# TREE DEPTH FOR MASKED DIFFUSION: EXACT UNMASKING IN FERROMAGNETIC ISING MODELS

**Anonymous authors**
Paper under double-blind review

## ABSTRACT

Masked models predict every unrevealed coordinate in one evaluation, but independent parallel commitments can distort the joint law even when the one-coordinate conditionals are exact. Block size or marginal confidence alone does not determine which variables are conditionally independent given prior reveals, or the minimum number of rounds for exact sampling. To characterize both from the dependency graph, we study ferromagnetic Ising models, classical graphical models from statistical physics with local attractive interactions. We prove that independent parallel sampling is exact if and only if removing previously revealed vertices leaves no path between any two variables sampled together. Layers of a minimum-height elimination forest attain the minimum round count for exact fixed commit-once sampling, equal to tree depth, the minimum forest height. Below this round threshold, every fixed schedule incurs positive forward Kullback–Leibler divergence, which measures distributional mismatch; for learned conditionals, an exact decomposition separates this factorization cost from model error. With planner-aware training, our schedules improve conditional prediction quality at matched model-call budgets and better preserve dependencies in generated samples. In-distribution comparisons with dependency-guided baselines show lower prediction error at comparable or shorter end-to-end decoding times.

## 1 INTRODUCTION

Masked generative models predict all unrevealed variables in a single evaluation (Chang et al., 2022; Nie et al., 2025). However, the model outputs one-coordinate conditionals rather than an explicit joint conditional distribution. Each prediction uses the same revealed context, without conditioning on values sampled at other positions in the current round. Drawing independently from these outputs forms a product distribution, even if the selected variables remain dependent under that context. This can distort the target joint law even when each one-coordinate conditional is exact (Azangulov et al., 2026; Zhang et al., 2026). Block selection must therefore account for dependencies that block size or marginal confidence alone cannot reveal. Sequential conditioning avoids this factorization error but forgoes parallelism. Figure 1 illustrates the exact-round threshold set by the dependency graph.

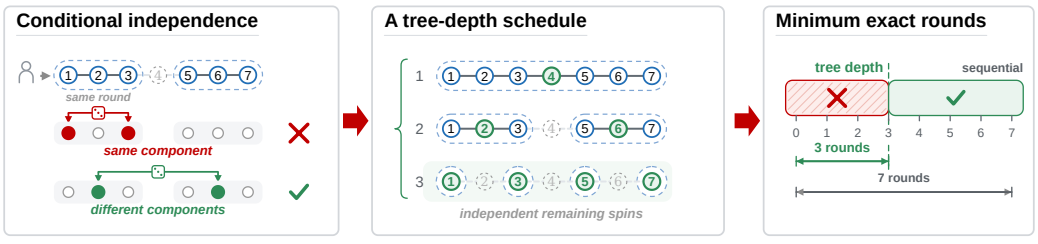


Figure 1: **Tree depth and the round complexity of exact parallel sampling.** The left panel compares parallel draws within and across the remaining components of a ferromagnetic Ising graph. In the middle, sampling uses exact single-spin conditionals and independent within-round draws, keeping sampled values fixed. The right panel marks the minimum round count for exact fixed schedules.

The central problem is to determine which variables can be committed together without changing the target joint law, and how few rounds suffice. Recent theory characterizes schedule divergence,



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.

separates conditional-model from factorization error, and optimizes block sizes through information profiles (Chen et al., 2026; Lavenant & Zanella, 2025); other methods learn or score position-specific reveal orders (Kim et al., 2025; Peng et al., 2025). We study sampling with exact one-coordinate conditionals, drawing each round's selected variables independently and never revising their sampled values. For ferromagnetic Ising targets, our method identifies exact parallel blocks by removing previously revealed vertices and choosing at most one variable from each remaining connected component.

For ferromagnetic Ising models, graphical separation gives independence across residual components (Lauritzen, 1996; Wainwright & Jordan, 2008), and classical faithfulness results give persistent dependence within each component after every reveal (Fallat et al., 2017). This property, called boundary-wise residual faithfulness, makes a block exact precisely when it meets each residual component at most once. Applying this component rule recursively yields an elimination forest whose depth layers form an exact parallel schedule (Figure 2 in Section 3). We prove that tree depth, the minimum forest height (Nešetřil & Ossona de Mendez, 2012), is the minimum exact round count for fixed schedules on arbitrary interaction graphs. Every trajectory of an exact deterministic value-adaptive planner obeys the same lower bound. The KL decomposition separates factorization from learned-conditional error (Lavenant & Zanella, 2025). Our experiments corroborate the exact-round threshold and the error decomposition. With masks drawn from the planned reveal order (planner-aware training; cf. Peng et al., 2026), the resulting schedules also improve parallel prediction quality with fewer decoding rounds.

Our contributions are as follows:

1. **Exact round characterization.** We identify tree depth with the minimum exact fixed-schedule round count for ferromagnetic Ising models on arbitrary interaction graphs, attained by elimination-forest layers.
2. **Lower bound under adaptation.** We prove that using observed values to choose future blocks cannot beat tree depth on any trajectory of an exact deterministic commit-once planner.
3. **Shorter schedules and estimated graphs.** We derive positive forward-KL lower bounds for shorter fixed schedules and transfer exactness to planning graphs containing all true edges, with bounded round inflation.
4. **Parallel prediction quality.** We evaluate prediction quality and decoding cost together, with shared adapters and training-objective controls.

Related work and the sampling model appear in Sections 2 and 3; Section 4 gives the theory, Section 5 evaluates quality and latency, and Section 6 concludes.

## 2 Related Work

**Factorized masked generation and dependence modeling.** Discrete diffusion replaces left-to-right order with denoising or masked prediction (Austin et al., 2021; Campbell et al., 2022; Nie et al., 2025), but factorized rounds sample selected positions independently. Within-step models and learned forward processes address that dependence (Liu et al., 2025; Bartosh et al., 2026). These methods change the within-step distribution or noising process; scheduling instead changes which variables are committed together.

**Schedule-dependent error.** Conditional total correlation measures within-block dependence (Cover & Thomas, 2006; Polyanskiy & Wu, 2025). Schedule analyses compare block sizes, orders, or KL directions (Li & Cai, 2025; Chen et al., 2026; Zhang et al., 2026); Lavenant & Zanella (2025, Proposition 2) separate model and factorization error. This distinguishes learned-conditional inaccuracies from dependence costs that can remain even with a perfect denoiser. For fixed-order random walks, Bansal et al. (2026, Section 4) use Markov separators to construct exact logarithmic-depth bisection samplers under perfect conditionals.

**Reveal orders, dependency scores, and corrections.** Methods learn reveal paths or dependency scores (Kim et al., 2025; Peng et al., 2025; Ringel et al., 2026; Zhou et al., 2026; Azangulov et al., 2026). Other approaches optimize unmasking policies (Hong et al., 2026), adapt inference block sizes (Lu et al., 2026), align training and sampling paths (Peng et al., 2026), or remask predictions (Wang et al., 2025; Frkovic et al., 2026). The revealed context changes with the schedule, so training–sampling alignment can affect conditional-model accuracy.

**Graphical dependence and tree depth.** Graphical separation gives independence across residual components (Lauritzen, 1996; Wainwright & Jordan, 2008); multivariate total positivity of order two (MTP2), a positive-dependence property, supplies the qualitative Ising premise through faithfulness (Fallat et al., 2017, Theorem 6.1). Closure under conditioning preserves this positive-dependence premise after each reveal (Fallat et al., 2017, Proposition 3.4). Our shortest-path proof quantifies this dependence. Tree depth is minimum elimination-forest height (Nešetřil & Ossona de Mendez, 2012), with exact and separator-based solvers (Reidl et al., 2014; Trimble, 2020b;a); our layered-separator elimination (LSE) construction verifies legality. Under graph-recovery assumptions, Ising model selection supplies edge coverage (Ravikumar et al., 2010); latent-tree and prediction-centric learning target distributional accuracy (Kandiros et al., 2023; Boix-Adserà et al., 2021).

**The missing gap and our motivation.** Separator constructions show that exact parallel sampling is possible; we ask how many sequential rounds the interaction graph makes unavoidable. Revealing a separator can change which remaining variables may be sampled together. The challenge is to relate these local choices to the minimum exact round count. For ferromagnetic Ising targets on arbitrary interaction graphs, we identify tree depth with the minimum exact fixed-schedule round count through matching constructions and lower bounds. The lower bound also holds for every trajectory of an exact deterministic value-adaptive planner, so using sampled values to choose future blocks cannot improve the optimum.

## 3 Factorized Unmasking and Residual Graphs

A committed block is exact precisely when it factors after the previous reveals. Forward KL measures violations of this condition, while the residual graph determines when factorization holds.

### 3.1 Factorized Sampling and Blockwise KL

A factorized sampler replaces a joint block conditional by the product of its coordinate conditionals. To define it, let $V$ be a finite set of coordinates, let $\mathcal{X}_i$ be a finite alphabet for coordinate $i$, and write $\mathcal{X} = \prod_{i \in V} \mathcal{X}_i$. We consider a random vector $X = (X_i)_{i \in V} \sim \mu$, where $\mu$ has full support, meaning $\mu(x) > 0$ for every $x \in \mathcal{X}$. Lowercase $x \in \mathcal{X}$ denotes a realization. For $A \subseteq V$, the notations $X_A = (X_i)_{i \in A}$ and $x_A$ denote the corresponding random subvector and its realization.

With this notation, a schedule groups variables into rounds and never changes a committed value. Formally, a fixed monotone schedule is an ordered partition $\Pi = (P_1, \dots, P_r)$ of $V$ into nonempty blocks. Before round $t$, the set $P_{<t} := \bigcup_{s<t} P_s$ has already been revealed. One conditional-model evaluation returns the one-coordinate conditionals for all unrevealed coordinates. The schedule selects the block $P_t$, draws its coordinates independently from those conditionals, and never revisits them. Under this interface, the $r = |\Pi|$ blocks require $r$ model evaluations, regardless of how many coordinates a block contains. Write $\nu_\Pi$ for the output law with exact one-coordinate conditionals; its product form is equation 5 in Appendix A. The independent draws within a block need not reproduce the target block conditional.

For probability laws $p$ and $q$ on the same finite space, write $D_{\mathrm{KL}}(p\|q)$ for the Kullback–Leibler divergence in nats, and write $\mathbb{E}_\mu$ for expectation under $\mu$. Forward KL places the target law first and, for a fixed schedule, decomposes by round.

**Proposition 1** (Blockwise KL Decomposition)**.** *For every full-support finite target $\mu$ and every ordered partition $\Pi$,*

$$D_{\mathrm{KL}}(\mu\|\nu_\Pi) = \sum_{t=1}^{r} \mathbb{E}_\mu D_{\mathrm{KL}}\left(\mu(X_{P_t} \mid X_{P_{<t}}) \,\middle\|\, \prod_{i \in P_t} \mu(X_i \mid X_{P_{<t}})\right). \tag{1}$$

*Each summand is the conditional total correlation of $X_{P_t}$ given $X_{P_{<t}}$. Hence $\nu_\Pi = \mu$ if and only if each block's coordinates are mutually independent given its revealed past.*

**Proof sketch.** The chain rule splits $\log(\mu(X)/\nu_\Pi(X))$ round by round. Each round compares the block conditional with the product of its coordinate conditionals, and taking expectations yields the displayed sum (Lavenant & Zanella, 2025; Chen et al., 2026). The full expansion is in Appendix A.

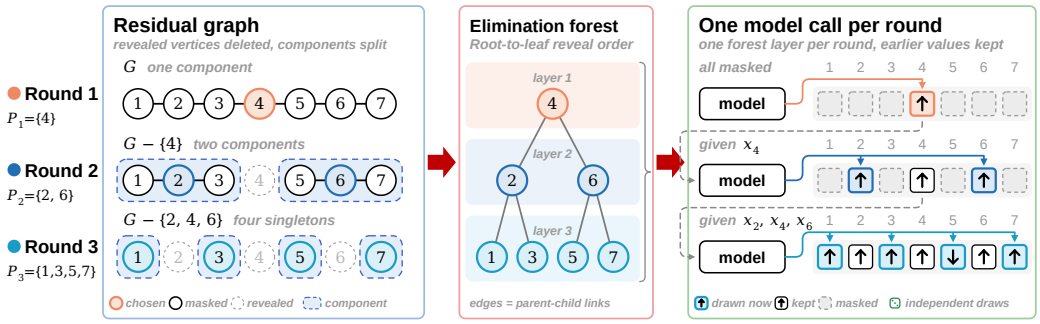


Figure 2: **Parallel sampling with elimination forests.** Revealing a separator splits the residual graph into components; each elimination-forest layer is one round, in which a single masked-model call on all earlier values commits the layer's spins independently. Exact conditionals preserve the target law; exact search minimizes forest height, while layered-separator elimination (LSE) constructs a legal forest.

The nonnegative terms in equation 1 prevent later rounds from canceling earlier factorization error. Exactness is also equivalent to zero total variation and, under full support, zero reverse KL, so the minimum exact round count is the same under these criteria.

### 3.2 Reveal Order and Conditional Exactness

Reveal order matters because exposing a separator can split a dependent set into conditionally independent residual components.

A revealed set $S$ and its observed values $x_S$ form a boundary. A deterministic value-adaptive planner uses this boundary to choose its next nonempty block $B(S, x_S) \subseteq V \setminus S$, uses the same product of true one-coordinate conditionals, and never remasks. Starting from $S = \emptyset$, it stops at $S = V$, possibly after a value-dependent number of rounds. It is exact when its output law is $\mu$; fixed partitions are the value-independent special case. Throughout, value-adaptive means this deterministic, commit-once structural benchmark.

### 3.3 Ising Models and Residual Components

The splitting premise holds for the Ising model, a binary pairwise graphical model with one variable at each vertex of a finite simple graph $G = (V, E)$. Let $n := |V|$, and place one binary spin at each vertex:

$$\mu(x) = \frac{1}{Z} \exp\left( \sum_{v \in V} h_v x_v + \sum_{\{u,v\} \in E} J_{uv} x_u x_v \right), \qquad x_v \in \{-1, +1\}, \tag{2}$$

where $Z$ is the normalizing constant (partition function), $\mathbf{h} = (h_v)_{v \in V} \in \mathbb{R}^V$ is the vector of finite external fields, or single-spin biases, and $\mathbf{J} = (J_{uv})_{\{u,v\} \in E} \in \mathbb{R}^E$ is the coupling vector. The ferromagnetic assumption is $J_{uv} > 0$ for every edge. Here *scalar* means that each graph vertex corresponds to exactly one binary coordinate.

The tree-depth reduction uses two properties that can be stated without the Ising parameterization.

**Definition 1** (Boundary-wise residual faithfulness)**.** *Let $\mu$ be a full-support law on a finite product space indexed by $V$, and let $G$ be a graph on $V$. The pair $(\mu, G)$ is* boundary-wise residual-faithful *if, for every $S \subseteq V$ and every $x_S$, the conditional law factors across the connected components of the induced graph $G - S := G[V \setminus S]$, while any two distinct coordinates in the same component remain dependent.*

For the Ising model, conditioning on $X_S = x_S$ preserves every edge among unrevealed vertices and changes only the field at $v \notin S$, to $h_v + \sum_{u \in S : \{u,v\} \in E} J_{uv} x_u$. The conditional law therefore factors across the components of $G - S$. Section 4.3 proves the within-component clause and its quantitative strengthening, so ferromagnetic Ising laws satisfy Definition 1. $G$ denotes the true interaction graph: construction uses its adjacency, while $\mathbf{J}$ and $\mathbf{h}$ enter the KL charges. Section 4.2 handles conservative graph estimates.

# 4 TREE DEPTH AND THE COST OF SHORTER SCHEDULES

Boundary-wise residual faithfulness makes exactness a component-by-component test: a block is exact if and only if it contains at most one vertex from each residual component. A residual collision means committing a pair of vertices from the same component in a round. Such collisions below tree depth incur factorization error, which we separate from learned-conditional error.

