Neural Boundary Integral Operators: Solver-Consistent Learning Across Geometry, Resolution, and Kernel Mismatch*

Anonymous authors
Paper under double-blind review

Abstract

Black-box neural operators such as FNO and DeepONet learn PDE solution maps from data, but they often struggle on irregular geometry, do not encode known singular operator structure, and do not natively handle unbounded domains without truncation, mapping, or specialized exterior treatment. We introduce the *Neural Boundary Integral Operator* (NBIO), a hybrid neural operator that embeds a boundary integral equation (BIE) and restricts learning to smooth model discrepancy. NBIO combines: (M1) an *analytically split kernel*, consisting of a known singular Green's kernel and a learned smooth correction; (M2) a *differentiable second-kind solve* for the boundary density; and (M3) discretization-independent continuous evaluation in boundary arc length and query location. This construction supports zero-shot transfer across geometry and resolution and across interior/exterior regimes in 2D; a jointly trained 3D proof of concept covers both regimes on deformed spheres. In the zero-correction limit, NBIO recovers a spectrally accurate boundary element solver on 2D Laplace problems (rel-$L_2 = 4.8 \times 10^{-7}$), while learned black-box baselines on the same data remain near 10^{-2}–10^{-1} and do not improve under boundary refinement. When the available classical prior is misspecified or incomplete, learning becomes useful: on a modified-Helmholtz task with a Laplace prior, the learned correction reduces error from 1.13 to 8.3×10^{-3}; on a variable-coefficient problem it improves a fixed prior from 9.0×10^{-2} to 3.4×10^{-3}, although its edge over a strong FNO is modest. NBIO also extends to interface problems with prescribed jumps and volume sources. Its zero-parameter structured limit matches the reference solver and is substantially more accurate than same-protocol black-box operators. A differentiable QBX module resolves near-boundary and near-touching evaluation. On real ABC and Fusion 360 CAD meshes, the zero-parameter BIE structure achieves median errors of 2.1×10^{-3} and 5.6×10^{-3}, respectively, 100–200× lower than same-protocol learned geometry operators, while a learned correction does not improve the known-kernel CAD solver. Overall, NBIO shows that exact numerical structure can provide solver consistency and geometry–resolution transfer, while learning is most valuable when the classical model is incomplete.

1 Introduction

Operator learning aims to approximate solution maps of PDEs, $\mathcal{G} : a \mapsto u$, from data, enabling fast surrogate solves across a family of inputs. Fourier Neural Operators (Li et al., 2021) and DeepONet (Lu et al., 2021) are the dominant paradigms, with a now-substantial theory of neural operators between function spaces (Kovachki et al., 2023). These are powerful but *black-box*: they learn a volumetric map on a (near-)regular grid, do not use the structure of the differential operator, staircase on curved boundaries, and represent unbounded exterior domains only through truncation, mapping, or a separate exterior treatment. Variants

*Large language models were used as assistive tools for manuscript editing, code review, and experiment-planning support. The authors designed and verified all methods, experiments, results, and claims, and take full responsibility for the content.

address geometry through graph kernels (Li et al., 2020), learned deformations (Li et al., 2023a), or 3D geometry encodings (Li et al., 2023b), but the volumetric, structure-agnostic nature remains.

For *linear elliptic* PDEs, classical numerical analysis offers a powerful alternative: the *boundary integral equation* (BIE) reformulation (Atkinson, 1997; Kress, 2014; Hsiao & Wendland, 2008). Using the Green's function, the interior solution is represented by an integral of an unknown density over the $(d-1)$-dimensional boundary; the density solves a well-conditioned second-kind Fredholm equation. BIE methods are dimension-reduced, spectrally accurate (Nyström (Nyström, 1930) with Kress quadrature (Kress, 1991)), and exact for exterior problems. This structure is absent from black-box operators.

Contributions. NBIO is a hybrid operator architecture that reduces to a classical boundary-integral solver when the governing kernel is known and uses learning for smooth model discrepancy when the kernel is misspecified or unknown. It assembles the known singular boundary operator directly and restricts data-driven capacity to a bounded smooth correction. This construction gives quadrature-consistent generalization across geometry and resolution, and across interior/exterior regimes in 2D, together with the sufficient-condition stability statement in Prop. 1.

We evaluate six aspects of the method. (i) On 2D Laplace, NBIO reaches rel-$L_2 \sim 10^{-7}$ and is 2.5×10^4 more accurate than the strongest same-data geometry-aware baseline (up to $10^5\times$ over simpler black-box operators), with zero-shot resolution transfer. (ii) Wrong-kernel and variable-coefficient tests identify when the learned correction is necessary. (iii) The formulation extends to piecewise-constant transmission, 3D surfaces, and interface problems with prescribed jumps and volume sources. (iv) An in-operator QBX module provides differentiable close evaluation at continuous query locations. (v) A near-linear $\mathcal{H}$-matrix + GMRES solver, validated to $n_b = 65{,}536$, removes the dense $O(n_b^3)$ solve; a rank-4 separable correction retains full solution accuracy in the known-kernel, low-frequency regime. For oscillatory Helmholtz, the off-diagonal rank instead grows approximately linearly with wavenumber. (vi) Used zero-shot inside a Picard iteration, NBIO solves a semilinear elliptic BVP to classical-solver accuracy, whereas a black-box operator's per-step error limits the outer solve. The largest gains occur in the BIE-native regime; on volumetric variable-coefficient problems, the improvement over FNO is modest.

2 Background and Related Work

Neural operators. FNO (Li et al., 2021), DeepONet (Lu et al., 2021), GNO (Li et al., 2020), Geo-FNO (Li et al., 2023a), GINO (Li et al., 2023b); theory in Kovachki et al. (2023); fair-comparison practice in Lu et al. (2022). PINNs (Raissi et al., 2019) solve single instances rather than the operator. **Learning Green's functions** is related but distinct: Boullé et al. (2022); Gin et al. (2021) learn a global Green's function from data, whereas NBIO uses the known singular Green's kernel as structure and learns only a smooth correction and the cross-geometry generalization. **Boundary integral methods**: second-kind theory (Atkinson, 1997; Kress, 2014; Hsiao & Wendland, 2008), logarithmic (Kress) quadrature (Kress, 1991), fast evaluation (Greengard & Rokhlin, 1987), and close/near-singular evaluation by Quadrature-by-Expansion (Klöckner et al., 2013), which we make differentiable and embed in the operator. Our strongest black-box baseline is an attention-based boundary operator in the spirit of Vaswani et al. (2017).

Boundary- and interface-aware neural operators. Several recent operators target boundaries or interfaces directly. IONet (Wu et al., 2024) and XI-DeepONet (Bi et al., 2025) use domain decomposition or level-set/interface conditioning to learn elliptic *interface* solution operators, reporting rel-$L_2 \sim 10^{-3}$–10^{-2} across 2D–6D with volume sources and interface jumps; the concurrent ϕ-DeepONet (Roy et al., 2026) uses a multi-branch DeepONet with nonlinear latent interface embedding, achieving rel-$L_2 \approx 2.7 \times 10^{-3}$ on 2D interface problems (comparable to IONet, and two orders better than standard DeepONet). BENO (Wang et al., 2024) embeds boundary values and geometry into a graph/message-passing operator for elliptic PDEs with body forcing on irregular corner domains.

The closest concurrent work to NBIO is in the BIE-learning subspace. LOB (Fang et al., 2024) reformulates the PDE as a BIE and trains a physics-informed operator from boundary points only, reporting $\approx 6 \times 10^{-2}$ on 2D parametric Laplace. MAD-BNO (Wu et al., 2026a) further trains a boundary operator using only

data synthesized from fundamental solutions (no solver required), reaching rel-$L_2 \approx 6 \times 10^{-3}$ on 2D Laplace; the BIE evaluation step recovers interior from boundary. BI-TDONet (Meng et al., 2024) uses SVD of the second-kind BIE operator to learn interior-to-boundary-to-interior maps, reporting rel-$L_2 \approx 4$–7×10^{-4} MRE on interior/exterior Dirichlet problems with a single training. BINO (Wu et al., 2026b) learns a geometry-aware Green's traction kernel for mesh deformation (different application domain: linear elasticity). Mousavi et al. (2026) study boundary-condition encodings for arbitrary backbones. Most directly related to our contraction and nonlinear-BIE claims, the concurrent FREDINO (Georgiou et al., 2026) learns *provably contractive* second-kind Fredholm integral operators and applies them to nonlinear elliptic PDEs via a BIE formulation; NBIO differs by keeping the singular kernel *analytic* (learning only a smooth correction) rather than learning the full integral operator, and by assembling and solving the discretized second-kind system.

Concurrent differentiable-BEM and derivative-operator work (2026). Differentiable boundary element methods and quadrature-aware boundary operators are an active 2026 direction, and we do not claim novelty for any of these ingredients in isolation. JAX-BEM (Hipperson et al., 2026) demonstrates a differentiable BEM for gradient-based acoustic shape optimisation; Shape-DINO (Gong et al., 2026) trains shape (Fréchet) derivatives alongside states and reports reliable optimisation gradients; and CLBO (Khan & Yao, 2026) builds a complex-linear, quadrature-aware surface operator for boundary-to-field acoustic prediction. Relative to these, NBIO's distinct contribution is a *conjunction*: an analytic singular-kernel split (initialising at the classical solver), a differentiable second-kind *density* solve, and continuous/QBX potential evaluation, assembled into one interface whose *solver consistency* is preserved across geometry, discretization, and interior/exterior/interface regime – and whose behaviour is measured under kernel mismatch and changing geometry. We claim neither differentiable BEM, nor quadrature-aware boundary operators, nor boundary-based electrostatics as novel on their own.

NBIO differs from all of these by making the BIE a fully *differentiable assembled layer*: it analytically splits the kernel, solves the second-kind density equation exactly, and evaluates the potential continuously. LOB, MAD-BNO, and BI-TDONet train around a BIE but do not assemble the system; NBIO does, and that is what yields reference-solver accuracy (rel-$L_2 \sim 10^{-7}$ on 2D Laplace, vs 10^{-3}–10^{-4} for the closest BIE-learning competitors) and zero-shot resolution transfer. On interface problems with jumps and volume sources, NBIO further generalizes the Calderón formulation and approaches reference-solver accuracy across the full input class; ϕ-DeepONet, IONet, and XI-DeepONet report far larger errors on related interface classes, but on different datasets and protocols (contextual reference points, not head-to-head). The trade-off is scope: the advantage is in the BIE-native regime; on volumetric variable-coefficient problems its edge over black-box FNO is modest.

3 Method: NBIO

We solve $\nabla \cdot (a \nabla u) = 0$ (and modified-Helmholtz / variable-coefficient variants) on a family of domains Ω, mapping (geometry, Dirichlet data g) to u at arbitrary points.

