LEAST-ACTION CAUSAL DISCOVERY VIA SDES FOR STOCHASTIC DYNAMICAL SYSTEMS

Anonymous authors
Paper under double-blind review

ABSTRACT

We introduce the Principle of Least-Action Causal Identification of Dynamics (PLACID), a controlled-SDE testbed for learning latent dynamics and examining sparse causal drift structure from irregularly sampled time series. Under explicit regularity assumptions, Girsanov's theorem relates path-space relative entropy to a quadratic steering cost, motivating a decomposition into reusable structural drift and data-dependent control. Theoretically, we give a conditional drift-consistency analysis and sufficient coverage, smoothness, and signal-margin conditions for oracle recovery, a counterexample shows that vanishing action and uniformly accurate drift can still retain a false edge. Empirically, within this controlled-SDE setting, Lorenz96 and Glycolysis experiments show drift-ranking signal, including for 50-variable Lorenz models, although fixed-threshold graph recovery remains poor in that high-dimensional setting. Checkpoint diagnostics show that similar prediction error can accompany different drift rankings. A matched Lorenz D10 comparison finds no resolved ranking benefit from control or stochastic diffusion over deterministic drift, a released-code NGM-GL comparator gives similar ranking with substantial checkpoint sensitivity. Ablations show that bounded control prevents constant-drift collapse in a zero-action-penalty stress test, while proximal sparsity improves Lorenz graph parsimony. Together, these results position PLACID as a testbed for examining how control regularization and sparse drift parameterization shape structure learning in stochastic dynamical systems.

1 INTRODUCTION

Accurate trajectory fits need not yield accurate causal graphs. For irregularly observed dynamical systems, a learned model can explain the observations without recovering which coordinates influence each component of the instantaneous drift. This distinction matters when continuous-time models are used for structural discovery rather than prediction alone. Classical time-series approaches target conditional independence or predictive dependence (Runge et al., 2019), here we examine the evidence needed to interpret a learned drift as a graph of local dependencies.

Continuous-time models accommodate physical observation times without requiring a fixed sampling interval. Neural ODEs provide one such parameterization (Chen et al., 2018), Neural Graphical Models (NGM) use sparse drift networks to estimate local dependencies, including in stochastic systems (Bellot et al., 2022). An SDE additionally represents process variability, but this representational advantage does not by itself imply better drift-graph recovery. Whether learned diffusion or trajectory-specific control improves structural ranking is an empirical question requiring matched deterministic and stochastic estimators.

We study this question in a controlled-SDE testbed. SCOTCH (Wang et al., 2024) combines latent SDEs with variational inference over graphs and paths. Our PLACID estimator instead uses continuous sparse drift parameters and source-conditioned interval fits, providing a setting in which to separate structural sparsity, stochastic integration, and inference control. The empirical findings characterize this setting, the mathematical obstruction and its stated oracle conditions concern action-based structural recovery more generally.

The drift–control decomposition also exposes a practical failure mode. A sufficiently expressive controller can explain observations while leaving the structural drift uninformative. We therefore

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.

study control regularization and bounded control alongside proximal sparsity, separating their effects on graph ranking, pruning and controller takeover. Our analysis shows that even vanishing action and uniformly accurate drift can preserve a false edge. This motivates evaluating structural pruning separately from trajectory fitting, and specifying the additional coverage and smoothness conditions that bridge the two in an oracle comparison.

We use the **P**rinciple of **L**east-**A**ction **C**ausal **I**dentification of **D**ynamics (PLACID). Our contributions are:

- A controlled-SDE testbed that separates structural drift from data-dependent steering and combines control-energy regularization with sparse drift parameterizations for irregular observations.
- An analysis of why action accuracy alone does not certify structural recovery: a persistent-false-edge counterexample, a quantitative oracle derivative bound, and a direct link between zero input groups and absent drift edges.
- Within this controlled-SDE setting, a matched three-arm comparison on five Lorenz D10 realizations at nominal $p = .3$ finds no resolved graph-ranking benefit from control or stochasticity. Additional Lorenz96 and Glycolysis panels distinguish pruning from ranking and document a controller-induced collapse under zero action weight. These diagnostics delimit the empirical evidence rather than establish the necessity of the components.

2 Related Work

Instantaneous drift dependence is related to causal intervention semantics for SDEs (Sokol & Hansen, 2014) and to local-independence graphs for partially observed stochastic systems (Mogensen et al., 2018). Conditional-independence tests for SDE coordinate processes provide a distinct structure-learning approach (Manten et al., 2025). Learning identifiable causal structures from continuous-time observations also extends time-series graph learning beyond a fixed sampling grid (Runge et al., 2019). Neural ODEs (Chen et al., 2018) and Neural Graphical Models (NGM) (Bellot et al., 2022) parameterize continuous-time drift dependencies using ODEs. NGM also targets graph recovery from stochastic observations, so an explicit diffusion model cannot be presumed necessary for structural ranking. Neural SDEs provide stochastic dynamics (Li et al., 2020; Kidger et al., 2021). SCOTCH (Wang et al., 2024) performs variational inference over graph structures and latent SDE paths. PLACID instead extracts a graph from continuous drift input-group scores and uses source-conditioned interval inference. The distinction is in graph parameterization, inference, and capacity regularization, not a new Girsanov identity. Section 5 reports local runtime evidence and a separate comparison with published accuracy values.

Inference in latent SDEs is challenged by intractable marginal likelihoods. Recent variational methods leverage Girsanov's theorem to express the path-space Kullback-Leibler (KL) divergence as a quadratic control cost (Li et al., 2020; Tzen & Raginsky, 2019). While this connection has been exploited for generative modeling (Huang et al., 2021; Donati et al., 2022) and Schrödinger bridges (Liu et al., 2024; Tong et al., 2024), standard bridges seek the minimal control effort to transport mass between fixed boundary marginals. PLACID uses this path-space action functional as a structural inductive bias: the fixed-law oracle analysis motivates penalizing additional steering, but joint training need not recover the true drift.

A central theoretical hurdle in latent variable modeling is orthogonal non-identifiability (Hyvarinen et al., 2023). While nonlinear Independent Component Analysis (ICA) (Hyvarinen & Morioka, 2016; 2017) and identifiable VAEs (Khemakhem et al., 2020) use auxiliary variables to achieve linear equivalence, resolving the remaining linear ambiguity requires further structure. Mechanism sparsity regularization (Lachapelle et al., 2022) establishes recovery under specified variability and graphical conditions. A transfer from discrete-time conditional mechanisms to instantaneous SDE drift support requires a separate argument. PLACID uses sparsity as an inductive bias, the present analysis does not establish those conditions for its benchmarks or prove identifiability of its learned coordinates.

3 DIAGNOSTIC FRAMEWORK

Figure 1 summarizes the estimator. This section specifies the implemented model and objective, Section 4 analyzes the separate oracle question of structural recovery from action accuracy.

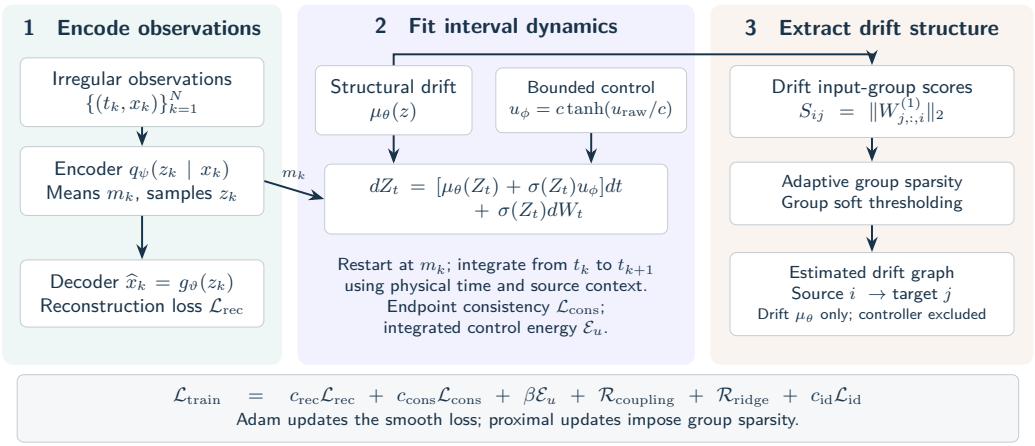

Figure 1: PLACID uses a drift network and an inference controller to fit irregular observations. Control energy regularizes the additional drift shift, input-group penalties encourage sparse drift dependencies. Integration uses physical observation times and source-conditioned interval inference.

3.1 DIAGNOSTIC ESTIMATOR: PLACID

PLACID separates reusable dynamics from trajectory-specific steering through a learned structural drift and a data-dependent controller. Reconstruction and interval consistency fit the observed trajectories, control energy regularizes the steering effort, and proximal sparsity encourages parsimonious drift dependencies.

3.1.1 CONTROL REGULARIZATION AND CAPACITY

Control energy penalizes trajectory-specific steering, encouraging the structural drift to account for the observations. The oracle comparison formalizes this motivation under a fixed posterior joint law, the role of control under joint finite-data training is tested empirically. To limit controller takeover, we use the smooth coordinatewise cap $u = c\tanh(u_{\text{raw}}/c)$ with $c = 2$ in the bounded experiments. The same effective control enters the drift shift and the action cost. In the zero-action-penalty stress test, the cap mitigates constant-drift collapse introduced by an unbounded controller. This diagnoses a failure of the controlled formulation and does not justify including a controller. The cap bounds u, not the product σu independently of learned diffusion, and does not fix the coordinate or diffusion gauges.

3.1.2 DRIFT AND CONTROL DECOMPOSITION

The controlled drift is $\mu_\theta + \sigma u_\phi$. The first term defines the candidate structural mechanisms and is used to extract the drift graph, the second supplies data-dependent corrections during inference. The policy uses physical time and source-observation context. This separation makes drift-graph recovery a distinct evaluation target from reconstruction or controlled trajectory fit. Each observed interval starts from its source encoder mean and is integrated to the next physical observation time. The controller receives time and source context, and the resulting endpoint is compared with the next encoder output. These independent interval restarts make irregular gaps tractable without interpolation, but they do not define one coherent posterior law for a complete trajectory. Thus the empirical endpoint-consistency term and the oracle path-law comparison answer different questions. The graph is read only from the reusable drift branch, including controller weights in an edge score would change the structural target.

3.1.3 PRACTICAL TRAINING OBJECTIVE

The implemented joint-training loss is

$$\mathcal{L}_{\text{train}} = c_{\text{rec}}\mathcal{L}_{\text{rec}} + c_{\text{cons}}\mathcal{L}_{\text{cons}} + \beta\mathcal{E}_u + \mathcal{R}_{\text{coupling}} + \mathcal{R}_{\text{ridge}} + c_{\text{id}}\mathcal{L}_{\text{id}}. \tag{1}$$

Here reconstruction fits decoded observations, interval consistency fits integrated endpoints and regularizes encoder variance, and control energy sums squared effective control across physical integration steps. The identity term is active for Glycolysis. An adaptive group proximal step follows each Adam update on the encoder, decoder, and drift input weights. Appendix B.3 defines every reduction and coefficient. This is a control-regularized interval-fitting objective, not an exact optimization of the fixed-law oracle path KL analyzed in Section 4.

3.1.4 SPARSE DRIFT PARAMETERIZATION

Proximal updates encourage structural zeros in source-target drift blocks, while a coupling penalty regularizes the encoder and decoder. These operations provide a tractable sparse parameterization, the additional assumptions needed for coordinate identification are stated in Section 4.

Adaptive Group Lasso and Proximal Optimization. To score candidate drift dependencies, we use the locally connected drift architecture. The first drift layer is organized into target-specific source blocks and can be viewed as a tensor $W^{(1)} \in \mathbb{R}^{D\times H\times D}$, where $W^{(1)}_{j,:,i}$ mediates the effect of source latent coordinate z_i on target drift component $\mu_{\theta,j}$. We define the continuous directed edge score $S_{ij} = \|W^{(1)}_{j,:,i}\|_2$ where S_{ij} corresponds to the candidate edge $z_i \to z_j$. We apply adaptive group lasso to these source-target blocks and optimize the resulting non-differentiable sparsity penalty using a block soft-thresholding proximal operator. The exact adaptive weights and proximal updates are given in Appendix B.3.

