QuOOD: Benchmarking Out-of-Distribution Property Prediction on Quantum Circuits

Anonymous authors
Paper under double-blind review

Abstract

Machine-learned surrogates that predict the noisy outputs of quantum circuits or mitigate their errors learn from classical simulation, whose cost grows exponentially with the number of qubits for general circuits. They are trained on circuits small or structured enough to simulate and deployed on circuits that are wider, deeper, from new families, or run on new devices. Distribution shift is therefore the normal deployment condition. Existing datasets test these surrogates in distribution or one shift at a time. We introduce QuOOD, a benchmark that makes performance under these shifts measurable: 99,386 circuits and 394,684 instances, split into an i.i.d. control and four shifts in depth, width, circuit family, and device noise, with every test regime kept within the range where simulation still gives exact or error-bounded labels. Because many target values are close to zero, a low average error can hide both a model that predicts nothing and one that invents structure. QuOOD checks its farthest test regimes for signal and for a clear margin over predicting zero, and adds a control stratum of qubit pairs whose correlations the circuit's structure certifies as near zero. Across thirteen predictors, tree ensembles and neural models swap places: trees over engineered features lead every neural model at 2–3$\times$ the training width, where each target depends only on a bounded set of gates, and trail them on the held-out family. Graph networks fall further behind as width grows, even our noise-conditioned transformer with its width inputs removed. On the controls, the transformers (ours and GraphGPS) emit more spurious correlation than their measured input carries, whereas the trees shrink it. Our transformer leads on held-out calibrated devices through per-block noise conditioning. On the held-out family, validation error stops tracking the test ranking (Spearman $\rho = 0.00$, against 0.99 in distribution), so models meant for shifted deployment must be tested under shift, which QuOOD enables. We plan to release the dataset, splits, labels, and reference runs upon acceptance.

1 Introduction

Quantum computers today are noisy: the value a device measures differs from what its circuit would ideally produce. Learned models help in two ways: predicting a device's noisy output without running it (Zlokapa & Gheorghiu, 2020; Vadali et al., 2024), and recovering the noise-free value from noisy measurements at a fraction of the cost of physics-based error mitigation (Liao et al., 2024; Wang et al., 2026; Patil et al., 2025). The input is a quantum circuit, a directed acyclic graph (DAG) of gates acting on qubits, so the task is graph-level regression by a classical model. What sets it apart is its label oracle: labels come from classically simulating the circuit, at a cost that grows exponentially with the number of qubits for general circuits. Training data can contain only circuits small or structured enough to simulate, while the circuits worth predicting are precisely the ones that are not. The label oracle runs out exactly where the model is needed, so such a surrogate is used out of distribution by construction, on more qubits, deeper circuits, new families, or unseen hardware.

Because circuits are generated and labeled by simulation, they are also an unusually clean testbed for distribution shift in general. Graph out-of-distribution benchmarks for molecules and social networks (Gui et al., 2022; Ji et al., 2023; Koh et al., 2021) have made distribution shift a first-class target, and size generalization is a known failure mode of graph networks (Yehudai et al., 2021; Bevilacqua et al., 2021). Circuits add a combination those benchmarks lack: width, depth,

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.

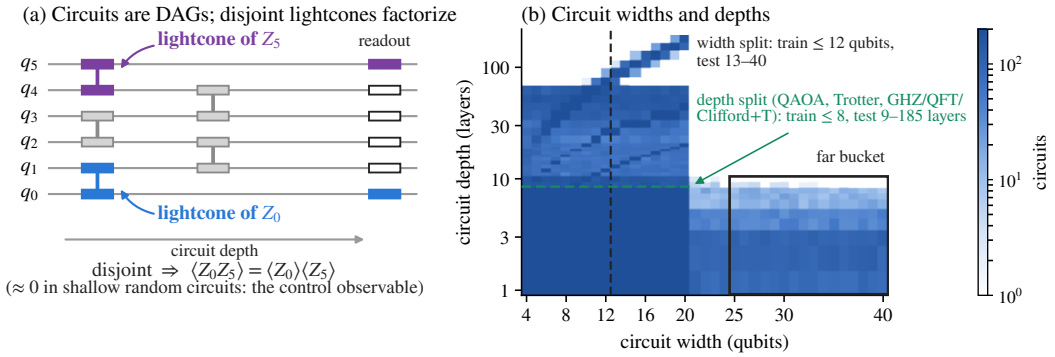

Figure 1: QuOOD in one view: certified controls (a) and test regions beyond the training range (b). **(a)** An observable depends only on gates inside its reverse lightcone; disjoint lightcones factorize the pair expectation, giving certified near-zero controls. **(b)** The regression circuits by width and depth (shading: circuits per cell, log scale). Dashed lines mark the width-split boundary (train $\leq$12 qubits) and the depth-split boundary (train $\leq$8 layers, drawn over that split's three families); the box is the width split's far bucket. Family and device shifts are held out separately (not shown).

family, and noise are explicit generation parameters with physical meaning; every label is exact or error-bounded, so benchmark error is essentially model error; and for some inputs the circuit's structure fixes the answer in advance, so spurious output can be measured directly. Existing circuit datasets do not exploit this: QASMBench and MQT Bench (Li et al., 2023; Quetschlich et al., 2023) carry no learned-prediction tasks, MNISQ (Placidi et al., 2026) evaluates in distribution, and QEM-Bench (Bao et al., 2025b) tests one shift at a time on hundreds of circuits.

QuOOD is built to test surrogates under shift (Figure 1). To study extrapolation without losing ground truth, it separates training and test regimes inside the range where reference labels can still be computed. It contains 99,386 circuits and 394,684 (circuit, observable, noise) instances with exact or error-bounded labels for two simulation-bound tasks, noisy-expectation prediction (Task A) and error mitigation (Task B, the main task), plus a compilation-cost task. Five splits cover an i.i.d. control and four shifts along depth, width (qubit count), circuit family, and device. Two instruments address a common trap: many measured quantities are close to zero, so a low average error can hide both a model that predicts zero and one that invents structure. We call the farthest test regime of the depth or width split its *far bucket* (for width, 2–3$\times$ the training width); it must contain enough non-zero signal, and its best model must beat predicting zero by a clear margin. In the widest circuits (up to 40 qubits), distant qubit pairs whose correlation the circuit's structure certifies as near zero (Figure 1a) form a control stratum that exposes invented structure.

We evaluate thirteen predictors under one protocol, from four feature-based models (ridge, random forest, XGBoost, and an MLP, all on 70 engineered features) to six graph networks (GCN, GIN, GatedGCN, DAGNN, GraphGPS, and ML-QEM of Liao et al., 2024) and a *noise-conditioned graph transformer* (NCGT), a reference model that injects the noise description into every block by feature-wise modulation (Perez et al., 2018) rather than only at the readout. Hybrid versions of the NCGT and GraphGPS also append the 70 engineered features before the readout.

The central lesson is that which model is best depends on the shift, and that validation on the training distribution does not reveal it; measuring each shift separately, with trustworthy labels, makes this visible. Our contributions are:

- **Benchmark.** A large-scale circuit-property benchmark with exact or error-bounded labels, four distribution-shift axes, and two instruments that make extrapolation claims falsifiable: non-triviality criteria for every far bucket and a certified near-zero control stratum, plus a split-leakage audit.
- **Evaluation.** Across 384 runs, tree ensembles and neural models swap places: trees over engineered features lead every neural model at 2–3$\times$ the training width, where each target depends only on a depth-bounded lightcone, and trail them on the held-out family. Against an MLP on the same features, the trees' far-width lead comes mostly from three

width-dependent inputs (qubit count and observable positions), which trees saturate and the MLP extrapolates linearly (Xu et al., 2021); without them the MLP closes most of the gap. Graph networks fall further behind as width grows, even the NCGT with its width inputs removed: a size-generalization gap (Yehudai et al., 2021) on targets whose locality the circuit certifies.

- **Model selection.** Under the family hold-out, validation error carries no information about the test ranking (Spearman $\rho = 0.00$, against 0.99 in distribution), unlike the usual case where out-of-distribution accuracy tracks in-distribution accuracy (Miller et al., 2021); a model meant for a shifted deployment has to be tested under shift, which QuOOD enables.
- **Reference model.** On noisy-expectation prediction the NCGT is significantly ahead of every feature-based baseline in distribution; on mitigation it leads on held-out calibrated devices, and its hybrid variant beats both tree ensembles in distribution. Ablations credit per-block noise conditioning ($+71\%$ far-bucket error without it).
- **Diagnostics.** On the certified controls the transformers (ours and GraphGPS) emit more spurious correlation than their measured input carries, whereas the trees shrink it; aggregate error cannot separate this excess from missing signal. A collapse audit flags the constant predictors that mean absolute error rewards when labels concentrate near zero.
- **Release.** We plan to release the dataset, frozen splits, labels, code, and reference runs with a data card upon acceptance (Appendix A).

