JOINT CONFORMAL SOURCE LOCALIZATION: SOURCE SET INFERENCE AND ORDER-ADAPTIVE SET SCORING

Anonymous authors
Paper under double-blind review

ABSTRACT

Source localization models assign scores to candidate origins of an observed spreading process on a network, but these scores alone provide no guarantee of recovering the true sources. Existing conformal methods provide source recall guarantees, yet selecting candidates by individual confidence can require large sets because it does not directly account for how their combined selection meets the recall target. We propose Joint Conformal Source Localization (JCSL), a post hoc framework that uses propagation constraints to infer a distribution over source sets and prioritizes nodes by their expected contribution to normalized source recovery. Order-adaptive set scoring uses information from the original model to determine how far to extend the refined ordering to meet the recall target after calibration. Theoretically, we establish finite-sample marginal recall guarantees under exchangeability and prove that each prefix of the proposed ordering maximizes the expected recall-aware objective among all candidate sets of the same size. Across 15 spreading conditions on three real-world networks, JCSL reduces mean set size by 3.45–95.35% against the best baseline in the main comparison. The mean rate of meeting the recall target is 0.8999 at nominal confidence 0.90. Ablations show that propagation constraints and order-adaptive scoring each reduce mean prediction set size.

1 INTRODUCTION

Understanding spreading processes on networks is important in applications such as epidemic control (Ferretti et al., 2020; 2024), cyberattack investigation (Cheng et al., 2024; Li et al., 2024), and water contamination tracing (Jerez et al., 2021; Ji et al., 2022). Source localization focuses on identifying the nodes where an observed spread originated. Source detectors identify likely sources from graph structure and propagation observations (Sha et al., 2021; Wang et al., 2022). However, different source sets can produce similar propagation observations, making source locations difficult to distinguish. Moreover, predictions from existing detectors do not by themselves provide a statistical guarantee on the fraction of true sources recovered. Such a guarantee matters for subsequent investigation: missing true sources leaves gaps, while examining too many candidates increases the cost. This motivates uncertainty quantification for existing source detectors through compact sets with guaranteed recall.

Conformal prediction (CP) constructs prediction sets with finite-sample marginal coverage guarantees under exchangeability (Vovk et al., 2005; Angelopoulos & Bates, 2023). APS and RAPS calibrate candidate label sets for classification (Romano et al., 2020; Angelopoulos et al., 2021). Conformal risk control generalizes coverage guarantees to bounds on expected loss (Angelopoulos et al., 2024). Conformal structured prediction constructs interpretable representations of prediction sets with coverage guarantees (Zhang et al., 2025). For multi-source localization, setCP (Jian et al., 2026) calibrates detector scores to return node sets with prescribed source recall and confidence levels.

In multi-source localization, existing conformal methods provide recall guarantees but can require large candidate node sets. Reducing the number of candidate nodes while preserving these guarantees presents two challenges. **(i) From node confidence to source recall.** Reducing set size requires prioritizing candidates that contribute most to the recall target. A high node score indicates that a candidate is likely a source, but does not directly indicate how much including it with other candidates

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.

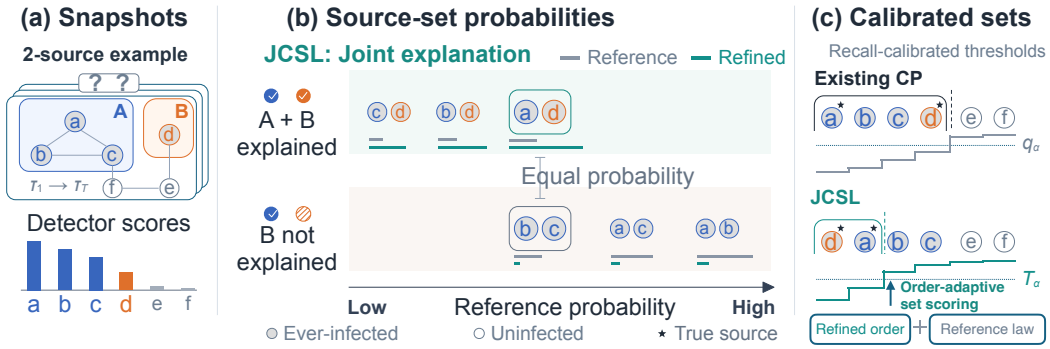


Figure 1: **From detector scores to calibrated source sets.** (a) Temporal observations in an example with two sources; question marks indicate unknown source locations. (b) The reference distribution P_0 assigns equal probability to $\{a, d\}$ and $\{b, c\}$, whereas JCSL favors $\{a, d\}$, whose candidates span observed regions A and B. Upper (Reference) and lower (Refined) bars show source set probabilities under P_0 and P_{R}. (c) Order-adaptive set scoring evaluates the refined order using P_0. At the same recall target and confidence level, the illustrated baseline returns $\{a, b, c, d\}$ and JCSL returns $\{a, d\}$. Stars in (c) mark true sources, unknown to both methods.

improves the chance of meeting that target. Joint source set inference allows these dependencies to inform candidate priorities, but must handle combinatorially many possible source sets.

(ii) From revised priorities to calibrated selection. When node priorities are revised to improve set efficiency, the resulting candidate sets change, while the detector's scores still express its original predictions for individual nodes. Using the new order alone loses score information; using the original scores alone does not incorporate the revised priorities. The challenge is to score the revised sets using detector information while preserving the recall guarantee.

We propose *Joint Conformal Source Localization* (JCSL), a post hoc conformal framework that performs **joint inference over source sets** using detector scores and propagation observations. JCSL constructs a reference distribution over source sets, then refines it by assessing whether candidate sources jointly explain the observed spread under uncertain propagation distances. From the refined distribution, we derive node priorities, maximizing expected recovery normalized by the required recovery count at each prediction set size. **Order-adaptive set scoring** calibrates how far to extend this order to meet the recall target, with *finite-sample marginal recall guarantees* under exchangeability. Figure 1 contrasts JCSL and existing conformal methods.

Our main contributions are as follows:

- **Methodologically,** we propose JCSL, a post hoc framework that jointly infers source sets from propagation observations to refine node priorities. Order-adaptive set scoring calibrates how far to extend this order to meet the recall target.
- **Theoretically,** we establish tractable exact computation of node priorities under retained constraints and optimal ordering for expected normalized recovery. With separate fitting and calibration, we obtain finite-sample marginal recall guarantees under exchangeability.
- **Experimentally,** we evaluate conformal prediction set size and inclusion rates using a shared, frozen source detector on three real-world networks across 15 simulated spreading conditions. Against the best baseline per condition in the main comparison, JCSL reduces mean prediction set size by 3.45–71.05% for random source counts and 82.19–95.35% for fixed counts, while preserving the source recall guarantee. A systematic recall/confidence grid evaluation shows reductions of 11.66–31.40%. Complementary ablations show that propagation conditioning and constraints involving multiple candidates reduce set size. Order-adaptive set scoring provides further gains across the evaluated groups of conditions.

