WHAT JOINT SCORE INFORMATION RETAINS FOR CAUSAL ORDERING

Anonymous authors
Paper under double-blind review

ABSTRACT

Score-based methods can identify leaves in nonlinear additive noise models, but recovering a full causal order requires repeating this decision after each removal. Can the expected joint score-Jacobian matrix alone support this recursion? We show that it cannot in general, even when the matrix is known exactly. The proof uses two three-variable nonlinear models with equal-variance Gaussian noise. Their joint matrices are identical and identify X_3 as a leaf in both models. After X_3 is removed, one model requires X_1 to precede X_2, while the other requires X_2 to precede X_1. Since the matrix cannot distinguish these orders, no rule using it alone can guarantee a valid order over the model class. This limitation also applies to Schur elimination, although it is exact in linear Gaussian models. An exact identity shows that the Schur update differs from the true marginal matrix by a scaled covariance of the removed leaf's mechanism gradient. This term vanishes when the gradient is constant. When nonzero, it need not change the selected leaf, so experiments with Score-Schur Topological Sort (SSTS) test when the uncorrected update still yields valid orders. Across five paired synthetic datasets per dimension, SSTS gives zero edge violations at 50 variables but 4.6 ± 2.4 at 100 variables, compared with 0.2 ± 0.4 for public SCORE/DAS ordering. Separate paired diagnostics show that lower matrix-estimation error can accompany more order violations. A guarantee over the stated nonlinear model class still requires information beyond this joint expectation or further model restrictions.

1 INTRODUCTION

In the chains $X_1 \to X_2 \to X_3$ and $X_2 \to X_1 \to X_3$, the variable X_3 has no children and is removed first by leaf-based ordering. The next decision must distinguish $X_1 \to X_2$ from $X_2 \to X_1$. Identifying the first leaf does not settle this decision. The information used to select it must also support ordering the variables that remain.

Score-based methods make these removal decisions from derivatives of the observational score (Rolland et al., 2022; Xu et al., 2024; Sanchez et al., 2023). Each removal changes the distribution used for the next decision. SCORE re-estimates score derivatives from reduced data; DiffAN's masking variant avoids retraining but still queries its learned representation after removals. Could the expected joint score-Jacobian matrix replace these later queries? If so, ordering could proceed by matrix operations without further access to sample-dependent scores.

This reduction is exact in linear Gaussian models with equal noise variances. The expected negative score Jacobian is then the precision matrix. Selecting its smallest diagonal entry and applying Schur elimination recovers a valid order recursively (Ghoshal & Honorio, 2018). Under the same noise assumptions, the nonlinear expected-diagonal leaf criterion is a special case of CaPS (Xu et al., 2024). This criterion identifies a current leaf, but it does not establish that the original nonlinear matrix remains sufficient after removal.

For the two chains above, we construct nonlinear models with the same expected score-Jacobian matrix (Figure 1). It identifies X_3 in both, but cannot distinguish the remaining edge $X_1 \to X_2$ from $X_2 \to X_1$. Their observational distributions differ; the shared matrix has lost the distinction. Theorem 3.1 shows that no rule given only this matrix can guarantee a valid order for both models. This excludes any matrix-only update, not just Schur elimination.

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

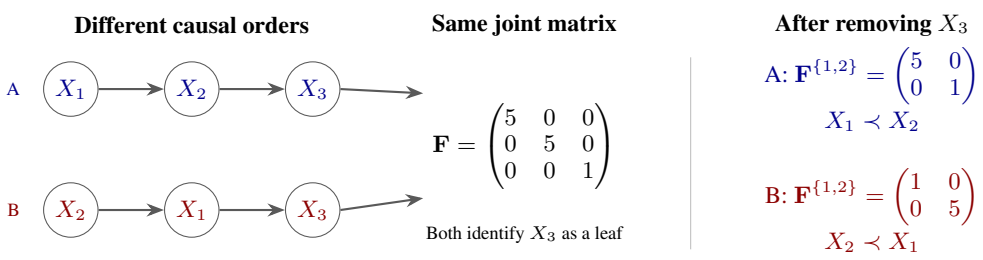


Figure 1: **The same first-leaf information, different remaining orders.** Two smooth nonlinear chains with unit-variance Gaussian noise have the same expected score-Jacobian matrix. It identifies their common leaf X_3, but their remaining orders are opposite. The right column shows the true marginal matrices, not Schur updates of the shared matrix. Matrix coordinates follow the fixed labels (X_1, X_2, X_3) and (X_1, X_2). Theorem 3.1 and Appendix C.5 give the construction.

Schur elimination can fail to reproduce the nonlinear marginal matrix. A leaf's mechanism gradient describes its response to its parents and can vary across observations. The joint entries linking the leaf to its parents encode the mean gradient, but the marginalization correction needs its second moment. Schur elimination uses the outer product of the mean instead. The discrepancy is exactly a scaled covariance of the gradient (Proposition 3.2). It vanishes when the gradient is constant, as in the linear case. The identity explains the error of this particular update; the counterexample establishes the limit on all rules restricted to the same matrix.

The covariance discrepancy need not change the selected leaf, so the uncorrected update may still recover an order on individual nonlinear models. We examine this possibility with Score-Schur Topological Sort (SSTS), which uses one estimated matrix without the covariance correction. It gives zero edge violations on the tested 50-variable synthetic graphs, while repeated-query SCORE/DAS ordering has lower error at 100 variables. Separate paired diagnostics show that a more accurate matrix estimate need not give a better order. These experiments measure when the approximation works. The population counterexample explains why estimation accuracy alone cannot provide a guarantee over the stated nonlinear model class: such a guarantee requires additional information or further structural restrictions.

2 Model, Information, and Ordering Task

The two chains require opposite positions for X_1 and X_2 after X_3 is removed. We now define the model class, the information retained in the joint matrix, and what it means to recover such an order. An order puts every direct cause before its effect; restricting parent selection to preceding variables then prevents cycles without yet determining the edges.

Model and observations. For d observed variables, write $[d] = \{1, \dots, d\}$ and let $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ be a directed acyclic graph (DAG) with node set $\mathcal{V} = [d]$. Node i has parents $\mathrm{pa}(i) = \{j : (j, i) \in \mathcal{E}\}$ and children $\mathrm{ch}(i) = \{j : (i, j) \in \mathcal{E}\}$; a root has no parents. An additive noise model (ANM) $M = (\mathcal{G}, (f_i)_{i=1}^d, \sigma^2)$ specifies $\mathbf{X} = (X_1, \dots, X_d)^\top$ by

$$X_i = f_i(\mathbf{X}_{\mathrm{pa}(i)}) + \epsilon_i, \qquad (\epsilon_i)_{i=1}^d \overset{\text{iid}}{\sim} \mathcal{N}(0, \sigma^2), \quad \sigma^2 > 0. \tag{1}$$

Here $f_i : \mathbb{R}^{|\mathrm{pa}(i)|} \to \mathbb{R}$ is the structural mechanism of node i, mapping parent values to the noise-free component of X_i. Write $p = p_M$ and $\mathcal{G}_M = \mathcal{G}$ for the model's density and graph, and $\partial_j = \partial/\partial x_j$. We define the class $\mathfrak{M}_d$ by the mechanism conditions

$$f_i \in C^2(\mathbb{R}^{|\mathrm{pa}(i)|}), \qquad 0 < \mathbb{E}_p\big[(\partial_j f_i(\mathbf{X}_{\mathrm{pa}(i)}))^2\big] < \infty \quad \forall (j, i) \in \mathcal{E},$$

and the integrability and boundary conditions in Assumption A.3. We call f_i linear when it is affine, and nonlinear otherwise. Observations are $\mathcal{D} = \{\mathbf{x}^{(n)}\}_{n=1}^N$, with $\mathbf{x}^{(n)} \overset{\text{iid}}{\sim} p$. Every coordinate is observed; there are no latent nodes (Hoyer et al., 2008; Peters et al., 2014).

Marginals and notation. Discarding data columns leaves observations from a marginal distribution. For $\mathcal{S} \subseteq \mathcal{V}$, $\mathbf{X}_\mathcal{S}$ is the subvector in increasing label order, and

$$p_\mathcal{S}(\mathbf{x}_\mathcal{S}) = \int_{\mathbb{R}^{|\mathcal{V}\setminus\mathcal{S}|}} p(\mathbf{x})\, d\mathbf{x}_{\mathcal{V}\setminus\mathcal{S}}, \qquad \mathbb{E}_{p_\mathcal{S}}[h] = \int h(\mathbf{x}_\mathcal{S}) p_\mathcal{S}(\mathbf{x}_\mathcal{S})\, d\mathbf{x}_\mathcal{S}.$$

The full density is $p = p_\mathcal{V}$; $\mathcal{G}[\mathcal{S}]$ retains the nodes in $\mathcal{S}$ and the edges between them. A set $\mathcal{S}$ is ancestor-closed when $\mathrm{pa}(i) \subseteq \mathcal{S}$ for every $i \in \mathcal{S}$; let $\mathfrak{A}(\mathcal{G})$ collect these nonempty sets. Leaf removal preserves this property: no retained equation uses that leaf as a parent. For $\mathcal{S} \in \mathfrak{A}(\mathcal{G})$, the unchanged equations generate $p_\mathcal{S}$; its leaves are $\mathcal{L}(\mathcal{S}) = \{i \in \mathcal{S} : \mathrm{ch}(i) \cap \mathcal{S} = \varnothing\}$. Bold symbols denote vectors or matrices; $\mathbf{I}_k$ is the $k \times k$ identity. Matrix subscripts select labeled rows and columns, and $\mathrm{sym}(\mathbf{A}) = (\mathbf{A} + \mathbf{A}^\top)/2$.

The retained statistic. The score measures how log density changes with the observed coordinates. Its negative Jacobian describes local curvature; the score-Jacobian information matrix (SJIM) averages that curvature. For each marginal, define

$$\mathbf{s}_\mathcal{S} = \nabla_{\mathbf{x}_\mathcal{S}} \log p_\mathcal{S}, \qquad \mathbf{H}^\mathcal{S} = -\nabla^2_{\mathbf{x}_\mathcal{S}} \log p_\mathcal{S}, \qquad \mathbf{F}^\mathcal{S} = \mathbb{E}_{p_\mathcal{S}}[\mathbf{H}^\mathcal{S}(\mathbf{X}_\mathcal{S})]. \tag{2}$$

For the joint density we omit the superscript: $\mathbf{s} = \mathbf{s}_\mathcal{V}$, $\mathbf{H} = \mathbf{H}^\mathcal{V}$, and $\mathbf{F} = \mathbf{F}^\mathcal{V}$. Under Assumption A.3, integration by parts gives the identity $\mathbf{F}^\mathcal{S} = \mathbb{E}_{p_\mathcal{S}}[\mathbf{s}_\mathcal{S}\mathbf{s}_\mathcal{S}^\top] \succeq 0$ (Hyvärinen, 2005). Marginalizing before computing curvature differs from deleting rows and columns afterward: $\mathbf{F}^\mathcal{R}$ need not equal the submatrix $\mathbf{F}^\mathcal{S}_{\mathcal{R}\mathcal{R}}$, where $\mathcal{R} \subseteq \mathcal{S}$.

Ordering task and information restriction. Let $\mathfrak{S}_d$ be the bijections $\pi : \mathcal{V} \to [d]$ assigning node positions; write $X_i \prec X_j$ when $\pi(i) < \pi(j)$. The valid orders are

$$\Pi(\mathcal{G}) = \{\pi \in \mathfrak{S}_d : \pi(j) < \pi(i) \ \ \forall (j,i) \in \mathcal{E}\}. \tag{3}$$

If $l_1, \ldots, l_d$ are successive true leaf removals, the list $\omega = (l_d, \ldots, l_1)$ defines a valid order by $\pi(\omega_k) = k$. A matrix-only rule is a function $T : \mathbb{S}^d_+ \to \mathfrak{S}_d$, with $\mathbb{S}^d_+$ the symmetric positive semidefinite matrices indexed by $\mathcal{V}$. Its only model-dependent input is $\mathbf{F}(M)$. Universal recovery would require

$$T(\mathbf{F}(M)) \in \Pi(\mathcal{G}_M) \qquad \forall M \in \mathfrak{M}_d. \tag{4}$$

Thus T has no separate access to $\mathcal{D}$, $\mathbf{s}(\mathbf{x})$, or marginal scores. The population question concerns exact $\mathbf{F}$; the fitted algorithm will first estimate it from $\mathcal{D}$.

In this model class, each expected diagonal is the common noise contribution σ^{-2} plus positive contributions from its children. Leaves have no such extra terms, so they attain the minimum (Theorem A.1; Xu et al., 2024). In a linear Gaussian ANM, $\mathbf{F} = \mathbf{\Sigma}^{-1}$, where $\mathbf{\Sigma} = \mathrm{Cov}_p(\mathbf{X})$. This precision matrix is the inverse covariance; its Schur update gives the next marginal matrix (Ghoshal & Honorio, 2018; Lauritzen, 1996). We ask whether the same joint statistic suffices when mechanisms are nonlinear.

3 A First Leaf Is Not a Complete Order

We separate two questions. Does the joint matrix contain enough information for any ordering rule to succeed? If not, what does the particular Schur update omit? The first result answers the information question; the second identifies the update error.

3.1 Identical Matrices, Incompatible Orders

Two models with the same input matrix force a matrix-only rule to return the same answer. Such a rule cannot be correct on both models if their valid orders are disjoint. The following construction makes this obstruction explicit within $\mathfrak{M}_3$.

Theorem 3.1 (Insufficiency of the joint SJIM)**.** *There exist $M_A, M_B \in \mathfrak{M}_3$, both with $\sigma^2 = 1$, such that*

$$\mathcal{G}_A : \ 1 \to 2 \to 3, \qquad \mathcal{G}_B : \ 2 \to 1 \to 3,$$
$$\mathbf{F}(M_A) = \mathbf{F}(M_B) = \mathrm{diag}(5,5,1), \qquad \Pi(\mathcal{G}_A) \cap \Pi(\mathcal{G}_B) = \varnothing.$$

Consequently,

$$\forall T : \mathbb{S}^3_+ \to \mathfrak{S}_3, \quad \exists k \in \{A, B\} : \ T(\mathbf{F}(M_k)) \notin \Pi(\mathcal{G}_k).$$

A limit of fixed-matrix ordering. The diagonal $(5, 5, 1)$ identifies X_3 in both models. The disjoint order sets require opposite positions for X_1 and X_2, yet identical inputs force any rule T to return the same order. The final statement therefore rules out universal recovery even without estimation error. CaPS supplies a current-leaf criterion (Xu et al., 2024); Theorem 3.1 shows why this does not make the original matrix sufficient for later decisions. It does not rule out recovery from observations, sample-dependent scores, or a more restricted model class.

Proof idea. Use quadratic mechanisms whose derivatives have mean zero and second moment 4, then exchange labels 1 and 2. The joint matrix stays $\mathrm{diag}(5, 5, 1)$, while the required order reverses. Appendix C.5 specifies the mechanisms and verifies the moments.

3.2 What Schur Elimination Omits

Fix an ancestor-closed active set $\mathcal{S} \in \mathfrak{A}(\mathcal{G})$, a true leaf $l \in \mathcal{L}(\mathcal{S})$, and $\mathcal{R} = \mathcal{S} \setminus \{l\} \neq \varnothing$. These symbols refer to the current marginal, not necessarily the joint distribution. Extend f_l to $\mathbb{R}^{|\mathcal{R}|}$ by ignoring non-parent coordinates, and set

$$\mathbf{g}_l(\mathbf{x}_\mathcal{R}) = \nabla_{\mathbf{x}_\mathcal{R}} f_l(\mathbf{x}_{\mathrm{pa}(l)}), \qquad \mathrm{Cov}_q(\mathbf{Y}) = \mathbb{E}_q[\mathbf{Y}\mathbf{Y}^\top] - \mathbb{E}_q[\mathbf{Y}]\mathbb{E}_q[\mathbf{Y}]^\top.$$

For a matrix $\mathbf{A}$ indexed by $\mathcal{S}$ with $A_{ll} \neq 0$, define

$$\mathrm{Schur}_l(\mathbf{A}) = \mathbf{A}_{\mathcal{R}\mathcal{R}} - \mathbf{A}_{\mathcal{R}l} A_{ll}^{-1} \mathbf{A}_{l\mathcal{R}}. \tag{5}$$

The next identity compares this update with $\mathbf{F}^\mathcal{R}$. It needs Gaussian noise and regularity, but not edge non-degeneracy: the identity still holds if a listed parent has zero derivative.

Proposition 3.2 (Nonlinear marginalization gap)**.** *Under Assumptions A.1 and A.3, for every $\mathcal{S}, l, \mathcal{R}$ as above,*

$$\mathrm{Schur}_l(\mathbf{F}^\mathcal{S}) - \mathbf{F}^\mathcal{R} = \mathbf{C}_l^\mathcal{S} := \sigma^{-2}\mathrm{Cov}_{p_\mathcal{S}}(\mathbf{g}_l(\mathbf{X}_\mathcal{R})) \succeq 0. \tag{6}$$

Moreover,

$$\forall i, j \in \mathcal{R}, \quad (i \notin \mathrm{pa}(l) \text{ or } j \notin \mathrm{pa}(l)) \implies (\mathbf{C}_l^\mathcal{S})_{ij} = 0,$$
$$\mathbf{g}_l(\mathbf{X}_\mathcal{R}) = \mathbf{c} \ \ p_\mathcal{S}\text{-a.s. for some constant } \mathbf{c} \implies \mathbf{C}_l^\mathcal{S} = \mathbf{0}.$$

What the covariance term changes. Proposition 3.2 gives an exact population identity, not an estimation bound. Its positive-semidefinite gap means that the update cannot underestimate marginal curvature along any direction. The support condition confines this one-step discrepancy to the removed leaf's parents; the constant-gradient condition recovers linear exactness. Each entry of $\mathbf{g}_l$ is a local response, so the covariance measures how those responses vary across observations. Linear precision elimination (Ghoshal & Honorio, 2018) has no such correction. In nonlinear models, correcting the update requires this covariance, which better estimation of $\mathbf{F}^\mathcal{S}$ alone cannot supply.

