

A Tangent Problems and Numerical Targets

We expand the formulation in Section 3 using the same notation: ϑ_0 denotes the physical base parameters, c their PDE context, and v a perturbation direction. The exact base state u_0 defines $J_{c,X}$; approximate base fields enter only the numerical targets and, where supplied, the model context.

A.1 Tangent equations and boundary data

Proposition A.1 (Tangent-operator linearity) *Under the differentiability and invertibility assumptions of Section 3, with P_X well defined and linear, $J_{c,X} = P_X L_c^{-1} \mathcal{B}_c$ is a linear map from $\mathbb{R}^d$ to $\mathbb{R}^Q$.*

Proof. Differentiating $\mathcal{F}(\mathcal{G}(\vartheta_0 + tv), \vartheta_0 + tv) = 0$ at $t = 0$ gives $L_c \delta u = \mathcal{B}_c v$, with both derivatives evaluated at (u_0, ϑ_0). Applying the linear maps L_c^{-1} and P_X proves the claim. □

For fixed geometry, the four 2D families have the tangent equations

$$(-\Delta + \lambda)\delta u = \delta f - u_0 \delta\lambda, \tag{11}$$

$$-\nabla \cdot (\kappa \nabla \delta u) + q\delta u = \delta f - u_0 \delta q + \nabla \cdot (\delta\kappa \nabla u_0), \tag{12}$$

$$[-\Delta + \lambda + 3\gamma u_0^2]\delta u = \delta f - u_0 \delta\lambda - u_0^3 \delta\gamma. \tag{13}$$

These correspond to SP-D/SP-DN, VC-DR, and SL-RD, respectively. Here $\kappa > 0$ is diffusivity, q a linear reaction coefficient, and γ the coefficient of the cubic term γu^3. Equation 11 is the interior equation in Equation 2, with the same fixed-boundary data $\delta u = \delta g_D$ on Γ_D and $\partial_n \delta u = \delta g_N$ on Γ_N. The 2D VC-DR and SL-RD benchmarks have zero tangent Dirichlet data; VC-DR uses $\delta\kappa = \kappa\, \delta \log \kappa$. CAD-DR uses Equation 12 in 3D with Dirichlet variation δg from its analytic fields.

For SP-D shape directions with zero base Dirichlet trace, let V_n be the prescribed normal boundary velocity. Differentiating the boundary trace along the deformation gives the Eulerian response

$$\delta u|_{\partial\Omega} = \delta g_D - V_n \partial_n u_0. \tag{14}$$

The training shape templates set $\delta g_D = 0$. The input supplies V_n, while $\partial_n u_0$ is fixed by the base problem; the tangent is therefore linear in $(\delta g_D, V_n)$. Only the outer boundary moves in the SP-D benchmark; fixed holes satisfy $V_n = \delta g_D = 0$ and hence $\delta u = 0$. This derivative describes infinitesimal boundary motion at fixed topology.

A.2 Variable-coefficient transformation

For VC-DR and CAD-DR, $U = \sqrt{\kappa}\, u$ satisfies $(-\Delta + Q_\kappa)U = f/\sqrt{\kappa}$, with

$$Q_\kappa = \frac{q}{\kappa} + \frac{1}{2}\Delta \log \kappa + \frac{1}{4}|\nabla \log \kappa|^2. \tag{15}$$

Writing $a = \log \kappa$ and $U_0 = \sqrt{\kappa}\, u_0$, differentiation gives

$$\begin{aligned} \delta Q_\kappa &= \frac{\delta q - q\delta a}{\kappa} + \tfrac{1}{2}\Delta\delta a + \tfrac{1}{2}\nabla a \cdot \nabla \delta a, \\ (-\Delta + Q_\kappa)\delta U &= \kappa^{-1/2}(\delta f - \tfrac{1}{2} f \delta a) - \delta Q_\kappa U_0, \\ \delta u &= \kappa^{-1/2}\delta U - \tfrac{1}{2} u_0 \delta a. \end{aligned} \tag{16}$$

All coefficients multiplying the perturbations are fixed at the base problem. The learning target remains the physical response δu.

A.3 Numerical targets and references

Fix c, X, and the numerical settings. Let $\widetilde{J}_{c,X}$ denote the response of the numerical tangent boundary-value problem and $A_\omega : \mathbb{R}^d \to \mathbb{R}^Q$ the estimator for one complete random realization over X (Appendix B.1). Its mean $\overline{J}_{\mathrm{num}} = \mathbb{E}A_\omega$ can differ from both $\widetilde{J}_{c,X}$ and the exact target $J_{c,X}$:

$$\overline{J}_{\mathrm{num}} - J_{c,X} = (\widetilde{J}_{c,X} - J_{c,X}) + (\overline{J}_{\mathrm{num}} - \widetilde{J}_{c,X}). \tag{17}$$

The first term includes approximate base states and fields; the second includes stopping and boundary-evaluation approximations. These biases persist as the path count increases.

All four 2D weak-label constructions use coarse quadratic finite-element (P2) states $\widetilde{u}_0$ and bilinearly interpolated 64×64 field grids. SP-D/SP-DN use the source $\delta f - \widetilde{u}_0 \delta\lambda$; SP-D shape responses use a shared nested estimate of $\partial_n u_0$. VC-DR assembles the transformed source

$$b_C(v) = \kappa^{-1/2}(\delta f - \tfrac{1}{2} f \delta a) - \delta Q_\kappa \sqrt{\kappa}\, \widetilde{u}_0, \tag{18}$$

then restores the physical response using Equation 16 with u_0 replaced by $\widetilde{u}_0$. SL-RD uses the potential $\lambda + 3\gamma\widetilde{u}_0^2$ and source $\delta f - \widetilde{u}_0\delta\lambda - \widetilde{u}_0^3\delta\gamma$. Only SL-RD supplies the coarse state explicitly to the learned model.

Finer P2 solves of the physical tangent equations provide validation and test references independently of the weak-label construction. CAD-DR instead uses analytic manufactured references (Appendix D.2).

B Stochastic Estimators and Directional Structure

B.1 Estimator construction

For a fixed context and query set, each component of $A_\omega v$ has the form

$$\begin{aligned} (A_\omega v)_i = &\sum_k t_{\omega k}(c) b_c(v; \xi_{\omega k}) \\ &+ \sum_j s_{\omega j}(c) g_c(v; z_{\omega j}) + \ell_{\omega,c}(v). \end{aligned} \tag{19}$$

The query index i is suppressed on the right-hand side. Here $\xi_{\omega k}$ and $z_{\omega j}$ are sampled interior and boundary locations, and $t_{\omega k}$ and $s_{\omega j}$ are path weights. The source b_c, boundary data g_c (including shape terms), and restoration $\ell_{\omega,c}$ implement the numerical tangent problem, possibly after transformation, and are linear in v. Locations, weights, termination, and validity decisions depend only on the base problem and randomness. Each query shares its complete realization across directions, including all nested estimates; stacking query components gives A_ω.

Writing D for the domain diameter, SP-D uses screened WoS and, for shape responses, differential WoS (Miller et al., 2024) with four shared nested primal walks at inward offset $0.01D$. SP-DN uses Walk on Stars (Sawhney et al., 2023); its Neumann queries estimate solution responses by $\delta u(z - \varepsilon n) + \varepsilon \delta g_N(z)$ with $\varepsilon = 0.002D$. VC-DR and SL-RD use signed delta tracking (Sawhney et al., 2022) with numerical potential $V_c = Q_\kappa$ and $V_c = \lambda + 3\gamma\widetilde{u}_0^2$, respectively. The reference potential is $m = \max\{10^{-4}, 1.01 \max_{\text{grid}} |V_c| + 10^{-6}\}$, and the volume multiplier is $1 - V_c/m$. Negative potentials can increase path throughput; the second-moment analysis below therefore assumes finite variance.

The 2D outer shell width is $10^{-3}D$ and the step cap is 512. Recorded training labels have zero cap-hit rates. Direction-independent selection preserves linearity; the $1/W$ variance identity below applies to a fixed number of complete iid realizations.

B.2 Shared realizations and second moments

Fix c, X, the numerical settings, and $V = [v_1, \ldots, v_R]$; write $J = J_{c,X}$. Assume that each complete realization defines a linear map A_ω.

Proposition B.1 (Shared-path linearity) *Let $\widehat{J}_W = W^{-1} \sum_{w=1}^{W} A_{\omega_w}$ and $Y_j = \widehat{J}_W v_j$. Then $Y = \widehat{J}_W V$, $Y\alpha = \widehat{J}_W(V\alpha)$, and $\ker V \subseteq \ker Y$. No independence, unbiasedness, or moment assumption is needed.*

Proof. Each column satisfies $Y_j = \widehat{J}_W v_j$. Hence $Y\alpha = \sum_j \alpha_j \widehat{J}_W v_j = \widehat{J}_W(V\alpha)$. If $V\alpha = 0$, the same identity gives $Y\alpha = 0$. □

For stochastic error, define $\mathsf{B}^{\text{num}} = \mathbb{E}A_\omega - J$ and $N_\omega = A_\omega - \mathbb{E}A_\omega$. For fixed $H \in \mathbb{R}^{Q\times Q}$ with $H \succeq 0$, let $\|z\|_H^2 = z^\top H z$ and $\Gamma_H = \mathbb{E}[N_\omega^\top H N_\omega] \in \mathbb{R}^{d\times d}$.

Proposition B.2 (Bias and covariance) *For iid complete realizations with $\mathbb{E}\|A_\omega\|_F^2 < \infty$,*

$$\mathbb{E}\|\widehat{J}_W v - Jv\|_H^2 = \|\mathsf{B}^{\text{num}} v\|_H^2 + W^{-1} v^\top \Gamma_H v, \tag{20}$$

$$\mathbb{E}[(Y - \mathbb{E}Y)^\top H (Y - \mathbb{E}Y)] = W^{-1} V^\top \Gamma_H V. \tag{21}$$

Proof. For $\overline{N}_W = W^{-1} \sum_w N_{\omega_w}$, centering and independence give $\mathbb{E}[\overline{N}_W^\top H \overline{N}_W] = \Gamma_H / W$. Expanding $\widehat{J}_W - J = \mathsf{B}^{\text{num}} + \overline{N}_W$ and using $Y - \mathbb{E}Y = \overline{N}_W V$ gives both identities. □

Equation 21 retains cross-direction covariance. Single-direction bias and variance follow Equation 20. The supervision study varies the number of training contexts N, directions per context R, and realizations per query W within this scheme.