2 PRELIMINARIES

This section establishes the source localization setting and partial-recall objective, then reviews conformal prediction. Appendix A summarizes the main notation.

2.1 NOTATION AND PROBLEM FORMULATION

Propagation observations. Let $G = (V, E)$ be the graph on which a spreading process takes place. At observation times $\tau_1 < \cdots < \tau_T$, we observe vectors of node states $\boldsymbol{z}_t \in \mathcal{Z}^{|V|}$, where t indexes the snapshot and $\mathcal{Z}$ is the state space. For example, epidemic simulations use susceptible, infected, and recovered states. Each labeled propagation instance consists of an input $x = (G, \boldsymbol{z}_{1:T})$ and a nonempty true source set $y \subseteq V$ containing the nodes that initiated the spread. Let (X, Y) denote a random observation and its true source set, and (x, y) a realization. Only x is observed at test time.

Source detector. A pretrained, frozen source detector f maps x to node scores $\boldsymbol{p}(x) = (p_v(x))_{v \in V} \in [0, 1]^{|V|}$, where a larger $p_v(x)$ indicates higher confidence that v is a source.

Problem formulation. Given the detector and labeled propagation instances, we seek a prediction set $C(x) \subseteq V$ that recovers a prescribed fraction of the true sources with a specified confidence. Define $\mathrm{rec}(C(x), y) = |C(x) \cap y|/|y|$. For a tolerated missed-source fraction $\beta \in [0, 1)$ and failure probability $\alpha \in (0, 1)$, the target is

$$\Pr\{\mathrm{rec}(C(X), Y) \geq (1 - \beta)\} \geq 1 - \alpha. \tag{1}$$

Here $1 - \beta$ is the required recall within a propagation instance, whereas $1 - \alpha$ is the minimum probability of meeting that target. The probability is taken over the calibration data, the test instance, and the auxiliary randomization used in calibration and prediction. Thus, the recall requirement is met with probability at least $1 - \alpha$ under these random draws; this is not a separate guarantee for each fixed calibration set or test input. Our objective is to minimize $\mathbb{E}[|C(X)|]$ under Equation 1.

2.2 CONFORMAL PREDICTION

We adopt split conformal prediction, which calibrates a fixed nonconformity score on a separate labeled dataset to obtain a finite-sample coverage guarantee (Vovk et al., 2005; Angelopoulos & Bates, 2023). Let $\mathcal{D}_{\mathrm{fit}}$ be a fitting split used to determine method-specific parameters and scoring rules, and let $\mathcal{D}_{\mathrm{cal}} = \{(X_i, Y_i)\}_{i=1}^{n}$ be a separate calibration set. Conditional on the fitting data and the frozen detector, we assume that the calibration pairs and the test pair (X_{n+1}, Y_{n+1}) are exchangeable. Each pair comprises a complete propagation observation and its source set.

A nonconformity score $S(x, y)$ measures how inconsistent a proposed source set y is with the predictions derived from x. Larger values indicate greater disagreement. Compute $s_i = S(X_i, Y_i)$ on $\mathcal{D}_{\mathrm{cal}}$. Let q_α be the $\lceil (n + 1)(1 - \alpha) \rceil$-th smallest value among $\{s_1, \ldots, s_n, +\infty\}$. Then

$$\Pr\{S(X_{n+1}, Y_{n+1}) \leq q_\alpha\} \geq 1 - \alpha. \tag{2}$$

Nested prediction sets. Conformal prediction also admits a nested-set formulation (Gupta et al., 2022). For a candidate set $V_x \subseteq V$ containing all true sources, an order π_x gives $C_k(x) = \pi_x[1{:}k]$, where $C_0(x) = \varnothing$ and $C_0(x) \subseteq \cdots \subseteq C_{|V_x|}(x)$. For s true sources, the required recovery count is $r_\beta(s) = \lceil (1 - \beta)s \rceil$. The position of the first set that meets the recall requirement determines a recall-based nonconformity score, whose calibrated threshold selects a prediction set (Jian et al., 2026). JCSL specifies its node order and set scoring rule within this framework.

3 JOINT CONFORMAL SOURCE LOCALIZATION (JCSL)

JCSL infers source set probabilities using detector scores and propagation observations. Recall-weighted node priorities define the order; set scores under the reference distribution P_0 determine the calibrated cutoff. Figure 2 summarizes the procedure.

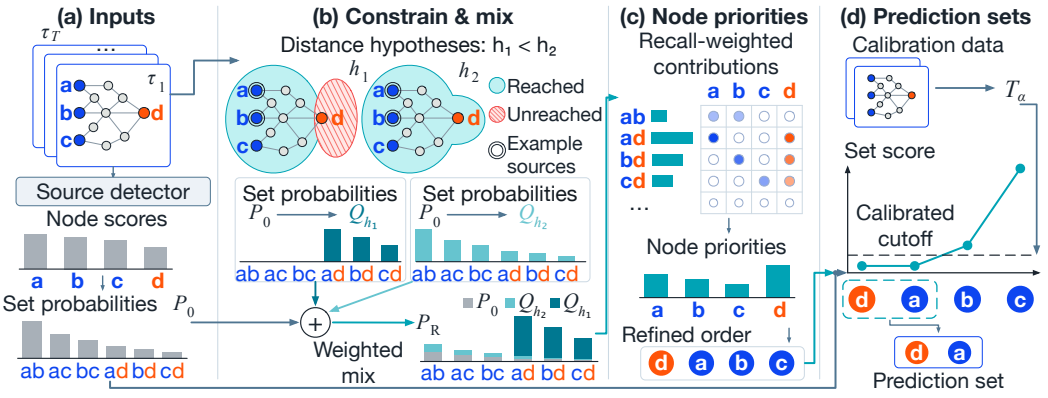


Figure 2: **Overview of JCSL.** (a) Temporal observations feed the source detector; corrected node weights and source count probabilities define the reference distribution P_0. (b) Upper graphs test the example source set $\{a, b\}$ (double rings) under propagation distance hypotheses $h_1 < h_2$. Bars show probabilities over candidate source sets, with ab denoting $\{a, b\}$. Snapshot constraints define Q_h; mixing with P_0 yields P_R. (c) Each set contributes its P_R probability divided by the required recovery count to its member nodes. Darker dots indicate larger contributions; column sums give node priorities. (d) P_0 scores prefixes along the refined order; calibration determines the cutoff.

3.1 Joint Source Set Inference

Reference distribution over source sets. A rule fixed before calibration selects the snapshot. Its affected nodes $V_x \subseteq V$ are assumed to contain all true sources, as in our fully observed SI/SIR simulations (Appendix C.1). Inference uses the induced graph $G_x = G[V_x]$.