Proof idea. Factor $p_\mathcal{S}$ into $p_\mathcal{R}$ and the Gaussian conditional of the leaf. Differentiating and averaging gives

$$\mathbf{F}^\mathcal{S}_{\mathcal{R}\mathcal{R}} = \mathbf{F}^\mathcal{R} + \sigma^{-2}\mathbb{E}[\mathbf{g}_l\mathbf{g}_l^\top], \qquad \mathbf{F}^\mathcal{S}_{\mathcal{R}l} = -\sigma^{-2}\mathbb{E}[\mathbf{g}_l], \qquad F^\mathcal{S}_{ll} = \sigma^{-2},$$

with expectations under $p_\mathcal{S}$. The residual-weighted Hessian of f_l has zero expectation because ϵ_l is independent of $\mathbf{X}_\mathcal{R}$. Substitution in Eq. 5 leaves the covariance. Appendix C.4 gives the full derivation.

Why retaining sample Hessians does not give an exact recursion. Could one instead update each Hessian before averaging? Under Assumptions A.1 and A.3, differentiating the current leaf conditional gives the pointwise identity derived in Appendix C.3, Eq. 15:

$$\mathrm{Schur}_l(\mathbf{H}^\mathcal{S}(\mathbf{X}_\mathcal{S})) = \mathbf{H}^\mathcal{R}(\mathbf{X}_\mathcal{R}) - \epsilon_l\sigma^{-2}\nabla^2_{\mathbf{x}_\mathcal{R}} f_l(\mathbf{X}_{\mathrm{pa}(l)}).$$

The residual has zero expectation, so one averaged update recovers $\mathbf{F}^\mathcal{R}$ (Lemma A.3). This is not a pointwise equality with the marginal Hessian. A later inverse can turn the residual into bias or make the expectation undefined, even with true Hessians and correct removals (Appendix G.2). Reusing the lemma requires the current marginal Hessians.

Unregularized block and scalar updates agree for fixed eliminated coordinates (Appendix G.3); batching cannot restore the covariance term. SSTS retains the uncorrected update, so we next ask when its leaf decisions remain valid.

4 SSTS: ORDERING FROM ONE ESTIMATED MATRIX

Given $\mathcal{D}$, SSTS fits a joint score model and extracts a candidate order from its averaged Jacobian. The unknown $\mathcal{G}$, mechanisms (f_i), and leaf sets are not inputs. Selection and elimination follow the linear Gaussian procedure. Block-SSTS may select several candidates per update; scalar SSTS removes one minimizer at a time.

4.1 FITTING AND AGGREGATION

We fit a multilayer perceptron (MLP) $\mathbf{s}_\theta : \mathbb{R}^d \to \mathbb{R}^d$, with parameters θ, by denoising score matching (DSM) with Gaussian corruption standard deviation $\tau > 0$ (Vincent, 2011). Input-column Group Lasso with weight $\lambda_{\text{sparse}} \geq 0$ regularizes the fit (Yuan & Lin, 2006) (Appendix B.1). We define the empirical matrix used for extraction by

$$\widehat{\mathbf{J}} = -\frac{1}{N}\sum_{n=1}^{N} \nabla_{\mathbf{x}} \mathbf{s}_\theta(\mathbf{x}^{(n)}), \qquad \hat{\mathbf{F}} = \operatorname{sym}(\widehat{\mathbf{J}}). \tag{7}$$

An ideal fixed-scale DSM fit learns the score of $p_\tau = p * \mathcal{N}(0, \tau^2 \mathbf{I}_d)$, where $*$ denotes convolution, not the score of p. Averaging at clean observations can introduce smoothing bias alongside fitting and sampling errors (Appendix G.4). Input-weight regularization neither enforces a sparse Jacobian nor guarantees that $\hat{\mathbf{F}}$ is positive definite.

4.2 SELECTING AND REMOVING CANDIDATES

Algorithm 1 maintains a symmetric matrix $\mathbf{M}$ on active labels $\mathcal{S}$. It groups diagonal entries within tolerance γ of the minimum and uses a ridge-regularized update. Candidates need not be leaves: increasing γ may include a parent. The tolerance is not a confidence level. If all nodes qualify, only one minimizer is kept. Node labels break ties.

Algorithm 1 Block Score-Schur Topological Sort (Block-SSTS)

Require: Symmetric estimate $\hat{\mathbf{F}}$, ridge $\lambda_{\text{ridge}} > 0$, tolerance $\gamma \in [0, 1)$.

$\mathbf{M} \leftarrow \hat{\mathbf{F}}$; $\mathcal{S} \leftarrow \mathcal{V}$; order list $\omega \leftarrow [\,]$.
while $\mathcal{S} \neq \varnothing$ **do**
  Stop with failure if $\mathbf{M}$ has a non-finite entry.
  $v \leftarrow \min_{i \in \mathcal{S}} M_{ii}$; $\mathcal{B} \leftarrow \{i \in \mathcal{S} : M_{ii} \leq v + \gamma|v|\}$.
  If $|\mathcal{B}| = |\mathcal{S}| > 1$, keep the smallest-label diagonal minimizer.
  Prepend $\mathcal{B}$ to ω, sorted by (M_{ii}, i); set $\mathcal{R} \leftarrow \mathcal{S} \setminus \mathcal{B}$.
  if $\mathcal{R} \neq \varnothing$ **then**
    Solve $(\mathbf{M}_{\mathcal{B}\mathcal{B}} + \lambda_{\text{ridge}} \mathbf{I}_{|\mathcal{B}|})\mathbf{Q} = \mathbf{M}_{\mathcal{B}\mathcal{R}}$; stop on failure.
    $\mathbf{M} \leftarrow \operatorname{sym}(\mathbf{M}_{\mathcal{R}\mathcal{R}} - \mathbf{M}_{\mathcal{R}\mathcal{B}}\mathbf{Q})$.
  end if
  $\mathcal{S} \leftarrow \mathcal{R}$.
end while
return Candidate order $\hat{\pi} \in \mathfrak{S}_d$, defined by $\hat{\pi}(\omega_k) = k$.

A sufficient condition for a correct removal. For $M \in \mathfrak{M}_d$, an active set $\mathcal{S} \in \mathfrak{A}(\mathcal{G})$, and a finite candidate matrix $\mathbf{M}$ indexed by $\mathcal{S}$, define the maximum diagonal error and the leaf separation by

$$e_{\text{diag}}(\mathbf{M}, \mathcal{S}) = \max_{i \in \mathcal{S}} |M_{ii} - F^{\mathcal{S}}_{ii}|, \qquad \delta_{\mathcal{S}} = \min_{i \in \mathcal{S} \setminus \mathcal{L}(\mathcal{S})} F^{\mathcal{S}}_{ii} - \sigma^{-2}, \tag{8}$$

with $\delta_{\mathcal{S}} = +\infty$ if all active nodes are leaves. These are definitions; the recovery condition is the implication proved in Proposition A.4: if $e_{\text{diag}} < \delta_{\mathcal{S}}/2$, every diagonal minimizer is a true leaf. A leaf diagonal can rise by at most e_{diag}, and a non-leaf diagonal can fall by the same amount. Errors below half the gap cannot reverse their order (Appendix C.6). A biased matrix can therefore support a correct removal without approximating every entry well.

This guarantee concerns one minimum-diagonal decision. Later Schur updates bring off-diagonal errors onto the diagonal. Proposition A.5 bounds this propagation and the new covariance discrepancy; Corollary A.6 gives a sufficient condition for a complete scalar, unregularized run. Both use unknown population marginals and do not certify the learned block procedure. A fixed ridge need not make its solves well-conditioned. We measure matrix error and order violations separately to test how estimation affects these decisions.

4.3 Edge Recovery and Cost

After ordering, candidate parents of node i are $P_i = \{j : \hat{\pi}(j) < \hat{\pi}(i)\}$. We follow the order-based pruning principle of causal additive models (CAM) (Bühlmann et al., 2014), using common linear Lasso instead of nonlinear pruning (Appendix B.2). This is a comparison protocol, not a new graph estimator. The returned graph is acyclic by construction; wrong orders or inadequate regression can still give wrong edges.

Jacobian aggregation needs an $\mathcal{O}(d^2)$ accumulator in addition to data, model, and autodifferentiation storage; a micro-batch of size b can materialize bd^2 entries. After aggregation, dense Schur elimination costs $\mathcal{O}(d^3)$ time and $\mathcal{O}(d^2)$ storage. Total runtime includes score fitting and pruning. Avoiding later score queries alone does not guarantee an end-to-end speedup.

5 Experiments

Theorem 3.1 rules out universal recovery, not success on individual models. We compare orders under common pruning, test fixed-matrix extraction against repeated queries under a matched schedule, and ask whether matrix error predicts ordering error. Higher-dimensional and Sachs checks test the scope of these findings.

5.1 Experimental Settings

Common-pruning benchmark. To compare orders without changing the edge-selection rule, we use common pruning in one nonlinear family. Sparse Erdős–Rényi (ER) DAGs have expected degree 2 and $d \in \{20, 50, 100\}$; each dataset has $N = 5000$ observations. Each weight w_{ji} on $j \to i$ has an independent sign and magnitude uniform on $[0.5, 1.5]$. We generate $X_i = \tanh(\sum_{j \in \mathrm{pa}(i)} w_{ji} X_j) + \epsilon_i$, with $\epsilon_i \sim \mathcal{N}(0, 0.1^2)$. We center columns and divide by one global standard deviation. Seeds 42–46 generate five structural causal models (SCMs), their data, and stochastic initializations. Methods share data within each seed.

Baselines are public CaPS ordering (Xu et al., 2024), dodiscover SCORE/DAS (Rolland et al., 2022; Montagna et al., 2023), and DiffAN-masking (Sanchez et al., 2023). All use common Lasso pruning (Appendix B.2), excluding CaPS's native parent-scoring and pruning. Saved native DAS graphs provide a pruning comparison. Public estimators retain their own capacities and training budgets (Appendix E).

Matched-schedule control. We compare fixed-matrix extraction with repeated masked queries under a common training schedule, using the same synthetic family and ten paired seeds 42–51. The control is ours, not DAS. Both models use two GELU hidden layers of width $\min(1024, \max(128, \lfloor 1.5d \rfloor))$, 200 epochs, batches of 1000, AdamW with initial rate 0.002 and weight decay 10^{-5}, a cosine schedule, and gradient clipping at 1. Both query all 5000 observations during ordering. SSTS uses $\tau = 0.2$, $\gamma = 0.01$, $\lambda_{\mathrm{ridge}} = 10^{-4}$, and $\lambda_{\mathrm{sparse}} = 0$ at $d = 20$ or $2 \times 10^{-4}\sqrt{d}$ at $d \geq 50$, also in the common-pruning benchmark. The control adds mask inputs and uses a different loss without the input penalty; parameter counts and objectives differ.

Paired-matrix diagnostic. This diagnostic asks whether a more accurate matrix gives a better order. We use five ER SCM/data seeds 42–46 at $d = 100$, with 5000 training observations and 2000 independent evaluation observations per seed. Here $f_i(\mathbf{x}_{\mathrm{pa}(i)}) = \sum_{j \in \mathrm{pa}(i)} w_{ji} \tanh(x_j)$, rather than $\tanh$ of the weighted sum. Evaluation data use the training means and global scale. At each $\tau \in \{0.05, 0.10, 0.20\}$, we use the same 200-epoch schedule and compute the averaged negative Jacobian and score outer product on the same evaluation data. The empirical reference $\mathbf{F}_{\mathrm{ref}}$ uses known mechanism-gradient moments on those samples, only for diagnosis (Appendix M).

Scope checks. At $d = 1000$, known-mechanism matrices vary SCM/data seeds 42–46; learned matrices fix SCM/data seed 42 and vary training seeds 101–105. We use the GraN-DAG release of Sachs data (Sachs et al., 2005; Lachapelle et al., 2020): 853 fixed observations, 11 variables, and a 17-edge reference graph, with column-wise standardization. Stochastic methods use seeds 42–46; deterministic SCORE, DAS, and TOPIC run once. Sachs comparisons also include public TOPIC, NOTEARS-MLP, and DAGMA-MLP (Xu et al., 2025; Zheng et al., 2020; Bello et al., 2022). Their pruning rules differ, so graph metrics compare complete pipelines. Appendix E.5 specifies the supplementary linear, ablation, mechanism-shift, and scaling settings, with their sample sizes and dispersion conventions.

Evaluation and runtime. Order error means edge violations,

$$\mathrm{EV}(\hat{\pi}, \mathcal{G}) = \sum_{(j,i)\in\mathcal{E}} \mathbf{1}\{\hat{\pi}(j) > \hat{\pi}(i)\},$$

where $\mathbf{1}$ is the indicator. This is Rolland et al. (2022)'s topological order divergence, zero for every valid order. Structural Hamming distance (SHD) counts edge additions, deletions, and reversals, each once. True positive rate (TPR) is the fraction of true directed edges recovered. Matrix error is $\|\hat{\mathbf{F}} - \mathbf{F}_{\mathrm{ref}}\|_F / \|\mathbf{F}_{\mathrm{ref}}\|_F$, using the Frobenius norm. We report mean $\pm$ sample standard deviation across paired seeds. GPU routines use an RTX 5090; dodiscover kernel estimators use CPU NumPy. SSTS ordering time T_{order} includes aggregation and Schur updates; T_{total} also includes training, pruning, and trial setup. The shared server was not verified idle (Appendices E and L).

5.2 Ordering in the Common-Pruning and Matched-Schedule Settings

Table 1: **Common-pruning benchmark.** Five paired SCM/data seeds, 42–46. All SHDs use the same order-constrained Lasso. SCORE and DAS return identical orders. CaPS denotes its public ordering prefix. Full results, including $d = 20$, are in Table 7.

Method	$d = 50$: EV	SHD	$d = 100$: EV	SHD
SSTS	0.0 ± 0.0	0.2 ± 0.4	4.6 ± 2.4	12.6 ± 8.4
CaPS ordering	4.2 ± 1.9	8.8 ± 8.6	3.8 ± 1.3	6.6 ± 3.2
SCORE / DAS	0.2 ± 0.4	0.8 ± 1.8	0.2 ± 0.4	0.8 ± 1.3
DiffAN-masking	7.2 ± 3.6	26.4 ± 21.8	13.8 ± 4.0	52.6 ± 23.4

In the common-pruning benchmark, SSTS and SCORE/DAS give zero EV and SHD at $d = 20$. SSTS gives zero EV on all five $d = 50$ graphs and SHD 0.2 ± 0.4: the uncorrected update can recover valid orders despite the population information limit. At $d = 100$, SCORE/DAS have lower EV than SSTS; CaPS also has lower mean EV and SHD (Table 1). Pruning changes graph accuracy even with the order fixed: DAS gives SHD 16.2 ± 5.1 with native pruning and 0.8 ± 1.3 with common Lasso at $d = 100$ (Table 8). This is why we separate order and graph metrics.

In the matched-schedule setting, SSTS has lower EV in eight pairs and ties in two at $d = 50$. At $d = 100$, it has higher mean EV but lower mean SHD (Table 2). Its lower median ordering time does not yield lower median total time; at $d = 20, 50$, ordering is slower. Removing later queries gives no uniform advantage in this control. Long timing tails reverse some mean–median rankings (Tables 9 and 11).

Table 2: **Matched-schedule control at $d = 100$.** Ten paired seeds, 42–51. EV and SHD are mean $\pm$ sample standard deviation; times are medians in seconds.

Method	EV	SHD	T_{order}	T_{total}
SSTS	3.8 ± 2.1	8.9 ± 7.0	0.177	9.30
Masked-score control	3.1 ± 2.1	11.2 ± 9.2	0.453	8.79

5.3 Results of the Paired-Matrix Diagnostic

The known-mechanism reference gives zero EV for all five SCMs. We test whether learned matrices closer to this successful reference in Frobenius norm also give better orders. Figure 2 shows that the criteria can disagree.

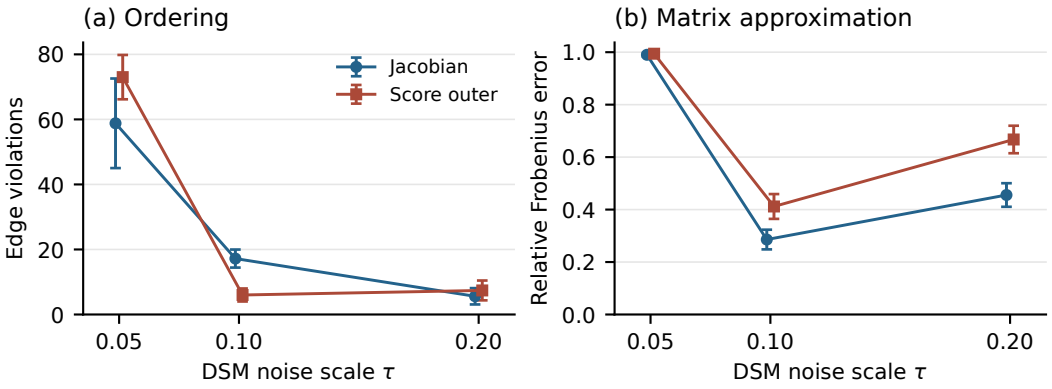


Figure 2: **Paired-matrix diagnostic at** $d = 100$**.** Means and sample standard deviations over five SCM/data seeds. Both estimators use the same fitted network and held-out observations at each scale. The empirical known-mechanism reference has zero EV; it is not an analytic population matrix.

At $\tau = 0.10$, the Jacobian has lower relative Frobenius error than at $\tau = 0.20$ (0.286 versus 0.456), but more violations (17.2 versus 5.6). At $\tau = 0.10$, the outer product has larger matrix error yet fewer violations than the Jacobian. Global matrix error and positive semidefiniteness alone do not rank these orders correctly. Both estimators fail at $\tau = 0.05$. This scale also makes the fixed penalty 16 times stronger relative to the unscaled fitting loss than at $\tau = 0.20$, so the comparison does not isolate smoothing bias from training effects.

All 15 Jacobian matrices are indefinite. We also check whether the sufficient diagonal-gap condition explains correct leaf decisions. On forced true-leaf paths, neither estimator passes the empirical test at any of 960 correlated states, although the diagonal minimizer remains a true leaf at $\tau = 0.10, 0.20$. Failure of this sufficient test therefore does not predict a wrong removal. These paths cover only the first 32 steps and use an empirical reference, not a certificate for a complete learned order (Appendix M; Table 13).