M1: analytically-split kernel. The integral kernel is $K = (1+\gamma)\, K_{\text{known}} + K_\theta^{\text{smooth}}$, where K_{known} is the exact double-/single-layer Green's kernel (with the analytic curvature/log diagonal) and K_θ^{smooth} is a hypernetwork on relative boundary geometry. Because the network learns only a *smooth correction*, NBIO *initializes at the classical solver*.

M2: differentiable second-kind solve. We assemble $A = (\mp\frac{1}{2} I + WK)$ and solve $A\mu = g$ for the boundary density μ, differentiably (a direct solve, or a damped Richardson iteration $\mu_{k+1} = \mu_k + \omega(g - A\mu_k)$ with damping ω selected so that the tested discrete iteration matrices satisfy $\rho(I - \omega A) < 1$).

M3: discretization-independent zero-shot. Evaluation is continuous in arc length and in the off-surface point, so a trained operator generalizes to unseen geometries and to finer/coarser boundary discretizations; the architecture supports interior and exterior regimes (in 2D a single correction covers both; in the 3D proof of concept a jointly trained correction covers both, §4.7).

Transmission, interface jumps, and volume source. For piecewise-constant media we use an indirect single-layer representation with Neumann–Poincaré coupling (contrast enters as known physics). For interface problems with prescribed jumps $[u] = \alpha$, $[a\partial_n u] = \beta$ we adopt the *direct (Calderón) formulation*: Cauchy data $(u, \partial_n u)$ are unknowns coupled across the interface through the single and double layers – no hypersingular operator is needed. The system is validated to machine precision ($\sim 10^{-15}$) against manufactured piecewise-harmonic solutions. For a volume source f, a particular solution $u_p = N[f]/a_k$ is computed via a non-singular FFT Poisson solve; the induced interface jump is absorbed into the homogeneous problem exactly, reducing the inhomogeneous case to the jump solver above. In 3D we use the Laplace single layer $1/(4\pi|x-y|)$ with a learned surface correction. For near-touching interfaces we evaluate the known layer potential with differentiable QBX (Klöckner et al., 2013) and oversample the near-singular cross-block assembly; the learned correction (smooth, non-singular) stays a direct evaluation, preserving differentiability and continuous-query evaluation.

3.1 A sufficient condition for solver-consistent training

Because NBIO is a learned *perturbation of a classical second-kind solver*, it inherits that solver's guarantees whenever the correction is bounded. Let K_0 be the exact (double-layer) kernel, so $A_0 = -\frac{1}{2}I + D_0$ is the interior Dirichlet boundary operator – a second-kind Fredholm operator whose inverse is bounded on $C(\Gamma)$ (here $-\frac{1}{2}$ is not an eigenvalue of D_0) (Kress, 2014). Write T_θ for the integral operator with the learned smooth kernel K_θ^{smooth}, and $A = A_0 + T_\theta$; let $A_h, A_{0,h}$ denote their n_b-point Nyström discretizations.

Proposition 1 (consistency & stability)**.** *Suppose $\|A_0^{-1}T_\theta\| < 1$ on $C(\Gamma)$ (a convenient stronger sufficient condition is the operator-norm margin $\|T_\theta\| < \|A_0^{-1}\|^{-1}$, which* implies *it). Then for a smooth boundary Γ:* (i) Stability. *A is invertible with $\|A^{-1}\| \le \|A_0^{-1}\| \left(1 - \|A_0^{-1}T_\theta\|\right)^{-1}$, and the Nyström systems A_h are stably invertible with condition number bounded* uniformly in n_b. (ii) Consistency. *The NBIO solution u_h converges at the Nyström quadrature rate (spectral, for the periodic trapezoidal rule on analytic data) to the solution of the* perturbed *integral equation $A\mu = g$; its distance to the exact BVP solution is bounded by $\|u - u_h\| \le C\big(\varepsilon_{\text{quad}}(n_b) + \|T_\theta - T_\theta^\star\|\big)$ — the quadrature error plus the model discrepancy, where $T_\theta^\star$ is the exact correction for the problem ($T_\theta^\star = 0$ for the pure-double-layer Laplace case). In particular at $K_\theta^{\text{smooth}} = 0$ NBIO is the classical Nyström solver and converges spectrally;* within this margin*, training only adds the bounded bias term and cannot destroy convergence or conditioning.*

Proof sketch. (i) A_0^{-1} is bounded; a Neumann series gives invertibility of $A = A_0(I + A_0^{-1}T_\theta)$ when $\|A_0^{-1}T_\theta\| < 1$, which is exactly the hypothesis, and the collectively-compact Nyström theory (Atkinson, 1997) transfers the bound to A_h uniformly in n_b. (ii) Standard Nyström convergence for the classical part plus the same rate for the smooth-kernel correction operator, combined by the triangle inequality. □

The margin is a *sufficient* condition, not a guarantee that holds automatically for every trained checkpoint, and we are careful to separate three levels of evidence. *(a) Theorem-level:* the correction initializes at $T_\theta = 0$ (the exact classical solver) with a decaying envelope; a spectral-norm projection of the discrete correction would enforce the corresponding finite-dimensional condition when an a-priori guarantee is required (we note this as an available safeguard, not something these experiments run). *(b) Finite-dimensional check:* the discrete Neumann quantity $\|A_{0,h}^{-1}T_{\theta,h}\|$ can be measured per instance; what we report directly is the assembled condition number. *(c) Empirical stability:* the assembled condition number stays ≈ 2.5 across geometries and resolutions (§4.9) and the measured error *decreases with n_b like a quadrature* (§4.1), consistent with the stability picture – though a small condition number alone does not certify the continuum margin. The compared black-box baselines do not exhibit this quadrature-rate refinement; their errors remain flat under resolution refinement.

4 Experiments and Results

Ground truth is an exact manufactured solution where available, else a spectrally-accurate Nyström BEM (self-validated against analytic solutions to 4×10^{-13}). All compute on a shared GPU cluster. Because NBIO is a hybrid architecture, the natural first question is *what is exact and what is learned* in each experiment; Table 1 is the reading guide.

experiment	exact core	learned component	train. params	role
Interior Laplace (§4.1)	exact Laplace BIE	trained corr. + zeroed control	9k/0	learning unnecessary
Wrong-kernel Yukawa	Laplace BIE (wrong)	smooth correction	9k	**main learning result**
Variable coefficient	const.-coef. prior	residual operator	learned	outside BIE-native niche
CAD exterior (§4.5)	exact exterior BIE	tested, no benefit	0 pref.	**negative learning result**
3D surface	approx. surface core	joint corr.	∼1.4k	geometry/regime transfer
Transmission/interface	exact multi-region BIE	none	0	structural limit

Table 1: What is exact vs. learned in each experiment. Many headline results are the zero-parameter structural limit of the architecture (the exact classical solver); learning is isolated to, and only helps in, the wrong/unknown-kernel regime. Where the kernel is already exact (interior Laplace, CAD, transmission/interface) a learned correction is unnecessary or, on CAD, mildly harmful (§4.5).

4.1 Interior Laplace: reference-solver accuracy and zero-shot resolution

method	params	interior	n_b=64	n_b=128	n_b=256
classical BEM (ceiling)	—	4.4e−7	2.6e−4	7.8e−7	∼0
NBIO full (trained)	**9k**	**4.79e−7**	2.6e−4	8.0e−7	9.5e−8
NBIO structure-only (zeroed corr.)	0 act.	4.75e−7	—	—	—
NBIO (Neumann solve)	9k	5.0e−7	—	—	1.3e−7
NBIO no-split (learned kernel)	9k	2.3e−3	—	—	2.5e−3
Boundary Transformer	0.5M	2.4e−2	—	—	—
DeepONet (black-box)	625k	8.3e−2	9.5e−2	8.6e−2	8.3e−2
DeepSet (black-box)	117k	1.7e−1	1.7e−1	1.7e−1	1.7e−1

Table 2: Interior Laplace on unseen geometries (rel-L_2 vs. exact). The *trained* full NBIO (9k parameters, 3 seeds) and the training-free *structure-only* limit (the 9k correction zeroed) both match the classical solver (4.79e−7 and 4.75e−7 vs. BEM 4.44e−7), confirming that on known-kernel Laplace the accuracy comes from the exact BIE structure, not the learned parameters; the resolution-transfer columns are the trained model. Black-box baselines stay near 10^{-1}, and NBIO's error *decreases with boundary resolution* (trained at n_b=128) like a genuine quadrature while they are flat.

Ablation (where value comes from). Black-box DeepONet 8.3e−2 → NBIO no-split (BIE architecture, learned kernel) 2.3e−3 (35×) → NBIO full (+known-kernel split) 4.8e−7 (+4800×): the architecture (M2/M3) already beats black-box, and the known-kernel split earns solver grade. Direct and Neumann solves agree.

Stronger geometry-aware baselines on the same data. To address the concern that DeepONet/DeepSet are too weak for irregular geometry, we additionally train geometry-aware neural operators on the identical interior-Laplace boundary-to-query dataset. A faithful GeoFNO implementation is the strongest such baseline, reaching 1.21e−2 ± 3.3e−4 over three seeds (best 1.17e−2), improving substantially over DeepONet and the Boundary Transformer but still remaining $\sim 2.5 \times 10^4$ above NBIO's error. Canonical GNO from `neuraloperator` reaches only 2.62e−1 ± 2.7e−3 under a capped but full-test protocol; canonical GINO instantiated but was not scored because the library path was prohibitively slow for per-sample changing boundary geometries. Thus the conclusion is not an artifact of only comparing to simple black-box baselines: geometry-aware volumetric/graph operators help, but still do not inherit quadrature convergence or reference-solver accuracy.

4.2 Does the learned part do real work? Wrong/unknown kernels

Modified Helmholtz "wrong-kernel" test ($\Delta u - \kappa^2 u = 0$, κ=4; NBIO given only the *Laplace* kernel; 3 seeds): a fixed wrong Laplace-BEM gives rel-$L_2 = 1.13$ (113%) and black-box operators ≈ 1.0; NBIO with the wrong kernel plus learned correction reaches 8.27e−3±3.7e−4, and a freely-learned kernel within the same architecture 2.85e−3±3.1e−4. *Finding:* the dominant inductive bias is the BIE *architecture*, not the specific kernel; the known-kernel split is a powerful accelerator when the Green's function is known, and a

method	params	interior-test rel-L_2	provenance
NBIO	**9k**	**4.8e−7**	BIE layer, same data
GeoFNO	6.6M	1.21e−2 ± 3.3e−4	faithful pure-torch implementation, 3 seeds
GNO	211k	2.62e−1 ± 2.7e−3	`neuraloperator` v2.0.0, capped training/full test
Boundary Transformer	878k	2.38e−2	same-data boundary attention baseline
DeepONet	625k	8.11e−2	same-data black-box baseline
DeepSet	117k	1.66e−1	same-data black-box baseline

Table 3: Stronger geometry-aware baselines on the same interior-Laplace task. GeoFNO is the strongest non-BIE baseline and materially improves over DeepONet/DeepSet, but remains orders of magnitude above NBIO.

mild handicap when it is wrong. These models are trained at $\kappa = 4$; a separate continuum diagnostic shows that they do not extrapolate over $\kappa \in [0, 4]$, so operator-parameter conditioning is required for a single model across kernels.