Theory versus implemented score. The oracle analysis below compares functional drifts and derivative support. The implemented group norm is a tractable structural surrogate: a zero group excludes a drift edge when the target branch has no bypass path, but a nonzero group need not imply functional dependence because downstream weights can cancel it. The oracle recovery bound does not certify the group-norm ranking.

Encoder–decoder coupling. A smooth penalty on products of encoder and decoder magnitude matrices regularizes the representation. It sums both multiplication orders across hidden channels, Appendix B.3 gives the exact expression and the remaining loss reductions. This is a representation bias, not a coordinate-identification guarantee.

4 WHEN DOES LEAST ACTION IMPLY STRUCTURAL RECOVERY?

4.1 LATENT DYNAMICS AND THE DRIFT GRAPH

On $\mathcal{C} = C([0,T],\mathbb{R}^D)$ with its canonical filtration, consider

$$dZ_t = \mu_\theta(Z_t)\,dt + \sigma(Z_t)\,dW_t, \qquad Z_0 \sim p_0. \tag{2}$$

For differentiable drift on an open domain, define $G_{ij} = 1$ when $\partial_i\mu_j$ is not identically zero there. Thus rows denote sources and columns targets. This is a graph of instantaneous drift dependence. Intervention semantics additionally require autonomous structural mechanisms, a specified intervention operation, and assumptions about noise and latent confounding. Independent Brownian components alone do not exclude cross-state dependence in the diffusion coefficients. Finite-gap transition dependence need not equal instantaneous drift dependence.

4.2 CONTROL ENERGY AND PATH RELATIVE ENTROPY

For a fixed observation x, let Q_x have initial law q_0^x and dynamics

$$dZ_t = [\mu_\theta(Z_t) + \sigma(Z_t)u_x(t, Z_{[0,t]})]dt + \sigma(Z_t)dW_t^{Q_x}. \tag{3}$$

Assume weak uniqueness, a common invertible diffusion, $q_0^x \ll p_0$, progressively measurable control, and a valid Girsanov change of measure whose exponential is a true martingale. Assume finite initial relative entropy and finite expected control energy. These are conditions on path laws, not consequences of fitting discrete observations. Under these conditions,

$$\mathrm{KL}(Q_x \| P_\theta) = \mathrm{KL}(q_0^x \| p_0) + \frac{1}{2}\mathbb{E}_{Q_x}\int_0^T \|u_x(t, Z_{[0,t]})\|^2 dt. \tag{4}$$

See Léonard (2012) for path-measure change of drift under finite entropy. The control energy is a relative-entropy cost. Interpreting it as physical heat or entropy production requires additional physical assumptions and units. It is not generally the entire path KL because of the initial term. For a componentwise cap $|u_k| \le c$, the pathwise energy is at most $TDc^2/2$. A sharper compensation statement holds pointwise without a uniform diffusion bound. For invertible $\sigma(z)$ and desired drift correction r, set $v = \sigma(z)^{-1}r$. Then

$$\min_{\|u\|_\infty \le c} \|\sigma(z)^{-1}(r - \sigma(z)u)\|_2^2 = \sum_{k=1}^{D}(|v_k| - c)_+^2, \tag{5}$$

where $(a)_+ = \max(a, 0)$. Thus exact compensation is possible within the box precisely when $\|\sigma(z)^{-1}r\|_\infty \le c$. This describes the capacity of all box-bounded controls, a neural policy may realize less. Learned diffusion can enlarge that capacity in drift units, so the cap alone does not force drift correctness. Appendix A.5 proves this identity and gives its diagonal-diffusion form.

4.3 Oracle drift comparison and its limits

Assumption 4.1 (Common diffusion and initial law)**.** The true law $P^\star$ and candidate laws P_θ satisfy the preceding change-of-measure conditions, share p_0 and a fixed known invertible σ, and the candidate class contains $\mu^\star$.

Assumption 4.2 (Fixed exact posterior joint law)**.** Fix a data distribution $\nu(dx)$ and conditional path laws Q_x independently of θ. Their aggregate $\bar{Q} = \int Q_x \nu(dx)$ equals $P^\star$. All relative entropies and the mutual information used below are finite.

Theorem 4.3 (Oracle drift consistency)**.** *Under Assumptions 4.1 and 4.2,*

$$\mathbb{E}_\nu \mathrm{KL}(Q_X \| P_\theta) = I_Q(X, Z_{[0,T]}) + \mathcal{A}(\theta), \tag{6}$$

$$\mathcal{A}(\theta) = \frac{1}{2}\mathbb{E}_{P^\star}\int_0^T \|\sigma(Z_t)^{-1}(\mu^\star(Z_t) - \mu_\theta(Z_t))\|^2 dt. \tag{7}$$

Every global minimizer has $\mu_\theta = \mu^\star$ almost everywhere for $\rho(B) = \mathbb{E}_{P^\star}\int_0^T \mathbf{1}_B(Z_t)dt$.

The proof is in Appendix A.1. The conditional control energy contains a constant information contribution, it need not vanish at the true drift. Jointly changing the encoder and prior does not keep that contribution fixed. Weak posterior convergence alone does not imply KL or action convergence. This theorem concerns a fixed-coordinate oracle comparison, not consistency of the implemented optimizer or recovery of a global derivative-support graph from finite data. Exact noiseless conditioning can violate the finite-entropy assumptions.

Corollary 4.4 (Oracle graph recovery on an occupied domain)**.** *Let U be open with $\rho(V) > 0$ for every nonempty open $V \subset U$. If $\mu^\star$ and μ_θ are C^1 on U, every global minimizer in Theorem 4.3 has $\mu_\theta = \mu^\star$ and $J_{\mu_\theta} = J_{\mu^\star}$ on U. Its drift graph restricted to U therefore equals the true graph there.*

The proof is in Appendix A.2. Full occupation support is an additional assumption, not established for our benchmarks. The result makes no claim outside U, from finite data, or across unknown coordinates.

4.4 Small action does not certify the graph

The exact-minimizer result leaves a practical question: does nearly minimal action imply nearly correct structure? The following obstruction shows why an action objective and a structural regularizer address different properties.

Proposition 4.5 (Vanishing action with a persistent false edge)**.** *Let $P^\star$ be the stationary two-dimensional SDE with $\mu^\star(z) = (-z_1, -z_2)$ and $\sigma = I$. With the same initial law, set*

$$\mu_n(z) = (-z_1 + n^{-1}\sin(nz_2), -z_2).$$

These drifts have a common global Lipschitz bound and satisfy $\|\mu_n - \mu^\star\|_\infty \le n^{-1}$ and $\mathcal{A}(\mu_n) \le T/(2n^2) \to 0$. Nevertheless, $\partial_2(\mu_n)_1(0) = 1$ for every n, so every candidate contains the false edge $2 \to 1$. For any fixed cap $c > 0$, the correction $u_n = \mu^\star - \mu_n$ also satisfies $\|u_n\|_\infty < c$ for all sufficiently large n and exactly restores the true total drift.

Appendix A.4 proves the claim. Thus even uniformly accurate, uniformly Lipschitz drift estimates and bounded, vanishing-energy corrections do not ensure graph recovery. The missing condition is control of how rapidly drift derivatives vary, not merely control of the drift's magnitude or Lipschitz constant. The obstruction also survives an integrated squared-sensitivity score: under the stationary true law, $\mathbb{E}_{P^\star}\int_0^T |\partial_2(\mu_n)_1(Z_t)|^2 dt = T(1 + e^{-n^2})/2$, which does not approach zero.

A sufficient bridge: coverage, smoothness, and signal. Theorem A.1 gives the complementary positive result. If occupation measure dominates volume on an open domain, diffusion is bounded above, and the drift-error Jacobian has a fixed Lipschitz bound, then on a compact interior set K,

$$\mathcal{A}(\theta) \le \varepsilon \quad \Longrightarrow \quad \max_{i,j} \sup_{z \in K} |\partial_i(\mu_\theta - \mu^\star)_j(z)| \le C\varepsilon^{1/(D+4)}$$

for sufficiently small positive ε, with C depending on the coverage and smoothness constants. Thresholding derivative scores at this bound recovers edges whose true strength exceeds twice the bound. The sine example violates the required uniform Jacobian smoothness. Together, these results distinguish action accuracy from structural accuracy, they are oracle statements, not finite-sample guarantees for joint training or certificates for the input-group scores below. The exponent $1/(D+4)$ and unmeasured constants can make this bound numerically uninformative at $D = 50$.

4.5 Sparse parameterization and coordinate scope

Unknown coordinate transformations present a separate ambiguity. Even exact drift recovery in one representation does not identify a native coordinate system. Appendix A.6 states a conditional support-separation criterion and an isotropic diffusion counterexample. Our practical sparsity mechanism instead has a direct, one-sided link to drift dependence:

Proposition 4.6 (Input-group control of drift sensitivity)**.** *Write a target branch as $\mu_j(z) = F_j(W_j z + b_j)$, with no bypass path, and let w_{ji} be column i of W_j. If F_j is differentiable with $\|\nabla F_j\|_2 \le C_j$, then $|\partial_i \mu_j(z)| \le C_j \|w_{ji}\|_2$. In particular, a zero input block excludes that drift edge.*

For the implemented ELU branches, C_j can be the product of downstream spectral norms, Appendix A.7 proves the bound. The converse fails because downstream weights and cancellations can eliminate functional dependence. Group norms are structural surrogates, not exact Jacobian support counts. We use them as a tractable structural bias, coordinate identification and known diffusion are not imposed by the implementation.

4.6 What the oracle bridge says about input groups

The derivative bound also yields a limited statement about the implemented parameterization. Let $S_{ij}^K(\mu) = \sup_{z \in K} |\partial_i \mu_j(z)|$ and suppose the assumptions of Theorem A.1 give a uniform derivative error bound E_h. Combining that bound with Proposition 4.6 gives

$$C_j \|w_{ji}\|_2 \ge \left(S_{ij}^K(\mu^\star) - E_h\right)_+. \tag{8}$$

Thus a sufficiently strong true derivative edge requires a nonzero input group in a candidate satisfying the oracle bound. The reverse implication fails: downstream cancellation permits nonzero groups with no functional edge, and learned C_j can differ across targets and fits. The result therefore supplies a necessary condition, not calibrated ranking or false-edge exclusion for our graph scores. Appendix A.7 proves the bound. The training loss in Eq. (1) uses interval endpoints and a weighted control energy, not the fixed-law path KL of Theorem 4.3, its optimization gives no reason to assume the oracle premise holds. Appendix B.7 states the narrower Euler-step KL identity.

Table 1: Inclusive AUROC, mean ± sample SD over five realizations. SCOTCH D10 and Glycolysis values are from Table 1 of Wang et al. (2024); D50 entries are local adapter runs stopped before producing graph estimates. NGM-GL denotes our released-code adaptation, with the matched same-split protocol documented for Lorenz D10, $p = 0.3$ in Appendix B.2. PLACID values are from this work. Protocols differ across methods; NR denotes no value reported.

Setting	SCOTCH	NGM-GL	PLACID (Ours)
Lorenz D10, $p = 0.3$	0.7279 ± 0.017	0.7358 ± 0.039	0.7456 ± 0.016
Lorenz D10, $p = 0.6$	0.6453 ± 0.014	0.6764 ± 0.025	0.7103 ± 0.051
Lorenz D50, $p = 0.3$	NR	0.7312 ± 0.042	0.7481 ± 0.0135
Lorenz D50, $p = 0.6$	NR	0.6705 ± 0.028	0.6935 ± 0.0309
Glycolysis	$0.7113 \pm .012$	0.6872 ± 0.019	$0.6528 \pm .0193$

5 EXPERIMENTS

We evaluate observed-coordinate drift graphs, the roles of controller capacity and proximal sparsity, and fitting cost on Lorenz96 and Glycolysis.

5.1 PRIMARY RESULTS

Five trajectory-disjoint realizations per setting use training-only preprocessing, validation-calibrated graph alignment, and fixed budgets. We score continuous drift-group AUROC including self-edges, matching the published SCOTCH convention. The comparison and secondary metrics are detailed in Appendix C. For Lorenz D10 at $p = .3$, off-diagonal AUROC is $.6608 \pm .0210$, below the inclusive $.7456 \pm .0157$. Across D10 and D50, AUROC ranges from .6935 to .7481, D50 retains ranking signal but has poor fixed-threshold graph recovery. Glycolysis AUROC is $.6528 \pm .0193$. Its (β, λ_4) setting was selected using graph labels on three development realizations and frozen for five new realizations from the same graph family, the no-control arm did not receive that search, limiting this comparison.