Scope beyond these settings. The high-dimensional diagnostic tests whether increasing sample size resolves learned-order errors; the observed trend is non-monotone (Table 21). Sachs tests transfer beyond the synthetic mechanisms. SSTS gives EV 11.4 ± 0.5, above the random-order expectation of 8.5, and SHD 16.8 ± 0.8 (Table 6). The synthetic recovery does not extend to this benchmark under the tested protocol. The linear checks test exactness, and the ablation and mechanism-shift checks test sensitivity (Appendix H).

6 Related Work

Precision Elimination and Expected Curvature. Ghoshal & Honorio (2018) give a recursive algorithm that selects a minimum precision diagonal and removes the corresponding terminal vertex by a Schur update. Their finite-sample analysis uses a modified precision-estimation step to control accumulated error. CaPS derives the expected score-Jacobian diagonal criterion for linear and nonlinear relations under conditions on noise scales and mechanism derivatives (Xu et al., 2024). Our equal-noise leaf formula is a special case of that criterion. These results do not by themselves justify recursive ordering from the original nonlinear matrix: identifying a leaf does not establish that enough information remains after its removal. Our construction and marginalization identity examine this step.

Access to Score Information. SCORE identifies leaves from score-Jacobian variation and repeats score estimation after removing a leaf (Rolland et al., 2022). DAS uses further score-Jacobian struc-

ture to restrict candidate parents before pruning (Montagna et al., 2023). DiffAN learns a diffusion-based score estimator and handles successive reduced systems through either a residue correction or a masking approximation (Sanchez et al., 2023). Its masking variant already avoids retraining. The restriction studied here is narrower: after aggregation, SSTS makes no further sample-level score queries. Montagna et al. (2025) analyze which causal relations are identifiable from scores in linear, non-linear, and latent-variable models. Our information limit concerns one joint expectation of score derivatives, not access to the score function itself.

Other Approaches to Ordering. Order-based methods search over permutations and select edges consistent with the chosen order (Teyssier & Koller, 2005; Bühlmann et al., 2014). Additive noise models supply observational identifiability under restrictions on the mechanisms and noise (Hoyer et al., 2008; Peters et al., 2014). Autoregressive flows offer another order-sensitive parameterization (Khemakhem et al., 2021). Xu et al. (2025) derive a Kolmogorov-complexity ordering criterion and implement a minimum-description-length approximation in TOPIC, treating continuous i.i.d. data and specified non-i.i.d. settings. These methods use criteria and representations beyond the matrix-only input of Theorem 3.1.

Continuous Causal Discovery. NOTEARS expresses acyclicity as a smooth equality constraint on a weighted adjacency matrix (Zheng et al., 2018). Later work extends continuous structure learning to non-linear neural mechanisms (Lachapelle et al., 2020), graph neural networks (Yu et al., 2019), likelihood-based objectives for linear DAG models (Ng et al., 2020), differentiable Bayesian formulations (Lorch et al., 2021), and log-determinant acyclicity characterizations (Bello et al., 2022). These methods learn graph parameters or distributions over graphs through objectives or priors that encode acyclicity. SSTS removes graph-constrained optimization from the ordering stage only; it does not replace the mechanism estimators used by these methods.

7 Discussion and Limitations

Information loss and approximation error. The counterexample shows that the joint population matrix can lose information needed for a complete order. The covariance identity identifies the discrepancy in a Schur update, but a nonzero discrepancy can leave the selected leaf unchanged. Estimation adds another source of error: a learned matrix can produce poor orders even when the known-mechanism matrix succeeds. A first-leaf criterion alone cannot justify replacing all later marginal queries with a fixed statistic; the statistic must retain the information needed for those decisions.

What an estimation guarantee would need to control. The paired diagnostics show that lower global matrix error can accompany more order violations. Our deterministic condition compares diagonal error with leaf separation, and the recurrence bounds propagated error and the covariance discrepancy. Both apply to scalar, unregularized elimination. A statistical guarantee for the fitted procedure would need to bound the initial smoothing, fitting, and averaging errors, then account for block selection and ridge regularization. A learned-score concentration rate remains missing, and the deterministic condition can be conservative even along a prescribed correct path.

Empirical scope. The public synthetic comparison covers one sparse `tanh` family. Common linear Lasso can miss symmetric nonlinear effects, and SSTS performs poorly on Sachs. Timings come from a shared server and combine differences in initialization, occupancy, and CPU/GPU execution paths. Some high-dimensional configurations complete with substantial order errors, so execution at these dimensions should not be read as accurate recovery. The guarantees exclude heteroscedastic, post-nonlinear, confounded, and error-in-variables models.

8 Conclusion

The expected joint score Jacobian can identify a first leaf without determining a complete order. Our construction proves this information limit; the covariance identity explains the nonlinear Schur error. SSTS recovers valid orders at 50 variables in the common-pruning benchmark, retains errors at 100, and performs poorly on Sachs. An ordering guarantee over the stated nonlinear model class requires information beyond the joint expectation or further model restrictions, not just better estimation of the same matrix.

REPRODUCIBILITY STATEMENT

Section 2 defines the model, notation, and ordering task. Appendix A states the regularity conditions and supporting results; Appendix C gives proofs. Algorithm 1 specifies extraction. Section 5.1 and Appendix E identify the experimental settings, baselines, and replication units. The code supplement includes saved server results, source fingerprints, baseline snapshots, environment requirements, and scripts that regenerate comparison tables and diagnostic figures without retraining. Summary-only supplementary records and their dispersion conventions are identified separately.

AI USE STATEMENT

Generative AI tools assisted with manuscript organization, drafting and editing, literature discovery, and the development and review of experimental code. We also used these tools to discuss theoretical claims, counterexample constructions, and proof steps, and to audit assumptions and derivations. For the experiments, assistance included designing diagnostic and sensitivity checks, preparing data-generation and evaluation scripts, and interpreting results. Synthetic observations were sampled from the structural models specified in the paper using numerical code. The authors independently checked and revised AI-assisted text and suggestions, verified references against the cited sources, examined the mathematical arguments, and tested the code by rerunning the relevant computations. The authors take full responsibility for the mathematical claims, experiments, code, and final text.

REFERENCES

Kevin Bello, Bryon Aragam, and Pradeep Ravikumar. DAGMA: Learning DAGs via M-matrices and a log-determinant acyclicity characterization. In *Advances in Neural Information Processing Systems*, volume 35, pp. 8226–8239, 2022.

Peter Bühlmann, Jonas Peters, and Jan Ernest. CAM: Causal additive models, high-dimensional order search and penalized regression. *The Annals of Statistics*, 42(6):2526–2556, 2014. doi: 10.1214/14-AOS1260.

Asish Ghoshal and Jean Honorio. Learning linear structural equation models in polynomial time and sample complexity. In *Proceedings of the Twenty-First International Conference on Artificial Intelligence and Statistics*, volume 84 of *Proceedings of Machine Learning Research*, pp. 1466–1475. PMLR, 2018. URL https://proceedings.mlr.press/v84/ghoshal18a.html.

Patrik O Hoyer, Dominik Janzing, Joris M Mooij, Jonas Peters, and Bernhard Schölkopf. Nonlinear causal discovery with additive noise models. In *Advances in Neural Information Processing Systems*, volume 21, pp. 689–696, 2008.

Aapo Hyvärinen. Estimation of non-normalized statistical models by score matching. *Journal of Machine Learning Research*, 6:695–709, 2005.

Ilyes Khemakhem, Ricardo Monti, Robert Leech, and Aapo Hyvärinen. Causal autoregressive flows. In *Proceedings of the 24th International Conference on Artificial Intelligence and Statistics*, volume 130 of *Proceedings of Machine Learning Research*, pp. 3520–3528. PMLR, 2021. URL https://proceedings.mlr.press/v130/khemakhem21a.html.

Sébastien Lachapelle, Philippe Brouillard, Tristan Deleu, and Simon Lacoste-Julien. Gradient-based neural DAG learning. In *International Conference on Learning Representations (ICLR)*, 2020. URL https://arxiv.org/abs/1906.02226.

Steffen L Lauritzen. *Graphical Models*. Number 17 in Oxford Statistical Science Series. Clarendon Press, Oxford, 1996.

Lars Lorch, Jonas Rothfuss, Bernhard Schölkopf, and Andreas Krause. DiBS: Differentiable bayesian structure learning. In *Advances in Neural Information Processing Systems*, volume 34, pp. 24111–24123, 2021.

Francesco Montagna, Nicoletta Noceti, Lorenzo Rosasco, Kun Zhang, and Francesco Locatello. Scalable causal discovery with score matching. In *Proceedings of the Second Conference on Causal Learning and Reasoning*, volume 213 of *Proceedings of Machine Learning Research*, pp. 752–771. PMLR, 2023. URL https://proceedings.mlr.press/v213/montagna23b.html.

Francesco Montagna, Philipp Michael Faller, Patrick Blöbaum, Elke Kirschbaum, and Francesco Locatello. Score matching through the roof: Linear, nonlinear, and latent variables causal discovery. In *Proceedings of the Fourth Conference on Causal Learning and Reasoning*, volume 275 of *Proceedings of Machine Learning Research*, pp. 552–605. PMLR, 2025. URL https://proceedings.mlr.press/v275/montagna25a.html.

Ignavier Ng, AmirEmad Ghassami, and Kun Zhang. On the role of sparsity and DAG constraints for learning linear DAGs. In *Advances in Neural Information Processing Systems*, volume 33, pp. 17943–17954, 2020.

Jonas Peters, Joris M Mooij, Dominik Janzing, and Bernhard Schölkopf. Causal discovery with continuous additive noise models. *Journal of Machine Learning Research*, 15:2009–2053, 2014. URL https://jmlr.org/papers/v15/peters14a.html.

Paul Rolland, Volkan Cevher, Matthäus Kleindessner, Chris Russell, Dominik Janzing, Bernhard Schölkopf, and Francesco Locatello. Score matching enables causal discovery of nonlinear additive noise models. In *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pp. 18741–18753. PMLR, 2022. URL https://proceedings.mlr.press/v162/rolland22a.html.

Karen Sachs, Omar Perez, Dana Pe'er, Douglas A Lauffenburger, and Garry P Nolan. Causal protein-signaling networks derived from multiparameter single-cell data. *Science*, 308(5721):523–529, 2005. doi: 10.1126/science.1105809.

Pedro Sanchez, Xiao Liu, Alison Q O'Neil, and Sotirios A Tsaftaris. Diffusion models for causal discovery via topological ordering. In *International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=Idusfje4-Wq.

Marc Teyssier and Daphne Koller. Ordering-based search: A simple and effective algorithm for learning bayesian networks. In *Proceedings of the Twenty-First Conference on Uncertainty in Artificial Intelligence*, Arlington, Virginia, 2005. AUAI Press. URL https://arxiv.org/abs/1207.1429.

Pascal Vincent. A connection between score matching and denoising autoencoders. *Neural Computation*, 23(7):1661–1674, 2011. doi: 10.1162/NECO_a_00142.

Sascha Xu, Sarah Mameche, and Jilles Vreeken. Information-theoretic causal discovery in topological order. In *Proceedings of the 28th International Conference on Artificial Intelligence and Statistics*, volume 258 of *Proceedings of Machine Learning Research*, pp. 2008–2016. PMLR, 2025. URL https://proceedings.mlr.press/v258/xu25d.html.

Zhuopeng Xu, Yujie Li, Cheng Liu, and Ning Gui. Ordering-based causal discovery for linear and nonlinear relations. In *Advances in Neural Information Processing Systems*, volume 37, pp. 4315–4340, 2024. doi: 10.52202/079017-0141. URL https://proceedings.neurips.cc/paper_files/paper/2024/hash/081b08068e4733ae3e7ad019fe8d172f-Abstract-Conference.html.

Yue Yu, Jie Chen, Tian Gao, and Mo Yu. DAG-GNN: DAG structure learning with graph neural networks. In *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pp. 7154–7163. PMLR, 2019. URL https://proceedings.mlr.press/v97/yu19a.html.

Ming Yuan and Yi Lin. Model selection and estimation in regression with grouped variables. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 68(1):49–67, 2006. doi: 10.1111/j.1467-9868.2005.00532.x.

Xun Zheng, Bryon Aragam, Pradeep K Ravikumar, and Eric P Xing. DAGs with NO TEARS: Continuous optimization for structure learning. In *Advances in Neural Information Processing Systems*, volume 31, pp. 9472–9483, 2018.

Xun Zheng, Chen Dan, Bryon Aragam, Pradeep Ravikumar, and Eric Xing. Learning sparse nonparametric DAGs. In *Proceedings of the Twenty Third International Conference on Artificial Intelligence and Statistics*, volume 108 of *Proceedings of Machine Learning Research*, pp. 3414–3425. PMLR, 2020. URL https://proceedings.mlr.press/v108/zheng20a.html.

A Assumptions and Supporting Results

A.1 Model and Regularity Conditions

We use the model, labeled coordinates, and ancestor-closed sets defined in Section 2. The observations contain every coordinate of each independent draw of $\mathbf{X}$; "fully observed" excludes latent nodes, not sampling error. The following conditions specify $\mathfrak{M}_d$.

Assumption A.1 (Common Gaussian noise)**.** For some $\sigma^2 > 0$,

$$\boldsymbol{\epsilon} = (\epsilon_1, \dots, \epsilon_d)^\top \sim \mathcal{N}(\mathbf{0}, \sigma^2 \mathbf{I}_d).$$

Assumption A.2 (Mechanism non-degeneracy)**.**

$$\forall (j,i) \in \mathcal{E}, \qquad 0 < \mathbb{E}_p\big[(\partial_j f_i(\mathbf{X}_{\mathrm{pa}(i)}))^2\big] < \infty.$$

For $i \in \mathcal{S}$, extend f_i to active coordinates by ignoring non-parents and write $r_i(\mathbf{x}_\mathcal{S}) = x_i - f_i(\mathbf{x}_{\mathrm{pa}(i)})$. Here $\|\cdot\|_F$ is the Frobenius norm; vector $\|\cdot\|_2$ is the Euclidean norm. Partial derivatives ∂_i refer to coordinate i, and $\partial_{ii} = \partial_i^2$. A multi-index $\alpha = (\alpha_j)_{j\in\mathcal{R}}$ has nonnegative integer entries, $|\alpha| = \sum_{j\in\mathcal{R}} \alpha_j$, and $\partial^\alpha = \prod_{j\in\mathcal{R}} \partial_j^{\alpha_j}$.

Assumption A.3 (Regularity)**.** For every $i \in \mathcal{V}$ and every $\mathcal{S} \in \mathfrak{A}(\mathcal{G})$,

$$f_i \in C^2(\mathbb{R}^{|\mathrm{pa}(i)|}), \qquad p_\mathcal{S} > 0, \quad p_\mathcal{S} \in C^2(\mathbb{R}^{|\mathcal{S}|}),$$

$$\mathbb{E}_{p_\mathcal{S}}\big[\|\mathbf{s}_\mathcal{S}\|_2^2 + \|\mathbf{H}^\mathcal{S}\|_F\big] < \infty, \quad \mathbb{E}_{p_\mathcal{S}}\big[\|\nabla_{\mathbf{x}_\mathcal{S}} f_i\|_2^2 + |r_i|\,\|\nabla^2_{\mathbf{x}_\mathcal{S}} f_i\|_F\big] < \infty \quad (i \in \mathcal{S}).$$

For all $i, j \in \mathcal{S}$ and almost every $\mathbf{x}_{\mathcal{S}\setminus\{i\}}$,

$$\lim_{x_i \to \pm\infty} \partial_j p_\mathcal{S}(\mathbf{x}_\mathcal{S}) = 0.$$

For $l \in \mathcal{L}(\mathcal{S})$, $\mathcal{R} = \mathcal{S} \setminus \{l\} \neq \varnothing$, and every multi-index α on $\mathcal{R}$ with $|\alpha| \le 2$,

$$\partial^\alpha p_\mathcal{R}(\mathbf{x}_\mathcal{R}) = \int_{\mathbb{R}} \partial^\alpha p_\mathcal{S}(\mathbf{x}_\mathcal{R}, x_l)\, dx_l, \qquad \int_{\mathbb{R}} |\partial^\alpha p_\mathcal{S}(\mathbf{x}_\mathcal{R}, x_l)|\, dx_l < \infty$$

for almost every $\mathbf{x}_\mathcal{R}$.

These conditions make the score integration-by-parts identity and the expectations in the proofs well-defined. They do not impose bounded derivatives: polynomial mechanisms in the counterexample are allowed. Non-degeneracy is needed to distinguish a non-leaf from a leaf, but not to compute the marginalization identity.

A.2 Background Results

Shared setting. Fix $\mathcal{S} \in \mathfrak{A}(\mathcal{G})$. The active ANM has density $p_\mathcal{S}$, matrix $\mathbf{F}^\mathcal{S}$, and leaf set $\mathcal{L}(\mathcal{S})$. For a removal, take $l \in \mathcal{L}(\mathcal{S})$ and $\mathcal{R} = \mathcal{S} \setminus \{l\} \neq \varnothing$. This setting also applies to the one-step identity below.

SCORE uses sample-wise diagonal variation (Rolland et al., 2022); CaPS uses its expectation (Xu et al., 2024). The equal-noise specialization of the latter shows exactly which nodes minimize the expected diagonal.

Theorem A.1 (Expected-diagonal criterion; cf. Xu et al., 2024)**.** *For $M \in \mathfrak{M}_d$ and every $\mathcal{S} \in \mathfrak{A}(\mathcal{G})$,*

$$F_{ii}^\mathcal{S} = \sigma^{-2} + \sigma^{-2} \sum_{j \in \mathrm{ch}(i)\cap\mathcal{S}} \mathbb{E}_{p_\mathcal{S}}\big[(\partial_i f_j(\mathbf{X}_{\mathrm{pa}(j)}))^2\big] \qquad \forall i \in \mathcal{S}, \tag{9}$$

$$\arg\min_{i\in\mathcal{S}} F_{ii}^\mathcal{S} = \{i \in \mathcal{S} : F_{ii}^\mathcal{S} = \sigma^{-2}\} = \mathcal{L}(\mathcal{S}).$$

Differentiating the ANM log-density gives the formula. Independence removes residual terms, and non-degeneracy makes the extra term positive for non-leaves (Appendix C.1).

