SCALABLE GAME-THEORETIC GRAPH DATA VALUATION VIA COORDINATED STRUCTURAL REUSE

Anonymous authors
Paper under double-blind review

ABSTRACT

As large-scale graph data becomes increasingly prevalent, assessing the quality and value of individual data points is critical for model developers and data marketplaces alike. Existing SOTA data valuation methods, such as Data Shapley and DataModels, are designed for i.i.d. settings and fail to capture the structural dependencies inherent in graph data. Precedence-Constrained Winter Value (PC-Winter) represents the first SOTA framework for graph-aware valuation; however, two shortcomings limit its effectiveness. First, its Monte Carlo estimator samples nodes without coordination within a labeled-node group, producing context-misaligned estimates that inflate the noise on intra-group members and harm ranking-based tasks. Second, while its algorithmic pipeline exploits some structural redundancy, substantial reuse opportunities remain unexploited in both signal propagation and classifier training. We address both shortcomings under a unifying principle, *Maximal Structural Reuse* (MSR), which exploits the structural commonalities across players at every component of the pipeline. Accordingly, we introduce **SGVR** (**S**calable **G**raph Data **V**aluation via Coordinated Structural **R**euse), built on three contributions: i) a structure-aware estimator based on hierarchical coordinated sampling across the members of each labeled-node group; we derive its exact variance when comparing nodes within the same group, which contracts in proportion to the covariance that coordination induces through shared context, and confirm this contraction empirically; ii) an optimal propagation procedure that updates only the nodes affected by each incremental change; and iii) a recursive least squares update rule that incrementally refines reusable model weights at negligible cost. Experiments on six standard benchmarks confirm that SGVR outperforms existing baselines on the unlabeled node selection task, with speedups that grow with graph size and reach two orders of magnitude on the largest benchmarks. Code and experiments will be publicly released upon publication.

1 INTRODUCTION

The rapid growth of unstructured data across science, social networks, and industry has made understanding the *value* of individual data points increasingly critical (Pei, 2020). Data valuation, the problem of quantifying how much each training instance contributes to model performance, has significant practical implications: it enables dataset pruning, mislabel detection, fair compensation in data markets, and robust model debugging (Ghorbani & Zou, 2019; Wang & Jia, 2023; Ilyas et al., 2022).

Graph-structured data represents a fundamentally richer modality than tabular or Euclidean data, encoding relational dependencies between entities that classical machine learning cannot directly exploit (Ju et al., 2026). Graph Neural Networks (GNNs) were developed to leverage this structure, learning node representations by iteratively aggregating information from neighboring nodes (Bruna et al., 2013; Kipf & Welling, 2016; Wu et al., 2019). This neighborhood aggregation violates the independent and identically distributed (i.i.d.) assumption underlying classical machine learning, i.e., nodes are not independent, and, as such, both labeled and unlabeled nodes participate in training in complementary ways; an unlabeled node carries an intrinsic structural signal that can provide an implicit supervision via indirect network structure (Ju et al., 2026).

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

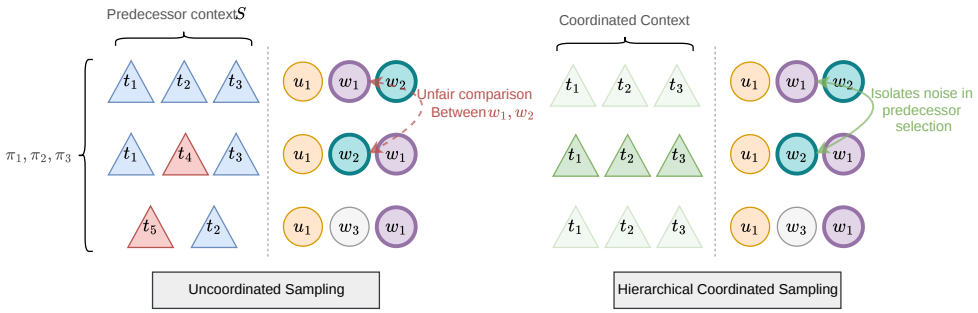


Figure 1: Uncoordinated against coordinated sampling for the target pair $\{w_1, w_2\}$. Each row is a sampled permutation; a sequence of triangles (subtrees t_i) forms the predecessor context, and circles mark the labeled node u_1 and the targets compared. **(Left)** PC-Winter draws π_1, π_2, π_3 independently, and as such w_1 and w_2 are compared under differing contexts and absorb sampling noise. **(Right)** Coordinated sampling fixes a shared context while permuting only the focal block, isolating the ranking signal.

The dominant data valuation paradigm assumes i.i.d. samples, estimating values via frameworks such as Data Shapley (Ghorbani & Zou, 2019) and DataModels (Ilyas et al., 2022). These methods fundamentally cannot handle the structural dependencies of graphs, where a node's contribution is entangled with its structural context. Precedence-Constrained Winter Value (PC-Winter) (Chi et al., 2025a) takes the first step toward graph-aware valuation by introducing a game-theoretic framework that attributes node contributions via the Winter value (Winter, 1989) over Level Coalition Structures (LCSs) induced by GNN computation trees. While Data Shapley estimates marginal contributions via Monte Carlo (MC) sampling over all *possible permutations* of the data, PC-Winter restricts this space to *permissible* permutations exhibited by the computation graph structure, imposing two natural constraints that enforce the hierarchical level ordering and precedence ordering among nodes (Chi et al., 2025a). (For more background on PC-Winter, see § 2.1).

However, this framework yields *context-misaligned* Monte Carlo estimates. Since permissible permutations are drawn i.i.d. from a prohibitively large space, there is no guarantee that sampled permutations share structural contexts, causing each marginal contribution to be evaluated against structurally misaligned predecessor subgraphs (Shapley, 1988). This conflates the contribution of local structure with global context variation across samples, introducing high variance that undermines cross-sample reliability (Wu et al., 2022). Nevertheless, permissible permutations are *not truly i.i.d.*: they share large portions of their subtree structure and exhibit common coalition subgraphs that PC-Winter leaves entirely unexploited. Materializing these shared structures offers a principled approach toward reducing context misalignment: rather than sampling i.i.d. permissible permutations without coordination, we sample directly over shared structural coalition contexts, ensuring marginal contributions of the target node are evaluated under consistent predecessor sets, effectively capturing the rich locality signals among neighboring nodes.

We introduce our estimator as **Coordinated Sampling**, the core contribution of our framework. Rather than sampling permissible permutations independently as in PC-Winter Chi et al. (2025a), we reformulate the game-theoretic allocation in terms of the structural contexts that players share, and estimate it with a hierarchical Monte Carlo scheme. By reusing each reference structure within every labeled-node group, coordinated sampling mitigates the noise that misaligned contexts inject into pairwise comparisons. We provide a theoretical analysis, confirmed empirically across our benchmarks, showing that this contracts the estimation variance on intra-group pairwise differences, which essentially governs the ranking quality among contributors. As illustrated in Figure 1, PC-Winter draws π_1, π_2, π_3 independently, and as such w_1 and w_2 are compared under differing contexts and absorb sampling noise, whereas coordinated sampling maintains a shared context and isolates the true ranking signal.

This perspective generalizes to a broader principle of *Maximal Structural Reuse*: with coalitions coordinated around shared contexts, the same redundancy can be exploited throughout the valuation

pipeline. We therefore introduce two further contributions. First, unlike the local propagation PC-Winter (Chi et al., 2025a) performs per insertion, we introduce **Minimal Propagation** which reuses propagation signals across coalitions, recomputing only the nodes each insertion affects. Second, relaxing the Simplified Graph Convolution (SGC) objective to ridge regression (Antonelli & Bojchevski, 2025) admits closed-form solutions per subgraph; **Recursive Model Retraining** replaces full retraining with a recursive least squares estimator (Plackett, 1950) that updates incrementally across shared contexts, recovering exact marginal contributions at a fraction of the cost. Furthermore, coordinated sampling provides a streaming hierarchical structure that maximizes the resubility gains from these two efficient optimizations.

We instantiate these three contributions in **SGVR** (**S**calable **G**raph Data **V**aluation via Coordinated Structural **R**euse). We evaluate unlabeled node contributions via node-selection experiments across six benchmark datasets, providing a faithful assessment of graph-aware valuation in the transductive semi-supervised setting for node classification, and demonstrate SGVR's ability to detect noisy edges relative to representative baselines (Chi et al., 2025a; Antonelli & Bojchevski, 2025). While Antonelli & Bojchevski (2025) offer the first benchmark of valuation methods on graph data, their pipeline runs each subgraph's computation in isolation with no reuse of intermediate results, incurring runtimes of *days* even on moderately sized graphs; Maximal Structural Reuse (MSR) collectively reduces this to *minutes or a few hours*, establishing SGVR as both an effective and efficient framework for graph-aware data valuation.

2 METHODOLOGY

2.1 PRELIMINARIES: PC-WINTER VALUE

PC-Winter (Chi et al., 2025a) formulates graph data valuation as a cooperative game by first observing that nodes in a graph contribute to GNN performance in a heterogeneous and multifaceted manner: unlabeled nodes influence model performance via message-passing, while labeled nodes provide direct supervision signals. To capture these diverse contributions in a fine-grained manner, PC-Winter decomposes each node into *duplicate players*, each uniquely identified by its path to a corresponding labeled node in the K-hop computation tree. These duplicates are organized into a *contribution tree* $\mathcal{T}$, constructed by attaching each labeled node's computation-tree root to a shared dummy objective node (Chi et al., 2025a). The players in $\mathcal{T}$ exhibit two structural properties. First, they form a *Level Coalition Structure*: players are naturally grouped by their parents at the bottom level, with parent nodes acting as delegates for their sub-coalitions at higher levels up to the root O. Second, they exhibit a *Unilateral Dependency Structure*: each player's contribution is contingent on its ancestors being present in the coalition, since a duplicate player accrues a non-zero marginal contribution only once its labeled ancestor appears (Chi et al., 2025a). PC-Winter addresses both properties simultaneously by extending the Winter value (Winter, 1989), a solution concept for games with level coalition structures, with an additional precedence constraint derived from the GNN message-passing topology (Chi et al., 2025a). This gives rise to two constraints on permissible orderings: the **Level Constraint** $\mathcal{C}_{\text{lev}}$ (Eq. (A.7)), which ensures that any player and all its descendants appear as an unbroken group in every permutation, and the **Precedence Constraint** $\mathcal{C}_{\text{prec}}$ (Eq. (A.8)), which requires every parent to precede its descendants. A permutation is *permissible* if and only if it satisfies both constraints simultaneously, giving $\Pi^{\mathcal{T}}(\mathcal{D}) = \mathcal{C}_{\text{lev}} \cap \mathcal{C}_{\text{prec}}$ (Eq. (A.9)), which corresponds exactly to the pre-order Depth-First Search (DFS) traversals of $\mathcal{T}$ (Chi et al., 2025a). The **PC-Winter value** of a duplicate player v is then its average marginal contribution over all permissible permutations:

$$\phi_v^{\text{PCW}}(\mathcal{D},\mathcal{U},\mathcal{T}) = \frac{1}{|\Pi^{\mathcal{T}}(\mathcal{D})|} \sum_{\pi \in \Pi^{\mathcal{T}}(\mathcal{D})} [\mathcal{U}(S_v^\pi \cup \{v\}) - \mathcal{U}(S_v^\pi)] \tag{1}$$

where $\mathcal{D}$ is the player set and $S_v^\pi = \{u \in \mathcal{D} \mid \pi(u) < \pi(v)\}$ is the predecessor set of v under permutation π. The utility $\mathcal{U}(S)$ is the validation performance of a model trained on coalition $S \subseteq \mathcal{D}$, so the **marginal contribution** $\mathcal{U}(S_v^\pi \cup \{v\}) - \mathcal{U}(S_v^\pi)$ measures the change attributable to adding v. The value of the original node v is obtained by aggregating the PC-Winter values of all its duplicate players (for formal details, see Appendix C.5).

To make computation tractable, PC-Winter employs three strategies:

1. **MC Permutation Sampling.** Exact averaging over $\Pi^{\mathcal{T}}(\mathcal{D})$ is combinatorially intractable, so PC-Winter estimates the sum in Eq. (1) via MC sampling, drawing each sample as a uniformly random pre-order DFS traversal consistent with $\mathcal{C}_{\text{lev}}$ and $\mathcal{C}_{\text{prec}}$ (Chi et al., 2025a). We refer to this scheme as *uncoordinated sampling* of permissible permutations, since each player's predecessor set is drawn from an independent permutation.
2. **Hierarchical Truncation.** GNN models exhibit *neighborhood saturation*, whereby satisfactory performance is achievable using only a small subset of neighbors, with diminishing returns beyond a certain point (Xie et al., 2020). Consequently, the marginal contributions of late-visited child players in a DFS traversal are negligible. Given level-wise truncation rates $\rho_1, \rho_2, \ldots, \rho_{K+1}$, PC-Winter computes actual marginal contributions only for players in the first $(1 - \rho_\ell)$ portion of each node's child subtrees at level ℓ, directly approximating the remaining contributions as zero (Chi et al., 2025a).
3. **Local Propagation.** Driven by $\mathcal{C}_{\text{lev}}$, once all players in a labeled node's subtree have been inserted, its aggregated representation is fixed and cached for the remainder of the DFS pass (Chi et al., 2025a). Only the subtree of the most recently visited labeled node requires active re-propagation, reducing cost to a local neighborhood.

2.2 Scalable Graph Data Valuation via Coordinated Structural Reuse

While PC-Winter (Chi et al., 2025a) provides a principled graph-aware valuation framework, its MC estimation (Eq. (1)) leaves the shared structure of $\mathcal{T}$ underutilized in two ways. Permissible permutations are sampled without coordination (§2.1), so a node is measured against inconsistent contexts across samples; and although PC-Winter reuses part of $\mathcal{T}$ across insertions, message-passing representations are only partially reused and model parameters are retrained from scratch at every marginal evaluation. Both stem from the structure shared across players being consumed in isolation rather than efficiently reused. Drawing a parallel to maximum sample reuse (Wang & Jia, 2022), we address both through a principle for graph valuation, *Maximal Structural Reuse* (MSR): the redundancy in $\mathcal{T}$ is exploited for both lower-variance sampling and a more efficient pipeline. We instantiate MSR in **SGVR** through three components: *coordinated sampling* (§2.3), *minimal propagation* (§2.4), and *recursive model training* (§2.5).

