PRESSURE–VORTICITY–REACTION GRAPH NETWORKS: STRUCTURED TOPOLOGICAL MEMORY AND SAFE LEARNED DYNAMICS

Anonymous authors
Paper under double-blind review

ABSTRACT

Learning constrained graph dynamics requires conservation, long-lived state, nonlinear response, and hard constraints. We introduce the Pressure–Vorticity–Reaction Graph Network (PVR-GN), which separates these roles. Pressure is a particular flow satisfying the current divergence; Vorticity is a persistent cycle-space state that stores topological memory without changing nodal conservation; and Reaction is a small causal proposal passed through a feasibility-preserving operator. We state the shared-budget allocation identity used by the cell, derive an allocation-free feasible set, and give fixed-active-set KKT sensitivities for the continuous projection branch. On a precommitted blind synthetic benchmark, PVR has zero observed rollout failures and favorable central comparisons with raw GNN/GRU baselines. A clustered reanalysis selects the 3,025-parameter PVR-Compact: its Blind-ID NMSE is lower than PVR-Strong by 0.001799 (95% CI $[0.000526, 0.003124]$); the pooled Blind-OOD difference is 0.000202 and its CI includes zero. In a separate control pilot, removing Vorticity or Reaction increases overloaded NMSE by 0.70706 or 0.02398; removing safe return yields 87.5% rollout failure under overload. A semantics-preserving same-A100 refactor reduces median paired Compact latency by 18.939%, although not uniformly. An independent static PGLib/MATPOWER DC-OPF study finds favorable held-out accuracy–compactness, numerically close scale transfer, and mixed rankings under severe congestion. In a separately frozen case57 test, a DC3-style DC-OPF adaptation (post-hoc) is natively feasible on both strata, while PVR-DC has lower dispatch MSE after common repair. PVR is a compact, auditable inductive bias for constrained graph dynamics, not a universal optimizer.

1 INTRODUCTION

Graph networks are increasingly used as learned simulators and controllers for physical and networked systems (Sanchez-Gonzalez et al., 2020; Pfaff et al., 2021; Li et al., 2018). Yet a generic latent state is often asked to do several mathematically different jobs at once: represent transport, remember history, adapt nonlinearly, and remain inside a constrained feasible set. When these roles are only encouraged by loss penalties or learned implicitly, failures can appear precisely under long horizons, congestion, or distribution shift.

For flow-like graph systems, a natural structural separation already exists. Nodal conservation depends on divergence. Circulations live in the null space of the incidence matrix and therefore change edge flows without changing divergence. Capacity constraints define a feasible set independently of any learned representation. These facts suggest an architecture in which the learned model is responsible only for the part of the dynamics that is not known analytically.

We propose the **Pressure–Vorticity–Reaction Graph Network** (PVR-GN). A PVR Cell decomposes the state into: **Pressure** P, a conservative particular flow; **Vorticity** $V = Z\omega$, a persistent cycle-space state; and **Reaction** R_θ, a causal learned modulation. The returned flow is produced by a feasibility-preserving layer rather than by the neural proposal alone. The forward computation is

$$x_t \to P_t + Z\omega_t \to R_\theta(x_t, P_t, Z\omega_t) \to \Pi_{\mathcal{F}}^{\text{safe}} \to f_t. \tag{1}$$

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

The design is intentionally hybrid: PVR spends parameters on the unknown reaction instead of relearning conservation, cycle geometry, and safety from data.

Our evidence has three layers. First, we derive the structural and constrained-flow properties of the cell. Second, we evaluate a precommitted dynamic synthetic benchmark designed to stress delayed cycle-memory recovery and constrained inference. Third, because a post-blind dependency analysis found that this controlled benchmark is strongly aligned with PVR's cycle-space inductive bias, we use a separately precommitted external validation based on PGLib/MATPOWER DC optimal power flow (DC-OPF). The external experiment has different topology, physics, solver, target construction, and task definition; its static nature intentionally disables any long-memory claim.

Our main contributions are:

- a Pressure–Vorticity decomposition in which $Z\omega$ stores divergence-free graph memory;
- an explicit shared-budget elimination identity and an allocation-free constrained set, with a pressure-safe fallback independent of learned weights;
- a small learned Reaction combined with piecewise-differentiable constrained inference and mechanistic P/V/R/safety decomposition;
- a precommitted blind synthetic evaluation separating RAW and safety-wrapped baselines, followed by Compact-versus-Strong and No-V/No-R closures that expose parameter efficiency and causal mechanism;
- a same-A100, semantics-preserving implementation comparison that quantifies latency reduction without turning it into an architecture-level speed claim;
- a post-blind task-alignment analysis and a separate PGLib/MATPOWER external campaign showing strong held-out accuracy, compactness, and scale transfer without universal-dominance claims.

2 Related Work

Graph learning for physical dynamics. Graph-network simulators and mesh-based models learn local interaction laws and can generalize across object counts, meshes, and initial conditions (Sanchez-Gonzalez et al., 2020; Pfaff et al., 2021). Recurrent graph architectures such as DCRNN combine message passing with temporal state for networked forecasting (Li et al., 2018). These methods demonstrate the value of graph inductive bias, but the hidden state typically mixes transport, memory, and correction in a learned representation. PVR instead assigns explicit graph-theoretic roles to these mechanisms and restricts learning primarily to Reaction.

Edge/Hodge representations. Edge signals admit a Hodge decomposition into gradient- and cycle-related components. Edge-space graph signal processing uses this geometry to smooth and denoise flows (Schaub and Segarra, 2018), while HodgeNet and related models exploit higher-order/edge Laplacians for learning on edge data (Roddenberry and Segarra, 2019; Yang et al., 2024; Hoppe and Schaub, 2024). PVR uses the same algebraic distinction but gives the cycle component a temporal role: $Z\omega_t$ is an explicit persistent state, not merely a feature subspace. The architecture then couples that state to capacity allocation and learned Reaction.

Differentiable constrained learning. Optimization layers such as OptNet differentiate through KKT systems (Amos and Kolter, 2017); differentiable convex optimization layers generalize this approach (Agrawal et al., 2019). DC3 explicitly addresses learning under equality and inequality constraints and includes optimal-power-flow applications (Donti et al., 2021). PVR is complementary: its constrained map is specialized to a pressure/cycle-space geometry and includes an analytic pressure-safe fallback. We therefore attribute guarantees to the returned constrained policy, not to the neural module itself. We also report a separately frozen *DC3-style DC-OPF adaptation (post-hoc)* on case57; it is not an official reproduction of DC3's AC-OPF experiment.