B.3 Recombination error and coverage

Fix the point sets $\mathcal{P} = (X, X_\Omega, X_\partial)$ and query grouping, as in Section 3. For a direction-linear predictor $F_\theta(c, v; \mathcal{P}) = A_\theta v$, set $R_\theta = A_\theta V - Y$. For CALTO, $A_\theta = \mathcal{A}_\theta(c; \mathcal{P})\mathcal{C}_{c,\mathcal{P}}$ is the full map from physical directions to query responses (Equation 3). Then

$$F_\theta(c, V\alpha; \mathcal{P}) - JV\alpha = (\widehat{J}_W - J)V\alpha + R_\theta \alpha. \tag{22}$$

Thus observation and fitting errors remain even with exact linearity. Coverage is limited to $\operatorname{span}(V)$: if $a \in \mathbb{R}^d$ is orthogonal to this span, then $(J + ba^\top)V = JV$ for any $b \in \mathbb{R}^Q$. Responses outside that span require additional information, such as learning across contexts.

C Input Representation and Architecture

C.1 Physical point-set inputs

The point sets $\mathcal{P}$ carry context vectors $(c_{\mathrm{q}}, c_{\Omega}, c_{\partial})$ and physical perturbation features $r = (r_{\mathrm{q}}, r_{\Omega}, r_{\partial}) = \mathcal{C}_{c,\mathcal{P}} v$. The features r sample input-field variations and shape velocities; they differ from $\mathcal{B}_c v$, the induced tangent-equation data. All directions at a base problem use the same point locations and context vectors, so $\mathcal{C}_{c,\mathcal{P}}$ is a fixed linear map.

In 2D, the context at each point contains its coordinates (x, y), outward normal (n_x, n_y), two indicators for Dirichlet and Neumann boundary type, and the base fields (f, λ, g_D, g_N). These fields are the source, screening coefficient, Dirichlet value and Neumann flux. Two further channels, $\sin(2\pi s)$ and $\cos(2\pi s)$, encode normalized arclength $s \in [0, 1)$ on each boundary component. Boundary-specific entries are zero in the interior, and other inapplicable entries are also zero. The base parameter vector described below is repeated at every point. Table 6 lists the additional context fields and the direction channels.

Table 6: 2D input features. Context counts include the common fields and base parameter vector. The listed direction channels are used on all three point sets.

Family	Context	Additional context	Direction channels
SP-D	37	None	δf, $\delta\lambda$, δg_D, V_n
SP-DN	40	None	δf, $\delta\lambda$, δg_D, δg_N
VC-DR	43	$\log\kappa$, κ, q and two supplied basis amplitudes	δf, $\delta\log\kappa$, δq
SL-RD	29	$\widetilde{u}_0$, $\lambda + 3\gamma\widetilde{u}_0^2$	δf, $\delta\lambda$, $\delta\gamma$

The direction vector has four channels for SP-D/SP-DN and three for VC-DR/SL-RD. Boundary-data variations are zero off their corresponding boundary, and normal velocity V_n is zero in the interior. For example, a pure SP-D screening change uses $(0, \delta\lambda, 0, 0)$ at every point. Shape velocities are evaluated on the fixed base boundary.

All four families include 13 source parameters: the offset, three amplitudes, six center coordinates and three log-widths. SP-D adds λ, seven Dirichlet Fourier coefficients and four shape slots set to zero at the base geometry. SP-DN adds λ and seven Fourier coefficients for each boundary type. VC-DR adds the six diffusivity coefficients, reaction offset and six reaction coefficients; SL-RD adds λ and γ. Appendix D.1 specifies these parameters and the shape basis. The two extra VC-DR context amplitudes multiply $\cos(2\pi x)$ and $\sin(2\pi y)$ in $\log\kappa$; they are zero unless supplied by the coefficient shift.

For CAD-DR, use the transformed potential Q_κ in Equation 15, the source $f_\kappa = f/\sqrt{\kappa}$, and the Dirichlet trace $g_\kappa = \sqrt{\kappa}\, g$ of $U = \sqrt{\kappa}\, u$. Query and interior-support contexts contain coordinates, five point-type indicators and $(\kappa, q, f, \log\kappa, Q_\kappa, f_\kappa)$. The indicators identify volume sampling or one of four near-boundary distance bands (Appendix D.2). Boundary-support contexts contain coordinates, normals and $(g, g_\kappa, \kappa, q, f, \log\kappa, Q_\kappa, f_\kappa)$. Both context vectors have 14 channels. Query and interior-support direction vectors have six channels, $(\delta\kappa, \delta q, \delta f, \delta\log\kappa, \delta Q_\kappa, \delta f_\kappa)$; boundary-support vectors also include $(\delta g, \delta g_\kappa)$, giving eight. These variations come from differentiating the analytic fields in Appendix D.2. We apply $\mathrm{sign}(z)\log(1 + |z|)$ to the base scalar fields, including g and g_κ. Coordinates, normals, indicators and direction channels are untransformed. The output target is the physical δu.

C.2 Network architecture

The 2D and 3D models use 48 response channels, four attention heads of 12 channels each, and four successive response layers. The following steps describe one perturbation direction. Other directions follow the same steps with the same context features, weights and gates. We reserve r for physical inputs and h for the 48-channel hidden responses.

Context encoders. In the context branch, query and interior-support context vectors pass through the same encoder. Boundary-support contexts use a separate encoder with its own parameters. Each

encoder applies a fully connected layer with 48 outputs, GELU, a second 48-output layer, GELU, then LayerNorm at each point. Its output is a 48-channel context vector per point, kept fixed throughout the response layers. For an input context vector z at one point, the encoded vector e is

$$e = \mathrm{LayerNorm}(\mathrm{GELU}(\mathrm{FC}_2(\mathrm{GELU}(\mathrm{FC}_1(z))))) \,.$$

Here z represents the PDE context c at that point, and e has 48 channels. FC_1 and FC_2 are the encoder's two learned affine layers, including biases; GELU is the activation, and LayerNorm normalizes the 48 context channels.

Attention weights and gates. The encoded contexts feed two sets of learned projections: one computes attention weights, and the other produces gates. For attention, the target context is projected into a query vector and each source context into a key vector. Both projections are affine and have 48 outputs, split into four heads. Each head takes the dot product between its query and each key, divides by $\sqrt{12}$, and applies softmax over source points. This gives one weight per source point for each target and head. For head a, the weight is

$$\alpha_{ij}^{(a)} = \frac{\exp\Big(\langle q_i^{(a)}, k_j^{(a)} \rangle / \sqrt{12}\Big)}{\sum_{m=1}^{N_s} \exp\Big(\langle q_i^{(a)}, k_m^{(a)} \rangle / \sqrt{12}\Big)}.$$

Here i indexes target points, j and m index the N_s source points, and $a \in \{1, 2, 3, 4\}$ indexes heads. The 12-channel vectors $q_i^{(a)}$ and $k_j^{(a)}$ are the projected target and source contexts; $\langle \cdot, \cdot \rangle$ is their dot product, and $\alpha_{ij}^{(a)}$ is the resulting attention weight.

A gate projection takes the context at one point and produces 48 numbers, one for each response channel. Sigmoid normalizes these numbers between zero and one. Gating multiplies the response feature vector element-wise by the context-dependent gate vector at each point. The state-retention gate adds 0.5 after sigmoid, so its values lie between 0.5 and 1.5. For an ordinary gate at one point,

$$g = \mathrm{sigmoid}(\mathrm{FC}_{\mathrm{gate}}(e)), \qquad \widetilde{h} = g \odot h.$$

Here $\mathrm{FC}_{\mathrm{gate}}$ is a learned affine map from the encoded context e to 48 channels, g is its gate vector, h is the hidden response at the same point, and $\widetilde{h}$ is the gated response. The symbol $\odot$ means element-wise multiplication. Initialization and each response layer have their own projection parameters. The projections depend only on context and are shared across directions.

To apply attention, a bias-free linear layer maps each source response to 48 value channels, split into four groups of 12. Each head uses its weights to sum these source values at every target. The four resulting 12-channel vectors are concatenated into 48 channels and passed through a bias-free output layer. Attention therefore returns one 48-channel response vector per target point. Each attention module has its own query, key, value and output projections, shared across points and directions. This aggregation can be written as

$$o_i^{(a)} = \sum_{j=1}^{N_s} \alpha_{ij}^{(a)} W_V^{(a)} h_j, \qquad t_i = W_O[\, o_i^{(1)}; o_i^{(2)}; o_i^{(3)}; o_i^{(4)} \,].$$

Here h_j is the 48-channel source response after input gating, $W_V^{(a)}$ is the 12×48 part of the value projection for head a, and $o_i^{(a)}$ is that head's 12-channel output. The brackets join the four outputs along the channel dimension; W_O is the 48×48 output matrix, and t_i is the resulting 48-channel response at target i. Both value and output projections have no bias.