Observation 1: Context Misalignment in Uncoordinated Sampling. A target node's marginal contribution is measured against its predecessor set; under uncoordinated sampling this set is redrawn independently for each sample, so two members of the same labeled-node group are, on average, measured against mismatched contexts. This misalignment inflates the variance of the estimated values, and a pairwise difference is then dominated by this noise rather than by the players' true relative worth. The effect is most pronounced for intra-group ranking, which depends precisely on these within-group differences (Figure 1). *The mismatch is an artifact of how permutations are sampled, not a property of the value itself*: the same context recurs across exponentially many permissible permutations, and exploiting this redundancy lets every contributor be measured against it consistently.

This recurrence is structural, not incidental; permissible permutations compose recursively over the sub-coalition blocks of $\mathcal{T}$ (Appendix C.6, Eq. (A.13)), so a single predecessor set is realized by exponentially many of them. We write Γ for the number of permissible permutations, which obeys a recurrence over the block structure of $\mathcal{T}$ (Eq. (A.13)). Permutations that induce the same predecessor set at a player's position are *prefix-equivalent* and yield an identical marginal contribution, since the utility depends on the coalition alone and not on its internal order Shapley (1988). Formally, π and π' are prefix-equivalent at position j when the player at that position shares the same predecessor set under both,

$$\pi \sim_j \pi' \iff \exists\, v \in \mathcal{D} \text{ s.t. } \pi(v) = j :\ S_v^{\pi} = S_v^{\pi'}, \tag{2}$$

so both contribute the identical marginal $\mathcal{U}(S_v^{\pi} \cup \{v\}) - \mathcal{U}(S_v^{\pi})$. Collapsing each equivalence class to a single representative therefore reorganizes PC-Winter from a sum over permutations into a weighted sum over distinct predecessor sets, which can be theoretically evaluated once per class in a Dynamic Programming (DP) framework Held & Karp (1962). We write Λ for the number of such distinct predecessor sets, which obeys its recurrence (Eq. (A.16), Appendix C.6) and is exponentially smaller than Γ (Appendix C.7).

Theorem 1 (Coalition Equivalence; informal, see Theorem 3)**.** *The permissible permutations $\Pi^{\mathcal{T}}(\mathcal{D})$ decompose recursively along the sub-coalitions of $\mathcal{T}$ (Eq.* (A.13)*), and permutations realizing a common predecessor set form prefix-equivalence classes (Eq.* (2)*). Algorithms 1 and 2 enumerate permutations and equivalence classes respectively; both return the exact PC-Winter value of Eq.* (1)*, at $|\mathcal{D}| \cdot |\Pi^{\mathcal{T}}(\mathcal{D})|$ and Λ marginal evaluations, respectively (Eq.* (A.16)*), the latter exponentially smaller.*

The theorem yields an exact reformulation of Eq. (1) as a sum over distinct predecessor sets. Let $\mathcal{S}_p$ be the distinct permissible predecessor sets of player p and let $M_p(S)$ count the permissible permutations realizing a given S. The PC-Winter value then admits the reformulation

$$\phi_p^{\mathrm{PCW}} = \frac{1}{|\Pi^{\mathcal{T}}(\mathcal{D})|} \sum_{S \in \mathcal{S}_p} M_p(S) \big[\mathcal{U}(S \cup \{p\}) - \mathcal{U}(S)\big], \tag{3}$$

derived together with the recursive form of M_p in Appendix C.7.

This reformulation is the basis for MSR. We develop its three realizations next, beginning with coordinated sampling and the variance guarantees that motivate it.

2.3 Coordinated Sampling

Despite the reduction in Eq. (3), exact evaluation sums over Λ predecessor sets and remains intractable at scale; we therefore adopt a refined MC strategy Chi et al. (2025a) that also addresses the variance identified in Observation 1. We introduce *coordinated sampling*, which shares a hierarchical context among the members of a group, inducing a covariance between their estimates that reduces the variance in their pairwise comparison. The estimand is unchanged: both estimators are unbiased and, by the law of large numbers, converge to the exact PC-Winter value as the budget N grows,

$$\lim_{N\to\infty} \mathbb{E}\big[\hat{\phi}_q^{\mathrm{unc}}\big] = \lim_{N\to\infty} \mathbb{E}\big[\hat{\phi}_q^{\mathrm{coord}}\big] = \phi_q^{\mathrm{PCW}}, \tag{4}$$

so coordinated sampling trades no bias for its variance reduction (Appendix C.8).

Sampling construction. We replace the per-player budget of N independent permutations with a nested factorization $N = h_1 h_2 \cdots h_{K+1}$[1], where h_ℓ is the number of child orderings sampled at level ℓ. Writing $\mathcal{B}_\ell^j$ for the sub-coalition rooted at the j-th node at level ℓ, we draw h_1 orderings at the root and, within each, h_ℓ orderings of every level-ℓ block nested inside the orderings already drawn at predecessor levels. By construction, the context above any sub-coalition $\mathcal{B}_\ell^j$ is drawn once and shared by every player it contains (Appendix C.8).

Covariance-induced reduction. Intra-group ranking is decided by pairwise differences $\hat{\phi}_i - \hat{\phi}_j$, whose reliability is governed by their variance

$$\mathrm{Var}\big(\hat{\phi}_i - \hat{\phi}_j\big) = \mathrm{Var}\big(\hat{\phi}_i\big) + \mathrm{Var}\big(\hat{\phi}_j\big) - 2\,\mathrm{Cov}\big(\hat{\phi}_i, \hat{\phi}_j\big). \tag{5}$$

Under uncoordinated sampling the two estimates are independent, so $\mathrm{Cov} = 0$ and the full variance of each survives in the difference. Coordinated sampling instead induces a covariance through the context i and j share above their Lowest Common Ancestor (LCA); whenever this covariance is positive, it offsets part of the variance the shared levels contribute, shifting weight toward the levels below the LCA, where i and j enter disjoint sub-coalitions. Nesting also raises each player's own variance, since fewer draws are independent at the upper levels; the covariance must outweigh this.

Theorem 2 (Pairwise variance reduction; informal, see Theorem 5)**.** *For any pair (i, j) with LCA at level L_{ij}, uncoordinated and coordinated sampling under a matched budget N satisfy*

$$\mathrm{Var}\big(\hat{\phi}_i^{\mathrm{unc}} - \hat{\phi}_j^{\mathrm{unc}}\big) = \frac{1}{N} \sum_{\ell} \big(\sigma_{i,\ell}^2 + \sigma_{j,\ell}^2\big), \tag{6}$$

$$\mathrm{Var}\big(\hat{\phi}_i^{\mathrm{coord}} - \hat{\phi}_j^{\mathrm{coord}}\big) = \mathrm{Var}\big(\hat{\phi}_i^{\mathrm{unc}} - \hat{\phi}_j^{\mathrm{unc}}\big) + \Delta_{ij} - 2\,\mathrm{Cov}_{\le L_{ij}}, \tag{7}$$

[1] h_1 corresponds to MC permutation sampling from the root of $\mathcal{T}$, equivalent to the original PC-Winter strategy (Chi et al., 2025a).

where $H_\ell = h_1 \cdots h_\ell$, $\Delta_{ij} = \sum_\ell (\sigma^2_{i,\ell} + \sigma^2_{j,\ell})(1/H_\ell - 1/N) \geq 0$ is the added per-player variance from nesting, and $\mathrm{Cov}_{\leq L_{ij}}$ is the covariance induced by the levels shared above the LCA (Theorem 5). Coordinated sampling reduces the pairwise variance if and only if $2\,\mathrm{Cov}_{\leq L_{ij}} > \Delta_{ij}$, which we check empirically across different benchmarks (Section D.2).

A lower pairwise variance makes the estimated ordering of i and j more reliable. We expect the effect to be strongest in heterophilous regimes Ju et al. (2026), where dissimilar neighbors inflate the deeper-level variance on which coordination concentrates its budget; §3.5 tests this by injecting heterophilous edges.

2.4 Minimal Propagation

At every incremental step a new leaf node v is inserted into the current coalition and its marginal contribution evaluated, requiring an update to node embeddings via message passing. Following PC-Winter (Chi et al., 2025a), we adopt SGC as the propagation rule. PC-Winter's Local Propagation re-propagates the entire subtree $\mathcal{T}_v$ of the most recently visited labeled node at each insertion, yet nodes unaffected by the message-passing signal of v need not be recomputed. We introduce **Minimal Propagation**, which restricts each update to the neighborhood affected by inserting v following a Breadth-First Search (BFS) frontier, carrying all remaining embeddings forward unchanged (Eq. (A.35)). On a d-regular graph this reduces per-insertion cost to $\mathcal{O}(d^{k-1})$ for a node added at the k-th level, against $\mathcal{O}(d^k)$ for Local Propagation. Full derivation is in Appendix C.9.

2.5 Recursive Model Training

At every incremental step the model parameters $\hat{\Theta}$ must be updated to reflect the changed embedding of the labeled node v of $\mathcal{T}_v$. Following Antonelli & Bojchevski (2025), we relax node classification to ridge regression with closed-form solution $\hat{\Theta} = (X^\top X + I)^{-1} X^\top Y$; recomputing this inverse from scratch at every step is expensive. Crucially, each insertion changes at most one labeled node's embedding, so the normal matrix $X^\top X + I$ is perturbed by a single rank-one term. This structure admits an exact incremental update: we maintain the inverse covariance P_{k-1} and apply a rank-one Recursive Least Squares (RLS) correction upon each embedding change,

$$P_k = P_{k-1} - K_k\, x_k^\top P_{k-1}, \qquad \hat{\Theta}_k = \hat{\Theta}_{k-1} + K_k \big(y_k - x_k^\top \hat{\Theta}_{k-1}\big)^\top, \tag{8}$$

where K_k is the Kalman gain (Plackett, 1950). This recovers the exact ridge-regularized solution at every step (Sherman & Morrison, 1950) while eliminating full matrix inversion: with P_{k-1} maintained across steps, each incremental update reduces to a matrix-vector product and a rank-one correction, at $\mathcal{O}(d^2)$ cost per node rather than the $\mathcal{O}(d^3)$ of a from-scratch inverse. Full derivation is in Appendix C.10.

3 Experimental Results and Analysis

3.1 Experimental Settings

We evaluate the valuation methods in the semi-supervised transductive setting following (Antonelli & Bojchevski, 2025): the labeled train and validation sets are fixed, the validation set is used to compute marginal contributions during valuation, and performance is reported on the test set (all remaining, unlabeled nodes). Dataset sizes and splits are in Appendix A.1. We compare SGVR against Leave-One-Out (LOO), DataModels, and Data Shapley (using the implementations of Antonelli & Bojchevski (2025)), PC-Winter (Chi et al., 2025a), and two trivial heuristics: random selection (RND) and degree-based selection (Degree). SGVR uses hierarchical sampling factors $(h_2, h_3) = (5, 2)$.[2] We define budget as the number of marginal-contribution evaluations. All methods are budgeted except LOO, RND, and Degree, which are computed deterministically; the four budgeted methods (DataModels, Data Shapley, PC-Winter, SGVR) share a common budget of 50k, and for SGVR and PC-Winter we additionally report a higher budget of 500k. SGVR and PC-Winter use identical level-wise truncation rates (Chi et al., 2025a) (ρ_1, ρ_2, ρ_3) for a fair comparison, chosen per dataset

[2] h_1, the number of permutations over labeled nodes, is not a free parameter: it is inferred from the evaluation budget.

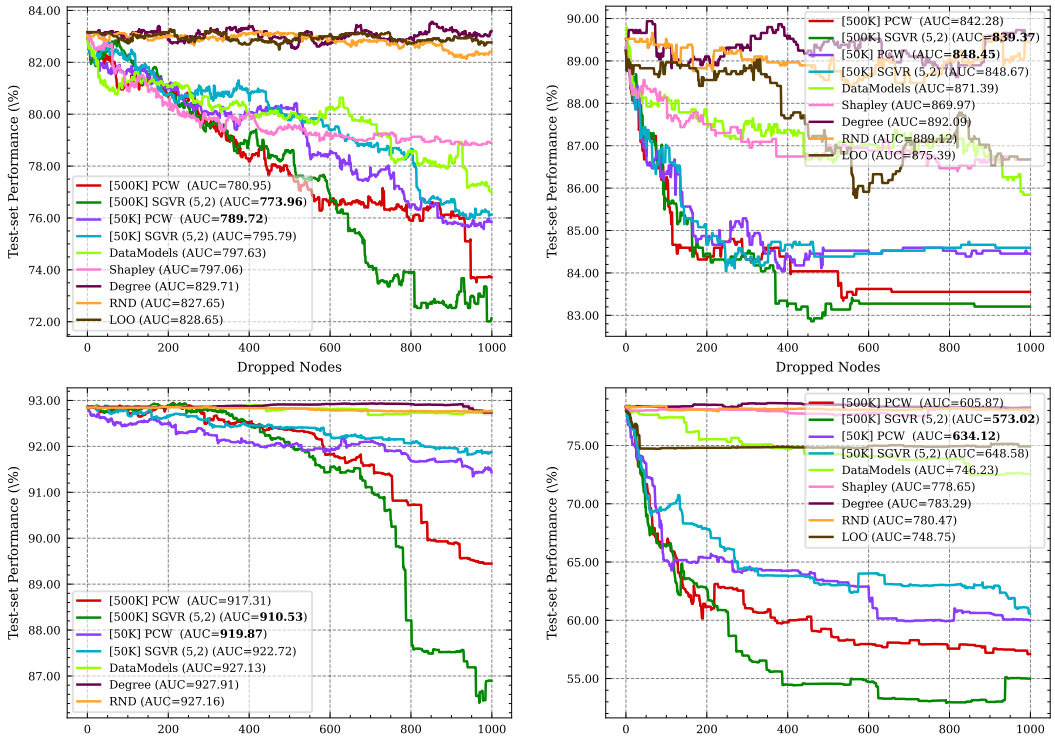


Figure 2: Most-influential-node-removal drop curves on Cora-ML, CiteSeer (top row), Physics, and PubMed (bottom row) (Sen et al., 2008; Shchur et al., 2018), summarized by AUC (lower is better; §3.1).

by graph size (node count): $(0.85, 0.85, 0.9)$ for Cora-ML, CiteSeer, and Photo; $(0.9, 0.925, 0.95)$ for PubMed and Computers; $(0.95, 0.975, 0.985)$ for Physics. We evaluate data selection quality via most-influential-node removal on unlabeled nodes: nodes are dropped in decreasing order of assigned value and test performance is recorded after each removal; lower node-drop curves indicate better valuation, summarized by their AUC (Chi et al., 2025b) (lower is better) over 3 seeds. LOO and Data Shapley are omitted from Physics due to computational cost at this scale (Table 2). All experiments run on a single machine with four NVIDIA A100 80GB GPUs and 1 TB of RAM; our implementations of all methods use multiprocessing to parallelize graph construction and model training, with the number of parallel jobs per method set to fully utilize the GPUs as memory permits.