Remark A.1 (What the leaf criterion supplies). Each squared derivative measures a child's local response to node i. Equal noise gives all nodes the same baseline; a leaf is distinguished by having no child contributions. This is the known current-leaf criterion, not a contribution of this paper. Our question concerns the information needed to reuse it: the theorem assumes access to the current marginal matrix, whereas Theorem 3.1 shows that the original joint matrix need not determine later orders. Neither result requires faithfulness, the assumption that every conditional independence in the distribution is implied by the graph.

Remark A.2 (Unequal noise variances). With independent noises $\epsilon_i \sim \mathcal{N}(0, \sigma_i^2)$ of unequal variances, diagonal magnitudes depend on noise scale and need not separate leaves. Appendix G.1 evaluates a scale-normalized diagonal-variance diagnostic on one nonlinear chain. It is not a general identification result.

For linear Gaussian models, the next matrix is available from the current one. This is the precision-matrix marginalization identity (Lauritzen, 1996) used in recursive leaf removal (Ghoshal & Honorio, 2018). Let $b_i \in \mathbb{R}$ be offsets and $\mathbf{B} \in \mathbb{R}^{d\times d}$ contain coefficients B_{ij} for $j \to i$, with $B_{ij} = 0$ when $j \notin \mathrm{pa}(i)$.

Theorem A.2 (Linear Schur update; cf. Ghoshal & Honorio, 2018)**.** *Under Assumption A.1, suppose* $f_i(\mathbf{x}_{\mathrm{pa}(i)}) = b_i + \sum_{j\in\mathrm{pa}(i)} B_{ij}x_j$ *for all* $i \in \mathcal{S}$*. Then, in the shared removal setting,*

$$\mathbf{F}^{\mathcal{R}} = \mathrm{Schur}_l(\mathbf{F}^{\mathcal{S}}) = \mathbf{F}^{\mathcal{S}}_{\mathcal{R}\mathcal{R}} - \mathbf{F}^{\mathcal{S}}_{\mathcal{R}l}(F^{\mathcal{S}}_{ll})^{-1}\mathbf{F}^{\mathcal{S}}_{l\mathcal{R}}. \tag{10}$$

The block factorization of the Gaussian precision cancels the removed leaf's contribution (Appendix C.2).

Remark A.3 (Why one matrix suffices in the linear case). The update returns another matrix of the same kind: the exact precision of the retained linear ANM, whose noise variance is unchanged. With nonzero edge coefficients, the leaf criterion can be applied again without new score queries. This known recursion is our reference case, not a new elimination method. Proposition 3.2 identifies the additional term that prevents the same recursion from remaining exact for general nonlinear mechanisms.

A.3 One-Step Sample-Wise Identity

For a nonlinear leaf, taking the Schur complement before averaging still recovers the next expected matrix in one step. The following statement uses the shared removal setting and the true sample Hessian, not a learned matrix.

Lemma A.3 (One-step expected Schur identity)**.** *Under Assumptions A.1 and A.3,*

$$H^{\mathcal{S}}_{ll}(\mathbf{x}_{\mathcal{S}}) = \sigma^{-2} \quad \forall \mathbf{x}_{\mathcal{S}}, \qquad \mathbb{E}_{p_{\mathcal{S}}}\left[\mathrm{Schur}_l(\mathbf{H}^{\mathcal{S}}(\mathbf{X}_{\mathcal{S}}))\right] = \mathbf{F}^{\mathcal{R}}.$$

The pivot is a positive constant, even if the full Hessian is indefinite. Differentiating the leaf conditional leaves a residual-weighted second derivative whose expectation vanishes (Appendix C.3).

Remark A.4 (Information for one step versus a full order). Keeping sample Hessians retains the variation lost by averaging. The lemma uses this information to correct the expectation for one removal, with the covariance supported on $\mathrm{pa}(l) \times \mathrm{pa}(l)$. It does not provide the sample Hessians needed to repeat the argument: each updated sample matrix differs from $\mathbf{H}^{\mathcal{R}}$ by a residual term. Appendix G.2 gives a counterexample to recursive use. Exactness of a single correction thus does not justify an exact nonlinear sorting algorithm based on repeated sample-wise Schur updates.

A.4 Stability Under Matrix Error

Correct selection does not require an exact matrix. We first compare the diagonal error with leaf separation, using e_{diag} and $\delta_{\mathcal{S}}$ from Eq. 8.

Proposition A.4 (Diagonal-gap stability)**.** *For* $M \in \mathfrak{M}_d$*,* $\mathcal{S} \in \mathfrak{A}(\mathcal{G})$*, and any finite real matrix* $\mathbf{M}$ *indexed by* $\mathcal{S}$*,*

$$e_{\mathrm{diag}}(\mathbf{M}, \mathcal{S}) < \delta_{\mathcal{S}}/2 \quad \Longrightarrow \quad \arg\min_{i\in\mathcal{S}} M_{ii} \subseteq \mathcal{L}(\mathcal{S}).$$

The proof bounds every estimated leaf diagonal above by $\sigma^{-2} + \delta_{\mathcal{S}}/2$ and every non-leaf diagonal below by the same threshold, with strict inequalities (Appendix C.6).

Remark A.5 (Accuracy relative to the ordering decision). The relevant comparison is between diagonal error and the separation created by the remaining child contributions. A biased update may preserve the leaf decision if this separation is large enough. The sufficient condition depends on the unknown population marginal and can be conservative. Section 5 examines how matrix error and ordering error differ in practice.

Sequential setting. Let $\mathcal{S}_t$ be the active set after t true leaf removals, with $\mathcal{S}_0 = \mathcal{V}$. Write $\mathbf{F}^{(t)} = \mathbf{F}^{\mathcal{S}_t}$ and let $\hat{\mathbf{F}}^{(t)}$ be the finite symmetric matrix produced by scalar, unregularized SSTS. For a candidate $l_t \in \mathcal{S}_t$, set $\mathcal{S}_{t+1} = \mathcal{R}_t = \mathcal{S}_t \setminus \{l_t\}$ and

$$m = \sigma^{-2}, \qquad \varepsilon_t = \|\hat{\mathbf{F}}^{(t)} - \mathbf{F}^{(t)}\|_2, \qquad a_t = \|\mathbf{F}^{(t)}_{\mathcal{R}_t l_t}\|_2,$$
$$c_t = \|\mathbf{C}^{\mathcal{S}_t}_{l_t}\|_2, \qquad L_t = 2 + 4a_t/m + 2(a_t/m)^2.$$

Matrix $\|\cdot\|_2$ denotes the spectral norm. The covariance term c_t is used only when $l_t \in \mathcal{L}(\mathcal{S}_t)$; it is defined by Eq. 6, with the gradient taken over $\mathcal{R}_t$. This true-leaf condition is an assumption of the next proposition, not a property assumed of every algorithmic selection.

Proposition A.5 (Sequential error recurrence)**.** *Under Assumptions A.1 and A.3, in the sequential setting,*

$$l_t \in \mathcal{L}(\mathcal{S}_t), \quad \varepsilon_t < m/2 \quad \Longrightarrow \quad \hat{F}^{(t)}_{l_t l_t} > 0, \qquad \left\|\mathrm{Schur}_{l_t}(\hat{\mathbf{F}}^{(t)}) - \mathbf{F}^{(t+1)}\right\|_2 \le L_t\varepsilon_t + c_t.$$

The proof first compares Schur updates of the estimated and population matrices, then applies Proposition 3.2 (Appendix C.7). The assumption that the selected node is a true leaf is essential.

Remark A.6 (Error propagated and error introduced). The term $L_t\varepsilon_t$ controls the effect of error already present, while c_t measures the discrepancy introduced by this removal. If $\varepsilon_t = 0$, the next error is exactly c_t by Eq. 6. Perfect estimation of the initial matrix cannot remove this source of nonlinear update error. In the linear reference case, $c_t = 0$ at every stage. The recurrence makes this distinction explicit without asserting a statistical rate for the learned score.

For a complete run, fix a deterministic tie-breaking rule and let the algorithm select

$$l_t \in \arg\min_{i \in \mathcal{S}_t} \hat{F}^{(t)}_{ii}.$$

At each stage reached after correct removals, define $\delta_t = \delta_{\mathcal{S}_t}$. The next corollary first establishes that this selected candidate is a true leaf, then uses L_t and c_t to propagate the error bound. These constants refer to the current population marginals, not just to the initial estimate.

Corollary A.6 (Sufficient condition for sequential recovery)**.** *Let $M \in \mathfrak{M}_d$ and $\hat{\mathbf{F}}^{(0)}$ be finite and symmetric. For scalar, unregularized SSTS, suppose*

$$\bar{\varepsilon}_0 \ge \|\hat{\mathbf{F}}^{(0)} - \mathbf{F}^{(0)}\|_2, \qquad \bar{\varepsilon}_{t+1} = L_t\bar{\varepsilon}_t + c_t,$$

$$\bar{\varepsilon}_t < \min\{m/2, \delta_t/2\} \quad (t = 0, \dots, d-2)$$

at every stage with $l_j \in \mathcal{L}(\mathcal{S}_j)$ for all $j < t$. Then

$$l_t \in \mathcal{L}(\mathcal{S}_t) \quad \forall t = 0, \dots, d-1, \qquad \hat{\pi} \in \Pi(\mathcal{G}).$$

Induction combines the diagonal-gap test with the recurrence; the final singleton is a leaf. The proof is in Appendix C.7.

Remark A.7 (Conditional recovery despite information loss). The two thresholds serve different purposes: $m/2$ keeps the scalar pivot positive, while $\delta_t/2$ makes the chosen diagonal minimizer a true leaf. Requiring both after every correct removal lets induction turn local decisions into a valid order. This is a conditional guarantee for individual models, not universal recovery contradicted by Theorem 3.1. Since $L_t \ge 2$, the bound can grow too quickly to explain successful runs. The learned block implementation also requires control of score-estimation error, block selection, and ridge regularization.

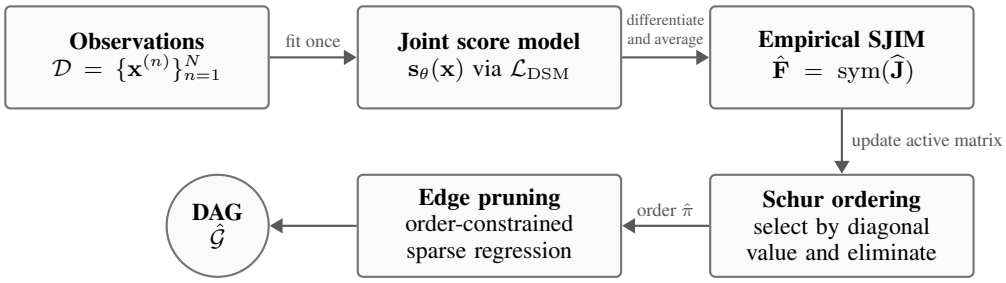


Figure 3: **SSTS pipeline.** The score model is fitted once. Its averaged Jacobian is used for leaf selection and Schur elimination; sparse regression is applied only after an order is obtained. Schur elimination equals leaf marginalization in linear Gaussian ANMs. In non-linear ANMs it omits the covariance term in Proposition 3.2.

B Score Fitting and Implementation Details

B.1 Score Fitting and Input-Weight Regularization

Let $\mathbf{Z} \sim \mathcal{N}(0, \mathbf{I}_d)$ independently of $\mathbf{X}$, and let $\tau > 0$. We fit $\mathbf{s}_\theta$ with objective $\mathcal{L}_{\mathrm{DSM}} + \mathcal{L}_{\mathrm{sparse}}$, where

$$\mathcal{L}_{\mathrm{DSM}}(\theta) = \tau^2 \mathbb{E}_{\mathbf{X},\mathbf{Z}} \|\mathbf{s}_\theta(\mathbf{X} + \tau\mathbf{Z}) + \mathbf{Z}/\tau\|_2^2, \qquad \mathcal{L}_{\mathrm{sparse}}(\theta) = \lambda_{\mathrm{sparse}} \sum_{j=1}^{d} \|[\mathbf{W}_1]_{:,j}\|_2. \tag{11}$$

Here $\mathbf{W}_1 \in \mathbb{R}^{h \times d}$ contains the input-layer weights, $[\mathbf{W}_1]_{:,j}$ is its j-th column, and h is the hidden width. Training replaces the data expectation by the empirical average. Without a model restriction or regularization, the population DSM minimizer is $\nabla \log p_\tau$, with $p_\tau = p * \mathcal{N}(0, \tau^2 \mathbf{I}_d)$ (Vincent, 2011). Averaging its Jacobian at clean observations targets $\mathbf{F}_{\tau|p} = \mathbb{E}_p[-\nabla^2 \log p_\tau]$, which can differ from $\mathbf{F}$. Appendix G.4 separates this smoothing bias from fitting and averaging errors.

The column penalty follows group-wise variable selection (Yuan & Lin, 2006). Later layers remain dense, so its effect on Jacobian sparsity is indirect. Equation 7 defines the symmetric estimate used for extraction. Figure 3 shows which stages access data and which operate only on that matrix.

B.2 Order-Constrained Pruning

For the estimated order $\hat{\pi}$, let $P_i = \{j : \hat{\pi}(j) < \hat{\pi}(i)\}$. These are permitted predecessors, not known ancestors. We follow the order-based pruning principle (Bühlmann et al., 2014), using linear Lasso on column-standardized observations $\mathbf{z}^{(n)}$. With intercept $a \in \mathbb{R}$, coefficients $\boldsymbol{\beta} \in \mathbb{R}^{|P_i|}$ indexed by P_i, and penalty weight $\alpha_i > 0$, solve

$$(\hat{a}_i, \hat{\boldsymbol{\beta}}_i) \in \arg\min_{a,\boldsymbol{\beta}} \frac{1}{2N} \sum_{n=1}^{N} \big(z_i^{(n)} - a - \boldsymbol{\beta}^\top \mathbf{z}_{P_i}^{(n)}\big)^2 + \alpha_i \|\boldsymbol{\beta}\|_1.$$

Three-fold cross-validation selects α_i. Write $\hat{\beta}_{ij}$ for the coefficient of predictor j in $\hat{\boldsymbol{\beta}}_i$. Retain $j \to i$ exactly when $j \in P_i$ and $|\hat{\beta}_{ij}| > 0.05$; the first variable has no candidate parents. The retained edge set $\hat{\mathcal{E}}$ defines $\hat{\mathcal{G}} = (\mathcal{V}, \hat{\mathcal{E}})$. Regression standardization occurs after ordering and does not change its input coordinates. This is not CAM's nonlinear pruning. Even a valid order need not give the correct graph under linear regression, and a wrong order changes the available regressors.

C Deferred Proofs and Derivations

C.1 Proof of Theorem A.1

Proof. Fix $\mathcal{S} \in \mathfrak{A}(\mathcal{G})$. Its structural factorization is

$$\log p_{\mathcal{S}}(\mathbf{x}_{\mathcal{S}}) = -\frac{|\mathcal{S}|}{2} \log(2\pi\sigma^2) - \frac{1}{2\sigma^2} \sum_{k \in \mathcal{S}} r_k(\mathbf{x}_{\mathcal{S}})^2, \quad r_k = x_k - f_k(\mathbf{x}_{\mathrm{pa}(k)}).$$

For $i \in \mathcal{S}$, $\partial_i r_i = 1$ and $\partial_i r_j = -\partial_i f_j$ for $j \in \mathrm{ch}(i) \cap \mathcal{S}$. Hence

$$s_{\mathcal{S},i}(\mathbf{x}_\mathcal{S}) = -\sigma^{-2} r_i + \sigma^{-2} \sum_{j \in \mathrm{ch}(i) \cap \mathcal{S}} r_j \partial_i f_j,$$

$$H^\mathcal{S}_{ii}(\mathbf{x}_\mathcal{S}) = \sigma^{-2} + \sigma^{-2} \sum_{j \in \mathrm{ch}(i) \cap \mathcal{S}} (\partial_i f_j)^2 - \sigma^{-2} \sum_{j \in \mathrm{ch}(i) \cap \mathcal{S}} r_j \partial_{ii} f_j. \tag{12}$$

At $\mathbf{X}_\mathcal{S}$, $r_j = \epsilon_j$. The derivative $\partial_{ii} f_j$ depends only on the parents of j, which are independent of ϵ_j. Assumption A.3 makes the product integrable, so its expectation is zero. Taking expectations proves Eq. 9. Non-degeneracy makes the sum positive for each non-leaf and the sum is empty for each leaf. □

C.2 Proof of Theorem A.2

Proof. Center the active Gaussian vector; the affine offsets do not affect its covariance or Hessian. With B_{ij} the coefficient of $j \to i$, its equations are $\mathbf{X}_\mathcal{S} - \mathbb{E}\mathbf{X}_\mathcal{S} = \mathbf{B}_{\mathcal{S}\mathcal{S}}(\mathbf{X}_\mathcal{S} - \mathbb{E}\mathbf{X}_\mathcal{S}) + \boldsymbol{\epsilon}_\mathcal{S}$. Let $\boldsymbol{\Sigma}_\mathcal{S} = \mathrm{Cov}_{p_\mathcal{S}}(\mathbf{X}_\mathcal{S})$ and $\mathbf{A}_\mathcal{S} = \mathbf{I}_{|\mathcal{S}|} - \mathbf{B}_{\mathcal{S}\mathcal{S}}$. A topological permutation makes $\mathbf{A}_\mathcal{S}$ triangular with unit diagonal. Thus

$$\boldsymbol{\Sigma}_\mathcal{S} = \sigma^2 \mathbf{A}_\mathcal{S}^{-1} \mathbf{A}_\mathcal{S}^{-\top}, \qquad \mathbf{F}^\mathcal{S} = \boldsymbol{\Sigma}_\mathcal{S}^{-1} = \sigma^{-2} \mathbf{A}_\mathcal{S}^\top \mathbf{A}_\mathcal{S}.$$