## 4.1 THE RESIDUAL-COMPONENT CRITERION

**Theorem 1** (Residual-component criterion). *Fix $S \subseteq V$ and a boundary assignment $x_S$ in the scalar ferromagnetic Ising model equation 2. The spins $(X_v)_{v\in B}$ in a block $B \subseteq V \setminus S$ are mutually independent given $X_S = x_S$ if and only if $B$ contains at most one vertex from each connected component of $G - S$.*

**Proof sketch.** Graphical separation proves one direction. For the converse, the MTP2 property of ferromagnetic Ising laws makes connectivity imply dependence (Fallat et al., 2017, Theorem 6.1); the shortest-path proof in Section 4.3 recovers this implication and also yields the boundary-uniform magnitude used by the KL bounds. Appendix B gives the complete argument.

To translate this local criterion into rounds, an *elimination forest* $F$ of $G$ is a rooted forest on $V$ in which the endpoints of every edge of $G$ have an ancestor–descendant relation in $F$. Let $H(F)$ be the largest number of vertices on a root-to-leaf path. The tree depth of $G$ is $\mathrm{td}(G) := \min_F H(F)$, where the minimum ranges over elimination forests. Equivalently, minimize over the first elimination vertex of a connected graph and run the remaining components in parallel; the deletion recurrence is equation 10 in Appendix B. Theorem 1 supplies the probabilistic premise: cross-component factorization gives a safe tree-depth schedule, while within-component dependence makes the bound sharp.

**Theorem 2** (Tree-depth upper bound and exact characterization). *Let $\mu$ be a full-support law on a finite product space, and suppose that after every boundary $(S, x_S)$, its conditional law factors across the connected components of $G - S$. Every elimination forest of $G$ gives an exact fixed monotone schedule, so the minimum exact round count is at most $\mathrm{td}(G)$. If $(\mu, G)$ is also boundary-wise residual-faithful, then*

$$\boxed{\min_{\Pi:\, \nu_\Pi = \mu} |\Pi| = \mathrm{td}(G).} \tag{3}$$

*Every realized trajectory of an exact deterministic value-adaptive monotone planner has at least $\mathrm{td}(G)$ rounds. The stronger premise holds for scalar ferromagnetic Ising models and for signed Ising models admitting signs $\sigma_v \in \{-1, +1\}$ such that $\sigma_u \sigma_v J_{uv} > 0$ on every edge. These* gauge-ferromagnetic *models are exactly the balanced signed graphs, in which the product of edge signs is positive on every cycle.*

**Proof sketch.** For the upper bound, reveal one depth layer of an elimination forest per round; its vertices lie in distinct residual components, so cross-component factorization makes every block exact. For the lower bound, residual faithfulness (Theorem 1 for Ising models) forces the first block of a connected residual graph to be a singleton; deleting it and running the remaining components in parallel gives recursion equation 10. The same rule holds at every boundary value for adaptive planners, and a vertex-wise sign flip extends it to balanced signed models. Appendix B supplies the complete proofs.

## 4.2 EXACT SCHEDULES FROM ELIMINATION FORESTS

Forest depth layers are component-legal and therefore exact. Conversely, recursively root each residual component at its earliest scheduled vertex. Block indices increase along every root–leaf path, yielding an elimination forest no taller than the schedule. This proves both directions of equation 3.

Depth-first search yields a legal forest in $O(n + |E|)$, whereas exact subset dynamic programming is exponential (Reidl et al., 2014; Trimble, 2020b; Bonnet et al., 2025). We use legality-checked LSE for construction and the exact solver Bute (Trimble, 2020b) as the minimum-height reference; Appendix C gives comparisons and Appendix B the exact program.

**Scheduling with conservative graph estimates.** Let the planning graph $\widehat{G} = (V, \widehat{E})$ contain all true edges. Every elimination forest of $\widehat{G}$ is also valid for $G$, so exactness transfers. Extra edges may raise height; Corollary 3 bounds the optimal inflation $\mathrm{td}(\widehat{G}) - \mathrm{td}(G)$ by their vertex-cover number, the fewest vertices meeting every added edge. Thus pre-generation edge coverage transfers an exactness guarantee.

### 4.3 Strict dependence within residual components

Fix a boundary $(S, x_S)$ and two vertices in the same component of $G - S$. A shortest path between them has no edges between nonconsecutive vertices, so conditioning on all spins outside the path leaves a ferromagnetic chain. Its endpoints have strictly positive covariance under every outside assignment. Releasing that conditioning preserves strict positivity: endpoint means are increasing functions of the outside spins, and the Fortuin–Kasteleyn–Ginibre (FKG) inequality makes their covariance nonnegative (Fortuin et al., 1971). Thus two vertices in one residual component cannot share an exact factorized block, proving the converse in Theorem 1. Lemma 3 in Appendix B gives the boundary-uniform path bound and the full covariance calculation.

### 4.4 Adaptive schedules and signed models

For a deterministic value-adaptive planner, Lemma 4 identifies each reachable history $h = (S, z)$ with $C_h = \{x : x_S = z\}$, the terminal assignments consistent with that history. If the terminal law is $\mu$, conditioning on these events makes the next product of one-coordinate conditionals equal the target block conditional. Every realized block is therefore component-legal and induces an elimination forest, so every trajectory has at least $\mathrm{td}(G)$ rounds. For signed models in Theorem 2, $Y_v = \sigma_v X_v$ makes all couplings positive while preserving factorized exactness, KL, and the block sequence. The conclusion also holds seed by seed when every positive-probability seed induces an exact conditional output law. Appendices B and C give the complete proofs and finite-state marginal-output controls.

### 4.5 Factorization Charges Below Tree Depth

For the ferromagnetic and gauge-ferromagnetic Ising targets of Theorem 2, every fixed schedule with fewer than $\mathrm{td}(G)$ rounds incurs strictly positive forward KL. Such a schedule must commit two vertices from one residual component together. Their conditional mutual information contributes a positive cost, and the costs of disjoint pairs add to a lower bound by Proposition 1.

The path bound in Lemma 3 makes these costs explicit in terms of couplings, fields, and residual distance. With bounded degree, fields, and coupling strengths bounded above and away from zero, a bounded-distance collision has a positive cost independent of total graph size. Corollary 4 in Appendix B.4 gives the uniform fixed-budget bound. Its graph-wide constant certifies positivity below the exact threshold; the local bounds retain the parameters of each colliding pair.

### 4.6 Schedule Comparison under Learned Conditionals

Let $q_i(\cdot \mid x_S, S)$ be full-support learned conditionals, and let $\nu_\Pi^q$ be the resulting law under a fixed schedule $\Pi$. The risk $\mathcal{R}_q(\Pi)$ sums target-averaged KL from true to learned coordinate conditionals along $\Pi$; Appendix A gives its expanded formula. The error decomposition of Lavenant & Zanella (2025, Proposition 2) then takes the following exact form.

**Proposition 2** (Exact KL decomposition with learned conditionals)**.** *For every fixed schedule* $\Pi$ *and full-support family of learned one-coordinate conditionals,*

$$\boxed{D_{\mathrm{KL}}(\mu \| \nu_\Pi^q) = \underbrace{D_{\mathrm{KL}}(\mu \| \nu_\Pi)}_{\textit{factorization error}} + \underbrace{\mathcal{R}_q(\Pi)}_{\textit{conditional-model error}}}. \tag{4}$$

For a shared target, expected sequence negative log-likelihood (NLL) equals target entropy plus forward KL, so differences in mean coordinate NLL equal forward-KL differences divided by $n$. A component-legal fixed schedule sets the factorization term to zero; if $\Pi$ is held fixed, changing $q$ changes only $\mathcal{R}_q(\Pi)$.

**Proof sketch.** Factoring $\mu/\nu_{\Pi}^{q}$ into the exact-schedule ratio and per-coordinate true-to-learned ratios gives the identity. Corollary 1 compares a tree-depth schedule with all shorter fixed schedules, while Corollary 2 supplies a finite-candidate held-out comparison when the target parameters are unknown.

## 5 EXPERIMENTS

Elimination-forest layers give exact parallel rounds under true conditionals. With learned models, we compare prediction quality, decoding cost, and generalization, then isolate structural and model errors through exact calculations.

**Experimental setup.** We fine-tune LLaDA-8B-Instruct and Dream-v0-Instruct-7B (Nie et al., 2025; Ye et al., 2025) on the same Ising corpus. The denoiser receives the graph, couplings, fields, and observed spins; the planner uses adjacency. Within each backbone and objective, policies share adapters, held-out states, and execution settings. In-distribution (ID) NLL averages three training seeds across 75 graphs; intervals bootstrap graph means with fitted adapters fixed. Appendix D gives corpus splits, encodings, training, validation selection, and timing details.

TD/LSE uses exact tree-depth (TD) scheduling when minimum height is computable and a checked layered-separator elimination (LSE) forest otherwise. It plans once, forms component-legal blocks, and makes one denoiser call per round; end-to-end (E2E) time includes planning. Sequential reveals one coordinate per round. Matched random preserves TD/LSE block sizes but randomizes positions; matched confidence assigns those sizes to the most confident positions; dynamic confidence selects a variable-size confident block. We also compare binary-spin implementations of PUNT's conditional-independence tests (Azangulov et al., 2026), DEMASK's learned dependency head (Ringel et al., 2026), and attention-guided DOS (Zhou et al., 2026). DOS-EB uses an entropy budget; DOS-matched uses the TD/LSE block sizes. For each adapter, validation selects thresholds and attention layers once; these remain fixed across test suites. Planner-aware training samples TD/LSE prefix masks and upweights the next block; uniform-mask training samples observed sets uniformly over cardinalities and positions.

**Prediction quality and generalization.** Under planner-aware training, TD/LSE has the lowest mean NLL on both backbones (Table 1). DOS-matched uses the same block sizes and model-call budget, yet has higher NLL, as do the random and confidence-matched controls. The advantage therefore concerns which variables are grouped, not simply how many are generated per round (Theorem 1). Under uniform-mask training, TD/LSE still improves on these matched controls. Planner-aware training improves the conditionals on TD/LSE's reveal contexts; for a fixed schedule, this changes model risk rather than factorization error (Proposition 2).

| | Planner-aware | | Uniform-mask | | | |
|---|---|---|---|---|---|---|
| **Policy** | **LLaDA** | **Dream** | **LLaDA** | **Dream** | **Mean rounds** | **Calls** |
| **TD/LSE** | **0.6288** | **0.6361** | 0.6603 | 0.6680 | 4.40 | 4.40 |
| Sequential | 0.6421 | 0.6531 | 0.6573 | 0.6646 | 31.47 | 31.47 |
| Matched random | 0.6581 | 0.6673 | 0.6692 | 0.6761 | 4.40 | 4.40 |
| Matched confidence | 0.6544 | 0.6657 | 0.6720 | 0.6810 | 4.40 | 4.40 |
| Dynamic confidence | 0.6422 | 0.6537 | 0.6570 | 0.6653 | 31.46 | 31.46 |
| PUNT | 0.6516 | 0.6616 | 0.6649 | 0.6707 | 3.25 | 15.16 |
| DEMASK | 0.6464 | 0.6532 | 0.6629 | 0.6716 | 18.95 | 18.95 |
| DOS-EB | 0.6466 | 0.6557 | 0.6569 | 0.6652 | 23.06 | 23.06 |
| DOS-matched | 0.6579 | 0.6610 | 0.6733 | 0.6797 | 4.40 | 4.40 |

Table 1: **Prediction quality and decoding cost.** NLL (nats/coordinate) averages 75 ID graphs, four states each, and three adapter seeds. Rounds and calls average both backbones and objectives; calls include auxiliary probes. Bold marks the lowest NLL per column.

TD/LSE forms component-legal blocks from each test graph's adjacency; Theorem 1 applies irrespective of graph family. In the out-of-distribution (OOD) suites, TD/LSE has the lowest NLL averaged across the two backbones under both topology and scale shifts (Table 2). Matched-size controls have similar latency but higher NLL. Under the scale shift, TD/LSE is $17.89\times$ faster than sequential decoding with lower NLL.

| | Mean NLL | | Speedup vs. sequential | |
|---|---|---|---|---|
| **Policy** | **Topology** | **Scale** | **Topology** | **Scale** |
| **TD/LSE** | **0.5743** | **0.6662** | 2.26 | 17.89 |
| Sequential | 0.5835 | 0.6683 | 1.00 | 1.00 |
| Matched random | 0.5883 | 0.6809 | 2.26 | 17.89 |
| Matched confidence | 0.6109 | 0.6893 | 2.26 | 17.90 |
| Dynamic confidence | 0.5829 | 0.6704 | 0.99 | 1.00 |
| PUNT | 0.5898 | 0.6804 | 0.69 | 3.24 |
| DEMASK | 0.5903 | 0.6740 | 1.25 | 1.37 |
| DOS-EB | 0.6005 | 0.6757 | 1.48 | 1.53 |
| DOS-matched | 0.6135 | 0.6844 | 2.25 | 17.87 |

Table 2: **Generalization to unseen graphs.** NLL (nats/coordinate) averages both backbones with planner-aware seed-zero adapters; bold marks the lowest mean. Speedup divides mean sequential by mean policy end-to-end time over matched cases, including planning. OOD suites: Appendix D.5.

**Decoding efficiency and sample quality.** TD/LSE makes one model call per legal block; the calls column of Table 1 shows the contrast directly: 4.40 calls at 4.40 rounds, against 15.16 calls at 3.25 rounds for PUNT and 23.06 for DOS-EB. In planner-aware ID evaluation, TD/LSE is 3.9–8.3× faster than PUNT, DEMASK, and DOS-EB while also lowering NLL; DOS-matched has essentially the same latency as TD/LSE. Figure 3 separates fixed-work timing from full-decoder speedup. TD/LSE's speedup grows with graph size as forest layers save model calls.

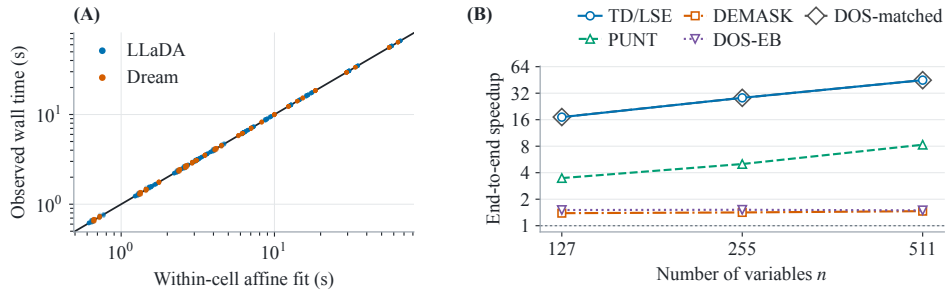


Figure 3: **Decoding latency and speedup.** **(A)** Observed times (seconds) versus within-cell affine fits for 2,700 synchronized fixed-work measurements. **(B)** End-to-end speedup is mean sequential time divided by mean policy time across both backbones and three graph families at each size. Dashed line: sequential decoding.

TD/LSE has the lowest edge-correlation and energy-distribution errors among learned policies (Table 16; Figure 4B). Unlike single-spin means, these metrics test dependencies that independent within-round draws can distort (Proposition 1).