3.2 Data Selection Quality

Figure 2 shows that the two structure-informed methods, SGVR and PC-Winter, produce the steepest drops (lowest AUC) on all four datasets, well below the trivial baselines (LOO, RND, Degree) and the structure-agnostic Data Shapley and DataModels. At the shared 50k budget, PC-Winter has a marginally lower AUC than SGVR on all four; at 500k, SGVR overtakes PC-Winter on every dataset, most clearly on PubMed (AUC 573.02 vs. 605.87). Results for Computers and Photo, which follow the same ordering with smaller margins between methods, are in Appendix D.1. On these two datasets, the positive shared-level covariance that Theorem 5 requires for variance reduction is weaker on Photo and almost absent on Computers, so coordination offers less advantage given these dataset characteristics (Appendix D.2).

3.3 Efficiency Analysis

Table 1 reports valuation time at the 50k budget. LOO, DataModels, and Data Shapley rebuild the graph and retrain from scratch at every evaluation, whereas SGVR inserts one node at a time, propagating only the affected neighborhood (Section 2.4) and updating the model with a rank-one RLS correction (Section 2.5). The speedup grows with graph size, from rough parity with Data

Table 1: Valuation time at the 50k budget, with the speedup of the full SGVR in parentheses; all runs use the same machine and implementation (§3.1). The two ablations replace minimal propagation with Local Propagation and RLS with retraining from scratch, respectively.

					SGVR		
Dataset	LOO	DataModels	Data Shapley	PC-Winter	w/o min. prop.	w/o RLS	full
Cora-ML	2m 16s (0.5×)	35m 05s (7.8×)	5m 55s (1.3×)	6m 50s (1.5×)	4m 43s (1.0×)	6m 25s (1.4×)	4m 30s
CiteSeer	38s (0.2×)	6m 55s (2.2×)	2m 59s (0.9×)	5m 30s (1.7×)	3m 48s (1.2×)	5m 05s (1.6×)	3m 12s
PubMed	3h 20m (71×)	2h 55m (62×)	3h 30m (74×)	7m 30s (2.6×)	4m 52s (1.7×)	6m 05s (2.1×)	2m 50s
Computers	40m 12s (12×)	1h 11m (21×)	1h 01m (18×)	9m 45s (2.8×)	4m 58s (1.4×)	7m 55s (2.3×)	3m 26s
Photo	5m 15s (2.0×)	31m 35s (12×)	12m 55s (4.9×)	6m 15s (2.4×)	3m 50s (1.5×)	4m 56s (1.9×)	2m 38s
Physics	est. 3d (476×)	1d 2h (173×)	17h 56m (118×)	31m 50s (3.5×)	17m 35s (1.9×)	23m 20s (2.6×)	9m 05s

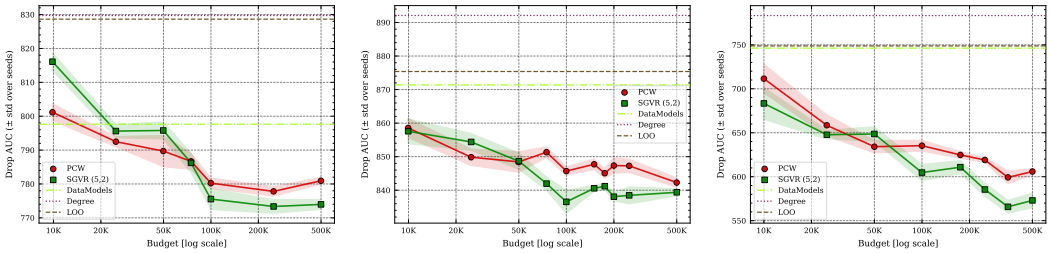


Figure 3: AUC of the node-drop curves (lower is better) under most-influential-node removal for SGVR and PC-Winter on Cora-ML, CiteSeer, and PubMed as the budget grows from 10k to 500k, averaged over 3 seeds with confidence intervals.

Shapley on CiteSeer to 62–74× over DataModels and Data Shapley on PubMed and 118–173× on Physics, where DataModels needs over a day against 9 minutes for SGVR. LOO, which needs one retraining per node regardless of the budget, is cheaper only on the two smallest graphs. SGVR is also 1.5–3.5× faster than PC-Winter.

The ablations isolate the two components: removing RLS slows SGVR by 1.4–2.6× and removing minimal propagation by 1.0–1.9×, with the largest effects on Physics.

3.4 Convergence with Budget

Here we extend the node-drop evaluation of §3.1 along the allocated budget, to see how the different methods converge as the budget grows.

Figure 3 reports the AUC of the node-drop curves under the same protocol as Figure 2, sweeping the budget from 10k to 500k marginal-contribution evaluations on three of our benchmarks, Cora-ML, CiteSeer, and PubMed; the remaining methods do not vary with the budget and are drawn as horizontal reference lines. At a matched budget, a lower curve means faster convergence: a better ranking for the same number of evaluations. As the budget increases, both graph-based methods improve proportionally, with SGVR outperforming PC-Winter at larger budgets, where ranking signals become more precise. Results are averaged over 3 seeds, with confidence intervals shown.

3.5 Data Selection under Heterophily Noise

The benchmarks above are largely homophilous: connected nodes tend to share a label, so message passing aggregates consistent signal. Under heterophily, neighbors carry dissimilar labels and features (Ju et al., 2026), so the value of an unlabeled node depends strongly on which of its neighbors are already in the coalition, which inflates the variance at the levels below the LCA that coordinated sampling targets (§2.3). We inject heterophily by adding 50% extra edges at random between nodes of different classes, simulating noisy connections that pass misleading signal to the labeled nodes, and repeat the budget sweep of §3.4 on the perturbed graphs (Figure 4).

Heterophily shifts the comparison in favor of SGVR. It is ahead or level at every budget on Cora-ML (547.9 vs. 562.5 at 500k), and far ahead on CiteSeer from 25k onward (364.3 vs. 448.9 at 250k), where the two are within a few AUC points without noise; on PubMed PC-Winter leads at small budgets

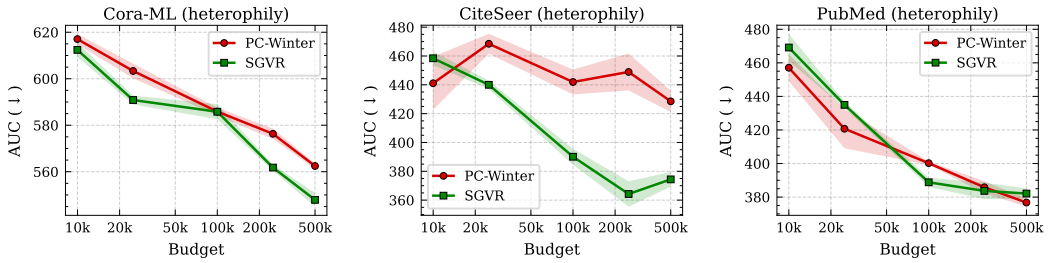


Figure 4: AUC of the node-drop curves (lower is better) for SGVR and PC-Winter across budgets with injected heterophily: 50% extra edges added at random between nodes of different classes.

and the two are on par from 100k. Ranking helpful unlabeled nodes first requires separating them from neighbors that only contribute noise, and evaluating those neighbors against shared contexts keeps the comparison from being dominated by context variation.

4 Related Work

Data Valuation. Game-theoretic frameworks such as Data Shapley (Ghorbani & Zou, 2019) and DataModels (Ilyas et al., 2022) quantify individual data contributions, with subsequent works improving robustness and scalability via semi-values and *maximal sample reuse* (Kwon & Zou, 2022; Wang & Jia, 2023; Li & Yu, 2023). All assume i.i.d. samples and fundamentally fail to capture structural dependencies in graph data (see Appendix B for a detailed overview).

Graph Data Valuation. PC-Winter (Chi et al., 2025a) introduces the first game-theoretic framework for graph data valuation via the Winter value (Winter, 1989) over GNN computation trees; later work extends graph valuation to heterogeneous graphs (Tang et al., 2026) and task-agnostic settings (Falahati & Amiri, 2024). Antonelli & Bojchevski (2025) benchmark a suite of valuation approaches for graph data, covering both i.i.d. and graph-aware methods, yet i.i.d. methods remain fundamentally agnostic to graph structure, treating nodes as exchangeable and ignoring relational dependencies (see Appendix B for further discussion).

Reusable Statistical Inference. RLS and the Sherman–Morrison formula enable exact incremental parameter updates without full recomputation (Plackett, 1950; Sherman & Morrison, 1950). SGVR leverages this linearity to enable streaming parameter updates, allowing efficient model retraining incrementally rather than from scratch at each valuation step (see Appendix B).

5 Conclusion

We introduced SGVR, a variance-reduced hierarchical sampling scheme for graph data valuation. Under the principle of MSR, coordinated sampling reuses the structural context shared by the members of a labeled-node group, so that intra-group comparisons are made against consistent predecessor sets, while minimal propagation and RLS updates reuse computation across evaluations. Across six benchmarks, SGVR is both effective and efficient against the strongest competitors, which are either graph-uninformed and costly (Data Shapley, DataModels) or graph-aware but less efficient (PC-Winter); its advantage grows with the budget and is wider under injected heterophily.

Two directions follow. First, adaptive tuning of the truncation and sampling rates, from the observed variance at each level, would remove the main manual choice in the method. Second, the same hierarchical sampling scheme extends to graph-level predictive valuation, where the players are graphs rather than nodes and structural context is shared across them in the same way.

AI Use Statement

We used generative AI assistants in a limited, non-research capacity: for language choice, paraphrasing, and proofreading of author-written text, and to help explore the related literature. All research contributions, including the problem formulation, theoretical results and proofs, algorithm design,

implementation, and experiments, were conceived, carried out, and verified by the authors. Every reference surfaced with AI assistance was read and checked against its original source before being cited. The authors take full responsibility for the final content of this work.

Ethics Statement

This work studies data valuation for graph-structured data and uses only publicly available benchmark datasets, which contain no personally identifiable information and involve no human subjects. We identify no potential harms, conflicts of interest, or fairness, privacy, or legal compliance concerns arising from this work.

Reproducibility Statement

All theoretical results are stated and proved in Appendix C. Implementation details, including the datasets, splits, evaluation budgets, sampling factors, truncation rates, and hardware, are given in §3.1, with dataset statistics in Appendix A.1 and per-component timings in §3.3. Anonymized code to reproduce all experiments is available at `https://anonymous.4open.science/r/SGVR/`.

References

Selahattin Akkas and Ariful Azad. Gnnshap: Scalable and accurate gnn explanation using shapley values. In WWW, 2024. doi: 10.1145/3589334.3645599.

Simone Antonelli and Aleksandar Bojchevski. Node-level data valuation on graphs. Transactions on Machine Learning Research, 2025. ISSN 2835-8856. URL https://openreview.net/forum?id=tNyApIqDSJ.

Juhan Bae, Nathan Ng, Alston Lo, Marzyeh Ghassemi, and Roger B. Grosse. If influence functions are the answer, then what is the question? In NeurIPS, 2022.

Samyadeep Basu, Phillip Pope, and Soheil Feizi. Influence functions in deep learning are fragile. In ICLR, 2021.

Joan Bruna, Wojciech Zaremba, Arthur Szlam, and Yann LeCun. Spectral networks and locally connected networks on graphs. International Conference on Learning Representations, 2013.

Ngoc Bui, Hieu Trung Nguyen, Viet Anh Nguyen, and Rex Ying. Explaining graph neural networks via structure-aware interaction index. In ICML, 2024.

F. Chantreuil. Axiomatics of level structure values. In M. J. Holler and G. Owen (eds.), Power Indices and Coalition Formation. Springer, Boston, MA, 2000. doi: 10.1007/978-1-4757-6221-1_4.

Jianfei Chen, Jun Zhu, and Le Song. Stochastic training of graph convolutional networks with variance reduction. International Conference on Machine Learning, 2017.

Zizhang Chen, Peizhao Li, Hongfu Liu, and Pengyu Hong. Characterizing the influence of graph elements. International Conference on Learning Representations, 2022. doi: 10.48550/arXiv.2210.07441.

Chirag Chhablani, Sarthak Jain, Akshay Channesh, Ian A. Kash, and Sourav Medya. Game-theoretic counterfactual explanation for graph neural networks. In WWW, 2024. doi: 10.1145/3589334.3645419.

Hongliang Chi, Wei Jin, Charu C. Aggarwal, and Yao Ma. Precedence-constrained winter value for effective graph data valuation. In ICLR, 2025a.

Hongliang Chi, Qiong Wu, Zhengyi Zhou, and Yao Ma. Shapley-guided utility learning for effective graph inference data valuation. In ICLR, 2025b.

Pradeep Dubey, Abraham Neyman, and Robert James Weber. Value theory without efficiency. Math. Oper. Res., 1981. doi: 10.1287/MOOR.6.1.122.

Alexandre Duval and Fragkiskos D. Malliaros. Graphsvx: Shapley value explanations for graph neural networks. In ECML/PKDD, 2021. doi: 10.1007/978-3-030-86520-7_19.