Place leaf l last, write $\mathbf{b} = \mathbf{B}_{l\mathcal{R}}^\top$, and set $\mathbf{A}_\mathcal{R} = \mathbf{I}_{|\mathcal{R}|} - \mathbf{B}_{\mathcal{R}\mathcal{R}}$. The leaf has no children, so

$$\mathbf{A}_\mathcal{S} = \begin{pmatrix} \mathbf{A}_\mathcal{R} & \mathbf{0} \\ -\mathbf{b}^\top & 1 \end{pmatrix}, \qquad \mathbf{F}^\mathcal{S} = \sigma^{-2} \begin{pmatrix} \mathbf{A}_\mathcal{R}^\top \mathbf{A}_\mathcal{R} + \mathbf{b}\mathbf{b}^\top & -\mathbf{b} \\ -\mathbf{b}^\top & 1 \end{pmatrix}. \tag{13}$$

The Schur update cancels $\sigma^{-2}\mathbf{b}\mathbf{b}^\top$ and leaves $\sigma^{-2}\mathbf{A}_\mathcal{R}^\top \mathbf{A}_\mathcal{R} = \mathbf{F}^\mathcal{R}$. The equality on the right follows because leaf removal leaves the structural equations on $\mathcal{R}$ unchanged. □

C.3 Proof of Lemma A.3

Proof. Use the shared removal setting from Appendix A.2. Since l is a leaf of $\mathcal{G}[\mathcal{S}]$,

$$\log p_\mathcal{S}(\mathbf{x}_\mathcal{S}) = \log p_\mathcal{R}(\mathbf{x}_\mathcal{R}) - \frac{r_l(\mathbf{x}_\mathcal{S})^2}{2\sigma^2} - \frac{1}{2}\log(2\pi\sigma^2).$$

With $\mathbf{g}_l = \nabla_{\mathbf{x}_\mathcal{R}} f_l$ as in Section 3, differentiation gives

$$\begin{aligned} \mathbf{H}^\mathcal{S}_{\mathcal{R}\mathcal{R}} &= \mathbf{H}^\mathcal{R} + \sigma^{-2}\mathbf{g}_l\mathbf{g}_l^\top - r_l\sigma^{-2}\nabla^2_{\mathbf{x}_\mathcal{R}} f_l, \\ \mathbf{H}^\mathcal{S}_{\mathcal{R}l} &= -\sigma^{-2}\mathbf{g}_l, \qquad H^\mathcal{S}_{ll} = \sigma^{-2}. \end{aligned} \tag{14}$$

Therefore,

$$\mathrm{Schur}_l(\mathbf{H}^\mathcal{S}(\mathbf{X}_\mathcal{S})) = \mathbf{H}^\mathcal{R}(\mathbf{X}_\mathcal{R}) - \epsilon_l \sigma^{-2} \nabla^2_{\mathbf{x}_\mathcal{R}} f_l(\mathbf{X}_{\mathrm{pa}(l)}). \tag{15}$$

No retained variable depends on ϵ_l. Its independence and zero mean imply that the integrable residual term has zero expectation. Averaging proves the lemma, without asserting a pointwise equality with the marginal Hessian. □

C.4 Proof of Proposition 3.2

Proof. Average the blocks in Eq. 14 under $p_\mathcal{S}$:

$$\mathbf{F}^\mathcal{S}_{\mathcal{R}\mathcal{R}} = \mathbf{F}^\mathcal{R} + \sigma^{-2}\mathbb{E}[\mathbf{g}_l\mathbf{g}_l^\top], \qquad \mathbf{F}^\mathcal{S}_{\mathcal{R}l} = -\sigma^{-2}\mathbb{E}[\mathbf{g}_l], \qquad F^\mathcal{S}_{ll} = \sigma^{-2}.$$

Substituting into Eq. 5 yields

$$\mathrm{Schur}_l(\mathbf{F}^\mathcal{S}) - \mathbf{F}^\mathcal{R} = \sigma^{-2}\big(\mathbb{E}[\mathbf{g}_l\mathbf{g}_l^\top] - \mathbb{E}[\mathbf{g}_l]\mathbb{E}[\mathbf{g}_l]^\top\big) = \mathbf{C}^\mathcal{S}_l.$$

A covariance matrix is positive semidefinite. For $i \notin \mathrm{pa}(l)$, $g_{l,i} = 0$, so the corresponding row and column of the covariance vanish. A constant gradient has zero covariance. Neither this calculation nor the preceding lemma uses Assumption A.2. □

C.5 Proof of Theorem 3.1

Proof. In each model take independent standard Gaussian noises and define

$$M_A: \quad X_1 = \epsilon_1, \quad X_2 = X_1^2 - 1 + \epsilon_2, \quad X_3 = (X_2^2 - 3)/\sqrt{3} + \epsilon_3,$$
$$M_B: \quad X_2 = \epsilon_2, \quad X_1 = X_2^2 - 1 + \epsilon_1, \quad X_3 = (X_1^2 - 3)/\sqrt{3} + \epsilon_3.$$

These give $1 \to 2 \to 3$ and $2 \to 1 \to 3$ on fixed labels. In M_A,

$$\mathbb{E}X_1 = 0, \quad \mathbb{E}X_1^2 = 1, \quad \mathbb{E}X_2 = 0, \quad \mathbb{E}X_2^2 = \mathrm{Var}(X_1^2) + 1 = 3.$$

The two nonzero mechanism derivatives, $u_2 = 2X_1$ and $u_3 = 2X_2/\sqrt{3}$, have mean zero and second moment 4. With $r_i(\mathbf{x}) = x_i - f_i(\mathbf{x}_{\mathrm{pa}(i)})$, the residual-Hessian terms vanish in expectation by noise independence. Hence

$$\mathbf{F}(M_A) = \sum_{i=1}^{3} \mathbb{E}[\nabla r_i \nabla r_i^\top] = \begin{pmatrix} 1 + \mathbb{E}u_2^2 & -\mathbb{E}u_2 & 0 \\ -\mathbb{E}u_2 & 1 + \mathbb{E}u_3^2 & -\mathbb{E}u_3 \\ 0 & -\mathbb{E}u_3 & 1 \end{pmatrix} = \mathrm{diag}(5, 5, 1).$$

Exchanging labels 1 and 2 leaves this matrix unchanged. The mechanisms are polynomial and their edge derivatives have positive finite second moments. The triangular transformation of Gaussian noise gives a positive smooth density with the required finite polynomial moments and vanishing boundary terms. Marginals after leaf removal have the same form, so both models belong to $\mathfrak{M}_3$.

The chains have unique orders, $(1, 2, 3)$ and $(2, 1, 3)$ as lists. A deterministic function of their common matrix returns the same list and fails on at least one. For a randomized rule, the disjoint success events have probabilities summing to at most one, so success is at most $1/2$ on at least one model. This is a statement about the common statistic, not equality of observational distributions. □

C.6 Proof of Proposition A.4

Proof. If $\mathcal{L}(\mathcal{S}) = \mathcal{S}$, any node is a leaf. Otherwise, $0 < \delta_{\mathcal{S}} < \infty$. Write $e = e_{\mathrm{diag}}(\mathbf{M}, \mathcal{S}) < \delta_{\mathcal{S}}/2$. For all $l \in \mathcal{L}(\mathcal{S})$ and $i \in \mathcal{S} \setminus \mathcal{L}(\mathcal{S})$,

$$M_{ll} \le \sigma^{-2} + e < \sigma^{-2} + \delta_{\mathcal{S}}/2 < \sigma^{-2} + \delta_{\mathcal{S}} - e \le M_{ii}.$$

The first and last inequalities use Theorem A.1. No non-leaf attains the minimum. □

C.7 Proof of Proposition A.5 and Corollary A.6

Proof. At a stage with true leaf l_t, partition the population and estimated matrices as

$$\mathbf{F}^{(t)} = \begin{pmatrix} \mathbf{A} & \mathbf{b} \\ \mathbf{b}^\top & m \end{pmatrix}, \qquad \hat{\mathbf{F}}^{(t)} = \begin{pmatrix} \mathbf{A} + \mathbf{E} & \mathbf{b} + \mathbf{e} \\ (\mathbf{b} + \mathbf{e})^\top & m + r \end{pmatrix}.$$

Each block error is a compression of $\hat{\mathbf{F}}^{(t)} - \mathbf{F}^{(t)}$, so $\|\mathbf{E}\|_2, \|\mathbf{e}\|_2, |r| \le \varepsilon_t$. The constant pivot $m = \sigma^{-2}$ follows from the leaf Hessian formula, without non-degeneracy. Since $\varepsilon_t < m/2$, $m + r > m/2 > 0$. With $a_t = \|\mathbf{b}\|_2$,

$$\begin{aligned} &\left\|\mathrm{Schur}_{l_t}(\hat{\mathbf{F}}^{(t)}) - \mathrm{Schur}_{l_t}(\mathbf{F}^{(t)})\right\|_2 \\ &\quad \le \|\mathbf{E}\|_2 + \left\| \frac{(\mathbf{b} + \mathbf{e})(\mathbf{b} + \mathbf{e})^\top}{m + r} - \frac{\mathbf{b}\mathbf{b}^\top}{m} \right\|_2 \\ &\quad \le \varepsilon_t + \frac{2a_t\varepsilon_t + \varepsilon_t^2}{m - \varepsilon_t} + \frac{a_t^2\varepsilon_t}{m(m - \varepsilon_t)} \\ &\quad \le \left[2 + 4a_t/m + 2(a_t/m)^2\right]\varepsilon_t = L_t\varepsilon_t. \end{aligned}$$

Proposition 3.2 gives

$$\mathbf{F}^{(t+1)} = \mathrm{Schur}_{l_t}(\mathbf{F}^{(t)}) - \mathbf{C}_{l_t}^{\mathcal{S}_t}.$$

The triangle inequality proves the recurrence.

For the corollary, start with $\varepsilon_0 \le \bar{\varepsilon}_0$ and proceed by induction. Spectral error bounds every diagonal error. Thus $\bar{\varepsilon}_t < \delta_t/2$ makes the selected minimizer a true leaf, and $\bar{\varepsilon}_t < m/2$ makes its pivot positive. The recurrence gives $\varepsilon_{t+1} \le \bar{\varepsilon}_{t+1}$. The final remaining node is a leaf, so reversing the removals yields an order in $\Pi(\mathcal{G})$. □

D TOPOLOGICAL AMBIGUITY AND KENDALL'S τ

In continuous causal discovery, an extracted sequence $\hat{\pi}$ is sometimes evaluated against a ground-truth topological order π^* using 1D rank correlation metrics, such as Kendall's τ. For sparse DAGs with parallel branches and wide leaf strata, this evaluation can penalize valid topological orders.

Use $\mathcal{G}$ and $\Pi(\mathcal{G})$ from Section 2. Write $u \rightsquigarrow v$ when a directed path from u to v exists in $\mathcal{G}$.

Definition D.1 (Topological Incomparability)**.** For distinct $u, v \in \mathcal{V}$, define

$$u \parallel_{\mathcal{G}} v \quad \Longleftrightarrow \quad u \not\rightsquigarrow v \text{ and } v \not\rightsquigarrow u.$$

Incomparable nodes have no graph-imposed relative order; this alone does not assert conditional independence. Comparing an estimate with one reference permutation nevertheless fixes their relative order. To isolate this effect, fix $P \subseteq \mathcal{V}$ of size W whose nodes are pairwise incomparable and can be freely permuted within fixed positions of a valid order. Keep every other pairwise order fixed. Let U, V be independent uniform permutations of these positions, producing valid position maps π_U, π_V. Write τ_{K} for Kendall's correlation, distinct from the DSM scale τ.

Proposition D.1 (Kendall's τ for an interchangeable stratum)**.** *For $d \geq 2$ in this fixed-position setting,*

$$\mathrm{EV}(\pi_U, \mathcal{G}) = \mathrm{EV}(\pi_V, \mathcal{G}) = 0, \qquad \mathbb{E}[\tau_{\mathrm{K}}(\pi_U, \pi_V)] = 1 - \frac{W(W-1)}{d(d-1)}. \tag{16}$$

Proof. Let K be the number of discordant pairs. Discordance can occur only within P. Each pair in P is discordant with probability $1/2$, so $\mathbb{E}[K] = \frac{1}{2}\binom{W}{2} = W(W-1)/4$. Substituting into $\tau_{\mathrm{K}} = 1 - 2K/\binom{d}{2}$ proves the result. □

Remark D.1 (Edge Violations). Proposition D.1 shows that rank correlation can depend on arbitrary choices inside an incomparable stratum. We also report **Edge Violations (EV)**, the number of true edges contradicted by an estimated order:

$$\mathrm{EV}(\hat{\pi}, \mathcal{G}) = \left|\{(j,i) \in \mathcal{E} \mid \hat{\pi}(j) > \hat{\pi}(i)\}\right| \tag{17}$$

EV evaluates the directional consistency of the extracted order $\hat{\pi}$. The final sparsity pattern is evaluated after structural pruning using SHD.

E EXPERIMENTAL SETTINGS AND PROTOCOLS

E.1 DATA, EVALUATION, AND REPLICATES

The common-pruning benchmark and matched-schedule control use the centered, globally scaled ER-$\tanh$ observations defined in Section 5.1. A seed determines the graph, mechanisms, and observations, and also initializes stochastic methods. Pairing is checked through data and graph hashes. Dispersion includes SCM and observation variation, not just optimization randomness. The known graph is used for evaluation, never supplied to a learned ordering method.

Common pruning uses the same linear Lasso rule for every order (Appendix B.2). The candidate predictors still change because different orders permit different parent sets. Within each trial, cross-validation uses observations alone; the coefficient threshold is fixed at 0.05 for all compared orders.

Metric definitions. Let $\mathcal{E}$ and $\hat{\mathcal{E}}$ be the true and estimated directed edge sets, $\mathcal{E}_{\mathrm{hit}} = \mathcal{E} \cap \hat{\mathcal{E}}$, and $\mathcal{E}_{\mathrm{rev}} = \{(j,i) \in \mathcal{E} \setminus \hat{\mathcal{E}} : (i,j) \in \hat{\mathcal{E}}\}$. We compute

$$\mathrm{EV}(\hat{\pi}, \mathcal{G}) = \sum_{(j,i)\in\mathcal{E}} \mathbf{1}\{\hat{\pi}(j) > \hat{\pi}(i)\}, \qquad \mathrm{NEV} = \frac{\mathrm{EV}}{|\mathcal{E}|},$$

$$\mathrm{SHD} = |\mathcal{E} \triangle \hat{\mathcal{E}}| - |\mathcal{E}_{\mathrm{rev}}|, \qquad \mathrm{TPR} = \frac{|\mathcal{E}_{\mathrm{hit}}|}{|\mathcal{E}|}, \qquad \mathrm{FDR} = \frac{|\hat{\mathcal{E}} \setminus \mathcal{E}|}{|\hat{\mathcal{E}}|}.$$

Here NEV denotes normalized edge violations, FDR denotes false discovery rate, and $\triangle$ is symmetric difference. Subtracting $|\mathcal{E}_{\mathrm{rev}}|$ counts a reversal once rather than as a deletion and addition. Ratios with zero denominators are set to zero. EV does not measure edge selection, while SHD, TPR, and FDR depend on ordering and pruning. Appendix D explains why one reference permutation can penalize a valid order.

E.2 Compared Procedures

Internal control. The masked-score model is a control constructed for this study, not a public baseline. It lets the ordering stage query different variable subsets, which SSTS cannot do after aggregation. SSTS and the control share hidden width $\min(1024, \max(128, \lfloor 1.5d \rfloor))$, two GELU hidden layers, AdamW with initial learning rate 0.002 and weight decay 10^{-5}, a cosine schedule, gradient clipping at 1, 200 epochs, and batch size 1000. Both use all 5000 observations for order evaluation. The SSTS and control parameter counts are 21,780/24,340 at $d = 20$, 29,490/35,890 at $d = 50$, and 52,900/67,900 at $d = 100$.

For each observation, the control samples a subset size uniformly from $\{1, \ldots, d\}$ and then a subset uniformly at that size. Its mask $\mathbf{m} \in \{0, 1\}^d$ has $m_i = 1$ exactly for retained coordinates. Write $\odot$ for elementwise multiplication and $[\mathbf{u}, \mathbf{v}]$ for vector concatenation. The network $\mathbf{q}_\phi : \mathbb{R}^{2d} \to \mathbb{R}^d$, with parameters ϕ, receives the masked observation and mask and minimizes

$$\mathcal{L}_{\text{mask}}(\phi) = \tau^2 \mathbb{E}_{\mathbf{X},\mathbf{Z},\mathbf{m}} \left\| \mathbf{m} \odot (\mathbf{q}_\phi([\mathbf{m} \odot (\mathbf{X} + \tau\mathbf{Z}), \mathbf{m}]) + \mathbf{Z}/\tau) \right\|_2^2 .$$

Masks and Gaussian corruption are independent of the data and graph. At extraction, we fix the active-set mask, evaluate the mean negative Jacobian diagonal on the observations, and remove one minimum-diagonal variable. We repeat this procedure with the reduced mask. The control has no input Group Lasso penalty. It matches SSTS in schedule and hidden width, but differs in parameter count, loss, regularization, and access to marginal information.

SSTS settings. For the common-pruning and matched-schedule settings, $\tau = 0.2$, $\gamma = 0.01$, and $\lambda_{\text{ridge}} = 10^{-4}$. The input penalty is zero at $d = 20$ and $2 \times 10^{-4}\sqrt{d}$ at $d \geq 50$. The same 200-epoch, batch-size-1000 schedule applies to their SSTS runs. The paired-matrix diagnostic and supplementary sweeps specify their changes separately.

Public implementations. CaPS (Xu et al., 2024) uses the unchanged ordering prefix from its pinned source, with Stein regularizers $\eta_G = \eta_H = 0.001$, followed by common Lasso. Its parent-score and native R pruning stages are excluded. SCORE (Rolland et al., 2022) uses the dodiscover ordering routine with native pruning disabled and the same Stein regularizers. DAS (Montagna et al., 2023) runs the dodiscover ordering and native pruning with $\alpha = 0.001$, cutoff 0.001, and parent limits $\min(5, d-1)$ and $\min(20, d-1)$; we save its native graph and also evaluate common Lasso on its order. Public kernel routines use all 5000 samples. The SCORE and DAS rows refer to these dodiscover implementations.