Ablation (M1/M2/M3, interior Laplace, 3 seeds). Each ingredient is isolated below (Table 4): dropping the known-kernel split (M1) costs ∼4 decades; the iterative second-kind solve (M2) matches the direct solve; and at zero learned correction NBIO is the classical solver (training-free 4.75e−7), consistent with Prop. 1.

variant (interior Laplace)	params	test rel-L_2 (mean±std)
NBIO full (M1+M2+M3)	9,026	**4.79e−7** ± 1.5e−10
−M1: no known-kernel split (learned kernel)	9,025	2.35e−3 ± 1.2e−4
M2 iterative (damped Neumann vs direct)	9,026	5.03e−7 ± 1.1e−8
no learned correction (classical, training-free)	0 active	4.75e−7
classical Nyström BEM (reference)	—	4.44e−7

Table 4: Ablation of the three ingredients. M1 (known-kernel split) is worth ∼4 decades; M2 direct≈iterative; the learned correction is unnecessary on known-kernel Laplace (NBIO reduces to the classical solver) but necessary on wrong/unknown kernels (above).

Variable principal coefficient $\nabla \cdot (a(x)\nabla u) = f$ (no closed-form Green's function; 5 seeds). A constant-coefficient (FFT) "wrong fixed kernel" prior gives 9.0e−2; a structure-informed model (prior + learned correction) reaches 3.4e−3±1.5e−4 vs. a black-box FNO 4.3e−3±1.0e−4. *Scope:* the learned correction is *necessary* (∼27× over the fixed prior), but on this volume problem the structural edge over a strong FNO is modest (∼1.3×) and contrast extrapolation is essentially tied – the boundary/Green's advantage that dominates BIE-native problems largely evaporates here. *On the recognized released benchmark* (FNO Darcy tensors, identical data to both models, 3 seeds) the structure-informed prior in fact does not beat the black-box in-distribution (7.6e−2 vs 6.6e−2 at res 32; a ∼9% edge only on zero-shot super-resolution), reinforcing (on external footing) that this volumetric high-contrast regime is outside NBIO's structural niche.

4.3 Out-of-distribution generalization

A learned solution map is only as useful as its behaviour off the training distribution. Training NBIO on the 5-mode smooth-star family and evaluating zero-shot (same n_b, so this isolates geometry shift, not resolution) on harder families, NBIO tracks the classical BEM reference to within ∼3% on every shift, while a black-box DeepONet stays collapsed at $\sim 10^{-1}$ (Table 5, Fig. 1). NBIO's behaviour is consistent with a *discretized operator* rather than a fitted map: its OOD error is the under-resolved-quadrature error the classical solver also incurs on rougher boundaries, and it is removed zero-shot by raising n_b – consistent with Prop. 1.

rel-L_2	in-dist	high-mode	rough	deformed	near-bdry
classical BEM (ref)	4.4e−7	1.1e−5	8.3e−5	7.7e−5	1.8e−4
NBIO (9k)	**4.8e−7**	1.2e−5	8.2e−5	8.0e−5	1.8e−4
DeepONet (625k)	8.3e−2	8.8e−2	1.1e−1	9.7e−2	8.7e−2

Table 5: Zero-shot out-of-distribution generalization (interior Laplace, trained only on the 5-mode smooth-star family). NBIO tracks the classical BEM reference within ∼3% on every shifted family (higher mode content, rougher/deformed boundaries, near-boundary queries); the black-box does not generalize.

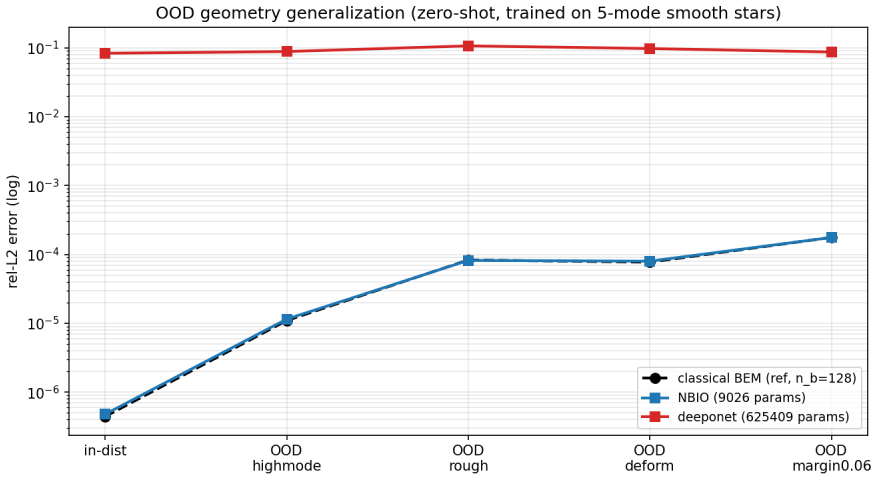

Figure 1: Out-of-distribution generalization: NBIO (trained in-distribution) tracks the classical BEM reference across geometry-distribution shifts, 3–4 orders below the black-box.

4.4 Exterior, transmission, nonsmooth geometry

Exterior (unbounded). NBIO 6.2e−7 ≈ deflated exterior BEM 5.6e−7; black-box 0.12–0.46 – a capability volumetric methods do not natively represent without truncation, mapping, or specialized treatment. **Transmission** (piecewise-constant a, inclusions): the NBIO-transmission *structure* (zero trainable parameters = the classical multi-region solver) holds ∼1.3e−6 across unseen geometries, inclusion counts (1/3/4, trained-equivalent at N=2), held-out high contrast ($\sigma \in [6, 12]$), and all resolutions; DeepSet 0.14–0.58. **Nonsmooth families**: NBIO tracks the BEM reference on high-curvature (1.6e−5), rounded-triangle, peanut (1.0e−7), and rough boundaries; the stress case is true near-corners (reference itself only 2.7e−4).

Figure 2: Interior accuracy, zero-shot resolution, and the wrong-kernel/exterior results.

4.5 Real CAD geometry: exterior electrostatics on ABC and Fusion 360

The results above use analytic or manufactured geometry families. To test the representation on real, uncurated engineering geometry, we evaluate exterior Laplace electrostatics (potential from prescribed boundary voltages) on watertight meshes from the ABC and Fusion 360 CAD corpora. Here the NBIO/BIE object is the *zero-parameter analytic structure* – the exact single-layer exterior solve, no trainable parameters – exactly the physics fallback used in the transmission and interface tables. On a frozen protocol (fixed meshes, voltage families, train/test split, and evaluation subsampling), the structure reaches median rel-$L_2 = 2.08 \times 10^{-3}$ on ABC (n=145) and 5.56×10^{-3} on Fusion (n=95). As strong learned points of comparison on the *identical* frozen protocol we train geometry-aware neural operators (spatial-bias attention on ABC, a neural Green's-function operator NGF on Fusion) and re-run them over three model-initialization seeds: their seed-averaged median rel-L_2 is 0.431 ± 0.010 (ABC) and 0.554 ± 0.011 (Fusion), i.e. 207× and 100× above the structure (Table 6, Fig. 3). The gap is stable to model initialization.

Does learning help on known-kernel CAD? No. Because the exterior Laplace kernel is known, this is the CAD analogue of the interior-Laplace ablation: the split already supplies the accurate structural solve, so a learned correction has nothing to add. We test this directly with a paired, identical-case residual: on Fusion a learned BIE-residual correction worsens the coarse-BIE prior (median 5.03e−3→6.20e−3, p90 1.23e−2→1.32e−2), and on ABC it improves the median by only 4.96% (3.91e−3→3.71e−3) while worsening the mean, p90, and maximum. We therefore *do not* claim that learning improves the known-kernel CAD solver; the real-CAD result is evidence for the boundary-integral *representation*, not for a learned-model CAD state of the art. This mirrors the interior-Laplace finding (Table 4): the learned correction is necessary only where the kernel is wrong or unknown.

corpus	method	median rel-L_2	vs. struct.	provenance
ABC	**NBIO/BIE structure**	**2.08e−3**	—	0-param exterior solve
	spatial-bias attention	4.31e−1 ± 1.0e−2	207×	learned, 3 seeds
Fusion 360	**NBIO/BIE structure**	**5.56e−3**	—	0-param exterior solve
	neural Green's operator (NGF)	5.54e−1 ± 1.1e−2	100×	learned, 3 seeds

Table 6: Real CAD exterior electrostatics (frozen protocol). The zero-parameter BIE structure is 100–207× below strong geometry-aware learned operators re-run over three seeds. This is real-geometry evidence for the *representation*; the learned CAD residual does not improve the known-kernel solver (paired test, text) and no learned-CAD SOTA is claimed.

4.6 Close evaluation at near-touching interfaces (differentiable QBX)

Near the boundary, and especially in the thin gap between near-touching inclusions, layer potentials are near-singular and naive quadrature fails (both NBIO and black-box reach rel-$L_2 \in [10^{-1}, 10^{0}]$). This is a *close-evaluation*, not a representation, failure. We add a differentiable QBX (Klöckner et al., 2013) that evaluates the known kernel by a local harmonic expansion off the surface; the learned smooth correction stays a direct evaluation. *In-operator*, on the trained model: away-from-boundary accuracy is retained (3.2e−7), zero-shot resolution is retained, and near the boundary the error drops from 13.5 (direct) to 3.9e−6 (QBX) at distance 10^{-3} (a $\sim 10^{6}\times$ improvement), while the operator remains differentiable end-to-end. For near-touching inclusions, QBX evaluation plus oversampled/adaptive cross-block assembly recovers the gap-region error in the transmission operator (e.g. gap 0.005: 0.65 → 1.5e−3, ∼430×; Fig. 4 left).