In the matched Lorenz D10, $p = 0.3$ NGM-GL comparison, validation-selected AUROC is $.7402 \pm .0436$ and final-checkpoint AUROC is $.6842 \pm .0216$, showing checkpoint sensitivity around the fixed-checkpoint value in Table 1. At the displayed means, PLACID has higher AUROC than NGM-GL in each Lorenz row, by 0.010 to 0.034. For Glycolysis, NGM-GL is higher by 0.034 (0.6872 versus 0.6528). These are descriptive cross-setting comparisons, not a claim of general superiority. We retain the full source and protocol distinction in Appendix B.2.

5.2 ABLATIONS AND FIT–STRUCTURE DIAGNOSTICS

In paired Lorenz runs, proximal updates produce 38–43 exact-zero drift groups and reduce mean SHD from 43.0 to 36.2, their ranking effect is uncertain, and no-control has the highest mean AUROC. On Glycolysis, the controlled versus no-control AUROC difference is .0278 (descriptive 95% interval $[-.0555, .1110]$), while no control has lower forecast MSE in all five realizations (.0421 versus .1905). Full paired results are in Appendix C.7.

Figure 2 shows the same separation across saved Lorenz checkpoints: prediction error changes little while drift-group ranking shifts. At step 2000, the weaker λ_4 arm retains 83 groups versus 60 for the bounded reference, yet has lower AUROC (0.7583 versus 0.7815).

5.3 CONTROLLER-CAPACITY STRESS TEST

Within our controlled-SDE Lorenz stress test, zero action weight with an unbounded controller yields constant drift and AUROC .5 in all five runs. Bounding the same controller restores state-dependent drift and mean AUROC .7436, with descriptive paired 95% improvement interval $[.2103, .2769]$. This is a controller-induced failure: the cap repairs the unpenalized controlled formulation, while removing the controller avoids that failure mechanism altogether. It does not establish that adding control improves graph recovery at the operating action weight.

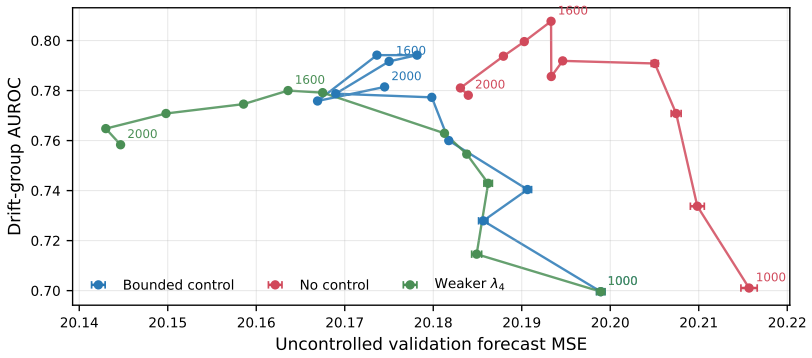

Figure 2: Prediction error against structural ranking across saved checkpoints from exposed Lorenz development seed 105. Points average three numerical forecast evaluations, horizontal bars show their sample SD, not variation across fitted systems. Lines connect checkpoints from 1000 to 2000 updates.

5.4 Matched stochastic and deterministic comparators

We isolate controller and diffusion effects on Lorenz D10 at nominal $p = .3$ using five realizations with byte-identical train, validation, and test splits. The SDE and deterministic arms share the encoder, decoder, sparse drift, proximal schedule, and fitting budget.

Table 2: Matched Lorenz D10 comparison, mean $\pm$ sample SD over five realizations. AUROC includes self-edges, off-diagonal AUROC excludes them. SHD uses the fixed binary threshold.

Arm	AUROC	Off-diag. AUROC	SHD
Full controlled SDE	$.7456 \pm .0157$	$.6608 \pm .0210$	35.0 ± 3.9
No-control SDE	$.7541 \pm .0111$	$.6722 \pm .0148$	36.0 ± 3.9
Deterministic comparator	$.7450 \pm .0182$	$.6601 \pm .0242$	35.2 ± 2.6

The no-control mean AUROC is numerically highest, but paired intervals for its difference from the full SDE and deterministic comparator include zero. The matched comparison therefore resolves no ranking benefit from control or stochastic diffusion in this setting.

5.5 Runtime and scaling

Fitting times and local SCOTCH adapter outcomes are summarized in Appendix B.5, these runs do not establish an equal-accuracy speed claim.

5.6 Scope of the graph measurements

The experiments score dependency among simulator coordinates. Lorenz observations expose the state directly, Glycolysis adds an identity anchor. Validation simulator states calibrate a permutation for graph alignment, which is available for these benchmarks but would not be available for an unlabeled real system. Consequently, the reported AUROC is a controlled measure of drift-graph ranking after benchmark alignment, not evidence that the learned latent coordinates are uniquely identified. The same graph is shared across trajectories within each simulator family, so held-out trajectories test new paths rather than new graph mechanisms.

On the five DREAM3 size-100 networks, one-seed exploratory runs yield mean off-diagonal AUROC of .4868 and .5028 under two frozen profiles, providing no evidence of useful transfer beyond the simulator benchmarks (Appendix C.8).

The cap was selected with development exposure, and Glycolysis's controlled hyperparameters were selected using graph labels. The fresh Glycolysis realizations follow that frozen choice but

use the same graph family. Independent per-trajectory masking, a matched published-configuration SCOTCH study, and additional standard baselines remain open evaluation gates. They matter for claims about irregular-sampling robustness and comparative accuracy, none is inferred from the contextual Table 1 or the local adapter's early stops.

6 Discussion and Conclusion

Our diagnostic study uses PLACID to separate structural drift from data-dependent steering using a control-regularized latent SDE and sparse drift input groups. Small action and accurate dynamics do not certify a graph: a persistent false edge can coexist with vanishing action and bounded control, while coverage, Jacobian smoothness, and a signal margin supply a sufficient oracle recovery condition. Experimentally, the unpenalized controller can cause collapse, bounding it mitigates this self-induced failure without establishing a benefit of control. Proximal updates improve Lorenz parsimony, and both 50-variable settings retain ranking signal despite poor fixed-threshold SHD.

At the operating weights, the component panels show no consistent ranking benefit from control, and the matched three-arm comparison resolves no controller or diffusion benefit on Lorenz D10 at nominal $p = .3$. The evidence is limited to observed-coordinate, development-exposed simulators, Glycolysis tuning is asymmetric, published SCOTCH values use different protocols, and the local adapter timing has one completed seed. Unknown mixtures, broader matched baselines, and guarantees for the interval estimator remain open.

Reproducibility Statement

Appendix B specifies data generation, trajectory splits, architectures, exact loss reductions, optimization schedules, graph alignment, scoring, and runtime boundaries. Appendix A provides proofs and assumptions, Appendix C retains complete component and diagnostic results. Reported runs retain frozen configurations, source and split checksums, and saved graph scores for verification.

AI Use Statement

We used AI tools to aid and polish writing, retrieve and discover related work, support research ideation and execution, and draft sections of this paper. The authors reviewed the AI-assisted text, analyses, proofs, code, and experimental artifacts and take responsibility for the final content.

References

Alexis Bellot, Kim Branson, and Mihaela van der Schaar. Neural graphical modelling in continuous-time: consistency guarantees and algorithms, 2022. arXiv:2105.02522.

Ricky T. Q. Chen, Yulia Rubanova, Jesse Bettencourt, and David K Duvenaud. Neural ordinary differential equations. In *NeurIPS*, 2018.

Luca Donati, Marcus Weber, and Bettina G. Keller. A review of girsanov reweighting and of square root approximation for building molecular markov state models. *Journal of Mathematical Physics*, 63(12), 2022.

Chin-Wei Huang, Jae Hyun Lim, and Aaron C Courville. A variational perspective on diffusion-based generative models and score matching. In *NeurIPS*, pp. 22863–22876, 2021.

Aapo Hyvarinen and Hiroshi Morioka. Unsupervised feature extraction by time-contrastive learning and nonlinear ica. In *NeurIPS*, 2016.

Aapo Hyvarinen and Hiroshi Morioka. Nonlinear ICA of Temporally Dependent Stationary Sources. In *AISTATS*, pp. 460–469, 2017.

Aapo Hyvarinen, Ilyes Khemakhem, and Hiroshi Morioka. Nonlinear independent component analysis for principled disentanglement in unsupervised deep learning, 2023. arXiv:2303.16535.

Ilyes Khemakhem, Diederik Kingma, Ricardo Monti, and Aapo Hyvarinen. Variational autoencoders and nonlinear ica: A unifying framework. In *AISTATS*, pp. 2207–2217, 2020.

Patrick Kidger, James Foster, Xuechen (Chen) Li, and Terry Lyons. Efficient and accurate gradients for neural sdes. In *NeurIPS*, pp. 18747–18761, 2021.

Sebastien Lachapelle, Pau Rodriguez, Yash Sharma, Katie E Everett, Rémi LE PRIOL, Alexandre Lacoste, and Simon Lacoste-Julien. Disentanglement via mechanism sparsity regularization: A new principle for nonlinear ICA. In Bernhard Schölkopf, Caroline Uhler, and Kun Zhang (eds.), *Proceedings of the First Conference on Causal Learning and Reasoning*, volume 177 of *Proceedings of Machine Learning Research*, pp. 428–484. PMLR, 11–13 Apr 2022. URL https://proceedings.mlr.press/v177/lachapelle22a.html.

Christian Léonard. Girsanov theory under a finite entropy condition. *Séminaire de Probabilités XLIV, Lecture Notes in Mathematics*, 2046:429–465, 2012. URL https://arxiv.org/abs/1101.3958.

Xuechen Li, Ting-Kam Leonard Wong, Ricky T. Q. Chen, and David Duvenaud. Scalable gradients for stochastic differential equations. In Silvia Chiappa and Roberto Calandra (eds.), *Proceedings of the Twenty Third International Conference on Artificial Intelligence and Statistics*, volume 108 of *Proceedings of Machine Learning Research*, pp. 3870–3882. PMLR, 26–28 Aug 2020. URL https://proceedings.mlr.press/v108/li20i.html.

Guan-Horng Liu, Yaron Lipman, Maximilian Nickel, Brian Karrer, Evangelos Theodorou, and Ricky T. Q. Chen. Generalized schrödinger bridge matching. In *ICLR*, 2024.

Georg Manten, Cecilia Casolo, Emilio Ferrucci, Søren Wengel Mogensen, Cristopher Salvi, and Niki Kilbertus. Signature kernel conditional independence tests in causal discovery for stochastic processes. In *ICLR*, 2025.

Søren Wengel Mogensen, Daniel Malinsky, and Niels Richard Hansen. Causal learning for partially observed stochastic dynamical systems. In *Proceedings of the 34th Conference on Uncertainty in Artificial Intelligence*, pp. 350–360, 2018. URL https://auai.org/uai2018/proceedings/papers/142.pdf.

Jakob Runge, Sebastian Bathiany, Erik Bollt, Gustau Camps-Valls, Dim Coumou, Ethan Deyle, Clark Glymour, Marlene Kretschmer, Miguel D. Mahecha, Jordi Muñoz-Marí, Egbert H. van Nes, Jonas Peters, Rick Quax, Markus Reichstein, Marten Scheffer, Bernhard Schölkopf, Peter Spirtes, George Sugihara, Jie Sun, Kun Zhang, and Jakob Zscheischler. Inferring causation from time series in earth system sciences. *Nature Communications*, 10(1):2553, 2019.

Alexander Sokol and Niels Richard Hansen. Causal interpretation of stochastic differential equations. *Electronic Journal of Probability*, 19(100):1–24, 2014.

Alexander Y. Tong, Nikolay Malkin, Kilian Fatras, Lazar Atanackovic, Yanlei Zhang, Guillaume Huguet, Guy Wolf, and Yoshua Bengio. Simulation-free Schrödinger bridges via score and flow matching. In Sanjoy Dasgupta, Stephan Mandt, and Yingzhen Li (eds.), *Proceedings of The 27th International Conference on Artificial Intelligence and Statistics*, volume 238 of *Proceedings of Machine Learning Research*, pp. 1279–1287. PMLR, 02–04 May 2024. URL https://proceedings.mlr.press/v238/tong24a.html.

Belinda Tzen and Maxim Raginsky. Neural stochastic differential equations: Deep latent gaussian models in the diffusion limit, 2019. arXiv:1905.09883.