DiffAN (Sanchez et al., 2023) uses its public masking variant, a maximum of 3000 epochs with its native checkpoint selection, learning rate 0.001, and training batch size 1024. Ordering keeps the implementation's 1024-sample cap and evaluation micro-batches of 128. No extra coordinate normalization is applied. Its representation and training budget are not matched to SSTS. This experiment does not evaluate the residue variant on the synthetic grid. The adapter allows micro-batched Jacobian evaluation and bypasses unused CAM imports; common Lasso replaces native pruning. Source versions and compatibility changes are documented with the bundled implementations. These comparisons hold input data and downstream Lasso fixed, not estimator capacity, optimization budget, or execution device.

E.3 Paired-Matrix Diagnostic Protocol

For seeds 42–46, fix an ER DAG at $d = 100$, signed weights of magnitude in $[0.5, 1.5]$, Gaussian noise standard deviation $\sigma = 0.1$, and $f_i(\mathbf{x}_{\text{pa}(i)}) = \sum_{j \in \text{pa}(i)} w_{ji} \tanh(x_j)$. Draw 5000 training observations and $N_e = 2000$ independent evaluation observations. Center and scale both using training column means and one global standard deviation. Data hashes agree across the three DSM scales within each seed. The common-pruning benchmark instead applies $\tanh$ after summation and aggregates Jacobians on training observations; these are different settings.

Let $\boldsymbol{\mu}$ be the training column means and $a > 0$ the global standard deviation of the centered training data. The scaled coordinates are $\mathbf{z} = (\mathbf{x} - \boldsymbol{\mu})/a$, with mechanisms and noise standard deviation

$$\widetilde{f}_i(\mathbf{z}_{\text{pa}(i)}) = \frac{f_i(\boldsymbol{\mu}_{\text{pa}(i)} + a\mathbf{z}_{\text{pa}(i)}) - \mu_i}{a}, \qquad \widetilde{\sigma} = \sigma/a.$$

Write $\mathcal{D}_e$ for the scaled evaluation observations.

Train one model per seed and scale for 200 epochs. On the same $\mathcal{D}_e$, compute

$$\hat{\mathbf{F}}_J = -\frac{1}{N_e}\sum_{\mathbf{z}\in\mathcal{D}_e}\mathrm{sym}(\nabla_{\mathbf{z}}\mathbf{s}_\theta(\mathbf{z})), \qquad \hat{\mathbf{F}}_O = \frac{1}{N_e}\sum_{\mathbf{z}\in\mathcal{D}_e}\mathbf{s}_\theta(\mathbf{z})\mathbf{s}_\theta(\mathbf{z})^\top.$$

Define the scaled residual $\widetilde{r}_i(\mathbf{z}) = z_i - \widetilde{f}_i(\mathbf{z}_{\mathrm{pa}(i)})$. The known-mechanism reference is

$$\mathbf{F}_{\mathrm{ref}} = \frac{1}{N_e\widetilde{\sigma}^2}\sum_{\mathbf{z}\in\mathcal{D}_e}\sum_{i=1}^{d}\nabla_{\mathbf{z}}\widetilde{r}_i(\mathbf{z})\nabla_{\mathbf{z}}\widetilde{r}_i(\mathbf{z})^\top.$$

The moment reference omits the residual-Hessian term, whose expectation is zero. A second reference averages the full known Hessian. Neither is an analytic population matrix. Matrix error is $\|\hat{\mathbf{F}} - \mathbf{F}_{\mathrm{ref}}\|_F/\|\mathbf{F}_{\mathrm{ref}}\|_F$ in these common coordinates. Five independently generated SCM/data settings are reused across scales, not fifteen independent datasets.

Forced-path check. At each of the first 32 steps, remove the smallest-label true leaf, without ridge, and compare the updated estimate with a fresh known-mechanism marginal moment matrix. Only this diagnostic uses the graph to prescribe removals. Apply the diagonal-gap condition from Proposition A.4 with the empirical reference in place of the population matrix. There are $15 \times 2 \times 32 = 960$ correlated states across fits, estimators, and path positions. This prescribed partial path cannot certify a complete learned order or the regularized block algorithm.

E.4 Record Provenance and Dispersion

Record provenance. The matched-schedule setting uses vectorized masks and a cosine schedule for both models. Archived control runs that generated masks per sample in Python and omitted this schedule remain in the code package for provenance, but are excluded from the comparison tables. The Sachs control has a separate vectorized implementation. The main synthetic records use Python 3.10.20 and PyTorch `2.12.0.dev20260323+cu128`; source and configuration fingerprints are stored per trial.

Individual JSON records, CSV summaries, and saved matrices are available for the common-pruning, matched-schedule, and paired-matrix settings. Some supplementary settings have only aggregate summaries and generating scripts; their per-seed raw records are unavailable. Appendix E.5 specifies these settings.

Dispersion conventions. For n replicate values, the common-pruning, matched-schedule, paired-matrix, Sachs, and high-dimensional matrix settings report sample standard deviations (variance denominator $n-1$). Supplementary graph-recovery, sample-size, pruning, checkpoint, ablation, mechanism-shift, topology/noise, order-scaling, and capacity records retain population standard deviations (variance denominator n). These rounded source summaries are neither converted nor pooled with the main trials.

E.5 Supplementary Setting Register

Section 5.1 specifies the common-pruning benchmark, matched-schedule control, and paired-matrix diagnostic. The settings below cover the supplementary experiments. Seeds are those specified in the generating scripts. Where only aggregate results are available, the script configuration does not verify individual trial outcomes.

Linear precision test. Five SCM/data seeds 42–46; $N = 10000$; $d \in \{10, 20, 50, 100, 500, 1000\}$. The saved records use ER graphs with expected degree 2, signed weights of magnitude in $[0.5, 1.5]$, and Gaussian noise standard deviation 0.1. Scalar Schur ordering is applied to both the analytic precision and inverse empirical covariance. The script's numerical ridge is 10^{-10}. No neural score is fitted. Results: Table 3.

Unequal-noise chain. A five-node chain uses $X_i = \tanh(2X_{i-1}) + \epsilon_i$ after a Gaussian root. The script specifies seed 42, $N = 20000$, 1000 training epochs, batch size 2000, and DSM scale 0.05. It compares equal noise standard deviations of 1 with $(5, 1, 1, 1, 0.2)$. Expected diagonal, absolute variance, and squared coefficient of variation are compared for the first leaf only. The retained table is an aggregate diagnostic, not a multi-seed guarantee (Table 4).

Sample-wise recursion. One seed, 42; $d \in \{50, 100\}$; $N = 5000$. Block-SSTS is compared with recursion on stored sample Hessians, with at most 1000 Hessian samples for the latter. Both use fitted scores; neither is an exact nonlinear marginalization reference. Results: Table 5.

Graph-recovery and sample-size sweeps. The ER-$\tanh$ SSTS tests use seeds 42–46. Graph recovery varies $d \in \{200, 500\}$ at $N = 5000$ (Table 14); the sample-size sweep fixes $d = 100$ and varies $N \in \{500, 1000, 2000, 5000, 10000, 20000\}$ (Table 15). These are separate fitted runs, not additional public-baseline comparisons. The order-source control fixes $d = 100$, $N = 5000$, and compares a true order, SSTS order, unrestricted random permutation, and empirical-precision order under the same Lasso (Table 16).

Checkpoint averaging. At $N = 5000$, average the last $K \in \{1, 3, 5\}$ checkpoint matrices for $d = 100$ using seed groups 42–46 and 47–51. At $d = 500$, use $K \in \{1, 3\}$ and seeds 42–44. A checkpoint is saved at the end of each epoch. The reported matrix is the arithmetic mean of SJIM estimates from the last K consecutive epoch-end checkpoints, evaluated on the same training observations. These checkpoints come from one fit and are not independent replicates. Results: Table 17.

Input-penalty and block-tolerance sweeps. The scripts specify seeds 42–46, 200 epochs, and DSM scale 0.05. The input-penalty setting fixes $d = 100$, $N = 500$, and $\gamma = 0.05$, and varies $\lambda_{\text{sparse}} \in \{0, 10^{-5}, 5 \times 10^{-5}, 10^{-4}, 1.1 \times 10^{-4}\}$. The block-tolerance setting fixes $d = 500$, $N = 5000$, and $\lambda_{\text{sparse}} = 0$, and varies $\gamma \in \{0, 0.01, 0.05, 0.15\}$. These settings differ from the main SSTS configuration. Results: Figure 4 and Table 18.

Mechanism and topology/noise shifts. Both use $d = 50$, $N = 5000$, seeds 42–46, 200 epochs, DSM scale 0.05, and input penalty 5×10^{-4}. With $u_i = \sum_{j \in \text{pa}(i)} w_{ji} X_j$, mechanism shifts compare additive $\tanh(u_i) + \epsilon_i$, multiplicative $\tanh(u_i) + (0.5 + 1.5/(1 + e^{-u_i}))\epsilon_i$, and post-nonlinear $(\tanh(u_i) + \epsilon_i)^3$ responses; roots remain Gaussian. The topology/noise setting crosses ER and scale-free graphs with Gaussian, Exponential, and Gumbel noise. Results: Tables 19 and 23.

Order scaling and capacity-constrained extraction. At $N = 5000$, the order-scaling setting uses seeds 42–46 for $d = 50, 100$ and seeds 42–44 for $d = 500, 1000$. It compares SSTS with unrestricted random permutations, without evaluating a pruned graph at $d = 1000$ (Table 20). The capacity-constrained setting uses seeds 42–46 and $d \in \{500, 1000, 5000\}$, with micro-batch Jacobians, scalar rank-one updates, and a hidden-width cap (Table 22). Its Schur timer excludes Jacobian construction and is not the main ordering timer.

High-dimensional matrix diagnostic. Use $d = 1000$ ER-$\tanh$ models, expected degree 2, noise standard deviation 0.1, $\lambda_{\text{ridge}} = 10^{-4}$, and $\gamma = 0.01$. Known-mechanism matrices vary SCM/data seeds 42–46 at $N \in \{5000, 10000, 20000, 100000\}$. The moment construction omits the residual-Hessian term, whose expectation is zero; the full-Hessian construction includes it. Learned matrices fix SCM/data seed 42, use nested samples of sizes 5000, 10000, and 20000, and vary training seeds 101–105. Each fit uses 200 epochs, batch size 1000, $\tau = 0.2$, hidden width capped at 1024, and $\lambda_{\text{sparse}} = 2 \times 10^{-4}\sqrt{d}$. Thus oracle dispersion includes graph/data variation, while learned dispersion measures initialization, batch order, and DSM corruption on fixed data. Results: Table 21.

Sachs observational benchmark. Use the 853-observation, 11-variable dataset and 17-edge graph from the GraN-DAG release (Sachs et al., 2005; Lachapelle et al., 2020). The files are `continuous/data1.npy`, `continuous/DAG1.npy`, and `sachs-header.npy` in `data/sachs.zip` at commit `d86ab5932573`. The supplement records the full commit and checksums. Standardize each column to unit variance, with no resampling or train/test split. SSTS, the masked-score control, both DiffAN variants, NOTEARS-MLP, and DAGMA-MLP use seeds 42–46; their dispersion reflects method randomness on fixed data. SCORE, DAS, and TOPIC are run once under deterministic configurations. SSTS, the control, SCORE, and DiffAN use common Lasso; DAS and TOPIC retain their public graph recovery; NOTEARS and DAGMA return graphs directly. For the latter two, EV is evaluated on one topological extension of the returned graph and is not invariant to that choice. Results: Table 6.

F Detailed Results for Linear Exact Recovery

Table 3 reports linear ANM evaluations ($N = 10000$). We evaluate topological extraction using the empirical precision matrix inverted from observational data and the population precision matrix derived from the structural equations.

Table 3: **Linear ANM recovery.** Edge violations recorded over Schur marginalization steps. The population matrix checks the algebra in Theorem A.2.

d	Empirical precision EV	Population precision EV
10	0	0 (Exact)
20	0	0 (Exact)
50	0	0 (Exact)
100	0	0 (Exact)
500	0	0 (Exact)
1000	0	0 (Exact)

Both matrices give zero EV at the tested dimensions. The population result checks the linear algebra. The empirical result applies to these runs at $N = 10000$; recovery for arbitrary finite samples would require a separate guarantee.

G Additional Theoretical Diagnostics

G.1 Scale-Normalized Diagonal Variance Diagnostic

In the unequal-noise chain setting, varying σ_i^2 modifies F_{ii} and can break the minimum-diagonal criterion. A true leaf has constant negative Hessian entry $H_{ll}(\mathbf{x}) = 1/\sigma_l^2$, giving zero sample-wise structural variance. Estimated absolute variance remains sensitive to scale and score-fitting error.

We examine whether normalizing by the squared mean helps in this setting, using the squared coefficient of variation (CV^2):

$$\hat{l} \in \arg\min_i CV^2(H_{ii}(\mathbf{X})), \qquad CV^2(H_{ii}(\mathbf{X})) := \frac{\mathrm{Var}_p(H_{ii}(\mathbf{X}))}{(\mathbb{E}_p[H_{ii}(\mathbf{X})])^2}. \tag{18}$$

Table 4 evaluates this diagnostic on a 5-node non-linear causal chain under homoscedastic ($\sigma_i \equiv 1.0$) and heteroscedastic ($\sigma_{\text{root}} = 5.0, \sigma_{\text{leaf}} = 0.2$) regimes. Its entries use empirical moments of $\hat{H}_{ii}(\mathbf{x}) = -\partial_i s_{\theta,i}(\mathbf{x})$, where $s_{\theta,i}$ is the i-th network output. The implementation adds 10^{-8} to the squared-mean denominator.

Table 4: **Unequal-noise chain diagnostic.** Evaluated on $X_1 \to \cdots \to X_5$, using one-based labels as in Section 2. The relative variance score selects the leaf in both regimes.

		Homoscedastic		Heteroscedastic		
Node	**Role**	**Mean**	**CV²**	**Mean**	**Variance**	**CV²**
X_1	Root	3.4629	1.7605	1.3369	16.0972	9.0064
X_2	Mid	2.9928	1.0626	10.5315	122.9990	1.1090
X_3	Mid	3.0887	1.1539	11.2526	144.2889	1.1395
X_4	Mid	2.8529	1.0561	51.7125	4371.3657	1.6347
X_5	Leaf	1.6575	0.0767	102.9850	565.3597	0.0533

In this chain, the expectation baseline selects the root under heteroscedasticity, and absolute variance is affected by scale. The CV^2 diagnostic selects the leaf in both regimes. A linear Gaussian model gives a counterexample to any general identification claim: all Hessian diagonals are constant, so their relative variances cannot distinguish leaves from non-leaves.

G.2 One-Step Correction and Failure of Sample-wise Recursion

Use the current-marginal setting $\mathcal{S}, l, \mathcal{R}$ of Proposition 3.2. The pivot $H^{\mathcal{S}}_{ll} = F^{\mathcal{S}}_{ll} = \sigma^{-2}$ is constant, and the correction needed to obtain the next expected matrix is

$$\mathbf{F}^{\mathcal{R}} - \mathrm{Schur}_l(\mathbf{F}^{\mathcal{S}}) = -\mathbf{C}^{\mathcal{S}}_l = -(F^{\mathcal{S}}_{ll})^{-1}\mathrm{Cov}_{p_{\mathcal{S}}}(\mathbf{H}^{\mathcal{S}}_{\mathcal{R}l}(\mathbf{X}_{\mathcal{S}})). \tag{19}$$

The empirical counterpart uses covariance normalized by the sample count to equal the difference of the two empirical operations. An arbitrary learned Hessian need not have a constant pivot, so substituting it in Eq. 19 does not retain an exact identity.

Equation 15 shows why one cannot repeat this argument on stored sample matrices: their Schur updates retain a residual-weighted second derivative and are not the next marginal Hessians. For example, take $X_1 = \epsilon_1$, $X_2 = X_1 + \epsilon_2$, and $X_3 = X_2^2 + \epsilon_3$, with independent standard normal noise. Eliminating leaf 3 gives

$$\mathrm{Schur}_3(\mathbf{H}(\mathbf{X})) = \begin{pmatrix} 2 & -1 \\ -1 & 1 - 2\epsilon_3 \end{pmatrix}, \qquad \mathbf{H}^{\{1,2\}}(X_1, X_2) = \begin{pmatrix} 2 & -1 \\ -1 & 1 \end{pmatrix}.$$

Eliminating node 2 from the first matrix yields $2-(1-2\epsilon_3)^{-1}$, whereas the true marginal information for X_1 is 1. The former has no ordinary expectation because of the pole at $\epsilon_3 = 1/2$. Repeated Schur updates of stored sample matrices are therefore not exact non-linear marginalization, even with the true joint score and a correct elimination order.

Table 5 compares two learned-matrix recursions. Both combine estimation error with an approximate recursive update, so their residual EV cannot be attributed solely to score estimation.

Table 5: **Sample-wise Schur recursion at $N = 5000$.** This single-seed comparison uses 1000 Hessian samples for the sample-wise method. Neither learned-matrix recursion is an exact non-linear marginalization reference.

	Block-SSTS		**Sample-wise Schur**	
Dimension (d)	**EV** $\downarrow$	$\mathbf{T}_{\mathrm{order}}$ **(s)**	**EV** $\downarrow$	$\mathbf{T}_{\mathrm{order}}$ **(s)**
50	0	0.15	0	0.10
100	2	0.05	4	0.71

G.3 Block Elimination Does Not Remove Aggregation Bias

Let $\mathcal{A}, \mathcal{B}$ be disjoint eliminated sets and $\mathcal{R}$ the remaining indices of a matrix $\mathbf{M}$. When the required pivots are invertible, define the block Schur operator by the middle expression below. The Schur quotient identity gives

$$\begin{aligned} \mathrm{Schur}_{\mathcal{B}}(\mathrm{Schur}_{\mathcal{A}}(\mathbf{M})) &= \mathbf{M}_{\mathcal{R}\mathcal{R}} - \mathbf{M}_{\mathcal{R}\mathcal{U}}\mathbf{M}^{-1}_{\mathcal{U}\mathcal{U}}\mathbf{M}_{\mathcal{U}\mathcal{R}} \\ &= \mathrm{Schur}_{\mathcal{U}}(\mathbf{M}), \qquad \mathcal{U} = \mathcal{A} \cup \mathcal{B}. \end{aligned}$$

To verify the identity, partition $\mathbf{M}_{\mathcal{U}\mathcal{U}}$ into its $\mathcal{A}, \mathcal{B}$ blocks and substitute its block inverse. The two terms are the successive elimination updates. Iteration proves the statement for any partition of a fixed eliminated set. Applied to a population SJIM, batching the same removals cannot repair its discrepancy from the true marginal SJIM. Adaptive selection, finite precision, and regularization must be examined separately. A reduction in update count is not evidence that the covariance term has decreased.