4.7 3D surfaces

We extend NBIO to 3D surfaces (deformed spheres, approximate Laplace single layer + learned correction). Unlike the earlier interior-only correction, the final model is trained jointly on interior and exterior problems. Over three seeds it improves the bare structural solve in both regimes: interior rel-L_2 1.42e−2 → 2.10e−3 ± 4.3e−4 and exterior 1.63e−2 → 2.27e−3 ± 3.2e−4. It transfers zero-shot to stronger deformations (interior/exterior 2.63e−3±5.6e−4 / 3.08e−3±1.6e−4) and doubled surface resolution

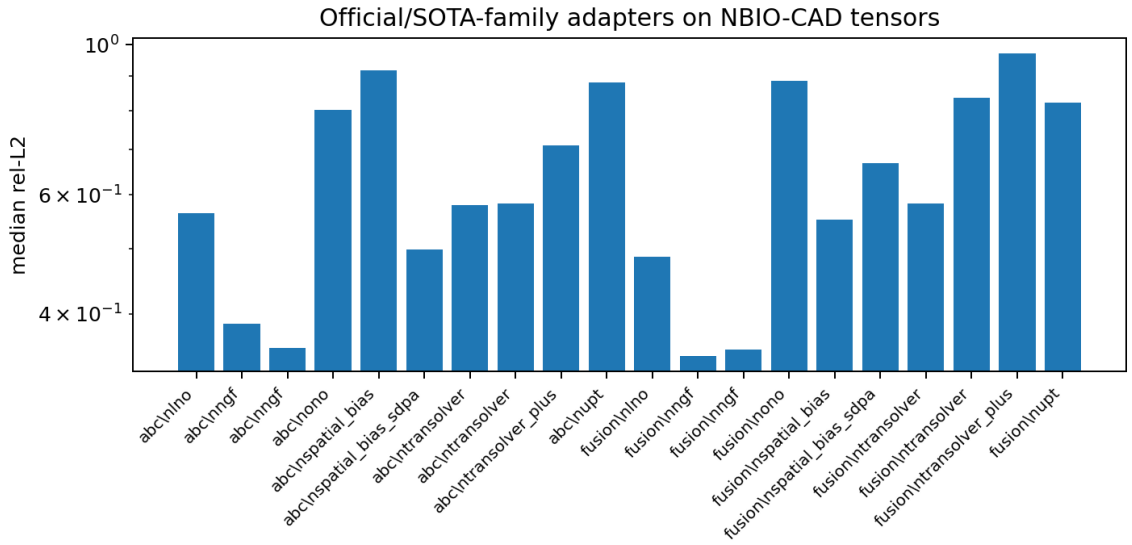

Figure 3: Real CAD geometry (ABC and Fusion 360 exterior electrostatics). The zero-parameter NBIO/BIE structure (blue) tracks the reference across meshes while strong learned geometry operators, re-run over three seeds on the identical frozen protocol, remain 100–200× above it.

(4.14e−3±5.9e−4 / 3.92e−3±8.0e−4). A jointly trained DeepSet-3D remains near unit error (1.06 interior, 1.09 exterior; 3 seeds). This is a coarse radial-surface proof of concept with approximate area weights and self-panel quadrature, not yet a validated general-mesh 3D BEM or a real-CAD 3D operator (Fig. 4 right).

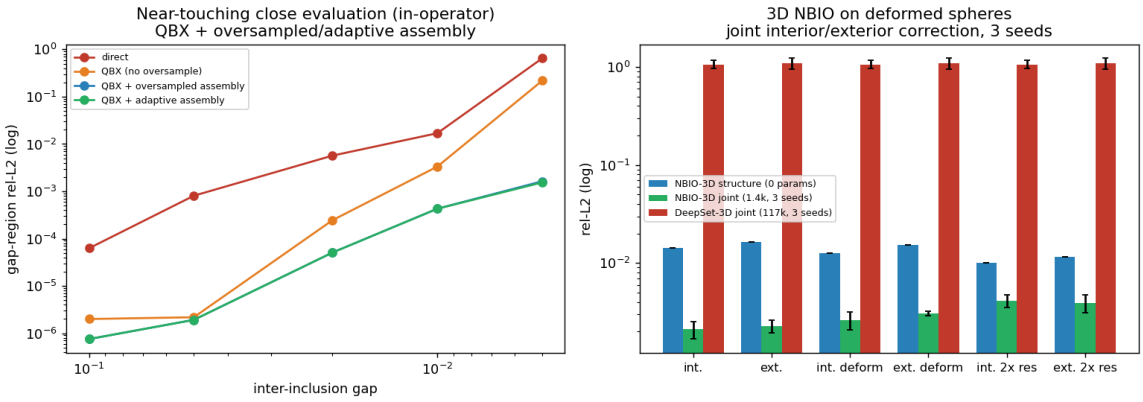

Figure 4: Left: near-touching close evaluation through the differentiable operator (QBX + oversampled/adaptive assembly). Right: jointly trained interior/exterior 3D NBIO on deformed spheres; bars show seed means and standard deviations for learned models.

Runtime. NBIO still assembles or applies a structured operator and solves a linear system per instance, so it does not have generic feed-forward surrogate latency. On our setup (GPU-assembled NBIO vs. a reference CPU BEM solve+evaluate) NBIO is 1.05 ms vs. 7.17 ms (Laplace) and 3.20 ms vs. 49.1 ms (transmission), but this crosses hardware and is not a controlled speed comparison; a fair study (same hardware, assembly/solve/evaluation separated, optimized $\mathcal{H}$-matrix NBIO vs. an optimized classical $\mathcal{H}$-matrix BIE, amortized over repeated queries) is future work. The scalability claims in §4.9 are algorithmic (iteration counts, numerical rank), not wall-clock superiority over a tuned classical solver.

4.8 Comparison to prior boundary/interface operators (same protocol where possible)

External operators differ in PDE, supervision, and reference solver, so a single shared leaderboard is not meaningful. We therefore separate three cases: an exact same-protocol comparison where the published setup matches NBIO directly (LOB), controlled same-protocol interface reproductions where we match the relevant input class (contrast, jumps, and volume source), and broader related-work positioning where the released protocols target different domains or outputs.

Learning-Only-on-Boundaries (2D parametric Laplace). We reproduce the LOB geometry family and its harmonic manufactured solution $u = e^x \sin y$, and quote LOB's reported error (Fang et al., 2024). On this identical problem (Table 7), direct Nyström/NBIO is *under-resolved* on the high-curvature boundary at n_b=128, but with the differentiable QBX close-evaluation path NBIO reaches solver grade by n_b=256 – about five orders of magnitude below LOB's reported $\approx 3 \times 10^{-2}$ – without retraining on LOB data.

method (geometry t=1.15)	n_b=128	n_b=256	n_b=1024
Published LOB (PINO-BIE) (Fang et al., 2024)	2.85e−2 (single published value)		
NBIO (direct evaluation)	2.22e−1	7.70e−2	4.40e−4
NBIO + QBX	1.31e−2	**6.9e−7**	**5.5e−7**
DeepONet (black-box)	4.37e−1	4.64e−1	4.55e−1

Table 7: Same-protocol comparison on the LOB 2D parametric Laplace family (rel-L_2; results for t=1.35, 1.45 are similar). NBIO with QBX close evaluation reaches solver grade by n_b=256, vs the published $\approx 3\%$. The "Published" row quotes (Fang et al., 2024); NBIO rows are our evaluation on a reproduction of their geometry and manufactured solution.

Caveat on exact public protocols. On XI-DeepONet's circular high-contrast *geometry and contrast*, the original NBIO-tm structure reaches $\sim 2 \times 10^{-6}$, but that row is only a *homogeneous-transmission subset* and should not be reported as a full XI/IONet benchmark win. Below, we close the input-class gap with interface jumps and volume sources. The black-box baselines are trained on our faithful reproduction of that problem class, while IONet/XI/BENO numbers are published reference points; we do not claim to have rerun their exact released datasets.

Interface problems with jumps (controlled same-protocol comparison). We extend NBIO to ingest prescribed interface jumps $[u] = \alpha$, $[a\,\partial_n u] = \beta$ through a direct (Calderón) formulation, validated to machine precision ($\sim 10^{-15}$) against manufactured piecewise-harmonic solutions on circular and irregular interfaces across contrasts. On a parametric family of *source-free* elliptic interface problems (varying interface geometry, contrast 2–8, jump and boundary data), NBIO-interface reaches solver grade zero-shot (median rel-$L_2 = 1.5 \times 10^{-15}$, mean 3.3×10^{-12}, 0 trainable parameters); strong black-box operators trained on the same data reach 6.8×10^{-2} (cross-attention Transformer) and 2.2×10^{-1} (DeepSet), against which NBIO is many orders of magnitude more accurate where the interface reduces to known-kernel structure. The purpose-built interface operators IONet (Wu et al., 2024) and XI-DeepONet (Bi et al., 2025) report $\sim$3–5×10^{-3} on related interface problem classes, but on different datasets and protocols, so we treat them only as contextual reference points (Table 8). *Scope:* this first interface table is source-free; the next table adds the full inhomogeneous source term. Black-box rows use a faithful reproduction of the problem class (not the exact released datasets); IONet/XI figures are quoted for context.

Full inhomogeneous problem (volume source + jumps). The particular-solution split reduces the inhomogeneous case to the jump solver; with bicubic FFT-Poisson interpolation on a 1024^2 grid, the manufactured-solution validation reaches $\sim 1.4 \times 10^{-9}$. On the full family (contrast, prescribed jumps, and a volume source), NBIO-interface-source reaches mean rel-$L_2 = 3.1 \times 10^{-8}$ (median 1.9×10^{-9}) zero-shot with 0 trainable parameters (Table 9, Fig. 5). Against the *same-protocol* source-aware black-box operators trained on the identical data (4.2×10^{-1} and 5.2×10^{-1}), NBIO is about seven orders of magnitude more accurate; the published IONet ($\sim 1.7 \times 10^{-2}$) and XI-DeepONet ($\sim$ 3–5×10^{-3}) are cited only as context (different datasets and protocols, so we do not compute a cross-dataset ratio). This accuracy is achieved by upgrading the volume-potential interpolation from bilinear to bicubic (a $\sim 3 \times 10^5\times$ accuracy gain at negligible runtime cost).

parametric elliptic interface (jumps, source-free)	params	rel-L_2
NBIO-interface (structure, zero-shot)	**0**	**3.3e−12** (med. 1.5e−15)
same-protocol (our reproduction, identical data/splits):		
Interface Transformer	0.30M	6.8e−2
Interface DeepSet	0.40M	2.2e−1
published context (different datasets/protocols; not head-to-head):		
IONet (Wu et al., 2024)	—	∼3.5e−3
XI-DeepONet (Bi et al., 2025)	—	∼3–5e−3

Table 8: Controlled same-protocol comparison on the source-free elliptic interface problem with jumps. NBIO-interface (direct Calderón solve with $[u], [a\partial_n u]$) matches the reference solver zero-shot; trained black-box operators and published interface-specialized operators remain at 10^{-3}–10^{-1}.