Initial query response. The response trunk starts with three separate bias-free linear layers that map the direction vectors at queries, interior supports and boundary supports to 48 channels. Query features are multiplied by a gate generated from the query context. Interior-support features are gated using interior context, aggregated at queries using attention, then gated using query context. Boundary-support features follow the same sequence, with boundary context for the input gate and query context for the output gate. The two attention modules have independent parameters. Initialization uses five separate gate projections in total: one for the local query features and two for each

support attention module. All three outputs have 48 channels at each query; their sum is the initial response b and the starting state h_0. Let $L_{\mathrm{q}}, L_\Omega, L_\partial$ be the separate bias-free lifts of $r_{\mathrm{q}}, r_\Omega, r_\partial$ to 48 channels. Then

$$\begin{aligned} b = {} & g_q \odot L_{\mathrm{q}} r_{\mathrm{q}} \\ & + g_\Omega^{\mathrm{out}} \odot \mathrm{Attn}_{\Omega\to q}(g_\Omega^{\mathrm{in}} \odot L_\Omega r_\Omega) \\ & + g_\partial^{\mathrm{out}} \odot \mathrm{Attn}_{\partial\to q}(g_\partial^{\mathrm{in}} \odot L_\partial r_\partial), \qquad h_0 = b. \end{aligned}$$

Here g_q is the local query gate. For each support set, g^{in} is the gate at its source points and g^{out} is the gate at the target queries. $\mathrm{Attn}_{\Omega\to q}$ and $\mathrm{Attn}_{\partial\to q}$ are the four-head attention operations above, from interior and boundary supports to queries. Their weights use the corresponding fixed encoded contexts. Each lift acts pointwise; each gate array has one value per point and response channel. Both b and h_0 have 48 channels at every query.

Response layers and readout. Each of the four layers takes the current 48-channel query state h_ℓ, the initial response b, and the fixed query context. A layer-specific affine projection maps each query context to 288 numbers, split into six 48-channel gates. These gates control four parallel operations:

1. **State retention** multiplies the current state by the state-retention gate g_ℓ^{keep}.
2. **Query attention** gates the current state, aggregates values from the queries in the same group using the attention operation above, and gates the output. These two gates are $g_\ell^{\mathrm{attn,in}}$ and $g_\ell^{\mathrm{attn,out}}$. Its queries and keys both come from the fixed query context.
3. **Channel mixing** gates the current state, applies a single bias-free linear layer from 48 to 48 channels at each query, and gates the output. The gates are $g_\ell^{\mathrm{mix,in}}$ and $g_\ell^{\mathrm{mix,out}}$, and the linear layer has weight matrix W_ℓ.
4. **Input reinjection** multiplies the original b by its own gate g_ℓ^{inject}.

The four outputs are added and scaled by $1/2$ to obtain $h_{\ell+1}$:

$$\begin{aligned} h_{\ell+1} = \tfrac{1}{2}\big[& g_\ell^{\mathrm{keep}} \odot h_\ell + g_\ell^{\mathrm{attn,out}} \odot \mathrm{Attn}_\ell(g_\ell^{\mathrm{attn,in}} \odot h_\ell) \\ & + g_\ell^{\mathrm{mix,out}} \odot W_\ell(g_\ell^{\mathrm{mix,in}} \odot h_\ell) + g_\ell^{\mathrm{inject}} \odot b\big]. \end{aligned}$$

Here $\ell = 0, 1, 2, 3$ indexes the four layers, and Attn_ℓ is layer ℓ's four-head attention among queries in the same group. The six gate arrays have the same shape as h_ℓ; W_ℓ is a 48×48 matrix applied independently at each query. This update runs four times, with independent parameters in each layer and the same b reinjected each time. All states retain 48 channels per query. Finally, one shared bias-free linear layer maps the fourth state h_4 to a scalar prediction of the physical δu at each query, $[F_\theta(c, v; \mathcal{P})]_i = w_{\mathrm{read}}^\top h_{4,i}$, where $h_{4,i}$ is the 48-channel final state at query i, and w_{read} is the shared learned 48-channel readout vector. Evaluating R directions returns R scalars per query.

C.3 Directional linearity and query grouping

Attention connects only queries processed together in the same group, so changing the group can change a prediction. Evaluation uses one group of 1024 queries for each 2D problem and fixed groups of 256 for CAD-DR, with supports fixed per context (Appendices D.1 and D.2). Groups and supports stay the same across directions.

Proposition C.1 (Network linearity) *For fixed network weights θ, context c, point sets $\mathcal{P}$, and query grouping, $F_\theta(c, v; \mathcal{P})$ is linear in v and satisfies Equation 5.*

Proof. Changing v leaves all context-derived attention weights and gates unchanged. The physical input $r = \mathcal{C}_{c,\mathcal{P}} v$ is linear in v, and its lifts, weighted aggregation, and fixed gates are linear. Thus b and $h_0 = b$ are linear in v. Each response layer is linear in (h_ℓ, b), so induction gives linearity of every h_ℓ and the bias-free readout. GELU, LayerNorm and sigmoid operate on context features; softmax operates on context-derived attention scores. The response features undergo no activation, normalization or additive bias. □

D Experimental Details and Additional Results

D.1 Two-dimensional benchmarks and references

Physical fields. The PDEs are defined in Section 5; Appendix A gives their tangents. SP-D imposes the prescribed Dirichlet trace on the outer boundary and $u = 0$ on fixed hole boundaries. SP-DN imposes $u = g_D$ on Γ_D and $\partial_n u = g_N$ on Γ_N, where $\Gamma_D \cup \Gamma_N = \partial\Omega$ and n is the outward unit normal. VC-DR and SL-RD impose $u = 0$ on every boundary component, including holes. All four families use a three-Gaussian source

$$f(x) = a_0 + \sum_{j=1}^{3} A_j \exp\left[-\frac{\|x - \mu_j\|_2^2}{2\sigma_j^2}\right], \tag{23}$$

with $a_0 \in [-0.2, 0.2]$, $\sigma_j/D \in [0.06, 0.14]$, and $A_j \in [0.5, 1.5]$ ($[0.25, 1]$ for SL-RD), where D is the domain diameter. Centers are uniform in the domain with boundary clearance at least $0.15D$ and pairwise separation at least $0.12D$. Unless stated otherwise, scalar draws are uniform on the specified intervals.

SP-D/SP-DN use $\lambda \in [0.5, 4]$ and boundary Fourier modes 1–3 in normalized arclength s. Explicitly, with $s \in [0, 1)$ measured separately on each boundary component,

$$g_D(s) = c_0 + \sum_{k\in\mathcal{K}} \left[c_k^{\mathrm{c}} \cos(2\pi k s) + c_k^{\mathrm{s}} \sin(2\pi k s)\right],$$
$$g_N(s) = d_0 + \sum_{k\in\mathcal{K}} \left[d_k^{\mathrm{c}} \cos(2\pi k s) + d_k^{\mathrm{s}} \sin(2\pi k s)\right].$$

For SP-D, these Fourier data apply only to the outer boundary; hole data and their perturbations are zero. Here $\mathcal{K} = \{1, 2, 3\}$ in ID; the high-mode boundary-OOD trace uses $\mathcal{K} = \{4, 5\}$ with its constant term set to zero. Dirichlet constants lie in $[-0.3, 0.3]$ and sine/cosine coefficients in $[-0.15/k, 0.15/k]$ for mode k. Neumann coefficients use $[-0.20/k, 0.20/k]$ and the same constant range. SP-DN has one or two Neumann arcs occupying $[0.20, 0.55]$ of the outer boundary; holes are Dirichlet and the partition remains fixed during differentiation. VC-DR uses

$$\log\kappa = \sum_{j=1}^{6} \xi_j \phi_j, \qquad q = q_0 + \sum_{j=1}^{6} \zeta_j \phi_j, \tag{24}$$

where $\phi = (\cos\pi x, \sin\pi x, \cos\pi y, \sin\pi y, \cos\pi x\cos\pi y, \sin\pi x\sin\pi y)$, $\xi_j \in [-0.06, 0.06]$, $q_0 \in [1.5, 4]$, and $\zeta_j \in [-0.15, 0.15]$. A 41×41 lattice screens for $0.70 \le \kappa \le 1.43$ and $q \ge 0.5$. SL-RD uses $\lambda \in [0.5, 2]$ and $\gamma \in [0.25, 2]$.

Geometry and splits. Domains in $[-1, 1]^2$ mix Fourier, superellipse, and periodic cubic-spline boundaries in proportions 40%, 30%, and 30%, with 512 outer vertices. Geometry OOD equally mixes toothed domains (6, 8, or 10 teeth) and domains with one or two elliptical holes, each discretized by 128 vertices. Each context has eight directions, 512 interior queries, and 512 boundary queries, sampled uniformly by area and arclength, respectively. The model uses 512 interior and 512 boundary supports; all 1024 queries form one evaluation group. Table 7 gives the context counts. Figure 5 compares ID and geometry-OOD examples in 2D and 3D.

Table 7: 2D context counts. The last column is the count for each OOD split.

Family	Train	Validation	Test ID	Per OOD split
SP-D	4000	500	1000	500
SP-DN	4000	500	1000	500
VC-DR	1024	128	256	128
SL-RD	1024	128	256	128

Physical directions and shifts. The direction dimensions are 25, 28, 26, and 15 for SP-D, SP-DN, VC-DR, and SL-RD. All four families vary the source offset, amplitudes, centers, and log-widths in Equation 23. SP-D additionally varies λ, Dirichlet Fourier coefficients, and the four shape

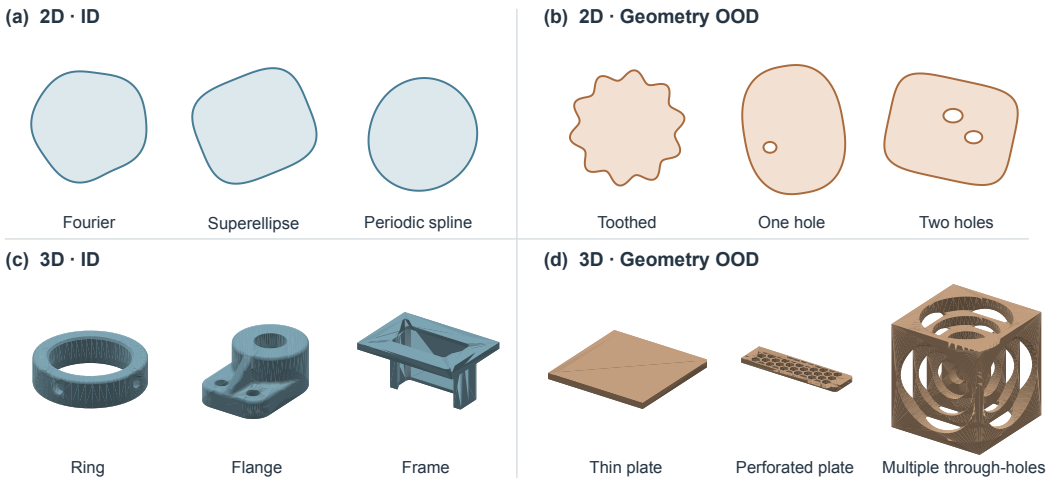

Figure 5: ID and geometry-OOD examples from the held-out test sets. (a) 2D ID domains use Fourier, superellipse, and periodic spline boundaries. (b) Geometry OOD adds toothed and perforated domains. (c,d) CAD-DR examples from the ID and geometry-OOD splits, with thin solids and more complex topology in the latter. Blue and orange distinguish the two splits.