The source count law $\eta_x(s)$ assigns probability to s sources, with support in $\{1, \ldots, |V_x|\}$. Let $\bar{p}_v(x) \in (0, 1)$ be the detector score $p_v(x)$ clipped away from 0 and 1. Under independent Bernoulli source indicators with probabilities $\bar{p}_v(x)$, source set probabilities share the factor $\prod_{v \in V_x}(1 - \bar{p}_v)$. Their relative weights are therefore products of source odds $\bar{p}_v/(1 - \bar{p}_v)$. We adjust log odds using a fitted term $g_v(x) \in \mathbb{R}$ that uses the observed states of neighboring nodes, giving positive weights

$$w_v(x) = \frac{\bar{p}_v(x)}{1 - \bar{p}_v(x)} \exp\{g_v(x)\}. \tag{3}$$

For nonempty $y \subseteq V_x$, these weights and the count law define the reference distribution

$$P_0(y \mid x) = \eta_x(|y|) \frac{\prod_{v \in y} w_v(x)}{Z^0_{x,|y|}}, \qquad Z^0_{x,s} = \sum_{\substack{B \subseteq V_x \\ |B| = s}} \prod_{v \in B} w_v(x). \tag{4}$$

$Z^0_{x,s}$ normalizes the conditional Bernoulli law for s sources (Hájek, 1964; Chen et al., 1994).

Conditioning on observed spread. We construct constrained versions of P_0 for alternative propagation hypotheses. Each hypothesis $h \in \mathcal{H}$ specifies a possible propagation distance. For each affected node u, its state determines a distance limit $\ell_{x,h}(u) \geq 0$, requiring a source within $\ell_{x,h}(u)$ graph hops. Define $N_{x,h}(u) = \{v \in V_x : \mathrm{dist}_{G_x}(v, u) \leq \ell_{x,h}(u)\}$. Graph distance is infinite between connected components. The indicator tests whether y intersects this neighborhood:

$$\psi_{u,h}(y) = \mathbf{1}\left\{ \sum_{v \in N_{x,h}(u)} y_v \geq 1 \right\}, \qquad y_v = \mathbf{1}\{v \in y\}. \tag{5}$$

Requiring these constraints to hold together links the source choices: whether one candidate is needed depends on which others are selected as sources.

Within each source count, we condition P_0 on selected propagation constraints. Normalization uses the exactly computed total weight of satisfying source sets. A constraint is retained only if the

computed elimination order stays within a fixed width budget (12 in our experiments), limiting the number of indicators handled jointly at each summation step. Let $J_{x,h}$ index the retained constraints, including singleton constraints that force their sole candidate to be a source. Their joint satisfaction event is $E^J_{x,h} = \{y \subseteq V_x : \psi_{u,h}(y) = 1 \text{ for all } u \in J_{x,h}\}$. A count s is feasible if $\eta_x(s) > 0$ and a size-s source set satisfies $E^J_{x,h}$. For feasible counts, define

$$\begin{aligned} Q^J_{x,h}(y \mid s) &= P_0(y \mid x,\ |Y| = s,\ E^J_{x,h}), \\ Q^J_{x,h}(y) &= \widetilde{\eta}_{x,h}(|y|)\, Q^J_{x,h}(y \mid |y|). \end{aligned} \tag{6}$$

Here $\widetilde{\eta}_{x,h}$ renormalizes η_x over feasible counts, preserving their relative probabilities. The second line defines $Q^J_{x,h}$ for source sets of feasible size; set $Q^J_{x,h}(y) = 0$ otherwise. This treats the count distribution as a separate modeling choice: constraints refine locations within each feasible count. Experiments in Appendices E.1–E.2 show that set efficiency is robust to replacing fitted count probabilities with uniform weights over the same possible counts, both on the main benchmark and on samples reweighted to favor fewer sources.

Combining propagation hypotheses. Several distances can be consistent with the observed spread. Let $\mathcal{F}_x \subseteq \mathcal{H}$ contain hypotheses admitting at least one feasible source count. Mixing their constrained distributions with P_0 gives the refined distribution:

$$P_{\mathrm{R}}(y \mid x) = \omega_{x,0} P_0(y \mid x) + \sum_{h \in \mathcal{F}_x} \omega_{x,h} Q^J_{x,h}(y), \qquad \omega_{x,0} + \sum_{h \in \mathcal{F}_x} \omega_{x,h} = 1. \tag{7}$$

Mixture parameters are fitted on $\mathcal{D}_{\mathrm{fit}}$ before calibration (Appendices C.2 and D.2). The weights are nonnegative with $\omega_{x,0} > 0$, preserving positive probability for every source set supported by P_0. The subscript R denotes this retention of reference support.

3.2 Recall-Weighted Node Priorities

For s sources, recall requires recovering $r_\beta(s) = \lceil (1-\beta)s \rceil$ sources. We measure progress toward this target by dividing the recovered count by $r_\beta(s)$. Each recovered source thus contributes $1/r_\beta(s)$, and its expected contribution under P_{R} defines candidate v's priority:

$$q^{\mathrm{R}}_v(x) = \mathbb{E}_{P_{\mathrm{R}}(\cdot \mid x)} \left[\frac{\mathbf{1}\{v \in Y\}}{r_\beta(|Y|)} \right]. \tag{8}$$

For any prediction set size k, the k nodes with highest priority maximize expected normalized recovery $\mathbb{E}_{P_{\mathrm{R}}(\cdot \mid x)}[|A \cap Y| / r_\beta(|Y|)]$ over all prediction sets $A \subseteq V_x$ with $|A| = k$ (Theorem 1). Sorting with fixed tie-breaking defines the order $\pi_1(x)$ and its prefixes $C_k(x) = \pi_1(x)[1{:}k]$.

Variable elimination computes these priorities without enumerating source sets. For each feasible count s, it computes $Z^J_{x,h,s}$, the total weight of size-s source sets satisfying $E^J_{x,h}$. Since $\partial \log Z^J_{x,h,s} / \partial \log w_v = Q^J_{x,h}(v \in Y \mid s)$, differentiating

$$\Phi_{x,h} = \sum_{s : \widetilde{\eta}_{x,h}(s) > 0} \frac{\widetilde{\eta}_{x,h}(s)}{r_\beta(s)} \log Z^J_{x,h,s},$$

with respect to log node weights gives all node priorities for hypothesis h in one backward pass, holding the constraints and count law fixed (Appendix B.4). By linearity of expectation, combining these priorities and those of P_0 with the weights defining P_{R} gives Equation 8.