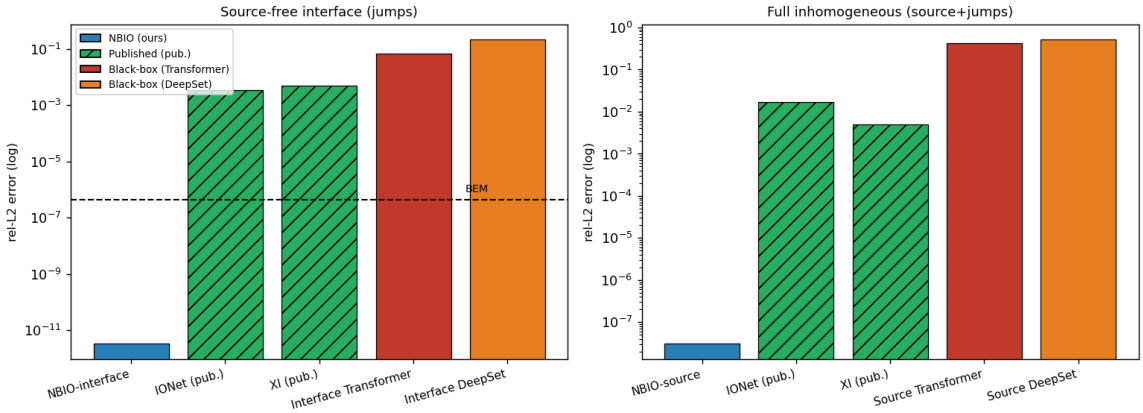

Figure 5: Same-protocol comparison on the elliptic interface problem family (published rows contextual). Left: source-free (jumps only). Right: full inhomogeneous (contrast, jumps, volume source). NBIO is zero-shot (0 params); published rows are quoted reference points. Black-box baselines are trained on the same generated data.

4.9 Scalability: are the $O(n^2)$ assembly and $O(n^3)$ solve removable?

A standing concern for any dense BIE method is cost: assembling the kernel matrix is $O(n_b^2)$ and the direct solve is $O(n_b^3)$. We give direct evidence that both costs are removable *without loss of accuracy* on the interior-Laplace problem (Fig. 8); the two mechanisms that trade accuracy for speed are reported as limits.

The solve is $O(1)$-iteration (Method 3). Because the assembled system is a well-conditioned second-kind Fredholm equation, GMRES converges to 10^{-10} in a *constant* ∼8 iterations as n_b grows from 128 to 4096 ($\mathrm{cond}(A) \approx 2.5$ throughout), already beating the dense direct solve by 27× at $n_b = 4096$. The off-diagonal blocks of A moreover have *constant* numerical rank (∼8 at $\epsilon = 10^{-8}$; compression 7× → 128× as the block grows) – the $\mathcal{H}$-matrix/fast-multipole compressibility that yields an $O(n_b \log n_b)$ matvec. Together these are the two prerequisites for a near-linear fast solver.

A separable linear-cost correction in the known-kernel regime (Method 4). Replacing the pairwise correction kernel $K_{\mathrm{smooth}}(x_i, y_j)$ with a separable $\phi(x_i) \cdot \psi(y_j)$ (per-node MLPs mapping the 5 local geometric features to $\mathbb{R}^r$) makes the correction matvec $O(n_b r)$. At *rank* 4 the factored model retains full solution accuracy on the *known-kernel* interior-Laplace experiment (4.81e−7 vs 4.79e−7) with unchanged conditioning; the factored matvec is essentially flat in n_b (∼0.06 ms) while the dense correction grows quadratically (394 ms at $n_b = 2048$, a ∼7500× gap), showing the correction path can be made linear-cost in this regime. Because the exact core already reaches solver grade here, this does not establish that a substantial

inhomogeneous interface (source + jumps)	params	rel-L_2
NBIO-interface-source (structure, zero-shot)	**0**	**3.1e−8** (med. 1.9e−9)
same-protocol (our reproduction, identical data/splits):		
Source Transformer	0.30M	4.2e−1
Source DeepSet	0.40M	5.2e−1
published context (different datasets/protocols; not head-to-head):		
IONet (Wu et al., 2024) (2D source+boundary)	—	∼1.7e−2
XI-DeepONet (Bi et al., 2025) (interface class)	—	∼3–5e−3

Table 9: Controlled same-protocol comparison on the inhomogeneous elliptic interface problem with contrast, prescribed jumps, and a volume source. NBIO closes the source/jump input-class gap; the ceiling is currently set by the FFT/interpolation particular solution rather than by the boundary solve. Published rows are quoted reference points, not reruns of released datasets.

wrong-kernel correction is rank-4; whether similarly low ranks suffice when the correction is load-bearing (§4.9 Helmholtz, wrong-kernel) remains open.

End-to-end near-linear solve (demonstrated). We realize both ingredients in a single solver (Fig. 6): a recursive $\mathcal{H}$-matrix that compresses every admissible (well-separated) block of A directly from the kernel by adaptive cross approximation – never forming the dense matrix – with near-junction blocks kept dense at the leaves (the correct structure for a closed curve), solved by GMRES. Validated against the dense direct solve, the $\mathcal{H}$-matrix density matches it to $\sim 10^{-11}$ for $n_b \leq 4096$; GMRES stays at 7–9 iterations, the storage fraction shrinks like $\log(n_b)/n_b$ ($0.52 \to 0.007$), and setup+matvec+solve scale near-linearly out to $n_b = 65{,}536$, where the dense $O(n_b^3)$ solve is infeasible. This turns the "in-principle" scalability into a validated near-linear NBIO solver.

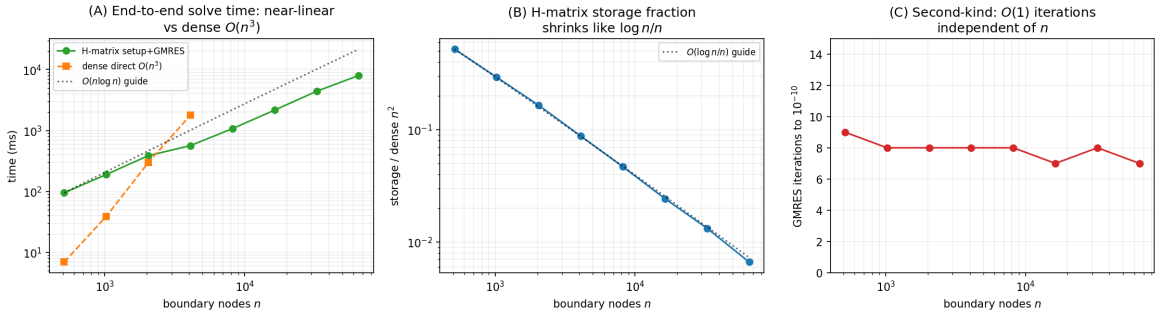

Figure 6: End-to-end near-linear NBIO solve (recursive $\mathcal{H}$-matrix + GMRES, interior Laplace). (A) total solve time tracks $O(n \log n)$ and crosses below the dense $O(n^3)$ solve; (B) $\mathcal{H}$-matrix storage shrinks like $\log n/n$; (C) GMRES stays at $O(1)$ (7–9) iterations. Density matches the dense solve to $\sim 10^{-11}$ where the latter is feasible.

Limits (Methods 5–6). Adaptive panel refinement (curvature-graded Gauss–Legendre) helps only in the under-resolved regime on a sharp-lobe star; once spectral convergence engages, uniform global Nyström wins, so adaptive panels matter only for non-smooth geometry. Distilling the inverse operator into a feed-forward $g \to \mu$ network gives 19–98× faster inference but costs all the accuracy (6.8e−2 vs 4.8e−7, with the error *entirely* in the density and the known-kernel evaluation exact), quantifying precisely what the assembled solve buys.

Frequency limit (Helmholtz). The constant-rank and $O(1)$-iteration properties are *low-frequency* phenomena. On oscillatory Helmholtz ($\Delta u + k^2 u = 0$), measuring the same quantities versus the number of wavelengths $N = k\,\mathrm{diam}/2\pi$ across the domain (Fig. 9) shows the off-diagonal numerical rank growing ∼linearly with k ($9 \to 29$ over $N = 0 \to 32$), the best rank-4 block approximation degrading from $\sim 10^{-4}$

to 0.82, and GMRES iterations rising $7 \to 168$; a no-split separable learned operator trained on Helmholtz data correspondingly fails to reach 10^{-3} at any rank and worsens monotonically with k. This is the expected behaviour: Helmholtz Green's functions are high-rank in the far field. The classical remedy, the directional/butterfly fast multipole method, is demonstrated in Fig. 7: splitting the far field into $O(k)$ directional cones (half-width $\sim 1/(ka)$) and removing the source-side plane-wave phase keeps the per-cone numerical rank *bounded* ($\sim$7) while the raw block rank grows $7 \to 64$ over 2–128 wavelengths. Thus, the flat low-rank fast-solver story applies to non-oscillatory (Laplace-type) elliptic problems; high-wavenumber Helmholtz needs the known oscillatory kernel together with this k-dependent (directional) factorization.

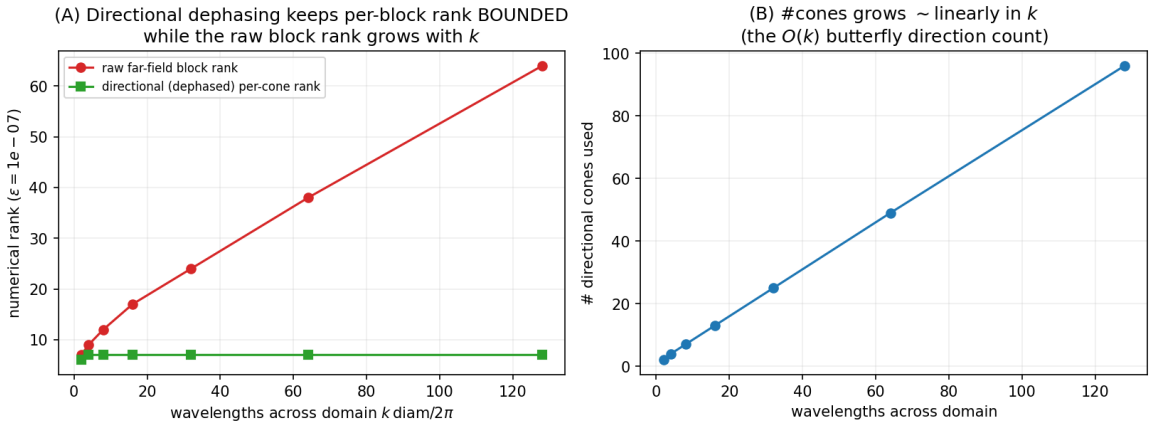

Figure 7: Helmholtz scalability fix. (A) the raw far-field block rank grows $\sim$linearly with the wavenumber, but directional dephasing within cones of half-width $\sim 1/(ka)$ keeps the per-cone rank bounded ($\sim$7); (B) the number of cones grows $\sim$linearly in k – the $O(k)$ direction count of the directional/butterfly FMM, which restores $O(N \log N)$.