Logan Engstrom, Axel Feldmann, and Aleksander Madry. Dsdm: Model-aware dataset selection with datamodels. In ICML, 2024.

Ali Falahati and Mohammad Mohammadi Amiri. Disentangled Structural and Featural Representation for Task-Agnostic Graph Valuation. arXiv preprint arXiv:2408.12659, 2024. doi: 10.48550/arXiv.2408.12659.

R. A. Fisher. Statistical methods for research workers. In Samuel Kotz and Norman L. Johnson (eds.), Breakthroughs in Statistics: Methodology and Distribution, pp. 66–70. Springer New York, New York, NY, 1992. ISBN 978-1-4612-4380-9. doi: 10.1007/978-1-4612-4380-9_6. URL https://doi.org/10.1007/978-1-4612-4380-9_6.

Amirata Ghorbani and James Y. Zou. Data shapley: Equitable valuation of data for machine learning. International Conference on Machine Learning, 2019.

William L. Hamilton, Z. Ying, and J. Leskovec. Inductive representation learning on large graphs. Neural Information Processing Systems, 2017.

S Haykin. Adaptive filter theory. Prentice-Hall, Inc., USA, 1986. ISBN 0130040525.

Michael Held and Richard M. Karp. A dynamic programming approach to sequencing problems. Journal of the Society for Industrial and Applied Mathematics, 10(1):196–210, 1962.

Andrew Ilyas, Sung Min Park, Logan Engstrom, Guillaume Leclerc, and Aleksander Madry. Datamodels: Understanding predictions with data and data with predictions. In ICML, 2022.

S. Jegelka. Theory of graph neural networks: Representation and learning. arXiv.org, 2022. doi: 10.48550/arXiv.2204.07697.

Wei Ju, Siyu Yi, Yifan Wang, Zhiping Xiao, Zhengyang Mao, Hourun Li, Yiyang Gu, Yifang Qin, Nan Yin, Senzhang Wang, Xinwang Liu, Philip S. Yu, and Ming Zhang. A survey of graph neural networks in real world: Imbalance, noise, privacy and ood challenges. IEEE Trans. Pattern Anal. Mach. Intell., 2026. doi: 10.1109/TPAMI.2025.3630673.

Thomas Kipf and M. Welling. Semi-supervised classification with graph convolutional networks. International Conference on Learning Representations, 2016.

Pang Wei Koh and Percy Liang. Understanding black-box predictions via influence functions. In ICML, 2017.

Yongchan Kwon and James Zou. Beta shapley: a unified and noise-reduced data valuation framework for machine learning. In AISTATS, 2022.

Weida Li and Yaoliang Yu. Robust data valuation with weighted banzhaf values. In NeurIPS, 2023.

Weida Li and Yaoliang Yu. One sample fits all: Approximating all probabilistic values simultaneously and efficiently. In NeurIPS, 2024.

Xin Liu, Mingyu Yan, Lei Deng, Guoqi Li, Xiaochun Ye, and Dongrui Fan. Sampling methods for efficient training of graph convolutional networks: A survey. IEEE/CAA Journal of Automatica Sinica, 2021. doi: 10.1109/JAS.2021.1004311.

Yurong Liu, R. Teal Witter, Flip Korn, Tarfah Alrashed, Dimitris Paparas, and Juliana Freire. Kernel banzhaf: A fast and robust estimator for banzhaf values., 2024. URL https://dblp.org/rec/journals/corr/abs-2410-08336. arXiv preprint CoRR.

Francesco Orabona. A modern introduction to online learning. ArXiv, abs/1912.13213, 2019. URL https://api.semanticscholar.org/CorpusID:209516176.

Sung Min Park, Kristian Georgiev, Andrew Ilyas, Guillaume Leclerc, and Aleksander Madry. Trak: Attributing model behavior at scale. In ICML, 2023.

J. Pei. A survey on data pricing: From economics to data science. IEEE Transactions on Knowledge and Data Engineering, 2020. doi: 10.1109/TKDE.2020.3045927.

R. L. Plackett. Some theorems in least squares. Biometrika, 37(1/2):149–157, 1950. ISSN 00063444, 14643510. URL http://www.jstor.org/stable/2332158.

Tom Rainforth, Robert Cornish, Hongseok Yang, and Andrew Warrington. On nesting monte carlo estimators. In Jennifer G. Dy and Andreas Krause (eds.), Proceedings of the 35th International Conference on Machine Learning, ICML 2018, Stockholmsmässan, Stockholm, Sweden, July 10-15, 2018, Proceedings of Machine Learning Research, pp. 4264–4273. PMLR, 2018. URL http://proceedings.mlr.press/v80/rainforth18a.html.

Prithviraj Sen, Galileo Namata, M. Bilgic, L. Getoor, Brian Gallagher, and Tina Eliassi-Rad. Collective classification in network data. The AI Magazine, 2008. doi: 10.1201/b17320-16.

L. Shapley. A value for n-person games. In Alvin E. Roth (ed.), The Shapley Value: Essays in Honor of Lloyd S. Shapley. Cambridge University Press, 1988. doi: 10.1017/CBO9780511528446.003.

Oleksandr Shchur, Maximilian Mumme, Aleksandar Bojchevski, and Stephan Günnemann. Pitfalls of graph neural network evaluation. arXiv.org, 2018.

Jack Sherman and Winifred J. Morrison. Adjustment of an inverse matrix corresponding to a change in one element of a given matrix. The Annals of Mathematical Statistics, 21(1):124–127, 1950. ISSN 00034851. URL http://www.jstor.org/stable/2236561.

Rachael Hwee Ling Sim, Xinyi Xu, and K. H. Low. Data valuation in machine learning: "ingredients", strategies, and open challenges. International Joint Conference on Artificial Intelligence, 2022. doi: 10.24963/ijcai.2022/782.

Xin Tang, Quanyan Gao, and Chao Li. LTSV: Layered Type-Constrained Shapley Value for Heterogeneous Graph Data Valuation. In Proceedings of the 32nd ACM SIGKDD Conference on Knowledge Discovery and Data Mining V.2, 2026. doi: 10.1145/3770855.3817695.

Petar Velickovic, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Liò, and Yoshua Bengio. Graph attention networks. International Conference on Learning Representations, 2017. doi: 10.17863/CAM.48429.

J. von Neumann and O. Morgenstern. Theory of Games and Economic Behavior: 60th Anniversary Commemorative Edition. Princeton University Press, 2020.

Jiachen T. Wang and R. Jia. Data banzhaf: A robust data valuation framework for machine learning. International Conference on Artificial Intelligence and Statistics, 2022.

Jiachen T. Wang and Ruoxi Jia. Data banzhaf: A robust data valuation framework for machine learning. In AISTATS, 2023.

Eyal Winter. A value for cooperative games with levels structure of cooperation. International Journal of Game Theory, 18(2):227–240, 1989. doi: 10.1007/BF01268161.

Felix Wu, Tianyi Zhang, A. Souza, Christopher Fifty, Tao Yu, and Kilian Q. Weinberger. Simplifying graph convolutional networks. International Conference on Machine Learning, 2019.

Mengmeng Wu, R. Jia, Changle Lin, Wei Huang, and Xiangyu Chang. Variance reduced shapley value estimation for trustworthy data valuation. Comput. Oper. Res., 159:106305, 2022. URL https://api.semanticscholar.org/CorpusID:257050876.

Yiqing Xie, Sha Li, Carl Yang, Raymond Chi-Wing Wong, and Jiawei Han. When do GNNs work: Understanding and improving neighborhood aggregation. In Christian Bessiere (ed.), Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence, IJCAI-20, pp. 1303–1309. International Joint Conferences on Artificial Intelligence Organization, 2020. doi: 10.24963/ijcai.2020/181. URL https://doi.org/10.24963/ijcai.2020/181.

Keyulu Xu, Weihua Hu, J. Leskovec, and S. Jegelka. How powerful are graph neural networks? International Conference on Learning Representations, 2018.

Lina Zhou, Shimei Pan, Jianwu Wang, and A. Vasilakos. Machine learning on big data: Opportunities and challenges. Neurocomputing, 2017. doi: 10.1016/j.neucom.2017.01.026.

A Experimental Details

A.1 Datasets

Table 2: Dataset statistics after taking the largest connected component and applying the transductive train/val/test split used in all experiments.

Dataset	Nodes	Edges	Mean deg.	Features	Classes	Train	Val	Test
Cora-ML	2,810	7,981	5.7	2,879	7	140	140	2,530
CiteSeer	1,681	2,902	3.5	602	6	120	120	1,441
PubMed	19,717	44,324	4.5	500	3	60	60	19,597
Computers	13,381	245,778	36.7	767	10	200	200	12,981
Photo	7,487	119,043	31.8	745	8	160	160	7,167
Physics	34,493	247,962	14.4	8,415	5	100	100	34,293

B Additional Related Works

Cooperative Game Theory. Cooperative games provide the theoretical foundation for attributing value to individual players within coalitions (Shapley, 1988; von Neumann & Morgenstern, 2020). The Shapley value satisfies a canonical set of fairness axioms; semi-values form a broader family by dropping the efficiency axiom from Shapley value assumptions (Dubey et al., 1981). The Winter value generalizes this to LCSs, decomposing contributions hierarchically across levels of coalition structure (Winter, 1989; Chantreuil, 2000), serving as the game-theoretic backbone of PC-Winter (Chi et al., 2025a) and SGVR.

Data Valuation. The need to quantify individual data point contributions has grown alongside large-scale ML data markets and dataset curation (Zhou et al., 2017; Pei, 2020; Sim et al., 2022). Classical influence functions approximate LOO retraining efficiently but rely on strong convexity assumptions (Koh & Liang, 2017; Basu et al., 2021; Bae et al., 2022). Data Shapley (Ghorbani & Zou, 2019) applies the Shapley value to training data, with Beta Shapley and Data Banzhaf improving robustness via semi-values (Kwon & Zou, 2022; Wang & Jia, 2023). Maximal Sample Reuse estimators improve scalability by reusing utility evaluations across MC coalitions, substantially improving sampling efficiency (Wang & Jia, 2023; Li & Yu, 2023; 2024; Liu et al., 2024). DataModels (Ilyas et al., 2022) fit a predictive proxy over massive model retrainings to estimate data contributions, with subsequent works reducing this retraining cost (Park et al., 2023; Engstrom et al., 2024). All of these methods assume i.i.d. samples and fundamentally fail to capture the structural dependencies inherent in graph-structured data Chi et al. (2025a).

Reusable Statistical Inference. RLS and the Kalman filter provide the classical framework for optimal online estimation, incrementally recovering exact linear model parameters as new observations arrive (Haykin, 1986; Plackett, 1950; Sherman & Morrison, 1950). The Sherman–Morrison formula further enables rank-one updates to matrix inverses in closed form, avoiding full recomputation (Sherman & Morrison, 1950). Streaming variants of ridge regression extend this to regularized objectives, maintaining exact solutions under sequential data additions (Orabona, 2019). SGVR leverages this linearity to enable streaming parameter updates, allowing efficient model retraining incrementally rather than from scratch at each valuation step.

Graph Neural Networks. GNNs extend deep learning to graph-structured data by iteratively aggregating neighborhood information via message passing (Bruna et al., 2013; Hamilton et al., 2017; Velickovic et al., 2017; Xu et al., 2018; Ju et al., 2026). Graph Convolutional Networks (GCNs) (Kipf & Welling, 2016) formalize this as a spectral graph convolution, enabling semi-supervised node classification by propagating features across the graph Laplacian. SGC (Wu et al., 2019) simplifies GCNs by collapsing the nonlinear activations between layers into a single linear transformation, yielding a model equivalent to ridge regression on fixed k-hop propagated features. SGVR directly exploits this linearity to enable exact incremental model updates. Training GNNs at scale introduces

further challenges around neighborhood explosion and evaluation (Chen et al., 2017; Liu et al., 2021; Shchur et al., 2018; Jegelka, 2022).

Influence and Explainability on Graphs. Influence functions approximate the contribution of graph elements via LOO perturbations (Chen et al., 2022). Alternatively, post-hoc Shapley methods attribute a fixed GNN's predictions to its input features across exponentially many feature coalitions (Duval & Malliaros, 2021; Akkas & Azad, 2024; Bui et al., 2024; Chhablani et al., 2024). While useful for explainability, neither approach quantifies how training graph data points contribute to GNN model performance, which requires the exhaustive subset analysis of exact game-theoretic data valuation (Chi et al., 2025a).

Graph Data Valuation. PC-Winter (Chi et al., 2025a) introduces the first game-theoretic framework for graph data valuation, attributing node contributions via the Winter value (Winter, 1989) over LCSs induced by GNN computation trees. Rather than averaging over all permutations as in Data Shapley Ghorbani & Zou (2019), PC-Winter restricts to *permissible* permutations satisfying two natural constraints that enforce the hierarchical coalition structure and the precedence ordering of the contribution tree. PC-Winter approximates these values via MC sampling of pre-order DFS traversals (Chi et al., 2025a). Furthermore, Chi et al. (2025b) extend graph valuation to inference time, applying Shapley-guided utility learning to quantify unlabeled node contributions at test time. Tang et al. (2026) extend Shapley-based valuation to heterogeneous graphs with type constraints, and Falahati & Amiri (2024) value graph data without a downstream task by disentangling structural and featural representations.

Antonelli & Bojchevski (2025) provide the first benchmark of valuation methods on graph data, evaluating a representative suite of i.i.d. methods, including Data Shapley, Data Banzhaf, DataModels, and LOO, alongside PC-Winter in the transductive semi-supervised setting. Notably, these i.i.d. methods are fundamentally agnostic to the structural nature of graph valuation: their sampling frameworks treat nodes as exchangeable, and their model training and signal propagation pipelines ignore the relational dependencies that graph-aware methods are specifically designed to exploit. Nevertheless, their benchmark concludes that graph-aware approaches do not consistently outperform i.i.d. counterparts, a conclusion that however reflects evaluation design choices more than a fundamental limitation of graph-aware valuation. Their framework conflates labeled and unlabeled nodes under a unified ranking task and moreover applies truncation ratios unrealistically high for graph-aware methods (Chi et al., 2025a), discarding the structural signals PC-Winter is designed to capture and rendering the comparison inconclusive. Driven by i.i.d. assumptions, their pipeline performs graph construction and model retraining in isolation for every coalition with no reuse of intermediate computations, resulting in runtimes of *days* even on moderately sized graphs, despite the rich structural redundancy that graphs naturally afford.