Benjie Wang, Joel Jennings, and Wenbo Gong. Neural structure learning with stochastic differential equations. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=V1GM9xDvIY.

A PATH-LAW IDENTITIES AND THE SCOPE OF STRUCTURAL RECOVERY

A.1 FIXED-JOINT-LAW KL DECOMPOSITION

Set $R(dx, dz) = \nu(dx)Q_x(dz)$ and let its path marginal be $\bar{Q}$. With finite divergences, factor the density as

$$\frac{dR}{d(\nu \otimes P_\theta)}(x, z) = \frac{dR}{d(\nu \otimes \bar{Q})}(x, z)\frac{d\bar{Q}}{dP_\theta}(z).$$

Integrating its logarithm proves

$$\mathbb{E}_\nu \mathrm{KL}(Q_X \| P_\theta) = I_R(X, Z) + \mathrm{KL}(\bar{Q} \| P_\theta).$$

The information term is constant only when the joint law R is held fixed. Holding its path marginal fixed does not suffice. For $\bar{Q} = P^\star$, Girsanov's identity and the common initial law give

$$\mathrm{KL}(P^\star \| P_\theta) = \frac{1}{2}\int \|\sigma(z)^{-1}(\mu^\star(z) - \mu_\theta(z))\|^2 \rho(dz).$$

This nonnegative quantity attains zero at $\mu^\star$. Invertibility implies that any minimizer agrees with $\mu^\star$ ρ-almost everywhere, proving Theorem 4.3. If initial laws differ, add $\mathrm{KL}(p_0^\star \| p_{0,\theta})$, an argument that omits it is invalid. For a fixed posterior joint law and fixed candidate initial law, the expected initial conditional KL is constant, so expected conditional energy has the same minimizers. It is not equal to the drift mismatch without that constant. No weak-convergence-to-KL step is used. Occupation-measure agreement alone neither controls derivatives nor proves finite-sample graph selection.

A.2 GRAPH RECOVERY ON AN OCCUPIED DOMAIN

Let μ_θ be a global minimizer in Theorem 4.3. Its difference from $\mu^\star$ is zero ρ-almost everywhere. If the continuous difference were nonzero at some $z \in U$, continuity would give a nonempty open neighborhood $V \subset U$ on which it remains nonzero. Full support gives $\rho(V) > 0$, a contradiction. Thus the two drifts agree at every point of U. Since U is open and both are C^1 there, their partial derivatives agree throughout U. Each source-target derivative is identically zero on U for one drift exactly when it is for the other. This proves Corollary 4.4. Without coverage, equality need not extend to an unvisited open region: smooth drift perturbations supported there leave the occupation integral unchanged.

A.3 FROM SMALL ORACLE EXCESS ACTION TO DERIVATIVE ACCURACY

The exact occupied-domain corollary alone gives no quantitative guarantee for approximate minimizers. The following result makes the extra coverage and smoothness requirements explicit. Let v_D be the volume of the unit Euclidean ball in $\mathbb{R}^D$.

Theorem A.1 (Quantitative derivative and thresholded-support bound)**.** *Under the oracle setup of Theorem 4.3, suppose $\mathcal{A}(\theta) \le \varepsilon$. Let $K \subset U$ be nonempty and compact, with U open and $B_r(x) \subset U$ for every $x \in K$, for some $r > 0$. Assume $\rho(B) \ge m\,\mathrm{Leb}(B)$ for all Borel $B \subset U$, with $m > 0$, and $\|\sigma(z)\|_{\mathrm{op}} \le s$ on U. The domain U must have finite volume: since $\rho(U) \le T$, this coverage condition implies $m\,\mathrm{Leb}(U) \le T$. Write $e = \mu_\theta - \mu^\star$, assume both drifts are C^1 on U, and assume each ∇e_j is L-Lipschitz on U. For every $0 < h \le r$,*

$$\max_{i,j}\sup_{x\in K}|\partial_i e_j(x)| \le E_h := s\sqrt{\frac{2(D+2)\varepsilon}{m v_D}}\,h^{-D/2-1} + \frac{\sqrt{D+2}\,Lh}{2}. \tag{9}$$

Define $S_{ij}(\mu) = \sup_{x\in K}|\partial_i \mu_j(x)|$ and the graph on K by $G_{ij}^K = \mathbf{1}\{S_{ij}(\mu^\star) > 0\}$. If every nonzero $S_{ij}(\mu^\star)$ exceeds $2E_h$, then $\mathbf{1}\{S_{ij}(\mu_\theta) > E_h\} = G_{ij}^K$ for all i, j.

Proof. Invertibility and the diffusion bound give $\|e(z)\|_2 \le s\|\sigma(z)^{-1}e(z)\|_2$ on U. For any $x \in K$, occupation coverage and the definition of $\mathcal{A}$ therefore imply

$$\int_{B_h(x)} e_j(z)^2\,dz \le \frac{1}{m}\int_{B_h(x)} \|e(z)\|_2^2\,\rho(dz) \le \frac{2s^2\varepsilon}{m}.$$

Let Y be uniform on $B_h(0)$. Lipschitz continuity of the gradient gives

$$e_j(x+Y) = e_j(x) + \nabla e_j(x)^\top Y + R(Y), \qquad |R(Y)| \le \tfrac{L}{2}\|Y\|_2^2.$$

Symmetry yields $\mathbb{E}Y_i = 0$ and $\mathbb{E}Y_iY_\ell = \mathbf{1}\{i = \ell\}h^2/(D+2)$. Multiplying by Y_i and taking expectations gives

$$|\partial_i e_j(x)| \le \frac{D+2}{h^2}\left(|\mathbb{E}[e_j(x+Y)Y_i]| + |\mathbb{E}[R(Y)Y_i]|\right).$$

After multiplication by $(D+2)/h^2$, the first contribution is at most $\sqrt{D+2}\,h^{-1}\sqrt{\mathbb{E}e_j(x+Y)^2}$ by Cauchy–Schwarz. For the second, $\|Y\| \le h$ and $\mathbb{E}|Y_i| \le h/\sqrt{D+2}$ give an upper bound $Lh\sqrt{D+2}/2$ after multiplying by $(D+2)/h^2$. Since the ball volume is $v_D h^D$, the integral bound proves (9), uniformly in x, i, j. Finally, $|S_{ij}(\mu_\theta) - S_{ij}(\mu^\star)| \le E_h$. An absent edge has estimated score at most E_h, a true edge with score strictly above $2E_h$ has estimated score strictly above E_h. □

For fixed D, m, s, L, r, choosing h proportional to $\varepsilon^{1/(D+4)}$ for sufficiently small positive ε gives an $O(\varepsilon^{1/(D+4)})$ derivative bound. This is a conditional oracle excess-action rate, not a sample-size rate or a bound on the current training loss. Coverage and smoothness constants can deteriorate with dimension. Recovery concerns the graph witnessed on K, to infer the graph on all of U, every edge active on U must have a sufficiently strong witness on K. The thresholded scores here are functional derivative scores, not the first-layer group norms used in the experiments.

A.4 Proof of the small-action counterexample

For Proposition 4.5, the true stationary initial law is $\mathcal{N}(0, I/2)$. The drift-error norm is bounded by $1/n$, so its integrated squared norm gives $\mathcal{A}(\mu_n) \le T/(2n^2)$. The Jacobian is

$$J_{\mu_n}(z) = \begin{pmatrix} -1 & \cos(nz_2) \\ 0 & -1 \end{pmatrix}.$$

Its operator norm is at most its Frobenius norm $\sqrt{3}$, uniformly in n and z. Thus all candidate SDEs have globally Lipschitz drifts with a common bound. The bounded drift difference satisfies Novikov's condition on every finite horizon, so the change-of-measure and action identities apply. Yet $\partial_2(\mu_n)_1(0) = 1$, while $\partial_2\mu_1^\star \equiv 0$. The compensating control is $u_n = (-n^{-1}\sin(nz_2), 0)$, whose infinity norm is at most $1/n < c$ when $n > 1/c$. Its energy vanishes even though the candidate graph is wrong. The Lipschitz constant of the drift-error gradient grows with n, explaining why this example does not contradict Theorem A.1. This is a statement about admissible smooth drifts and controls, not a claim that a particular neural architecture represents every member of the sequence exactly.

A.5 Sharp compensation limits of a componentwise cap

If $|u_k(t)| \le c$, then $\|u_t\|_2^2 \le Dc^2$ pointwise, giving $\frac{1}{2}\int_0^T \|u_t\|_2^2 dt \le TDc^2/2$, expectation preserves the bound. For invertible $\sigma(z)$ and $v = \sigma(z)^{-1}r$, the squared residual in diffusion coordinates is $\|v-u\|_2^2$. Minimization over the box $[-c,c]^D$ separates by coordinate. Each minimizer is $u_k = \mathrm{clip}(v_k, -c, c)$, with squared residual $(|v_k| - c)_+^2$. Summing proves (5). In particular, any admissible controller leaves at least this residual, and a coordinate with $|v_k| > c$ cannot be compensated exactly. The clipped vector need not be representable by the chosen neural policy, the equality is a capacity statement over all vectors in the closed box. For the tanh policy, boundary optima may only be approached, since finite inputs give strict componentwise inequalities.

For diagonal diffusion $\sigma(z) = \mathrm{diag}(g_1, \ldots, g_D)$, with $g_k > 0$ as in the implemented softplus parameterization, the residual in drift units also separates, giving the sharper state-dependent identity

$$\min_{\|u\|_\infty \le c} \|r - \sigma(z)u\|_2^2 = \sum_{k=1}^{D} (|r_k| - cg_k)_+^2.$$

No uniform upper bound on g_k is needed for this pointwise identity. If $\|\sigma(z)\|_{\mathrm{op}} \le s$ is additionally available, the earlier conservative bound follows from the reverse triangle inequality:

$\|r - \sigma(z)u\|_2 \geq (\|r\|_2 - s\sqrt{D}\,c)_+$. The cap's Euclidean radius grows as $\sqrt{D}$, and learned diffusion can enlarge the allowable correction in drift units. These facts prevent the cap alone from establishing drift recovery. The actual discrepancy r is unknown during training, so the identities are not empirical error certificates for our fits.

A.6 Affine transformations and support separation

Assumption A.2 (Hypothetical coordinate gauge)**.** The admissible affine representations $\hat{Z} = LZ + c$ remove diagonal scaling ambiguity, so admissible monomial matrices reduce to signed permutations.

Assumption A.3 (Strict support separation)**.** Among admissible invertible L, every nonmonomial transformation has strictly more globally active drift-Jacobian entries than the native representation.

Proposition A.4 (Conditional support-count implication)**.** *Suppose affine equivalence is already established and Assumptions A.2 and A.3 hold. Minimizing the exact global Jacobian support count over those representations selects a signed permutation, with translations unresolved.*

Ito's formula for $\hat{Z} = LZ + c$ yields

$$\hat{\mu}(\hat{z}) = L\mu^\star(L^{-1}(\hat{z} - c)), \quad J_{\hat{\mu}}(\hat{z}) = LJ_{\mu^\star}(L^{-1}(\hat{z} - c))L^{-1}.$$

The diffusion transforms too: $\hat{\sigma}(\hat{z}) = L\sigma(L^{-1}(\hat{z}-c))$, and the decoder becomes $\hat{g}(\hat{z}) = g(L^{-1}(\hat{z}-c))$. The initial law is pushed forward. For fixed isotropic covariance $a = \sigma_0^2 I$, retaining that covariance requires $LL^\top = I$. Marginal variance normalization alone does not remove orthogonal ambiguity.

Define $N(\mu) = \sum_{i,j} \mathbf{1}\{\partial_i \mu_j \not\equiv 0\}$ on the transformed domain, including diagonal entries. Monomial transforms preserve this count. Under strict support separation all other transforms have larger count, so every minimizer is monomial. The assumed gauge reduces these to signed permutations. This proves Proposition A.4 only under its stated separation assumption. It does not prove a group-lasso relaxation preserves minimizers.

For a counterexample without separation, take the stationary isotropic Ornstein–Uhlenbeck model $dZ_t = -aZ_t dt + \sigma_0 dW_t$, $a > 0$. Every orthogonal L leaves the drift $-a\hat{Z}$, isotropic diffusion and isotropic stationary initial law unchanged. Dense rotations have the same Jacobian support count. Thus sparsity and an isotropic diffusion do not universally select native coordinates. A continuous-time identification result needs additional verifiable conditions, no benchmark witness is established here.