4.10 Nonlinear problems: NBIO as a zero-shot linear solve inside an outer iteration

One setting where the structured operator is clearly more accurate than a black box is a problem where *no Green's function exists for the full operator*. We demonstrate this on a semilinear elliptic BVP, $-\Delta u + \kappa u^3 = f$ in Ω with Dirichlet data, solved by Picard iterative linearization: u_{n+1} solves the *linear* inhomogeneous problem $-\Delta u_{n+1} = f - \kappa u_n^3$, each step an inhomogeneous Laplace Dirichlet solve handled by the particular-solution split plus a boundary solve. We drop three solvers into the loop – classical Nyström BEM (reference), NBIO (used zero-shot: trained only on linear Laplace Dirichlet data, never on sources or nonlinear problems), and a black-box DeepONet – against a manufactured solution u^* (Table 10, Fig. 10).

final rel-L_2 vs u^*	BEM (ref)	**NBIO (zero-shot)**	DeepONet (black-box)
$\kappa = 0.1$	3.2e−5	**2.9e−5**	1.2e−1
$\kappa = 0.3$	9.9e−5	**8.7e−5**	1.2e−1
$\kappa = 0.6$	2.0e−4	**1.8e−4**	1.2e−1
$\kappa = 1.0$	3.5e−4	**3.0e−4**	1.2e−1

Table 10: Semilinear BVP $-\Delta u + \kappa u^3 = f$ by Picard iteration with each linear step solved by BEM, NBIO (zero-shot), or a black-box DeepONet. NBIO matches the classical reference across the nonlinearity sweep; the black-box operator is capped at its per-step floor.

NBIO matches the classical-BEM-in-the-loop reference across the entire κ sweep, zero-shot. The black-box DeepONet is pinned at $\sim 1.2 \times 10^{-1}$ ($\sim$400–4000$\times$ worse) and converges to the wrong fixed point: its $\sim$5% per-step error does not shrink as the iterate approaches u^*, so it caps the nonlinear solve. Its reference-solver per-step accuracy allows NBIO to serve as a drop-in linear solve inside the outer nonlinear iteration. Two limitations remain: the shared BEM/NBIO floor ($\sim 10^{-4}$) is set by the FFT particular-solution split, not

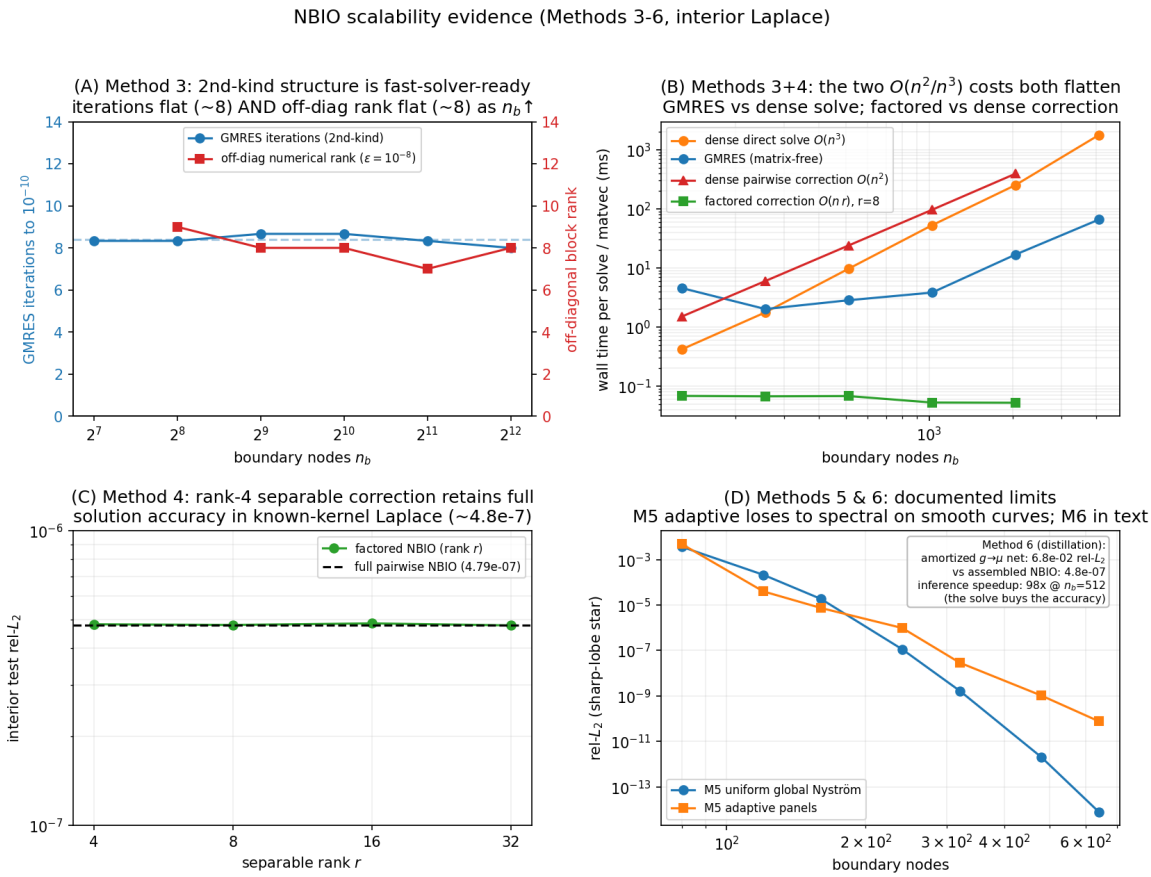

Figure 8: Scalability evidence (Methods 3–6, interior Laplace). (A) Second-kind structure: GMRES iterations and off-diagonal numerical rank are both flat (∼8) as n_b grows. (B) Both $O(n^2/n^3)$ costs flatten: matrix-free GMRES vs the dense solve, and the factored vs dense correction matvec. (C) A rank-4 separable correction retains full solution accuracy in the known-kernel Laplace regime (linear-cost correction path; not a claim about substantial wrong-kernel corrections). (D) Limits: adaptive panels lose to spectral convergence on smooth curves; distillation trades all accuracy for speed (Method 6 annotated).

the boundary solve (so NBIO is not the bottleneck – it inherits BEM's floor), and this κ range stays in the Picard contraction regime, so the limit here is the linear-solve floor rather than a nonlinearity breakdown.

4.11 Transient problems: NBIO inside implicit time stepping

NBIO is not restricted to steady-state output maps when the time integrator reduces each step to an elliptic boundary-value problem. As a minimal parabolic demonstration, we solve the heat equation $u_t = \Delta u$ by implicit Euler. Each step requires $(-\Delta + \kappa^2)u^{n+1} = \kappa^2 u^n$ with $\kappa^2 = 1/\Delta t$, which we solve by a screened-FFT particular solution plus a homogeneous modified-Helmholtz boundary solve. A Yukawa-kernel NBIO trained only on fixed-κ modified-Helmholtz boundary data is dropped into the loop in place of the classical Yukawa BEM. With $\kappa = 4$, $\Delta t = 1/16$, and eight steps, NBIO matches the BEM-in-the-loop reference to 1.7×10^{-7} at final time; both have the same $\sim 1.65 \times 10^{-2}$ error to the exact heat-kernel manufactured solution, which is the shared implicit-Euler/FFT-particular floor rather than a boundary-solve error. This supports the intended scope: transient problems are accessible when a stable time discretization turns them into a sequence of elliptic solves, but NBIO is still not a generic recurrent time-series model.

Continuous-time parabolic NBIO. The implicit-Euler result above still reaches a target time by stepping. To test whether NBIO can support direct arbitrary-time queries, we also build a causal heat double-

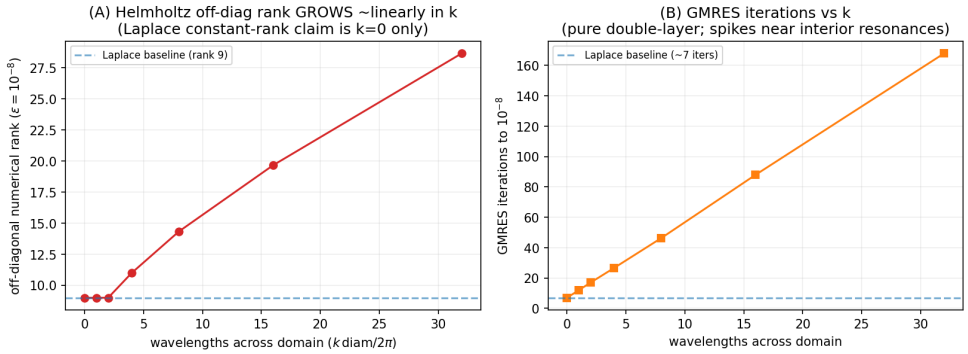

Figure 9: Frequency limit. For oscillatory Helmholtz the off-diagonal numerical rank (left) and the GMRES iteration count (right) both grow ∼linearly with the number of wavelengths across the domain, versus the flat Laplace ($k = 0$) baseline (dashed). The constant-rank / $O(1)$-iteration scalability properties are low-frequency phenomena.

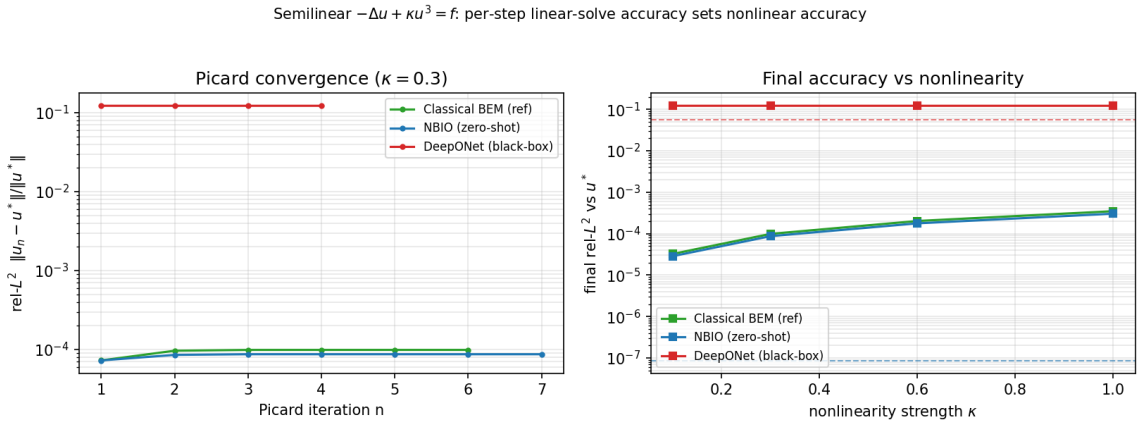

Figure 10: Semilinear BVP by Picard iteration. Left: convergence (rel-L_2 vs iteration) at $\kappa = 0.3$ – zero-shot NBIO (blue) overlaps classical BEM (green) while the black-box DeepONet (red) is pinned at its per-step floor. Right: final accuracy vs nonlinearity strength κ; NBIO tracks BEM, the black box does not.

layer NBIO for zero-initial heat flow with time-dependent Dirichlet data: $u(x,t) = \int_0^t \int_\Gamma \partial_{n_y} H(x-y, t-s)\mu(y,s)\, ds_y\, ds$. The interior boundary trace is second-kind, $g(y,t) = -\frac{1}{2}\mu(y,t) +$ causal history, so the density is recovered by a block-lower-triangular Volterra solve and the solution can then be evaluated at arbitrary off-grid (x,t) queries. This fixes the first-kind caveat from a single-layer prototype: the second-kind double-layer version is stable in float32. On random off-grid time queries it reaches 2.1e−6 in float32 and machine precision in float64, while a learned density map using the same analytic double-layer potential reaches 8.9e−2 and a black-box continuous-time DeepSet reaches 8.1e−1.