C Formal Foundations

Our notation borrows formulations and notational standards from Chi et al. (2025a); Antonelli & Bojchevski (2025).

C.1 Cooperative Games and Shapley Value

Let $\mathcal{D} = \{1, \ldots, n\}$ be a set of n players (data points) and let $\mathcal{U} : 2^{\mathcal{D}} \to \mathbb{R}$ be a utility function mapping any subset $S \subseteq \mathcal{D}$ to a scalar score, defined as the performance of a model trained on S and evaluated on a held-out validation set. The tuple $(\mathcal{D}, \mathcal{U})$ defines a *cooperative game*.

The **Shapley value** Shapley (1988) of player $i \in \mathcal{D}$ is the weighted average of its marginal contributions over all possible subsets:

$$\phi_i(\mathcal{D}, \mathcal{U}) = \sum_{S \subseteq \mathcal{D} \setminus \{i\}} \frac{|S|!\,(n - |S| - 1)!}{n!} \left[\mathcal{U}(S \cup \{i\}) - \mathcal{U}(S)\right] \tag{A.1}$$

Equivalently, ϕ_i can be expressed as an unweighted average over all $n!$ permutations $\pi \in \Pi(\mathcal{D})$:

$$\phi_i(\mathcal{D},\mathcal{U}) = \frac{1}{n!} \sum_{\pi \in \Pi(\mathcal{D})} [\mathcal{U}(S_i^\pi \cup \{i\}) - \mathcal{U}(S_i^\pi)] \tag{A.2}$$

where $S_i^\pi = \{j \in \mathcal{D} : \pi(j) < \pi(i)\}$ denotes the *predecessor set* of i in permutation π, i.e., the set of players appearing before i, and $\pi(i)$ denotes the position of player i in π.

C.2 Semi-Values

Relaxing the *efficiency axiom* of the Shapley value axioms Shapley (1988), which requires $\sum_{i\in\mathcal{D}} \phi_i = \mathcal{U}(\mathcal{D})$, yields the broader family of **semi-values** Dubey et al. (1981). A semi-value assigns to each player i a value of the form:

$$\psi_i^{w_n}(\mathcal{D},\mathcal{U}) = \sum_{S \subseteq \mathcal{D}\setminus\{i\}} w_n(|S|)\,[\mathcal{U}(S \cup \{i\}) - \mathcal{U}(S)] \tag{A.3}$$

where $w_n : \{0,\dots,n-1\} \to \mathbb{R}_{\geq 0}$ is a weighting function satisfying:

$$\sum_{k=0}^{n-1} \binom{n-1}{k} w_n(k) = 1 \tag{A.4}$$

Different choices of w_n recover well-known valuation methods:

- **Shapley value** Shapley (1988): $w_n(k) = \dfrac{k!\,(n-k-1)!}{n!}$
- **Data Banzhaf** Wang & Jia (2023): $w_n(k) = \dfrac{1}{2^{n-1}}$
- **LOO** Antonelli & Bojchevski (2025): $w_n(k) = \mathbb{1}_{k=n-1}$

C.3 Datamodels

Datamodels Ilyas et al. (2022) treat data valuation as a regression problem over subsets. Each subset $S \subseteq \mathcal{D}$ is encoded as a binary indicator vector $\mathbf{1}_S \in 0,1^n$; the utility is approximated by a linear surrogate

$$\mathcal{U}(S) \approx \mathbf{1}_S^\top \beta + \beta_0, \tag{A.5}$$

trained on many $(S,\mathcal{U}(S))$ pairs Ilyas et al. (2022); Wang & Jia (2022). Here, β_0 captures the null performance, and the learned coefficient β_i is interpreted as the datamodel value of player i, summarizing its average marginal contribution to $\mathcal{U}(S)$ under the chosen subset-sampling scheme and regularization Ilyas et al. (2022).

Antonelli & Bojchevski (2025) extend datamodels to graph data at the node level, despite the non-i.i.d. nature of graph-structured data. Subsets are constructed by sampling each node independently with inclusion probability $\alpha \in (0,1)$, which governs the expected fraction of nodes retained per subset, thereby controlling the scale at which marginal contributions are estimated.

C.4 Winter Value

A **Level Coalition Structure** Winter (1989) organizes $\mathcal{D}$ into a sequence of $L+1$ nested partitions $\mathcal{B} = (\mathcal{B}_0, \mathcal{B}_1, \dots, \mathcal{B}_L)$, where at each level ℓ the players are partitioned into m_ℓ sub-coalitions $\mathcal{B}_\ell = \{\mathcal{B}_\ell^1, \dots, \mathcal{B}_\ell^{m_\ell}\}$. The sequence satisfies $\mathcal{B}_0 \succeq \cdots \succeq \mathcal{B}_L$ under the coarsening relation (i.e., $\mathcal{B}_\ell \succeq \mathcal{B}_{\ell+1}$ iff $\forall\, \mathcal{B}_{\ell+1}^k,\ \exists\, j\ :\ \mathcal{B}_{\ell+1}^k \subseteq \mathcal{B}_\ell^j$). The *ground coalition* $\mathcal{B}_0 = \{\mathcal{D}\}$ is the root level, containing all players in a single coalition, and $\mathcal{B}_L = \{\{i\} : i \in \mathcal{D}\}$ is the leaf level, treating each player as a singleton.

The **Winter value** Winter (1989); Chantreuil (2000) restricts the Shapley average (Eq. (A.2)) to the subset $\Pi^{\mathcal{B}}(\mathcal{D}) \subseteq \Pi(\mathcal{D})$ of permutations where members of the same sub-coalition at every level appear contiguously (See Eq. (A.7)):

$$\phi_i^{\mathcal{B}}(\mathcal{D},\mathcal{U}) = \frac{1}{|\Pi^{\mathcal{B}}(\mathcal{D})|} \sum_{\pi \in \Pi^{\mathcal{B}}(\mathcal{D})} [\mathcal{U}(S_i^\pi \cup \{i\}) - \mathcal{U}(S_i^\pi)] \tag{A.6}$$

Unlike semi-values Dubey et al. (1981), which are characterized by a universal weight kernel $w_n(k)$ depending only on coalition size, the Winter value does not reduce to such a kernel; the effective weight of each subset is determined by the levels structure $\mathcal{B}$, and thus varies across instances.

C.5 PC-Winter Value

PC-Winter Chi et al. (2025a) extends the Winter value to graph data valuation by instantiating $\mathcal{B}$ from the GNN computation tree $\mathcal{T}$. Each labeled node $v \in \mathcal{V}_L$ induces an independent computation tree $\mathcal{T}_v$ of depth K, where each node is a *duplicate player* encoding a graph node's contribution through some path at a specific propagation hop. Accordingly, the level structure $\mathcal{B} = (\mathcal{B}_0, \mathcal{B}_1, \ldots, \mathcal{B}_K)$ consists of $K+1$ levels, where each sub-coalition $\mathcal{B}_\ell^j$ corresponds to the set of induced duplicate nodes in the subtree rooted at the j-th node at hop depth ℓ in $\mathcal{T}$. This subtree structure means every $\mathcal{B}_\ell^j$ has a unique root node $p_\ell^j = \mathrm{lca}(\mathcal{B}_\ell^j)$, defined as the lowest common ancestor of all nodes in $\mathcal{B}_\ell^j$, which induces a natural partial order among sub-coalitions that directly motivates the two constraints below.

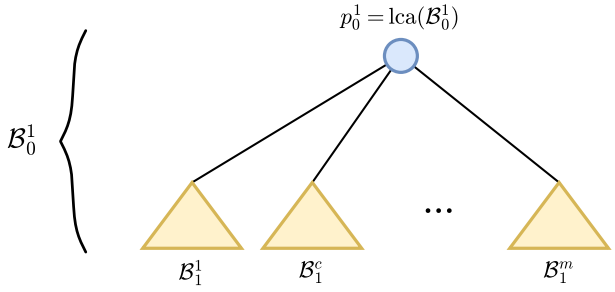


Figure 5: The $\mathcal{B}$ breakdown of a computation tree: each sub-coalition $\mathcal{B}_\ell^j$ (triangle) is the subtree rooted at duplicate player p_ℓ^j (circle), as in the union identity of Eq. (A.12).

The **Level Constraint** $\mathcal{C}_{\text{lev}}$, introduced by Chi et al. (2025a) as an extension of the level coalition structure Winter (1989) to the GNN setting, requires all members of every sub-coalition $\mathcal{B}_\ell^j$ together with their lowest common ancestor p_ℓ^j to appear contiguously in π:

$$\mathcal{C}_{\text{lev}} = \left\{ \pi : \forall \ell,\, \forall \mathcal{B}_\ell^j,\, \max_{x \in \mathcal{B}_\ell^j \cup \{p_\ell^j\}} \{\pi(x)\} - \min_{x \in \mathcal{B}_\ell^j \cup \{p_\ell^j\}} \{\pi(x)\} = |\mathcal{B}_\ell^j| \right\} \tag{A.7}$$

The **Precedence Constraint** $\mathcal{C}_{\text{prec}}$ encodes the partial order induced by $\mathcal{T}$: every parent node must precede its children in π, since unlabeled nodes cannot affect model performance without a path to a labeled node Chi et al. (2025a). Letting $\rho(c)$ denote the direct parent of node c in $\mathcal{T}$:

$$\mathcal{C}_{\text{prec}} = \{\pi : \forall\, c \in \mathcal{T},\ \pi(\rho(c)) < \pi(c)\} \tag{A.8}$$

The PC-Winter value is thus a Winter value further refined by $\mathcal{C}_{\text{prec}}$: a permutation π is *permissible* iff it satisfies both constraints simultaneously Chi et al. (2025a), giving:

$$\Pi^{\mathcal{T}}(\mathcal{D}) = \mathcal{C}_{\text{lev}} \cap \mathcal{C}_{\text{prec}} \subseteq \Pi^{\mathcal{B}}(\mathcal{D}) \subseteq \Pi(\mathcal{D}) \tag{A.9}$$

The **PC-Winter value** of duplicate node v is then:

$$\phi_v^{\text{PCW}}(\mathcal{D},\mathcal{U},\mathcal{T}) = \frac{1}{|\Pi^{\mathcal{T}}(\mathcal{D})|} \sum_{\pi \in \Pi^{\mathcal{T}}(\mathcal{D})} [\mathcal{U}(S_v^\pi \cup \{v\}) - \mathcal{U}(S_v^\pi)] \tag{A.10}$$

The node value of v aggregates its duplicate players across all hops Chi et al. (2025a). Exact computation grows factorially with the number of players, and Chi et al. (2025a) approximate

via Monte Carlo sampling, drawing pre-order DFS traversals of $\mathcal{T}$ that satisfy both constraints by construction. We refer to this baseline as *uncoordinated* sampling, since each duplicate player's predecessor set is drawn from an independent permutation. In Section C.6, we first reformulate PC-Winter as a sum over distinct *permissible predecessor sets* rather than permissible permutations, with each set weighted by its multiplicity, and show that this reformulation is equivalent to Eq. (A.10). Building on this, we then introduce *coordinated* sampling, a hierarchical Monte Carlo estimator that shares predecessor sets across duplicate players among different samples, yielding tighter variance bounds for ranking-based downstream tasks.

C.6 Hierarchical Reusable Structures

In this section we approach Eq. (A.10) from a counting perspective, deriving recurrence rules along the structure of $\mathcal{T}$ that expose the redundancy in summing over all permissible permutations. We first count $|\Pi^{\mathcal{T}}(\mathcal{D})|$, then observe that many permissible permutations produce the same predecessor set at the same position, inducing identical model states and therefore redundant marginal evaluations, and count the distinct predecessor sets they generate. The gap between the two recurrences quantifies the redundancy and provides the foundation for the reformulation and the hierarchical Monte Carlo estimator (see Section C.8).

Let $\mathcal{K}(\mathcal{B}_\ell^j)$ denote the immediate children of sub-coalition $\mathcal{B}_\ell^j$ at level $\ell{+}1$:

$$\mathcal{K}(\mathcal{B}_\ell^j) = \left\{ \mathcal{B}_{\ell+1}^c : c \in \{1, \dots, m_{\ell+1}\},\ \mathcal{B}_{\ell+1}^c \subseteq \mathcal{B}_\ell^j \right\} \tag{A.11}$$

Each sub-coalition equals the disjoint union of its children together with its root player p_ℓ^j:

$$\mathcal{B}_\ell^j = \{p_\ell^j\} \ \sqcup \bigsqcup_{\mathcal{B}_{\ell+1}^c \in \mathcal{K}(\mathcal{B}_\ell^j)} \mathcal{B}_{\ell+1}^c \tag{A.12}$$

Counting permissible permutations. The level constraint forces every child sub-coalition to appear as a contiguous block, with internal orderings independent across blocks (Figure 6). By the product rule, the number of permissible permutations within $\mathcal{B}_\ell^j$ satisfies the recurrence

$$\Gamma(\mathcal{B}_\ell^j) = |\mathcal{K}(\mathcal{B}_\ell^j)|! \prod_{\mathcal{B}_{\ell+1}^c \in \mathcal{K}(\mathcal{B}_\ell^j)} \Gamma\big(\mathcal{B}_{\ell+1}^c\big) \tag{A.13}$$

with base case $\Gamma(\mathcal{B}_L^j) = 1$ for singleton leaves, so $|\Pi^{\mathcal{T}}(\mathcal{D})| = \Gamma(\mathcal{B}_0^1)$. Each permissible permutation requires $|\mathcal{D}| + 1$ model retrainings, giving total evaluation complexity

$$O\left(|\mathcal{D}| \cdot |\mathcal{K}(\mathcal{B}_0^1)|! \prod_{\mathcal{B}_1^c \in \mathcal{K}(\mathcal{B}_0^1)} \Gamma(\mathcal{B}_1^c) \right). \tag{A.14}$$

This multiplicative scaling leads to factorial growth in the branching at every level.