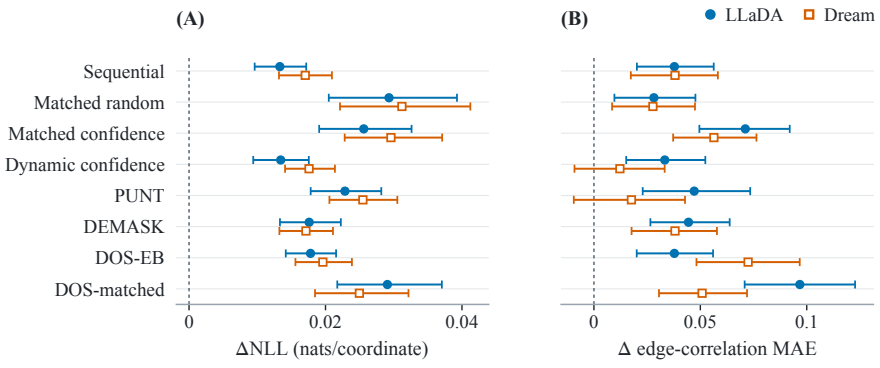


Figure 4: **Error reductions with planner-aware training.** **(A)** NLL (nats/coordinate), averaging three adapters on 75 graphs; **(B)** edge-correlation MAE on 30 graphs (seed-zero adapters). Positive baseline-minus-TD/LSE values favor TD/LSE. Bars: 95% paired graph-bootstrap intervals with adapters and generated samples fixed.

**Structural checks.** Tree-depth layers have zero factorization KL on all four targets; merging the final two layers or randomizing positions at fixed block sizes introduces error (Figure 5A). With exact conditionals, the positive KL is entirely a cost of the schedule. The shortened schedule tests the exact-round threshold (Theorem 2), while fixed-size randomization tests the component criterion at the same round count (Theorem 1). For independent edges, enumeration agrees with the additive pairwise charges in equation 11: KL grows linearly with the number of colliding pairs and increases with coupling strength (Figure 5B).

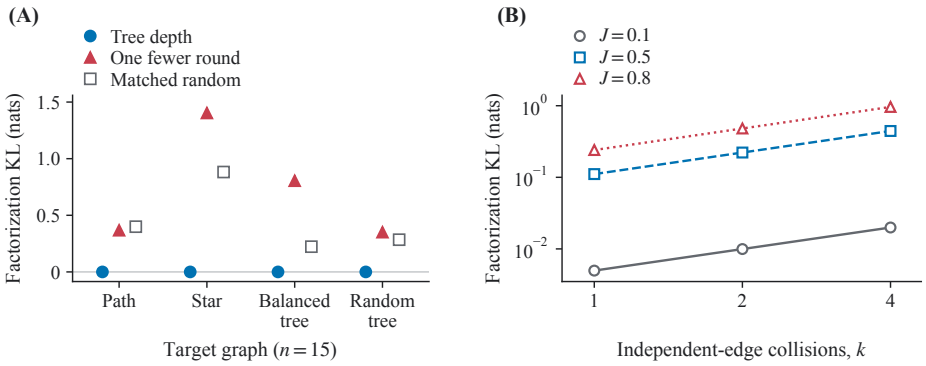


Figure 5: **Exact round limits and collision costs. (A)** Four 15-vertex targets. Tree-depth schedules use $\mathrm{td}(G)$ rounds with zero factorization KL; randomizing positions at fixed layer sizes introduces error, as does merging the final two layers to remove one round. **(B)** For $k$ independent zero-field edges with coupling $J$, one-round KL agrees with analytic lines (markers: exact enumeration). Two-round controls have zero KL.

In separate ablations, removing a round or a true edge creates residual collisions, whereas adding an edge preserves legality (Table 14). Corollary 3 explains this asymmetry and bounds optimal round inflation by the vertex-cover number of the added edges. Complete-state sums on the 15-vertex path $P_{15}$ verify Proposition 2. The compact masked-model control separates zero structural error from learned-conditional risk; TD/LSE has the lowest total KL with fewer rounds than sequential decoding (Table 3). The decomposition exposes why partial metrics mislead: the shortened schedule cuts model risk from 0.0301 to 0.0188, yet its factorization term jumps from 0 to 0.7246, raising total KL by a factor of 24. Model-risk-based selection would therefore pick the shortened schedule, which the exact decomposition (Proposition 2) shows to be strictly worse in total KL than the component-legal tree-depth schedule.

| Schedule | Rounds | Factor KL | Model risk | Total KL |
|---|---|---|---|---|
| **Tree-depth** | 4 | 0 | 0.0301 | **0.0301** |
| Sequential | 15 | 0 | 0.0734 | 0.0734 |
| One fewer round | 3 | 0.7246 | 0.0188 | 0.7434 |
| Matched random | 4 | 0.7661 | 0.0483 | 0.8144 |

Table 3: **KL decomposition on $P_{15}$.** Means over five compact masked multilayer-perceptrons, in nats per joint sample.

## 6 Conclusion

Residual faithfulness identifies tree depth as the exact fixed-schedule round minimum for ferromagnetic Ising models and lower-bounds every trajectory of an exact deterministic value-adaptive planner. Minimum-height elimination forests attain this bound by revealing one depth layer per round. Shorter fixed schedules necessarily incur positive forward KL, and shortest-path dependence estimates give explicit lower bounds on this cost. The characterization extends to balanced signed models; conservative graph estimates preserve exactness with bounded inflation in the optimal round count. For learned conditionals, the exact KL decomposition separates structural factorization error from model error and yields risk-dependent comparisons between schedules. Numerical results support the predicted round threshold and error decomposition. With planner-aware training, dependency-aware schedules improve parallel prediction quality with fewer decoding rounds. These results give tree depth an operational meaning as the optimal round count for exact fixed schedules.

## AI USE STATEMENT

Generative AI tools assisted language editing and formatting.

## REFERENCES


Jacob Austin, Daniel D. Johnson, Jonathan Ho, Daniel Tarlow, and Rianne van den Berg. Structured denoising diffusion models in discrete state-spaces. In *Advances in Neural Information Processing Systems*, volume 34, pp. 17981–17993, 2021. URL https://arxiv.org/abs/2107.03006.

Iskander Azangulov, Teodora Pandeva, Niranjani Prasad, Javier Zazo, and Sushrut Karmalkar. Parallel sampling from masked diffusion models via conditional independence testing. In *International Conference on Learning Representations*, 2026. URL https://arxiv.org/abs/2510.21961.

Vansh Bansal, Cho Cholyeon, Syamantak Kumar, Sujay Sanghavi, and Purnamrita Sarkar. Understanding parallel samplers in masked diffusion via random walks on graphs. *arXiv preprint arXiv:2606.22976*, jun 2026. doi: 10.48550/arXiv.2606.22976. URL https://arxiv.org/abs/2606.22976.

Grigory Bartosh, Teodora Pandeva, Sushrut Karmalkar, and Javier Zazo. Forward-learned discrete diffusion: Learning how to noise to denoise faster. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=45EtKUdgbJ.

Enric Boix-Adserà, Guy Bresler, and Frederic Koehler. Chow-Liu++: Optimal prediction-centric learning of tree Ising models. In *2021 IEEE 62nd Annual Symposium on Foundations of Computer Science*, pp. 417–426. IEEE, 2021. doi: 10.1109/FOCS52979.2021.00049. URL https://arxiv.org/abs/2106.03969.

Édouard Bonnet, Daniel Neuen, and Marek Sokołowski. Treedepth inapproximability and exponential ETH lower bound. *arXiv preprint arXiv:2507.13818*, 2025. URL https://arxiv.org/abs/2507.13818.

Andrew Campbell, Joe Benton, Valentin De Bortoli, Tom Rainforth, George Deligiannidis, and Arnaud Doucet. A continuous time framework for discrete denoising models. In *Advances in Neural Information Processing Systems*, volume 35, pp. 28266–28279, 2022. doi: 10.52202/068431-2049. URL https://arxiv.org/abs/2205.14987.

Huiwen Chang, Han Zhang, Lu Jiang, Ce Liu, and William T. Freeman. MaskGIT: Masked generative image transformer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 11315–11325, 2022. doi: 10.1109/CVPR52688.2022.01103. URL https://doi.org/10.1109/CVPR52688.2022.01103.

Sitan Chen, Kevin Cong, and Jerry Li. Optimal inference schedules for masked diffusion models. In Steve Hanneke and Tor Lattimore (eds.), *Proceedings of the Thirty-Ninth Conference on Learning Theory*, volume 336 of *Proceedings of Machine Learning Research*, pp. 1279–1311. PMLR, 2026. URL https://proceedings.mlr.press/v336/chen26e.html.

Thomas M. Cover and Joy A. Thomas. *Elements of Information Theory*. Wiley-Interscience, 2 edition, 2006. doi: 10.1002/047174882X. URL https://doi.org/10.1002/047174882X.

Shaun Fallat, Steffen Lauritzen, Kayvan Sadeghi, Caroline Uhler, Nanny Wermuth, and Piotr Zwiernik. Total positivity in markov structures. *The Annals of Statistics*, 45(3):1152–1184, 2017. doi: 10.1214/16-AOS1478. URL https://doi.org/10.1214/16-AOS1478.

C. M. Fortuin, P. W. Kasteleyn, and J. Ginibre. Correlation inequalities on some partially ordered sets. *Communications in Mathematical Physics*, 22(2):89–103, 1971. doi: 10.1007/BF01651330. URL https://doi.org/10.1007/BF01651330.

Stipe Frkovic, Metod Jazbec, Dan Zhang, Christian A. Naesseth, Ilija Bogunovic, and Eric Nalisnick. Re-evaluating confidence remasking in masked diffusion language models. *arXiv preprint arXiv:2606.12232*, 2026. URL https://arxiv.org/abs/2606.12232.

Chunsan Hong, Seonho An, Min-Soo Kim, and Jong Chul Ye. Improving discrete diffusion unmasking policies beyond explicit reference policies. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=on6cb46OhD.

Vardis Kandiros, Constantinos Daskalakis, Yuval Dagan, and Davin Choo. Learning and testing latent-tree ising models efficiently. In *Proceedings of the Thirty-Sixth Conference on Learning Theory*, volume 195 of *Proceedings of Machine Learning Research*, pp. 1666–1729. PMLR, 2023. URL https://proceedings.mlr.press/v195/kandiros23a.html.

Jaeyeon Kim, Kulin Shah, Vasilis Kontonis, Sham M. Kakade, and Sitan Chen. Train for the worst, plan for the best: Understanding token ordering in masked diffusions. In Aarti Singh, Maryam Fazel, Daniel Hsu, Simon Lacoste-Julien, Felix Berkenkamp, Tegan Maharaj, Kiri Wagstaff, and Jerry Zhu (eds.), *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 30749–30768. PMLR, 2025. URL https://proceedings.mlr.press/v267/kim25ah.html.

Steffen L. Lauritzen. *Graphical Models*. Oxford Statistical Science Series. Oxford University Press, 1996. doi: 10.1093/oso/9780198522195.001.0001. URL https://doi.org/10.1093/oso/9780198522195.001.0001.

Hugo Lavenant and Giacomo Zanella. Error bounds and optimal schedules for masked diffusions with factorized approximations. *arXiv preprint arXiv:2510.25544*, 2025. URL https://arxiv.org/abs/2510.25544.

Gen Li and Changxiao Cai. Breaking AR's sampling bottleneck: Provable acceleration via diffusion language models. *Advances in Neural Information Processing Systems*, 2025. URL https://arxiv.org/abs/2505.21400.

Anji Liu, Oliver Broadrick, Mathias Niepert, and Guy Van den Broeck. Discrete copula diffusion. In *International Conference on Learning Representations*, pp. 88953–88979, 2025. URL https://openreview.net/forum?id=FXw0okNcOb.

Guanxi Lu, Hao Mark Chen, Yuto Karashima, Zhican Wang, Daichi Fujiki, and Hongxiang Fan. AdaBlock-dLLM: Semantic-aware diffusion LLM inference via adaptive block size. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=0Cv9PwL7cI.

Jaroslav Nešetřil and Patrice Ossona de Mendez. *Sparsity: Graphs, Structures, and Algorithms*, volume 28 of *Algorithms and Combinatorics*. Springer, 2012. doi: 10.1007/978-3-642-27875-4. URL https://doi.org/10.1007/978-3-642-27875-4.

Shen Nie, Fengqi Zhu, Zebin You, Xiaolu Zhang, Jingyang Ou, Jun Hu, Jun Zhou, Yankai Lin, Ji-Rong Wen, and Chongxuan Li. Large language diffusion models. *arXiv preprint arXiv:2502.09992*, 2025. URL https://arxiv.org/abs/2502.09992.

Fred Zhangzhi Peng, Zachary Bezemek, Sawan Patel, Jarrid Rector-Brooks, Sherwood Yao, Avishek Joey Bose, Alexander Tong, and Pranam Chatterjee. Path planning for masked diffusion model sampling. *arXiv preprint arXiv:2502.03540*, 2025. URL https://arxiv.org/abs/2502.03540.

Fred Zhangzhi Peng, Zachary Bezemek, Jarrid Rector-Brooks, Shuibai Zhang, Anru R. Zhang, Michael Bronstein, Alexander Tong, and Avishek Joey Bose. Planner aware path learning in diffusion language models training. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=lAlI5FuIf7.

Yury Polyanskiy and Yihong Wu. *Information Theory: From Coding to Learning*. Cambridge University Press, 2025. doi: 10.1017/9781108966351. URL https://doi.org/10.1017/9781108966351.

Pradeep Ravikumar, Martin J. Wainwright, and John D. Lafferty. High-dimensional Ising model selection using $\ell_1$-regularized logistic regression. *The Annals of Statistics*, 38(3):1287–1319, 2010. doi: 10.1214/09-AOS691. URL https://arxiv.org/abs/1010.0311.

Felix Reidl, Peter Rossmanith, Fernando Sánchez Villaamil, and Somnath Sikdar. A faster parameterized algorithm for treedepth. In *Automata, Languages, and Programming*, volume 8572 of *Lecture Notes in Computer Science*, pp. 931–942. Springer, 2014. doi: 10.1007/978-3-662-43948-7_77. URL https://arxiv.org/abs/1401.7540.

Liran Ringel, Ameen Ali, and Yaniv Romano. Dependency-guided parallel decoding in discrete diffusion language models. *arXiv preprint arXiv:2604.02560*, 2026. doi: 10.48550/arXiv.2604.02560. URL https://arxiv.org/abs/2604.02560.

James Trimble. PACE solver description: Bute-Plus: A bottom-up exact solver for treedepth. In *15th International Symposium on Parameterized and Exact Computation*, volume 180 of *Leibniz International Proceedings in Informatics*, pp. 34:1–34:4. Schloss Dagstuhl–Leibniz-Zentrum für Informatik, 2020a. doi: 10.4230/LIPIcs.IPEC.2020.34. URL https://arxiv.org/abs/2006.09912.

James Trimble. An algorithm for the exact treedepth problem. In *18th International Symposium on Experimental Algorithms*, volume 160 of *Leibniz International Proceedings in Informatics*, pp. 19:1–19:14. Schloss Dagstuhl–Leibniz-Zentrum für Informatik, 2020b. doi: 10.4230/LIPIcs.SEA.2020.19. URL https://doi.org/10.4230/LIPIcs.SEA.2020.19.

Martin J. Wainwright and Michael I. Jordan. Graphical models, exponential families, and variational inference. *Foundations and Trends in Machine Learning*, 1(1–2):1–305, 2008. doi: 10.1561/2200000001. URL https://doi.org/10.1561/2200000001.

Guanghan Wang, Yair Schiff, Subham Sekhar Sahoo, and Volodymyr Kuleshov. Remasking discrete diffusion models with inference-time scaling. *arXiv preprint arXiv:2503.00307*, 2025. URL https://arxiv.org/abs/2503.00307.

Jiacheng Ye, Zhihui Xie, Lin Zheng, Jiahui Gao, Zirui Wu, Xin Jiang, Zhenguo Li, and Lingpeng Kong. Dream 7b: Diffusion large language models. *arXiv preprint arXiv:2508.15487*, 2025. URL https://arxiv.org/abs/2508.15487.

Shaorong Zhang, Longxuan Yu, Rob Brekelmans, Luhan Tang, Salman Asif, and Greg Ver Steeg. Generation order and parallel decoding in masked diffusion models: An information-theoretic perspective. *arXiv preprint arXiv:2602.00286*, 2026. URL https://arxiv.org/abs/2602.00286.