3.3 Order-Adaptive Set Scoring

Evaluating the refined sets. The order from P_{R} incorporates propagation constraints. Using corrected node weights and count probabilities, P_0 scores each prefix by its probability of meeting recall, giving the nondecreasing curve

$$H_x(k) = P_0\{|C_k(x) \cap Y| \geq r_\beta(|Y|) \mid x\}, \qquad k = 0, \ldots, |V_x|. \tag{9}$$

The product weights of P_0 permit prefix and suffix recursions over source counts inside and outside $C_k(x)$. Combining their sums gives all $H_x(k)$ in $O(|V_x| s^2_{\max})$ operations, with $s_{\max}$ the largest represented source count (Appendix C.3). P_{R} can couple source choices across this boundary. Since H_x uses P_0 probabilities, a shared order and threshold can yield different set sizes across inputs.

Scoring the required prefix. Calibration labels identify the first prefix that meets the recall target. For a pair (x, y) with $\varnothing \neq y \subseteq V_x$ and an order π_x of V_x, the required prefix size is

$$K_\pi(x, y) = \min\{k \in \{1, \ldots, |V_x|\} : |\pi_x[1{:}k] \cap y| \geq r_\beta(|y|)\}. \tag{10}$$

The probability that P_0 assigns to meeting recall before this position gives the nonconformity score

$$S_{\mathrm{OA}}(x, y) = H_x(K_{\pi_1}(x, y) - 1). \tag{11}$$

A high score means P_0 assigns high probability to meeting recall before this order recovers enough sources in y. We score the last failing prefix, at $K_{\pi_1}(x, y) - 1$.

Calibration and prediction. With fitting complete, we score $\mathcal{D}_{\mathrm{cal}}$ and draw independent $U_i \sim \mathrm{Unif}[0, 1]$ for calibration examples. Order $(S_{\mathrm{OA}}(X_i, Y_i), U_i)$ lexicographically: first by score, then by U_i. Their randomized $(1-\alpha)$-quantile gives T_α using randomized rounding of the calibration rank (Appendix C.4). At test time, we use $b_x(k) = H_x(k-1)$ and draw one independent $U \sim \mathrm{Unif}[0, 1]$ per example, reusing it at every position:

$$\widehat{k}(x) = \max\{k \in \{1, \ldots, |V_x|\} : (b_x(k), U) \leq_{\mathrm{lex}} T_\alpha\}, \qquad C(x) = C_{\widehat{k}(x)}(x). \tag{12}$$

Here $\max \varnothing = 0$ and $C_0(x) = \varnothing$; Theorem 2 establishes the marginal recall guarantee.

4 Analysis

We analyze exact computation of normalizers and node priorities, optimal ordering for expected normalized recovery, and calibrated recall, with proofs in Appendix B.

4.1 Joint Inference

Conditioning incorporates propagation constraints while minimally changing the reference distribution. Among distributions satisfying the retained constraints and preserving the relative probabilities of feasible counts, $Q^J_{x,h}$ is closest in KL divergence to P_0 restricted to those counts (Proposition 3). The mixture then preserves all source sets supported by P_0, since $P_{\mathrm{R}}(y \mid x) \geq \omega_{x,0} P_0(y \mid x)$ (Proposition 4). Thus, a source set excluded by a distance hypothesis retains positive probability in the mixture whenever its reference probability is positive.

A fixed budget caps the elimination width and controls inference cost. Computing normalizers for all source counts is #P-hard even for unrestricted one-hop neighborhoods. Given the retained factors and an elimination order of width κ, however, JCSL computes their normalizers and node priorities exactly in $O(m_x 2^{\kappa+1} s^2_{\max})$ arithmetic operations per distance hypothesis h, where $m_x = |V_x|$ and $s_{\max}$ is the largest represented source count (Theorem 5).

4.2 Optimality of Recall-Weighted Ordering

The priorities optimize expected recovery normalized by the required recovery count.

Theorem 1 (Optimal prefixes for normalized recovery)**.** *For every $k \in \{0, \ldots, |V_x|\}$, the prefix $C_k(x)$ of the order sorted by decreasing $q^{\mathrm{R}}_v(x)$ maximizes*

$$\mathbb{E}_{P_{\mathrm{R}}(\cdot|x)}\left[\frac{|A \cap Y|}{r_\beta(|Y|)}\right] \quad \textit{over } A \subseteq V_x \textit{ with } |A| = k. \tag{13}$$

A single order therefore provides optimal prefixes at every size, forming a nested family for calibration. Among sets of the same size, each prefix also maximizes a lower bound on the probability, under $P_{\mathrm{R}}(\cdot \mid x)$, of meeting the recall target (Corollary 6).

4.3 Calibrated Recall

Recall is met exactly when the predicted prefix contains at least $K_{\pi_1}(x, y)$ nodes. By the prediction rule in Equation 12, this occurs if and only if $(H_x(K_{\pi_1}(x, y) - 1), U) \leq_{\mathrm{lex}} T_\alpha$. The equivalence holds even where H_x is constant, since H_x is nondecreasing and all prefixes share the same U.

Theorem 2 (Finite-sample partial-recall validity)**.** *Condition on the fitting data, frozen detector, and all parameters and scoring rules fixed before calibration. Assume exchangeable calibration and test pairs with $\varnothing \neq Y \subseteq V_X$ almost surely. Using the same score rule for every example and independent tie and rank randomization (Appendix C.4), JCSL satisfies*

$$\Pr\{\mathrm{rec}(C(X_{n+1}), Y_{n+1}) \geq 1 - \beta\} = 1 - \alpha. \tag{14}$$

Probability is over calibration data, the test pair, and tie/rank randomization; ranks $0, n+1$ give $T_\alpha = -\infty, +\infty$, respectively.

Recall remains valid with inaccurate source set probabilities under P_0 or P_{R}, though they affect set size. Enlarging $C(x)$ preserves the recall lower bound (Appendix D.4).

5 Experiments

5.1 Experimental Setup

Tasks and baselines. Following Jian et al. (2026), JCSL and all detector-based baselines share predictions from the same pretrained source detector on three empirical contact networks: highSchool (Salathé et al., 2010), bkFratB (Bernard et al., 1980), and SFHH (Génois & Barrat, 2018). Each network has three susceptible–infected–recovered (SIR) conditions (slow, medium, fast) with varying source counts. Six additional highSchool conditions use one, seven, or ten sources under SIR or susceptible–infected (SI) propagation. Source set inference uses the first observed snapshot, taken two simulator steps after seeding for tasks with one source and one step otherwise. Baselines are setCP-rec, setCP-pre, and setCP-min (Jian et al., 2026; 2025), ArbiTree (Cauchois et al., 2021), FavMac-complement (Lin et al., 2023) adapted to missed-source fraction, and simulation-based ADiT-DSI (Dawkins et al., 2021) for SI with one source.

Evaluation. Following Jian et al. (2026), we evaluate mean prediction set size and *inclusion rate* over 50 paired repeats per condition, with 400 test examples per repeat. Inclusion rate is the fraction of test cases attaining recall at least $1 - \beta$, with nominal target $1 - \alpha$. Each repeat provides another 7,600 examples: 1,520 for JCSL fitting and 6,080 for calibration. Main experiments use $\alpha = 0.10$, with $\beta = 0.30$ for multiple sources and $\beta = 0$ for one source. Appendix D.1 reports network and candidate sizes and data splits; Appendices D.2–D.3 detail fitting and statistical procedures.