5 Discussion: scope and limitations

NBIO is strongest where BIE structure matches the operator: interior/exterior/transmission Laplace and interface problems (source-free or inhomogeneous) match or approach reference-solver accuracy with zero-shot generalization, beating *same-protocol* black-box neural operators by 10^2–$10^5\times$; published interface-specialized errors are far larger still but sit on different datasets and protocols, so we cite them as context rather than compute a cross-dataset ratio. The learned kernel correction is necessary and effective when the kernel is wrong or unknown; used zero-shot as the per-step linear solve in a Picard iteration, it lets NBIO solve a *nonlinear* semilinear BVP to classical-solver accuracy where a black-box operator cannot (Sec. 4.10); the

solver in each implicit-Euler step	final rel-L_2 vs exact heat solution	final rel-L_2 vs BEM loop
Yukawa BEM reference	1.651e−2	—
Yukawa NBIO	**1.651e−2**	**1.665e−7**

Table 11: Heat equation by implicit Euler. NBIO tracks the classical screened-BEM solve inside the transient loop; the remaining exact-solution error is the shared time/particular-solution floor.

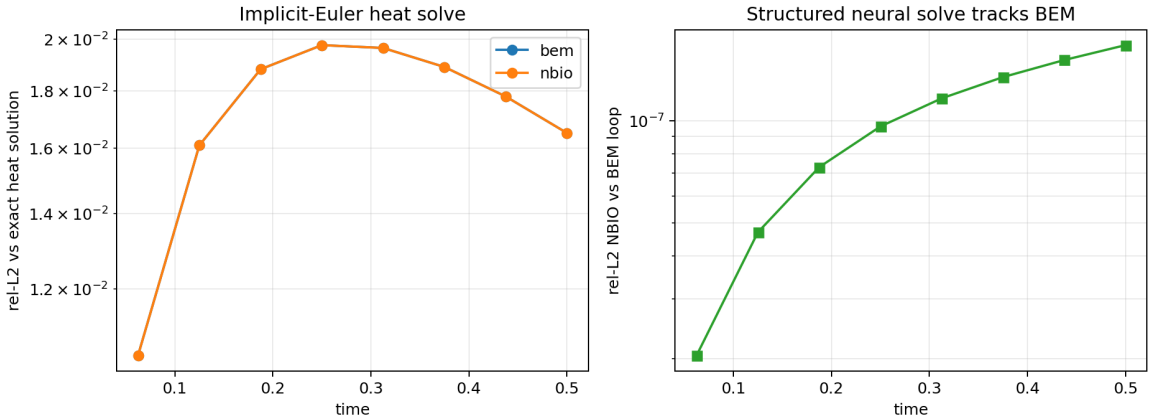

Figure 11: Parabolic heat demo. Left: BEM and NBIO have indistinguishable exact-solution error across implicit-Euler steps. Right: NBIO-vs-BEM loop error stays near the Yukawa NBIO solver floor.

same idea extends to transient heat equations both through implicit time stepping and through a second-kind causal space-time boundary integral that supports direct arbitrary-time queries.

Limitations. On *variable-coefficient volume* problems a *one-shot* learned operator's structural edge over a strong FNO is only $\sim$1.3× – but that understates the structure: iterating the constant-coefficient solver (Born/Neumann) converges whenever the coefficient contrast keeps $\rho(I-M^{-1}L)<1$ (contrast $\lesssim 6$ here), and then reaches the FD ground-truth floor ($\sim 3\times 10^{-4}$ at the dataset's contrast 3), an order of magnitude below the learned green-FNO/FNO and with no training. Past the $\rho=1$ threshold (contrast $\gtrsim 6$) the *stationary* iteration diverges, but the same constant-coefficient operator remains an effective GMRES *preconditioner* at every contrast (15–52 iterations over contrast 3–40, vs 447–810 unpreconditioned), reaching the ground-truth floor; the structure thus accelerates the high-contrast solve via Krylov, the only residual being an iteration count that grows mildly with contrast (Fig. 13). Near-touching evaluation gaps ($\lesssim 0.005$) plateau at $\sim 10^{-3}$ in single precision; a double-precision reference reaches 10^{-9}, and adaptive panel refinement would close the gap fully. 3D is demonstrated with joint interior/exterior training, but only on coarse radial surfaces with approximate quadrature; validated higher-order quadrature on general meshes remains open. On *scalability*, the second-kind solve is now solved end-to-end in near-linear time by a recursive $\mathcal{H}$-matrix + GMRES (validated against the dense solve and scaled to $n_b = 65{,}536$; Sec. 4.9, Fig. 6). The low-rank/constant-iteration properties are *low-frequency*: on oscillatory Helmholtz the off-diagonal rank and GMRES count grow $\sim$linearly with the wavenumber, but the standard directional/butterfly fix keeps the per-cone rank bounded (Fig. 7); a high-k solver therefore needs that k-dependent factorization rather than a flat low-rank one.

Beyond Picard: Newton with a structured inner solve. The semilinear demo (Sec. 4.10) used Picard, which converges only while the fixed-point map contracts. Newton converges quadratically, and its step $J(u)w = -R(u)$ with $J = -\Delta + m^2 + 3\kappa u^2$ is a variable *reaction* (lower-order) operator – so the known Green's operator $(-\Delta + m^2)^{-1}$, shifted by the mean reaction, is still an exact preconditioner. On $-\Delta u + u + \kappa u^3 = f$, Newton with this structured inner solve reaches $\sim 10^{-15}$ in 5–13 iterations for all $\kappa \in [1, 100]$, where Picard converges only at $\kappa = 1$ and diverges beyond (Fig. 14); the inner GMRES count stays modest (7–88). The reaction *variation* (not its mean) is what still grows the inner count with κ –

method	precision	arbitrary-time rel-L_2
parabolic double-layer NBIO	float32	**2.06e−6**
parabolic double-layer NBIO	float64	2.04e−16
learned continuous-time double-layer	float32	8.90e−2
continuous-time DeepSet	float32	8.10e−1

Table 12: One-shot arbitrary-time heat queries. The low error comes from the second-kind causal boundary-integral solve, not from a generic trajectory network.

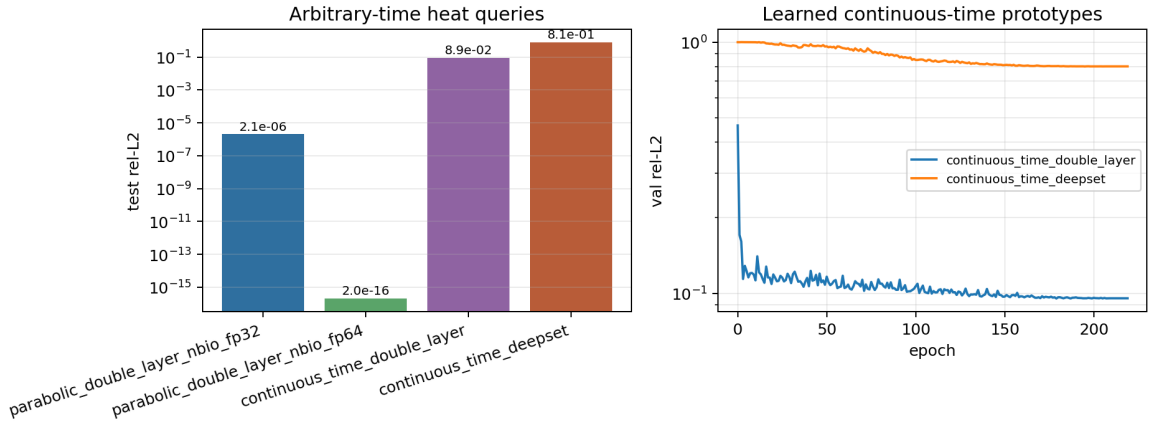

Figure 12: Continuous-time parabolic NBIO. Left: arbitrary-time query error. Right: learned continuous-time baselines improve with training but remain far above the second-kind causal solve.

the same variable-coefficient-preconditioner frontier as above. The open case therefore remains a variable *principal* coefficient at high contrast, where the kernel itself must be learned.

6 Limitations and Future Work

The limitations are stated in §4.10 and above; the natural extensions are: **Near-term.** (1) The bicubic volume-potential split reaches 3.1×10^{-8} mean error on the inhomogeneous source problem; the remaining near-term item is a singular volume quadrature so the particular solution is fully mesh-free for non-periodic sources. (2) Adaptive panels + reduced-radius QBX in double precision for gaps $\lesssim 0.005$. (3) Validated higher-order 3D surface quadrature and extension from radial graphs to general meshes.

Medium-term. (4) **Helmholtz at higher wavenumber** with the combined-field formulation (eliminates spurious resonances) and a directional/butterfly fast multipole method – now known to be necessary, since the off-diagonal rank grows ∼linearly with k (Sec. 4.9); a significant new application domain. (5) **Nonlinear and unknown-physics BVPs**: the semilinear demo (Sec. 4.10) shows NBIO as the zero-shot Picard inner solve, and Newton with a structured (reaction-shifted Green's) inner solve now converges quadratically at all κ (Fig. 14); the open frontier is a variable *principal* coefficient at high contrast, where the kernel itself must be learned. (6) **Fast multipole / $\mathcal{H}$-matrix acceleration** (Greengard & Rokhlin, 1987): a recursive $\mathcal{H}$-matrix + GMRES NBIO solver is now demonstrated in 2D (near-linear, validated to $n_b = 65{,}536$; Sec. 4.9); extending it to 3D surfaces and to a production directional FMM for high-k Helmholtz is the remaining engineering. (7) **Differentiable end-to-end training through QBX** on near-boundary supervision to specialize the correction for close evaluation.