2 Related work

Learned quantum error mitigation and property prediction. Clifford data regression (Czarnik et al., 2021) fits a mitigation map on classically simulable near-Clifford circuits. Liao et al. (2024) train random forests and graph networks to map noisy to ideal expectation values, and report that simple models such as random forests often suffice in distribution. Other learned models, mostly graph networks and transformers, target mitigation (Bao et al., 2025a; Wang et al., 2026; Tousi et al., 2025; Patil et al., 2025), expectation-value prediction (Liu et al., 2026), fidelity estimation (Zlokapa & Gheorghiu, 2020; Wang et al., 2022; Vadali et al., 2024), performance bounds under drifting noise (Li et al., 2026), and magic estimation (Lipardi et al., 2026). QEM-Bench (Bao et al., 2025b) gives this line a shared testbed: 22 datasets of hundreds to a thousand circuits each, nine of which hold out deeper Trotter steps, larger gate counts, or unseen observables. Outside QEM-Bench, each work evaluates on its own corpus, i.i.d. or under light shift, and few report seed spreads or significance tests. QuOOD adopts the noisy-to-ideal input pairing of Liao et al. (2024) and differs from QEM-Bench in scale and design: four shift axes held out separately, far buckets with non-triviality criteria, a certified control stratum, and a seed-level statistical protocol.

Quantum circuit datasets. Table 1 positions QuOOD among existing circuit resources.

Graph out-of-distribution benchmarks. GOOD (Gui et al., 2022), DrugOOD (Ji et al., 2023), and WILDS (Koh et al., 2021) established covariate- and concept-shift splits for graph and real-world data. On many natural shifts, out-of-distribution accuracy tracks in-distribution accuracy

Table 1: Circuit resources for machine learning. "Prop. labels": circuit properties; "Noise cond.": noise-conditioned instances; "OOD": out-of-distribution hold-outs; "(✓)": single-axis hold-outs.

Resource	Scale	Prop. labels	Noise cond.	OOD
QASMBench (Li et al., 2023)	curated suite	–	–	–
MQT Bench (Quetschlich et al., 2023)	>70K circuits	–	–	–
SupermarQ (Tomesh et al., 2022)	application suite	–	–	–
MNISQ (Placidi et al., 2026)	4.95M instances	–	–	–
KetGPT (Apak et al., 2024)	generative (no fixed set)	–	–	–
ML-QEM data (Liao et al., 2024)	10^2–10^3 circ.	✓	✓	–
QEM-Bench (Bao et al., 2025b)	22 sets of 10^2–10^3 circ.	✓	✓	(✓)
QuOOD (ours)	99K circ. / 395K inst.	✓	✓	✓ (4 axes)

closely across models (Miller et al., 2021), and an out-of-distribution protocol is incomplete without a model-selection rule (Gulrajani & Lopez-Paz, 2021). Caro et al. (2023) prove out-of-distribution bounds for learning quantum dynamics, and classical learners that exploit locality provably learn gapped ground-state properties with sample complexity that falls from polynomial (Huang et al., 2022) to logarithmic (Lewis et al., 2024) to constant (Wanner et al., 2024) in system size; QuOOD tests whether locality also buys extrapolation across size.

3 THE QUOOD BENCHMARK

3.1 CIRCUITS, NOISE, AND LABELS

Circuits. Seven families cover the main uses of near-term devices: a structured family of GHZ state preparation, quantum Fourier transforms, and Clifford+T circuits; QAOA on random MaxCut graphs (Farhi et al., 2014); Trotterized Ising evolution; hardware-efficient variational ansätze (Kandala et al., 2017; Peruzzo et al., 2014); random layered circuits; Clifford circuits (Aaronson & Gottesman, 2004); and a wide-shallow brickwork family that extends to 40 qubits. Circuits are generated with Qiskit (Javadi-Abhari et al., 2024) from randomized parameterized templates.

Noise. Each circuit is simulated under one of three noise settings. N1 combines depolarizing, thermal-relaxation (T_1/T_2), and readout error with rates sampled per circuit from realistic ranges; N2 uses the full per-qubit, per-gate noise model of one of eight superconducting devices, taken from its public calibration data; N3 adds a coherent over-rotation on every two-qubit gate to N1.

Labels. Every label is computed by exact or error-bounded simulation chosen by regime: statevector simulation for ideal values, density-matrix simulation for noisy values up to a qubit threshold, adaptive trajectory sampling above it, and matrix-product-state simulation (Vidal, 2003) with a truncation-error guard for the wide-shallow family above the density-matrix threshold. Shot noise (4,096 shots) is applied on top of the noisy value, so the mitigation task has a well-defined measured input and target. Trajectory-sampled labels store their standard-error floor, and model errors on that subset are 3–11$\times$ the floor, so comparisons are not limited by label noise.

3.2 TASKS

Every instance pairs a circuit, a directed acyclic graph of gates on n qubits, with an observable O and the noise metadata of its setting. O is a single-qubit Z_i or a two-qubit Z_iZ_j (Pauli-Z operators with outcomes ± 1, so every target lies in $[-1, 1]$), with up to two of each per circuit.

Task A: noisy-expectation prediction. From these inputs, predict the noisy expectation value $\langle O \rangle_{\text{noisy}}$, the expected outcome of measuring O under the instance's noise.

Task B: error mitigation (main task). Given the same inputs plus a shot-limited measured estimate $\hat{y}_S$, predict the ideal value $y^\star = \langle O \rangle_{\text{ideal}}$. This is learned error mitigation in the formulation of Liao et al. (2024), and the natural point of comparison is zero-noise extrapolation (ZNE), which reruns the circuit at amplified noise and extrapolates to zero (Temme et al., 2017; Li & Benjamin, 2017; Kandala et al., 2019). The dataset includes digital-ZNE (Giurgica-Tiron et al., 2020; LaRose et al., 2022) baselines for 74.5% of instances (Appendix A).

Task C: compilation-cost prediction. Given a logical circuit and a target coupling map, predict post-routing depth and two-qubit-gate count (200,000 instances). Its labels need no quantum simulation and are cheap at any scale, which separates the representation question from the simulability constraint. The benchmark includes its split and reference runs of the four feature-based models (Appendix A), and the experiments below address the two simulation-bound tasks.

The metric throughout is mean absolute error against each task's target, $\mathrm{MAE} = \mathbb{E}\,|\hat{y} - y|$. Task B also reports two deployment quantities: the error-reduction ratio $\mathbb{E}|\hat{y} - y^\star| / \mathbb{E}|\hat{y}_S - y^\star|$ and the fraction of ZNE-covered instances on which the model is closer to $y^\star$ than ZNE.

3.3 SPLITS AND NON-TRIVIALITY CRITERIA

Five splits share one test protocol (sizes in Appendix A):

- **i.i.d.**: a random 80/10/10 split grouped by circuit, the control condition.
- **Depth**: within the QAOA, Trotterized-Ising, and GHZ/QFT/Clifford+T families, train on circuits of at most 8 layers and test on deeper ones, as when a Trotterization or ansatz is extended to more steps (composition in Appendix B).
- **Width** (qubit count): train on $\leq$12 qubits, test up to 40, the regime that motivates learned surrogates. Its *far bucket* (wide-shallow circuits at $\geq$25 qubits with two-qubit observables) sits at 2–3$\times$ the training width.
- **Family**: hold out the Trotterized-Ising family, as when a new algorithm class arrives.
- **Device** (noise hold-out): test on noise the model never saw, namely two of the eight calibrated devices and the over-rotation setting N3, as when a model moves to another fleet or to a drifted device (Senapati et al., 2023; Li et al., 2026).

An extrapolation bucket is only meaningful if it cannot be solved by predicting zero. We therefore require of each far bucket that (i) at least 25% of instances carry signal $|y^\star| > 0.1$, and (ii) the best model beats the constant-zero predictor by at least 30%. Both far buckets pass (Appendix B); the depth-split one is restricted to the GHZ and Trotter circuits, because the deep QFT and Clifford+T circuits have almost-all-zero labels. The Clifford family, whose ideal labels are -1, 0, or $+1$ and almost all zero, is a classification sub-benchmark excluded from the regression tables.

3.4 NEAR-ZERO CONTROLS, LEAKAGE AUDIT, AND REPORTING PROTOCOL

A measured qubit can be influenced only by the gates in its reverse lightcone, those linked to it through earlier gates, and in a 40-qubit shallow circuit most random qubit pairs have disjoint lightcones. In a brickwork circuit of depth D, the reverse lightcone of a measured qubit spans at most D neighbors to each side, so observables Z_i and Z_j at circuit distance $\delta > 2D$ (index separation along the chain, or ring distance for ring circuits) depend on disjoint sets of gates. Their joint state is then a product state, and $\langle Z_i Z_j \rangle = \langle Z_i \rangle \langle Z_j \rangle$ exactly. The benchmark's noise settings preserve this relation because their channels act locally. Single-qubit expectations in shallow random circuits concentrate near zero, and vanish by symmetry in the QAOA and Trotter sub-families, so the product is near zero in turn.

QuOOD draws a quarter of the wide-shallow family's two-qubit observables at distance $\delta > 4$ (Appendix A). These distant pairs are the controls: a calibrated model must output near zero on them while keeping its signal on nearby pairs. In the far bucket 85% of them satisfy $\delta > 2D$ and factorize exactly, and the reference labels confirm that the whole stratum is near zero (mean $|y^\star| = 0.0014$; Appendix B).

Leakage audit. The frozen splits group by circuit identifier, and because GHZ and QFT generators are deterministic in the qubit count, some test rows repeat a training circuit's content. Removing them changes no model ranking (Appendix C).

Reporting protocol. A result on QuOOD reports test MAE on every split with the Clifford family excluded, the far-bucket MAE with its two non-triviality checks, the mean predicted magnitude on the control stratum, and the collapse audit of Section 5.6, over at least three seeds with cross-seed Welch tests, all computed from per-sample test predictions.