A.7 Input-group sensitivity bound

For $\mu_j(z) = F_j(W_j z + b_j)$, the chain rule gives $\partial_i \mu_j(z) = \nabla F_j(W_j z + b_j)^\top w_{ji}$. Cauchy–Schwarz proves Proposition 4.6. In the implemented drift network, all layers after W_j are specific to target j, with unit-parameter ELU activations. Their scalar derivative lies in $(0, 1]$. If $A_{j,1}, \ldots, A_{j,q}$ are the downstream linear maps in that branch, submultiplicativity of operator norms gives

$$\|\nabla F_j\|_2 \leq C_j := \prod_{\ell=1}^{q} \|A_{j,\ell}\|_{\mathrm{op}}, \qquad S_{ij}(\mu_\theta) \leq C_j \|w_{ji}\|_2.$$

The experimental raw edge score is precisely $\|w_{ji}\|_2$. Combining this inequality with Theorem A.1, under that theorem's additional assumptions, yields

$$C_j \|w_{ji}\|_2 \geq \big(S_{ij}(\mu^\star) - E_h\big)_+.$$

Thus an edge with true derivative strength above E_h cannot have a zero input block in a candidate satisfying the stated excess-action bound. This is only a necessary condition: downstream cancellations can make a nonzero group functionally inactive, and C_j varies across targets and fits. The result establishes neither false-edge exclusion from nonzero scores nor calibrated ranking by group norms.

A.8 Limits of an amortization-error bound

For the fixed-law oracle functional, differentiating under the integral gives

$$\nabla_\theta \mathcal{A} = -\mathbb{E}_{P^\star} \int_0^T (D_\theta \mu_\theta)^\top (\sigma\sigma^\top)^{-1} (\mu^\star - \mu_\theta)\, dt.$$

If a residual in control coordinates is perturbed by δ, with $\|\delta\| \le \epsilon$, the corresponding fixed-law gradient expression changes in norm by at most

$$\epsilon\, \mathbb{E}_{P^\star} \int_0^T \|(D_\theta \mu_\theta)^\top \sigma^{-\top}\|_{\mathrm{op}}\, dt.$$

This assumes that the law and diffusion are fixed and differentiation is justified. It is not a bound on the full implemented training gradient, which also depends on the learned path distribution and inference parameters. It implies neither coordinatewise gradient signs nor recovery of graph edges above an unspecified noise floor.

B Reproducibility: data, estimator, evaluation, and runtime

This section specifies the frozen estimator behind the reported results. Dataset-specific settings are intentional and are not pooled across cohorts. The theoretical oracle and the implemented interval estimator are distinguished in Section 4 and Appendix B.7.

B.1 Data generation and trajectory splits

We generate 100 independent trajectories per realization, with 100 observation times on $\mathrm{linspace}(0, 100, 100)$, using Euler–Maruyama with nominal solver step .005. Lorenz96 uses cyclic indexing and dynamics

$$dX_{t,d} = [(X_{t,d+1} - X_{t,d-2})X_{t,d-1} - X_{t,d} + 10]dt + .5\, dW_{t,d}, \tag{10}$$

with independent standard-normal initial states. For nominal $p \in \{.3, .6\}$, a shared random time mask is followed by a seeded row subsample retaining at most 7000 or 4000 observations. The second stage can create trajectory-specific grids, so p is not the exact retained fraction.

The Glycolysis generator is the seven-variable SCOTCH-derived yeast model of Wang et al. (2024), implemented in `scotch/dataset_generation/example_sdes/yeast_glycolysis.py`. It retains all 10,000 observations and uses noise scale .01. Exact kinetic constants, initial-state ranges, and simulator settings are frozen in that source and the archived run configurations. Its stored dependency matrix is target-row and is transposed for our source-row scores. Glycolysis has 16 off-diagonal edges among 42 candidates, Lorenz96 has $3D$ among $D(D-1)$, plus D self-edges.

Trajectory IDs are shuffled with a dedicated split seed and partitioned 60/20/20 into training, validation, and test sets. Row masks use a separate seed. Glycolysis coordinates are standardized with population means and standard deviations from retained training observations, Lorenz is not standardized. No interval crosses a trajectory boundary. Validation states calibrate alignment, and test trajectories are excluded from hyperparameter and checkpoint selection. Each simulator family retains one graph across trajectories, so trajectory splits do not create new graph families.

Benchmark seeds 101–105 use separately seeded data, masks, and splits. Glycolysis selects (β, λ_4) from six prespecified combinations on seeds 101–103, then freezes $(1, 5 \cdot 10^{-4})$ for seeds 111–115. The main, component, and comparator cohorts retain their archived split hashes, the matched three-arm Lorenz comparison uses byte-identical split artifacts.

B.2 Architectures and initialization

Latent and observation dimensions agree. Lorenz uses linear $D \to D$ encoder mean and log-variance heads and a linear decoder. Glycolysis uses coordinate-specific ELU encoder maps and target-specific decoder maps. Encoder samples use $z = m + \exp(\ell/2) \odot \epsilon$, alignment uses means.

The sparse drift has target-specific branches with widths $10, 1$ for Lorenz and $7, 7, 1$ for Glycolysis, without cross-target bypasses. Diagonal diffusion uses the same branch widths and a final softplus.

The ELU controller receives state, sinusoidal time features, and fixed source context, it has two width-64 hidden layers for Lorenz and one width-8 layer for Glycolysis, with componentwise output $2\tanh(u_{\mathrm{raw}}/2)$.

First-layer structural weights use the documented small initialization, controller weights use Xavier initialization. Exact layer maps, initialization constants, numerical dtypes, and seeds are retained in source and archived configurations. The matched deterministic comparator keeps the encoder, decoder, drift, proximal schedule, and budget while disabling control, Brownian increments, diffusion in the integration step, and the diffusion penalty. It is not an implementation of the original NGM method.

NGM-GL released-code adaptation. The matched NGM comparison uses the released irregular-time Lorenz trainer at upstream commit `6020e6859d8eec08d3d216e347f2458c67036a91`, with width $[D, 12, 1]$, fixed group penalty 0.05, threshold 0.09, and validation plateau stopping. Each trajectory is fitted as an independent sequence so that concatenation introduces no artificial transitions. The five runs use the same saved Lorenz D10 trajectory splits as the PLACID comparison. The primary NGM row reports the saved 2000-update checkpoint, validation-selected and final checkpoints are summarized in the main text. This is a released-code adaptation and is not the adaptive-group variant described in the original paper appendix.

B.3 Exact losses, reductions, and updates

Let N_o be the number of retained training observations, N_I the number of valid within-trajectory intervals, m_k, ℓ_k the encoder mean and log-variance, and $\widetilde{z}_{k+1}$ the integrated endpoint initialized at m_k. Define $a_D = 1$ for Lorenz and $a_D = D$ for Glycolysis. The losses are

$$\mathcal{L}_{\mathrm{rec}} = \frac{1}{2N_o a_D} \sum_k \|g_\vartheta(z_k) - x_k\|_2^2, \tag{11}$$

$$\mathcal{L}_{\mathrm{cons}} = \frac{1}{2N_I a_D} \sum_{k \in \mathcal{I}} \left[\|m_{k+1} - \widetilde{z}_{k+1}\|_2^2 + \sum_j \{\exp(\ell_{k+1,j}) - 1 - \ell_{k+1,j}\} \right], \tag{12}$$

$$\mathcal{L}_{\mathrm{id}} = \frac{1}{2N_o D} \sum_k \|z_k - x_k\|_2^2. \tag{13}$$

Thus Lorenz sums over coordinates before averaging observations, whereas Glycolysis averages both. Reconstruction and identity terms use sampled encoder states, interval starts use encoder means. The consistency loss is an endpoint discrepancy plus variance regularization, not a transition KL.

For each eligible trajectory r, control energy sums across its intervals and Euler substeps:

$$\mathcal{E}_u = \frac{1}{N_{\mathrm{tr}}} \sum_r \sum_{k \in \mathcal{I}_r} \sum_q \frac{h_{kq}}{2} \|u_{kq}\|_2^2.$$

The final substep is shortened to the observation time. There is no energy term at terminal observations and no division by total duration in the training objective. All active intervals are advanced together, with fresh Brownian increments of covariance $h_{kq} I$. The diffusion schedule scales both the Brownian term and the control shift.

For hidden-channel magnitude matrices $E_h \in \mathbb{R}^{D \times D}$ and $D_h \in \mathbb{R}^{D \times D}$ formed respectively from the encoder mean head and decoder first layer, the coupling term is

$$\mathcal{R}_{\mathrm{coupling}} = \frac{\lambda_1}{2} \sum_h (\|E_h D_h\|_F^2 + \|D_h E_h\|_F^2).$$

The ridge term is $\mathcal{R}_{\mathrm{ridge}} = \frac{\lambda_2}{2}\|W_{\mu,1}\|_F^2 + \frac{\lambda_5}{2}\|W_{\sigma,1}\|_F^2$. These are explicit first-layer penalties, not optimizer weight decay over all parameters. Biases and downstream layers are not included in this ridge term.

Table 3: Frozen fitting settings. K denotes joint-training updates, pretraining updates are additional. Lorenz dimensions use the same settings within each missingness level.

Setting	Lorenz $p = .3$	Lorenz $p = .6$	Glycolysis selected
Pretraining / K	3000 / 2000	3000 / 2500	5000 / 1000
Maximum integration step	.05	.1	.01
$c_{\mathrm{rec}}, c_{\mathrm{cons}}, c_{\mathrm{id}}$	$1, 1, 0$	$1, 1, 0$	$30, .25, 15$
β	.01	.01	1
$\lambda_1, \lambda_2, \lambda_5$	$.1, .05, .05$	$.1, .05, .05$	$1, .01, .01$
λ_3, λ_4	$.01 \to .04$	$.01 \to .04$	$.0005, .0005$
Adaptive exponent γ	.5	.5	.25
Diffusion scale	1	1	$.01 + .99 \min(k/3000, 1)$
Binary threshold	.1	.1	.2

Pretraining uses a separate Adam optimizer for encoder and decoder. Lorenz minimizes $\mathcal{L}_{\mathrm{rec}}$ plus $\frac{1}{2N_o}\sum_{k,j}(m_{k,j}^2 + \exp \ell_{k,j} - 1 - \ell_{k,j})$, Glycolysis minimizes $\mathcal{L}_{\mathrm{rec}} + 5\mathcal{L}_{\mathrm{id}}$. Joint training uses a fresh Adam optimizer over all networks, learning rate .003, default moments $(.9, .999)$, and numerical constant 10^{-8}. Each update uses the full retained training set and all its valid intervals, there is no trajectory minibatching in PLACID. The configured learning-rate decays occur after the reported budgets, so the rate remains .003.

Adaptive group proximal updates. After each Adam step, adaptive weights are recomputed and proximal steps with $\eta = .01$ are applied to the encoder mean head, decoder first layer, and drift input groups. For Lorenz, $\lambda_3 = \lambda_4 = .01$ up to zero-based step $k = 1000$ and $.01 + .03(k - 1000)/(K - 1000)$ thereafter. Glycolysis keeps these penalties fixed, its final diffusion scale is .33967. Drift weights are $w_{ij} = 1/\|W^{(1)}_{j,:,i}\|_2^{2\gamma}$. Encoder/decoder adaptive weights use the reciprocal γ power of the product of paired encoder and decoder group norms. The adaptive denominator floor is zero in these runs. A zero denominator therefore yields an infinite weight, subtracting its positive threshold and clamping at zero removes the group. Direction normalization uses a separate 10^{-12} safeguard. For a drift block v, the implemented update is

$$v^+ = \frac{v}{\max(\|v\|_2, 10^{-12})} \max(0, \|v\|_2 - \eta\lambda_4 w_{ij}). \tag{14}$$

For finite fixed weights and norms above the normalization safeguard, this is group soft thresholding. Adaptive reweighting and joint optimization provide no exact support-selection guarantee. No-control sets both the drift correction and action to zero. No-proximal ablations skip all three proximal updates while retaining coupling and ridge. The same saved splits and remaining schedules are used within each comparison.

B.4 Graph scoring, alignment, and uncertainty

Raw source-row scores are $S_{ij} = \|W^{(1)}_{j,:,i}\|_2$. A Hungarian assignment maximizes the sum of absolute Pearson correlations between validation encoder means and validation simulator coordinates. With the resulting permutation P, scores and binary adjacency are aligned as $PSP^\top$ and $P\widehat{G}P^\top$. No graph labels enter this alignment, but it uses simulator coordinates and is a benchmark calibration procedure.