Learning for optimal power flow. Prior work learns OPF solution maps directly with dense networks, exploits predicted active sets, or uses graph neural networks to transfer across network structure (Deka and Misra, 2019; Pan et al., 2021; Owerko et al., 2020). These studies motivate learned surrogates and feasibility post-processing for power systems. Our DC-OPF campaign is instead a static external transfer test for the compact Pressure/Reaction bias; it is not the target application of PVR and does not validate Vorticity's temporal-memory mechanism.

Positioning. The novelty is not Hodge theory, recurrent graph learning, or differentiable optimization in isolation. It is the integration of (i) conservative Pressure, (ii) cycle-space Vorticity as a persistent state, (iii) learned causal Reaction, and (iv) an explicit feasibility mechanism in one graph cell. The external DC-OPF study additionally tests whether the resulting inductive bias transfers beyond the synthetic task that motivated the architecture.

3 Pressure–Vorticity–Reaction Graph Networks

3.1 Pressure and cycle-space memory

Let $G = (\mathcal{V}, \mathcal{E})$ have incidence matrix $B \in \mathbb{R}^{n\times m}$. Let $Z \in \mathbb{R}^{m\times r_c}$ span the cycle space, so $BZ = 0$. For a required nodal divergence s, choose a particular flow p with $Bp = s$. Every flow with the same divergence can be represented as

$$f = p + Z\omega, \qquad Bf = Bp = s. \tag{2}$$

Thus Vorticity can modify circulation while leaving conservation unchanged. The coordinates ω depend on the chosen basis, whereas the physical circulation $Z\omega$ does not: for any invertible T, if $Z' = ZT$ and $\omega' = T^{-1}\omega$, then $Z'\omega' = Z\omega$.

A minimal memory law is

$$\omega_{t+1} = \omega_t + w_t - \nu_t D_t \omega_t, \qquad D_t \succeq 0, \tag{3}$$

where w_t is the Reaction write/erase term. If $w_t = 0$ and $\nu_t = 0$, memory is held exactly. With $w_t = 0$ and $0 \le \nu_t \le 2/\lambda_{\max}(D_t)$, the coordinate energy $\|\omega_t\|_2^2$ is non-increasing; when $D_t = 0$, this step-size condition is vacuous. This coordinate energy is not the physical circulation energy $\|Z\omega_t\|_2^2$ unless the basis is orthonormal. Full statements and proofs are in Appendix A.

3.2 Shared-capacity reaction and safe fallback

Let $u_0 \ge 0$ be base edge capacities, $\beta > 0$ a shared augmentation budget, and $a \in \Delta_m = \{a \ge 0 : \mathbf{1}^\top a = 1\}$. The augmented capacity is $u_0 + \beta a$. In the dynamic model, directed channels carry nonnegative flow; for a nonnegative raw flow r, define deficits $d_r = [r - u_0]_+$ and $\chi_r = \mathbf{1}^\top d_r/\beta$. A signed physical edge can instead be represented by two directed channels or an absolute-value capacity set. The external DC-OPF study uses its separate signed branch-flow formulation.

The surrounding positive-simplex, ℓ_1-budget, and singly constrained resource-allocation geometry is standard (Duchi et al., 2008; Kiwiel, 2008). We state the elementary elimination used by PVR for completeness.

Lemma 1 (Shared-budget elimination)**.** *There exists $a \in \Delta_m$ with $r \le u_0 + \beta a$ iff $\mathbf{1}^\top [r - u_0]_+ \le \beta$. When feasible,*

$$a = \frac{d_r}{\beta} + (1 - \chi_r)b, \qquad b \in \Delta_m, \tag{4}$$

exactly covers all deficits.

The necessity follows by summing the required componentwise augmentation; sufficiency follows from eq. (4). Eliminating a yields the equivalent flow-only feasible set

$$\mathcal{F}(s) = \{f \ge 0 : Bf = s,\ \mathbf{1}^\top [f - u_0]_+ \le \beta\}. \tag{5}$$

This supports two complementary mechanisms. A joint instantaneous oracle projects $p + Z\omega$ onto eq. (5); a pressure-safe fallback first reserves enough budget to preserve the current pressure flow p, then allocates residual capacity to memory. Whenever the pressure flow itself satisfies the budget condition, the returned fallback is feasible regardless of Reaction weights. The guarantee belongs to the wrapper, not to R_θ.

3.3 Learned Reaction and differentiable constrained inference

The learned Reaction reweights a safe allocation rather than inventing capacity from scratch:

$$a_\theta = \mathrm{softmax}(\log(a_{\mathrm{PS}} + \varepsilon) + r_\theta)\,. \tag{6}$$

In the dynamic implementation, Reaction also controls write/erase of cycle memory. Unsafe or instantaneously unacceptable proposals can fall back to the pressure-safe action, yielding an anytime interpretation: a learned proposal may improve the safe base, but need not be trusted to maintain feasibility.

Using $f = p + Zz$, the native projection becomes a constrained problem in cycle coordinates. On a fixed active set its KKT system has the form

$$\begin{bmatrix} H & A^\top \\ A & 0 \end{bmatrix} \begin{bmatrix} z \\ \lambda \end{bmatrix} = \begin{bmatrix} -g \\ b \end{bmatrix}. \tag{7}$$

If this system is nonsingular and the active set is unchanged, implicit differentiation gives local derivatives with respect to smooth parameters. Globally, the projection branch is piecewise smooth because active-set switches can create nondifferentiable boundaries. The discrete proposal selector and accept/fallback wrapper remain outside this differentiation claim; we do not claim global smoothness of the full policy.

3.4 One PVR step and computational structure

A dynamic step can be summarized as: (1) compute a pressure particular flow p_t from the current transport requirement; (2) form the raw state $r_t = p_t + Z\omega_t$; (3) evaluate Reaction features and propose a reweighting/write action; (4) return the proposal if it satisfies the required checks, otherwise apply pressure-safe fallback/repair; and (5) update ω_{t+1} from the accepted circulation. This separation matters for ablation: removing Vorticity deletes persistent cycle memory; removing Reaction leaves an analytic safe controller; removing the wrapper tests the learned proposal without inherited safety.

The trainable parameter count can be independent of graph size when Reaction weights are shared, but the full algorithm is not size-free. An explicit dense cycle basis costs $O(mr_c)$ storage, and dense KKT solves can dominate runtime. We therefore report parameter efficiency and runtime separately rather than treating one as a proxy for the other.