Xueyu Zhou, Yangrong Hu, and Jian Huang. DOS: Dependency-oriented sampler for masked diffusion language models. In *Findings of the Association for Computational Linguistics: ACL 2026*, pp. 17404–17419. Association for Computational Linguistics, 2026. doi: 10.18653/v1/2026.findings-acl.861. URL https://aclanthology.org/2026.findings-acl.861/.

# A Proofs of the KL Decompositions

We derive the two KL identities used in the main text by expanding the target-to-sampler likelihood ratio round by round.

## A.1 Exact conditionals

For exact conditionals, the likelihood ratio splits across rounds, and each term measures the factorization error of one committed block. For the fixed schedule $\Pi = (P_1, \dots, P_r)$, independent draws within each committed block give the output law

$$\nu_\Pi(x) = \prod_{t=1}^{r} \prod_{i \in P_t} \mu(x_i \mid x_{P_{<t}}). \tag{5}$$

*Proof of Proposition 1.* Let $X \sim \mu$. For a past assignment $z = x_{P_{<t}}$, define the product conditional law

$$\bar{\mu}_t^z(x_{P_t}) := \prod_{i \in P_t} \mu(x_i \mid z).$$

The probability chain rule for $\mu$ and the sampler definition equation 5 give

$$\mu(x) = \prod_{t=1}^{r} \mu(x_{P_t} \mid x_{P_{<t}}), \qquad \nu_\Pi(x) = \prod_{t=1}^{r} \bar{\mu}_t^{x_{P_{<t}}}(x_{P_t}).$$

Therefore

$$\begin{aligned} D_{\mathrm{KL}}(\mu \| \nu_\Pi) &= \mathbb{E}_\mu \log \frac{\mu(X)}{\nu_\Pi(X)} \\ &= \sum_{t=1}^{r} \mathbb{E}_\mu \log \frac{\mu(X_{P_t} \mid X_{P_{<t}})}{\bar{\mu}_t^{X_{P_{<t}}}(X_{P_t})} \\ &= \sum_{t=1}^{r} \mathbb{E}_\mu D_{\mathrm{KL}}\Big(\mu(X_{P_t} \in \cdot \mid X_{P_{<t}}) \,\Big\|\, \bar{\mu}_t^{X_{P_{<t}}}\Big), \end{aligned}$$

which is equation 1. Full support makes every displayed conditional probability positive. Each summand is nonnegative and vanishes exactly when the conditional block law equals the product of its one-coordinate marginals. Thus the full divergence vanishes exactly when each block's coordinates are mutually independent given its past.

### A.2 Learned conditionals

With learned conditionals, each round adds marginal conditional-model terms to the same block factorization term. Let $q_i(\cdot \mid x_S, S)$ be full-support one-coordinate conditionals for every revealed context. For a fixed schedule, define

$$\nu_\Pi^q(x) := \prod_{t=1}^{r} \prod_{i \in P_t} q_i(x_i \mid x_{P_{<t}}, P_{<t})$$

and the population conditional-model risk

$$\mathcal{R}_q(\Pi) := \sum_{t=1}^{r} \sum_{i \in P_t} \mathbb{E}_\mu D_{\mathrm{KL}}(\mu(X_i \in \cdot \mid X_{P_{<t}}) \,\|\, q_i(\cdot \mid X_{P_{<t}}, P_{<t})). \tag{6}$$

This risk averages only over the contexts and coordinates queried by $\Pi$.

*Proof of Proposition 2.* For a fixed past value $z = x_{P_{<t}}$, write

$$q_t^z(x_{P_t}) := \prod_{i \in P_t} q_i(x_i \mid z, P_{<t}).$$

The chain rule applied to $\mu$ and $\nu_\Pi^q$ gives

$$D_{\mathrm{KL}}(\mu \| \nu_\Pi^q) = \sum_{t=1}^{r} \mathbb{E}_\mu D_{\mathrm{KL}}\Big(\mu(X_{P_t} \in \cdot \mid X_{P_{<t}}) \,\Big\|\, q_t^{X_{P_{<t}}}\Big).$$

For a fixed $z$, insert the exact product conditional $\bar{\mu}_t^z$:

$$\begin{aligned} &D_{\mathrm{KL}}(\mu(X_{P_t} \in \cdot \mid z) \,\|\, q_t^z) \\ &\quad = D_{\mathrm{KL}}(\mu(X_{P_t} \in \cdot \mid z) \,\|\, \bar{\mu}_t^z) + \mathbb{E}_{\mu(\cdot \mid z)} \log \frac{\bar{\mu}_t^z(X_{P_t})}{q_t^z(X_{P_t})} \\ &\quad = D_{\mathrm{KL}}(\mu(X_{P_t} \in \cdot \mid z) \,\|\, \bar{\mu}_t^z) + \sum_{i \in P_t} D_{\mathrm{KL}}(\mu(X_i \in \cdot \mid z) \,\|\, q_i(\cdot \mid z, P_{<t})). \end{aligned}$$

The last equality holds because the marginal of the conditional block law on coordinate $i$ is $\mu(X_i \in \cdot \mid z)$. Averaging over $z = X_{P_{<t}}$ and summing over $t$ proves equation 4.

Combining this decomposition with the fixed-budget factorization floor in Corollary 4, proved in Appendix B, yields a population-level comparison between learned schedules. The graph-wide charge $\Lambda$ and minimum collision count $\mathfrak{v}_r$ are defined in equation 12.

**Corollary 1** (Population comparison for learned schedules)**.** *Assume $E \neq \emptyset$, and let $\Pi_{\mathrm{td}}$ be a tree-depth schedule. For $1 \leq r < \mathrm{td}(G)$, if*

$$\mathcal{R}_q(\Pi_{\mathrm{td}}) < \Lambda(G, \mathbf{J}, \mathbf{h})\, \mathfrak{v}_r(G),$$

*then the learned tree-depth sampler has smaller forward KL than every learned fixed-schedule factorized sampler using at most $r$ rounds. Taking $r = \mathrm{td}(G) - 1$ covers every shorter fixed schedule.*

*Proof.* By equation 4, the tree-depth schedule has total forward KL $\mathcal{R}_q(\Pi_{\mathrm{td}})$. Corollary 4 gives every schedule of at most $r$ rounds factorization error at least $\Lambda(G, \mathbf{J}, \mathbf{h})\mathfrak{v}_r(G)$; its conditional-model risk is nonnegative.

### A.3 Held-out schedule comparison

The unknown target entropy cancels when two learned schedules are evaluated on the same validation observations, leaving a bounded empirical KL difference.

**Corollary 2** (Held-out comparison of fixed learned schedules)**.** *Fix a learned conditional family $q$, a reference schedule $\Pi_0$, and a nonempty finite candidate family $\mathcal{C}$, all before observing a validation set. Assume every binary outcome probability queried by these schedules is at least a known $\alpha \in (0, 1/2)$. For independent $X^{(1)}, \dots, X^{(m)} \sim \mu$, define*

$$\widehat{L}_m(\Pi) := -\frac{1}{m} \sum_{j=1}^{m} \log \nu^q_\Pi(X^{(j)}), \qquad R_\alpha := n \log \frac{1-\alpha}{\alpha}.$$

*For every $\delta \in (0, 1)$, with probability at least $1 - \delta$,*

$$\min_{\Pi \in \mathcal{C}} \big\{ \widehat{L}_m(\Pi) - \widehat{L}_m(\Pi_0) \big\} > R_\alpha \sqrt{\frac{2 \log(|\mathcal{C}|/\delta)}{m}} \quad \Longrightarrow \quad D_{\mathrm{KL}}(\mu \| \nu^q_{\Pi_0}) < \min_{\Pi \in \mathcal{C}} D_{\mathrm{KL}}(\mu \| \nu^q_\Pi). \tag{7}$$

*Proof of Corollary 2.* Write $\ell_\Pi(x) := -\log \nu^q_\Pi(x)$. Every schedule commits exactly $n$ binary variables. Since both outcomes have probability at least $\alpha$, each queried probability is also at most $1 - \alpha$, and hence

$$n \log \frac{1}{1-\alpha} \leq \ell_\Pi(x) \leq n \log \frac{1}{\alpha}.$$

Thus $Z_\Pi(x) := \ell_\Pi(x) - \ell_{\Pi_0}(x)$ lies in $[-R_\alpha, R_\alpha]$.

$$\mathbb{E}_\mu Z_\Pi(X) = D_{\mathrm{KL}}(\mu \| \nu^q_\Pi) - D_{\mathrm{KL}}(\mu \| \nu^q_{\Pi_0}),$$

because the entropy of $\mu$ cancels. The one-sided Hoeffding inequality and a union bound imply, simultaneously for all $\Pi \in \mathcal{C}$,

$$\mathbb{E}_\mu Z_\Pi(X) \geq \widehat{L}_m(\Pi) - \widehat{L}_m(\Pi_0) - R_\alpha \sqrt{\frac{2 \log(|\mathcal{C}|/\delta)}{m}}$$

with probability at least $1 - \delta$. Condition equation 7 makes every right-hand side positive and proves the claim.

For $\varepsilon > 0$, the deviation radius in equation 7 is at most $\varepsilon$ whenever

$$m \geq \frac{2n^2 \log^2((1-\alpha)/\alpha) \log(|\mathcal{C}|/\delta)}{\varepsilon^2}.$$

Thus the candidate-family size enters only logarithmically, whereas the explicit $n$ and $\alpha$ factors are the cost of uniformly bounding a full-sequence log-loss difference. Candidate construction and held-out certification are therefore separate stages.

# B PROOFS OF THE TREE-DEPTH CHARACTERIZATION

Residual dependence yields tree-depth optimality, exact forest schedules, and the quantitative KL bounds.

## B.1 STRICT RESIDUAL DEPENDENCE

Graphical separation gives independence across residual components; the converse below proves boundary-uniform dependence within each component.

**Lemma 1** (Conditional and marginal association)**.** *Let $\mu$ be the scalar ferromagnetic Ising law in equation 2, with finite fields and $J_{uv} > 0$ on every edge. Fix $S \subseteq V$ and $x_S$. The conditional law on $V \setminus S$ is associated: for any coordinatewise nondecreasing functions $f$ and $g$,*

$$\mathrm{Cov}\big(f(X_{V\setminus S}), g(X_{V\setminus S}) \mid X_S = x_S\big) \geq 0.$$

*Every marginal and every further conditional law has the same property.*

*Proof.* Conditioning on $x_S$ replaces $h_i$ by $h_i + \sum_{j\in S:\{i,j\}\in E} J_{ij}x_j$ for each $i \notin S$. All remaining pair couplings stay positive. On $\{-1,+1\}^2$, unary terms have zero mixed second difference and the term $J_{ij}x_ix_j$ contributes $4J_{ij}$. The density is therefore log-supermodular: its log has nonnegative mixed differences. The FKG inequality then gives association. Applying the same inequality to functions that ignore marginalized coordinates proves the marginal statement; further conditioning only updates the unary fields.

**Lemma 2** (Monotonicity of endpoint means)**.** *Let $\mu$ be the scalar ferromagnetic Ising law in equation 2, with finite fields and $J_{uv} > 0$ on every edge. Fix $S, x_S$, let $P$ be an induced path in $G - S$, and write $V(P)$ for its vertex set. Put $R := V \setminus (S \cup V(P))$. For either endpoint $a$ of $P$, the function $m_a(y_R) := \mathbb{E}[X_a \mid X_S = x_S, X_R = y_R]$ is coordinatewise nondecreasing on $\{-1,+1\}^R$.*

*Proof.* It suffices to flip one coordinate $w \in R$ from $-1$ to $+1$ while holding all other boundary spins fixed. Let $\pi_-$ and $\pi_+$ be the resulting conditional laws on $X_{V(P)}$. Their likelihood ratio is proportional to $L_w(x_{V(P)}) := \exp(2\sum_{i\in V(P):\{i,w\}\in E} J_{iw}x_i)$, which is coordinatewise nondecreasing. By Lemma 1, $\pi_-$ is associated. Hence, for the increasing endpoint spin $X_a$,

$$\mathbb{E}_{\pi_+} X_a - \mathbb{E}_{\pi_-} X_a = \frac{\mathrm{Cov}_{\pi_-}(X_a, L_w)}{\mathbb{E}_{\pi_-} L_w} \geq 0.$$

Successive single-coordinate flips prove the claim.

**Lemma 3** (Boundary-uniform path dependence)**.** *Let $\mu$ be the scalar ferromagnetic Ising law in equation 2, with finite fields and $J_{uv} > 0$ on every edge. Fix $S \subseteq V$, $x_S$, and distinct $u, v \notin S$. If $u$ and $v$ lie in the same component of $G - S$, let $P = (v_0 = u, v_1, \ldots, v_\ell = v)$ be a shortest path, put $V(P) := \{v_0, \ldots, v_\ell\}$, and write $J_j := J_{v_{j-1}v_j}$. Define*

$$\begin{aligned} b_i(P) &:= |h_i| + \sum_{\substack{\{i,w\}\in E \\ w\notin V(P)}} |J_{iw}|, \\ \Gamma(P) &:= 2^{-2\ell} \exp\left(-2 \sum_{i\in V(P)} b_i(P)\right) \prod_{j=1}^{\ell} (1 - e^{-4J_j}). \end{aligned} \tag{8}$$

*Here $b_i(P)$ bounds the effective field at $i \in V(P)$ under any outside assignment, and $I_\mu$ denotes conditional mutual information under $\mu$. Then, uniformly in $x_S$,*

$$\mathrm{Cov}(X_u, X_v \mid X_S = x_S) \geq \Gamma(P) > 0, \qquad I_\mu(X_u; X_v \mid X_S = x_S) \geq \frac{\Gamma(P)^2}{2}.$$

*Proof.* A shortest path is chordless. Put $R := (V \setminus S) \setminus V(P)$, and fix $X_R = y_R$. The remaining law on the path is a chain with couplings $J_j := J_{v_{j-1}v_j} > 0$ and effective fields

$$\alpha_i := h_{v_i} + \sum_{\substack{\{v_i,w\}\in E \\ w\notin V(P)}} J_{v_i w} x_w, \qquad |\alpha_i| \le b_{v_i}(P), \qquad 0 \le i \le \ell,$$

where $x_w$ is the fixed value from either $x_S$ or $y_R$. In the row and column order $(-1,+1)$, the unnormalized endpoint table is

$$M = D_0 K_1 D_1 \cdots K_\ell D_\ell, \quad D_i := \operatorname{diag}(e^{-\alpha_i}, e^{\alpha_i}), \quad K_j := \begin{pmatrix} e^{J_j} & e^{-J_j} \\ e^{-J_j} & e^{J_j} \end{pmatrix}.$$

Matrix multiplication sums over the internal path spins. Since $\det D_i = 1$,

$$\det M = \prod_{j=1}^{\ell} (e^{2J_j} - e^{-2J_j}) = e^{2\sum_{j=1}^{\ell} J_j} \prod_{j=1}^{\ell} (1 - e^{-4J_j}).$$

Let $Z_P(y_R)$ be the chain partition function. Summing its $2^{\ell+1}$ configurations gives

$$Z_P(y_R) \le 2^{\ell+1} \exp\left( \sum_{i=0}^{\ell} |\alpha_i| + \sum_{j=1}^{\ell} J_j \right).$$

For a normalized binary endpoint table, covariance is four times its determinant. Therefore

$$\begin{aligned} &\operatorname{Cov}(X_u, X_v \mid X_S = x_S, X_R = y_R) \\ &\quad = \frac{4 \det M}{Z_P(y_R)^2} \ge 2^{-2\ell} e^{-2\sum_{i\in V(P)} b_i(P)} \prod_{j=1}^{\ell} (1 - e^{-4J_j}) = \Gamma(P). \end{aligned}$$

By Lemma 2, the endpoint means $m_u(y_R)$ and $m_v(y_R)$ are nondecreasing. The marginal law of $X_R \mid X_S = x_S$ is associated by Lemma 1. The conditional law of total covariance consequently yields

$$\begin{aligned} \operatorname{Cov}(X_u, X_v \mid X_S = x_S) = \mathbb{E}[\operatorname{Cov}(X_u, X_v \mid X_S, X_R) \mid X_S = x_S] \\ + \operatorname{Cov}(m_u(X_R), m_v(X_R) \mid X_S = x_S) \ge \Gamma(P) > 0. \end{aligned} \tag{9}$$

For binary spins, the total variation distance between their joint law and the product of its marginals is one half of the absolute covariance. Pinsker's inequality in nats then gives the mutual-information bound.