5.2 Prediction Set Size and Recall

JCSL has the smallest mean set in all 15 main study conditions (Tables 1 and 2), reducing size by 3.45–71.05% for random counts and 82.19–95.35% for fixed counts. With shared score correction, JCSL remains 3.19–73.79% smaller than the best corrected baseline over ten paired repeats (Appendix D.6). All 76 main study comparisons favor JCSL in each of the 50 paired repeats, with negative one-sided 95% bootstrap upper bounds and Holm-adjusted $p < 0.05$. Gains grow with propagation speed on highSchool and SFHH and are smaller on bkFratB. Inclusion ranges from 0.8929 to 0.9046 (mean 0.8999), within 0.008 of the nominal 0.90. Randomized calibration permits empty outputs; enlarging fixed-count outputs to $r_\beta(s)$ nodes keeps 81.47–95.06% reductions (Appendix D.4).

The grid on highSchool SIR slow evaluates all recall/confidence combinations with $\alpha \in \{0.05, 0.10, 0.15\}$ and $\beta \in \{0.10, 0.30, 0.50, 0.70\}$. JCSL has the smallest mean set in all 12 settings, reducing size by 11.66–31.40% against the smallest baseline mean; all 60 comparisons favor JCSL in every repeat, with Holm-adjusted $p < 0.05$. Empirical inclusion rates range from 0.9449 to 0.9501, 0.8929 to 0.8988, and 0.8427 to 0.8460 for nominal targets of 0.95, 0.90, and 0.85, respectively, each within 0.008 of its target (Tables 9–10).

Restricting the outputs of ArbiTree and all three setCP variants to the observed candidate set leaves their recall unchanged. JCSL retains smaller mean sets in all 15 main conditions and 12 grid settings (Holm-adjusted $p \leq 0.0006$; Appendix D.5).

5.3 Ablation Study

Three recalibrated variants share splits over ten paired repeats per condition (Table 3; Appendix D.7). *No conditioning* uses P_0 priorities, retaining the fitted score correction, count probabilities, and

Table 1: SIR with random source counts over 50 paired repeats at $(\alpha, \beta) = (0.10, 0.30)$. Entries are means. Bold values mark the smallest set size and underlining the best baseline. Reduction is the percentage decrease from that baseline. Appendix D.4 reports full results with standard deviations.

	Set size ↓ (top) Inclusion target: 0.90 (bottom)						
Condition	setCP-rec	setCP-pre	setCP-min	ArbiTree	FavMac-comp.	JCSL	**Reduction ↑**
highSchool							
Slow	16.47 0.9014	18.73 0.9027	15.87 0.9012	741.12 0.9744	<u>14.83</u> 0.9017	**10.33** 0.8929	30.39%
Medium	26.68 0.8968	32.23 0.8971	26.26 0.8965	760.53 0.9909	<u>25.24</u> 0.8995	**10.48** 0.8996	58.50%
Fast	36.78 0.9002	47.08 0.9028	37.52 0.9010	760.96 0.9889	<u>35.45</u> 0.9024	**10.26** 0.8975	71.05%
bkFratB							
Slow	13.38 0.8963	16.09 0.9002	13.97 0.8984	56.55 0.9869	<u>12.14</u> 0.9013	**11.72** 0.9012	3.45%
Medium	19.51 0.9010	25.15 0.8978	21.59 0.9008	57.34 0.9932	<u>18.75</u> 0.9048	**17.28** 0.9046	7.83%
Fast	24.14 0.9001	32.19 0.8986	26.87 0.9004	57.75 0.9967	<u>23.59</u> 0.8988	**21.03** 0.9024	10.84%
SFHH							
Slow	14.77 0.9017	16.13 0.8989	14.08 0.8991	383.20 0.9705	<u>13.18</u> 0.8998	**11.59** 0.8992	12.05%
Medium	22.80 0.9005	25.70 0.8989	21.97 0.8982	390.19 0.9778	<u>21.39</u> 0.8996	**14.98** 0.8986	29.97%
Fast	30.45 0.9007	36.75 0.9010	30.51 0.9024	394.44 0.9863	<u>29.23</u> 0.9010	**16.76** 0.9027	42.67%

Table 2: highSchool with fixed source counts over 50 paired repeats at $\alpha = 0.10$. We use $\beta = 0$ for one source and 0.30 otherwise. ADiT-DSI applies only to SI with one source (size 24.22, inclusion 0.9864). Reporting follows Table 1.

	Set size ↓ (top) Inclusion target: 0.90 (bottom)						
Condition	setCP-rec	setCP-pre	setCP-min	ArbiTree	FavMac-comp.	JCSL	**Reduction ↑**
SI, $\lvert Y\rvert = 1$	23.30 0.9009	25.44 0.8960	<u>20.25</u> 0.8951	626.84 0.9008	20.71 0.8971	**0.94** 0.9005	95.35%
SI, $\lvert Y\rvert = 7$	<u>28.98</u> 0.8986	37.51 0.8981	30.24 0.8984	770.66 0.9999	29.01 0.8986	**4.70** 0.9015	83.78%
SI, $\lvert Y\rvert = 10$	<u>40.08</u> 0.9004	51.61 0.8978	41.43 0.8986	771.00 0.9998	40.11 0.9003	**6.87** 0.9022	82.86%
SIR, $\lvert Y\rvert = 1$	22.83 0.9038	23.63 0.8998	<u>19.17</u> 0.9019	608.59 0.9009	19.70 0.9024	**0.94** 0.8962	95.09%
SIR, $\lvert Y\rvert = 7$	<u>28.19</u> 0.9002	34.38 0.9005	29.04 0.9032	769.22 0.9999	28.21 0.9004	**4.64** 0.8982	83.54%
SIR, $\lvert Y\rvert = 10$	<u>37.78</u> 0.8992	49.48 0.8959	41.32 0.8982	770.00 1.0000	37.79 0.8994	**6.73** 0.9016	82.19%

scoring rule. Removing propagation conditioning increases mean size by 1.02–47.86% for random counts and 165.69–175.31% for fixed counts. *Unary only* retains constraints involving a single candidate. Excluding tasks with one source, mean size increases by 0.98–36.74% for random counts and 133.41–151.79% for fixed counts, supporting joint constraints. *Prefix fraction score* replaces $H_x(k)$ with $k/|V_x|$ along JCSL's order, increasing mean size by 2.53–15.81% across five groups.

Raw detector and diffusion orders yield larger mean sets and lower mean recall at matched budgets in all 15 conditions (Appendix D.8). Appendix D.7 reports separate ablations of propagation

Table 3: Mechanism ablations over ten paired repeats per condition. Size changes are paired means (positive: larger sets). Inclusion: full → ablated JCSL. Unary ablations exclude one-source tasks.