Mechanistic auditability. A PVR decision exposes the structural origin of its output: Pressure transport, cycle-space memory, learned Reaction, and safety correction are separately observable. Useful diagnostics include Reaction magnitude, memory retention/sacrifice, active KKT constraints, and fallback events. This is structural explainability rather than a claim of complete causal interpretability.

4 Experimental Design

4.1 Controlled dynamic benchmark

The synthetic benchmark stresses delayed recovery of a cycle-space quantity under dynamic pressure/capacity perturbations. Final evaluation was precommitted before opening two seed domains: Blind-ID changes evaluation seeds within predefined in-distribution graph families, while Blind-OOD uses a disjoint seed domain and unseen graph-family/rank configurations. For late-query target y and prediction $\hat{y}$, the episode metric is

$$\mathrm{NMSE}(\hat{y}, y) = \frac{\mathrm{mean}((\hat{y} - y)^2)}{\max\{\mathrm{mean}(y^2), 10^{-12}\}}. \tag{8}$$

The statistical unit is the independent evaluation-seed $\times$ graph-instance cluster; paired uncertainty resamples clusters with replacement. We compare PVR-Strong, PVR-Compact, GNN and GRU baselines in RAW and common safety-wrapped forms. RAW denotes the learned proposal without the common safe-return wrapper; wrapped baselines use that same wrapper. RAW failure is reported separately from performance on successful RAW rollouts because the archived flag can include execution failure modes beyond measured residuals.

A post-blind dependency audit found that the late target is generated before policy rollout and is therefore policy-independent, but the task generator and teacher are strongly aligned with cycle-space memory. We consequently interpret this benchmark as *controlled mechanistic evidence*, not as sufficient external validity by itself.

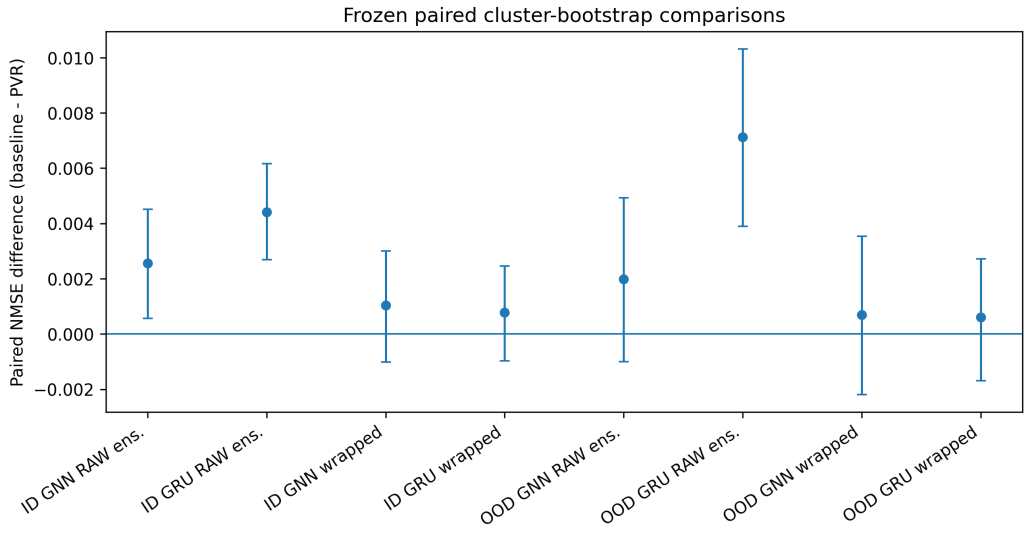

Figure 1: Paired blind comparisons, Δ = baseline NMSE − PVR NMSE; positive values favor PVR. Confidence intervals use the precommitted cluster bootstrap.

4.2 External PGLib/MATPOWER campaign

External validation uses PGLib-OPF v23.07 and MATPOWER 8.1 DC-OPF (Babaeinejadsarookolaee et al., 2019; Zimmerman and Murillo-Sanchez, 2025; Zimmerman et al., 2011). Phase A freezes perturbed oracle datasets; Phase B trains/selects only on IEEE 14/30/57 TRAIN/DEV; Phase C opens TEST once after model/checkpoint hashes are frozen. PVR-DC uses Pressure-derived physical context plus a small Reaction head; Vorticity memory is disabled because DC-OPF is static. Thus the external campaign tests topology/physics transfer, compactness, feasibility behavior, and scale/congestion shift—not long-memory transfer.

Three TEST groups are opened once: held-out typical 14/30/57, unseen 118-bus scale OOD, and API congestion variants 14/30/57/118. Accuracy is computed on the same oracle-successful scenarios for every model and is *not* conditioned on RAW model feasibility. Oracle-unsuccessful scenarios form a separate stratum. Multi-topology group summaries are equal-weight macro averages across cases, not pooled scenario averages. A common generator box/sum projection is applied to all RAW outputs; a common numerical line-feasibility repair is reported separately and is called *constraint-repaired*, not mathematically certified.

5 Blind Synthetic Results

On Blind-ID, the PVR-Strong ensemble improves late-query NMSE over the RAW GNN ensemble ($\Delta = 0.002554$, 95% CI $[0.000563, 0.004511]$) and RAW GRU ensemble ($\Delta = 0.004414$, CI $[0.002693, 0.006158]$). The RAW best-checkpoint contrasts are likewise positive: 0.002815 $[0.000706, 0.004780]$ for GNN and 0.005411 $[0.003335, 0.007538]$ for GRU. Against wrapped ensembles, the ID intervals include zero. On Blind-OOD, PVR improves over RAW GRU pooled ($\Delta = 0.007124$, CI $[0.003892, 0.010317]$); for RAW GNN pooled, $\Delta = 0.001989$ and the CI $[-0.001000, 0.004927]$ includes zero, so no pooled difference is detected. The wrapped pooled intervals also include zero. At horizon 128, the RAW GRU contrast is 0.014144 $[0.007830, 0.021016]$.

Reliability changes the interpretation of these accuracy numbers. PVR and wrapped baselines have zero observed rollout failures in 648 Blind-ID and 432 Blind-OOD episodes, whereas RAW GNN/GRU ensembles fail on roughly 20–24% of pooled episodes. The successful RAW outputs that are scored have machine-precision capacity/divergence residuals; the failure-rate endpoint is therefore kept separate rather than described as a universal feasibility violation rate.

Table 1: Blind synthetic summary. Pooled RAW deltas are paired on mutually successful RAW episodes; failure rates use all attempted episodes.