4 A NOISE-CONDITIONED GRAPH TRANSFORMER

The reference architecture, the noise-conditioned graph transformer (NCGT), treats a circuit as a DAG whose nodes are gates and whose edges follow qubit wires. Each block combines local message passing over wire edges with dense self-attention, in the style of GraphGPS (Rampášek et al., 2022). Two additions target what separates circuit-property prediction from generic graph regression: the target depends on a noise setting that is not part of the graph, and the graph has a direction, from preparation to measurement. Each addition tests a hypothesis: that noise should enter every

Table 2: Task B (error mitigation): test MAE ($\times 10^{-2}$, mean $\pm$ s.d. over 3–10 seeds; Appendix D); bold = best per split. The device column merges the held-out devices with the coherent setting; Appendix B separates them. One of the MLP's ten depth seeds fails to extrapolate (12.1; median 1.88). GraphGPS is our backbone without the lightcone mask, positional encoding, or conditioning (Appendix D). [a]Stopped after 12 of 60 epochs; "timeout": budget exceeded; "—": not run.

Model	i.i.d.	Depth	Width	Family	Device
Identity (output $\hat{y}_S$)	4.07	5.46	3.07	6.15	4.39
Constant zero	7.64	10.05	5.62	5.57	7.69
Ridge	3.80±0.00	5.67±0.00	3.28±0.00	6.41±0.00	4.21±0.00
Random forest (Breiman, 2001)	1.93±0.00	**2.17±0.01**	1.58±0.00	5.70±0.01	2.40±0.00
XGBoost (Chen & Guestrin, 2016)	2.14±0.00	2.24±0.00	1.66±0.00	5.52±0.00	2.74±0.00
MLP (features)	1.87±0.05	3.09±3.21	1.42±0.07	4.32±0.28	**2.22±0.13**
GCN (Kipf & Welling, 2017)	3.65±0.33	2.92±0.33	2.82±0.27	4.12±0.35	3.27±0.35
GIN (Xu et al., 2019)	3.00±0.26	2.68±0.24	2.24±0.45	4.36±0.61	3.48±0.22
GatedGCN (Bresson & Laurent, 2017)	2.84±0.19	2.67±0.30	2.12±0.12	4.58±0.42	3.29±0.29
ML-QEM GNN (Liao et al., 2024)	2.01±0.04	3.36±0.34	1.94±0.42	5.40±0.39	3.62±0.87
DAGNN (Thost & Chen, 2021)	timeout	2.48±0.13	2.17±0.16[a]	timeout	timeout
GraphGPS (Rampášek et al., 2022)	1.85±0.04	3.24±0.79	1.67±0.12	4.73±0.47	2.56±0.09
NCGT (ours)	1.87±0.17	3.26±0.22	1.49±0.14	**3.92±0.52**	2.48±0.19
GraphGPS + features	1.84±0.01	—	1.62±0.17	—	—
NCGT + features (ours)	**1.78±0.02**	3.71±0.14	**1.41±0.04**	4.14±0.44	2.51±0.06

block rather than only the readout, and that DAG structure is a useful prior. Section 5.5 confirms the first and finds no support for the second.

Noise conditioning. Circuits under different noise must map to different outputs, and noise acts throughout the computation: every gate's contribution to the measured expectation is attenuated by the error rates of the qubits it touches. The baselines receive the noise description as a graph-level vector concatenated at the readout, which can rescale the output but not the representation. We instead embed the noise metadata (setting, calibration vector) into a vector z and inject it into every block through feature-wise linear modulation (FiLM; Perez et al., 2018), which maps each block's node features h to $(1 + \gamma(z)) \odot h + \beta(z)$, so that noise can rescale features at every depth.

DAG positional encoding (DAG-PE). Attention is permutation-invariant, so nodes carry their topological depth and their distance to the measurement layer, giving the model access to circuit time in both directions (Thost & Chen, 2021). Attention also carries a learned lightcone (LC) key mask, a per-block additive penalty on keys outside the measured qubits' reverse lightcone.

Node embeddings combine gate type, gate parameters, qubit indices, and the gate's calibrated error rate and duration; the readout pools the nodes of the observable's qubits and all nodes, appends the noise vector and global circuit statistics (for mitigation also $\hat{y}_S$), and maps them to the prediction (Appendix D). A *hybrid* variant ("+ features" in tables and figures) concatenates the pooled representation with the 70 engineered features used by the feature-based baselines before the readout. The NCGT, its hybrid, and GraphGPS (the same backbone without the two additions) are the dense-attention models below, "the transformers" for short. A final LayerNorm (Ba et al., 2016) precedes pooling, and the head's last layer is zero-initialized. Section 5.6 shows that these two choices decide whether dense-attention training reaches a non-constant predictor at all.

5 Results

Each neural model trains under one fixed budget with at least three seeds (Appendix D); feature-based models use the benchmark's feature set. Significance statements are cross-seed Welch t-tests, ablation p-values are Holm-corrected, and $p \geq 0.05$ counts as a tie. The text reports the findings; Tables 2, 3, and 7 hold the full numbers, and Appendix D lists the test behind every comparison.

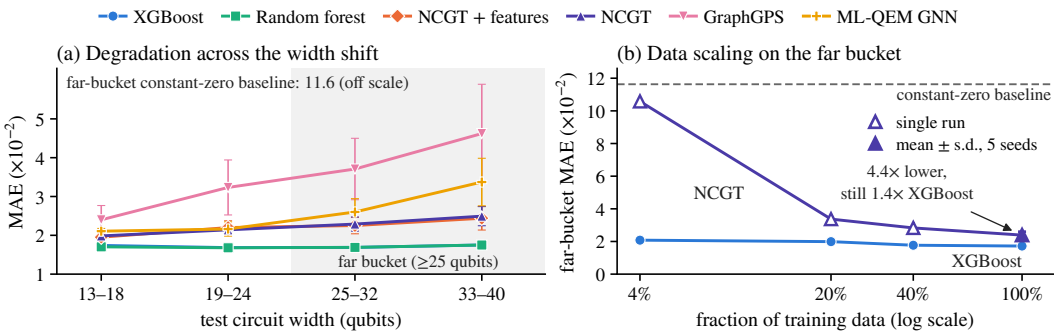

Figure 2: The width split's far regime (wide-shallow circuits, two-qubit observables). **(a)** MAE (mean ± s.d. over seeds) versus test width, training on $\leq$12 qubits: trees stay flat while the four graph models shown degrade (1.3–1.9×); shading marks the far bucket ($\geq$25 qubits). **(b)** Far-bucket MAE versus training fraction: the NCGT improves 4.4× but XGBoost stays ahead with 4% of the data (open markers: single NCGT runs).

5.1 Models that are close in distribution part ways under shift

In distribution the six best models are nearly tied (1.78–1.93×10^{-2}; Table 2), led by the hybrid NCGT, which beats both tree ensembles and ties the MLP. Under shift the lowest error moves to a different model on each axis: the random forest under depth shift (DAGNN is statistically tied), the NCGT on the held-out family, and the feature MLP under device shift. On the held-out family the tree ensembles fall to the level of predicting zero, consistent with trees memorizing family-specific input–output relations. On the width split as a whole the hybrid NCGT and the MLP tie for best, but most of its test rows lie below 25 qubits, and at far width the order reverses (Section 5.2). The device column merges two shifts with opposite winners: the NCGT and its hybrid lead on the held-out calibrated devices, the feature MLP on the coherent setting, a noise mechanism absent from training (Appendix B). A leaderboard built in distribution would therefore mislead: the random forest ranks only sixth there but first under depth shift.

Validation on the training distribution cannot correct for this. It ranks the models almost exactly as test error does in distribution (Spearman $\rho = 0.99$) but carries no information about the ranking on the held-out family ($\rho = 0.00$). A surrogate meant for a shifted deployment has to be chosen by testing under that shift.

The learned correction also pays off in absolute terms: the best model more than halves the error of the raw measurement in distribution and under depth and width shift, and beats per-instance ZNE on about 80% of the instances where ZNE is available, with one forward pass instead of additional circuit runs (Appendix D).

5.2 Far width: a size-generalization gap for graph models

Far width is where the benchmark's locality guarantee is strongest: at 2–3× the training width, every target still depends only on a depth-bounded lightcone (Section 3.4). It is also where the tree ensembles win. Both reach 1.72×10^{-2}, ahead of every graph model and significantly ahead of the NCGT and its hybrid (2.39 and 2.34; Table 7). The gap is not a failure to fit: inside the training width the dense-attention models match or beat the tree ensembles on validation (Appendix D). It opens outside the training range and grows with width (Figure 2a), and more data does not close it: XGBoost trained on 4% of the data already beats the NCGT trained on all of it (Figure 2b). For a surrogate deployed at 2–3× its training width, a tree ensemble is today the safer choice.

The explanation lies in the few inputs that leave the training range. Most engineered features are *local* statistics, such as gate counts inside the measured qubits' lightcone and error rates, that do not grow with width, and nearly all of them stay inside their training range at far width (Appendix B). The main exceptions are the qubit count and the two observable positions. A tree's response is constant beyond its last training split, so it saturates on these inputs, whereas a ReLU network extrapolates them linearly (Xu et al., 2021). Retrained without the three inputs, the trees stay ahead

Table 3: Task A (noisy-expectation prediction): test MAE ($\times 10^{-2}$, mean $\pm$ s.d., 3 seeds); bold = best. Graph models were run on the i.i.d. and width splits only (DAGNN not at all; compute budget, Appendix D); one of the MLP's three depth seeds fails to extrapolate. "—": not run.