Counting distinct predecessor sets. Accordingly, these permutations introduce redundancy: distinct $\pi, \pi' \in \Pi^{\mathcal{T}}(\mathcal{D})$ frequently produce the same predecessor set at the same position, inducing identical model states and therefore yielding identical marginal contributions. We say π and π' are *prefix-equivalent at position j* if

$$\pi \sim_j \pi' \iff \exists\, v \in \mathcal{D} \text{ s.t. } \pi(v) = j :\ S_v^\pi = S_v^{\pi'}. \tag{A.15}$$

Given this structural redundancy, a prefix-equivalence class need only be evaluated once. Let $\Lambda(\mathcal{B}_\ell^j)$ denote the number of distinct predecessor sets generated within $\mathcal{B}_\ell^j$ under maximal structural reuse. Since $\mathcal{C}_{\text{lev}}$ treats each subtree of $\mathcal{B}_\ell^j$ as a complete atomic block, the children $\mathcal{K}(\mathcal{B}_\ell^j)$ can be processed independently[3]: no multiplicative interaction arises between subtrees, and the $|\mathcal{K}(\mathcal{B}_\ell^j)|!$

[3]Unlike $\Gamma(\mathcal{B}_\ell^j)$, which counts permutations and scales by permutation length to obtain retraining cost, $\Lambda(\mathcal{B}_\ell^j)$ directly counts model retrainings, one per unique predecessor set rather than one per permutation position.

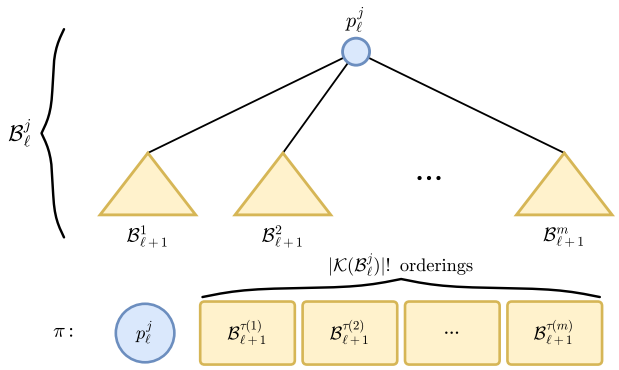


Figure 6: Counting permissible permutations: child blocks are permuted in $|\mathcal{K}(B)|!$ ways, each internally ordered in $\Gamma(\cdot)$ ways, as in the recurrence of Eq. (A.13).

child orderings reduce to $2^{|\mathcal{K}(\mathcal{B}_\ell^j)|-1}$ distinct predecessor sets per level (Figure 7), analogous to the DP of Held & Karp (1962). This yields the recurrence

$$\Lambda(\mathcal{B}_\ell^j) = 1 + 2^{|\mathcal{K}(\mathcal{B}_\ell^j)|-1} \sum_{\mathcal{B}_{\ell+1}^c \in \mathcal{K}(\mathcal{B}_\ell^j)} \Lambda\big(\mathcal{B}_{\ell+1}^c\big) \tag{A.16}$$

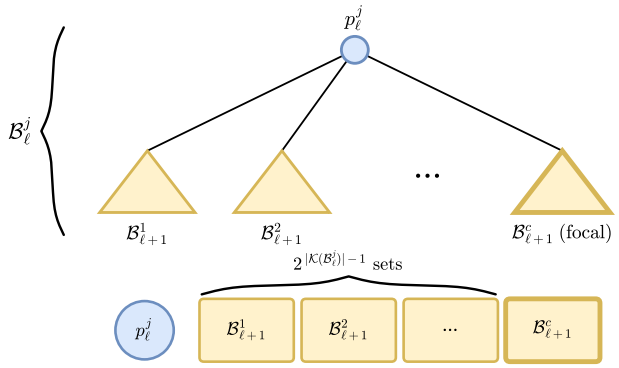


Figure 7: Counting distinct predecessor sets: for a focal child, each non-focal sibling is independently included or excluded, giving $2^{|\mathcal{K}(B)|-1}$ inclusion patterns (predecessor sets) per level instead of $|\mathcal{K}(B)|!$ permutations, as in the recurrence of Eq. (A.16).

with base case $\Lambda(\mathcal{B}_L^j) = 1$, and total complexity

$$O\left(2^{|\mathcal{K}(\mathcal{B}_0^1)|-1} \sum_{\mathcal{B}_1^c \in \mathcal{K}(\mathcal{B}_0^1)} \Lambda\big(\mathcal{B}_{\ell+1}^c\big)\right) \tag{A.17}$$

once each predecessor set is evaluated exactly once.

Complexity on a d-regular graph. Consider a d-regular graph with L labeled nodes under a 2-layer GNN ($K = 2$). Each labeled node has d hop-1 neighbors, each with d hop-2 neighbors, giving $|\mathcal{D}| = 1+L+dL+d^2L$ duplicate players. The permissible-permutation count from Eq. (A.14) gives

$$\big(1 + L + dL + d^2L\big) \cdot L! \cdot (d!)^{L(d+1)} \tag{A.18}$$

retrainings, while the distinct predecessor sets from Eq. (A.17) reduce this to

$$\underbrace{1+2^{L-1}}_{\text{labeled}}\cdot\sum_{l=1}^{L}\left(\underbrace{1+2^{d-1}}_{\text{hop-1}}\cdot\sum_{h=1}^{d}\left(\underbrace{1+2^{d-1}}_{\text{hop-2}}\cdot\sum_{h'=1}^{d}1\right)\right)=1+2^{L-1}L+2^{L+d-2}Ld+2^{L+2d-3}Ld^2, \tag{A.19}$$

collapsing the cost from $O\big(d^2L\cdot L!\,(d!)^{L(d+1)}\big)$ to $O\big(2^{L+2d-3}Ld^2\big)$, a superexponential-to-exponential reduction in L and d. The same counting gap underlies the predecessor-set reformulation of Eq. (A.10) in Section C.7 and the hierarchical Monte Carlo estimator that materializes this shared structure in Section C.8.

C.7 Equivalence of Both Formulations

Building on the recurrences of Section C.6, we now present their realizations as algorithms and establish their equivalence. Algorithm 1 (*permutation-based enumeration*) instantiates Eq. (A.13) by recursively constructing every $\pi\in\Pi^{\mathcal{T}}(\mathcal{D})$ from the permutations of its child sub-coalitions, and traversing each π left to right to accumulate marginals. Algorithm 1 incurs the $|\mathcal{D}|\cdot\Gamma(\mathcal{B}_0^1)$ cost of Eq. (A.14). Algorithm 2 (*subset-based enumeration*) instantiates Eq. (A.16) by recursively enumerating, at each block, every subset of siblings included in the predecessor set of the *focal* child, and aggregating contributions over distinct predecessor sets, incurring the $\Lambda(\mathcal{B}_0^1)$ cost of Eq. (A.17). Theorem 3 establishes that the two procedures yield the same PC-Winter value, from which we obtain an exact reformulation of Eq. (A.10) as a sum over distinct predecessor sets.

By Eq. (A.15), permissible permutations that are prefix-equivalent at the position of player v produce the same predecessor set S_v^π and therefore the same marginal contribution $\mathcal{U}(S_v^\pi\cup\{v\})-\mathcal{U}(S_v^\pi)$, due to the order-invariance property of $\mathcal{U}$. We write

$$\delta_v(S)\coloneqq\mathcal{U}(S\cup\{v\})-\mathcal{U}(S) \tag{A.20}$$

for the marginal contribution of player v to coalition S, so that player v's marginal contribution under π is $\delta_v(S_v^\pi)$. We call any set of the form S_v^π realized by some $\pi\in\Pi^{\mathcal{T}}(\mathcal{D})$ a *permissible predecessor set*, and denote the collection of permissible predecessor sets reachable for player p by

$$\mathfrak{S}_p=\{S_p^\pi:\pi\in\Pi^{\mathcal{T}}(\mathcal{D})\},\qquad|\mathfrak{S}_p|\le\Lambda(\mathcal{B}_0^1). \tag{A.21}$$

For fixed $q\in\mathcal{D}$ and $S\subseteq\mathcal{D}$, define $M(\mathcal{B}_0^1,q,S)$ as the number of permissible permutations of $\mathcal{B}_0^1$ in which q's permissible predecessor set is exactly S; this is the multiplicity with which the pair (q,S) appears across $\Pi^{\mathcal{T}}$.

For some q let

$$\mathcal{B}_0^1\supset\mathcal{B}_1^{j_1}\supset\cdots\supset\mathcal{B}_\ell^{j_\ell}\ni q$$

be the unique chain of nested ancestor blocks terminating at $\mathcal{B}_\ell^{j_\ell}$, the block whose root player is q, i.e., $p_\ell^{j_\ell}=q$. At each block $\mathcal{B}_k^{j_k}$, the chain descends into a unique *focal child* $\mathcal{B}_{k+1}^{j_{k+1}}$; the remaining children of $\mathcal{B}_k^{j_k}$ partition into a subset Σ_k of *included siblings*, whose sub-coalitions enter the predecessor set of the focal child, and a complementary set of *excluded siblings* (Figure 8). By the product rule, the number of permissible orderings of $\mathcal{B}_k^{j_k}$ consistent with a given Σ_k is

$$w(\Sigma_k)=|\Sigma_k|!\,\big(|\mathcal{K}(\mathcal{B}_k^{j_k})|-|\Sigma_k|-1\big)!\prod_{B'\in\mathcal{K}(\mathcal{B}_k^{j_k})\setminus\{\mathcal{B}_{k+1}^{j_{k+1}}\}}\Gamma(B'), \tag{A.22}$$

counting $|\Sigma_k|!$ orderings of the included siblings, $(|\mathcal{K}(\mathcal{B}_k^{j_k})|-|\Sigma_k|-1)!$ orderings of the excluded siblings, and $\Gamma(B')$ internal orderings of each off-chain sibling B'.

The chain $(\Sigma_0,\dots,\Sigma_{\ell-1})$ uniquely determines the predecessor set

$$S=\bigcup_{k=0}^{\ell-1}\big(\{p_k^{j_k}\}\cup\bigsqcup_{B'\in\Sigma_k}B'\big). \tag{A.23}$$

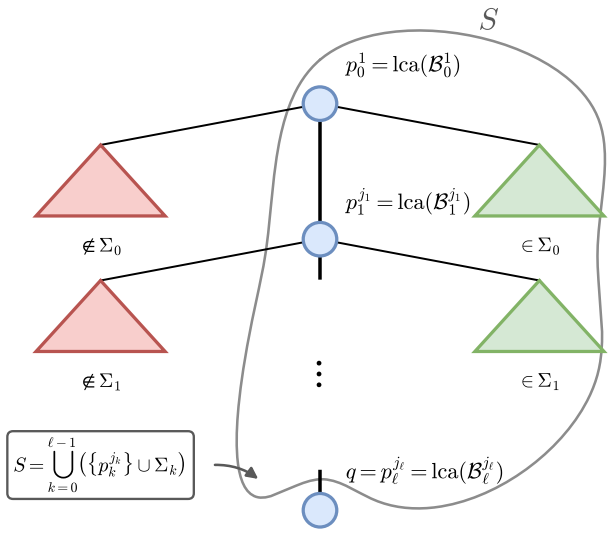


Figure 8: The ancestor chain to q: at each level the focal child continues the descent while siblings split into included (Σ_k, entering S) and excluded, as in the construction of S in Eq. (A.23).

Since the local orderings at distinct blocks are chosen independently, the total count of permissible permutations realizing S is the product of local counts along the chain, multiplied by $\Gamma(\mathcal{B}_\ell^{j_\ell})$ to account for the internal orderings of q's own block:

$$M(\mathcal{B}_0^1, q, S) \;=\; \Gamma(\mathcal{B}_\ell^{j_\ell}) \prod_{k=0}^{\ell-1} w(\Sigma_k). \tag{A.24}$$

Summing the multiplicity over all permissible predecessor sets of q recovers the total number of permissible permutations:

$$\sum_{S \in \mathfrak{S}_q} M(\mathcal{B}_0^1, q, S) = \Gamma(\mathcal{B}_0^1), \quad \forall\, q \in \mathcal{D}. \tag{A.25}$$

Theorem 3 (Equivalence of permutation-based and subset-based enumeration)**.** *For every $p \in \mathcal{D}$, Algorithms 1 and 2 return the same ϕ_p^{PCW}. Algorithm 1 performs $|\mathcal{D}| \cdot \Gamma(\mathcal{B}_0^1)$ marginal evaluations; Algorithm 2 performs $\Lambda(\mathcal{B}_0^1)$.*

Proof sketch. By order-invariance of $\mathcal{U}$, the marginal at π depends only on S_p^π, so both algorithms compute a weighted sum of marginals over predecessor sets, and it suffices to show that the weight assigned to each (p, S) pair equals $M(\mathcal{B}_0^1, p, S)$. Algorithm 1 attains this weight by direct enumeration: P-ENUM generates each π exactly once, contributing a single marginal at S_p^π, so the cumulative weight on (p, S) equals $|\{\pi : S_p^\pi = S\}| = M(\mathcal{B}_0^1, p, S)$. Algorithm 2 attains it by induction on p's ancestor chain: each recursive call into a focal child multiplies the running weight m by $w(\Sigma_k)$, and the accumulation step at p contributes $m \cdot \Gamma(\mathcal{B}_\ell^{j_\ell})$ to ϕ_p, recovering Eq. (A.24). The level constraint ensures that the chain $(\Sigma_0, \ldots, \Sigma_{\ell-1})$ uniquely determines S, so each (p, S) pair is reached along exactly one descent path. The complexity bounds follow from the recurrences of Section C.6. □

PC-Winter as a sum over permissible predecessor sets. The equivalence yields an exact reformulation of Eq. (A.10): substituting M into the permutation sum and collapsing permissions with identical predecessor sets,

$$\phi_p^{\mathrm{PCW}} = \frac{1}{\Gamma(\mathcal{B}_0^1)} \sum_{S \,\in\, \mathfrak{S}_p} M(\mathcal{B}_0^1, p, S) \left[\mathcal{U}(S \cup \{p\}) - \mathcal{U}(S)\right]. \tag{A.26}$$

The sum runs over $|\mathfrak{S}_p| \le \Lambda(\mathcal{B}_0^1)$ terms, replacing the $|\mathcal{D}| \cdot \Gamma(\mathcal{B}_0^1)$ marginals of the permutation form. This subset form is the foundation for the hierarchical Monte Carlo estimator in Section C.8.