coefficients described below; SP-DN varies λ and both Dirichlet and Neumann Fourier coefficients on a fixed geometry and partition. VC-DR varies (ξ_j, q_0, ζ_j), and SL-RD varies (λ, γ); their zero Dirichlet data remain fixed. ID directions mix normalized dense Gaussian, three-sparse, canonical-axis, and magnitude-paired draws in proportions 50%, 25%, 15%, and 10%. Magnitude pairs scale an existing dense direction by 0.5 or 1.5. Normalized directions z map to physical directions as $v = Sz$, without subsequent renormalization. Diagonal scales are half the ID sampling ranges, except for source centers (0.25 per coordinate) and shape coefficients ($0.02D$); log-width scales are $\frac{1}{2}\log(0.14/0.06)$. In 20% of eligible SP-D contexts, the base Dirichlet trace is zero and two direction slots include shape variations in the four sine/cosine modes $k = 1, 2$ of normal boundary velocity. Query locations remain fixed during differentiation. Table 8 summarizes the non-geometric shifts.

Table 8: 2D OOD shifts. Unspecified quantities retain their ID distributions. Each listed change is sampled equally often, up to rounding; direction OOD uses two directions of each type.

Split	Shift from ID
Parameter, SP-D/SP-DN	Four test cases: $\lambda \in [0, 0.2]$; $\lambda \in [5, 8]$; source amplitudes $[1.75, 2.5]$; source widths $\sigma/D \in [0.035, 0.05]$.
Boundary data, SP-D/SP-DN	Replace modes 1–3 by 4–5, or multiply coefficients by 1.75. SP-DN applies these changes separately to Dirichlet and Neumann data (four cases).
Boundary partition, SP-DN	Three Neumann arcs with total fraction $[0.25, 0.55]$, or one/two arcs with fraction $[0.60, 0.75]$. Holes remain Dirichlet.
Coefficient, VC-DR	Three test cases (43/43/42 contexts): $\xi_j \in [-0.09, 0.09]$; add $\cos 2\pi x, \sin 2\pi y$ to $\log \kappa$ with amplitudes $[-0.06, 0.06]$; or $q_0 \in [4.5, 6]$. ID acceptance bounds remain in force.
Nonlinearity, SL-RD	$\gamma \in [2.5, 4]$ or source amplitudes $[1.1, 1.5]$.
Direction	Four-sparse; balanced Rademacher; factors 0.25 and 2 on a common base-only Rademacher parent; equivalent-reparameterization carriers. The equivalent-reparameterization result reports aggregate prediction error.

Weak labels and references. Weak W4 targets average two independent two-path blocks, sharing the complete random realization across directions at each query. Shape estimates also require nested primal walks; deterministic Dirichlet traces require no walks. Coarse P2 states use $h/D = 0.08$, whereas physical tangent references use 0.02. The SP-D direction/geometry refinement protocol is specified below; the other archived refinement checks use 0.01. SL-RD solves the base nonlinear

problem by damped Newton iteration before linearization. The weak estimators and their numerical bias are specified in Appendices B.1 and A.3.

For both SP-D OOD splits, the reference solve imposes the complete Eulerian trace $\delta u = \delta g_D - V_n \partial_n u_0$ on the outer boundary and $u_0 = \delta u = 0$ on stationary holes. All 500 contexts in each split are evaluated, including all 250 perforated geometry-OOD domains. A preregistered 10% context sample compares $h/D = 0.08, 0.04, 0.02$. Between the last two resolutions, the 95th percentiles of context-level relative differences are 0.0018% for direction OOD and 0.0076% for geometry OOD, both below the 0.5% tolerance. All reported models use their original checkpoints and input representations; Raw W4 uses its archived estimates. No retraining or checkpoint reselection uses these OOD references.

D.2 CAD DIFFUSION–REACTION (CAD-DR)

We consider

$$-\nabla\cdot(\kappa\nabla u) + qu = f \quad \text{in } \Omega \subset \mathbb{R}^3, \qquad u = g \quad \text{on } \partial\Omega, \tag{25}$$

on irregular CAD geometries. As in VC-DR, we use the Liouville transformation $u = e^{-\ell/2}U$ with $\kappa = e^{\ell}$, under which the variable-coefficient equation can be written in terms of the transformed field U as

$$-\Delta U + VU = F, \qquad V = \frac{q}{\kappa} + Q_\ell, \qquad Q_\ell = \frac{1}{2}\Delta\ell + \frac{1}{4}\|\nabla\ell\|_2^2. \tag{26}$$

For the 3D CAD benchmark, we use this relation in reverse to construct manufactured problems with exact solution and tangent references. Specifically, for independent $\vartheta_1, \vartheta_2 \sim \mathcal{U}[-0.45, 0.45]$, we prescribe

$$U = U_0 + \vartheta_1 W_1 + \vartheta_2 W_2, \qquad \ell = \ell_0 + \vartheta_2 L_2, \qquad V = V_0 + \vartheta_2 Z_2, \tag{27}$$

and recover the physical coefficients and forcing through

$$\kappa = e^{\ell}, \qquad q = \kappa(V - Q_\ell), \qquad F = -\Delta U + VU, \qquad f = e^{\ell/2}F, \tag{28}$$

with

$$\begin{aligned} u &= e^{-\ell/2}U, \\ g(x) &= e^{-\ell(x)/2}\big[U_0(x) + \vartheta_1 W_1(x) + \vartheta_2 W_2(x)\big], \quad x \in \partial\Omega. \end{aligned} \tag{29}$$

This construction guarantees that u exactly satisfies the physical diffusion–reaction equation while providing analytic reference tangents. Writing $L_1 = 0$, differentiation with respect to ϑ_r gives

$$\delta_r u = e^{-\ell/2}\left(W_r - \frac{1}{2}UL_r\right), \qquad r \in \{1, 2\}. \tag{30}$$

Thus CAD-DR retains the same transformed diffusion–reaction structure as VC-DR, while avoiding an additional high-accuracy 3D tangent solve for ground-truth evaluation. The source/boundary direction (S/B, $r = 1$) fixes κ, q; the coefficient direction (Coef., $r = 2$) changes κ, q and also induces changes in f, g. The two directions vary ϑ_1 and ϑ_2, respectively, with geometry and the generating fields fixed. The fields $h \in \{U_0, W_1, W_2, \ell_0, L_2\}$ defining the boundary data have the form

$$h(x) = b_h + a_h^\top x + \sum_{j=1}^{m_h} c_{hj}\sin(k_{hj}^\top x + \varphi_{hj}) + d_h \exp\left[-\frac{\|x-\mu\|_2^2}{2\sigma^2}\right].$$

There are three low-frequency modes per field, except for L_2, which has two. Linear terms $a_h^\top x$ occur only in U_0, W_1, W_2, Gaussian terms only in U_0, W_1, and $b_{L_2} = 0$. This gives generally nonzero Dirichlet data on the entire CAD boundary. ID log-diffusivity harmonic magnitudes are $[0.025, 0.065]$, the potential offset is $[3, 4]$, and Gaussian widths are $[0.216, 0.630]$. A 13^3 lattice screens for $0.5 \le \kappa \le 2$, contrast at most 4, $q > 0$, and $0.1 \le V \le 8$.

Geometry, splits, and point sets. The benchmark contains 750 connected, watertight solids from ABC (Koch et al., 2019), centered and scaled to longest extent 1.8. To prevent geometry leakage, CAD models derived from the same source design or identified as duplicates are grouped together, yielding 736 disjoint geometry groups. All members of a group are assigned to the same split.

Table 9: CAD-DR splits. Parameter and joint OOD each contain 33 contrast, 33 localization, and 34 composition contexts.

Split	Geometries	Contexts
Training	500	1000
Validation	50	50
Test ID	100	200
Geometry OOD	100	200
Parameter OOD	100 training geometries	100
Joint OOD	100 geometry-OOD geometries	100

Geometry OOD selects extreme complexity, topology, thickness, aspect, and curvature descriptors. Table 9 gives the split sizes; test-ID geometries are unseen during training.

Stored training/evaluation contexts have 1024/4096 queries: half in the volume and half equally divided among four distance bands near the boundary, plus 1024 surface points. The bands group points by distance to the closest boundary and contract for thin solids. Each training exposure samples 256 queries and 256 interior/256 surface supports; point-set methods share the frozen sampling schedule. Evaluation uses all 4096 queries in fixed groups of 256, with supports fixed per context. This grouping matters because query self-attention couples predictions within each group.

Supervision and training. W10 labels average ten outer native parameter-DWoS paths per stochastic query, using the transformed equation and restoring the physical response as

$$\widehat{\delta_r u} = e^{-\ell/2}\left(\widehat{\partial_{\vartheta_r} U} - \tfrac{1}{2} L_r \widehat{U}\right). \tag{31}$$

The estimator uses majorant 8, boundary tolerance 0.0018, and at most 1024 sphere steps. The 3D CALTO has 140,640 parameters and the architecture in Appendix C.2. CALTO-3D trains for 25,000 updates with batch size two, AdamW learning rate 0.005, weight decay 10^{-4}, and cosine decay to 5% of the initial rate. FNO+AD uses a 24^3 grid and W10 primal-only supervision for the same update budget.