Size change (%) ↓ (top) Inclusion target: 0.90 (bottom)			
Condition group	No conditioning	Unary only	Prefix fraction score
Random highSchool	$+47.86$ $0.8974 \to 0.8996$	$+36.74$ $0.8974 \to 0.8971$	$+10.94$ $0.8974 \to 0.8968$
Random bkFratB	$+1.02$ $0.9070 \to 0.9073$	$+0.98$ $0.9070 \to 0.9074$	$+2.53$ $0.9070 \to 0.9044$
Random SFHH	$+14.65$ $0.9004 \to 0.9013$	$+13.31$ $0.9004 \to 0.9001$	$+9.54$ $0.9004 \to 0.8973$
Fixed counts, SI	$+175.31$ $0.8994 \to 0.8997$	$+151.79$ $0.8986 \to 0.9009$	$+15.36$ $0.8994 \to 0.8989$
Fixed counts, SIR	$+165.69$ $0.8962 \to 0.9018$	$+133.41$ $0.9006 \to 0.8998$	$+15.81$ $0.8962 \to 0.8959$

conditioning and score correction on the same ten repeats. Appendices E.1–E.2 show robustness to count probabilities, including reweighting toward fewer sources.

Computational cost. On highSchool (774 nodes), JCSL fitting/calibration and testing 400 examples take 8.839 s and 0.319 s for SI with one source (five workers, Intel Xeon Platinum 8358P CPU), and 665.286 s and 46.232 s for SIR with ten sources (RTX 4090, five CPU workers). Appendix D.9 (Table 18) reports single-source baseline test times from 0.009 s (FavMac) to 911.701 s (ADiT-DSI).

6 Related Work

Source localization. OJC (Zhu et al., 2017) uses graph coverage and BOSouL (Zhang et al., 2024) diffusion simulation. DIPT (Memon et al., 2026) reconstructs propagation trees. Belief propagation (Altarelli et al., 2014) and dynamic message passing (Lokhov et al., 2014) infer epidemic origins. SL-VAE (Ling et al., 2022), DDMSL (Yan et al., 2023), and SIDSL (Chen et al., 2025) model source set distributions. ADiT-DSI (Dawkins et al., 2021) constructs single-source confidence sets. JCSL turns frozen detector scores into prediction sets with calibrated partial recall guarantees.

Set prediction and graph CP. ArbiTree (Cauchois et al., 2021) uses label trees for inner and outer confidence sets. FavMac (Lin et al., 2023) controls cost violation probabilities, using missed-source fraction in our complement adaptation. Fisch et al. (2022) optimize expected recall under false positive control, whereas JCSL seeks smaller sets under recall guarantees. Graph CP diffuses or aggregates node scores (H. Zargarbashi et al., 2023; Lam & Anh, 2026; Song et al., 2024), or calibrates locally (Baghershahi et al., 2026). setCP (Jian et al., 2026) uses detector rankings. JCSL instead uses propagation constraints to infer joint source set probabilities, derives node priorities from these probabilities, and calibrates prediction sets with order-adaptive scoring.

Inference and efficiency. Information projection characterizes constrained distributions (Khanna et al., 2017). JCSL computes exact priorities by variable elimination (Peyrard et al., 2019) and cardinality recursions (Tarlow et al., 2012). Prior analyses minimize expected set size under coverage (Sadinle et al., 2019), characterize it (Dhillon et al., 2024), and establish restricted volume optimality (Gao et al., 2025). JCSL instead maximizes expected normalized recovery under P_{R} at each size.

7 Conclusion

JCSL combines joint source set inference and order-adaptive scoring for compact prediction sets. We establish tractable inference under retained constraints, optimal ordering for expected normalized recovery, and finite-sample marginal recall guarantees under exchangeability. Across 15 conditions, JCSL yields smaller sets than evaluated baselines; ablations support joint conditioning and order-adaptive scoring. Future work targets later or partial observations, broader networks, and faster inference on larger graphs.

AI USE STATEMENT

Generative AI tools were used to assist with literature search, manuscript editing, and research code implementation and debugging. They also supported the checking of mathematical arguments and proofs, provided feedback on experimental design, and assisted with the analysis and interpretation of experimental results. The authors reviewed all AI-assisted material incorporated into this work and take responsibility for the final content.

REPRODUCIBILITY STATEMENT

Appendix B provides complete proofs of the theoretical results. Appendix C details the implementation and the fitting, calibration, and prediction procedure. Appendix D specifies the datasets, simulation settings, data splits, baseline adaptations, parameter selection, and evaluation protocols. Anonymized code that reproduces JCSL's results on representative settings for all three networks is available at `https://anonymous.4open.science/r/jcsl-review`.

REFERENCES

Fabrizio Altarelli, Alfredo Braunstein, Luca Dall'Asta, Alejandro Lage-Castellanos, and Riccardo Zecchina. Bayesian inference of epidemics on networks via belief propagation. *Physical Review Letters*, 112:118701, 2014. doi: 10.1103/PhysRevLett.112.118701.

Anastasios N. Angelopoulos and Stephen Bates. Conformal prediction: A gentle introduction. *Foundations and Trends in Machine Learning*, 16(4):494–591, 2023. doi: 10.1561/2200000101.

Anastasios N. Angelopoulos, Stephen Bates, Jitendra Malik, and Michael I. Jordan. Uncertainty sets for image classifiers using conformal prediction. In *International Conference on Learning Representations*, 2021.

Anastasios N. Angelopoulos, Stephen Bates, Adam Fisch, Lihua Lei, and Tal Schuster. Conformal risk control. In *The Twelfth International Conference on Learning Representations*, 2024.

Peyman Baghershahi, Fangxin Wang, Debmalya Mandal, and Sourav Medya. GRAPHLCP: Structure-aware localized conformal prediction on graphs. *arXiv preprint arXiv:2605.08074*, 2026.

H. Russell Bernard, Peter D. Killworth, and Lee Sailer. Informant accuracy in social network data IV: A comparison of clique-level structure in behavioral and cognitive network data. *Social Networks*, 2(3):191–218, 1980. doi: 10.1016/0378-8733(79)90014-5.

Maxime Cauchois, Suyash Gupta, and John C. Duchi. Knowing what you know: Valid and validated confidence sets in multiclass and multilabel prediction. *Journal of Machine Learning Research*, 22 (81):1–42, 2021.

Hongyi Chen, Jingtao Ding, Xiaojun Liang, Yong Li, and Xiao-Ping Zhang. Structure-prior informed diffusion model for graph source localization with limited data. In *Proceedings of the 34th ACM International Conference on Information and Knowledge Management*, pp. 250–259. Association for Computing Machinery, 2025. doi: 10.1145/3746252.3761146.