**Local scale under bounded geometry.** If the maximum degree is $\Delta$, $|h_i| \le H$, and $0 < J_{\min} \le |J_e| \le J_{\max}$, set $B := H + \Delta J_{\max}$. For a violating pair at residual distance $\ell$, every path budget obeys $b_i(P) \le B$, so Lemma 3 gives, uniformly over the revealed boundary,

$$I_\mu(X_u; X_v \mid X_S = x_S) \ge 2^{-4\ell-1} e^{-4(\ell+1)B} \left(1 - e^{-4J_{\min}}\right)^{2\ell}.$$

Consequently, bounded-distance conflicts have a positive charge independent of the total graph size; dependence on $n$ enters only when a single graph-wide constant must cover paths of every possible length.

### B.2 The component criterion and exact round complexity

The path-dependence lemma and blockwise KL accounting yield the component criterion and tree-depth optimality.

*Proof of Theorem 1.* Fix $X_S = x_S$. The conditional density factors over the connected components of $G - S$, so spins chosen from distinct components are mutually independent. Conversely, if a block contains two vertices from one component, Lemma 3 shows that their two-coordinate marginal is not a product; hence the full block conditional cannot be a product. This proves the residual-component criterion. Together with Proposition 1, it is also the exact scheduling criterion.

To extend the fixed-schedule argument to deterministic value-adaptive planners, we first identify each reachable history with a cylinder event of the terminal output.

**Lemma 4** (Prefix cylinders for deterministic planners)**.** *Let $\mathcal{A}$ be a deterministic value-adaptive commit-once planner with terminal law $\nu_{\mathcal{A}}$. For a reachable history $h = (S, z)$, set*

$$C_h := \{x \in \mathcal{X} : x_S = z\}.$$

*As events of the terminal output, the planner reaches $h$ if and only if its output lies in $C_h$. If $B = B(S, z)$ is the next block, $y \in \mathcal{X}_B$, and $C_{h,y} := \{x \in \mathcal{X} : x_S = z,\ x_B = y\}$, then*

$$\nu_{\mathcal{A}}(C_{h,y} \mid C_h) = \prod_{i \in B} \mu(X_i = y_i \mid X_S = z).$$

*Proof.* Induct on the number of rounds preceding the history. The initial history $(\emptyset, \emptyset)$ corresponds to the whole output space. Suppose the claim holds at $h = (S, z)$. On $C_h$, the induction hypothesis says that the planner reaches $h$; determinism then fixes the same next block $B(S, z)$ for every such output. Because committed values are never changed, the successor with block value $y$ is reached if and only if the terminal output belongs to $C_{h,y}$. This proves the event identity at every successor. Conditional on reaching $h$, the sampler draws the coordinates in $B$ independently from their exact one-coordinate conditionals, which gives the displayed transition probability.

*Proof of Theorem 2.* Let $r_\star(G)$ be the minimum number of rounds among exact fixed monotone schedules on graph $G$. Assume first only that every boundary conditional factors across the components of $G - S$. Let $F$ be any elimination forest and reveal its depth layers in order. Vertices in one layer belong to distinct components after the preceding layers have been removed, so every block factors and the schedule is exact. Hence $r_\star(G) \le H(F)$, and minimizing over $F$ gives $r_\star(G) \le \mathrm{td}(G)$.

Now suppose that $(\mu, G)$ is boundary-wise residual-faithful. A block conditional factors exactly when the block contains at most one coordinate from each residual component: cross-component factorization gives sufficiency, while within-component pair dependence gives necessity. Theorem 1 establishes this rule for the Ising specialization. If $G$ is disconnected, schedules for its components can run in parallel, while restricting any exact global schedule to one component remains exact. Thus

$$r_\star(G) = \max_{C \text{ component of } G} r_\star(C).$$

If $G$ is connected and nonempty, the component criterion forces the first block to be a singleton $\{v\}$. After revealing $v$, the components of $G - \{v\}$ can run in parallel for every revealed value because the conditional target remains boundary-wise residual-faithful, so

$$r_\star(G) = 1 + \min_{v \in V} r_\star(G - \{v\}).$$

Together with $r_\star(\emptyset) = 0$, these are precisely the tree-depth recurrences in equation 10. Induction on $|V|$ gives $r_\star(G) = \mathrm{td}(G)$.

For a gauge-ferromagnetic model, set $Y_v := \sigma_v X_v$. This coordinatewise bijection changes the field to $\sigma_v h_v$ and the coupling on $\{u, v\}$ to $\sigma_u \sigma_v J_{uv} = |J_{uv}| > 0$. It preserves conditional independence, mutual information, factorized exactness, and KL divergence, so the fixed-schedule conclusion transfers. The existence of such signs is equivalent to a positive product of edge signs on every cycle: one direction follows because each vertex sign occurs twice around a cycle, while the other follows by choosing signs along a spanning forest and checking each fundamental cycle.

Finally, let $\mathcal{A}$ be a deterministic value-adaptive monotone planner with terminal law $\nu_{\mathcal{A}} = \mu$, and fix a reachable history $h = (S, z)$. Full support gives $\mu(C_h) > 0$. For the next block $B = B(S, z)$, Lemma 4 and equality of the terminal laws imply, for every $y \in \mathcal{X}_B$,

$$\begin{aligned} \mu(X_B = y \mid X_S = z) &= \frac{\mu(C_{h,y})}{\mu(C_h)} = \frac{\nu_{\mathcal{A}}(C_{h,y})}{\nu_{\mathcal{A}}(C_h)} \\ &= \prod_{i \in B} \mu(X_i = y_i \mid X_S = z). \end{aligned}$$

Thus the target block conditional factors at every reachable history. Boundary-wise residual faithfulness (Theorem 1 in the Ising case) places at most one chosen vertex in each component of $G - S$. Along any realized trajectory, the resulting ordered partition obeys the residual-component rule at every round; the schedule-to-forest construction consequently gives an elimination forest whose height is at most the trajectory length. Every realized trajectory has at least $\mathrm{td}(G)$ rounds, while the fixed optimal schedule attains that number. Conjugating an adaptive planner and its target by the same coordinatewise sign map preserves its block sequence, terminal exactness, and round count, so this adaptive lower bound transfers to the gauge-ferromagnetic case as well.

### B.3 Elimination-Forest Schedules and Tree-Depth Computation

The deletion recurrence used in the preceding proof is

$$\mathrm{td}(\emptyset) = 0, \qquad \mathrm{td}(G) = \begin{cases} \max_{C \text{ component of } G} \mathrm{td}(C), & \text{if } G \text{ is disconnected,} \\ 1 + \min_{v \in V} \mathrm{td}(G - \{v\}), & \text{if } G \text{ is connected and nonempty} \end{cases}. \tag{10}$$

It gives both forest–schedule maps and the exact subset dynamic program. Let $F$ be an elimination forest and reveal all vertices at forest depth $t$ in round $t$. After shallower vertices are removed, distinct depth-$t$ vertices root incomparable residual subtrees. The definition of an elimination forest forbids graph edges between those subtrees. The vertices therefore lie in distinct residual components and form an exact block.

Conversely, start from an exact ordered partition. In each connected component, make its unique earliest-scheduled vertex the root. Delete that vertex, recurse inside every residual component, and attach the resulting forests below the root. Along each root-to-leaf chain, block indices strictly increase, so a vertex's forest depth is at most its scheduled round. Every graph edge has ancestor-comparable endpoints, and no root-to-leaf path uses more levels than the schedule has blocks.

For exact computation, process all subsets $U \subseteq V$ in increasing cardinality. If $G[U]$ is disconnected, store the maximum value over its components. If it is connected and nonempty, store $1 + \min_{v \in U} \mathrm{td}(G[U \setminus \{v\}])$ and a minimizing vertex. Connectivity and the candidate deletions can be handled in $O(n(n + |E|))$ time per subset by a direct implementation, giving $O(n(n + |E|)2^n)$ time and $O(2^n)$ memory. Recursing through the stored choices recovers an optimal forest.

The forest construction also makes exactness robust to conservative overestimation of the interaction graph.

**Corollary 3** (Exact Scheduling under Conservative Supergraphs)**.** *Let $\mu$ be a scalar ferromagnetic or gauge-ferromagnetic Ising model with true interaction graph $G = (V, E)$. Let $\widetilde{G} = (V, \widetilde{E})$ satisfy $E \subseteq \widetilde{E}$. Every elimination forest $F$ of $\widetilde{G}$ gives an exact depth-layer schedule for $\mu$.*

*Let $\tau^+(\widetilde{G}, G)$ be the minimum vertex-cover size of the false-positive edge graph $(V, \widetilde{E} \setminus E)$. Then*

$$0 \le \mathrm{td}(\widetilde{G}) - \mathrm{td}(G) \le \min\Big\{\tau^+(\widetilde{G}, G),\ |V| - \mathrm{td}(G)\Big\}.$$

*Consequently, if an estimate $\widehat{G} = (V, \widehat{E})$, produced before generation, satisfies*

$$\Pr\Big(E \subseteq \widehat{E},\ \tau^+(\widehat{G}, G) \le k\Big) \ge 1 - \delta,$$

*then its minimum-height forest schedule is exact and uses at most $\mathrm{td}(G) + k$ rounds with probability at least $1 - \delta$ over graph estimation. Edge coverage alone still transfers exactness.*

*Proof.* Every elimination forest of $\widetilde{G}$ is also one of $G$, proving exactness and $\mathrm{td}(G) \le \mathrm{td}(\widetilde{G})$. Let $C$ cover all false-positive edges. Place the vertices of $C$ in an arbitrary chain above a minimum-height elimination forest of $G - C$. Every false-positive edge meets $C$, all edges incident to $C$ have comparable endpoints, and the remaining edges are handled by the forest of $G - C$. The resulting forest is valid for $\widetilde{G}$ and has height at most

$$|C| + \mathrm{td}(G - C) \le |C| + \mathrm{td}(G).$$

Minimizing over $C$ and using $\mathrm{td}(\widetilde{G}) \le |V|$ proves the inflation bound. Applying the deterministic statement on the displayed estimation event proves the probability claim; this outer randomness belongs to graph estimation, not online block selection.

### B.4 Quantitative KL Bounds

The same dependence floor yields the fixed-budget lower bound and uniform learned-schedule comparison.

Fix $\Pi = (P_1, \dots, P_r)$, let $S_t = P_{<t}$, and write $\mathrm{comp}(K)$ for component vertex sets and $I_\mu(\cdot;\cdot \mid \cdot)$ for conditional mutual information under $\mu$. For every $C \in \mathrm{comp}(G - S_t)$, choose disjoint same-block pairs

$$M_{t,C} \subseteq \big\{\{u, v\} : u, v \in P_t \cap C,\ u \neq v\big\}$$

as a matching. Entropy subadditivity and Proposition 1 give

$$D_{\mathrm{KL}}(\mu \| \nu_\Pi) \geq \sum_{t=1}^{r} \sum_{C \in \mathrm{comp}(G - S_t)} \sum_{\{u,v\} \in M_{t,C}} I_\mu(X_u; X_v \mid X_{P_{<t}}). \tag{11}$$

Lemma 3 lower-bounds each pair's conditional mutual information using its shortest residual path.

For a schedule-uniform statement with $E \neq \emptyset$, set $J_{\min} := \min_{e \in E} |J_e|$ and $B_{\max} := \max_{i \in V}(|h_i| + \sum_{\{i,w\} \in E} |J_{iw}|)$, and define

$$\begin{aligned} \Lambda(G, \mathbf{J}, \mathbf{h}) &:= \frac{1}{2} e^{-4B_{\max}} \left[\frac{e^{-2B_{\max}}(1 - e^{-4J_{\min}})}{4}\right]^{2(n-1)}, \\ \mathfrak{v}_r(G) &:= \min_{\Pi:\, |\Pi| \leq r} \sum_{t=1}^{|\Pi|} \sum_{C \in \mathrm{comp}(G - P_{<t})} \left\lfloor \frac{|P_t \cap C|}{2} \right\rfloor. \end{aligned} \tag{12}$$

$B_{\max}$ bounds the absolute unary field after any boundary assignment. $\Lambda$ is a schedule-independent KL charge per collision, while $\mathfrak{v}_r$ is the fewest disjoint collisions forced within $r$ rounds. When $E = \emptyset$, every schedule is exact.

**Corollary 4** (Uniform positivity below tree depth)**.** *For every ferromagnetic or gauge-ferromagnetic scalar Ising model with at least one edge and every $1 \leq r \leq n$,*

$$\min_{\Pi:\, |\Pi| \leq r} D_{\mathrm{KL}}(\mu \| \nu_\Pi) \geq \Lambda(G, \mathbf{J}, \mathbf{h})\, \mathfrak{v}_r(G), \qquad \mathfrak{v}_r(G) = 0 \iff r \geq \mathrm{td}(G). \tag{13}$$

*In particular, every fixed schedule with $r < \mathrm{td}(G)$ incurs forward KL of at least $\Lambda(G, \mathbf{J}, \mathbf{h})$.*

For disjoint $A, S \subseteq V$, write

$$\omega_G(A \mid S) := \sum_{C \in \mathrm{comp}(G - S)} \left\lfloor \frac{|A \cap C|}{2} \right\rfloor.$$

Thus $\omega_G$ counts disjoint residual-component collisions in one block, and equation 12 minimizes their total over schedules using at most $r$ rounds.

*Proof of Corollary 4.* Fix a round $t$, let $S_t = P_{<t}$, and fix a boundary value $x_{S_t}$. For $A \subseteq P_t$, abbreviate the conditional law of $X_A$ by $\mu_A^{x_{S_t}} := \mu(X_A \in \cdot \mid X_{S_t} = x_{S_t})$. Partition the coordinates of $P_t$ into the disjoint pairs in $(M_{t,C})_C$ and the remaining singletons. Conditional entropy subadditivity gives the pointwise inequality

$$D_{\mathrm{KL}}\left(\mu_{P_t}^{x_{S_t}} \,\middle\|\, \prod_{i \in P_t} \mu_{\{i\}}^{x_{S_t}}\right) \geq \sum_{C \in \mathrm{comp}(G - S_t)} \sum_{\{u,v\} \in M_{t,C}} D_{\mathrm{KL}}\left(\mu_{\{u,v\}}^{x_{S_t}} \,\middle\|\, \mu_{\{u\}}^{x_{S_t}} \mu_{\{v\}}^{x_{S_t}}\right).$$

Taking expectation over $X_{S_t}$ turns each pair divergence into conditional mutual information. Summing over rounds then proves equation 11.

For the uniform constant, a shortest residual path has $\ell \leq n - 1$, each $b_i(P) \leq B_{\max}$, and each transformed coupling is at least $J_{\min}$. Hence

$$\begin{aligned} \Gamma(P) &\geq e^{-2B_{\max}} \left[\frac{e^{-2B_{\max}}(1 - e^{-4J_{\min}})}{4}\right]^{\ell} \\ &\geq e^{-2B_{\max}} \left[\frac{e^{-2B_{\max}}(1 - e^{-4J_{\min}})}{4}\right]^{n-1}, \end{aligned}$$

because the bracket lies in $(0, 1)$. Squaring and dividing by two gives $\Lambda(G, \mathbf{J}, \mathbf{h})$.