Comparison	Mean Δ	95% CI	RAW fail.
ID: GNN ens. − PVR	0.00255	[0.00056, 0.00451]	20.37%
ID: GRU ens. − PVR	0.00441	[0.00269, 0.00616]	23.61%
OOD: GNN ens. − PVR	0.00199	[-0.00100, 0.00493]	21.53%
OOD: GRU ens. − PVR	0.00712	[0.00389, 0.01032]	24.07%

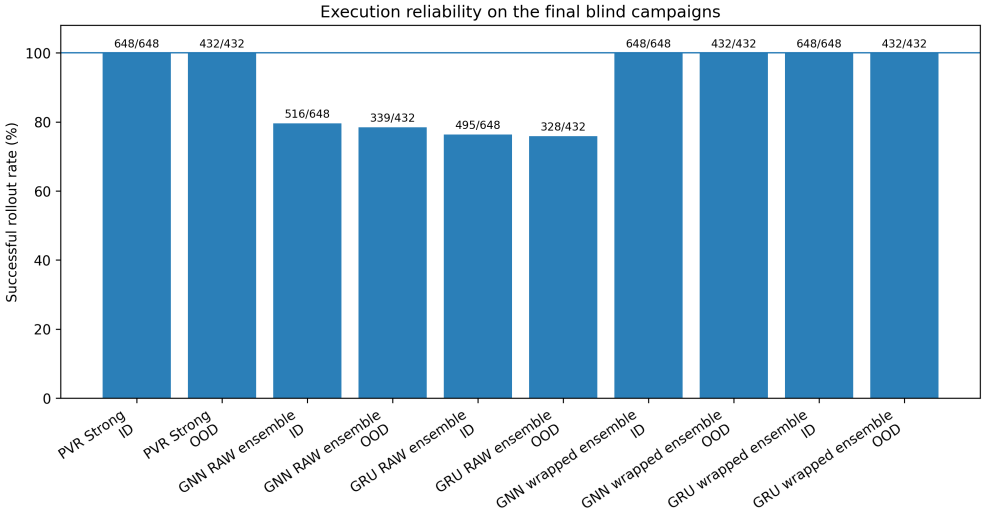

Figure 2: Rollout success rates computed from attempted blind episodes. PVR and safety-wrapped baselines complete every pooled ID/OOD episode; RAW learned baselines exhibit substantial execution failure rates. Exact counts are annotated and provided in the supplement.

PVR is also parameter-efficient: PVR-Strong has 10,657 trainable parameters, versus 38,657 for GNN-Strong and 57,601 for GRU-Strong; PVR-Compact has only 3,025. Thus the generic GNN/GRU baselines are 3.63×/5.40× larger than PVR-Strong and 12.8×/19.0× larger than PVR-Compact.

The within-family closure therefore selects PVR-Compact as the preferred parameter-efficient variant: it is better than Strong on Blind-ID, while the Blind-OOD difference is 0.000202 with CI $[-0.001889, 0.002267]$, which includes zero; Compact uses 71.6% fewer trainable parameters. Strong remains the predeclared comparator for the baseline contrasts above. The mechanistic controls are sharper: Full PVR wins all 54 No-V clusters and 50 of 54 No-R clusters under genuine overload. Removing Vorticity collapses the controlled memory mechanism; removing Reaction causes a smaller but statistically supported loss beyond the analytic pressure-safe controller.

A separate safety-removal control retains Reaction but removes certified non-worsening/fallback behavior. In the historical control pilot it has 189/216 runner-declared rollout failures under overload (87.5%), no failures in 432 nominal/stressed episodes, and 4,285/9,036 candidate steps (47.4%) worsen the objective. These are execution failures and loss of safe return, not measured capacity violations; successful outputs retain machine-precision residuals.

We do *not* equate parameter efficiency with runtime efficiency. A same-A100 engineering comparison evaluated the reference PVR-Compact path and a semantics-preserving factorization cache on 36 fresh non-blind episodes in 12 rank–horizon clusters. It found zero classification or step-count mismatches, zero maximum final-flow difference, and a median paired latency reduction of 18.939% (cluster-bootstrap 95% CI $[15.505, 21.032]\%$), corresponding to 1.234× median speedup; 94.4% of episodes were faster. The improvement is not uniform: at rank 64/horizon 128, only one of three episodes is faster. This is an implementation result, not evidence that PVR is intrinsically faster than GNN/GRU.

Table 2: Read-only clustered closures. For Compact–Strong, $\Delta = \mathrm{NMSE_C} - \mathrm{NMSE_S}$, so negative favors Compact. For controls, $\Delta = \mathrm{NMSE_{ablation}} - \mathrm{NMSE_{Full}}$, so positive favors Full PVR. Control-pilot results remain separate from the final blind campaign.

Contrast	Scope	Pairs/clusters	Mean Δ [95% CI]
Compact–Strong	Blind-ID	648/54	-0.001799 $[-0.003124, -0.000526]$
Compact–Strong	Blind-OOD	432/24	0.000202 $[-0.001889, 0.002267]$
No-V–Full	overloaded	388/54	0.707060 $[0.685708, 0.726184]$
No-R–Full	overloaded	388/54	0.023978 $[0.018757, 0.029356]$

6 Static External Validation

Model selection. Phase B uses three predetermined seeds per architecture and selects checkpoints only by macro DEV dispatch MSE on 14/30/57. The selected seeds are PVR-DC 7103, GNN 7201, and GRU 7302; three-seed ensembles are also fixed before TEST. PVR-DC has 3,361 trainable parameters, GNN 46,913, and GRU 41,729. PVR-DC's extra physical context (economic anchor, DC angles/flows, and local loading) is computed only from the scenario load and static network data, never from oracle P_g, P_f, or voltage-angle targets.

Endpoints. The primary external accuracy endpoint is generator-dispatch MSE normalized by base MVA. Secondary endpoints include branch-flow MSE, generation-cost regret, RAW line feasibility, repair success, repair magnitude, and runtime. Pairwise uncertainty uses 10,000 scenario-bootstrap replicates with a fixed bootstrap seed. For multi-case summaries, each case receives equal macro weight. Feasibility requires generator bounds, total-generation balance, and DC line limits within the precommitted numerical tolerance; the common repair enforces and rechecks these conditions.

6.1 Held-out typical grids

Among scenarios for which the MATPOWER oracle returns a successful solution, PVR-DC best has the lowest equal-weight macro raw dispatch MSE across held-out 14/30/57 cases: 5.76×10^{-4}, versus 8.18×10^{-4} for GRU best and 1.087×10^{-3} for GNN best. Paired scenario bootstraps against PVR exclude zero for both differences. The same ordering holds after the common constraint repair. PVR-DC uses 3,361 trainable parameters; the GRU and GNN baselines use 12.4$\times$ and 14.0$\times$ more, respectively.