Xiang-Hui Chen, Arthur P. Dempster, and Jun S. Liu. Weighted finite population sampling to maximize entropy. *Biometrika*, 81(3):457–469, 1994. doi: 10.1093/biomet/81.3.457.

Zijun Cheng, Qiujian Lv, Jinyuan Liang, Yan Wang, Degang Sun, Thomas Pasquier, and Xueyuan Han. KAIROS: Practical intrusion detection and investigation using whole-system provenance. In *2024 IEEE Symposium on Security and Privacy (SP)*, pp. 3533–3551, 2024. doi: 10.1109/SP54263.2024.00005.

Imre Csiszár. I-divergence geometry of probability distributions and minimization problems. *The Annals of Probability*, 3(1):146–158, 1975. doi: 10.1214/aop/1176996454.

Quinlan E. Dawkins, Tianxi Li, and Haifeng Xu. Diffusion source identification on networks with statistical confidence. In *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pp. 2500–2509, 2021.

Guneet S. Dhillon, George Deligiannidis, and Tom Rainforth. On the expected size of conformal prediction sets. In *Proceedings of the 27th International Conference on Artificial Intelligence and Statistics*, volume 238 of *Proceedings of Machine Learning Research*, pp. 1549–1557, 2024.

Luca Ferretti, Chris Wymant, Michelle Kendall, Lele Zhao, Anel Nurtay, Lucie Abeler-Dörner, Michael Parker, David Bonsall, and Christophe Fraser. Quantifying SARS-CoV-2 transmission suggests epidemic control with digital contact tracing. *Science*, 368(6491):eabb6936, 2020. doi: 10.1126/science.abb6936.

Luca Ferretti, Chris Wymant, James Petrie, Daphne Tsallis, Michelle Kendall, Alice Ledda, Francesco Di Lauro, Adam Fowler, Andrea Di Francia, Jasmina Panovska-Griffiths, Lucie Abeler-Dörner, Marcos Charalambides, Mark Briers, and Christophe Fraser. Digital measurement of SARS-CoV-2 transmission risk from 7 million contacts. *Nature*, 626:145–150, 2024. doi: 10.1038/s41586-023-06952-2.

Adam Fisch, Tal Schuster, Tommi Jaakkola, and Regina Barzilay. Conformal prediction sets with limited false positives. In *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pp. 6514–6532, 2022.

Chao Gao, Liren Shan, Vaidehi Srinivas, and Aravindan Vijayaraghavan. Volume optimality in conformal prediction with structured prediction sets. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 18495–18527, 2025. URL https://proceedings.mlr.press/v267/gao25k.html.

Mathieu Génois and Alain Barrat. Can co-location be used as a proxy for face-to-face contacts? *EPJ Data Science*, 7(1):11, 2018. doi: 10.1140/epjds/s13688-018-0140-1.

Chirag Gupta, Arun K. Kuchibhotla, and Aaditya Ramdas. Nested conformal prediction and quantile out-of-bag ensemble methods. *Pattern Recognition*, 127:108496, 2022.

Soroush H. Zargarbashi, Simone Antonelli, and Aleksandar Bojchevski. Conformal prediction sets for graph neural networks. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pp. 12292–12318, 2023. URL https://proceedings.mlr.press/v202/h-zargarbashi23a.html.

Jaroslav Hájek. Asymptotic theory of rejective sampling with varying probabilities from a finite population. *The Annals of Mathematical Statistics*, 35(4):1491–1523, 1964. doi: 10.1214/aoms/1177700375.

D. J. Jerez, H. A. Jensen, M. Beer, and M. Broggi. Contaminant source identification in water distribution networks: A Bayesian framework. *Mechanical Systems and Signal Processing*, 159: 107834, 2021. doi: 10.1016/j.ymssp.2021.107834.

Yiran Ji, Feifei Zheng, Jiawen Du, Yuan Huang, Weiwei Bi, Huan-Feng Duan, Dragan Savic, and Zoran Kapelan. An effective and efficient method for identification of contamination sources in water distribution systems based on manual grab-sampling. *Water Resources Research*, 58(11): e2022WR032784, 2022. doi: 10.1029/2022WR032784.

Xingchao Jian, Purui Zhang, Lan Tian, Feng Ji, Wenfei Liang, Wee Peng Tay, Bihan Wen, and Felix Krahmer. Conformal prediction for multi-source detection on a network. *arXiv preprint arXiv:2511.08867*, 2025.

Xingchao Jian, Purui Zhang, Lan Tian, Feng Ji, Wenfei Liang, Wee Peng Tay, Bihan Wen, and Felix Krahmer. Conformal prediction for multi-source detection on a network. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 36663–36670, 2026. doi: 10.1609/aaai.v40i43.40990.

Rajiv Khanna, Joydeep Ghosh, Russell Poldrack, and Oluwasanmi Koyejo. Information projection and approximate inference for structured sparse variables. In *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics*, volume 54 of *Proceedings of Machine Learning Research*, pp. 1358–1366, 2017.

Shuji Kijima, Yoshio Okamoto, and Takeaki Uno. Dominating set counting in graph classes. In *Computing and Combinatorics*, volume 6842 of *Lecture Notes in Computer Science*, pp. 13–24, 2011. doi: 10.1007/978-3-642-22685-4_2.

Phan Binh Nguyen Lam and Nguyen Thai Anh. HeAD-CP: Heterophily-aware diffused conformal prediction sets for graph neural networks. *arXiv preprint arXiv:2607.25273*, 2026.

Shaofei Li, Feng Dong, Xusheng Xiao, Haoyu Wang, Fei Shao, Jiedong Chen, Yao Guo, Xiangqun Chen, and Ding Li. NODLINK: An online system for fine-grained APT attack detection and investigation. In *Proceedings of the Network and Distributed System Security Symposium (NDSS)*, 2024. doi: 10.14722/ndss.2024.23204.

Zhen Lin, Shubhendu Trivedi, Cao Xiao, and Jimeng Sun. Fast online value-maximizing prediction sets with conformal cost control. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pp. 21182–21203, 2023. URL https://proceedings.mlr.press/v202/lin23j.html.

Chen Ling, Junji Jiang, Junxiang Wang, and Liang Zhao. Source localization of graph diffusion via variational autoencoders for graph inverse problems. In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pp. 1010–1020, 2022. doi: 10.1145/3534678.3539288.

Andrey Y. Lokhov, Marc Mézard, Hiroki Ohta, and Lenka Zdeborová. Inferring the origin of an epidemic with a dynamic message-passing algorithm. *Physical Review E*, 90:012801, 2014. doi: 10.1103/PhysRevE.90.012801.

Zeeshan Memon, Chen Ling, Ruochen Kong, Vishwanath Seshagiri, Andreas Züfle, and Liang Zhao. Deep identification of propagation trees in graph diffusion. In *Proceedings of the Thirty-Fifth International Joint Conference on Artificial Intelligence*, pp. 4660–4668, 2026. doi: 10.24963/ijcai.2026/519.