G.4 DSM Bias and the Initial Matrix Error

The conditional least-squares solution of Eq. 11 is

$$\mathbf{s}_\tau(\mathbf{y}) = \mathbb{E}[-\mathbf{Z}/\tau \mid \mathbf{X} + \tau\mathbf{Z} = \mathbf{y}] = \nabla_{\mathbf{y}} \log p_\tau(\mathbf{y}),$$

under the differentiation conditions for the Gaussian convolution (Vincent, 2011). Define $\mathbf{F}_\theta = \mathbb{E}_p[-\mathrm{sym}(\nabla \mathbf{s}_\theta)]$ and $\mathbf{F}_{\tau|p} = \mathbb{E}_p[-\nabla^2 \log p_\tau]$. Write $\hat{\mathbf{F}}_\theta$ for Eq. 7 to make its fitted-model dependence explicit. Then

$$\hat{\mathbf{F}}_\theta - \mathbf{F} = \underbrace{\hat{\mathbf{F}}_\theta - \mathbf{F}_\theta}_{\text{empirical averaging}} + \underbrace{\mathbf{F}_\theta - \mathbf{F}_{\tau|p}}_{\text{fitting and regularization}} + \underbrace{\mathbf{F}_{\tau|p} - \mathbf{F}}_{\text{smoothing bias}}.$$

The three matrix terms can be dependent, and their error magnitudes need not add. Reusing training data for the empirical average also prevents an automatic conditional-unbiasedness argument for the first term. At fixed $\tau > 0$, the smoothing term need not vanish as N increases. The experiments do not estimate all three population terms separately.

For example, if $p = \mathcal{N}(\mathbf{0}, \mathbf{\Sigma})$, then $\mathbf{F}_{\tau|p} = (\mathbf{\Sigma} + \tau^2 \mathbf{I}_d)^{-1}$ rather than $\mathbf{\Sigma}^{-1}$. This gives a diagnostic of smoothing bias without neural fitting. An alternative estimator is

$$\hat{\mathbf{F}}_O = \frac{1}{N} \sum_{n=1}^{N} \mathbf{s}_\theta(\mathbf{x}^{(n)}) \mathbf{s}_\theta(\mathbf{x}^{(n)})^\top.$$

It avoids differentiation and is positive semidefinite, but $\mathbf{F} = \mathbb{E}_p[\mathbf{s}\mathbf{s}^\top]$ requires the true score of p. For the ideal fixed-scale Gaussian DSM score, the outer-product expectation is $(\mathbf{\Sigma}+\tau^2\mathbf{I}_d)^{-1}\mathbf{\Sigma}(\mathbf{\Sigma}+\tau^2\mathbf{I}_d)^{-1}$, not $\mathbf{F}_{\tau|p}$. Section 5.3 compares both estimators from the same fit and held-out data. Their rankings by matrix error and EV differ; neither is uniformly preferable in the tested sweep.

H Scope Checks

The settings in Appendix E.5 check boundaries beyond the main comparisons. The linear precision test gives zero EV through $d = 1000$ (Table 3), consistent with the known Gaussian identity. The nonlinear high-dimensional matrix diagnostic gives zero EV for both known-mechanism constructions, but learned matrices retain EV 83.2, 131.6, and 116.2 at $N = 5000, 10000, 20000$ (Table 21). The two groups vary different sources of randomness; their difference is not a paired error decomposition, and the learned trend does not show that more samples alone resolve the errors.

The input-penalty and block-tolerance sweeps measure sensitivity, not the covariance term itself (Table 18). The cubic post-nonlinear setting changes both additivity and score regularity and gives poor recovery (Table 19). On Sachs, SSTS has TPR 0.176 ± 0.000 and EV above the random-order expectation. TOPIC, NOTEARS, and DAGMA have lower SHD in these runs (Table 6). The order errors precede pruning; the contributions of noise mismatch, scaling, and score fitting remain unresolved.

I Standard Sachs Results

The Sachs setting in Appendix E.5 tests transfer from synthetic ANMs to a fixed biological benchmark. Table 6 compares the specified complete pipelines; their pruning rules differ. Total time includes all stages needed to return a graph. Dispersion measures method randomness on the same 853 observations, not variation across datasets.

Table 6: **Standard observational Sachs benchmark ($N = 853$, $d = 11$, 17 reference edges).** Mean $\pm$ Std is reported for five-run methods. A uniformly random order has expected EV 8.5.

Method	Runs	EV ↓	SHD ↓	TPR ↑	FDR ↓	$\mathbf{T}_{\text{total}}$ (s) ↓
SSTS	5	11.4 ± 0.5	16.8 ± 0.8	0.176 ± 0.000	0.692 ± 0.026	2.88 ± 0.28
Masked score (matched)	5	12.4 ± 0.9	16.8 ± 1.1	0.165 ± 0.026	0.708 ± 0.038	2.32 ± 0.17
SCORE + Lasso	1	12.0	16.0	0.118	0.714	**1.49**
DAS	1	12.0	17.0	0.059	0.875	2.78
TOPIC	1	**6.0**	**12.0**	**0.412**	**0.364**	1.59
DiffAN (mask) + Lasso	5	10.0 ± 0.7	15.8 ± 0.4	0.071 ± 0.026	0.829 ± 0.064	4.24 ± 0.83
DiffAN (residue) + Lasso	5	8.8 ± 0.8	15.2 ± 1.1	0.141 ± 0.053	0.696 ± 0.092	8.73 ± 0.89
NOTEARS-MLP	5	6.8 ± 1.8	14.2 ± 0.4	0.224 ± 0.026	0.457 ± 0.064	11.10 ± 0.57
DAGMA-MLP	5	9.4 ± 0.5	13.4 ± 0.5	0.212 ± 0.032	0.400 ± 0.091	184.01 ± 30.30

The SSTS TPR corresponds to three recovered reference edges in every run. Variation in SHD comes from the number and orientation of the remaining estimated edges. Its five learned orders differ, but all place several downstream signaling variables before the measured variables labeled PKA or PLCγ, producing 11 or 12 edge violations. Every block is a singleton at $\gamma = 0.01$, so candidate merging cannot explain these errors. The repeated ordering errors suggest a mismatch between the learned diagonals and the reference ordering across these fits, beyond one unfavorable initialization.

J Synthetic Comparison Details

The common-pruning benchmark contains 75 successful trials; the matched-schedule setting contains 60. Public methods receive identical data and graph hashes at each dimension and seed. The two grids share SSTS runs with identical orders and structural metrics; these count as the same replicates. Every recorded EV was checked against its stored order and reconstructed graph.

Table 7: **Public ordering implementations with common Lasso pruning.** $N = 5000$, seeds 42–46. Mean $\pm$ sample standard deviation. SCORE and DAS have identical orders in all 15 paired settings.

d	Ordering	EV	SHD	TPR	FDR
20	SSTS	0.0 ± 0.0	0.0 ± 0.0	1.000 ± 0.000	0.000 ± 0.000
20	CaPS ordering	4.0 ± 1.2	6.6 ± 2.5	0.778 ± 0.074	0.315 ± 0.117
20	SCORE ordering	0.0 ± 0.0	0.0 ± 0.0	1.000 ± 0.000	0.000 ± 0.000
20	DAS ordering	0.0 ± 0.0	0.0 ± 0.0	1.000 ± 0.000	0.000 ± 0.000
20	DiffAN-masking	3.0 ± 1.0	7.0 ± 2.9	0.835 ± 0.053	0.310 ± 0.104
50	SSTS	0.0 ± 0.0	0.2 ± 0.4	1.000 ± 0.000	0.004 ± 0.008
50	CaPS ordering	4.2 ± 1.9	8.8 ± 8.6	0.916 ± 0.036	0.148 ± 0.113
50	SCORE ordering	0.2 ± 0.4	0.8 ± 1.8	0.996 ± 0.008	0.014 ± 0.032
50	DAS ordering	0.2 ± 0.4	0.8 ± 1.8	0.996 ± 0.008	0.014 ± 0.032
50	DiffAN-masking	7.2 ± 3.6	26.4 ± 21.8	0.857 ± 0.068	0.338 ± 0.189
100	SSTS	4.6 ± 2.4	12.6 ± 8.4	0.955 ± 0.022	0.110 ± 0.066
100	CaPS ordering	3.8 ± 1.3	6.6 ± 3.2	0.962 ± 0.014	0.064 ± 0.029
100	SCORE ordering	0.2 ± 0.4	0.8 ± 1.3	0.998 ± 0.004	0.008 ± 0.013
100	DAS ordering	0.2 ± 0.4	0.8 ± 1.3	0.998 ± 0.004	0.008 ± 0.013
100	DiffAN-masking	13.8 ± 4.0	52.6 ± 23.4	0.862 ± 0.032	0.363 ± 0.112

Table 8: **DAS pruning comparison.** The same public DAS order is used in both columns. Native pruning and common Lasso are different graph estimators. Seeds 42–46; mean $\pm$ sample standard deviation.

d	Native SHD	Native TPR	Lasso SHD	Lasso TPR
20	2.0 ± 1.7	0.937 ± 0.046	0.0 ± 0.0	1.000 ± 0.000
50	8.2 ± 4.5	0.875 ± 0.053	0.8 ± 1.8	0.996 ± 0.008
100	16.2 ± 5.1	0.883 ± 0.044	0.8 ± 1.3	0.998 ± 0.004

At $d = 100$, the paired difference (SSTS minus control) is $+0.7$ EV and -2.3 SHD. Descriptive 95% paired t intervals are $[-1.02, 2.42]$ and $[-8.32, 3.72]$. With ten pairs and no multiple-comparison adjustment, these intervals are not evidence of equivalence or of a stable SSTS accuracy advantage.

K Seed Sensitivity

To check whether the internal comparison depends on one set of generated problems, Table 10 separates seeds 42–46 from seeds 47–51. Hyperparameters are unchanged between groups. Both groups have slightly higher mean SSTS EV and lower mean SSTS SHD than the control. The

Table 9: **Internal masked-score control.** Ten paired seeds, 42–51, with common Lasso. Training schedules and evaluation counts match; parameter counts and losses do not. Mean ± sample standard deviation.

d	Method	EV	SHD	TPR
20	SSTS	0.0 ± 0.0	0.0 ± 0.0	1.000 ± 0.000
20	Masked control	1.0 ± 1.2	2.2 ± 2.7	0.955 ± 0.054
50	SSTS	0.0 ± 0.0	0.1 ± 0.3	1.000 ± 0.000
50	Masked control	2.0 ± 1.4	4.9 ± 3.6	0.960 ± 0.029
100	SSTS	3.8 ± 2.1	8.9 ± 7.0	0.962 ± 0.019
100	Masked control	3.1 ± 2.1	11.2 ± 9.2	0.969 ± 0.020

second group has lower errors for both methods, illustrating sensitivity to the sampled SCM and observations. These are new synthetic problems, not additional fits to a fixed dataset.

Table 10: **Seed groups at $d = 100$, $N = 5000$.** Matched-schedule setting; mean ± sample standard deviation. These are subsets of Table 9, not additional replicates.

Seeds	Method	EV	SHD	TPR
42–46	SSTS	4.6 ± 2.4	12.6 ± 8.4	0.955 ± 0.022
42–46	Masked control	3.8 ± 2.9	15.4 ± 11.7	0.963 ± 0.026
47–51	SSTS	3.0 ± 1.6	5.2 ± 2.2	0.969 ± 0.016
47–51	Masked control	2.4 ± 0.9	7.0 ± 2.7	0.974 ± 0.011

L Runtime and Memory Measurements

We time training (T_{rep}), ordering (T_{order}), and common pruning separately to distinguish ordering cost from total runtime (T_{total}). GPU stage boundaries are synchronized, and total time includes setup within the trial. SSTS ordering comprises one Jacobian aggregation and all Schur updates. The control evaluates masked Jacobians throughout its recursion. Each trial uses a separate process; the stage measurements retain library-initialization costs where they occur.

Table 11: **Internal-control runtime in seconds.** Ten seeds, no discarded trials. Mean ± sample standard deviation and median are both shown because the records have long tails.

d	Method	T_{order}	Median	T_{total}	Median
20	SSTS	0.996 ± 1.623	0.178	6.88 ± 1.64	5.94
20	Masked control	0.061 ± 0.012	0.055	7.47 ± 3.66	6.12
50	SSTS	1.061 ± 2.129	0.147	7.73 ± 2.02	6.98
50	Masked control	0.120 ± 0.008	0.118	16.72 ± 28.48	6.79
100	SSTS	0.797 ± 1.314	0.177	11.51 ± 5.68	9.30
100	Masked control	0.456 ± 0.015	0.453	21.14 ± 26.47	8.79

At $d = 100$, the ordering mean is 0.797 seconds for SSTS and 0.456 for the control, whereas the medians are 0.177 and 0.453 seconds. Total-time means are 11.51 and 21.14 seconds, but the medians are 9.30 and 8.79 seconds. These reversals preclude a single stable speedup factor. At $d = 20, 50$, the median SSTS ordering time is higher than the control's.

The dodiscover SCORE/DAS paths use CPU NumPy kernel estimation. Their long runtimes do not constitute a matched-GPU comparison. CaPS, SSTS, and DiffAN use GPU routines, but differ in estimator and training budget. CaPS is faster end to end than SSTS at $d = 20$ in these records. Its peak PyTorch allocation at $d = 100$ is 28.163 GiB versus 0.817 GiB for SSTS. The final CaPS trials all complete; earlier shared-memory out-of-memory errors do not establish failure on an idle device.

Table 12: **Recorded public-implementation runtime and GPU allocation.** Five seeds; total seconds are mean ± sample standard deviation and median. DAS total includes its native pruning and an additional common-Lasso evaluation. GPU GiB denotes the maximum per-process PyTorch allocated memory, not board usage or host memory. CPU marks the NumPy kernel path.

d	Method	T_{total}	Median	GPU GiB
20	SSTS	6.20 ± 0.23	6.21	0.171
20	CaPS ordering	3.80 ± 0.10	3.86	5.809
20	SCORE ordering	322.01 ± 9.95	324.48	CPU
20	DAS ordering	519.67 ± 12.90	524.92	CPU
20	DiffAN-masking	13.14 ± 0.82	13.42	0.204
50	SSTS	7.05 ± 0.27	7.10	0.355
50	CaPS ordering	9.34 ± 0.47	9.23	14.191
50	SCORE ordering	895.14 ± 28.11	909.20	CPU
50	DAS ordering	1791.48 ± 845.83	1418.25	CPU
50	DiffAN-masking	27.70 ± 14.49	18.10	0.397
100	SSTS	11.72 ± 4.43	10.46	0.817
100	CaPS ordering	18.25 ± 0.41	18.42	28.163
100	SCORE ordering	2585.57 ± 968.66	2158.54	CPU
100	DAS ordering	3758.88 ± 1173.93	3230.80	CPU
100	DiffAN-masking	42.47 ± 16.11	35.10	0.764

The experiments ran on a shared server. The records omit continuous occupancy traces and the effective CPU/BLAS thread count; exclusive device use was not verified. Successful retries complete the accuracy grid, but the runtimes remain shared-server measurements. For example, two identical SSTS $d = 100$, seed-42 runs return the same order but take 9.05 and 19.35 seconds in total. We retain both measurements. The records cannot separate hardware competition, host scheduling, and initialization costs.

M Paired Learned-Matrix Diagnostics

The paired-matrix setting is specified in Appendix E.3. Table 13 reports both estimators and the reference. The moment and full-Hessian references differ by a mean relative Frobenius norm of about 2.1×10^{-4} on these samples. The moment reference gives zero EV, but the learned estimates need not preserve that order.

Table 13: **Paired estimators at $d = 100$.** Mean ± sample standard deviation over five seeds. Error is the relative Frobenius norm against the reference moment matrix. Its EV is zero for all five distinct SCMs; the same oracle result is reused across scales.

τ	Jacobian EV	Outer EV	Jacobian error	Outer error
0.05	58.8 ± 13.8	73.0 ± 6.8	0.990 ± 0.007	0.994 ± 0.005
0.10	17.2 ± 2.8	6.0 ± 2.0	0.286 ± 0.038	0.412 ± 0.047
0.20	5.6 ± 2.5	7.4 ± 3.0	0.456 ± 0.045	0.667 ± 0.052

At $\tau = 0.05$, Jacobian EV is 58.8, above the average random-order expectation of 50.2 over these graphs. The smaller scale also increases the relative strength of the fixed input penalty: dividing the regularized objective by τ^2 gives a penalty coefficient $\lambda_{\text{sparse}}/\tau^2$. Smoothing and training effects vary together in this sweep, so the failure cannot be attributed to either one alone.

All 15 Jacobian aggregates are indefinite. Their minimum eigenvalues range from -0.212 to -0.007. At $\tau = 0.20$, they range from -0.028 to -0.007, so symmetry and a fixed ridge cannot be treated as positive-definiteness guarantees. The score outer product is mathematically positive semidefinite; one small-scale stored matrix has a minimum eigenvalue near -1.9×10^{-8}, consistent with floating-

point accumulation. It is not a uniformly better ordering statistic: it improves EV at $\tau = 0.10$ but not at the other tested scales.

Forced-path diagnostic. On the forced paths defined in Appendix E.3, none of the $15 \times 2 \times 32 = 960$ correlated states passes the empirical diagonal-gap test. At $\tau = 0.10, 0.20$, the diagonal minimizer is still a true leaf at every recorded state. The condition is conservative on these paths. The empirical reference and prescribed partial path do not certify a complete learned order or the regularized block procedure.

Algebra checks. Gauss–Hermite quadrature reproduces the common matrix $\mathrm{diag}(5, 5, 1)$ in Theorem 3.1. For M_A, the signed correction $\mathbf{F}^{\{1,2\}} - \mathrm{Schur}_3(\mathbf{F})$ is $\mathrm{diag}(0, -4)$, the negative of the covariance term in Proposition 3.2. The fixed-set block/scalar Schur identity agrees to about 9×10^{-16}. These calculations check implementations; the proofs establish the identities. The non-existence of the inverse-pivot expectation in stored-sample recursion is an analytic conclusion, not a Monte Carlo estimate.