For every $t$ and component $C$, pairing the vertices of $P_t \cap C$ gives $\lfloor |P_t \cap C|/2 \rfloor$ disjoint pairs. The uniform bound therefore implies

$$D_{\mathrm{KL}}(\mu \| \nu_{\Pi}) \geq \Lambda(G, \mathbf{J}, \mathbf{h}) \sum_{t=1}^{|\Pi|} \omega_G(P_t \mid P_{<t}).$$

Minimizing over schedules with at most $r$ blocks proves the first claim of equation 13. The violation count is zero exactly for a schedule that is legal for every residual component. By Theorem 2, such a schedule exists with at most $r$ blocks exactly when $r \geq \mathrm{td}(G)$.

# C Finite-State and Algorithmic Evaluation

The evaluation covers exact schedule enumeration, planning-graph perturbations and recovery, scalable legal-schedule construction, randomized policies near the tree-depth boundary, and learned conditionals on small graphs. All exact-oracle calculations sum target and sampler probabilities over the complete state space rather than using Monte Carlo estimates. The pseudocode below records the construction and optimization routines; the surrounding paragraphs state the randomness, numerical tolerances, and learning configuration used in the reported comparisons.

## C.1 Exact-Conditionals Figure Protocol

Figure 5A uses the four 15-vertex targets from the exact-KL evaluation in Appendix D: one path, star, balanced binary tree, and random tree. All target and sampler probabilities are summed over the complete state space. The shorter control merges the last two tree-depth layers; the random control preserves the original layer sizes and randomizes vertex positions. The structural values depend only on the targets and schedules, so each target is plotted once.

For Figure 5B, the graph consists of $k$ disjoint edges with common coupling $J > 0$ and zero fields. Committing all vertices in one round replaces each edge's joint law by the product of its uniform one-spin marginals. Independence across edges gives

$$D_{\mathrm{KL}}(\mu \| \nu_{\text{one-round}}) = k\big[J \tanh J - \log \cosh J\big]. \tag{14}$$

The saved exact-enumeration grid uses $k \in \{1, 2, 4\}$ and $J \in \{0.1, 0.5, 0.8\}$. Enumerated and analytic KL agree within $10^{-14}$ nats at every point. The two-round control reveals one endpoint of every edge first and the other second; its KL is numerically zero.

## C.2 Exact Schedule Enumeration

Let $G = P_7$, set every edge coupling to $J_e = 0.5$, and set every field to zero. We enumerate all $2^7 = 128$ spin configurations and all 105 ordered partitions with block sizes $(1, 2, 4)$. For each schedule, we compute forward KL both from the target and sampler probability tables and as the sum of roundwise conditional total correlations in Proposition 1; the two calculations agree within $2 \times 10^{-14}$ nats.

| Schedule | Actual KL | Paired violations | Local certificate | $\Lambda\times$ count |
|---|---|---|---|---|
| Tree-depth schedule | 0 | 0 | 0 | 0 |
| Best nonexact | $2.1939\times10^{-2}$ | 1 | $1.9996\times10^{-5}$ | $3.5990\times10^{-21}$ |
| Largest KL | $3.9190\times10^{-1}$ | 3 | $4.0178\times10^{-3}$ | $1.0797\times10^{-20}$ |

Table 4: Exact enumeration of representative $(1, 2, 4)$ schedules on the seven-vertex path $P_7$. Paired violations count disjoint residual-component collisions. The local column sums $\Gamma(P)^2/2$ over those pairs as in equation 11; the final column applies the graph-wide constant.

The tree-depth row in Table 4 is $(\{4\}, \{2, 6\}, \{1, 3, 5, 7\})$, the only exact schedule among the 105. One best nonexact schedule is $(\{2\}, \{4, 6\}, \{1, 3, 5, 7\})$; its sole collision is connected by the residual path $4-5-6$. The largest value in the table is attained, for example, by $(\{1\}, \{2, 3\}, \{4, 5, 6, 7\})$. Across all schedules, the mean, median, and maximum KL are $0.1829$, $0.1691$, and $0.3919$ nats.

The local certificate retains the residual path and its parameters, whereas $\Lambda$ provides one guarantee uniform over schedules and boundaries.

This exact finite-state check isolates the theorem's Ising regime and verifies both the KL identity and the separation between schedule-local and graph-wide certificates.

### C.3 Graph Families and Planning-Graph Perturbations

The stress test covers six graph and parameter settings: the path $P_7$, star $S_7$, balanced binary tree $T_7$, cycle $C_7$, a $3 \times 3$ grid, and a balanced signed version of $T_7$. In lexicographic edge order, the coupling magnitudes cycle through $0.30, 0.38, 0.46, 0.54, 0.62, 0.70$; vertex fields cycle through $-0.18, -0.06, 0.06, 0.18$. The signed instance applies a fixed vertex gauge to both couplings and fields. Subset dynamic programming constructs a minimum-depth elimination forest, depth-first search (DFS) supplies a linear-time exact schedule, and a fixed seed (1729) supplies a matched-random schedule with the same block profile. The shorter schedule merges the deepest two optimal layers. Every optimal and DFS schedule has KL below $2 \times 10^{-11}$ nats.

| Graph | $n$ | Minimum/DFS rounds | KL (short) | KL (random) | Local certificate |
|---|---|---|---|---|---|
| Path $P_7$ | 7 | 3/7 | $4.003\times10^{-1}$ | $2.440\times10^{-1}$ | $2.706\times10^{-3}$ |
| Star $S_7$ | 7 | 2/2 | $6.604\times10^{-1}$ | $4.174\times10^{-1}$ | $1.246\times10^{-6}$ |
| Binary tree $T_7$ | 7 | 3/3 | $5.096\times10^{-1}$ | $3.447\times10^{-1}$ | $7.604\times10^{-4}$ |
| Cycle $C_7$ | 7 | 4/7 | $2.376\times10^{-1}$ | $1.045\times10^{-1}$ | $7.415\times10^{-4}$ |
| Grid $3 \times 3$ | 9 | 5/9 | $2.327\times10^{-1}$ | $2.835\times10^{-1}$ | $9.425\times10^{-5}$ |
| Signed binary tree | 7 | 3/3 | $5.096\times10^{-1}$ | $3.447\times10^{-1}$ | $7.604\times10^{-4}$ |

Table 5: Exact-oracle stress test across six graph families. "Short" merges the deepest two tree-depth layers; "random" preserves that schedule's block profile. The local certificate refers to the short schedule. The first round count is the minimum tree-depth schedule and the second is DFS; both are exact in every row, and Signed $T_7$ checks gauge invariance.

Table 5 shows positive factorization KL for every short and random schedule, while retaining zero KL at the predicted threshold across non-path graphs, heterogeneous parameters, and a gauge-equivalent signed model. The signed row isolates invariance under the gauge transform on the same dependency graph. The schedule-local certificate tracks the colliding residual path and its parameters.

To separate target specification from schedule construction, we fix the heterogeneous $T_7$ target and construct elimination forests from three planning graphs. Vertices are numbered in breadth-first order. All KL and certificate values are then evaluated under the same true target.

| Planning graph | Rounds | True-graph collisions | Actual KL | Local cert. | Global cert. |
|---|---|---|---|---|---|
| True binary tree | 3 | 0 | 0 | 0 | 0 |
| Tree $+\{4, 6\}$ | 4 | 0 | 0 | 0 | 0 |
| Tree $-\{1, 2\}$ | 2 | 1 | $5.237\times10^{-3}$ | $1.636\times10^{-8}$ | $1.601\times10^{-31}$ |

Table 6: Planning-graph perturbations for a fixed binary-tree target $(G, \mathbf{J}, \mathbf{h})$. Only the graph used to construct the schedule changes; collisions and all numerical quantities are evaluated on the true graph.

The added planning edge preserves exactness and increases the constructed schedule from three to four rounds. For the same fixed target, omitting the active edge $\{1, 2\}$ reduces the constructed schedule to two rounds and creates one true-component collision with positive KL.

### C.4 Planning-Graph Recovery

We test the conservative-graph workflow on two enumerated Ising targets by estimating the planning graph from samples. The targets are the heterogeneous 15-vertex path $P_{15}$ and $3 \times 3$ grid models, with the coupling and field patterns used above. For each sample of size $N$ and each vertex, we fit an $\ell_1$-regularized logistic neighborhood model with penalty $\lambda_N = \sqrt{\log(n)/N}$, include an intercept

feature scaled by 100, and OR-symmetrize the selected neighborhoods (take the union over both edge directions) (Ravikumar et al., 2010). The solver uses a $10^{-8}$ tolerance and at most 2,000 iterations. For every sample size, we run 20 independent replications, construct the LSE schedule defined in Section C.5 from each estimated graph, and evaluate its collisions and forward KL exactly under the true target. Finite-sample neighborhood-selection theory derives edge-recovery guarantees from degree, signal, and incoherence conditions (Ravikumar et al., 2010); Corollary 3 converts any resulting edge-coverage probability directly into an exact-schedule probability. Tree-structured Ising learning with latent variables or prediction-centric loss addresses different statistical targets (Kandiros et al., 2023; Boix-Adserà et al., 2021).

| Graph | $N$ | Recall | FP | Coverage | Rounds | Exact | Mean KL |
|---|---|---|---|---|---|---|---|
| Path $P_{15}$ | 100 | 0.736 | 0.60 | 0.00 | 3.50 | 0.00 | $7.883\times10^{-2}$ |
| Path $P_{15}$ | 250 | 0.989 | 0.25 | 0.85 | 4.35 | 0.85 | $2.563\times10^{-3}$ |
| Path $P_{15}$ | 500 | 1.00 | 0.10 | 1.00 | 4.10 | 1.00 | $< 2\times10^{-11}$ |
| Path $P_{15}$ | 1,000 | 1.00 | 0.25 | 1.00 | 4.35 | 1.00 | $< 2\times10^{-11}$ |
| Path $P_{15}$ | 2,000 | 1.00 | 0.20 | 1.00 | 4.15 | 1.00 | $< 2\times10^{-11}$ |
| Grid $3\times3$ | 100 | 0.842 | 2.65 | 0.05 | 5.50 | 0.50 | $9.879\times10^{-3}$ |
| Grid $3\times3$ | 250 | 0.983 | 2.50 | 0.80 | 6.05 | 0.95 | $8.080\times10^{-4}$ |
| Grid $3\times3$ | 500 | 1.00 | 2.05 | 1.00 | 5.95 | 1.00 | $< 2\times10^{-11}$ |
| Grid $3\times3$ | 1,000 | 1.00 | 2.10 | 1.00 | 6.15 | 1.00 | $< 2\times10^{-11}$ |
| Grid $3\times3$ | 2,000 | 1.00 | 1.80 | 1.00 | 5.75 | 1.00 | $< 2\times10^{-11}$ |

Table 7: Sample-to-schedule graph recovery over 20 replications. Recall, false positives (FP), full-edge coverage, layered-separator elimination (LSE) rounds, exact-schedule rate, and forward KL are means over 20 replications. Coverage and Exact are fractions; Exact means true-target KL below $2\times10^{-11}$ nats.

At $N = 500$, all 40 estimated graphs cover every true edge and every constructed schedule is exact. At smaller $N$, the grid's exact-schedule rate can exceed its full-coverage rate because edge coverage is sufficient, not necessary: a missed edge is harmless when the resulting schedule still avoids every true residual collision. False positives leave exactness intact and are reflected in the attained round count. Together with Table 6, this sweep tests both sides of the planning-graph asymmetry using a fixed estimator.

### C.5 Layered-Separator Elimination

Once a planning graph is available, the remaining task is to construct a legal schedule without exponential subset dynamic programming. Layered-separator elimination (LSE) recursively places a scored separator on one ancestor chain and processes the resulting components in parallel. The following rule records every candidate, score, ordering, and tie break used by the implementation.

Every returned forest is legal by construction, and this direct implementation has the conservative worst-case bound $O(n(n + |E|)\log n)$. The exact subset dynamic program is used only as an independent comparator. To reduce traversal-order sensitivity, multi-start DFS reports the shallowest forest from at most five deterministic starts—the four landmarks plus a double-sweep pseudo-center—and four neighbor orders based on vertex index or induced degree; index-order DFS uses one ascending-index traversal. On an $n$-vertex forest, every centroid step halves the largest residual component, so LSE uses at most $\lceil \log_2(n+1) \rceil$ rounds; paths attain this bound.

**Benchmark setup.** Grid vertices use row-major order and complete binary trees use breadth-first order. Random-tree instances use an i.i.d. uniform Prüfer sequence from a Mersenne–Twister generator seeded by 20,260,806. Sparse-$n$ starts from that tree and, with the same generator seeded by 20,260,807, adds $n/2$ distinct uniformly sampled nonedges. The circulant family is $C_n(1, s)$, with edges at cyclic offsets 1 and $s$, where $s = 3, 11, 23, 45$ for $n = 12, 128, 512, 2048$, respectively. Construction times are medians of three deterministic repeats. Exact subset dynamic programming is applied to the seven small controls; the larger rows report constructed height and direct legality checks, with path and cycle formulas and completed external controls providing the stated optima.

As an external exact control, we use the dependency-free Bute variant because it gives a lightweight, directly checkable optimum certificate (Trimble, 2020b); Bute-Plus is the corresponding published

**Pseudocode 1: Layered-separator elimination** $\mathrm{LSE}(H)$**.** *Input:* an induced residual graph $H$.

1. If $H$ is empty, return the empty forest. If it is disconnected, recursively process every component, merge equal-depth layers in parallel, and return.
2. If $H$ is a tree, set $A = \{v\}$, where $v$ is a centroid (a vertex whose deletion leaves no component larger than half the tree), breaking ties by vertex index; proceed to Step 5. Otherwise form up to four distinct landmarks: the smallest vertex, the maximum induced-degree vertex, and the two endpoints of a double sweep (two breadth-first search (BFS) passes that approximate the diameter); all ties use the smallest index.
3. For each landmark $r$, let $L_{r,d}$ be its BFS layer at distance $d$, and let $b^-_{r,d}$ and $b^+_{r,d}$ count vertices at smaller and larger distances. Select the layer with lexicographically smallest score
$$\big(|L_{r,d}| + m_{r,d},\ m_{r,d},\ |L_{r,d}|, |b^-_{r,d} - b^+_{r,d}|,\ d\big), \qquad m_{r,d} = \max\{b^-_{r,d}, b^+_{r,d}\}.$$
Use these layers and the singleton landmarks as candidates.
4. For each candidate $A$, let $M(A)$ be the largest component size in $H - A$. Choose the lexicographic minimum of
$$\big(|A| + M(A),\ M(A),\ |A|, \mathrm{sort}(A)\big).$$
5. Order $A$ by decreasing induced degree and then increasing index, place its vertices on a singleton ancestor chain, recursively process the components of $H - A$, and merge their layers in parallel below that chain.
6. Return the forest depth layers as the unmasking schedule.

| Graph | $n$ | $\lvert E\rvert$ | LSE | Multi-start DFS | Index-order DFS |
|---|---|---|---|---|---|
| Path | 15 | 14 | **4** | 8 | 15 |
| Cycle | 12 | 12 | **5** | 12 | 12 |
| Grid $3 \times 4$ | 12 | 17 | **7** | 9 | 12 |
| Binary tree | 15 | 14 | **4** | **4** | **4** |
| Random tree | 15 | 14 | **4** | 5 | 6 |
| Circulant | 12 | 24 | **7** | 12 | 12 |
| Sparse | 14 | 20 | **6** | 7 | 9 |

Table 8: LSE against DFS controls on seven small graphs. Bold marks the shallowest displayed construction in each row. Subset dynamic programming certifies LSE as optimal in six controls; on Grid $3 \times 4$, the exact depth is 6 and LSE returns 7.

practical solver (Trimble, 2020a). The exact control confirms the six LSE-optimal rows in Table 8 and returns depth 6 on Grid $3 \times 4$. On the larger structured controls, Bute certifies depths 7, 8, and 7 for $P_{127}$, $C_{128}$, and the 127-vertex binary tree, respectively, matching LSE in 35.0, 136.7, and 7.7 ms. Under a five-second cap it returns no certificate for Grid $8 \times 8$, for which LSE produces a directly verified height-17 schedule. All returned forests pass independent height and edge-legality checks. The two solvers are complementary: Bute certifies minimum height on completed instances, while LSE directly constructs a legal schedule.