For binary selection, each score column is transformed as

$$\widetilde{S}_{ij} = \frac{S_{ij} - \min_i S_{ij}}{\max_i S_{ij} - \min_i S_{ij} + 10^{-5}}, \qquad \widehat{G}_{ij} = \mathbf{1}\{\widetilde{S}_{ij} > \tau\}.$$

AUROC and average precision (reported as AUPRC) use raw continuous scores, not this thresholded matrix. Main ranking metrics and reported component SHD include self-edges, off-diagonal analyses explicitly remove them. Directed SHD is false positives plus false negatives, so a reversal costs two. FDR is false positives divided by the number of predicted edges, with a denominator of at least one. Paired 95% intervals use a Student t distribution with four degrees of freedom over five realization-level differences. They are descriptive and unadjusted for development selection or multiple comparisons. Monte Carlo rollout repetitions are not treated as independent model fits.

Table 4: Measured PLACID fitting time, including pretraining, over five realizations per setting. These are observed wall times, not extrapolations.

Lorenz setting	Joint updates	Mean $\pm$ SD (min)	Range (min)
D10, $p = .3$	2000	17.9 ± 4.9	13.5–23.1
D10, $p = .6$	2500	23.9 ± 6.3	16.9–32.6
D50, $p = .3$	2000	16.9 ± 4.4	12.4–21.4
D50, $p = .6$	2500	23.5 ± 5.3	17.2–30.0

B.5 Runtime measurements and local SCOTCH configuration

PLACID timing surrounds the complete `learn` call, including pretraining, joint training, logging, and in-training diagnostics, but excludes data generation and final graph export. Cohort provenance records a Tesla V100-SXM2-32GB GPU. Runs span allocations and variable physical-time grids, the similar D10 and D50 times do not imply dimension-independent complexity.

The completed local SCOTCH D10 seed-103 run uses exactly the same saved training split as its PLACID counterpart: 60 trajectories, 3840 observations, and one grid of 64 times. It groups these into six optimizer packs of ten. SCOTCH's timer surrounds `Trainer.fit`, including checkpointing every 250 updates and training-framework overhead, but excluding data preparation and final graph export. Its recorded GPU execution takes 9188.55 seconds for 5000 updates, PLACID takes 807.32 seconds for 3000 pretraining plus 2000 joint updates, giving $11.38\times$. The SCOTCH result record does not itself store the GPU model, so this is an observed local pipeline comparison, not a hardware-normalized algorithm benchmark. Neither timing includes hyperparameter search. The seed was selected for its single shared grid before its training outcome was observed.

The local SCOTCH adapter uses learning rate $.003$, graph sparsity weight 500, solver step 1, residual connections off, and one-step learning-rate warm-up. SCOTCH's published Lorenz configuration instead describes residual connections and 100 epochs of warm-up and 5000 training epochs. Its Section 6.1 states ten trajectories, whereas Appendices D.3 and D.4 state 100 generated trajectories. We do not resolve that internal sample-count discrepancy or assume the published cohort equals our saved split. Therefore the local run and the published Table 1 reference are distinct comparisons. At D50, the four post-pilot runs stop at 168, 159, 36, and 36 updates after respectively 3066, 391, 83, and 602 seconds. Three reach the frozen loss-ratio stop and one has a nonfinite gradient. These are time-to-stop measurements, no full-budget SCOTCH D50 time or speedup ratio is inferred.

B.6 Artifacts and verification

The recorded D10 PLACID environment uses PyTorch 2.7.1, NumPy 1.26.4, SciPy 1.13.1, and torchsde 0.2.6. The local SCOTCH execution uses a separate environment, this software difference is another reason to interpret the timing ratio as a pipeline measurement. The experiment archives retain frozen source bundles, configuration JSON, source checksums, split membership and array hashes, continuous graph scores, binary adjacency, and model checkpoints. Objective and metric versions are `placid-physical-time-v2` and `placid-graph-v2`, the split protocol is `trajectory-split-v1`. The benchmark runner is `run_revision.py`, the fixed search and harder Lorenz panel are specified by `run_harder_glycolysis.py`, component results retain their separate source snapshots. Saved arrays and scores, with the corrected Glycolysis orientation, are the basis of table verification. The manuscript build does not rerun training or select checkpoints. Exact regeneration requires the source snapshot and environment recorded for the relevant cohort, rather than an arbitrary later working-tree version.

B.7 Euler-step control KL and implementation limits

At a fixed source state z and step length $h > 0$, let both Euler kernels have covariance $ha(z)$ with $a = \sigma\sigma^\top$ positive definite. Their means are $z + h\mu(z)$ and $z + h[\mu(z) + \sigma(z)u]$. The Gaussian KL formula gives

$$\mathrm{KL}(K_h^u(z,\cdot)\|K_h^0(z,\cdot)) = \frac{h}{2}u^\top\sigma^\top a^{-1}\sigma u = \frac{h}{2}\|u\|^2, \tag{15}$$

where the last equality assumes square invertible σ. For rectangular or degenerate diffusion the equality needs a range/minimum-energy formulation. For a single discretized path law, the chain rule adds its initial KL and expected conditional step KLs. Refinement approaches the continuous energy only with appropriate convergence and integrability conditions.

The identity is exact for the local Euler kernels. It is not generally an exact Gaussian formula for finite-gap SDE transition laws. The corrected implementation explicitly integrates control energy over physical within-path intervals, including partial final solver steps, and averages trajectory integrals. Its endpoint consistency loss is separate. Restarting intervals from encoder means does not establish one global posterior path law, and endpoint MSE is not interchangeable with either conditional transition KL or Girsanov energy. Learned diagonal diffusion is shared between controlled and uncontrolled rollouts of each fitted model, but it is not a known diffusion held fixed across the oracle comparisons in Theorem 4.3.

C Complete experimental results and diagnostics

C.1 Protocol and reporting scope

We report two separate panels. The main-settings panel contains five bounded fits per dataset: Lorenz-96 with nominal missingness $p = 0.3$ and Glycolysis without additional missingness. The matched action/control panel contains five arms on each of five realizations per dataset. Its saved reference data and original dataset-specific settings are shared across arms, it is not pooled with the main-settings panel. Observations expose simulator coordinates, and Glycolysis additionally uses an identity anchor. These experiments measure observed-coordinate drift-graph recovery rather than disentanglement of unknown mixtures.

Splits use disjoint trajectories and normalization uses training data only. Graph alignment is calibrated on validation simulator states. AUROC and AUPRC use continuous first-layer group scores, thresholded adjacency uses the existing column-normalized rule. Primary graph scores include diagonal entries, we also report off-diagonal AUROC. Directed SHD counts insertions and deletions, so reversing an edge costs two. For Lorenz D10, the empty graph has directed SHD 40 against 40 true edges including self-edges. The single completed local SCOTCH adapter D10 seed-103 run has directed SHD 42 at its saved threshold, its configuration differs from the published model, so this is diagnostic only. Forecasts start from source encoder means and use source context, averaging eight decoded SDE draws for the matched panel. Errors average coordinates, intervals within trajectories, then trajectories. Lorenz errors use its observed units and Glycolysis uses training-standardized units, so MSE magnitudes should not be compared across datasets.

The initial main and matched-panel realizations have development exposure. Seed 101 selected the cap, seeds 102–105 are subsequent matched development realizations, not an untouched final test. Means use sample SD. Paired intervals below are descriptive, unadjusted 95% Student intervals over five realizations, not corrections for multiple comparisons or adaptive development. Primary endpoints remain 2000 main steps for Lorenz and 1000 for Glycolysis. All 50 matched records and ten main-setting records passed configuration, split-hash and saved-graph checks. These counts describe separate reporting panels, not 60 independent datasets.

The following development cohorts are separate from the frozen selected-setting evaluation.

Dataset	n	AUROC	AUPRC	Off-diagonal AUROC
Lorenz $p = 0.3$	5	0.7456 ± 0.0157	0.6993 ± 0.0276	0.6608 ± 0.0210
Glycolysis Full	5	0.5719 ± 0.0603	0.5590 ± 0.0285	0.5851 ± 0.0530

Table 5: Verified bounded-control main settings, mean and sample SD over five exposed development realizations. Fixed final checkpoints, no matched external-baseline claim.

C.2 Main settings and controller-induced collapse

The main-settings AUROCs are 0.7456 ± 0.0157 for Lorenz and 0.5719 ± 0.0603 for Glycolysis (Table 5). Off-diagonal Lorenz AUROC is lower, 0.6608 ± 0.0210, showing the contribution of

Dataset	Arm	n	AUROC	F1	Prior MSE
Lorenz	Original	5	0.7462 ± 0.0438	0.6192 ± 0.0294	20.1596 ± 0.2982
Lorenz	Zero action weight	5	0.5000 ± 0.0000	0.0000 ± 0.0000	47.4179 ± 2.8743
Lorenz	No control	5	0.7639 ± 0.0425	0.6288 ± 0.0344	20.1669 ± 0.3041
Lorenz	Bounded, zero action weight	5	0.7436 ± 0.0268	0.6500 ± 0.0281	20.5645 ± 0.2436
Lorenz	Bounded, original weight	5	0.7540 ± 0.0400	0.6279 ± 0.0293	20.1589 ± 0.2989
Glycolysis	Original	5	0.5729 ± 0.0573	0.5456 ± 0.0463	0.1990 ± 0.0687
Glycolysis	Zero action weight	5	0.5831 ± 0.0469	0.5690 ± 0.0355	1.5135 ± 0.3853
Glycolysis	No control	5	0.5886 ± 0.0178	0.5693 ± 0.0484	0.0421 ± 0.0014
Glycolysis	Bounded, zero action weight	5	0.6054 ± 0.0310	0.5970 ± 0.0358	0.0637 ± 0.0088
Glycolysis	Bounded, original weight	5	0.5722 ± 0.0603	0.5571 ± 0.0381	0.1905 ± 0.0609

Table 6: Matched action/control development panel. Original dataset-specific settings and final budgets, mean and sample SD over five realizations. Seed 101 is candidate-discovery evidence. These settings are not pooled with the separate main-settings panel.

Table 7: Secondary metrics for the PLACID benchmark rows of Table 1. Mean $\pm$ sample SD over five realizations. AUPRC and directed SHD include self-edges, off-diagonal AUROC excludes them. SHD uses the declared fixed binary threshold.

Setting	AUPRC	Off-diagonal AUROC	Directed SHD
Lorenz D10, $p = .3$	$.6993 \pm .0276$	$.6608 \pm .0210$	35.0 ± 3.9
Lorenz D10, $p = .6$	$.6832 \pm .0350$	$.6138 \pm .0684$	33.4 ± 4.7
Lorenz D50, $p = .3$	$.4716 \pm .0208$	$.6657 \pm .0192$	366.2 ± 34.1
Lorenz D50, $p = .6$	$.4014 \pm .0686$	$.5945 \pm .0376$	343.4 ± 38.0
Glycolysis, selected	$.6001 \pm .0445$	$.6558 \pm .0243$	21.6 ± 1.1

diagonal entries to the headline score. Glycolysis graph recovery remains weak. We do not compare these corrected scores directly with historical binary-score AUROC or quoted baseline results under unmatched protocols.

In the matched Lorenz stress test, setting the action weight to zero while retaining unbounded control produces constant drift and AUROC 0.5 in all five realizations. The smooth cap $u = 2\tanh(u_{\mathrm{raw}}/2)$ restores state-dependent drift and mean AUROC 0.7436. The paired AUROC gain is 0.2436, with descriptive interval $[0.2103, 0.2769]$. The corresponding Glycolysis comparison changes AUROC from 0.5831 to 0.6054 and uncontrolled MSE from 1.5135 to 0.0637. Its paired AUROC gain is 0.0222, with descriptive interval $[-0.0462, 0.0907]$, so the graph-ranking benefit is unresolved. Unbounded Glycolysis retains state-dependent drift, so this is controller reliance rather than the exact constant-drift collapse seen in Lorenz.

The cap is not a complete resolution of the drift/control decomposition. In the Glycolysis zero-weight comparison, mean diffusion RMS rises from 0.0604 to 0.1135. Bounding u does not bound σu independently of diffusion, and neither effect establishes an identifiable coordinate gauge.

C.3 Does retaining control help at the original action weight?