Physical shifts and scope. To evaluate out-of-distribution (OOD) robustness, we construct three types of physical distribution shifts from the in-distribution (ID) contexts:

(i) *Contrast OOD* amplifies log-diffusivity harmonics to $[0.22, 0.30]$ and potential offsets to $[4.5, 5.5]$, enforcing medium contrasts > 4 with $\kappa \in [1/3, 3]$ and $q > 0$;

(ii) *Localization OOD* contracts Gaussian source widths to $[0.108, 0.216]$; and

(iii) *Composition OOD* fixes Fourier mode directions while expanding amplitude-phase ratios to 3–4 (versus ≤ 2.4 in ID).

To decouple distributional shift from trivial energy scaling, each OOD draw is paired with an ID reference: the solution RMS and secondary tangent coefficients are strictly matched, with samples retained only if field-covariate norm ratios across lattice and CAD meshes satisfy predetermined acceptance bounds.

Scope and limitations. Critically, Geometry OOD relies on manufactured solutions: restricting a common global analytic field to trimmed CAD boundaries leaves the interior ground-truth solution unchanged at overlapping coordinates. Consequently, this benchmark tests a model's adaptability to irregular CAD discretizations and tangent boundary conditions, rather than true geometry-induced alterations of the underlying physics. Furthermore, because supervision is restricted to two parameter coordinates, this does not validate high-dimensional recovery ($R \ll d$) in 3D. The 2D and 3D models are trained independently.

D.3 Learning settings

CALTO uses AdamW with initial learning rate 0.005 and weight decay 10^{-4}. The 2D runs use 100 epochs with batch size 32: 12,500 updates for SP-D/SP-DN and 3,200 for VC-DR/SL-RD. CAD-DR uses 25,000 updates. All experiments were conducted on an Nvidia RTX 4090D.

SC-FNO, DE-DeepONet, and DINO obtain JVPs by differentiating state surrogates; their losses compare these JVPs with fixed weak JVP targets. GINO, Transolver, and Transolver++ predict tangent responses directly using the same targets. All six tangent-prediction baselines therefore use JVP-only supervision. FNO+AD uses primal supervision. The state-surrogate JVPs are linear in the perturbation direction. They differentiate the normalized physical parameters while holding geometry, query locations, D/N partitions, and context fixed. For SL-RD, all methods receive the same coarse state $\widetilde{u}_0$ as fixed context. Strong validation responses select checkpoints. Table 10 lists the model capacities.

Table 10: Trainable parameter-tensor elements (thousands) for the benchmark-specific model configurations. Each complex entry counts once, corresponding to two real scalars. Input-channel dimensions vary by PDE family.

Method	SP-D	SP-DN	VC-DR	SL-RD
FNO+AD	342.6	344.4	342.8	342.2
SC-FNO	108.6	109.3	108.6	108.7
DE-DeepONet	250.3	254.9	250.7	773.3
DINO	30.7	30.7	30.7	30.7
GINO	1380.1	1380.2	1380.1	1380.0
Transolver	756.5	766.5	757.5	751.9
Transolver++	633.6	643.6	634.6	629.0
CALTO	142.5	142.8	142.9	141.6

For R^2, let y_j and p_j denote the reference and predicted responses, respectively, where j indexes all query–direction pairs across all contexts in a split. Let $\bar{y}$ be the mean of all y_j:

$$R^2 = 1 - \frac{\sum_j (p_j - y_j)^2}{\sum_j (y_j - \bar{y})^2}. \tag{32}$$

D.4 Detailed two-dimensional OOD results

Table 11 gives all values underlying Figure 2: nine methods across 15 OOD splits.

D.5 Representative OOD field visualizations

Figures 6–8 complement the aggregate OOD results with tangent-response fields and spatial error maps for individual contexts. Figure 6 shows one case for each benchmark's representative OOD split, and Figure 7 shows one case for each benchmark with Geometry OOD. Figure 8 follows the main 3D taxonomy of Geometry, Parameter, and Joint OOD, which describe context shifts independently of the tangent direction (Appendix D.2). All four columns show coefficient tangents. In Figures 6 and 8, error colors show pointwise absolute error divided by the reference field's RMS over the case's queries, in percent; the numbers beneath the maps give L2RE over those queries.

Table 11: Detailed two-dimensional OOD results underlying Figure 2. Metrics are reported in context-averaged L2RE (%).

SP-D				
Method	Parameters	Boundary data	Directions	Geometry
CALTO	1.20	0.72	0.65	21.11
SC-FNO	5.88	6.54	4.96	29.06
DE-DeepONet	2.84	3.49	1.95	21.36
DINO	6.04	5.58	5.13	22.23
GINO	9.65	8.94	10.08	25.75
Transolver	2.33	1.57	1.90	21.98
Transolver++	2.94	2.05	2.41	22.73
FNO+AD	24.04	63.39	35.27	44.46
Raw W4	16.02	15.61	14.20	25.46

SP-DN					
Method	Parameters	Boundary data	D/N partition	Directions	Geometry
CALTO	4.59	3.24	4.46	3.12	17.73
SC-FNO	16.82	14.88	30.46	14.85	30.09
DE-DeepONet	10.03	9.06	15.99	8.60	23.14
DINO	16.20	14.43	21.02	14.32	29.09
GINO	24.45	21.50	27.12	24.16	36.17
Transolver	8.88	6.70	10.22	8.67	20.79
Transolver++	8.83	6.67	10.32	8.84	20.52
FNO+AD	42.48	41.95	45.45	38.19	47.16
Raw W4	44.84	44.43	53.33	41.98	40.63

VC-DR			
Method	Coefficients	Directions	Geometry
CALTO	9.29	8.20	30.72
SC-FNO	29.61	30.65	66.58
DE-DeepONet	28.56	27.50	62.14
DINO	37.29	36.60	73.30
GINO	25.07	29.60	56.69
Transolver	23.86	27.81	48.02
Transolver++	25.16	28.79	48.34
FNO+AD	61.29	61.32	88.72
Raw W4	82.94	79.69	77.93

SL-RD			
Method	Nonlinearity	Directions	Geometry
CALTO	11.59	8.20	41.74
SC-FNO	28.98	26.97	73.49
DE-DeepONet	27.80	23.02	57.66
DINO	55.84	50.66	86.14
GINO	25.09	24.18	77.89
Transolver	25.25	24.51	44.04
Transolver++	25.47	24.79	44.72
FNO+AD	92.62	91.31	88.63
Raw W4	85.53	74.85	71.97

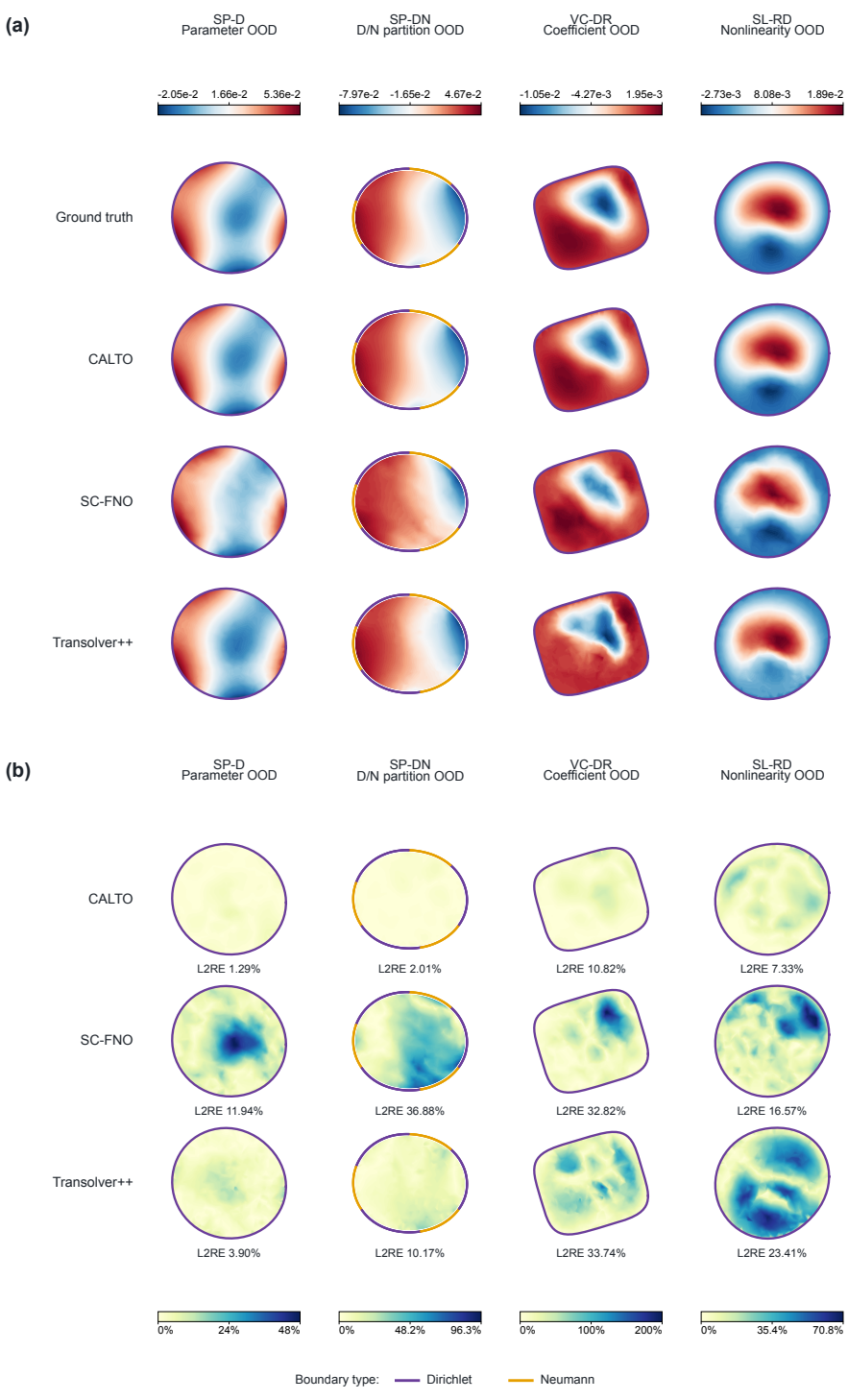

Figure 6: Representative 2D OOD tangent-response fields, with one selected physical direction per case. Within each column, all methods share the context, queries, and direction. Columns show SP-D parameter OOD, SP-DN D/N-partition OOD, VC-DR coefficient OOD, and SL-RD nonlinearity OOD. (a) Ground truth and predictions from CALTO, SC-FNO, and Transolver++. (b) Normalized pointwise errors, with query-set L2RE beneath each map (Appendix D.5). Each benchmark has its own complete field-value and error scales. Purple and orange denote Dirichlet and Neumann boundary respectively.