LSE reaches the optimum in six of seven small controls and has ratio $7/6$ on Grid $3 \times 4$. Across all 28 instances, its median round reduction is $63.6\%$ relative to multi-start DFS and $71.8\%$ relative to index-order DFS; it is shallower in 25 instances and ties in three. Every constructed output passes ordered-partition, forest-height, edge ancestor–descendant, and residual-component checks. The largest observed LSE construction time across the 28 instances is $0.139$ seconds.

### C.6 Randomized-Policy Evaluation

The preceding schedule constructions are deterministic. Whether randomization can cross the same round boundary on finite graphs is a separate question. Conditioning on a random seed reduces seed-wise exactness to the deterministic theorem, whereas the following finite optimization evaluates the broader marginal output criterion over randomized policies. Fix a budget of at most $R$ rounds and let $s = (t, S, x_S)$ be a nonterminal state. An action is a nonempty block $B \subseteq V \setminus S$; at $t = R - 1$, the only action is $B = V \setminus S$, which enforces completion. The transition kernel is $K(y_B \mid s, B) = \prod_{i \in B} \mu(X_i = y_i \mid X_S = x_S)$. Extend $K$ to the induced successor-state kernel and set it to zero unless the indicated transition is feasible; for a completing action, write $K(x \mid s, A)$ for

| Graph | $n$ | $\lvert E\rvert$ | LSE | Multi-start DFS | Index-order DFS |
|---|---|---|---|---|---|
| Path (127) | 127 | 126 | **7** | 64 | 127 |
| Path (511) | 511 | 510 | **9** | 256 | 511 |
| Path (2,047) | 2,047 | 2,046 | **11** | 1,024 | 2,047 |
| Cycle (128) | 128 | 128 | **8** | 128 | 128 |
| Cycle (512) | 512 | 512 | **10** | 512 | 512 |
| Cycle (2,048) | 2,048 | 2,048 | **12** | 2,048 | 2,048 |
| Grid $8 \times 8$ | 64 | 112 | **17** | 59 | 64 |
| Grid $16 \times 16$ | 256 | 480 | **41** | 243 | 256 |
| Grid $32 \times 32$ | 1,024 | 1,984 | **91** | 995 | 1,024 |
| Binary tree (127) | 127 | 126 | **7** | **7** | **7** |
| Binary tree (511) | 511 | 510 | **9** | **9** | **9** |
| Binary tree (2,047) | 2,047 | 2,046 | **11** | **11** | **11** |
| Random tree (128) | 128 | 127 | **7** | 16 | 22 |
| Random tree (512) | 512 | 511 | **9** | 29 | 35 |
| Random tree (2,048) | 2,048 | 2,047 | **11** | 77 | 92 |
| Circulant (128) | 128 | 256 | **38** | 124 | 128 |
| Circulant (512) | 512 | 1,024 | **85** | 503 | 512 |
| Circulant (2,048) | 2,048 | 4,096 | **206** | 2,025 | 2,048 |
| Sparse (128) | 128 | 191 | **42** | 47 | 68 |
| Sparse (512) | 512 | 767 | **171** | 196 | 263 |
| Sparse (2,048) | 2,048 | 3,071 | **694** | 756 | 1,065 |

Table 9: LSE and DFS schedule depths on larger graphs. Bold marks the lowest reported constructed depth in each row; ties are all bold. Every returned forest passes the residual-component legality check. Path and cycle values match their standard optima; the remaining rows report constructed heights.

its terminal-state probability. Let $s_0 = (0, \emptyset, \emptyset)$ be the initial state. The state–action flow $z(s, B) \geq 0$ is the probability mass of reaching $s$ and choosing $B$. Root normalization and conservation are

$$\sum_B z(s_0, B) = 1, \qquad \sum_B z(s, B) = \sum_{s', A} z(s', A) K(s \mid s', A)$$

for every later nonterminal state. The terminal law and its $\ell_1$ linearization are

$$q(x) = \sum_{s,A} z(s, A) K(x \mid s, A), \qquad q(x) - e_x^+ + e_x^- = \mu(x), \qquad \min \sum_x (e_x^+ + e_x^-), \qquad e_x^\pm \geq 0.$$

Normalizing the outgoing flows at each reachable state recovers a randomized behavioral policy, meaning a state-dependent distribution over next blocks. Conversely, $(t, S, x_S)$ is a controlled Markov state, so this occupancy-flow representation preserves the terminal law of every full-history randomized monotone policy.

**Pseudocode 2: Randomized-policy threshold check.** *Input:* an exact target $\mu$, graph $G$, and round budget $R$.

1. Enumerate every reachable state $(t, S, x_S)$; enumerate every nonempty block of unrevealed vertices, forcing the final action to reveal all that remain.
2. Evaluate each transition $K(y_B \mid s, B)$ from the exact one-coordinate conditionals and create one nonnegative flow variable per state–action pair.
3. Add root normalization, flow conservation, and the terminal-law equalities above; introduce $e_x^+, e_x^-$ to linearize terminal $\ell_1$ error.
4. Solve the sparse linear program and report its terminal $\ell_1$ objective. Repeat for $R = \mathrm{td}(G) - 1$ and the $R = \mathrm{td}(G)$ control.

Target masses and transitions are evaluated by complete state-space summation in double precision. The sparse LP uses HiGHS with primal and dual feasibility tolerances $10^{-9}$ and interior-point

optimality tolerance $10^{-10}$; a terminal objective below $10^{-8}$ is reported as exact. The uniform parameterization sets $J_e = 0.5$ and $h_i = 0$. For the heterogeneous parameterization, let $g$ be the canonical graph-atlas index in the standard enumeration of connected unlabeled graphs, and let $e_k$ be the $k$-th edge in lexicographic order, both zero based. We set

$$J_{e_k} = 0.25 + 0.10\big((g+k) \bmod 5\big), \qquad h_i = (-0.30, -0.15, 0, 0.15, 0.30)_{(g+i) \bmod 5}.$$

We test all 30 connected unlabeled graphs with $2 \le n \le 5$, each with a uniform and a heterogeneous strictly ferromagnetic parameterization. Thus the optimization covers all randomized value-adaptive monotone policies within each of 60 finite instances, including mixtures of fixed schedules chosen before decoding. We report terminal $\ell_1$ error and total variation (TV), which is one half of that error.

None of the 60 $\mathrm{td}(G)-1$-round optimizations is exact. The smallest terminal $\ell_1$ error is $4.0504\times10^{-3}$, or $2.0252\times10^{-3}$ in TV, and occurs for the uniform complete graph $K_5$. All 60 $\mathrm{td}(G)$-round controls recover their target distributions, with maximum terminal coordinate residual $2.06\times10^{-13}$. The sub-threshold minimum therefore remains more than ten orders of magnitude above the largest control residual. These computations provide a finite-state randomized-policy comparison alongside the deterministic guarantee in Theorem 2.

### C.7 Learned Conditionals on Small Graphs

We examine how factorization and conditional-model error contribute to the total KL of learned samplers. The target is the zero-field 15-vertex path $P_{15}$ Ising model with $J_e = 0.5$, evaluated exactly over all $2^{15}$ states. For training seeds $0, \dots, 4$, we draw 50,000 target samples per seed with replacement and train one shared masked multilayer perceptron with dimensions $2n \to 64 \to 64 \to n$ and a SiLU activation after each hidden layer. Training uses 3,000 AdamW steps with batch size 512, constant learning rate $2 \times 10^{-3}$, weight decay $10^{-5}$, $(\beta_1, \beta_2) = (0.9, 0.999)$, and $\epsilon = 10^{-8}$. The input concatenates the observed spins and their binary mask, and the $n$ logits give all one-coordinate conditionals. Observed-set cardinalities are sampled uniformly from 0 through $n-1$; conditional on the cardinality, the observed subset is uniform. Binary cross-entropy is summed over masked coordinates and normalized by their total count in the batch. We clip query probabilities to $[0.01, 0.99]$ and draw 100,000 independent validation states after fitting. All reported learned-sampler probabilities, population risks, total KL values, held-out losses, and comparison statistics use these clipped conditionals. Clipping is part of the evaluated sampler and enforces the probability floor $\alpha = 0.01$ used by Corollary 2.

**Pseudocode 3: Masked conditional training and evaluation.**

1. Draw training states from $\mu$. At each update, sample a batch, then sample an observed-set size and a uniformly random subset of that size.
2. Feed the masked spins and mask to the shared network, average binary cross-entropy over unrevealed coordinates, and take one AdamW step.
3. After 3,000 steps, fix the network and cache its clipped conditionals for every context queried by the pre-specified schedules.
4. Sum over the complete target state space for factorization KL, model risk, and total KL. Draw validation states only after fixing the network and apply the simultaneous held-out comparison.

The six schedules are fixed before training. The tree-depth schedule uses median elimination on $P_{15}$, and the sequential control reveals one coordinate per round. Three-round, two-round, and single-round variants coarsen successive tree-depth layers. The matched-random schedule uses the $(1, 2, 4, 8)$ block profile with a random permutation fixed by seed 314159. The main-text control in Table 3 reports exact factorization KL, exact population model risk, and their sum from Proposition 2. The two-round layer merge has factorization KL 1.2730, model risk 0.0063, and total KL 1.2793. The single-round block has factorization KL 1.5532, model risk 0.0019, and total KL 1.5551.

To test whether learned confidence can replace the graph-derived order, we also evaluate matched confidence, a deterministic value-adaptive policy with the same $(1, 2, 4, 8)$ profile. At each history it ranks remaining positions by $\max\{q_i(+1 \mid x_S), q_i(-1 \mid x_S)\}$, equivalently by increasing binary entropy, with index tie breaking. The ranking reuses the current one-coordinate predictions, so the policy still takes four evaluations; complete-state enumeration verifies normalization of the adaptive

sampler. Its mean total forward KL is $0.8526$ over five fitted networks, compared with $0.0301$ for the tree-depth schedule.

On this same path target, the schedule-local certificates for the three-round merge, two-round merge, single-round block, and matched-random schedule are respectively $7.180 \times 10^{-3}$, $8.036 \times 10^{-3}$, $8.464 \times 10^{-3}$, and $5.302 \times 10^{-3}$ nats. They retain each collision's path and parameter information. All four fixed path candidates clear the simultaneous held-out comparison; the $5/5$ entries in Table 10 record the corresponding non-path results.

For this $P_{15}$ target, the uniform three-round floor $\Lambda\mathfrak{v}_3$ is $1.036 \times 10^{-45}$ nats. The population certificate in Corollary 1 requires model risk below this floor, compared with the measured mean risk of $0.0301$. The four fixed-candidate comparisons use Corollary 2: their mean held-out gaps of $0.7130$–$1.5241$ nats exceed the simultaneous radius of $0.6453$ nats.

**Binary-tree and grid results.** We repeat the experiment on the heterogeneous seven-vertex binary tree and $3 \times 3$ grid from Table 5, using the same training and clipping protocol. Evaluation sums over all $2^7$ or $2^9$ states, followed by 500,000 independent validation states per run. Alongside the tree-depth reference and sequential control, the pre-specified candidates are a one-round-short schedule and a matched-random order from seed 1729; the simultaneous test uses $\alpha = 0.01, \delta = 0.05$.

| Schedule | Rounds Tree, Grid | Factor KL Tree, Grid | Model risk Tree, Grid | Total KL Tree, Grid | Held-out Tree, Grid |
|---|---|---|---|---|---|
| | 3 | 0 | 0.0053 | 0.0053 | reference |
| Tree-depth | 5 | 0 | 0.0157 | 0.0157 | reference |
| | 2 | 0.5096 | 0.0025 | 0.5121 | 5/5 |
| One round shorter | 4 | 0.2327 | 0.0112 | 0.2439 | 5/5 |
| | 3 | 0.3447 | 0.0077 | 0.3524 | 5/5 |
| Matched random | 5 | 0.2835 | 0.0178 | 0.3013 | 5/5 |

Table 10: Mean learned-conditional results on the binary-tree and grid targets. Each cell lists the tree first and grid second. Factorization KL is exact; model risk and total KL are averaged across fitted networks. Held-out comparisons are simultaneous over the two candidates within each run and target.

The exact sequential controls have zero factorization KL and mean total KL $0.0088$ on $T_7$ and $0.0200$ on the grid. For the short and random candidates, respectively, mean held-out gaps are $0.5078$ and $0.3479$ nats on $T_7$, compared with a simultaneous radius of $0.1236$; on the grid they are $0.2281$ and $0.2863$ nats, compared with radius $0.1589$. Both fixed comparisons clear the pre-specified simultaneous threshold.

**Model risk across exact schedules.** The preceding comparisons vary factorization error. To isolate schedule-dependent model risk among structurally exact alternatives, we enumerate the 48 distinct minimum-height elimination-forest layerings produced by the exact tree-depth recurrence for the heterogeneous $3 \times 3$ grid. Every layering uses five rounds and has factorization KL below $2 \times 10^{-10}$. We reuse the five trained grid networks above and sum over the complete state space to evaluate every schedule. Because all factorization terms vanish, the entries below are both learned risk and total forward KL.

| Statistic | Across-network schedule mean |
|---|---|
| Minimum | 0.01319 |
| Median | 0.01561 |
| Maximum | 0.01807 |
| Max/min | 1.37 |

Table 11: Learned risk across 48 exact five-round grid schedules. Each schedule is first averaged over the five fitted networks, after which the listed statistics are computed across schedules.

The best and worst schedules after averaging over the five fitted networks have risks $0.01319$ and $0.01807$, respectively. The tree-depth schedule used in Table 10 has mean risk $0.01573$ and ranks 27th of 48. Thus equal round count and exactness do not determine learned-model error: the queried contexts still select among valid tree-depth schedules, as captured by equation 4.

# D MASKED-MODEL EXPERIMENTS

## D.1 EXPERIMENTAL SETUP

The synthetic corpus contains 742,400 states from 1,450 Ising instances on distinct graphs (Table 12). Random trees are uniform labeled trees; couplings and fields are drawn independently from $\mathrm{U}[0.10, 0.80]$ and $\mathrm{U}[-0.25, 0.25]$. Each graph supplies 512 exact states, sampled by sum-product on trees and enumeration otherwise. Evaluation selects five graphs per family-size cell for ID and topology OOD (75 and 40 graphs), and three per family at $n = 127$ for scale OOD (nine graphs). ID and topology OOD use four states per graph, scale OOD one.

| **Graph family** | $n$ | **Train** | **Val.** | **ID** |
|---|---|---|---|---|
| Path | 15, 31, 63 | 40 | 10 | 20 |
| Star | 15, 31, 63 | 20 | 0 | 10 |
| Balanced binary tree | 15, 31, 63 | 40 | 10 | 20 |
| Random tree | 15, 31, 63 | 80 | 20 | 40 |
| Ladder | 8, 12, 16 | 20 | 0 | 20 |

| **Shift** | **Graph family** | $n$ | **OOD** |
|---|---|---|---|
| Topology | Cycle, wheel | 8, 12, 16 | 20 |
| Topology | Grid | 9, 16 | 20 |
| Scale | Path, balanced binary tree | 127, 255, 511 | 20 |
| Scale | Random tree | 127, 255, 511 | 40 |

Table 12: **Corpus splits.** Counts per graph family and listed size.

We use LLaDA-8B-Instruct (revision `08b83a6feb34`) and Dream-v0-Instruct-7B (revision `05334cb9faaf`) (Nie et al., 2025; Ye et al., 2025). The denoiser receives a graph prefix followed by one token per spin in vertex order. Topology uses family identifiers for paths, stars, and balanced binary trees, parent lists for random trees, and edge lists otherwise. Couplings and fields are rounded to four decimals. Spins $-1, +1$ use tokens `0`, `1`; unknown spins use the backbone's mask token. For logits $z_{i,0}, z_{i,1}$ at spin slot $i$, binary readout is

$$q_i(+1 \mid x_S, S) = \mathrm{clip}_{[\epsilon, 1-\epsilon]} \frac{e^{z_{i,1}}}{e^{z_{i,0}} + e^{z_{i,1}}}, \qquad \epsilon = 10^{-6},$$

with $q_i(-1 \mid x_S, S) = 1 - q_i(+1 \mid x_S, S)$. The planner uses adjacency; the denoiser also sees target parameters and observed values.