At the original action weight, the cap changes mean AUROC from 0.7462 to 0.7540 on Lorenz and from 0.5729 to 0.5722 on Glycolysis. The Lorenz paired gain is 0.0079, interval $[0.0012, 0.0146]$, the corrected Glycolysis comparison is approximately flat. These development comparisons describe sensitivity to controller capacity, they do not establish a reason to retain control.

The controller-disabled comparator is particularly informative. Lorenz mean AUROC is 0.7639 without control versus 0.7540 with bounded control, their paired difference, bounded minus no control, is -0.0098, interval $[-0.0330, 0.0133]$. On Glycolysis the corresponding values are 0.5886 versus 0.5722, with paired difference -0.0164 and interval $[-0.0957, 0.0630]$. No control also yields lower Glycolysis uncontrolled prediction MSE: 0.0421 versus 0.1905. The paired MSE difference is 0.1485, interval $[0.0734, 0.2235]$, and favors no control in every realization. Off-diagonal Glycolysis ranking has paired difference 0.0231 and interval $[-0.0464, 0.0926]$. We therefore report the full metric pattern rather than claiming uniform domination or a universal benefit from removing control.

Arm	Selected step	Selected AUROC	Final AUROC	ΔMSE
Bounded reference	1900	0.7758	0.7815	-0.00758
No control	1900	0.7810	0.7781	-0.00089
Held λ_4	1900	0.7648	0.7583	-0.00167

Table 8: Saved-checkpoint development diagnostic. Each evaluation seed and selection using the mean MSE across seeds chooses step 1900. AUROC includes diagonal entries and uses continuous scores. ΔMSE is selected minus final, averaged across evaluation seeds. This is not a validated stopping rule.

The cap repairs an unpenalized-controller stress case, while the original-weight controlled model has no demonstrated advantage over the controller-disabled comparator in these panels. This distinction motivates interpreting the cap as capacity regularization rather than a universal accuracy improvement.

C.4 Corrected checkpoint and pruning diagnostics

We analyze one exposed Lorenz development realization (seed 105), using saved checkpoints from the original bounded and controller-disabled fits and a matched bounded continuation from step 1000. The continuation holds $\lambda_4 = 0.01$ while retaining the λ_3 ramp and the original 2000-step horizon. This is separate from the five-realization main-paper settings batch, no datasets or training settings are pooled.

For each arm, selection minimizes uncontrolled decoded validation interval MSE among checkpoints 1000, 1100, . . ., 2000, breaking ties toward earlier steps. Each estimate averages 16 decoded SDE draws, with matched Brownian streams across arms and checkpoints, for each of three evaluation seeds. The held branch shares the reference's step-1000 state. MSE averages coordinates, then intervals within each trajectory, then 20 validation trajectories. Neither test trajectories nor graph truth enters selection. Three Monte Carlo seeds are numerical repetitions on the same fitted realization, not independent training replications.

Selection is stable to these Monte Carlo repetitions but does not consistently improve graph ranking (Table 8). Descriptive paired trajectory-bootstrap 95% intervals for ΔMSE are $[-0.01822, 0.00305]$, $[-0.00962, 0.00776]$ and $[-0.01425, 0.01103]$, respectively. These conditional, post-selection intervals use 5000 resamples of trajectories, they are not selection-adjusted inference or uncertainty across independent fitted systems. All contain zero.

Pruning and rank distortion. Between steps 1500 and 2000, all saved validation permutations remain unchanged. Fixing each arm's step-1500 permutation therefore gives the same late graph metrics. The bounded reference loses one true off-diagonal raw input group in each of the intervals 1800–1900 and 1900–2000, alongside three false groups in each. True-versus-false score ties rise from 140 at step 1800 to 231 at step 2000. These endpoint observations do not identify the immediate optimizer or proximal cause. For no control, the raw support and tie count are unchanged from 1900 to 2000, yet off-diagonal AUROC falls from 0.7081 to 0.7042, with 11 strict pair reversals against true edges and four in their favor. Ranking changes therefore occur even without net support changes between saved checkpoints.

Holding λ_4 retains 83 raw groups at step 2000 versus 60 in the bounded reference, but worsens AUROC from 0.7815 to 0.7583 and directed SHD from 33 to 42. Thus preserving more groups does not repair graph recovery in this realization. We retain fixed-budget reporting and do not promote the weaker penalty or prediction-minimum stopping as a structural recovery remedy.

Fixed-budget selection versus graph-guided stopping. Step 1600 gives AUROC 0.7917 for the bounded reference and 0.8077 without control in seed 105, compared with final values 0.7815 and 0.7781. This checkpoint was nominated after inspecting known-graph metrics. The fixed step-1600 comparison on seed 102 below assesses repeatability within exposed development realizations, it is not an oracle checkpoint selected separately for every run. Both comparisons retain the original 2000-step schedule, because shortening that horizon would also change proximal annealing. These curves do not validate an early-stopping rule.

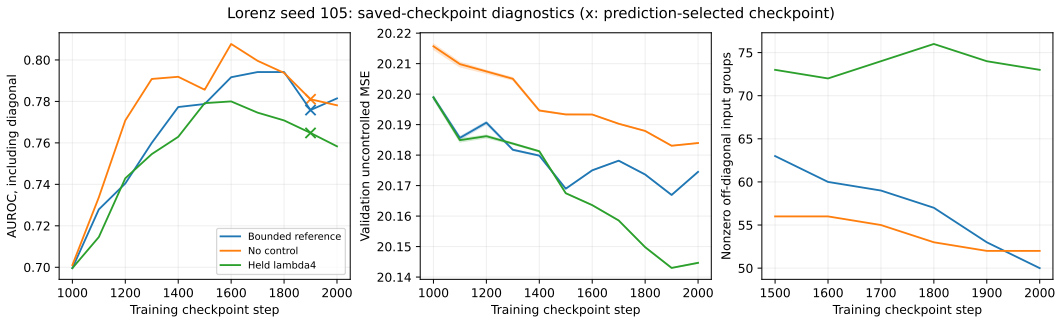

Figure 3: Corrected saved-checkpoint diagnostics for development seed 105. Crosses mark prediction-selected checkpoints. The MSE band spans the three Monte Carlo evaluation seeds, not a confidence interval across fitted systems. Raw support uses a 10^{-12} group-norm threshold, binary graph adjacency uses the separate existing column-normalized threshold.

Second development realization. The fixed step-1600 comparison repeats on seed 102. Bounded-control AUROC is 0.7121 at step 1600 versus 0.7046 at step 2000, without control it is 0.7335 versus 0.7027. All three 16-draw prediction evaluations instead select step 2000 in both arms. Together with seed 105, this supports reporting sensitivity to the fixed training budget, but does not validate prediction-guided early stopping. The candidate step was nominated from seed 105 and both realizations have development exposure. All multi-realization comparisons retain the original fixed training budgets.

C.5 Additional checkpoint evidence

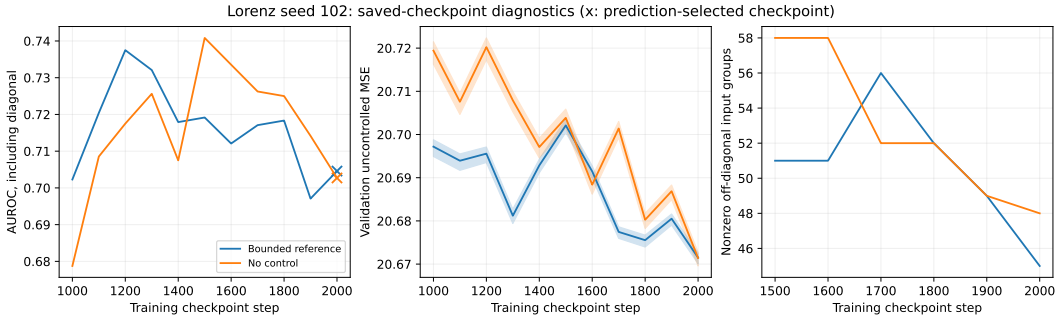

Figure 4: Second development realization, seed 102. The fixed step-1600 graph improvement repeats, while prediction selection chooses the final checkpoint. Bands span Monte Carlo seeds rather than independent fitted systems.

C.6 External comparison protocol and frozen Glycolysis search

Table 1 combines published SCOTCH D10 and Glycolysis values with local D50 adapter outcomes. The published values use different cohorts and training settings from ours. Local D50 runs stopped before producing graph estimates and are reported separately from the published numbers.

For a matched $D = 50$, $p = .3$ Lorenz execution check, SCOTCH used the same saved training splits as five completed PLACID fits. SCOTCH requires one time grid per batch, we grouped trajectories with identical physical times, with no interpolation or discarded observations. The upstream Lorenz source defaults were learning rate 0.003, graph sparsity weight 500, and solver step 1, with a ceiling of 5000 optimizer updates. The adapter used source-default residual connections off and one optimizer-step learning-rate warm-up. The SCOTCH paper instead describes residual connections and 100 epochs of warm-up, so this is not an exact Table 1 reproduction. Seed 101 showed escalating finite loss and was treated as a pilot. We then froze a rule that stops a run from epoch

5 onward if its median epoch loss reaches 10^6 times the epoch-0 median. Seeds 102, 104 and 105 crossed that rule at steps 168, 36 and 36, seed 103 developed a nonfinite gradient at step 159. No graph estimate was produced under that stopping policy. An initial $D = 10$, $p = .3$ check was stopped by the same loss rule at epoch 33. A fresh $D = 10$ seed-103 run omitted only that ratio stop while retaining nonfinite checks. Its median loss later recovered and it completed 5000 updates with AUROC 0.5246, AUPRC 0.4293 and directed SHD 42, versus paired PLACID 0.7529, 0.6772 and 30 at 2000 updates. This is a one-seed positive control, not a multi-seed D10 comparison. Its recovery shows that the three $D = 50$ ratio stops alone do not prove irreversible divergence, the separate nonfinite-gradient D50 run is a numerical failure. Stopped runs are reported as missing estimates, not as zero AUROC values.

Glycolysis already has dataset-specific architecture and regularization, we therefore performed a pre-specified finite search over $\beta \in \{0.1, 1, 10\}$ and drift penalty $\lambda_4 \in \{10^{-4}, 5 \cdot 10^{-4}\}$, keeping the cap, identity anchor, 1000-step main budget and diffusion schedule fixed. Selection uses mean final AUROC on three development realizations. The selected setting, incumbent and no-control comparator are then evaluated on five fresh simulator realizations without further selection. Those realizations share the same benchmark graph, they do not establish transfer to new graph families. The search selected $\beta = 1$ and $\lambda_4 = 5 \cdot 10^{-4}$. On the five fresh realizations, selected, incumbent and no-control AUROC are 0.6528 ± 0.0193, 0.5746 ± 0.0386 and 0.6251 ± 0.0519, respectively. The selected-minus-incumbent difference is 0.0783, with descriptive paired 95% interval $[0.0357, 0.1208]$ and five positive pairs. The selected-minus- no-control difference is 0.0278, interval $[-0.0555, 0.1110]$, and four of five pairs improve. Off-diagonal AUROC differs by 0.0577, interval $[-0.0338, 0.1492]$, AUPRC differs by 0.0319, interval $[-0.0792, 0.1431]$. These unadjusted, five-realization summaries support a selected-setting improvement over the incumbent and an uncertain control effect, rather than universal control necessity or superiority to an external baseline. All 15 fresh records passed graph recomputation and split-integrity checks, the three arms share identical split artifacts within each realization. Harder Lorenz runs cover $D = 10, p = .6$ and $D = 50, p \in \{.3, .6\}$, five realizations each. Following the original setting-specific configuration, $p = .6$ uses 2500 main steps, $p = .3$ retains 2000. All 15 harder fits completed. At $D = 50$, AUROC is 0.7481 ± 0.0135 for $p = .3$ and 0.6935 ± 0.0309 for $p = .6$, off-diagonal AUROC is 0.6657 ± 0.0192 and 0.5945 ± 0.0376, respectively. These results demonstrate successful fitting at 50 variables under the stated budgets, with weaker recovery in the more incomplete setting.

Table 9: Fixed-budget benchmark and tuning results. All displayed cohorts are complete. Development graph scores determine the selected candidate. Fresh evaluation follows a frozen selection.