Algorithm 1 Permutation-based enumeration

Require: Tree $\mathcal{T}$, root $\mathcal{B}_0^1$, oracle $\mathcal{U}$
$\phi_p \leftarrow 0$ for all $p \in \mathcal{D}$
for each $\pi \in$ P-ENUM$(\mathcal{B}_0^1)$ **do** ▷ enumerate all permissible permutations
  $S \leftarrow \emptyset$
  for each v in π left to right **do**
    $\phi_v \mathrel{+}= \delta_v(S)$ ▷ accumulate marginal at predecessor set S
    $S \leftarrow S \cup \{v\}$
  end for
end for
return $\phi_p/\Gamma(\mathcal{B}_0^1)$ for all p

procedure P-ENUM(B) ▷ realizes Eq. (A.13)
  if $\mathcal{K}(B) = \emptyset$ **then return** $\{[p_B]\}$ ▷ leaf: a singleton sequence of length one
  end if
  $L_c \leftarrow$ P-ENUM(c) for each $c \in \mathcal{K}(B)$ ▷ recursively enumerate per child
  return $\big\{\, [p_B] \,\|\, \pi_1 \,\|\, \cdots \,\|\, \pi_m \;:$
  $(c_1, \ldots, c_m)$ is an ordering of $\mathcal{K}(B)$, $\pi_i \in L_{c_i} \,\big\}$
  ▷ $\|$ denotes sequence concatenation; root p_B placed first, then children in order
end procedure

Algorithm 2 Subset-based enumeration

Require: Tree $\mathcal{T}$, root $\mathcal{B}_0^1$, oracle $\mathcal{U}$
$\phi_p \leftarrow 0$ for all $p \in \mathcal{D}$
S-ENUM$(\mathcal{B}_0^1, \emptyset, 1)$ ▷ begin descent at root with empty predecessor set and weight 1
return $\phi_p/\Gamma(\mathcal{B}_0^1)$ for all p

procedure S-ENUM(B, S, m) ▷ B: current block; S: predecessor set at p_B; $m = \prod_k w(\Sigma_k)$ accumulated along the descent
  $\phi_{p_B} \mathrel{+}= m \cdot \Gamma(B) \cdot \delta_{p_B}(S)$ ▷ contribute $M(\mathcal{B}_0^1, p_B, S)$-weighted marginal (Eq. (A.24))
  for each focal child $\mathcal{B}_{\ell+1}^c \in \mathcal{K}(B)$ **do**
    for each subset $\Sigma \subseteq \mathcal{K}(B) \setminus \{\mathcal{B}_{\ell+1}^c\}$ **do** ▷ Σ are the included siblings preceding the focal child
      $S_c \leftarrow S \cup \{p_B\} \cup \bigsqcup_{B' \in \Sigma} B'$
      S-ENUM$(\mathcal{B}_{\ell+1}^c, S_c, m \cdot w(\Sigma))$
    end for
  end for
end procedure

In both algorithms, $\delta_v(S)$ (Eq. (A.20)) denotes the marginal contribution of player v to coalition S.

C.8 HIERARCHICAL MONTE CARLO ESTIMATION

Despite the reduction afforded by Eq. (A.26), exact evaluation requires summing over $\Lambda(\mathcal{B}_0^1)$ permissible predecessor sets, which remains intractable at scale. The original PC-Winter estimator of Chi et al. (2025a) approximates Eq. (A.10) by drawing N permissible permutations of $\Pi^{\mathcal{T}}(\mathcal{D})$ uniformly at random and independently for each player; we refer to this scheme as *uncoordinated sampling*, since no randomness is shared across players.

We propose an alternative estimator based on *nested Monte Carlo* Rainforth et al. (2018) with shared draws across players, which we call *coordinated sampling*. The same per-player budget N is redistributed across the levels of the descent according to a factorization $N = h_1 h_2 \cdots h_L$, with

h_ℓ orderings drawn at level ℓ. Concretely, at each block $\mathcal{B}_\ell^{j_\ell}$ along q's ancestor chain we draw h_ℓ permissible orderings of its children $\mathcal{K}(\mathcal{B}_\ell^{j_\ell})$ uniformly at random, with each draw extending the orderings drawn at shallower levels, and the resulting nested draws are consumed by subsequent players. A single round produces $h_1 h_2 \cdots h_L = N$ descent paths, each terminating in a focal player and a permissible predecessor set; the estimator $\hat{\phi}_q$ is the empirical mean of the marginal contributions across these paths. Each path is marginally uniform over $\Pi^{\mathcal{T}}(\mathcal{D})$ by Eq. (A.13), so the induced sampling frequency over predecessor sets matches the weight $M(\mathcal{B}_0^1, q, S)/\Gamma(\mathcal{B}_0^1)$ required by Eq. (A.26). Both estimators are therefore unbiased for the exact PC-Winter value and share the same expected limit:

$$\lim_{N\to\infty} \mathbb{E}\big[\hat{\phi}_q^{\mathrm{unc}}\big] \;=\; \lim_{N\to\infty} \mathbb{E}\big[\hat{\phi}_q^{\mathrm{coord}}\big] \;=\; \phi_q^{\mathrm{PCW}}. \tag{A.27}$$

The two modes differ only in how the budget N is allocated across players: uncoordinated sampling spends it as N independent permutations per player, while coordinated sampling spends it as a nested factorization $h_1 h_2 \cdots h_L$ whose intermediate draws are shared across players. Any two players i, j therefore traverse identical sibling orderings at every block above their lowest common ancestor and increasingly share predecessor sets in deeper levels. The two modes agree in the limit but differ in finite-sample behavior. Theorem 4 characterizes the per-player variance of $\hat{\phi}_q$, and Theorem 5 compares the pairwise variance of the two modes under the matched budget N.

Theorem 4 (Per-player variance of the nested estimator)**.** *For the coordinated estimator with per-player budget $N = h_1 h_2 \cdots h_L$, the variance of $\hat{\phi}_q$ decomposes across the levels of q's ancestor chain as*

$$\mathrm{Var}\big(\hat{\phi}_q\big) = \frac{\sigma_{q,1}^2}{h_1} + \frac{\sigma_{q,2}^2}{h_1 h_2} + \cdots + \frac{\sigma_{q,L_q}^2}{h_1 h_2 \cdots h_{L_q}}, \tag{A.28}$$

where $\sigma_{q,\ell}^2$ is the conditional variance of the marginal contribution introduced by the draw at level ℓ, L_q is the depth of q's ancestor chain, and the total single-chain variance $\sigma_q^2 = \mathrm{Var}\big[\delta_q(S_q^\pi)\big]$, with $\delta_q(\cdot)$ the marginal-contribution function of Eq. (A.20)*, decomposes as*

$$\sigma_q^2 = \sum_{\ell=1}^{L_q} \sigma_{q,\ell}^2. \tag{A.29}$$

Proof sketch. Apply the law of total variance recursively from the innermost level outward. At the deepest level L_q, the within-group average has variance $\sigma_{q,L_q}^2/h_{L_q}$. Moving outward one level introduces a new contribution $\sigma_{q,\ell}^2/h_\ell$ and, by the variance of a sample mean of h_ℓ inner groups, divides every deeper term by an additional h_ℓ. Across all levels, the level-ℓ variance is divided by the cumulative product $h_1 h_2 \cdots h_\ell$, equal to the number of averaging operations applied to it on the descent to the root, which is the ANOVA decomposition of the variance across the levels of q's ancestor chain Fisher (1992). The identity in Eq. (A.29) follows as the special case $h_\ell = 1$. □

Theorem 5 (Coordinated sampling reduces pairwise variance)**.** *Let $\hat{\phi}_q^{\mathrm{unc}}$ denote the uncoordinated estimator of player q run for N independent permutations per player, and let $\hat{\phi}_q^{\mathrm{coord}}$ denote the coordinated estimator with the same budget $N = h_1 h_2 \cdots h_L$ redistributed across levels. For any pair $(i, j) \in \mathcal{D}$ with lowest common ancestor at level L_{ij}, let $\mathrm{Cov}_{\le L_{ij}} := \mathrm{Cov}\big(\hat{\phi}_i^{\mathrm{coord}}, \hat{\phi}_j^{\mathrm{coord}}\big)$ denote the covariance between the coordinated estimators, which arises entirely from the draws shared at and above the LCA. Then*

$$\mathrm{Var}\big(\hat{\phi}_i^{\mathrm{unc}} - \hat{\phi}_j^{\mathrm{unc}}\big) \;=\; \frac{\sigma_i^2 + \sigma_j^2}{N}, \tag{A.30}$$

$$\mathrm{Var}\big(\hat{\phi}_i^{\mathrm{coord}} - \hat{\phi}_j^{\mathrm{coord}}\big) \;=\; \mathrm{Var}\big(\hat{\phi}_i^{\mathrm{unc}} - \hat{\phi}_j^{\mathrm{unc}}\big) \;+\; \Delta_{ij} \;-\; 2\,\mathrm{Cov}_{\le L_{ij}}, \tag{A.31}$$

with $H_\ell = h_1 \cdots h_\ell$ and $\Delta_{ij} = \sum_\ell (\sigma_{i,\ell}^2 + \sigma_{j,\ell}^2)(1/H_\ell - 1/N) \ge 0$. Coordinated sampling therefore reduces pairwise variance if and only if $2\,\mathrm{Cov}_{\le L_{ij}} > \Delta_{ij}$. Below the LCA, i and j enter disjoint sub-coalitions and their draws are independent, so the level-wise contributions below the LCA contribute the entire independent portion of $\mathrm{Var}(\hat{\phi}_i^{\mathrm{coord}} - \hat{\phi}_j^{\mathrm{coord}})$, yielding the lower bound

$$\mathrm{Var}\big(\hat{\phi}_i^{\mathrm{coord}} - \hat{\phi}_j^{\mathrm{coord}}\big) \;\ge\; \sum_{\ell > L_{ij}} \frac{\sigma_{i,\ell}^2 + \sigma_{j,\ell}^2}{h_1 h_2 \cdots h_\ell}. \tag{A.32}$$

Equality in Eq. (A.32) *holds only under strong structural conditions on the value function (Remark 1); in general the right-hand side is a lower bound rather than the exact value. We check empirically whether the condition* $2\,\mathrm{Cov}_{\le L_{ij}} > \Delta_{ij}$ *holds in practice (Section D.2, Table 3).*

Remark 1 (When the floor is exact)**.** *The lower bound in Eq.* (A.32) *is attained exactly when the shared draws above the LCA contribute zero variance to* $\hat{\phi}_i^{\mathrm{coord}}$ *and* $\hat{\phi}_j^{\mathrm{coord}}$ *individually. A sufficient condition is that the value function is additively separable across the LCA split, i.e.,* $\mathcal{U}(S) = \mathcal{U}_A(S \cap A) + \mathcal{U}_B(S \cap B)$ *where* A *and* B *are the sub-coalitions containing* i *and* j *respectively; then the shared context affects* i *and* j *through disjoint channels and does not contribute variance to either. This condition does not hold for the GNN-based value functions considered here: the shared upstream context has genuine effect on both estimators, and the floor is a lower bound rather than an equality. Section D.2 instead checks the variance comparison directly on the collected estimates.*

Proof sketch. For both estimators, the variance of the difference decomposes as

$$\mathrm{Var}(\hat{\phi}_i - \hat{\phi}_j) = \mathrm{Var}(\hat{\phi}_i) + \mathrm{Var}(\hat{\phi}_j) - 2\,\mathrm{Cov}(\hat{\phi}_i, \hat{\phi}_j).$$

Under uncoordinated sampling, $\hat{\phi}_i$ and $\hat{\phi}_j$ use independent permutations, so $\mathrm{Cov} = 0$ and $\mathrm{Var}(\hat{\phi}_i - \hat{\phi}_j)$ equals $(\sigma_i^2 + \sigma_j^2)/N$ by Eq. (A.29), giving Eq. (A.30). Under coordinated sampling, the shared nested draws mean i and j traverse identical sibling orderings at every level $\ell \le L_{ij}$, while below the LCA their draws enter disjoint sub-coalitions and are independent, contributing $(\sigma_{i,\ell}^2 + \sigma_{j,\ell}^2)/(h_1 \cdots h_\ell)$ per level (Theorem 4) with zero cross-covariance. All covariance between $\hat{\phi}_i^{\mathrm{coord}}$ and $\hat{\phi}_j^{\mathrm{coord}}$ therefore comes from the shared levels $\ell \le L_{ij}$. Combining the variance-of-a-difference identity with the per-player variances of Eq. (A.28) gives $\sum_\ell (\sigma_{i,\ell}^2 + \sigma_{j,\ell}^2)/H_\ell - 2\,\mathrm{Cov}_{\le L_{ij}}$; subtracting Eq. (A.30) leaves $\Delta_{ij} - 2\,\mathrm{Cov}_{\le L_{ij}}$, which is Eq. (A.31). Splitting each single-estimator variance in Eq. (A.28) at level L_{ij} isolates the shared-level terms in the pairwise variance:

$$\mathrm{Var}\big(\hat{\phi}_i^{\mathrm{coord}} - \hat{\phi}_j^{\mathrm{coord}}\big) = \underbrace{\left[\sum_{\ell \le L_{ij}} \frac{\sigma_{i,\ell}^2 + \sigma_{j,\ell}^2}{h_1 \cdots h_\ell} \;-\; 2\,\mathrm{Cov}_{\le L_{ij}}\right]}_{\text{shared-level residual}} + \sum_{\ell > L_{ij}} \frac{\sigma_{i,\ell}^2 + \sigma_{j,\ell}^2}{h_1 \cdots h_\ell}.$$

By Cauchy–Schwarz, $2\,\mathrm{Cov}_{\le L_{ij}} \le \sum_{\ell \le L_{ij}} (\sigma_{i,\ell}^2 + \sigma_{j,\ell}^2)/(h_1 \cdots h_\ell)$, so the bracketed shared-level residual is non-negative; dropping it yields the floor in Eq. (A.32) directly. The bound is tight exactly when the residual is zero, i.e. when the shared-level contributions coincide (Remark 1). We do not assume this in general; instead Section D.2 compares the variances directly on the collected estimates. □

C.9 Minimal Propagation

Let $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ be a graph with feature matrix $\mathbf{X} \in \mathbb{R}^{|\mathcal{V}| \times F}$. We maintain k hop-indexed embedding graphs $\{\mathcal{G}^{(\ell)}\}_{\ell=0}^{k}$, where node u in $\mathcal{G}^{(\ell)}$ stores $\mathbf{h}_u^{(\ell)}$. Following (Chi et al., 2025a; Antonelli & Bojchevski, 2025), we adopt the SGC propagation rule (Wu et al., 2019):

$$\mathbf{H}^{(\ell)} = \hat{\mathbf{A}}\,\mathbf{H}^{(\ell-1)}, \quad \ell = 1, \ldots, k, \tag{A.33}$$

where $\hat{\mathbf{A}}$ is one of the aggregation matrices:

$$\hat{\mathbf{A}}^{\mathrm{asym}} = \tilde{\mathbf{D}}^{-1}\tilde{\mathbf{A}}, \qquad \hat{\mathbf{A}}^{\mathrm{sym}} = \tilde{\mathbf{D}}^{-\frac{1}{2}}\tilde{\mathbf{A}}\tilde{\mathbf{D}}^{-\frac{1}{2}}, \tag{A.34}$$

with $\tilde{\mathbf{A}} = \mathbf{A} + \mathbf{I}$, $\tilde{\mathbf{D}}$ the degree matrix, and $\mathbf{H}^{(0)} = \mathbf{X}$. Each application extends the receptive field by one hop, so $\mathbf{H}^{(\ell)}$ aggregates within $\mathcal{N}_\ell(v) = \{u \mid \mathrm{dist}(u, v) \le \ell\}$.