6.2 Unseen 118-bus scale shift

Without retraining, PVR-DC and GNN best are numerically close in raw dispatch MSE (1.61450 versus 1.61446), while GRU best is 1.75392. The GNN–PVR difference is statistically detected but numerically tiny; without a predeclared equivalence margin, the supported claim is scale competitiveness rather than equivalence or superiority. All methods are RAW line-infeasible on all 150 oracle-successful 118-bus scenarios and become feasible after the common repair, showing that generalization of dispatch accuracy does not imply direct generalization of line feasibility.

6.3 Severe congestion shift

Congestion OOD produces a deliberately mixed result. GNN best has the lowest equal-weight macro raw dispatch MSE (0.451 versus 1.026 for GRU and 1.357 for PVR), whereas PVR best has the lowest macro raw generation-cost regret point estimate among best-single models (2.71% versus 3.94% for GRU and 12.12% for GNN). RAW feasibility is topology dependent: on 14-API, PVR best is feasible on 141/142 oracle-successful scenarios (99.3%); on 30-API, PVR is feasible on 104/111 (93.7%) while GNN best is 0/111; on 57-API, GNN best is 150/150 while PVR/GRU best are 0/150; on 118-API, all best models are 0/83. The common repair succeeds for every oracle-successful model/scenario evaluated.

Oracle availability is itself reported rather than hidden. Typical 14/30/57 have 150/150, 149/150, and 150/150 successful oracles; scale-118 has 150/150. Under API congestion, successful oracles are

Table 3: External best-single results. Multi-case rows are equal-weight macro averages across cases. MSEs are conditional only on oracle success, never on model RAW feasibility.

Group/model	Disp. MSE	Cost regret	RAW feas.	Params
Typical PVR-DC	**0.000576**	**0.090%**	73.9%	**3,361**
Typical GRU	0.000818	0.173%	**81.7%**	41,729
Typical GNN	0.001087	0.439%	78.8%	46,913
118 PVR-DC	1.61450	13.965%	0%	**3,361**
118 GRU	1.75392	14.178%	0%	41,729
118 GNN	**1.61446**	**13.965%**	0%	46,913
API PVR-DC	1.35669	**2.707%**	48.2%	**3,361**
API GRU	1.02617	3.941%	37.9%	41,729
API GNN	**0.45085**	12.119%	**48.8%**	46,913

Table 4: Separate *DC3-style DC-OPF adaptation (post-hoc)* on case57. PVR rows are frozen; repaired rows use the common repair. This is not part of confirmatory Table 3.

Stratum	Method/output	Dispatch MSE	Strictly feasible
Typical57	PVR-DC RAW	0.001627	121/150
	PVR-DC repaired	**0.000785**	150/150
	DC3-style native	0.945444	150/150
API57	PVR-DC RAW	4.686966	0/150
	PVR-DC repaired	**1.814662**	150/150
	DC3-style native	4.999352	150/150

142/150, 111/150, 150/150, and 83/150 for 14/30/57/118 respectively. Thus 114/600 API scenarios are oracle-unsuccessful and receive no model accuracy score. A dtype-safe recomputation confirms that for every model/case, the accuracy denominator equals the number of oracle-successful scenarios; RAW-infeasible model outputs remain in the accuracy set.

Post-hoc case57 comparison. DC3-style was trained self-supervised on the same 300 TRAIN/75 DEV case57 scenarios in PVR-DC's pool, selected without TEST access, and evaluated once on the already-open Typical57/API57 rows. PVR-DC also used labeled 14/30-bus data, so this is neither an official DC3 reproduction nor a matched-budget contest. DC3-style is natively feasible on 150/150 in each stratum; its correction stops after the mandated first step with zero displacement because RAW is already feasible. Under common repair, PVR-DC's paired dispatch-MSE advantage is 0.944659 (95% CI $[0.751890, 1.152039]$) on Typical57 and 3.184690 ($[2.991490, 3.374497]$) on API57. Appendix F gives both output conventions, costs, protocol boundaries, and hashes. Batched DC3-style and per-scenario PVR timings are not compared.

7 Discussion and Limitations

A key evaluation choice is to separate *representation quality* from *constraint handling*. RAW predictions expose what each learned architecture proposes after the common generator box/sum projection. Constraint-repaired results ask how much accuracy remains when every architecture receives the same external line-feasibility correction. The synthetic study uses the analogous RAW-versus-wrapped distinction. This prevents a generic baseline from being penalized merely because PVR contains a safety mechanism, while still allowing reliability itself to be measured as an architectural endpoint.

The combined evidence supports a narrower and more useful claim than universal benchmark dominance. PVR's structural separation yields reliable constrained dynamic behavior with a small learned Reaction, and the No-V/No-R controls support the roles assigned to topological memory and learned adaptation. The Compact–Strong closure shows better Blind-ID accuracy and no detected pooled Blind-OOD difference. A related compactness–accuracy pattern appears in the separate static DC-OPF transfer study, but PVR-DC is not the same dynamic Compact variant. Severe congestion also exposes cases where generic baselines are better on dispatch error or topology-specific RAW

feasibility. The post-hoc case57 result is similarly mixed: DC3-style has stronger native feasibility, whereas PVR-DC is more accurate after common repair. Different supervision and topology scope preclude an equal-budget interpretation. The architecture contributes an inductive bias and a safety mechanism, not an oracle for every graph optimization problem.

The synthetic benchmark establishes mechanisms but is not architecture-neutral: its late target is policy-independent, yet its teacher/task definition is strongly aligned with cycle-space memory. The external benchmark addresses topology and physics externality but is static, so it does not validate the Vorticity memory mechanism. A domain-natural external long-memory task is therefore future work rather than an extrapolated claim. Likewise, current numerical stress tests support feasibility at tested scales, not arbitrary-size solver guarantees. Parameter counts are independent of graph size for shared Reaction weights, but state, basis storage, and constrained-solver costs are not; sparse cycle bases and localized or iterative projections are concrete scaling directions. The same-A100 refactor improves the tested implementation overall, but the largest rank–horizon cell shows no consistent gain; neither parameter count nor this engineering comparison establishes architecture-wide speed dominance.

Equation eq. (4) is an elementary bounded-simplex allocation identity, not a new resource-allocation theorem. Our contribution is its integration with Pressure, Vorticity, Reaction, and the pressure-safe fallback.