Model	i.i.d.	Depth	Width	Family	Device
Ridge	8.02±0.00	42.74±0.00	10.66±0.00	7.24±0.00	8.43±0.00
Random forest	3.52±0.00	4.77±0.15	3.33±0.02	6.12±0.01	3.78±0.00
XGBoost	3.75±0.00	**4.62±0.00**	3.50±0.00	**6.04±0.00**	3.90±0.00
MLP (features)	3.45±0.01	6.10±4.26	3.11±0.05	6.05±0.02	**3.73±0.07**
GCN	3.82±0.02	—	3.70±0.19	—	—
GIN	3.88±0.08	—	3.53±0.01	—	—
GatedGCN	3.77±0.08	—	3.64±0.09	—	—
ML-QEM GNN	3.64±0.06	—	3.36±0.07	—	—
GraphGPS	3.44±0.27	—	3.30±0.03	—	—
NCGT (ours)	**3.02±0.03**	—	**3.02±0.04**	—	—

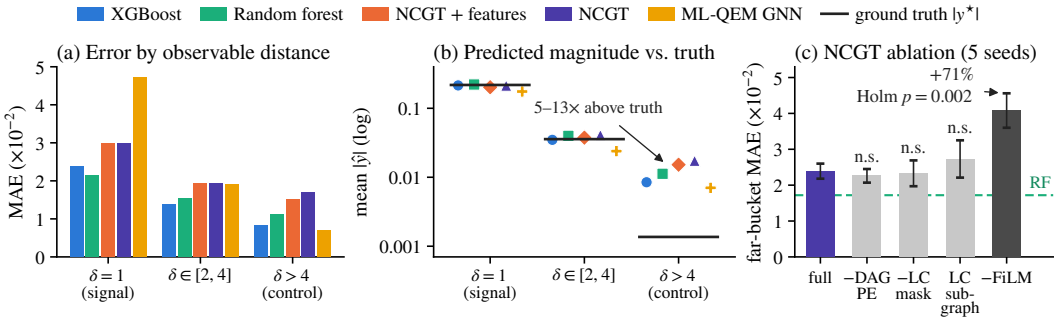

Figure 3: Far-bucket error by observable distance (Task B, width split). **(a)** MAE per stratum (mean over seeds); relative to the trees, the NCGT's error is largest on the controls. **(b)** Mean $|\hat{y}|$ against ground truth (black lines): on the $\delta > 4$ controls trees emit 0.8–1.1×10^{-2} and transformers 1.5–1.7×10^{-2}. **(c)** Far-bucket ablation of the NCGT (5 seeds, mean $\pm$ s.d., Holm-corrected): only removing noise conditioning matters; the dashed line marks the random forest.

of every graph model and the feature MLP closes three quarters of its gap to the random forest. The NCGT does not recover the same way: restricted to its lightcone subgraph, and then also stripped of its width inputs, it still trails the trees, and its error still rises with width in every seed (Table 7). Locality thus buys size extrapolation for learners whose inputs stay inside their training range, but not yet for the graph networks we tested.

5.3 Noisy-expectation prediction favors the transformer

Without a measured input to correct, the transformer pulls ahead. On Task A the NCGT has the lowest error on both splits where graph models were run, significantly so in distribution (3.02 against 3.45 for the best feature-based model; Table 3). On Task B the measured input $\hat{y}_S$ already carries most of the answer, so the learned part is a correction on which feature engineering is competitive; Task A must produce the value from circuit structure and noise alone, and there the NCGT justifies its cost. The margin comes from conditioning, not from the gate angles that graph models see and the engineered features omit. The message-passing baselines, ML-QEM among them, see the same angles yet cluster with the feature-based models, and GraphGPS, which differs from the NCGT only in the conditioning and in structural priors that Section 5.5 finds inert, trails it under width shift.

5.4 Certified controls expose invented signal

The control stratum separates two failures that aggregate error conflates: missing signal and invented signal (Figure 3). On nearest-neighbor pairs, trees and transformers alike recover the signal

magnitude. On the certified near-zero controls they part ways. The truth there averages 0.0014 in magnitude, while the shot-limited input $\hat{y}_S$ averages 0.0125. The trees shrink this input; the transformers output more than they receive (Figure 3b). On Task A, which has no measured input at all, the NCGT still predicts correlation for these pairs, so the excess is not shot noise passed through: at out-of-distribution widths the transformer invents correlations that the circuit's structure certifies as near zero (Appendix B). The failure weighs more than its share of rows: the controls hold 26% of the far bucket but 34% of the NCGT's gap to XGBoost.

5.5 Ablation: noise conditioning is the component that matters

Of the NCGT's components, only noise conditioning matters (Figure 3c). Without FiLM the far-bucket error rises by 71% (Holm-corrected $p = 0.002$), to the level of GraphGPS, which receives the noise vector only at the readout. The structural components (DAG positional encoding, the lightcone key mask, and restricting the input to the lightcone subgraph) add nothing measurable on any of the three splits tested (Appendix D); edges follow wires, so the wire graph already carries the reverse lightcone. FiLM's benefit is specific to shift: undetectable in distribution, largest at far width, significant on the held-out devices, and absent on the coherent setting, whose over-rotation parameter never varies in training (Table 11). Conditioning helps within the ranges the calibration vector covers in training, which the held-out devices stay inside (Appendix B).

5.6 Training stability is a first-order variable

With a bounded (tanh) readout and no terminal normalization, 44 of 91 dense-attention development runs collapse to a constant predictor. Two changes remove it: with a final LayerNorm and a zero-initialized last readout layer, none of the 133 full-data dense-attention runs collapses at the same learning rate (six are over-dispersed instead; Appendix D). A comparison that skips this check can mistake a training failure for an architectural one, so the benchmark includes a collapse audit (prediction spread below 15% of label spread) as a reporting requirement.

6 Limitations

QuOOD is simulation-based by necessity: exact labels exist only where simulation is possible, so its device-calibrated setting stands in for hardware, and conclusions about a specific device still call for on-device validation. Its far-width bucket covers the low-entanglement, Pauli-noise regime that can be labeled at 40 qubits; the depth split measures depth under a compound family shift, which Appendix B isolates on a restricted bucket; and the family hold-out uses a single family. All models run under fixed budgets without per-model tuning (Appendix D).

7 Conclusion

QuOOD turns a routine claim, that learned circuit-property models generalize beyond their training distribution, into something falsifiable, and a systematic evaluation shows where and how it breaks. Which model to trust depends on the shift: tree ensembles and neural models swap places between the far-width bucket and the held-out family, and validation error on the training distribution does not predict the ranking on that family. At far width, locality buys extrapolation across size for tree ensembles over local features, but not yet for the graph networks we tested, not even the NCGT with its width inputs removed. On the certified controls, spurious output accounts for a disproportionate share of the NCGT's gap to XGBoost. Among the ablated components only per-block noise conditioning matters, and its benefit is a shift effect: it carries the NCGT's far-width margin over its backbone and its lead on held-out devices.

The collapse audit also applies beyond circuits: under mean absolute error, any target concentrated near zero rewards a constant predictor. The results translate into per-shift guidance (Table 12). Three problems remain open: learning graph representations that generalize across size, closing the gap between i.i.d. and family-hold-out error (the benchmark's hardest axis), and suppressing spurious output on the certified controls. Once released, the dataset, splits, and labels will make each of them testable on the same footing.

AI USE STATEMENT

In this work, we used generative AI tools for refining and debugging code for author-designed models and data-generation pipelines, and for preparing job-submission scripts. We have not used generative AI tools for developing the conceptual framework, designing the benchmark and its experiments, proposing hypotheses, or designing the NCGT architecture. Nor did we use them to generate data: all circuits come from the authors' parameterized templates and all labels from classical simulation. Qualitative or thematic data analysis is not applicable to this work. Additionally, we used generative AI tools for writing plotting scripts for figures, polishing the writing of the paper, searching for and summarizing literature, and formatting references. While drafting the paper, we used them to refine English expression, split long sentences, and organize the text into sections. We have reviewed all AI-assisted work: the labels were audited against truncation-free simulation, every reported number was recomputed from the per-sample test predictions, and the authors verified the references. We take responsibility for the final content of this work, including text, claims, or artifacts produced with the aid of generative AI.

ETHICS STATEMENT

QuOOD contains no human-derived, personal, or proprietary data; all circuits are generated from parameterized templates and all labels are produced by classical simulation. Device noise parameters, where used, come from publicly documented calibration data. A data card documenting composition, provenance, known limitations, and intended use will accompany the release. The benchmark's intended use, evaluating surrogate models for quantum-circuit properties, carries no direct dual-use concern beyond quantum computing research generally.

REPRODUCIBILITY STATEMENT

Upon acceptance we plan to release the dataset with its frozen v1.0 splits and corrected labels, the code, and the reference runs with their per-sample test predictions, together with a data card and the scripts behind the tables and figures. Appendix A documents label generation, the label audit, and the planned release, and Appendix D lists every budget, seed count, and deviation. Results reproduce up to seed-level variation (GPU and CPU-thread nondeterminism).

REFERENCES

Scott Aaronson and Daniel Gottesman. Improved simulation of stabilizer circuits. *Physical Review A*, 70:052328, 2004.

Boran Apak, Medina Bandic, Aritra Sarkar, and Sebastian Feld. KetGPT: Dataset augmentation of quantum circuits using transformers. In *International Conference on Computational Science*, 2024.