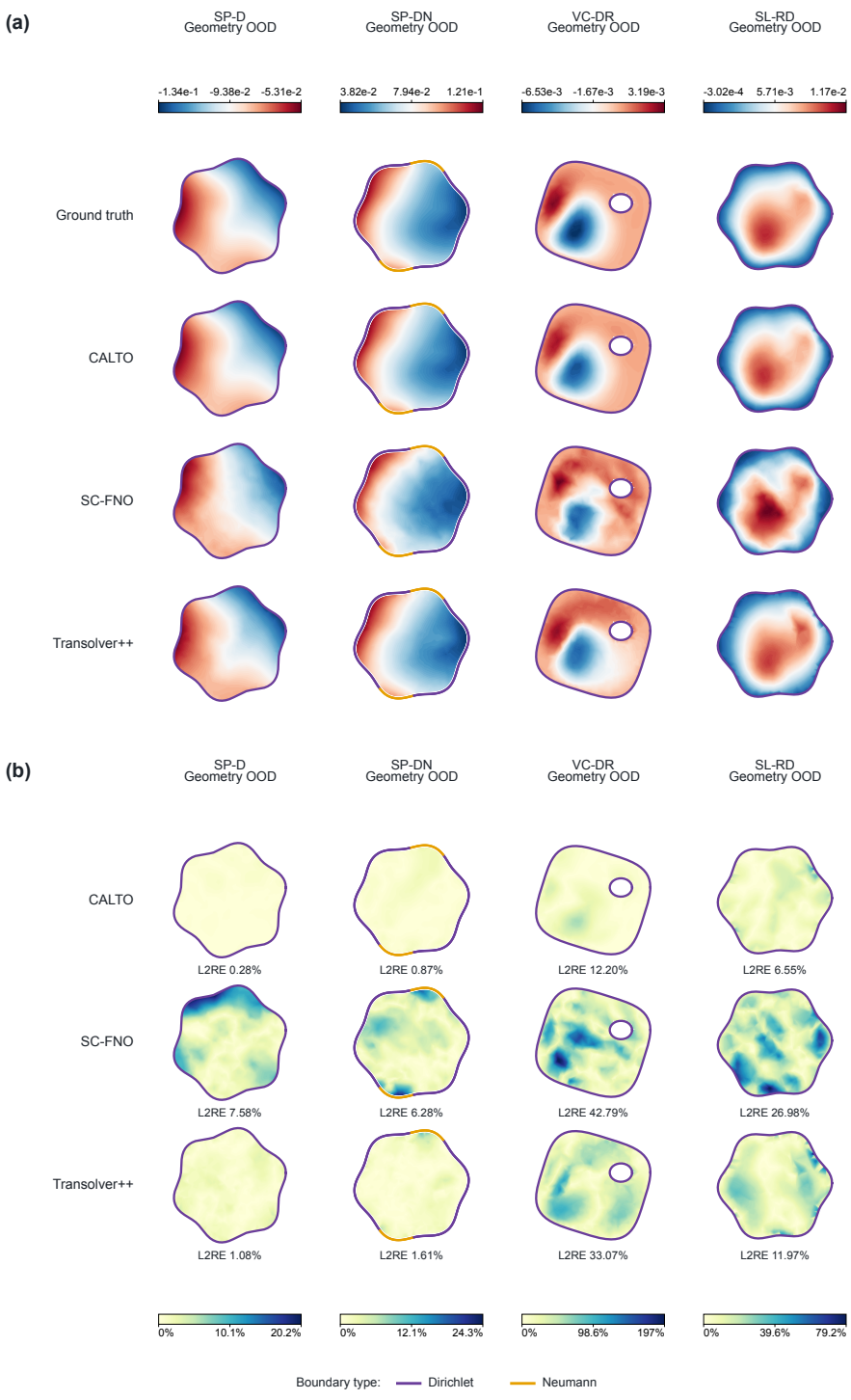

Figure 7: Two-dimensional Geometry OOD tangent-response fields: toothed domains for SP-D, SP-DN, and SL-RD, and a perforated domain for VC-DR.

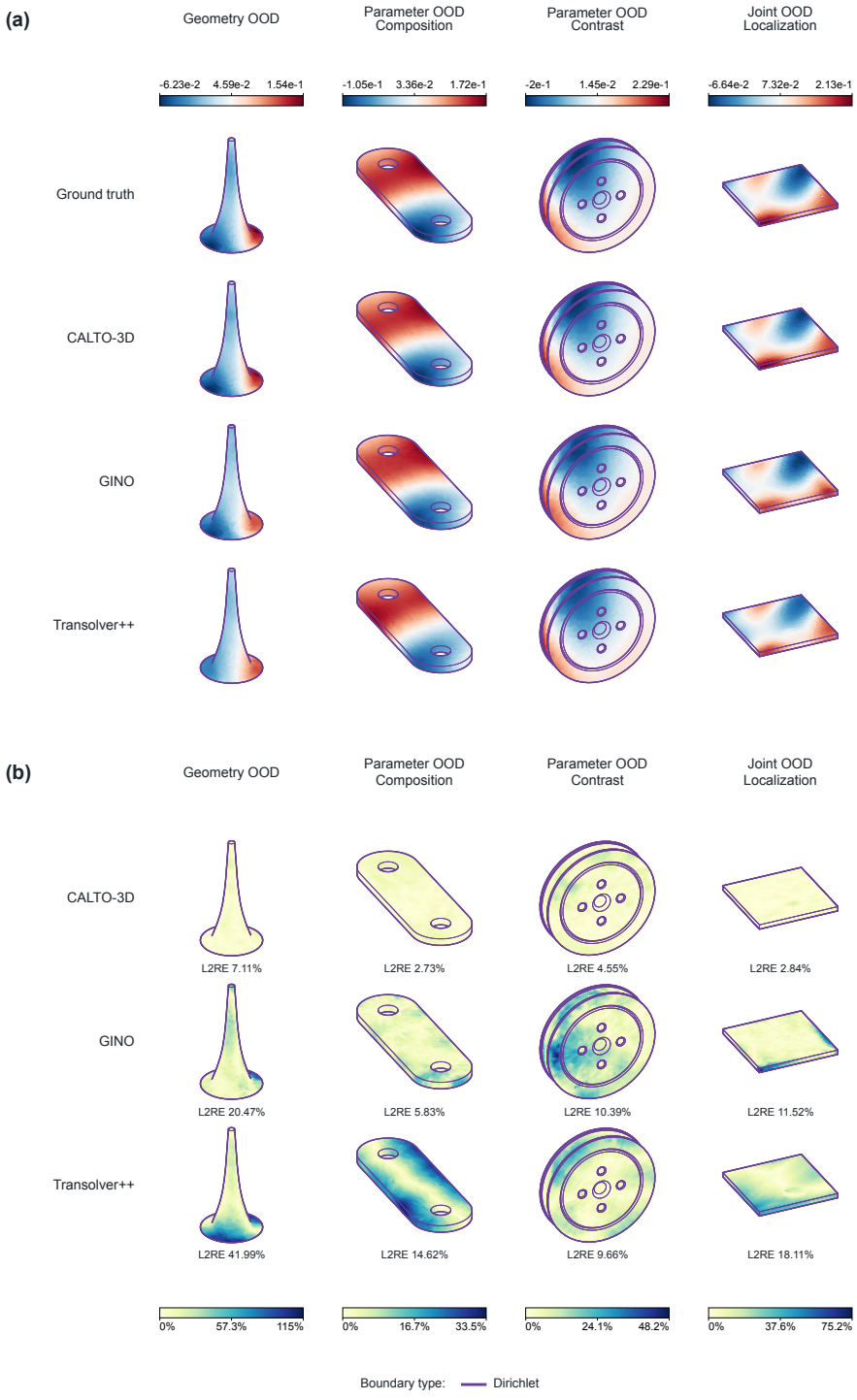

Figure 8: Representative 3D OOD tangent-response fields. Columns show Geometry OOD, Parameter OOD with a composition shift, Parameter OOD with a contrast shift, and Joint OOD with a localization shift. The two Parameter OOD columns use different contexts; all four columns show coefficient tangents. (a) Ground truth and predictions from CALTO-3D, GINO, and Transolver++. (b) Normalized pointwise errors, with query-set L2RE beneath each map (Appendix D.5)

.

D.6 SUPERVISION SENSITIVITY

The SP-DN study varies one of $(N, R, W) = (4000, 8, 4)$ at a time, or replaces weak labels with P2 tangent responses. All runs use 100 epochs, AdamW with initial learning rate 0.005 and weight decay 10^{-4}, cosine decay, and batch size 32. Table 12 gives the values underlying Figure 4.

Table 12: SP-DN supervision sensitivity for the point-set CALTO model in Figure 4. Entries are context-averaged L2RE (%) on ID and direction-OOD splits. Each row changes one quantity from $(N, R, W) = (4000, 8, 4)$; P2 replaces weak labels with strong tangent responses.