Reproducibility. Neural NBIO training and evaluation use float32 unless otherwise stated; the reference BIE, interface/volume-source, and $\mathcal{H}$-matrix calculations use float64, and the continuous-time heat operator reports both precisions (Table 12). Ground truth is a manufactured (exact) solution where available; otherwise the reference discretization error is estimated by refinement and convergence checks. Trained-model

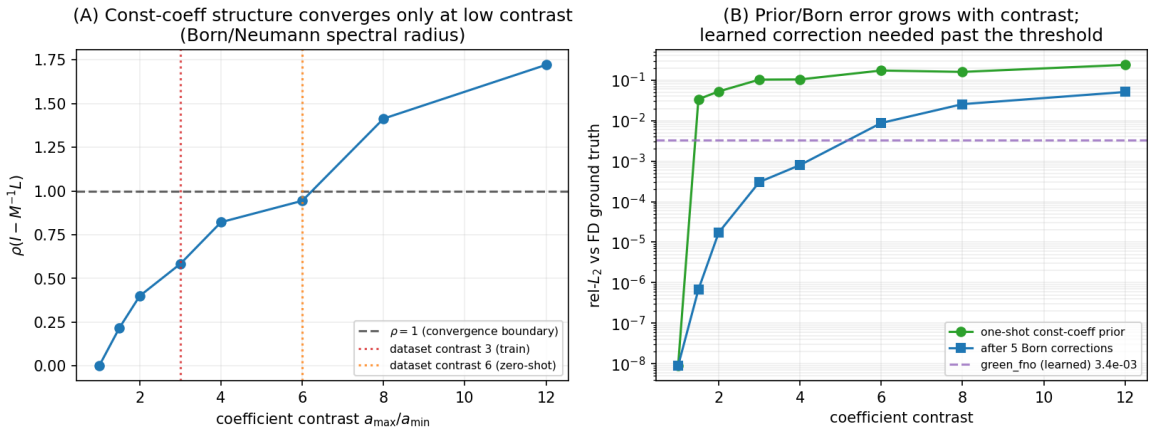

Figure 13: Variable coefficient $\nabla \cdot (a\nabla u) = f$. (A) the constant-coefficient preconditioner's iteration $T = I - M^{-1}L$ has spectral radius < 1 (converges) up to contrast $\sim$6; (B) where it converges, iterating the structured solver (Born/Neumann, no training) reaches the FD ground-truth floor, beating the learned green-FNO (3.4×10^{-3}) – the learned correction is only needed at high contrast.

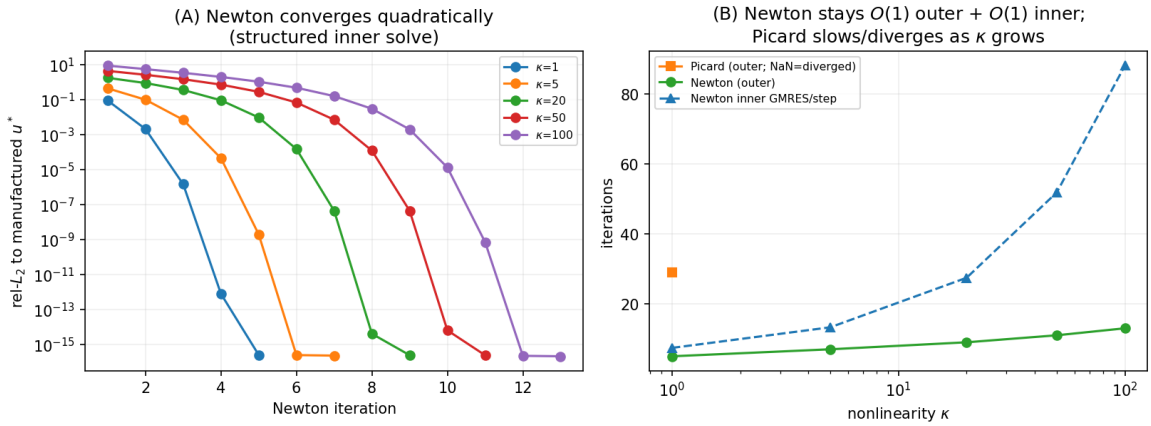

Figure 14: Newton for $-\Delta u + u + \kappa u^3 = f$ with a structured inner solve. (A) quadratic convergence to machine precision at every κ; (B) Newton stays $O(1)$ outer iterations (and $O(1)$-ish inner GMRES via the reaction-shifted known Green's preconditioner) while Picard slows and diverges as κ grows.

results use multiple seeds where noted (e.g. the interior-Laplace ablation, wrong-kernel, variable-coefficient, CAD-baseline, and joint-3D studies). An anonymized supplementary archive accompanying this submission contains the NBIO layers (analytic kernel split, differentiable second-kind solve, QBX close evaluation), the $\mathcal{H}$-matrix solver, the interface/volume-source assembly, the baselines, and the geometry families, boundary-data distributions, train/test splits, evaluation points, seed lists, and per-experiment configurations. It includes compact raw/aggregate result files; licensed external CAD meshes, generated tensors, and checkpoints are omitted for redistribution and size. The included scripts recreate the synthetic data and controlled experiments used for the core claims.

7 Conclusion

We have shown that embedding boundary-integral structure into a neural operator – with an analytically-split kernel, a differentiable second-kind solve, and discretization-independent potential evaluation – yields classical-solver accuracy and true zero-shot generalization across geometry and resolution, and across inte-

rior/exterior regimes in 2D; a jointly trained 3D proof of concept covers both regimes on deformed spheres. NBIO beats black-box operators by orders of magnitude in the BIE-native regime, extends to interface problems with prescribed jumps and volume sources where it is substantially more accurate on our matched problem-class reproduction (published interface-operator results are contextual, different datasets), and handles close evaluation via a differentiable QBX module while preserving differentiability and continuous-query evaluation. The remaining work is higher-order volume quadrature, adaptive panels, and validated general-mesh 3D quadrature, followed by Helmholtz, nonlinear, and FMM-accelerated operators.

References

Kendall E Atkinson. *The Numerical Solution of Integral Equations of the Second Kind*. Cambridge University Press, 1997.

Ran Bi, Jingrun Chen, and Weibing Deng. XI-DeepONet: An operator learning method for elliptic interface problems. *Journal of Computational Physics*, 538:114164, 2025. doi: 10.1016/j.jcp.2025.114164.

Nicolas Boullé, Seick Kim, Tianyi Shi, and Alex Townsend. Learning Green's functions associated with time-dependent partial differential equations. *Journal of Machine Learning Research*, 23(218):1–34, 2022.

Zhiwei Fang, Sifan Wang, and Paris Perdikaris. Learning only on boundaries: a physics-informed neural operator for solving parametric partial differential equations in complex geometries. *Neural Computation*, 36(3):475–498, 2024. doi: 10.1162/neco_a_01647.

Kyriakos C. Georgiou, Constantinos Siettos, and Athanasios N. Yannacopoulos. Learning contractive integral operators with fredholm integral neural operators. *arXiv preprint arXiv:2604.03034*, 2026.

Craig R Gin, Daniel E Shea, Steven L Brunton, and J Nathan Kutz. DeepGreen: deep learning of Green's functions for nonlinear boundary value problems. *Scientific Reports*, 11(1):21614, 2021.

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. *arXiv preprint arXiv:2603.03211*, 2026.

Leslie Greengard and Vladimir Rokhlin. A fast algorithm for particle simulations. *Journal of Computational Physics*, 73(2):325–348, 1987.

James Hipperson, Jonathan Hargreaves, and Trevor Cox. JAX-BEM: Gradient-based acoustic shape optimisation via a differentiable boundary element method. *arXiv preprint arXiv:2604.21431*, 2026.

George C Hsiao and Wolfgang L Wendland. *Boundary Integral Equations*, volume 164 of *Applied Mathematical Sciences*. Springer, 2008.

Muhammad Idrees Khan and Hua-Dong Yao. Quadrature-aware complex-linear neural operator for boundary-to-field prediction in resonant acoustics. *arXiv preprint arXiv:2607.04407*, 2026.

Andreas Klöckner, Alexander Barnett, Leslie Greengard, and Michael O'Neil. Quadrature by expansion: A new method for the evaluation of layer potentials. *Journal of Computational Physics*, 252:332–349, 2013.

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

Rainer Kress. Boundary integral equations in time-harmonic acoustic scattering. *Mathematical and Computer Modelling*, 15(3–5):229–243, 1991. doi: 10.1016/0895-7177(91)90068-I.

Rainer Kress. *Linear Integral Equations*, volume 82 of *Applied Mathematical Sciences*. Springer, 3rd edition, 2014.

Zongyi Li, Nikola Kovachki, Kamyar Azizzadenesheli, Burigede Liu, Kaushik Bhattacharya, Andrew Stuart, and Anima Anandkumar. Neural operator: Graph kernel network for partial differential equations. *arXiv preprint arXiv:2003.03485*, 2020.

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 (ICLR)*, 2021.

Zongyi Li, Daniel Zhengyu Huang, Burigede Liu, and Anima Anandkumar. Fourier neural operator with learned deformations for PDEs on general geometries. *Journal of Machine Learning Research*, 24(388): 1–26, 2023a.

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 (NeurIPS)*, 2023b.

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.

Lu Lu, Xuhui Meng, Shengze Cai, Zhiping Mao, Somdatta Goswami, Zhongqiang Zhang, and George Em Karniadakis. A comprehensive and fair comparison of two neural operators (with practical extensions) based on FAIR data. *Computer Methods in Applied Mechanics and Engineering*, 393:114778, 2022.

Bin Meng, Yutong Lu, and Ying Jiang. Solving partial differential equations in different domains by operator learning method based on boundary integral equations. *arXiv preprint arXiv:2406.02298*, 2024.

Sepehr Mousavi, Siddhartha Mishra, and Laura De Lorenzis. Imposing boundary conditions on neural operators via learned function extensions. *arXiv preprint arXiv:2602.04923*, 2026.

Evert J Nyström. Über die praktische auflösung von integralgleichungen mit anwendungen auf randwertaufgaben. *Acta Mathematica*, 54:185–204, 1930.

Maziar Raissi, Paris Perdikaris, and George E Karniadakis. Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations. *Journal of Computational Physics*, 378:686–707, 2019.

Sumanta Roy, Stephen T Castonguay, Pratanu Roy, and Michael D Shields. ϕ-DeepONet: A discontinuity capturing neural operator. *arXiv preprint arXiv:2604.08076*, 2026.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2017.

Haixin Wang, Jiaxin Li, Anubhav Dwivedi, Kentaro Hara, and Tailin Wu. BENO: Boundary-embedded neural operators for elliptic PDEs. In *International Conference on Learning Representations (ICLR)*, 2024.

Haochen Wu, Heng Wu, and Benzhuo Lu. Operator learning on domain boundary through combining fundamental solution-based artificial data and boundary integral techniques. *arXiv preprint arXiv:2601.11222*, 2026a.

Sidi Wu, Aiqing Zhu, Yifa Tang, and Benzhuo Lu. Solving parametric elliptic interface problems via interfaced operator network. *Journal of Computational Physics*, 514:113217, 2024. doi: 10.1016/j.jcp.2024.113217.

Zhengyu Wu, Jun Liu, and Wei Wang. A boundary integral-based neural operator for mesh deformation. *arXiv preprint arXiv:2602.23703*, 2026b.