Low-rank adaptation (LoRA) covers attention and feed-forward projections with rank 16, alpha 32, and dropout 0.05. Each fit uses 400 AdamW updates, batch size eight, constant learning rate $2 \times 10^{-4}$, weight decay 0.01, and gradient-norm clipping at 1, on 3,200 fresh states from 600 graphs. Both objectives use vocabulary cross-entropy on masked spin slots. Planner-aware training samples a legal prefix in proportion to its next block's squared size and doubles that block's loss weight. Uniform-mask training samples the observed-set size uniformly from $0, \ldots, n-1$, then a uniform subset of that size.

**Dependency-guided policies.** We adapt PUNT, DEMASK, and DOS to the same binary readout and shared adapters (Azangulov et al., 2026; Ringel et al., 2026; Zhou et al., 2026). PUNT uses sampled-anchor conditional-independence tests; DEMASK predicts pairwise influence from final hidden states; DOS ranks positions from attention. TD/LSE uses graph adjacency directly, while these selectors infer dependencies from model signals. Each DEMASK head trains for five epochs on five masks per graph from the 600 training graphs; 120 validation graphs select the checkpoint. PUNT tests thresholds in $\{0.001, 0.004, 0.01, 0.04, 0.16, 0.64\}$; DEMASK uses dependency thresholds in $\{0.01, 0.04, 0.16, 0.64\}$ and confidence thresholds in $\{0.5, 0.7, 0.9\}$. DOS-EB selects among five evenly spaced attention layers and entropy budgets $\{0.01, 0.5, 2, 8\}$. A family-size screen retains three configurations per method for full validation. We choose the fastest configuration within 0.01 nats per coordinate of sequential validation NLL, or the lowest-NLL configuration if none qualifies. DOS-matched reuses the chosen attention layer and the TD/LSE block sizes. Selection uses only validation graphs and is repeated for each fitted adapter. The nine-policy comparisons use batch-one inference with a bfloat16 backbone and float64 LoRA contractions, identically across policies.

The ID estimand is mean coordinate NLL conditioned on true earlier reveals, evaluated on four target states per graph and averaged over three training seeds. For randomized probe-based selectors, this is a conditional-prediction score along the sampled reveal order. Paired NLL differences average states and seeds within each graph before 5,000 percentile bootstrap resamples of the 75 graph means (Table 13). Timing repeats are reduced to a median per record before averaging states, graphs, and adapter seeds. OOD, systems, and free generation use the planner-aware seed-zero adapter; graph-bootstrap intervals condition on the fitted adapters. The pre-specified noninferiority rule declares TD/LSE noninferior when the lower endpoint of the 95% graph-bootstrap interval for baseline-minus-TD/LSE NLL is at least $-0.01$ nats per coordinate.

| | LLaDA ΔNLL | | | Dream ΔNLL | | |
|---|---|---|---|---|---|---|
| **Baseline minus TD/LSE** | **Mean** | **Lower** | **Upper** | **Mean** | **Lower** | **Upper** |
| Sequential | +0.0133 | +0.0096 | +0.0172 | +0.0170 | +0.0132 | +0.0209 |
| Matched random | +0.0293 | +0.0205 | +0.0393 | +0.0312 | +0.0221 | +0.0412 |
| Matched confidence | +0.0256 | +0.0191 | +0.0326 | +0.0296 | +0.0228 | +0.0371 |
| Dynamic confidence | +0.0134 | +0.0094 | +0.0176 | +0.0176 | +0.0141 | +0.0214 |
| PUNT | +0.0229 | +0.0178 | +0.0282 | +0.0255 | +0.0206 | +0.0305 |
| DEMASK | +0.0176 | +0.0133 | +0.0222 | +0.0171 | +0.0132 | +0.0211 |
| DOS-EB | +0.0178 | +0.0142 | +0.0216 | +0.0196 | +0.0156 | +0.0239 |
| DOS-matched | +0.0291 | +0.0217 | +0.0371 | +0.0250 | +0.0185 | +0.0322 |

Table 13: **NLL differences from TD/LSE.** Values (nats/coordinate) are baseline minus TD/LSE under planner-aware training, averaged over three seeds; positive values favor TD/LSE. Lower and Upper bound 95% graph-bootstrap intervals with fitted adapters fixed.

### D.2 Latency and Scaling

All policies in a comparison use matched execution settings. End-to-end times include online probes, dependency features, selection, and planning amortized over the states of the same graph. Offline head training is separate. Planner-aware seed-zero ID timing repeats the first state of each graph ten times; the other states and adapter cells use one measurement. All reported means give states, graphs, and adapter seeds equal weight after repeat reduction.

The affine experiment uses three independent process starts and path, balanced-binary-tree, and random-tree graphs at $n \in \{31, 63, 127\}$, five fixed round budgets per cell, and ten synchronized repeats per budget. Figure 3 compares the observed wall times with within-cell affine fits. This fixed-work control and the refinement and exact-KL controls retain their original measurements; the nine-policy comparisons are remeasured together.

The separate systems suite uses homogeneous targets on three families and $n \in \{127, 255, 511\}$, for nine graph cells per model and three repeats. At each size, Figure 3B divides mean sequential time by mean policy time over the two backbones and three graph families. Homogeneous parameters have a shorter serialization than the heterogeneous scale-OOD targets, so the two suites measure different input workloads.

### D.3 Ablation Studies

Dynamic confidence ranks unrevealed positions by $\max\{q_i(+1), q_i(-1)\}$, commits those at or above 0.8 up to the largest TD/LSE block size for that graph, and breaks ties by vertex order. If none passes, it commits the most confident position. This threshold is fixed across the reported comparisons.

Table 14 collects the dedicated ablation runs, with their matched TD/LSE reference and structural checks.

Additional dependence-regime tests use 20 graphs per cell. Under planner-aware training, removing one round raises mean NLL across backbones by $+0.0466$ at $J = 0.5$ and $+0.0680$ at $J = 0.8$. Every shortened schedule is structurally illegal, so the sign of a learned NLL change does not test legality.

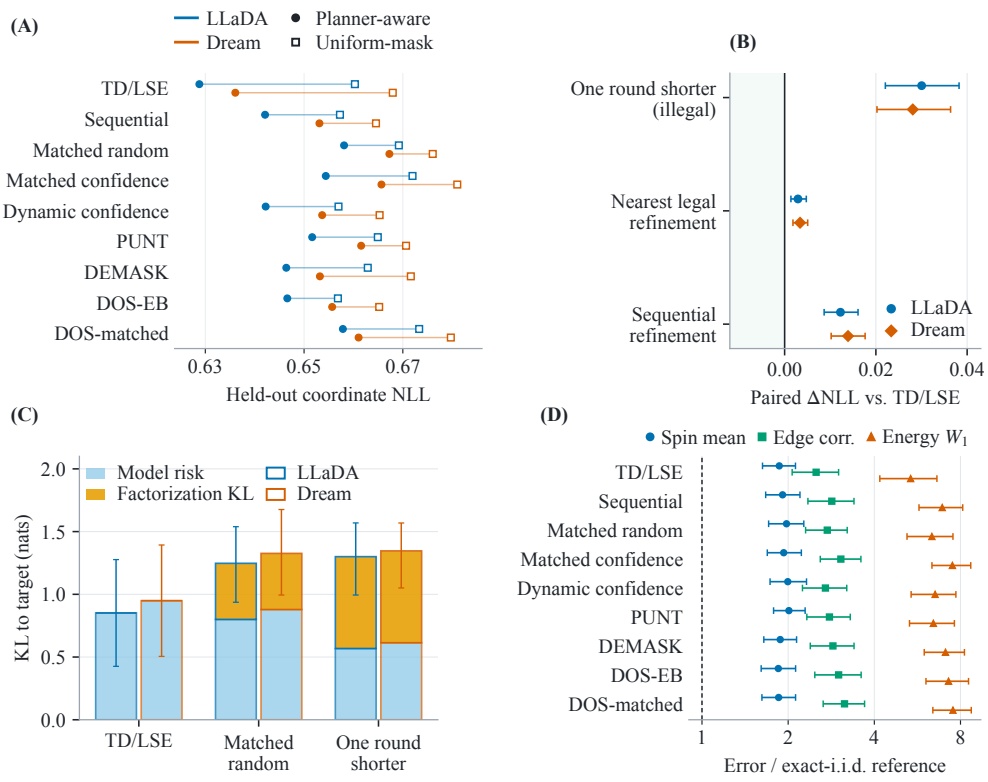


Figure 6: **Training and schedule effects.** **(A)** Nine-policy NLL averaged over three adapters: filled markers use planner-aware training; open markers use uniform masks. **(B)** Three-run NLL changes from TD/LSE under planner-aware training, for illegal shortening and two legal refinements. The nearest exact refinement is the first pre-specified grid point above TD/LSE (median $+4$ rounds); sequential uses $n$ rounds. **(C)** Exact joint KL on four $n = 15$ graph families, with 95% graph-bootstrap intervals. **(D)** Errors divided by the matched exact-target i.i.d. reference for finite-sample error, averaged over backbones within graph, then over graphs. The 95% graph-bootstrap bars condition on fitted adapters, generated sample sets, and reference estimates.

| | Planner-aware NLL | | Uniform-mask NLL | | | | |
|---|---|---|---|---|---|---|---|
| **Variant** | **LLaDA** | **Dream** | **LLaDA** | **Dream** | **Round change** | **Residual legal** | **True edges covered** |
| **TD/LSE** | **0.6289** | **0.6364** | 0.6604 | 0.6681 | 0 | 75/75 | 75/75 |
| One fewer round | 0.6590 | 0.6645 | 0.6721 | 0.6765 | −1.00 | 0/75 | 75/75 |
| Sequential | 0.6420 | 0.6533 | **0.6574** | **0.6642** | +27.07 | 75/75 | 75/75 |
| Extra planning edge | 0.6299 | 0.6366 | 0.6600 | 0.6674 | +0.60 | 75/75 | 75/75 |
| Missing true edge | 0.6353 | 0.6435 | 0.6600 | 0.6678 | −0.13 | 0/75 | 0/75 |

Table 14: **Schedule and planning-graph ablations.** NLL averages three adapter runs. Each ablation adds its graph-paired change to this cohort's model- and objective-matched TD/LSE mean. Mean round changes are relative to TD/LSE; legality and edge coverage count the 75 ID graphs.

### D.4 Exact KL Decomposition

Across the independent adapter runs, we enumerate all $2^{15}$ states of one path, star, balanced-binary-tree, and random-tree target. The structural term is model- and run-invariant: numerically zero for TD/LSE and positive for both residual-illegal alternatives (Table 15). The maximum absolute residual in $\mathrm{KL}_{\mathrm{total}} - \mathrm{KL}_{\mathrm{factor}} - \mathrm{Risk}_q$ over 72 cells is $8.9 \times 10^{-16}$.

| Schedule | Factor KL | Mean model risk | Mean total KL |
|---|---|---|---|
| **TD/LSE** | 0.000 | 0.900 | **0.900** |
| Matched random | 0.447 | 0.839 | 1.287 |
| One round shorter | 0.732 | 0.591 | 1.323 |

Table 15: **Factorization error and learned-model risk.** Forward KL is in nats per joint $n = 15$ sample. Model risk and total KL average LLaDA and Dream, four fixed graph families, and independent adapter runs.

### D.5 Out-of-Distribution Evaluation

All nine policies are evaluated on both backbones in the 40-graph topology-OOD suite and the nine-graph $n = 127$ scale-OOD suite (Table 2). On unseen topologies, TD/LSE has the lowest mean NLL for Dream; the best LLaDA result is within 0.0011 nats per coordinate of TD/LSE. DOS-EB is faster on Dream topology OOD, but with higher NLL. At $n = 127$, TD/LSE has the lowest mean NLL for both models. The graph-based schedule transfers without refitting a dependency predictor.

### D.6 Sample Quality

For each of 30 matched ID and topology-OOD graphs with $n \leq 16$, every learned policy generates one 32-sample set per graph. A separate exact sampler averages 100 independent 32-sample draws per graph. Spin mean absolute error (MAE) averages absolute errors in node means; edge MAE does the same for edge correlations. Energy $W_1$ is the 1-Wasserstein distance between the generated and exact energy distributions. Table 16 separates marginal and dependence errors. DOS-EB has the lowest pooled spin MAE, while TD/LSE has the smallest edge-correlation and energy errors among the learned policies for both backbones. Matching single-spin means therefore does not imply matching the dependencies that define the target joint law. Table 17 gives paired error differences from the same generated samples. We average the two backbone differences within each graph and form 95% percentile intervals from 10,000 resamples of the 30 matched graphs, holding the fitted adapters fixed.

| Policy | Spin MAE | Edge MAE | Energy $W_1$ | Pooled samples/s |
|---|---|---|---|---|
| **TD/LSE** | 0.2556 | **0.2807** | **2.3090** | 1.228 |
| Sequential | 0.2619 | 0.3187 | 2.9385 | 0.459 |
| Matched random | 0.2706 | 0.3087 | 2.7043 | 1.228 |
| Matched confidence | 0.2642 | 0.3445 | 3.1459 | 1.227 |
| Dynamic confidence | 0.2730 | 0.3035 | 2.7617 | 0.459 |
| PUNT | 0.2763 | 0.3131 | 2.7351 | 0.332 |
| DEMASK | 0.2573 | 0.3220 | 3.0131 | 0.576 |
| DOS-EB | **0.2532** | 0.3359 | 3.0621 | 0.686 |
| DOS-matched | 0.2540 | 0.3545 | 3.1730 | 1.225 |

Exact i.i.d. reference: spin MAE 0.1383; edge MAE 0.1183; energy $W_1$ 0.3895.

Table 16: **Recovering marginals and dependencies.** Each policy generates 32 samples per graph. MAE denotes mean absolute error; $W_1$ is the 1-Wasserstein distance between energy distributions. Learned rows average LLaDA and Dream with one sample set per graph; the exact-target i.i.d. reference averages 100 independent sets and measures finite-sample error. Bold marks the lowest learned-policy error. Throughput is total samples divided by total sampling time, excluding precomputed planning.

| | ΔEdge MAE | | | ΔEnergy $W_1$ | | |
|---|---|---|---|---|---|---|
| **Baseline minus TD/LSE** | **Mean** | **Lower** | **Upper** | **Mean** | **Lower** | **Upper** |
| Sequential | +0.0379 | +0.0220 | +0.0542 | +0.6294 | +0.4630 | +0.8040 |
| Matched random | +0.0279 | +0.0112 | +0.0455 | +0.3953 | +0.2057 | +0.5978 |
| Matched confidence | +0.0638 | +0.0462 | +0.0819 | +0.8369 | +0.6234 | +1.0612 |
| Dynamic confidence | +0.0227 | +0.0066 | +0.0394 | +0.4527 | +0.3046 | +0.6028 |
| PUNT | +0.0324 | +0.0123 | +0.0533 | +0.4261 | +0.2293 | +0.6322 |
| DEMASK | +0.0413 | +0.0252 | +0.0580 | +0.7040 | +0.5091 | +0.9071 |
| DOS-EB | +0.0551 | +0.0384 | +0.0721 | +0.7530 | +0.5832 | +0.9271 |
| DOS-matched | +0.0738 | +0.0549 | +0.0932 | +0.8640 | +0.6282 | +1.1109 |

Table 17: **Generation-error differences from TD/LSE.** Differences (baseline minus TD/LSE) average both backbones within the 30 graphs in Table 16; positive values favor TD/LSE. Lower and Upper bound 95% paired graph-bootstrap intervals with adapters and generated samples fixed.