Contexts N	Directions R	Supervision	ID	Direction OOD
4,000	8	W4	3.03	3.12
4,000	8	W8	2.59	2.70
4,000	8	W16	2.31	2.38
1,000	8	W4	4.83	5.07
200	8	W4	11.86	12.61
4,000	2	W4	4.06	4.29
4,000	4	W4	3.43	3.70
4,000	8	P2	1.89	1.97

D.7 STRUCTURAL ABLATIONS

We test four design choices: global attention, attention weights computed jointly from query and source contexts, successive response layers, and gates conditioned on physical data. Table 13 reports four controls for the model described in Appendix C.2. Each control retains exact directional linearity and matches the full model's parameter count, width, attention heads, and number of response blocks within each benchmark.

All variants use the same W4 supervision, eight directions, 1024 queries, and 512 supports each in the interior and on the boundary. We trained each variant for 100 epochs with batch size 32, and the same AdamW schedule, minimizing weak-target L2RE and selecting the checkpoint with the lowest strong-validation L2RE.

Table 13: Structural ablations for CALTO, metrics are reported in L2RE (%). All ablations preserve directional linearity and the parameter count of the full model within each benchmark.

Model	SP-D	SP-DN	VC-DR	SL-RD
CALTO	**0.71**	**3.03**	**8.88**	**8.30**
32-neighbor attention	1.48	15.94	10.78	11.18
Separable global kernel	1.07	4.04	12.83	11.08
Parallel response blocks	0.91	3.64	11.67	10.29
Geometry-only modulation	0.77	3.69	9.83	9.25

The 32-neighbor attention control limits each query to its 32 nearest source points by Euclidean distance and normalizes attention weights within that neighborhood. This applies to both support-to-query and query-to-query attention, testing the benefit of directly aggregating distant responses at the same network depth. Error increased on every benchmark: L2RE rose from 3.03% to 15.94% on SP-DN, the largest increase, and from 0.71% to 1.48% on SP-D. These results support global aggregation at the depth used here.

The separable global kernel keeps every query connected to all sources. For each head and channel, source contexts determine one weighted summary of the source responses, which every query scales by a gain from its own context. The relative source weights are therefore shared across queries within each channel. This tests whether aggregation benefits from weights that depend jointly on each query and source. CALTO reduced L2RE by 25.0% to 33.5% relative to this control across the four benchmarks, supporting this joint dependence over shared global summaries.

To test the value of successive updates, the parallel response blocks all act independently on the initial response b. Each block uses b as both its current state and its reinjected input; their outputs are

averaged before the common readout. This preserves the four blocks and their internal operations while removing the transfer of updated states between them. CALTO reduced error by 16.8% to 23.9% relative to this control, supporting the benefit of refining already aggregated responses over successive layers.

Geometry-only modulation generates gates through the same context encoders, using only coordinates, normals, boundary position and boundary membership. Physical fields and parameters are removed from the gate inputs, and boundary-condition types are merged into a single boundary indicator. Attention still uses the full PDE context, and perturbation inputs are unchanged. This tests whether physical information improves gating beyond its use in attention. Geometry-only gates increased SP-DN L2RE from 3.03% to 3.69%. Across all four benchmarks, full physical conditioning reduced error by 7.5% to 18.0%. These gains show that gates benefit from physical information even when attention already uses it.

D.8 Full-parameter Jacobians and VJPs

Table 4 uses the first 128 ID contexts per 2D family, with 1,024 queries per context. Each Jacobian has one column per parameter, and each VJP has one entry per parameter. SP-D has 21 parameters on a fixed geometry, plus four shape parameters in 22 contexts; SP-DN, VC-DR, and SL-RD have 28, 26, and 15 parameters, respectively. The reference is the P2 tangent Jacobian at $h/D = 0.02$.

VJPs are computed by matrix transposition as in Equation 9. Each context uses four Rademacher weights and four signed Gaussian weights with spatial width $\sigma = 0.25$. All weights have unit Euclidean norm and are shared across methods. Prediction and reference use the parameter scales in Appendix D.1, with $K_i^\star = J_{i,X} S_i$. For $N = 128$ contexts, the reported errors are

$$E_{\mathrm{Jac}} = \frac{1}{N} \sum_{i=1}^{N} \frac{\|\widehat{K}_i - K_i^\star\|_F}{\max(\|K_i^\star\|_F, 10^{-24})},$$

$$E_{\mathrm{VJP}} = \frac{1}{8N} \sum_{i=1}^{N} \sum_{k=1}^{8} \frac{\|\widehat{K}_i^\top w_{ik} - (K_i^\star)^\top w_{ik}\|_2}{\max(\|(K_i^\star)^\top w_{ik}\|_2, 10^{-24})}.$$

The physical-parameter Jacobian is recovered as $\widehat{J}_i = \widehat{K}_i S_i^{-1}$, with VJP $\widehat{J}_i^\top w$.

D.9 Full-parameter derivative cost

Table 14 reports mean full-parameter GPU timings from prepared device inputs, including context encoding for the learned methods. Matrix construction and reuse for eight VJPs are timed separately.

Table 14: Full-Jacobian construction / eight cached VJPs (ms) on an RTX 4090 D. Entries are means over four fixed contexts per family and ten synchronized repetitions per context.

Family	CALTO	SC-FNO	Raw W4
SP-D	2.13 / 0.014	10.19 / 0.014	19.46 / 0.026
SP-DN	2.19 / 0.018	16.86 / 0.017	31.48 / 0.021
VC-DR	2.11 / 0.014	9.86 / 0.014	8.07 / 0.017
SL-RD	2.27 / 0.014	10.81 / 0.013	4.10 / 0.017

D.10 DIFFUSION-COEFFICIENT IDENTIFICATION

The experiments in Section 5.4 recover the six diffusivity coefficients in Equation 24, with the source, reaction coefficient, domain, and homogeneous Dirichlet data known. We optimize $z = \xi/0.06$ within $[-1, 1]^6$, subject to the coefficient constraints in Appendix D.1. Each of the 16 VC-DR validation instances uses two initializations: zero coefficients and a feasible random vector. Observations and forward evaluations use P2 FEM with $h/D = 0.01$. Of the 512 interior queries, 128 enter the least-squares objective and 384 evaluate reconstruction.

SC-FNO, DE-DeepONet, and CALTO-Hybrid use equally weighted context-averaged relative-error losses for weak primal values and JVPs. CALTO-Hybrid shares its context-dependent layers between a primal output and a direction-linear JVP output. The standalone CALTO uses JVP supervision. Jointly trained models are selected by the sum of primal and JVP validation errors and remain fixed during inversion. FEM tangent derivatives provide the reference Jacobian through six discrete tangent solves.

Inversion uses damped Gauss–Newton steps with a trust region and backtracking to satisfy the coefficient constraints. For surrogate tangent correction, Equation 10 defines the local sensor-response model. Value and derivative corrections remain fixed during each local solve and are recomputed after an accepted step. Step acceptance requires a decrease in the FEM objective. Results use the last accepted estimate within a budget of 16 FEM forward evaluations, including the initial state and rejected trials.

Table 15 reports reconstruction at the 384 unused points and parameter recovery. The unused-point residual is normalized by its initial value; parameter error is $\varepsilon_\xi = \|\xi - \xi^\star\|_2/(0.06\sqrt{6})$. CALTO tangent correction reduces both errors for each primal surrogate. All 32 inversions for each corrected surrogate reach $E_{16} \le 0.1$, compared with 21 for SC-FNO and 6 for DE-DeepONet under value correction alone.

Table 15: Reconstruction after at most 16 FEM forward evaluations. Entries are means over 16 validation instances and two initializations per instance. Both errors are lower-is-better; correction labels follow Table 5.

Surrogate	Correction	Unused-point residual ratio	ε_ξ
SC-FNO	V	1.64×10^{-1}	2.00×10^{-1}
	V+CALTO	1.43×10^{-3}	6.06×10^{-3}
	V+FEM	4.00×10^{-6}	5.48×10^{-6}
DE-DeepONet	V	3.31×10^{-1}	4.16×10^{-1}
	V+CALTO	1.31×10^{-3}	5.62×10^{-3}
	V+FEM	2.54×10^{-6}	3.52×10^{-6}

D.11 FIXED-GEOMETRY INVERSE CONTROL

These experiments extend the action-derivative tests to repeated control updates on fixed domains. For physical controls α, CALTO predicts the response columns $\widehat{A}_{\alpha,X_\mathrm{d}}$ at design points X_d. At each accepted state, a ridge step fits the target-field residual:

$$\Delta\alpha = \arg\min_z \left\| \widehat{A}_{\alpha,X_\mathrm{d}} z - [y^\star - P_{X_\mathrm{d}} u(\alpha)] \right\|_2^2 + \lambda_\mathrm{r} \|z\|_2^2. \tag{33}$$

Step limits and forward-solve backtracking enforce decreasing design error; context and responses are recomputed after each update. Disjoint observations measure held-out RMSE. The displayed final fields come from forward evaluation. These fixed-domain examples assess the prescribed control families. The optimizer uses target observations; CALTO receives the current PDE context and control directions.

2D Dirichlet control. Three independent screened-Poisson solutions form a continuous three-channel field on each fixed domain. Each channel has seven Dirichlet Fourier coefficients (a constant and sine/cosine modes 1–3), giving 21 controls. Source and screening remain fixed; optimization starts from a constant trace and targets a nonconstant trace within this family. Figures 9 and 10

show six domains. Held-out RMSE decreases from 0.183–0.215 to 5.95×10^{-6}–4.22×10^{-5}. Field panels show the channel mean, while optimization and errors use all three channels. Held-out RMSE averages squared physical-value errors over 480 disjoint observation points and all three channels before taking the square root.