Jimmy Lei Ba, Jamie Ryan Kiros, and Geoffrey E. Hinton. Layer normalization. *arXiv preprint arXiv:1607.06450*, 2016.

Tianyi Bao, Xinyu Ye, Hang Ruan, Chang Liu, Wenjie Wu, and Junchi Yan. Beyond circuit connections: A non-message passing graph transformer approach for quantum error mitigation. In *International Conference on Learning Representations*, 2025a.

Tianyi Bao, Ruizhe Zhong, Xinyu Ye, Yehui Tang, and Junchi Yan. QEM-Bench: Benchmarking learning-based quantum error mitigation and QEMFormer as a multi-ranged context learning baseline. In *International Conference on Machine Learning*, 2025b.

Beatrice Bevilacqua, Yangze Zhou, and Bruno Ribeiro. Size-invariant graph representations for graph classification extrapolations. In *International Conference on Machine Learning*, 2021.

Leo Breiman. Random forests. *Machine Learning*, 45(1):5–32, 2001.

Xavier Bresson and Thomas Laurent. Residual gated graph ConvNets. *arXiv preprint arXiv:1711.07553*, 2017.

Matthias C. Caro, Hsin-Yuan Huang, Nicholas Ezzell, Joe Gibbs, Andrew T. Sornborger, Lukasz Cincio, Patrick J. Coles, and Zoë Holmes. Out-of-distribution generalization for learning quantum dynamics. *Nature Communications*, 14:3751, 2023.

Tianqi Chen and Carlos Guestrin. XGBoost: A scalable tree boosting system. In *ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2016.

Piotr Czarnik, Andrew Arrasmith, Patrick J. Coles, and Lukasz Cincio. Error mitigation with Clifford quantum-circuit data. *Quantum*, 5:592, 2021.

Edward Farhi, Jeffrey Goldstone, and Sam Gutmann. A quantum approximate optimization algorithm. *arXiv preprint arXiv:1411.4028*, 2014.

Tudor Giurgica-Tiron, Yousef Hindy, Ryan LaRose, Andrea Mari, and William J. Zeng. Digital zero noise extrapolation for quantum error mitigation. In *IEEE International Conference on Quantum Computing and Engineering*, 2020.

Shurui Gui, Xiner Li, Limei Wang, and Shuiwang Ji. GOOD: A graph out-of-distribution benchmark. In *Advances in Neural Information Processing Systems (Datasets and Benchmarks Track)*, 2022.

Ishaan Gulrajani and David Lopez-Paz. In search of lost domain generalization. In *International Conference on Learning Representations*, 2021.

Hsin-Yuan Huang, Richard Kueng, Giacomo Torlai, Victor V. Albert, and John Preskill. Provably efficient machine learning for quantum many-body problems. *Science*, 377(6613):eabk3333, 2022.

Hsin-Yuan Huang, Yunchao Liu, Michael Broughton, Isaac Kim, Anurag Anshu, Zeph Landau, and Jarrod R. McClean. Learning shallow quantum circuits. In *ACM Symposium on Theory of Computing*, 2024.

Ali Javadi-Abhari et al. Quantum computing with Qiskit. *arXiv preprint arXiv:2405.08810*, 2024.

Yuanfeng Ji et al. DrugOOD: Out-of-distribution dataset curator and benchmark for AI-aided drug discovery – a focus on affinity prediction problems with noise annotations. In *AAAI Conference on Artificial Intelligence*, 2023.

Abhinav Kandala, Antonio Mezzacapo, Kristan Temme, Maika Takita, Markus Brink, Jerry M. Chow, and Jay M. Gambetta. Hardware-efficient variational quantum eigensolver for small molecules and quantum magnets. *Nature*, 549:242–246, 2017.

Abhinav Kandala, Kristan Temme, Antonio D. Córcoles, Antonio Mezzacapo, Jerry M. Chow, and Jay M. Gambetta. Error mitigation extends the computational reach of a noisy quantum processor. *Nature*, 567:491–495, 2019.

Thomas N. Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. In *International Conference on Learning Representations*, 2017.

Pang Wei Koh et al. WILDS: A benchmark of in-the-wild distribution shifts. In *International Conference on Machine Learning*, 2021.

Ryan LaRose et al. Mitiq: A software package for error mitigation on noisy quantum computers. *Quantum*, 6:774, 2022.

Laura Lewis, Hsin-Yuan Huang, Viet T. Tran, Sebastian Lehner, Richard Kueng, and John Preskill. Improved machine learning algorithm for predicting ground state properties. *Nature Communications*, 15:895, 2024.

Ang Li, Samuel Stein, Sriram Krishnamoorthy, and James Ang. QASMBench: A low-level QASM benchmark suite for NISQ evaluation and simulation. *ACM Transactions on Quantum Computing*, 4(2):1–26, 2023.

Jinyang Li, Samudra Dasgupta, Yuhong Song, Lei Yang, Travis Humble, and Weiwen Jiang. Computational performance bounds prediction in quantum computing with unstable noise. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 45(2):969–982, 2026.

Ying Li and Simon C. Benjamin. Efficient variational quantum simulator incorporating active error minimization. *Physical Review X*, 7:021050, 2017.

Haoran Liao, Derek S. Wang, Iskandar Sitdikov, Ciro Salcedo, Alireza Seif, and Zlatko K. Minev. Machine learning for practical quantum error mitigation. *Nature Machine Intelligence*, 6: 1478–1486, 2024.

Vincenzo Lipardi, Domenica Dibenedetto, Georgios Stamoulis, Evert van Nieuwenburg, and Mark H. M. Winands. Nonstabilizerness estimation using graph neural networks. *Machine Learning: Science and Technology*, 7(2):025027, 2026.

Yuxiang Liu, Fanxu Meng, Lu Wang, Yi Hu, Zaichen Zhang, and Xutao Yu. Output prediction of quantum circuits based on graph neural networks. *Frontiers of Physics*, 21(6):063201, 2026.

John P. Miller, Rohan Taori, Aditi Raghunathan, Shiori Sagawa, Pang Wei Koh, Vaishaal Shankar, Percy Liang, Yair Carmon, and Ludwig Schmidt. Accuracy on the line: On the strong correlation between out-of-distribution and in-distribution generalization. In *International Conference on Machine Learning*, 2021.

Srushti Patil, Dibyendu Mondal, and Rahul Maitra. Machine learning approach toward quantum error mitigation for accurate molecular energetics. *The Journal of Chemical Physics*, 163(2): 024129, 2025.

Ethan Perez, Florian Strub, Harm de Vries, Vincent Dumoulin, and Aaron Courville. FiLM: Visual reasoning with a general conditioning layer. In *AAAI Conference on Artificial Intelligence*, 2018.

Alberto Peruzzo, Jarrod McClean, Peter Shadbolt, Man-Hong Yung, Xiao-Qi Zhou, Peter J. Love, Alán Aspuru-Guzik, and Jeremy L. O'Brien. A variational eigenvalue solver on a photonic quantum processor. *Nature Communications*, 5:4213, 2014.

Leonardo Placidi, Ryuichiro Hataya, Toshio Mori, Koki Aoyama, Hayata Morisaki, Kosuke Mitarai, and Keisuke Fujii. MNISQ: A large-scale quantum circuit dataset for machine learning in the NISQ era. *Scientific Data*, 13(1):810, 2026.

Nils Quetschlich, Lukas Burgholzer, and Robert Wille. MQT Bench: Benchmarking software and design automation tools for quantum computing. *Quantum*, 7:1062, 2023.

Ladislav Rampášek, Mikhail Galkin, Vijay Prakash Dwivedi, Anh Tuan Luu, Guy Wolf, and Dominique Beaini. Recipe for a general, powerful, scalable graph transformer. In *Advances in Neural Information Processing Systems*, 2022.

Priyabrata Senapati, Zhepeng Wang, Weiwen Jiang, Travis S. Humble, Bo Fang, Shuai Xu, and Qiang Guan. Towards redefining the reproducibility in quantum computing: A data analysis approach on NISQ devices. In *IEEE International Conference on Quantum Computing and Engineering*, 2023.

Kristan Temme, Sergey Bravyi, and Jay M. Gambetta. Error mitigation for short-depth quantum circuits. *Physical Review Letters*, 119:180509, 2017.

Veronika Thost and Jie Chen. Directed acyclic graph neural networks. In *International Conference on Learning Representations*, 2021.

Teague Tomesh, Pranav Gokhale, Victory Omole, Gokul Subramanian Ravi, Kaitlin N. Smith, Joshua Viszlai, Xin-Chuan Wu, Nikos Hardavellas, Margaret R. Martonosi, and Frederic T. Chong. SupermarQ: A scalable quantum benchmark suite. In *IEEE International Symposium on High-Performance Computer Architecture*, 2022.

Seyed Mohamad Ali Tousi, Adib Bazgir, Yuwen Zhang, and G. N. DeSouza. Quantum error mitigation with attention graph transformers for Burgers equation solvers on NISQ hardware. *arXiv preprint arXiv:2512.23817*, 2025.

Avi Vadali, Rutuja Kshirsagar, Prasanth Shyamsundar, and Gabriel N. Perdue. Quantum circuit fidelity estimation using machine learning. *Quantum Machine Intelligence*, 6(1):1, 2024.

Guifré Vidal. Efficient classical simulation of slightly entangled quantum computations. *Physical Review Letters*, 91:147902, 2003.