N High-Dimensional Graph-Recovery Setting

Table 14 reports the graph-recovery setting from Appendix E.5, including Lasso pruning. This is a separate SSTS stress test, not a public-baseline comparison. Dispersions are population standard deviations, as in the source summaries.

Table 14: **High-dimensional graph recovery with SSTS ($N = 5000$).** Evaluated over 5 random seeds (Mean $\pm$ Std).

Dimension (d)	**EV ↓**	**SHD ↓**	**TPR ↑**	**Block Iters ↓**	**$\mathbf{T}_{\mathrm{total}}$ (s)**
200	8.6 ± 1.2	14.4 ± 3.3	0.957 ± 0.004	136.0 ± 2.9	11.54 ± 0.85
500	13.6 ± 2.4	26.4 ± 6.1	0.971 ± 0.007	266.0 ± 4.2	46.07 ± 1.37

O Sample-Size Setting

In the sample-size setting at $d = 100$ (Table 15), EV decreases with sample size, while SHD is not monotone. The sweep measures the complete fitted procedure; it does not isolate estimation variance or establish a sample-complexity rate. Its population standard deviations are kept separate from the main comparison.

Table 15: **Sample-size sweep at $d = 100$.** Evaluated over 5 random seeds (Mean $\pm$ Std).

Samples (N)	**EV ↓**	**SHD ↓**	**TPR ↑**	**Block Iters**	**$\mathbf{T}_{\mathrm{total}}$ (s)**
500	14.8 ± 3.4	51.0 ± 13.8	0.851 ± 0.033	90.6 ± 1.2	2.94 ± 0.94
1000	10.0 ± 3.3	34.0 ± 15.0	0.901 ± 0.028	88.6 ± 5.2	2.68 ± 0.04
2000	8.8 ± 1.3	20.6 ± 6.4	0.911 ± 0.020	87.0 ± 2.1	3.59 ± 0.06
5000	4.6 ± 2.2	12.6 ± 7.5	0.955 ± 0.019	76.8 ± 3.3	6.16 ± 0.10
10000	1.2 ± 1.2	2.2 ± 2.4	0.986 ± 0.015	59.2 ± 6.2	10.12 ± 0.23
20000	0.8 ± 0.7	5.6 ± 5.0	0.969 ± 0.030	50.4 ± 5.4	18.22 ± 0.12

P Order-Source Control

Can pruning produce errors even when the order is correct? Table 16 compares four order sources under the same Lasso at $d = 100$. The true-order row supplies a valid order from the generating graph; the random row uses an unrestricted permutation; the empirical-precision row applies Schur ordering to an inverse sample covariance. The true order still has nonzero mean SHD after pruning. SSTS has fewer EV than random and empirical-precision ordering. Differences in SHD are not an additive error decomposition because changing the order changes the regression candidates.

Table 16: **Order and pruning decomposition at $d = 100$ ($N = 5000$).** Evaluated over 5 random seeds (Mean $\pm$ Std).

Order Source	EV ↓	SHD ↓	TPR ↑	$\mathbf{T}_{\text{total}}$ (s)
True order	0.0 ± 0.0	$\mathbf{1.6 \pm 3.2}$	$\mathbf{0.984 \pm 0.031}$	2.39 ± 0.08
Learned SSTS	4.6 ± 2.2	12.6 ± 7.5	0.955 ± 0.019	6.50 ± 0.91
Random order	46.2 ± 5.1	219.4 ± 46.7	0.536 ± 0.022	2.39 ± 0.12
Empirical precision	61.2 ± 6.9	193.6 ± 42.9	0.386 ± 0.062	3.16 ± 0.60

Q Checkpoint-Averaging Setting

We average the SJIM over the last K consecutive epoch-end checkpoints of one training run, with $K \in \{1, 3, 5\}$ for $d = 100$ and $K \in \{1, 3\}$ for $d = 500$. EV, SHD, and TPR are unchanged across the tested values of K within each setting (Table 17).

Table 17: **Late-checkpoint SJIM averaging.** Mean $\pm$ Std over seeds. $K = 1$ is the standard SSTS run; $K > 1$ averages the SJIM over the last K checkpoints before Schur extraction.

Setting	Method	EV ↓	SHD ↓	TPR ↑	$\mathbf{T}_{\text{total}}$ (s)
Main $d = 100$	SSTS ($K = 1$)	4.6 ± 2.2	12.6 ± 7.5	0.955 ± 0.019	6.90 ± 1.11
Main $d = 100$	SJIM avg. ($K = 3$)	4.6 ± 2.2	12.6 ± 7.5	0.955 ± 0.019	6.85 ± 1.00
Main $d = 100$	SJIM avg. ($K = 5$)	4.6 ± 2.2	12.6 ± 7.5	0.955 ± 0.019	6.85 ± 1.00
Held-out $d = 100$	SSTS ($K = 1$)	3.0 ± 1.4	5.8 ± 1.5	0.962 ± 0.008	6.63 ± 0.81
Held-out $d = 100$	SJIM avg. ($K = 3$)	3.0 ± 1.4	5.8 ± 1.5	0.962 ± 0.008	6.62 ± 0.75
Held-out $d = 100$	SJIM avg. ($K = 5$)	3.0 ± 1.4	5.8 ± 1.5	0.962 ± 0.008	6.61 ± 0.77
$d = 500$	SSTS ($K = 1$)	12.3 ± 2.4	23.3 ± 5.2	0.974 ± 0.006	47.57 ± 1.19
$d = 500$	SJIM avg. ($K = 3$)	12.3 ± 2.4	23.3 ± 5.2	0.974 ± 0.006	48.04 ± 1.03

R Input-Penalty and Block-Tolerance Settings

Figure 4 and Table 18 report the two sweeps specified in Appendix E.5. Mean SHD varies little across the tested input penalties relative to its dispersion. At $d = 500$, increasing the block tolerance reduces update count, but mean SHD is non-monotone. Neither sweep measures the covariance gap separately, so changes in SHD cannot be assigned to that gap.

S Mechanism-Shift Setting

We compare additive, multiplicative-noise, and post-nonlinear mechanisms. For each non-root node, write $u_i = \sum_{j \in \mathrm{pa}(i)} w_{ji} x_j$ and draw $\epsilon_i \sim \mathcal{N}(0, 0.1^2)$ independently. The three generators are

$$\begin{aligned} \text{ANM}: &\quad x_i = \tanh(u_i) + \epsilon_i, \\ \text{MNM}: &\quad x_i = \tanh(u_i) + \big(0.5 + 1.5/(1 + e^{-u_i})\big)\epsilon_i, \\ \text{PNL}: &\quad x_i = \big(\tanh(u_i) + \epsilon_i\big)^3. \end{aligned}$$

Roots remain $x_i = \epsilon_i$ in all three settings. MNM makes conditional noise scale depend on parents. PNL also applies a cubic outer transformation, whose derivative vanishes at zero; it is not a smooth-density control that changes only additivity. Table 19 uses the mechanism-shift configuration from Appendix E.5, with DSM scale 0.05 and input penalty 5×10^{-4}, not the main SSTS configuration.

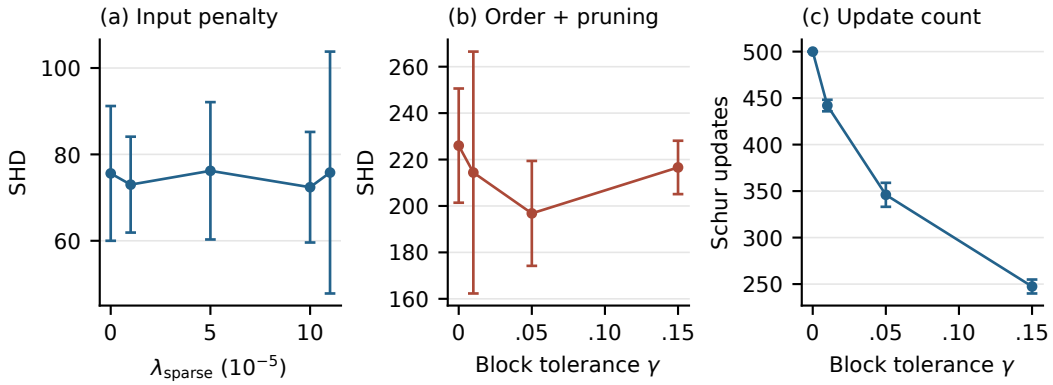


Figure 4: **Input-penalty and block-tolerance sweeps.** (a) Input penalty at $d = 100$, $N = 500$. (b,c) SHD and update counts for block tolerance at $d = 500$, $N = 5000$, on separate axes. Error bars are population standard deviations over five seeds. The plot uses the same rounded summaries as Table 18, not additional trials.

Table 18: **Dual-regime ablation studies.** Evaluated over 5 random seeds (Mean ± Std).

(a) Exp A: Group Lasso penalty (λ_{sparse}) in a low-sample regime ($d = 100, N = 500, \gamma = 0.05$).

λ_{sparse}	EV ↓	SHD ↓	TPR ↑	Extraction time (s) ↓
0.0 (Baseline)	22.6 ± 4.0	75.6 ± 15.6	0.78 ± 0.04	0.07 ± 0.04
1.0×10^{-5}	23.8 ± 3.5	73.0 ± 11.1	0.77 ± 0.04	0.05 ± 0.00
5.0×10^{-5}	24.0 ± 5.9	76.2 ± 15.9	0.77 ± 0.07	0.05 ± 0.00
1.0×10^{-4} (Best in sweep)	$\mathbf{21.6 \pm 2.2}$	$\mathbf{72.4 \pm 12.8}$	$\mathbf{0.79 \pm 0.03}$	$\mathbf{0.05 \pm 0.00}$
1.1×10^{-4}	21.8 ± 5.9	75.8 ± 28.0	$\mathbf{0.79 \pm 0.05}$	$\mathbf{0.05 \pm 0.00}$

(b) Exp B: Block tolerance (γ) at $d = 500$ ($N = 5000, \lambda_{\text{sparse}} = 0.0$).

γ	EV ↓	SHD ↓	TPR ↑	Block iterations ↓	Extraction time (s) ↓
0.00 (Sequential)	86.0 ± 6.4	226.0 ± 24.6	0.83 ± 0.01	500.0 ± 0.0	0.42 ± 0.00
0.01	82.0 ± 10.2	214.4 ± 52.1	$\mathbf{0.84 \pm 0.01}$	442.0 ± 6.2	0.39 ± 0.01
0.05 (Best in sweep)	$\mathbf{77.4 \pm 4.8}$	$\mathbf{196.8 \pm 22.6}$	$\mathbf{0.84 \pm 0.01}$	346.0 ± 12.9	0.33 ± 0.01
0.15	86.4 ± 3.3	216.6 ± 11.5	0.83 ± 0.01	$\mathbf{247.4 \pm 7.5}$	$\mathbf{0.27 \pm 0.01}$

T Order-Scaling Setting

We evaluate Block-SSTS order extraction on high-dimensional DAGs up to $d = 1000$ ($N = 5000$). Graph-level recovery at $d = 500$ is reported in Appendix N; at $d = 1000$ we report order-level metrics only.

Order-Level Scaling. At $d = 1000$, Block-SSTS uses 645.0 ± 4.2 block inversions and obtains normalized EV 0.091 ± 0.005, compared with a random-order baseline of about 0.501.

Every valid order has zero EV. For a fixed nonempty graph, a uniformly random permutation has expected NEV $1/2$ because each edge is reversed with probability $1/2$.

Dimension Dependence. Normalized EV is below the random-order expectation at $d = 1000$, but increases from 0.024 at $d = 500$ to 0.091 at $d = 1000$. This trend at fixed $N = 5000$ does not separate score fitting, smoothing bias, and elimination error.

Table 19: **Failure mode analysis: sensitivity to mechanism shifts ($d = 50, N = 5000$).** Evaluated over 5 random seeds (Mean ± Std).

Data-Generating Mechanism	Edge Violations ↓	SHD ↓	TPR ↑
ANM (Additive Noise, Control)	1.6 ± 1.0	3.6 ± 2.9	0.95 ± 0.04
MNM (Multiplicative Noise)	$\mathbf{0.0 \pm 0.0}$	$\mathbf{0.4 \pm 0.8}$	$\mathbf{0.99 \pm 0.02}$
PNL (Post-Nonlinear Model)	47.0 ± 2.4	94.4 ± 8.4	0.03 ± 0.02

Table 20: **Order-level scaling of Block-SSTS ($N = 5000$).** $d = 50, 100$ use 5 seeds; $d = 500, 1000$ use 3 seeds. NEV is EV divided by the number of true edges. The random baseline samples unrestricted permutations, not valid topological orders.

Dimension (d)	EV ↓	NEV ↓	Random NEV	Block Iters	$\mathbf{T}_{\text{order}}$ (s)
$d = 50$	0.0 ± 0.0	0.000 ± 0.000	0.499 ± 0.002	26.4 ± 3.1	0.02 ± 0.00
$d = 100$	4.6 ± 2.2	0.045 ± 0.019	0.500 ± 0.001	76.8 ± 3.3	0.04 ± 0.00
$d = 500$	12.3 ± 2.4	0.024 ± 0.005	0.501 ± 0.001	263.7 ± 4.0	0.35 ± 0.09
$d = 1000$	90.3 ± 7.8	0.091 ± 0.005	0.501 ± 0.001	645.0 ± 4.2	0.95 ± 0.01

U High-Dimensional SJIM Diagnostics

Table 21 reports the high-dimensional matrix setting from Appendix E.5. The known-mechanism group varies SCM/data seeds, while the learned group varies training seeds on one fixed SCM and nested dataset. Their difference is not a paired decomposition of estimation and elimination errors.

Table 21: **Detailed oracle–learned diagnostic at $d = 1000$ and $\gamma = 0.01$.** Oracle means use five SCM and data seeds; learned means use five network seeds with SCM and data seed 42 fixed.

	Oracle moment		Oracle full Hessian		Learned SJIM
N	EV	Block Iters	EV	Block Iters	EV / Block Iters
5,000	0.0 ± 0.0	86.6 ± 15.2	0.0 ± 0.0	85.2 ± 15.1	83.2 ± 5.2 / 638.2 ± 7.7
10,000	0.0 ± 0.0	86.6 ± 15.4	0.0 ± 0.0	85.8 ± 13.3	131.6 ± 6.5 / 655.6 ± 6.5
20,000	0.0 ± 0.0	86.4 ± 15.6	0.0 ± 0.0	85.6 ± 17.1	116.2 ± 3.2 / 700.0 ± 4.3
100,000	0.0 ± 0.0	85.8 ± 15.1	0.0 ± 0.0	85.6 ± 15.0	–

With $\gamma = 0$, both oracle constructions also give zero EV at every sample size and use about 566 updates. At $\gamma = 0.05$, the mean update count falls to about 29, but one of the five runs has one edge violation in some conditions. The $\gamma = 0.01$ setting reduces the number of oracle updates while retaining zero edge violations in these runs. The learned matrices do not exhibit the same block structure: they require 638–700 updates and retain non-zero EV.

The learned trend is not monotone in sample size and occurs across the five training seeds. Separate checks with a fixed update budget, no input Group Lasso, and $\gamma \in \{0.005, 0.01, 0.02, 0.05\}$ retain the deterioration from $N = 5000$ to the larger samples. Its cause remains unresolved, but the trend persists beyond one training seed or block tolerance.

V Capacity-Constrained Setting

The capacity-constrained setting uses micro-batch Jacobians, scalar rank-one Schur updates, and capped hidden width through $d = 5000$. Its T_{Schur} timer starts after Jacobian construction, unlike the complete ordering time in the main comparison.

This configuration completes at the tested dimensions with peak allocation below 1 GiB and substantial order errors. The runs neither isolate the effect of capacity capping nor compare other methods on the same hardware.

Table 22: **Capacity-constrained extraction ($N = 5000$).** Five seeds. T_{Schur} excludes Jacobian construction. Memory is per-process peak PyTorch allocation in GiB, not total device usage.

d	**EV**	T_{rep} (s)	T_{Schur} (s)	**Peak GPU GiB**
$d = 500$	210.6 ± 71.3	3.72 ± 0.69	0.145 ± 0.013	0.95
$d = 1000$	301.0 ± 16.9	3.46 ± 0.07	0.378 ± 0.002	0.97
$d = 5000$	1994.6 ± 669.2	3.65 ± 0.04	6.570 ± 0.015	0.93

W Topology and Noise-Shift Setting

We vary the graph topology and noise distribution at $d = 50$ ($N = 5000$):

1. **Topological shifts:** We compare Erdős-Rényi (ER) graphs with scale-free (SF) networks, whose power-law degree distribution gives a different pattern of connectivity.
2. **Distributional shifts:** We replace Gaussian noise with Exponential and Gumbel distributions. Both violate the Gaussian assumption even after centering.

Table 23: **Topology and noise distribution shifts ($d = 50, N = 5000$).** Evaluated over 5 random seeds (Mean $\pm$ Std).

Graph Topology	**Noise Distribution**	**Edge Violations** $\downarrow$	**SHD** $\downarrow$	**TPR** $\uparrow$
Erdős-Rényi (ER)	Gaussian	1.6 ± 1.0	3.6 ± 2.9	0.95 ± 0.04
	Exponential	1.6 ± 1.4	5.4 ± 6.0	0.97 ± 0.02
	Gumbel	$\mathbf{0.2 \pm 0.4}$	$\mathbf{0.8 \pm 1.0}$	$\mathbf{1.00 \pm 0.01}$
Scale-Free (SF)	Gaussian	2.2 ± 1.9	21.4 ± 14.4	0.97 ± 0.01
	Exponential	1.4 ± 1.7	26.6 ± 24.1	0.96 ± 0.04
	Gumbel	$\mathbf{0.2 \pm 0.4}$	$\mathbf{16.8 \pm 21.4}$	$\mathbf{0.98 \pm 0.04}$

EV remains low in the tested topology/noise settings. Scale-free graphs have higher SHD than ER graphs despite comparable EV and TPR. The aggregate metrics do not locate the extra errors in hub neighborhoods or establish robustness to arbitrary non-Gaussian noise.