8 Conclusion

PVR-GN separates conservative transport, divergence-free topological memory, learned adaptation, and feasibility into auditable computational roles. The blind synthetic benchmark combines competitive accuracy with zero observed PVR rollout failures and strong parameter efficiency; clustered closures identify the 3,025-parameter Compact variant as preferred and verify the roles of Vorticity, Reaction, and safe fallback. A same-A100 refactor reduces Compact latency without changing tested outputs, while its rank–horizon boundary prevents an intrinsic-speed claim. In a separate static PGLib/MATPOWER transfer study, PVR-DC has the lowest equal-weight macro held-out 14/30/57 dispatch MSE, remains numerically close to GNN on the unseen 118-bus case, and exhibits mixed economic/feasibility tradeoffs under severe congestion. A separately frozen post-hoc case57 comparison adds a natively feasible DC3-style reference and preserves the same accuracy–feasibility tradeoff. The results support structured learning for constrained graph dynamics while delimiting where the evidence does and does not establish superiority.

Reproducibility Statement

Headline synthetic and confirmatory external evaluations were frozen before their blind/test openings. Compact–Strong, No-V/No-R, safety-removal, and runtime closures did not rerun training, selection, blind inference, or external TEST inference. The labeled DC3-style comparison used a DEV-selected checkpoint and distinct one-shot post-hoc lock on already-open case57 rows, without post-TEST tuning. Appendix G records units, counts, denominator conventions, hashes, and freeze boundaries; the external campaign uses frozen PGLib-OPF v23.07 and MATPOWER 8.1 sources. Proofs, KKT details, audits, and additional tables are in the appendices.

AI Use Statement

Generative AI tools assisted with literature discovery, experimental-design feedback, implementation and analysis code, checking derivations and proofs, interpreting archived results, drafting and revising text, and formatting tables and figures. All AI-assisted material was reviewed against proofs, source code, primary bibliographic records, and archived experiment artifacts; no reported measurement was generated by an AI tool. The author takes responsibility for the mathematical claims, experimental design, software verification, interpretation, and final manuscript content.

REFERENCES

A. Sanchez-Gonzalez, J. Godwin, T. Pfaff, R. Ying, J. Leskovec, and P. Battaglia. Learning to simulate complex physics with graph networks. *ICML*, 2020.

T. Pfaff, M. Fortunato, A. Sanchez-Gonzalez, and P. W. Battaglia. Learning mesh-based simulation with graph networks. *ICLR*, 2021.

Y. Li, R. Yu, C. Shahabi, and Y. Liu. Diffusion convolutional recurrent neural network: Data-driven traffic forecasting. *ICLR*, 2018.

M. T. Schaub and S. Segarra. Flow smoothing and denoising: Graph signal processing in the edge-space. In *IEEE GlobalSIP*, 2018.

T. M. Roddenberry and S. Segarra. HodgeNet: Graph neural networks for edge data. In *53rd Asilomar Conference on Signals, Systems, and Computers*, pages 220–224, 2019. doi:10.1109/IEEECONF44664.2019.9049000.

M. Yang, V. Borovitskiy, and E. Isufi. Hodge-compositional edge Gaussian processes. In *AISTATS*, PMLR 238:3754–3762, 2024.

J. Hoppe and M. T. Schaub. Representing edge flows on graphs via sparse cell complexes. In *Proceedings of the Second Learning on Graphs Conference*, PMLR 231:1:1–1:22, 2024.

B. Amos and J. Z. Kolter. OptNet: Differentiable optimization as a layer in neural networks. *ICML*, 2017.

A. Agrawal, B. Amos, S. Barratt, S. Boyd, S. Diamond, and J. Z. Kolter. Differentiable convex optimization layers. *NeurIPS*, 2019.

J. Duchi, S. Shalev-Shwartz, Y. Singer, and T. Chandra. Efficient projections onto the ℓ_1-ball for learning in high dimensions. *ICML*, pages 272–279, 2008.

K. C. Kiwiel. Breakpoint searching algorithms for the continuous quadratic knapsack problem. *Mathematical Programming*, 112:473–491, 2008.

P. L. Donti, D. Rolnick, and J. Z. Kolter. DC3: A learning method for optimization with hard constraints. *ICLR*, 2021.

D. Deka and S. Misra. Learning for DC-OPF: Classifying active sets using neural nets. In *IEEE Milan PowerTech*, pages 1–6, 2019. doi:10.1109/PTC.2019.8810819.

X. Pan, T. Zhao, and M. Chen. DeepOPF: Deep neural networks for optimal power flow. In *ACM e-Energy*, 2021. doi:10.1145/3486611.3492390.

D. Owerko, F. Gama, and A. Ribeiro. Optimal power flow using graph neural networks. In *IEEE ICASSP*, pages 5930–5934, 2020. doi:10.1109/ICASSP40776.2020.9053140.

S. Babaeinejadsarookolaee et al. The Power Grid Library for benchmarking AC optimal power flow algorithms. arXiv:1908.02788, 2019. PGLib-OPF release v23.07 used in this work.

R. D. Zimmerman and C. E. Murillo-Sanchez. MATPOWER (Version 8.1) [Software], 2025. https://matpower.org.

R. D. Zimmerman, C. E. Murillo-Sanchez, and R. J. Thomas. MATPOWER: Steady-state operations, planning, and analysis tools for power systems research and education. *IEEE Transactions on Power Systems*, 26(1):12–19, 2011. doi:10.1109/TPWRS.2010.2051168.

A Full Theory and Proof Details

A.1 Divergence decoupling and memory

Because $BZ = 0$, $B(p + Z\omega) = Bp$. If $\omega_{t+1} = \omega_t + w_t - \nu_t D_t \omega_t$, then $w_t = \nu_t = 0$ gives exact hold. For $w_t = 0$ and $D_t \neq 0$, diagonalize $D_t = Q\Lambda Q^\top$. The update operator $I - \nu_t D_t$ has spectral radius at most one whenever $0 \leq \nu_t \leq 2/\lambda_{\max}(D_t)$, proving non-increase of the coordinate energy $\|\omega_t\|_2^2$. If $D_t = 0$, the update is the identity and the step-size condition is vacuous. A physical-energy statement for $\|Z\omega_t\|_2^2$ additionally requires an orthonormal cycle basis (or the corresponding basis-aware metric).