Hanrui Wang, Pengyu Liu, Jinglei Cheng, Zhiding Liang, Jiaqi Gu, Zirui Li, Yongshan Ding, Weiwen Jiang, Yiyu Shi, Xuehai Qian, David Z. Pan, Frederic T. Chong, and Song Han. QuEst: Graph transformer for quantum circuit reliability estimation. In *IEEE/ACM International Conference on Computer-Aided Design*, 2022.

Huaxin Wang, Xinge Wu, Jiajun Liu, Ruiqing He, Jiandong Shang, Hengliang Guo, and Qiang Chen. Scalable quantum error mitigation with physically informed graph neural networks. *arXiv preprint arXiv:2604.16815*, 2026.

Marc Wanner, Laura Lewis, Chiranjib Bhattacharyya, Devdatt Dubhashi, and Alexandru Gheorghiu. Predicting ground state properties: Constant sample complexity and deep learning algorithms. In *Advances in Neural Information Processing Systems*, 2024.

Keyulu Xu, Weihua Hu, Jure Leskovec, and Stefanie Jegelka. How powerful are graph neural networks? In *International Conference on Learning Representations*, 2019.

Keyulu Xu, Mozhi Zhang, Jingling Li, Simon S. Du, Ken-ichi Kawarabayashi, and Stefanie Jegelka. How neural networks extrapolate: From feedforward to graph neural networks. In *International Conference on Learning Representations*, 2021.

Gilad Yehudai, Ethan Fetaya, Eli Meirom, Gal Chechik, and Haggai Maron. From local structures to size generalization in graph neural networks. In *International Conference on Machine Learning*, 2021.

Alexander Zlokapa and Alexandru Gheorghiu. A deep learning model for noise prediction on near-term quantum devices. *arXiv preprint arXiv:2005.10811*, 2020.

A Dataset details

Scale and splits. 99,386 circuits (about 20,000 CPU-hours of simulation); 394,684 (circuit, observable, noise) instances for Tasks A and B; 200,000 compilation instances for Task C, with reference runs of the four feature-based models (Table 6); the random forest is best, at a test MAE of 1.6 layers for post-routing depth and 1.3 gates for two-qubit-gate count ($R^2 \geq 0.99$). Table 4 lists the split sizes. Regression tables exclude the Clifford classification family throughout (Section 3.3).

Table 4: Split sizes: circuits per partition (Clifford included) and Task B test instances (excluding the Clifford family and the circuits held out by the label audit). Totals differ because the depth split uses only the QAOA, Trotterized-Ising, and GHZ/QFT/Clifford+T families and the family split excludes the Clifford family; each shift split also leaves one training-side circuit unassigned by the rounding of its 90/10 train/validation fractions.

Split	Train	Val	Test	Test instances
i.i.d.	79,508	9,938	9,940	34,315
Depth	20,959	2,328	21,704	86,128
Width	38,432	4,270	56,683	176,072
Family	65,104	7,233	17,118	67,917
Device	70,883	7,875	20,627	80,999

Families. GHZ preparation, QFT, and Clifford+T circuits (the structured family); QAOA on Erdős–Rényi and Barabási–Albert MaxCut graphs; Trotterized transverse-field Ising evolution; hardware-efficient ansätze; random layered circuits; Clifford circuits (the classification family; 98.5% of its

ideal labels are zero); and wide-shallow brickwork circuits of 6–40 qubits and at most 10 layers (Huang et al., 2024). The wide-shallow family is 73.5% shallow ring-QAOA and shallow Trotter circuits from the parent families' generators, the ring being the topology that keeps the matrix-product-state bond dimension bounded, and 26.5% random brickwork. Its two-qubit observables are drawn 50% from interaction-graph edges, 25% at distance 2–4, and 25% at distance above 4 (the controls of Section 3.4). The other regression families span 4–20 qubits. For algorithmic families initialized in uniform superposition with a symmetric Hamiltonian, single-qubit $\langle Z \rangle$ is exactly zero; such rows are excluded from the width far bucket, which uses two-qubit observables only.

Table 5: Circuit-store composition: circuits archived per family and circuits used by the v1.0 property tables. In the five 4–20-qubit families the difference is the 50,000 separate circuits of the compilation task (Task C), plus one QAOA circuit without labels; in the wide-shallow family it is the 613 circuits rejected by the matrix-product-state guards.

Family	Archived	In v1.0 tables
Random layered	29,636	17,044
Trotterized Ising	29,567	17,118
QAOA (MaxCut)	24,801	14,896
Hardware-efficient ansätze	20,427	12,896
Structured (GHZ, QFT, Clifford+T)	20,502	12,978
Wide-shallow brickwork	15,137	14,524
Clifford (classification)	9,930	9,930
Total	150,000	99,386

Label generation. Ideal values by statevector simulation, so the Task B targets of the 4–20-qubit families are exact; noisy values by density-matrix simulation up to 14 qubits and adaptive trajectory sampling at 15–20 qubits, where trajectories are added until every observable's standard error falls below half the shot-noise standard deviation at 4,096 shots (0.0078 at zero expectation), capped at 2,000. The 18.1% of trajectory rows that hit the cap have a median standard error of 0.87 times that target; every row stores its standard error, and model errors on trajectory-labeled rows are several times that error (Section 3.1). The wide-shallow family above 14 qubits uses matrix-product-state simulation for both values and rejects any circuit whose discarded weight exceeds $0.5/\sqrt{4096}$ or whose projected simulation cost exceeds a fixed budget (613 circuits in total). On the far bucket the noisy runs discard no weight, and the ideal runs discard none on 99.6% of rows and at most 2.5×10^{-7} on the rest; on the $\delta > 4$ controls, whose near-zero values follow analytically for the 85% of pairs with $\delta > 2D$, the figures are 99.3% and 2.5×10^{-7}. The Clifford and wide-shallow families use setting N1 restricted to depolarizing and readout channels, the noise model their stabilizer and matrix-product-state simulators support. Circuits are simulated on their own qubit indices without routing to a device; under N2 the backend's gate errors act on the qubits and couplings the device defines, and its readout error on every measured qubit. ZNE baselines are computed by digital gate folding at $3\times/5\times$ with linear extrapolation (folded circuits use 200 trajectories above the density-matrix threshold), covering 74.5% of instances (all but the Clifford family and the circuits whose folds were not simulated, mostly at the top of each simulator's qubit window).

Label audit. Qiskit Aer's default qubit truncation returns wrong expectation values for circuits in which some qubit is not linked by two-qubit gates to the measured ones. An audit against truncation-free simulation found such labels on 2.5% of the instances, most of them in the wide-shallow family (37% of its random brickwork sub-family). The planned release contains corrected labels for them, and every number in this paper excludes the affected circuits from the test sets (26% of the pre-audit far bucket, at most 1% of the depth, family, and device test sets; the training sets keep them, 0.8% of the width split's training rows). Rankings on the full and the cleaned test sets agree.

Compilation labels (Task C). Qiskit 1.4's transpiler at optimization level 3 maps each logical circuit (4–12 qubits) into the $\{R_z, \sqrt{X}, X, \mathrm{CX}\}$ basis on each of four coupling maps sized to the circuit (linear chain, square grid, heavy-hex, all-to-all); the label is the post-routing depth and two-qubit-gate count averaged over five transpiler seeds, with the seed spread stored.

Planned release. The planned data release comprises the circuit store (OpenQASM 3; Table 5), sample tables for all three tasks, the frozen split definitions, generation metadata, the reference runs

Table 6: Task C (compilation cost) on the i.i.d. test split: MAE in layers (post-routing depth) and in gates (two-qubit-gate count), with R^2; the MLP is mean $\pm$ s.d. over 3 seeds; the other models are reported from one run each; bold = best MAE.

	Post-routing depth		Two-qubit gates	
Model	MAE	R^2	MAE	R^2
Ridge	21.65	0.649	27.71	0.022
Random forest	**1.62**	0.993	**1.29**	0.997
XGBoost	1.89	0.993	1.46	0.997
MLP (features)	2.86±0.55	0.989	2.80±0.63	0.993

with their per-sample test predictions, and a data card documenting composition, provenance, known limitations, and intended use; we intend to use the MIT license for the code and CC BY 4.0 for the data. Data generation is deterministic in (seed, circuit index), so the corpus can be regenerated or extended from the generation configurations, and the frozen v1.0 splits keep results comparable.

B FAR BUCKETS AND SPLIT DECOMPOSITIONS

Width-split far bucket. Main-text numbers use the full far bucket: wide-shallow circuits, $\geq$25 qubits, two-qubit observables, 9,721 instances (843 from random brickwork) from 4,882 circuits. It passes both non-triviality criteria: 36.8% of its instances carry signal, and the best model is 85% below constant zero. Restricting to the QAOA and Trotter circuits (8,878 instances, 38.8% signal) changes no conclusion: the trees remain significantly best ($p \leq 0.006$ against the next two models) and the order of the six leading models is unchanged.

Feature ranges under width shift. On the far bucket, 64 of the 70 engineered features lie inside their training range on every row. The qubit count leaves it on every row, the two observable positions on 60–64% of rows, the lightcone's qubit count on 11%, and two noise-vector fields on fewer than 0.1%. On each of these a tree's response is constant beyond its last training split, so the prediction saturates rather than diverging. Retraining with the same fit procedure after removing the qubit count and the observable positions gives far-bucket MAE 2.05 (XGBoost) and 1.74 ± 0.03 (random forest, three seeds); removing all six features with any out-of-range row gives 1.87 and 1.84 ± 0.02, against 1.72 for both trees with the full set and 2.34 for the hybrid NCGT. The feature MLP, retrained the same way (three seeds), reaches 2.02 ± 0.10 without the qubit count and positions and 2.15 ± 0.17 without all six, against 2.92 ± 0.27 on the full set in the same procedure (2.87 for the reference runs of Table 7); a ReLU network extrapolates an out-of-range input linearly (Xu et al., 2021) where a tree saturates it.