PC-Winter introduced **Local Propagation** (Chi et al., 2025a), which re-propagates the entire labeled subtree upon each insertion, costing $\mathcal{O}(d^k)$ on a d-regular graph in every incremental step. In this work, we introduce **Minimal Propagation**, which restricts each update to the affected frontier:

$$\mathbf{h}_u^{(\ell)} \leftarrow \sum_{w \in \tilde{\mathcal{N}}(u)} \hat{a}_{uw}\,\mathbf{h}_w^{(\ell-1)}, \qquad \forall\, u \in \mathcal{N}_\ell(v), \tag{A.35}$$

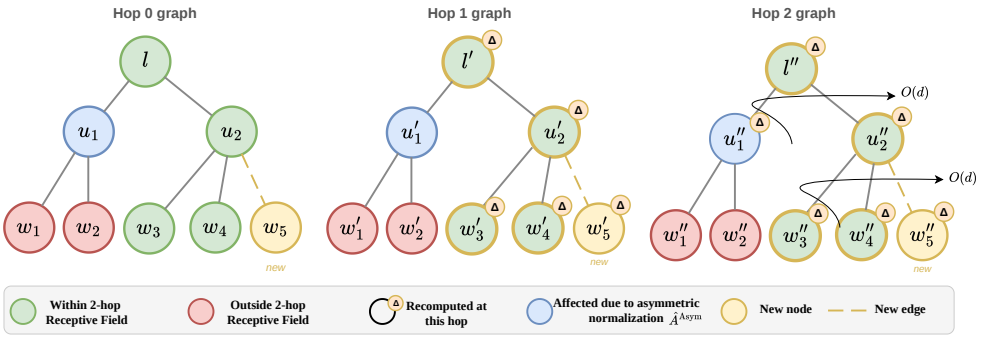


Figure 9: Minimal Propagation under $\hat{\mathbf{A}}^{\text{sym}}$ after inserting leaf w_5. Green nodes lie within the extended frontier $\mathcal{N}_{k+1}(v)$; red nodes are outside and carried forward unchanged. In $\mathcal{G}^{(1)}$, $\{l', u'_2, w'_3, w'_4\}$ are recomputed in-place; in $\mathcal{G}^{(2)}$, $\{u''_1, l'', u''_2, w''_3, w''_4\}$ are updated. All remaining embeddings are carried forward unchanged.

with all nodes outside the frontier carried forward unchanged . Under $\hat{\mathbf{A}}^{\text{asym}}$, row normalization $\tilde{\mathbf{D}}^{-1}$ depends solely on the source node, so inserting v affects exactly $\mathcal{N}_k(v)$: any u with $\text{dist}(u, v) > k$ is guaranteed to remain unchanged and the receptive field boundary is strictly respected. Under $\hat{\mathbf{A}}^{\text{sym}}$, however, inserting edge (u, v) increments $\tilde{d}_u$ and $\tilde{d}_v$, perturbing $\hat{a}_{zw} = (\tilde{d}_z\, \tilde{d}_w)^{-1/2}$ for all edges incident to $\mathcal{N}_1(u) \cup \mathcal{N}_1(v)$; the true update frontier therefore extends one hop beyond the nominal receptive field to $\mathcal{N}_{k+1}(v)$. The update propagates outward via BFS, making in-place changes exclusively to affected nodes at each hop, leaving the rest of the embedding graphs untouched. In this work, we adopt $\hat{\mathbf{A}}^{\text{sym}}$, as it is the standard choice in the literature (Wu et al., 2019; Chi et al., 2025a); Figure 9 illustrates the resulting frontier expansion after inserting leaf w_5. The subsequent complexity analysis assumes $\hat{\mathbf{A}}^{\text{sym}}$, where the true update frontier extends to $\mathcal{N}_{k+1}(v)$: at hop 1, $\{l', u'_2, w'_3, w'_4\}$ are recomputed, and at hop 2 the frontier expands to $\{u''_1, l'', u''_2, w''_3, w''_4\}$, while all remaining nodes are never touched. The per-insertion cost is $\mathcal{O}(d^k)$ in the worst case; for leaf insertions, however, $|\mathcal{N}_1(v)| = 1$ collapses the frontier at each hop, reducing the cost to $\mathcal{O}(d^{k-1})$, a strict improvement over full subtree re-propagation. Minimal Propagation is particularly well-suited for incremental leaf-node insertions, and is algorithm-agnostic and can be incorporated into any online or incremental GNN training procedure that maintains per-hop embedding graphs.

C.10 Recursive Model Training

In this work, we relax node classification to linear regression as in Antonelli & Bojchevski (2025): given labeled node embeddings $X \in \mathbb{R}^{n\times d}$ and one-hot label matrix $Y \in \mathbb{R}^{n\times c}$, we seek a readout $\Theta \in \mathbb{R}^{d\times c}$ minimizing the ridge-regularized objective with unit regularization,

$$\Theta^* = \arg\min_{\Theta} \|X\Theta - Y\|_F^2 + \|\Theta\|_F^2, \tag{A.36}$$

whose closed-form solution is $\hat{\Theta} = (X^\top X + I)^{-1} X^\top Y$. Introducing a step index for the recursive formulation, we initialize $\hat{\Theta}_0 = P_0 X^\top Y$ with $P_0 = (X^\top X + I)^{-1}$.

As labeled nodes arrive sequentially, recomputing this inverse from scratch is inefficient. RLS avoids this by maintaining P_{k-1} and applying a rank-one update upon each new sample. Denoting by $x_k \in \mathbb{R}^d$ the features of the k-th arriving sample, incorporating it into the normal matrix yields the updated inverse covariance

$$P_k = \left(X_{k-1}^\top X_{k-1} + I + x_k x_k^\top\right)^{-1}, \text{ and as such, } P_{k-1} = \left(X_{k-1}^\top X_{k-1} + I\right)^{-1}, \tag{A.37}$$

where $x_k x_k^\top$ is a rank-one correction to the normal matrix. Applying the **Sherman–Morrison formula** Sherman & Morrison (1950) avoids recomputing this inverse explicitly,

$$P_k = P_{k-1} - \frac{P_{k-1}\, x_k\, x_k^\top P_{k-1}}{1 + x_k^\top P_{k-1}\, x_k}. \tag{A.38}$$

Defining the **Kalman gain** Plackett (1950)

$$K_k = \frac{P_{k-1}\, x_k}{1 + x_k^\top P_{k-1}\, x_k}, \tag{A.39}$$

and noting that $X_k^\top Y_k = X_{k-1}^\top Y_{k-1} + x_k y_k^\top$, substituting into $\hat{\Theta}_k = P_k X_k^\top Y_k$ yields

$$P_k = P_{k-1} - K_k\, x_k^\top P_{k-1}, \qquad \hat{\Theta}_k = \hat{\Theta}_{k-1} + K_k \big(y_k - x_k^\top \hat{\Theta}_{k-1}\big)^\top, \tag{A.40}$$

where K_k scales the correction by the residual $y_k - x_k^\top \hat{\Theta}_{k-1}$, the prediction error on the new sample. This corresponds to RLS with forgetting factor $\lambda = 1$ (no forgetting), equivalently the Kalman filter for a static parameter, yielding the **exact** ridge-regularized solution at every step Plackett (1950).

PC-Winter guarantees that each structural update affects exactly one node embedding, making a single RLS step sufficient to recover the exact updated solution, eliminating model retraining and matrix inversion. Inverting $X^\top X + I \in \mathbb{R}^{d\times d}$ from scratch costs $\mathcal{O}(d^3)$; the RLS update instead applies the matrix-vector products of Sherman–Morrison directly to the maintained P_{k-1}, at $\mathcal{O}(d^2)$ cost per node.

D Additional Experiments

D.1 Additional Drop-Curve Results

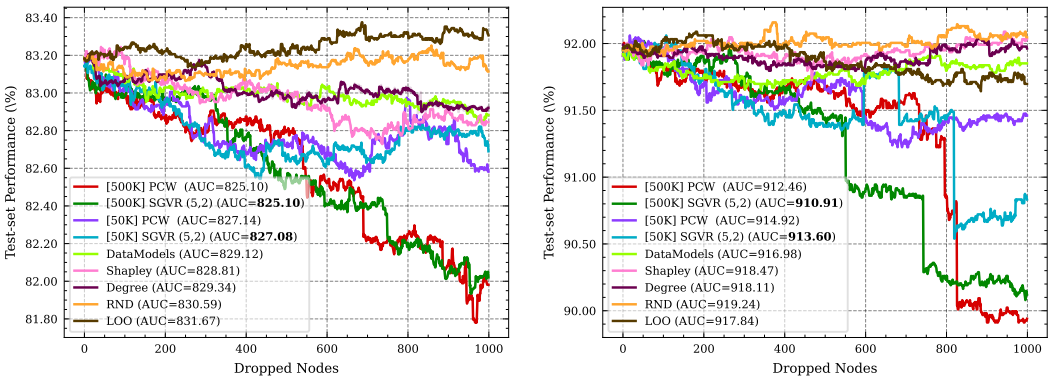


Figure 10: Most-influential-node-removal drop curves on Computers and Photo (Shchur et al., 2018), same protocol as Figure 2.

Computers and Photo follow the same ordering as Figure 2 but with much smaller margins between methods: on Computers, all budgeted methods cluster within 2 AUC points of each other, with SGVR and PC-Winter essentially tied at both budgets; on Photo, SGVR stays marginally ahead of PC-Winter at both budgets, again within a narrow margin relative to the overall AUC scale.

D.2 Variance Reduction Validation

Theorem 5 reduces pairwise variance if and only if $2\,\mathrm{Cov}_{\le L_{ij}} > \Delta_{ij}$; since this depends on the value function rather than on coordination itself (Remark 1), we test it empirically. For each node we run both estimators independently under the same settings as §3.1 (coordinated $h = (5, 2)$), and compare the resulting per-node variances directly: each node favors whichever scheme attains the strictly lower variance, or ties when both are exactly zero (which happens when too few repeated draws are available to resolve a nonzero estimate).

Coordinated sampling favors more nodes than it loses on Cora-ML, CiteSeer, PubMed, and Physics (Wilcoxon $p \le 3.4 \times 10^{-14}$), consistent with the covariance offset outweighing Δ_{ij} in aggregate. As Theorem 5 predicts, coordination helps a majority of nodes (55.6–74.4%), not all of them: whether a given node benefits depends on its own local shared-context structure, not the dataset as a whole.

Table 3: Per-node outcome of coordinated ($h = (5, 2)$) vs. uncoordinated sampling (§3.1): each node favors whichever scheme has lower variance, or ties if both are exactly zero. Wilcoxon is paired by node (H_1: coordinated is stochastically smaller). Avg. samples/path is the average repeated draws per node (coordinated / uncoordinated); mean degree is the graph's mean node degree (Table 2). Rows ordered by avg. samples/path (coordinated), descending.

Dataset	Favor $(5, 2)$	Tie	Favor $(1, 1)$	Wilcoxon p	Avg. samples/path	Mean deg.
PubMed	61.8%	≈0%	38.2%	2.6×10^{-32}	748.9 / 696.4	4.5
CiteSeer	70.9%	0%	29.2%	2.5×10^{-36}	642.0 / 547.0	3.5
Cora-ML	74.4%	≈0%	25.6%	1.0×10^{-137}	145.4 / 127.9	5.7
Physics	55.6%	1.7%	42.7%	3.4×10^{-14}	60.8 / 54.7	14.4
Photo	52.8%	6.8%	40.4%	1.2×10^{-10}	13.4 / 12.3	31.8
Computers	42.2%	11.0%	46.8%	0.993	12.0 / 11.4	36.7

On Computers and Photo the effect weakens and, for Computers, mildly reverses (42.2% vs. 46.8% favoring the baseline; Wilcoxon $p = 0.993$). Both also show a markedly higher tie rate (6.8% and 11.0%, vs. $\leq 1.7\%$ elsewhere), which requires both schemes to measure exactly zero variance, i.e. too few repeated draws to resolve a nonzero estimate. This tracks mean node degree (Table 2): Computers and Photo are considerably denser than the other four datasets (36.7 and 31.8 vs. 3.5–14.4), so a fixed budget covers fewer repeat draws per node once neighborhoods are this large.