3D mixed physical control. Six selected OOD CAD/PDE contexts use the two registered directions in Appendix D.2. The controls jointly vary source, Dirichlet, diffusion, and reaction data; geometry remains fixed. Targets lie within this two-coordinate family, and initial controls extend beyond the training amplitude range, with all-volume initial RMSE set to one. A common target-independent admissible set bounds the controls. CALTO supplies design-volume responses, the exact manufactured forward field accepts steps, and disjoint volume points measure RMSE (Figures 11 and 12). Final held-out RMSE is 1.11×10^{-6}–8.93×10^{-6}. These experiments evaluate repeated, derivative-assisted updates within the prescribed control families on fixed geometries.

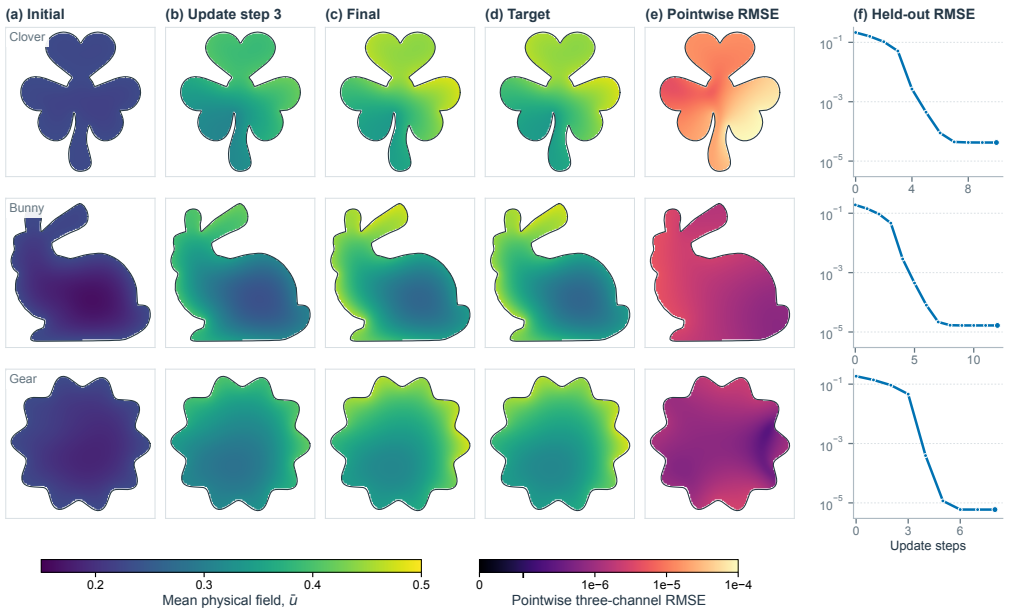

Figure 9: **2D Dirichlet control on complex domains.** Rows: clover, bunny silhouette, and gear. Columns: (a) initial field, (b) third accepted update, (c) forward-evaluated optimized field, (d) target, (e) pointwise three-channel RMSE $[\frac{1}{3}\sum_{a=1}^{3}(u_a - u_a^\star)^2]^{1/2}$, and (f) held-out RMSE. Here u_a and $u_a^\star$ are the final and target physical solution values in channel a. Field panels show the channel mean only; optimization and errors use all three channels. Geometry is fixed; 21 Fourier coefficients are optimized.

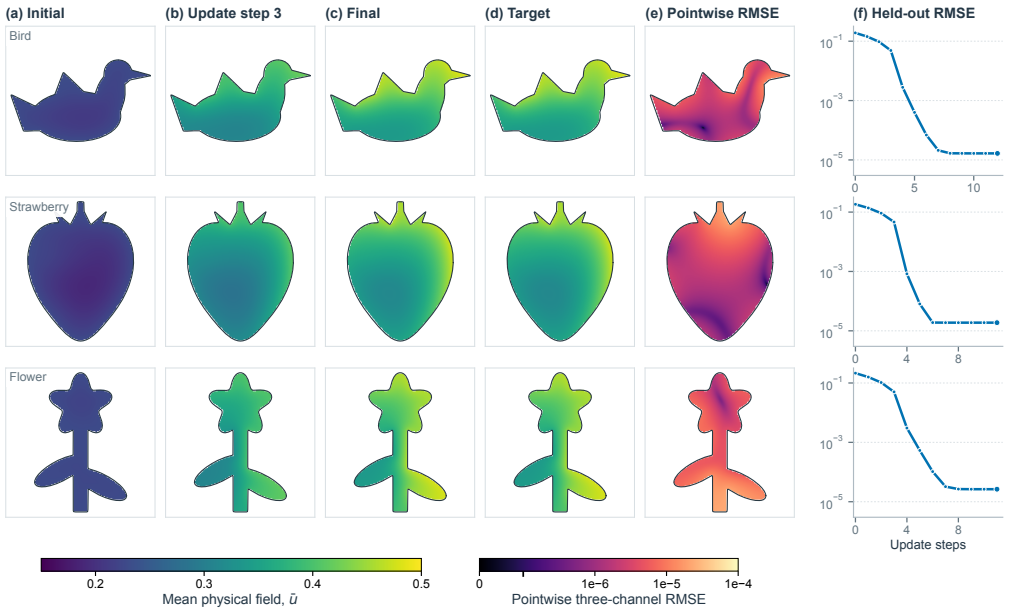

Figure 10: **2D Dirichlet control on silhouette domains.** Rows: bird, strawberry, and flower. Columns show the initial field, third update, final field, target, pointwise three-channel RMSE, and held-out RMSE, as in Figure 9. All three channels enter optimization and error evaluation; geometry and interior PDE data remain fixed.

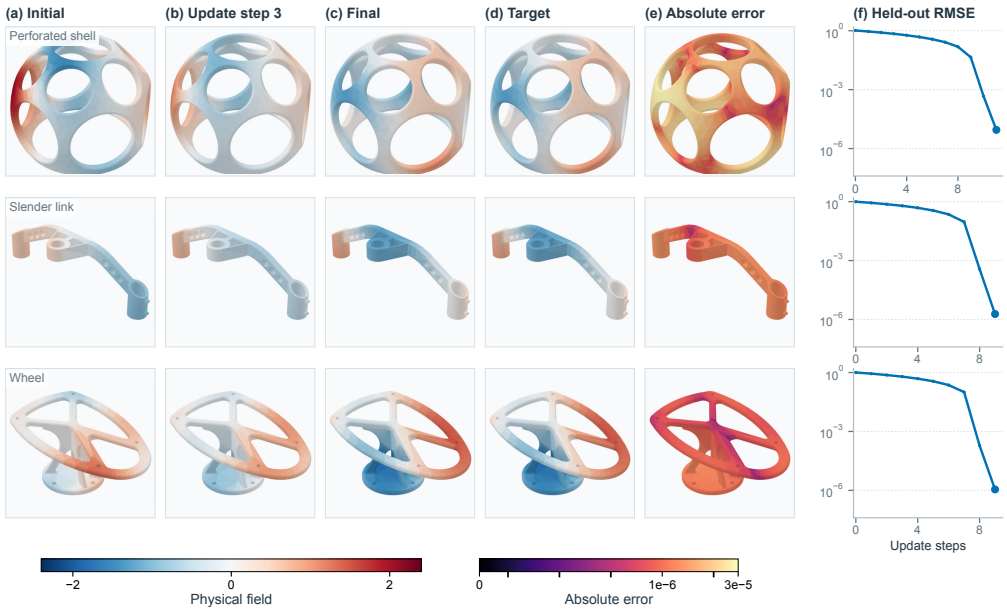

Figure 11: **Fixed-geometry control on 3D OOD CAD domains.** Rows: perforated shell, slender link, and wheel. Columns: initial surface field, third accepted update, forward-evaluated optimized field, exact target, absolute surface error, and held-out volume RMSE. Initial all-volume RMSE is one. The two physical controls include source, boundary, diffusion, and reaction data.

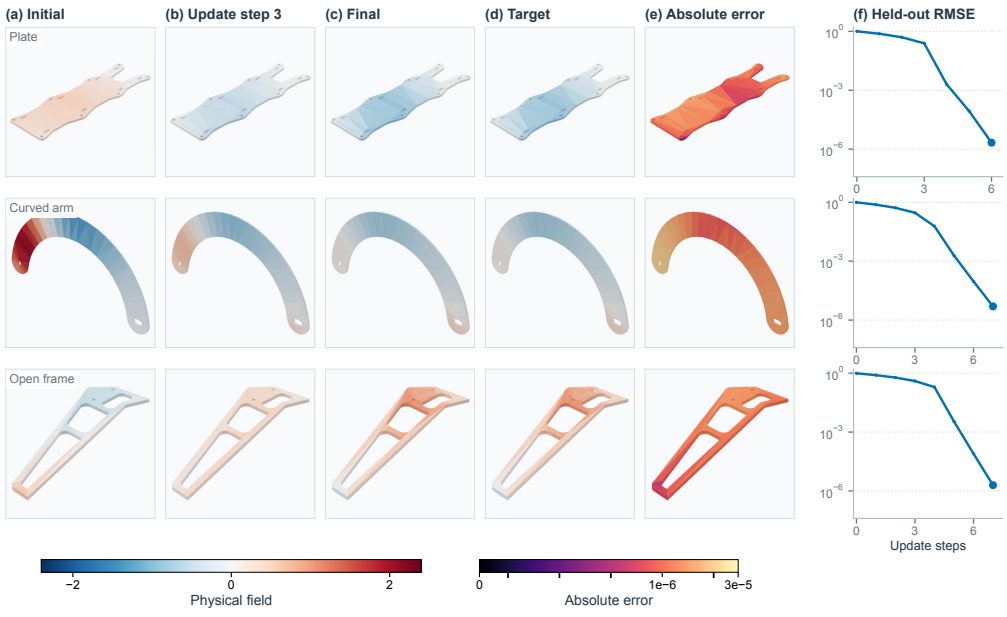

Figure 12: **Fixed-geometry control on high-response 3D OOD CAD domains.** Rows: plate, curved arm, and open frame. CALTO supplies tangent responses and the exact manufactured forward field accepts steps. Targets are attainable within the registered two-coordinate family, using common target-independent control bounds.