Control stratum. On the far bucket's $\delta > 4$ controls the reference labels have median $|y^\star| = 0$, 96% of them lie below 0.01, and their mean is 0.0014. At $\delta = 1$ the models' mean $|\hat{y}|$ is 0.19–0.22 against a true 0.22; on the controls the trees emit 0.008–0.011 and the transformers 0.015–0.017, against a shot-limited input of 0.0125. On Task A, whose target is the noisy value (truth 0.0008 on these controls), the NCGT emits 0.026. By mechanism, the controls comprise 2,301 pairs from the QAOA and Trotter sub-families, where symmetry fixes the single-qubit terms to zero and the reference labels average 0.0002 in magnitude, and 238 pairs from random brickwork (mean 0.0126). The transformers' spurious output is not a family effect: on the symmetric QAOA and Trotter pairs the NCGT, its hybrid, and GraphGPS emit 0.015–0.017 and the trees 0.007–0.011; on the random pairs 0.017–0.025 against 0.014–0.018. Regressing each model's control-stratum output on the shot-limited input $\hat{y}_S$ (mean magnitude 0.0125 on these rows) gives slopes of 0.7–0.9 for the trees, 0.25 for the MLP, and 1.0–1.1 for the NCGT and GraphGPS (0.8 for the hybrid).

Sample efficiency. Table 8 lists the numbers behind Figure 2b.

Depth split. The depth threshold (the median, 8 layers) leaves every QAOA circuit on the training side, and half of the deep test set consists of QFT and Clifford+T circuits whose deep-regime labels are almost all zero (signal fractions 0.0% and 0.5%). The far bucket is therefore defined on the GHZ and Trotter circuits (35,255 instances at depths 13–20, 26.3% signal), where both non-triviality criteria hold (the best model is 80% below constant zero). The depth column of Table 2 uses the full

Table 7: Width-split far bucket, all models and ablation variants: MAE ($\times 10^{-2}$, mean $\pm$ s.d. over seeds) and ratio to XGBoost (from unrounded means). [a] 12 of 60 epochs.

Model	MAE	Seeds	×XGBoost
XGBoost	1.72±0.00	3	1.00
Random forest	1.72±0.02	3	1.00
NCGT −DAG-PE	2.26±0.19	5	1.31
NCGT −LC mask	2.33±0.36	5	1.36
NCGT + features (ours)	2.34±0.26	5	1.36
NCGT (ours)	2.39±0.21	5	1.39
GraphGPS + features	2.50±0.43	3	1.45
NCGT LC subgraph	2.73±0.52	5	1.59
NCGT −width inputs	2.78±0.65	3	1.62
Ridge	2.83±0.00	3	1.64
MLP (features)	2.87±0.18	3	1.67
ML-QEM GNN	2.97±0.47	3	1.73
DAGNN[a]	3.52±0.40	3	2.05
GatedGCN	3.84±0.96	3	2.23
NCGT −FiLM	4.08±0.48	5	2.37
GraphGPS	4.15±1.00	5	2.41
GIN	4.28±0.41	3	2.49
GCN	5.28±1.02	3	3.07
Constant zero	11.63	–	6.77

Table 8: Far-bucket MAE ($\times 10^{-2}$) versus training fraction. Subsampled NCGT points are single runs, all above the collapse threshold (prediction-to-label spread 0.88–1.00 on the Clifford-excluded test set); XGBoost is deterministic given the data.

Training fraction	4%	20%	40%	100%
XGBoost	2.08	1.99	1.77	1.72
NCGT	10.58	3.37	2.83	2.39
NCGT seeds	1	1	1	5

test set, whose family composition differs from the training side (no QAOA), so it mixes family shift into the depth axis; for GHZ and QFT, whose generator depth grows with width, it also mixes in width shift. Figure 4 traces error along the depth axis on the restricted bucket. Every model degrades with depth between the shallowest and deepest test bins: DAGNN, the bucket's best model, and the trees by 1.2–1.4×, the other message-passing baselines (not plotted) by about 1.3×, the MLP by 1.6×, and the dense-attention models and ML-QEM by 1.6–2.0×.

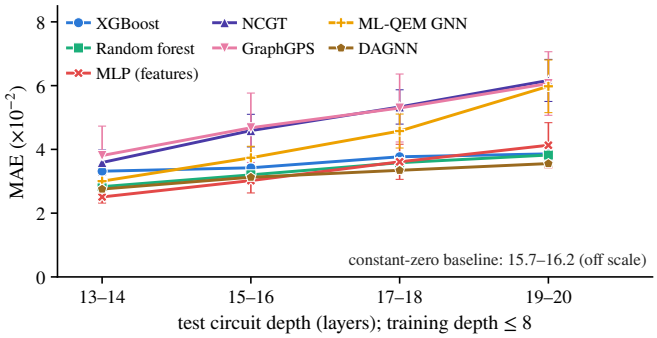

Figure 4: Depth split, restricted far bucket (GHZ and Trotter circuits; training depth $\leq$8 layers): test MAE by depth bin, mean $\pm$ s.d. over seeds (3; 5 for NCGT and GraphGPS, 10 for the MLP). The constant-zero baseline lies at 15.7–16.2 $\times$ 10^{-2} in every bin; the MLP's outlying depth seed (Table 2) peaks at 6.0 here.

Family and device splits. Table 9 splits the family and device columns of Table 2. The held-out Trotterized-Ising family shares its generator with the shallow Trotter circuits of the wide-shallow family, which have at most 5 steps; Table 9 divides the held-out family at 10 steps, so its deeper half has no counterpart of comparable depth in training, and the tree-versus-neural ordering is the same on both halves except that ML-QEM ties XGBoost on the deep half. The device split merges two shifts. On the two held-out calibrated devices the NCGT and its hybrid are best (1.59 and 1.57; for the NCGT, $p = 0.002$ against the MLP, the strongest feature-based model there, 0.0001 against the random forest, and 0.003 against GraphGPS). On the coherent over-rotation setting, whose parameter is constant in every training row, the feature MLP leads the models of Table 9 (2.40; $p = 0.02$ against the NCGT, 0.001 against GraphGPS). There the NCGT without FiLM (2.60 ± 0.12) sits below the full model (2.82 ± 0.27), though not significantly ($p = 0.15$). The held-out devices' calibration vectors fall inside the training range on every field of the noise vector, whereas the coherent setting activates two fields, its over-rotation angle and its setting indicator, that never vary in training. This second part holds 73% of the merged column's rows and sets its ranking.

Table 9: Task B MAE ($\times 10^{-2}$, mean $\pm$ s.d. over seeds) on the two halves of the family hold-out and the two parts of the device hold-out; bold = best per column; DAGNN timed out.

	Family (Trotterized Ising)		Device	
Model	$\leq$10 steps	$>$10 steps	Held-out devices	Coherent setting
Ridge	4.79±0.00	8.00±0.00	3.47±0.00	4.50±0.00
Random forest	3.96±0.00	7.40±0.01	1.79±0.00	2.62±0.00
XGBoost	3.94±0.00	7.07±0.00	1.91±0.00	3.05±0.00
MLP (features)	3.61±0.43	**5.01±0.18**	1.73±0.06	**2.40±0.17**
GCN	2.94±0.26	5.29±0.44	2.76±0.49	3.46±0.30
GIN	2.95±0.57	5.74±0.66	2.59±0.46	3.81±0.46
GatedGCN	3.27±0.20	5.87±0.64	2.64±0.25	3.54±0.33
ML-QEM GNN	3.63±0.33	7.13±0.44	1.98±0.32	4.24±1.08
GraphGPS	3.29±0.40	6.15±0.53	1.75±0.07	2.86±0.12
NCGT (ours)	**2.67±0.36**	5.15±0.68	1.59±0.03	2.82±0.27
NCGT + features (ours)	2.82±0.34	5.43±0.54	**1.57±0.01**	2.86±0.09

C LEAKAGE AUDIT

GHZ and QFT generators are deterministic functions of the qubit count, so the store's 150,000 circuit identifiers (of which the property tables use 99,386) resolve to 136,332 distinct circuit texts. The v1.0 splits group by identifier, which separates identifiers but not content: 9.9% of the i.i.d. regression test rows use a circuit whose text also appears in training, and 11.6% of the device-hold-out test rows share circuit content, none sharing the noise vector. The depth, width, and family splits are unaffected (their split axes are constant within duplicate groups). Table 10 reports Task B on the deduplicated test sets: the leading models' MAE rises by 10–13%, and no ranking changes; in particular the hybrid NCGT still significantly beats the random forest in distribution ($p = 0.0001$) and remains tied with the MLP ($p = 0.08$).

D TRAINING PROTOCOL AND DEVIATIONS

Features. The feature-based models use 70 engineered features per instance: ten gate-type counts; fifteen circuit statistics (log gate count, depth, layered depth, width, two-qubit-gate count, adjacent two-qubit-gate pairs, mean, maximum, and s.d. of the per-qubit gate and two-qubit-gate loads, observable arity and qubit positions); the same ten counts plus gate and qubit totals restricted to the depth-3 reverse lightcone of the measured qubits; the 32-dimensional noise vector; and the measured estimate $\hat{y}_S$, zeroed for Task A.