A.2 Shared-budget elimination proof

Let $d_r = [r - u_0]_+$ and assume $r \geq 0$. If $r \leq u_0 + \beta a$ for some $a \in \Delta_m$, then $d_r \leq \beta a$ and summing gives $\mathbf{1}^\top d_r \leq \beta$. Conversely, when $\chi_r = \mathbf{1}^\top d_r/\beta \leq 1$, choose $a = d_r/\beta + (1 - \chi_r)b$ for arbitrary $b \in \Delta_m$. Then $a \geq 0$, $\mathbf{1}^\top a = 1$, and $\beta a \geq d_r$, hence $r \leq u_0 + \beta a$.

For a pressure flow p, $q = [p - u_0]_+$ and $\chi_p = \mathbf{1}^\top q/\beta$. The pressure-safe allocation

$$a_{\mathrm{PS}} = q/\beta + (1 - \chi_p)b \tag{9}$$

satisfies $u_0 + \beta a_{\mathrm{PS}} \geq p$ whenever $\chi_p \leq 1$. This proves safety of the fallback independently of learned weights.

A.3 Valve elimination

The allocation variable exists iff $f \geq 0$ and $\mathbf{1}^\top [f - u_0]_+ \leq \beta$. Necessity follows as above. Sufficiency follows by constructing an allocation from the deficits of f. This equivalence yields the flow-only feasible set in eq. (5) and supports the joint instantaneous projection

$$f^\star = \operatorname*{arg\,min}_{f \in \mathcal{F}(s)} \tfrac{1}{2} \|f - r\|_{K^{-1}}^2 . \tag{10}$$

The oracle is instantaneous; it is not claimed to solve a temporal global optimum.

A.4 Fixed-active-set KKT differentiation

Write the cycle-coordinate problem locally as $\frac{1}{2} z^\top H(\xi) z + g(\xi)^\top z$ subject to active affine constraints $A(\xi)z = b(\xi)$. The KKT system is $M(\xi)y(\xi) = c(\xi)$ with

$$M = \begin{bmatrix} H & A^\top \\ A & 0 \end{bmatrix}, \quad y = \begin{bmatrix} z \\ \lambda \end{bmatrix}, \quad c = \begin{bmatrix} -g \\ b \end{bmatrix}. \tag{11}$$

On a nonsingular fixed-active-set region,

$$\frac{dy}{d\xi} = M^{-1} \left(\frac{dc}{d\xi} - \frac{dM}{d\xi} y \right). \tag{12}$$

This derivative need not extend smoothly across active-set switches. It describes the continuous fixed-active-set projection branch only; the discrete proposal selector and accept/fallback wrapper are not differentiated by this argument.

B Progressive Validation and Numerical Audits

The internal development ladder is retained as a progressive validation record rather than as methodological version labels. The Joint Instantaneous Oracle gave an 8.24% gain over pressure-safe (95% CI [5.71,10.88]); the Safe-Anytime controller 7.68% [4.47,10.82]; Learned Reaction 7.56% [5.42,9.68]; and the differentiable active-set implementation 7.35% [5.67,9.08]. The active-set gradient audit produced median relative error 2.55×10^{-9} and maximum 3.52×10^{-8} against finite differences on stable active sets. A native stress campaign covered 108 rollouts on 12 topologies with cycle ranks 8–64 and horizons 18/64, with zero observed feasibility failures and residuals at machine precision.

C Additional Frozen Synthetic Results

Table 5: Certified/wrapped point estimates and runtimes from the frozen synthetic benchmark. Runtime is implementation-specific and is not used as a complexity claim.

Policy	ID NMSE	OOD NMSE	OOD ms
PVR-Strong ensemble	0.18535	**0.28410**	1335.9
PVR-Compact	**0.18355**	0.28430	4700.0
GNN wrapped ensemble	0.18639	0.28478	1637.9
GRU wrapped ensemble	0.18614	0.28470	2143.2
Pressure-safe	0.20041	0.29436	**258.3**

These historical point estimates predate the semantics-preserving runtime refactor and remain unchanged. Dense storage of a cycle basis can scale as $O(mr_c)$ and a dense KKT solve can scale cubically in the cycle-coordinate dimension in the worst case. Therefore parameter efficiency does not establish computational efficiency.

Table 6: Read-only mechanistic closures from the frozen control pilot. Positive $\Delta = \mathrm{NMSE}_{\mathrm{ablation}} - \mathrm{NMSE}_{\mathrm{Full}}$ favors Full PVR.

Control	Scope	Pairs	Clusters	Mean Δ [95% CI]
No-V	all	648	54	0.819994 [0.798773, 0.838918]
No-V	overloaded	388	54	0.707060 [0.685708, 0.726184]
No-R	all	648	54	0.013947 [0.011120, 0.016858]
No-R	overloaded	388	54	0.023978 [0.018757, 0.029356]

The explicit No-R run retains Pressure and its analytic safe return while removing learned Reaction. Full PVR wins 54/54 overloaded No-V clusters and 50/54 overloaded No-R clusters. In the separate safety-removal run, 189/216 overloaded rollouts fail, versus 0/432 nominal/stressed failures; among the 27 completed overloaded rollouts, NMSE is 0.938179. Candidate steps worsen the objective in 4,285/9,036 cases. The archive supports a loss-of-safe-return conclusion, not a capacity-violation interpretation.

Table 7: Same-A100 PVR-Compact implementation audit on 36 fresh non-blind episodes in 12 rank–horizon clusters. C1 caches a repeated factorization; the warm-start-only C2 candidate was rejected.

Endpoint	Result
Reference / C1 median latency	1946.028 / 1478.404 ms
Median paired reduction	18.939% [15.505, 21.032]%
Median paired speedup	1.234× [1.184, 1.266]×
Episodes with C1 faster	94.4%
Classification / step-count mismatches	0 / 0
Maximum final-flow ℓ_∞ difference	0
Maximum audited divergence residual	1.665×10^{-16}

The C3 end-to-end audit establishes semantic equivalence at the recorded tolerances. The gain is not uniform: in the rank-64/horizon-128 cell, only 1/3 episodes is faster and descriptive medians change from 5973.715 to 6005.549 ms. We therefore report an accepted implementation improvement, not a universal speed claim.

D Post-Blind Synthetic Validity Audit

The frozen audit traced the late target to exogenous graph/node quantities generated before policy rollout and verified that the policy cannot alter the target. Late query signals are projected into

the cycle space and no learned baseline receives the target as input. This supports low metric-contamination risk. However, the teacher and task definitions use cycle projection and pressure-safe/safe-anytime structure, yielding strong teacher/task alignment with PVR. We therefore classify the synthetic benchmark as controlled mechanistic evidence and rely on the separately frozen DC-OPF campaign for external topology/physics evidence.