Setting	n	AUROC	Off-diagonal AUROC
Lorenz D=10, p=0.6	5	0.7103 ± 0.0515	0.6138 ± 0.0684
Lorenz D=50, p=0.3	5	0.7481 ± 0.0135	0.6657 ± 0.0192
Dev. β=10.0, λ_4=0.0005	3	0.5468 ± 0.0699	0.5697 ± 0.0677
Dev. β=10.0, λ_4=0.0001	3	0.5479 ± 0.0700	0.5705 ± 0.0682
Dev. β=1.0, λ_4=0.0005	3	0.6282 ± 0.0285	0.6282 ± 0.0398
Dev. β=1.0, λ_4=0.0001	3	0.6260 ± 0.0259	0.6258 ± 0.0361
Dev. β=0.1, λ_4=0.0005	3	0.6014 ± 0.0190	0.5769 ± 0.0181
Dev. β=0.1, λ_4=0.0001	3	0.6031 ± 0.0174	0.5793 ± 0.0158
Glycolysis fresh selected	5	0.6528 ± 0.0193	0.6558 ± 0.0243
Glycolysis fresh incumbent	5	0.5746 ± 0.0386	0.6000 ± 0.0740
Glycolysis fresh no_control	5	0.6251 ± 0.0519	0.5981 ± 0.0509
Lorenz D=50, p=0.6	5	0.6935 ± 0.0309	0.5945 ± 0.0376

Table 10: Complete matched component panel. Mean AUROC, AUPRC, directed SHD and FDR across five realizations. Lorenz and Glycolysis use separate matched cohorts. All four conditions are reported.

Dataset	Condition	AUROC	AUPRC	SHD	FDR
Lorenz	Full	0.7540	0.7335	36.2	0.467
Lorenz	No control	0.7639	0.7449	37.2	0.477
Lorenz	No proximal	0.7395	0.7344	43.0	0.523
Lorenz	Neither	0.7553	0.7438	40.8	0.506
Glycolysis	Full	0.6528	0.6001	21.6	0.480
Glycolysis	No control	0.6251	0.5681	21.2	0.468
Glycolysis	No proximal	0.6542	0.6014	21.6	0.479
Glycolysis	Neither	0.6254	0.5675	21.6	0.475

C.7 Matched component ablations

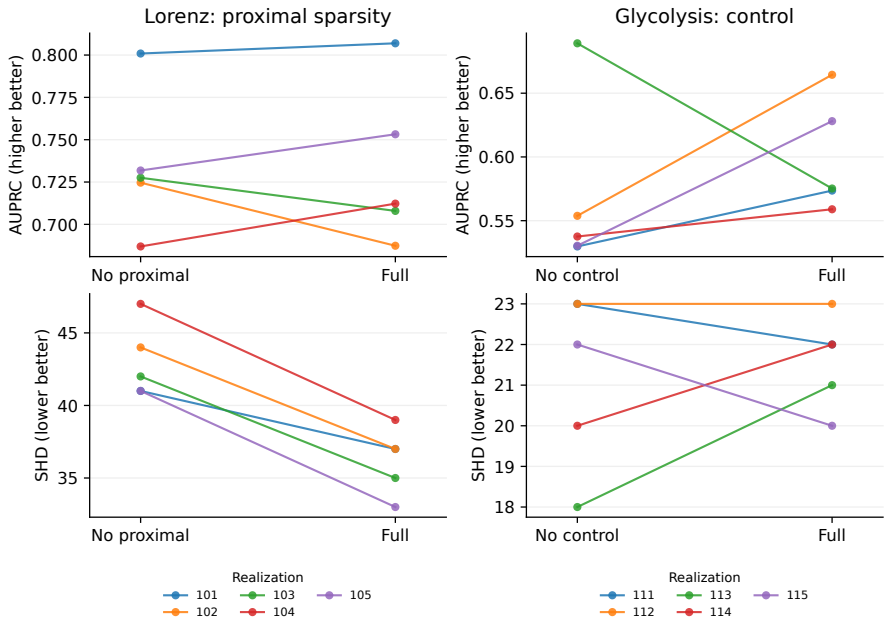

Figure 5: Paired component effects across five realizations. Each line uses identical data and remaining settings across its conditions. Lorenz shows the pruning effect of proximal sparsity, Glycolysis control effects vary across realizations.

We isolate proximal sparsity and control in a four-condition panel: full, no control, no proximal sparsity, and neither. Removing proximal sparsity disables encoder, decoder and drift proximal updates while retaining coupling and L_2 regularization. The Lorenz panel extends the existing matched development realizations, Glycolysis extends the frozen selected setting on realizations 111–115, now exposed for this follow-up. Full and no-control fits are reused, and the two additional conditions use identical saved splits, training budgets and remaining settings. The panel contains 40 verified records, including 20 reused references. On Lorenz, full versus no proximal sparsity has mean AUROC difference 0.0145 with descriptive paired interval $[-0.0151, 0.0442]$. Mean directed SHD improves from 43.0 to 36.2 and FDR from 0.5227 to 0.4672 at the retained threshold. The full model has 38–43 exact zero raw edge scores per realization, whereas the no-proximal arms have none, supporting a pruning effect. On Glycolysis, full and no proximal sparsity have mean AUROC 0.6528 and 0.6542, both with mean SHD 21.6, neither produces exact zero raw edge scores. The current proximal settings therefore provide little measurable benefit. With proximal updates, full versus no control has mean AUPRC 0.6001 versus 0.5681 and mean AUROC 0.6528 versus 0.6251,

but both paired intervals span zero. Without proximal updates, AUPRC is 0.6014 versus 0.5675. The controlled model's SHD is 21.6 versus 21.2 without control. These findings distinguish ranking, pruning and binary graph accuracy rather than establishing joint necessity of all components.

Paired effects clarify the component interaction. With control present, proximal updates reduce Lorenz SHD by 6.8 edges on average, descriptive interval $[4.76, 8.84]$, without control they reduce SHD by 3.6 edges, interval $[1.18, 6.02]$. The difference between these effects is 3.2 edges, interval $[-0.57, 6.97]$, so this small panel does not resolve a Lorenz interaction. For Glycolysis, the AUPRC difference between full and no control is 0.0319, interval $[-0.0792, 0.1431]$. The corresponding difference without proximal updates is 0.0339, interval $[-0.0781, 0.1460]$. Neither is a consistent five-realization gain.

At Glycolysis's fixed column-normalized threshold of 0.2, full predicts 33.4 edges on average versus 27.4 without control. The extra six edges contain 2.8 true positives and 3.2 false positives on average. Mean SHD rises from 21.2 to 21.6, a small difference with a paired interval spanning zero. Figure 6 shows the five off-diagonal precision-recall traces per arm and descriptive mean graph-size and SHD curves across thresholds. Only 0.2 was used for the reported binary metrics, the other thresholds use evaluation graph labels for diagnosis and were not selected as a new operating point.

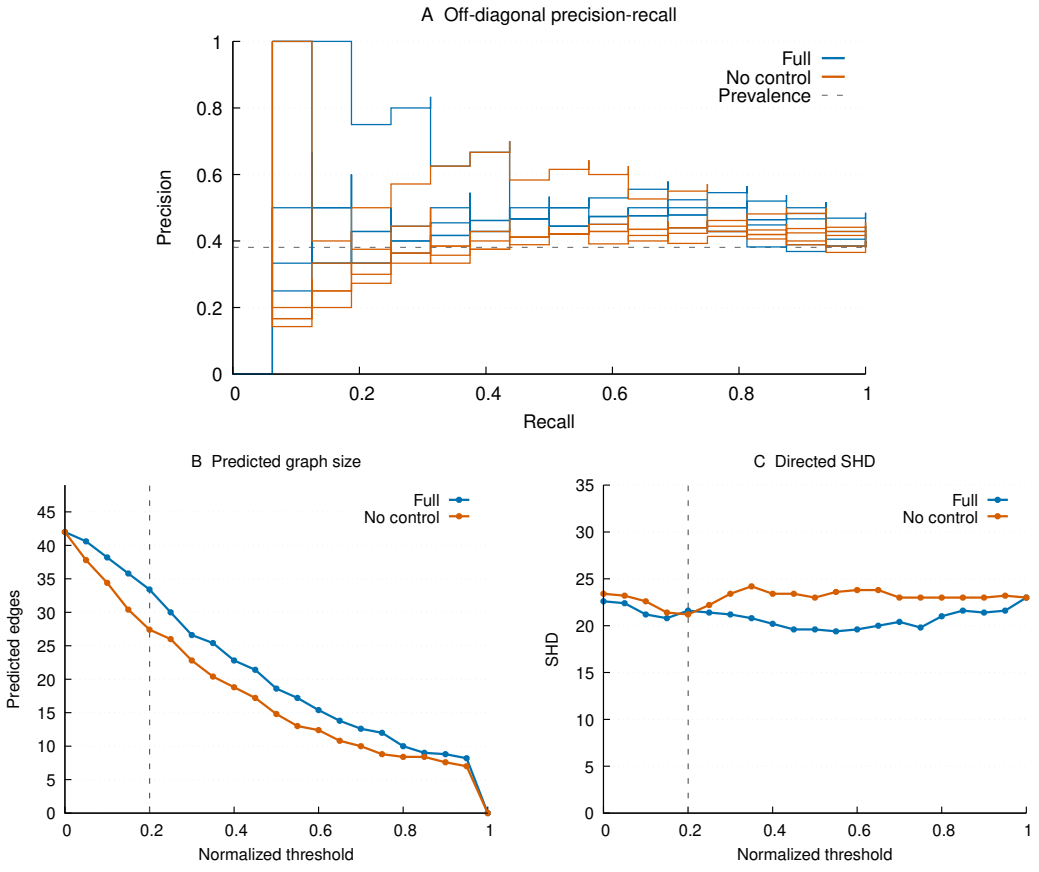

Figure 6: Glycolysis control ablation across five paired realizations. Blue is full and orange is no control. Precision-recall uses continuous off-diagonal scores, the gray line marks the positive-edge prevalence $16/42$. The other panels show means across realizations under the saved column-normalized threshold rule. The dashed vertical line marks the fixed reported threshold 0.2. Curves at other thresholds are descriptive, not tuned results.

Table 11: Matched component ablations. Lorenz uses the existing development panel, Glycolysis uses the frozen selected setting on realizations 111–115. No proximal sparsity removes encoder, decoder and drift proximal updates, retaining other regularizers. Each condition has five realizations.

Dataset	Component setting	n	AUROC	Off-diag. AUROC
lorenz96	Full	5	0.7540 ± 0.0400	0.6721 ± 0.0533
lorenz96	No control	5	0.7639 ± 0.0425	0.6852 ± 0.0566
lorenz96	No proximal sparsity	5	0.7395 ± 0.0372	0.6527 ± 0.0495
lorenz96	Neither	5	0.7553 ± 0.0222	0.6738 ± 0.0296
glycolysis	Full	5	0.6528 ± 0.0193	0.6558 ± 0.0243
glycolysis	No control	5	0.6251 ± 0.0519	0.5981 ± 0.0509
glycolysis	No proximal sparsity	5	0.6542 ± 0.0203	0.6577 ± 0.0260
glycolysis	Neither	5	0.6254 ± 0.0552	0.5990 ± 0.0567

C.8 Exploratory DREAM3 transfer pilot

We also ran one seed (42) on each of five DREAM3 size-100 in silico networks, using all 46 trajectories and 21 regular observation times per network. Inputs were standardized per variable. The two configurations were transferred from an archived DREAM3 setup and a frozen Lorenz setup, without a DREAM3 parameter search or repeated seeds. Table 12 reports source-row, off-diagonal AUROC after correcting the checked-in target-row graph orientation. The first Ecoli1 runs were rescored from saved models, with no retraining. The original TSV inputs used by the published SCOTCH runner were unavailable for byte-level comparison. Both transferred profiles recover little off-diagonal structure on this benchmark. Because there was no DREAM3-specific parameter search, weak transfer is unsurprising, that fact does not establish why the fits failed. This pilot is not a tuned benchmark or an independent biological validation: DREAM3 supplies simulated gene-network trajectories. The saved first-layer group-norm score remains the primary PLACID score. A post hoc path-sum score improves some networks, but an encoder-decoder-only diagnostic also improves, so neither is promoted as confirmed drift-graph recovery.

Table 12: Exploratory DREAM3 size-100 pilot. One seed per network, off-diagonal AUROC. Neither PLACID profile was tuned on DREAM3.

Network	Archived DREAM3 profile	Frozen Lorenz profile
Ecoli1	.420	.505
Ecoli2	.533	.505
Yeast1	.479	.501
Yeast2	.511	.504
Yeast3	.490	.499
Across-network mean	.487	.503