Budgets. All neural models minimize a Huber loss with threshold 0.1. Dense-attention models (NCGT, GraphGPS, and hybrids): 4 layers, 128 hidden units, 4 heads, batch 256, learning rate 10^{-4}, 40 epochs, patience 8. Message-passing models: learning rate 3×10^{-4}, batch 128, 60 epochs,

Table 10: Task B MAE ($\times 10^{-2}$) with content-duplicate test rows removed; bold = best per column; orderings match Table 2.

Model	i.i.d. (dedup.)	Device (dedup.)
Ridge	3.90±0.00	4.43±0.00
Random forest	2.13±0.00	2.70±0.00
XGBoost	2.34±0.00	3.05±0.00
MLP (features)	2.06±0.05	**2.45±0.15**
GCN	3.49±0.16	3.54±0.27
GIN	3.23±0.30	3.79±0.29
GatedGCN	3.10±0.17	3.59±0.21
ML-QEM GNN	2.23±0.04	4.03±0.87
GraphGPS	2.05±0.04	2.87±0.10
NCGT (ours)	2.07±0.17	2.79±0.21
GraphGPS + features	2.04±0.01	—
NCGT + features (ours)	**1.98±0.02**	2.80±0.08

patience 10. Feature-based models: random forest (500 trees; maximum depth from $\{\text{none}, 20, 40\}$ on validation), XGBoost (1,000 trees; learning rate from $\{0.03, 0.1\}$, early stopping on validation), and MLP (3×512 ReLU, dropout 0.1, AdamW at 10^{-3}, early stopping). Graph models use AdamW (weight decay 10^{-5}), 5% warm-up, cosine decay, and gradient-norm clipping at 5; all predictions are clipped to $[-1, 1]$; the hybrids standardize the engineered features on the training set and clip them at ± 5 standard deviations. The dense-attention models' smaller budget is a memory constraint: a 6-layer/256-unit configuration exceeds 80 GB on the largest graphs. DAGNN needs $\sim$12 hours per epoch on the largest splits: it trained fully on the depth split, for 12 of 60 epochs on the width split (Table 2), and exceeded the budget elsewhere. Task A graph models were run on the i.i.d. and width splits only. On the width split's validation set ($\leq$12 qubits) the dense-attention models reach 2.72–2.80×10^{-2} against 2.78 for the random forest, 2.83 for the MLP, and 2.99 for XGBoost. Seed counts: five for NCGT and GraphGPS on every Task B split, for NCGT + features on the i.i.d. and width splits, for the MLP on the device split (ten on the depth split), for every ablation variant on the width split (three for the variant without width inputs), and for −FiLM on the device split; three elsewhere. The 235 graph-model runs took about 5,300 GPU-hours in total (one A100 or H100 per run) and the 149 feature-based runs about 19 CPU-node-hours.

Architecture details. Each NCGT block is a GraphGPS-style layer: a GINE (edge-featured GIN) message-passing step over wire edges with two-dimensional edge attributes (a wire indicator and a flag for edges incident to a two-qubit gate), dense multi-head self-attention with the lightcone key mask of Section 4 (one learned scalar per block, initialized near 4), and a feed-forward step, all pre-LayerNorm. FiLM is applied to the block output with the (γ, β) of all blocks produced by a two-layer MLP from the 32-dimensional noise vector z. The GraphGPS baseline is this backbone with the lightcone mask, FiLM, and the DAG positional features switched off, without Laplacian or random-walk positional encodings. Node features are the gate-type one-hot, gate-parameter sine and cosine, a 16-dimensional sinusoidal encoding of the qubit indices (averaged over the gate's qubits), the gate's calibrated error rate and duration, topological depth, an observed-measurement flag, and the distance to the measurement layer; the −DAG-PE ablation zeroes the depth and distance features. Every graph model, including each ablated variant, receives the noise vector and the ten circuit statistics (normalized qubit count, gate depth, layer depth, log counts of all, one-, and two-qubit gates, gates per qubit, observable arity, and the two observable positions) at the readout, so the FiLM ablation compares per-layer conditioning against readout concatenation rather than against no noise information. The variant without width inputs (Section 5.2) runs on the lightcone subgraph and also zeroes the qubit-index encoding and the six width-dependent circuit statistics (normalized qubit count, the three log gate counts, and the two observable positions); its three runs come in addition to the 384 runs.

Stabilization. The collapse audit of Section 5.6 flags runs whose prediction spread falls below 15% of the label spread (Figure 5). The six full-data runs that are over-dispersed instead have spread ratios of 1.3–2.2 and remain in all tables.

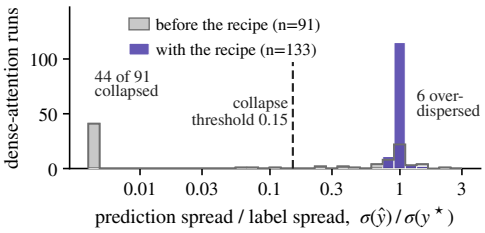

Figure 5: Collapse audit: prediction spread over label spread on the test set for every full-data dense-attention run (NCGT and GraphGPS, all splits and variants), before and after the stabilization recipe. Runs below 0.15 are constant predictors; 41 of the 44 collapsed pre-recipe runs have near-zero spread and sit in the leftmost bin. Pre-recipe runs are development runs outside the 384.

Statistics. All comparisons are cross-seed Welch t-tests on per-seed test MAE; reported s.d. are sample standard deviations over seeds; the four-way ablation uses Holm correction. A bootstrap over test instances (1,000 resamples) gives 95% intervals of ± 0.01–0.04×10^{-2} on the full splits and ± 0.03 on the far bucket (± 0.06 when whole circuits are resampled); no reported ranking changes within these intervals. Spearman correlation between validation and test MAE across the models of Table 2: 0.99 (i.i.d.), 0.92 (width), 0.76 (device), 0.42 (depth), and 0.00 (family). Table 11 reports the FiLM ablation by split; the $-$DAG-PE, $-$LC mask, and LC-subgraph variants (the last restricts the input graph to the measured qubits' reverse lightcone) were also run on the i.i.d. and device splits (three seeds): 1.86, 1.89, and 1.78 against 1.87×10^{-2} for the full model in distribution, and 2.73, 2.62, and 2.48 against 2.48 under device shift; no difference is significant.

Tests behind Section 5. Task B: the hybrid NCGT against both tree ensembles in distribution and on the width split, $p \leq 0.0005$ (surviving Holm correction); the random forest against DAGNN under depth shift, $p = 0.053$; the MLP against the best graph model under device shift, $p = 0.034$; on the held-out devices the NCGT and its hybrid against every feature-based model, $p < 0.003$; on the coherent setting the MLP against the NCGT, $p = 0.02$; on the far bucket both trees against the NCGT and its hybrid, $p \leq 0.006$, and against the NCGT restricted to its lightcone subgraph (2.73), $p = 0.01$; FiLM ablation, Table 11. Task A: the NCGT against the best feature-based model in distribution, $p = 0.001$, and against GraphGPS under width shift (3.30), $p < 0.001$. On the family hold-out the best model improves on constant zero by 30%, the smallest margin of any split.

Mitigation quality metrics. On the i.i.d. split the NCGT, its hybrid, and the random forest cut the unmitigated error to 0.459 / 0.438 / 0.474 of its size on the rows of Table 2, and beat ZNE on 77.3% / 78.5% / 77.3% of the ZNE-covered subset; on the shift splits, 76–82%.

Table 11: FiLM ablation by split: MAE ($\times 10^{-2}$); 5 seeds per arm, 3 for $-$FiLM on the i.i.d. split.

Split	Full	$-$FiLM	Welch p
Width far bucket	2.39	4.08	0.0005 (Holm 0.002)
i.i.d.	1.87	1.86	0.91
Device	2.48	2.36	0.23
held-out devices	1.59	1.71	0.0001
coherent setting	2.82	2.60	0.15

E GUIDANCE SUMMARY

Table 12 collects the practical guidance of Section 5 with the evidence behind each entry.

Table 12: Model class favored under each deployment shift (Task B unless noted; MAE $\times 10^{-2}$).

Deployment shift	Favored	Evidence
Wider, shallow circuits	tree ensembles	width far bucket: trees 1.72, an MLP without width inputs 2.02, hybrid NCGT 2.34 ($p \leq 0.006$ vs. trees)
Deeper circuits	no clear winner	depth split: trees 2.17–2.24 vs. 2.48 for the best graph model ($p = 0.053$); MLP median 1.88, one of ten seeds fails to extrapolate
New circuit family	neural models	family hold-out: NCGT 3.92, MLP 4.32, trees 5.52–5.70 (MLP vs. trees $p \leq 0.02$)
New calibrated device	NCGT or hybrid	held-out devices: 1.59 and 1.57 vs. 1.73 for the MLP ($p < 0.003$)
Unseen noise mechanism	feature MLP	coherent setting: 2.40 vs. 2.62 (random forest) and 2.82–2.86 (NCGT, hybrid, GraphGPS; $p \leq 0.04$)
No measured input (Task A)	NCGT	i.i.d. 3.02 vs. 3.45 ($p = 0.001$); width 3.02 vs. 3.11 (n.s.)
In distribution	hybrid NCGT or MLP	1.78 vs. 1.93 for the random forest ($p < 0.0001$); hybrid and MLP tied
Any architecture comparison	collapse audit first	44 of 91 pre-recipe dense-attention runs collapsed (Section 5.6)