E External Protocol and Denominator Audit

Phase A froze PGLib/MATPOWER cases, perturbation manifests, source versions, and oracles before learned evaluation. Phase B trained three seeds for each architecture on 14/30/57 TRAIN and selected checkpoints only on 14/30/57 DEV. The frozen best seeds are PVR-DC 7103, GNN 7201, and GRU 7302; ensembles average the three frozen seed proposals before the common dispatch projection. Phase C verified all hashes, created a persistent one-shot TEST lock, and then opened typical, scale-OOD, and congestion-OOD TEST once.

The external parameter counts are 3,361 (PVR-DC), 46,913 (GNN), and 41,729 (GRU). PVR-DC physical features are computed only from causal case/scenario information: an economic anchor, DC angles/flows, and local loading derived without oracle $P_g/P_f/V_a$ inputs. This is an explicit inductive bias, not target leakage.

Table 8: Oracle availability and RAW-feasible counts for frozen best-single models. Accuracy uses all oracle-successful scenarios regardless of RAW feasibility.

Case	Oracle ok	PVR feas.	GNN feas.	GRU feas.	Attempted
Typical 14	150	150	150	150	150
Typical 30	149	61	67	98	150
Typical 57	150	121	137	119	150
Scale 118	150	0	0	0	150
API 14	142	141	135	77	150
API 30	111	104	0	108	150
API 57	150	0	150	0	150
API 118	83	0	0	0	150

A dtype-safe read-only recomputation from the frozen scenario CSV verifies for every model and case that `accuracy_n` = `oracle_successful`. Model RAW infeasibility is therefore an endpoint, not a survival filter. The common constraint repair succeeds on all oracle-successful model/scenario rows in this campaign. We use "oracle-unsuccessful" rather than "oracle-infeasible" because the stored MATPOWER success flag alone does not prove mathematical infeasibility.

F Post-Hoc DC3-Style Case57 Audit

The *DC3-style DC-OPF adaptation (post-hoc)* follows the equality-completion and inequality-correction structure of DC3 (Donti et al., 2021) on the frozen PGLib case57 DC map. It is not an execution of the official DC3 AC-OPF benchmark. Three 53,806-parameter, float64 models were trained for 1,000 epochs on the 300 case57 TRAIN scenarios with self-supervised normalized generation cost and inequality penalties; selection used only 75 case57 DEV scenarios. Seed 7401 at epoch 980 was selected by the precommitted lexicographic DEV rule: maximize corrected feasibility, then minimize normalized generation cost, then retain the earliest epoch. Oracle dispatch was excluded from both loss and selection. PVR-DC used the identical case57 rows but also labeled 14/30-bus TRAIN/DEV rows, so sample budget, supervision, objective, and topology scope differ.

DC3-style achieves strict native feasibility on 150/150 Typical57 and 150/150 API57 scenarios, with dispatch NMSE 0.061501 and 0.341150, flow MSE 0.040926 and 0.304353, and generation-cost regret 5.173% and 12.427%, respectively. Its correction records one batch step but zero displacement on every scenario because all raw outputs are already feasible; the common repair changes it only at numerical precision. PVR-DC RAW feasibility is 121/150 on Typical57 and 0/150 on API57; the common repair succeeds on all 300 rows. PVR-DC repaired cost regret is 0.112% and 8.655%. Its negative RAW API57 regret is not interpreted because every RAW prediction in that stratum is

Table 9: Paired case57 post-hoc comparison. Δ = DC3-style MSE − PVR-DC MSE; positive values favor PVR-DC. CIs are 10,000-replicate paired scenario bootstraps with seed 94001. "DC3 lower" is descriptive, not a predeclared test.

Stratum	Output convention	DC3-style	PVR-DC	Δ [95% CI]	DC3 lower
Typical57	Native/RAW	0.945444	0.001627	0.943818 [0.750879, 1.150396]	0/150
Typical57	Common repair	0.945444	0.000785	0.944659 [0.751890, 1.152039]	0/150
API57	Native/RAW	4.999352	4.686966	0.312386 [0.007311, 0.619390]	66/150
API57	Common repair	4.999352	1.814662	3.184690 [2.991490, 3.374497]	0/150

infeasible. The common-repair comparison therefore isolates feasible dispatch accuracy, while native feasibility remains a separate endpoint. DC3-style timing is batched and PVR-DC timing is recorded per scenario, so no runtime ranking is reported.

The method-faithful training archive was frozen before evaluation; preflight reproduced all non-runtime DEV fields to within 7.28×10^{-12} and verified that the evaluator had not parsed TEST. The two TEST hashes and static DC-map adapter checks then matched exactly; all 300 paired keys are unique, and an independent recomputation reproduces the four bootstrap rows to machine precision. Full digests and the compact summary, decision, preflight, and manifest records accompany the supplement.

G Reproducibility and Freeze Boundaries

The synthetic evidence package records the final blind evaluation clusters, bootstrap rule, episode CSVs, failure counts, checkpoint hashes, and safety residuals. The read-only ablation-closure archive (SHA-256 prefix/suffix `22a2ebea...d5a89`) supplies the Compact–Strong and No-V/No-R clustered statistics. The separate historical control archive (`837263b3...decc0`) is explicitly pilot-only and supplies the safety-removal endpoints. Complete digests are recorded in the supplementary `data/README.md`. The runtime record is a read-only cluster bootstrap from frozen same-A100 per-episode timing rows on fresh non-blind audit inputs. None of these closures loaded checkpoints, reran inference, or selected a policy from test performance.

The external evidence is split into independently frozen Phase A (oracle/data), Phase B (TRAIN/DEV selection), and Phase C (one-shot TEST and denominator audit). The separate case57 post-hoc archive and final-manifest digests are `c0007381...342ad` and `bb995abd...c40f1`; the method-faithful training archive and selected-checkpoint digests are `8cc690ea...c2951` and `57ca4ee1...7e1d`. Full hashes are in supplementary `data/README.md`. Confirmatory Table 3 was not altered or rerun; Table 4 is explicitly post-hoc. The control-pilot, runtime, and DC3-style numbers are not numerically merged with the final Blind-ID/Blind-OOD campaign or confirmatory external table. Internal development identifiers are intentionally omitted from the scientific narrative; the reproducible stages are Joint Instantaneous Oracle, Safe-Anytime Temporal Controller, Learned Reaction, Differentiable Active-Set Projection, and Native Safety-Repair Stress Test.