Nathalie Peyrard, Marie-Josée Cros, Simon de Givry, Alain Franc, Stéphane Robin, Régis Sabbadin, Thomas Schiex, and Matthieu Vignes. Exact or approximate inference in graphical models: Why the choice is dictated by the treewidth, and how variable elimination can be exploited. *Australian & New Zealand Journal of Statistics*, 61(2):89–133, 2019. doi: 10.1111/anzs.12257.

Yaniv Romano, Matteo Sesia, and Emmanuel J. Candès. Classification with valid and adaptive coverage. In *Advances in Neural Information Processing Systems*, volume 33, pp. 3581–3591, 2020.

Mauricio Sadinle, Jing Lei, and Larry Wasserman. Least ambiguous set-valued classifiers with bounded error levels. *Journal of the American Statistical Association*, 114(525):223–234, 2019.

Marcel Salathé, Maria Kazandjieva, Jung Woo Lee, Philip Levis, Marcus W. Feldman, and James H. Jones. A high-resolution human contact network for infectious disease transmission. *Proceedings of the National Academy of Sciences*, 107(51):22020–22025, 2010. doi: 10.1073/pnas.1009094108.

Hao Sha, Mohammad Al Hasan, and George Mohler. Source detection on networks using spatial temporal graph convolutional networks. In *2021 IEEE 8th International Conference on Data Science and Advanced Analytics (DSAA)*, pp. 1–11, 2021. doi: 10.1109/DSAA53316.2021.9564188.

Jianqing Song, Jianguo Huang, Wenyu Jiang, Baoming Zhang, Shuangjie Li, and Chongjun Wang. Similarity-navigated conformal prediction for graph neural networks. In *Advances in Neural Information Processing Systems*, volume 37, pp. 48541–48567, 2024. doi: 10.52202/079017-1538.

Daniel Tarlow, Kevin Swersky, Richard S. Zemel, Ryan P. Adams, and Brendan J. Frey. Fast exact inference for recursive cardinality models. In *Proceedings of the Twenty-Eighth Conference on Uncertainty in Artificial Intelligence*, pp. 825–834, 2012.

Vladimir Vovk, Alexander Gammerman, and Glenn Shafer. *Algorithmic Learning in a Random World*. Springer, 2005. doi: 10.1007/b106715.

Junxiang Wang, Junji Jiang, and Liang Zhao. An invertible graph diffusion neural network for source localization. In *Proceedings of the ACM Web Conference 2022*, pp. 1058–1069. Association for Computing Machinery, 2022. doi: 10.1145/3485447.3512155.

Xin Yan, Hui Fang, and Qiang He. Diffusion model for graph inverse problems: Towards effective source localization on complex networks. In *Advances in Neural Information Processing Systems*, volume 36, pp. 22326–22350, 2023. doi: 10.52202/075280-0980.

Botong Zhang, Shuo Li, and Osbert Bastani. Conformal structured prediction. In *The Thirteenth International Conference on Learning Representations*, 2025.

Zonghan Zhang, Zijian Zhang, and Zhiqian Chen. Multiple-source localization from a single-snapshot observation using graph Bayesian optimization. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pp. 22538–22546, 2024. doi: 10.1609/aaai.v38i20.30262.

Kai Zhu, Zhen Chen, and Lei Ying. Catch'em all: Locating multiple diffusion sources in networks with partial observations. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 31, pp. 1676–1682, 2017. doi: 10.1609/aaai.v31i1.10746.

A NOTATION

Table 4 summarizes the symbols used throughout the method. Uppercase (X, Y) denote random quantities and lowercase (x, y) their realizations; y is a set. Bold letters denote vectors. The labels R and OA denote retention of the reference support in the mixture and order-adaptive set scoring, respectively.

Table 4: Main notation, organized by the stages of JCSL.

Symbol	Object	Meaning
Inputs and recall objective		
$G = (V, E)$	Graph	Propagation network, with nodes V and edges E.
$\tau_t,\ T;\ \boldsymbol{z}_t, \mathcal{Z}$	Time, count; states	Time; snapshot count; state vector; node state space.
$(X, Y);\ (x, y)$	Random pair; realization	Random observation and true source set; their realized values.
$V_x,\ G_x,\ m_x$	Set, graph, count	Affected nodes in the selected snapshot, assumed to contain all sources; $G[V_x]$; $\lvert V_x\rvert$.
$f;\ p_v(x), \bar{p}_v(x)$	Detector; scalars	Frozen source detector; node score; score clipped to $(0, 1)$.
$\mathcal{D}_{\mathrm{fit}}, \mathcal{D}_{\mathrm{cal}};\ n$	Datasets; count	Fitting and calibration splits; calibration sample size.
$\alpha,\ \beta$	Tolerances	Failure probability and missed-source fraction.
$\mathrm{rec}(A, y),\ r_\beta(s)$	Fraction; count	$\lvert A \cap y\rvert/\lvert y\rvert$; sources required: $\lceil (1-\beta)s \rceil$.
Source set inference		
$g_v(x),\ w_v(x)$	Scalars	Fitted log odds correction; positive node weight.
$d_v(x),\ a$	Fraction; scalar	Fraction of affected neighbors; fitted coefficient.
$s,\ \eta_x(s),\ \widetilde{\eta}_{x,h}(s)$	Count; masses	Source count; count probability; its renormalization over counts feasible under h.
$h,\ \mathcal{H},\ \mathcal{F}_x$	Hypothesis; sets	Propagation distance hypothesis; family; feasible subset.
$\ell_{x,h}(u),\ N_{x,h}(u)$	Limit; node set	Allowed graph distance and possible sources within it.
$y_v,\ \psi_{u,h}(y)$	Indicators	Whether $v \in y$; whether y covers node u under h.
$J_{x,h},\ E^J_{x,h}$	Index set; event	Retained constraints and their joint satisfaction event.
$P_0(\cdot \mid x)$	Distribution	Reference source set law from w_v and η_x.
$Q^J_{x,h}(\cdot \mid s),\ Q^J_{x,h}$	Distributions	Constrained law at count s, then mixed over feasible counts (Q_h in Figure 2).
$\omega_{x,0}, \omega_{x,h};\ P_{\mathrm{R}}$	Weights; law	Mixture weights; refined source set distribution.
$Z^0_{x,s},\ Z^J_{x,h,s}$	Normalizers	Total size-s set weight, before and after constraints.
$\lambda_0, \lambda_h;\ \epsilon_h$	Weights; fraction	Initial mixture weights; fraction of component h's weight allocated to P_0.
$\rho,\ \Lambda_x$	Masses	Total unconstrained mass; retained mixture mass.
$\kappa,\ s_{\max}$	Width; count bound	Induced elimination width after fixing forced sources; largest represented source count.
$\theta_v,\ \Phi_{x,h}$	Scalar; function	$\log w_v(x)$; recall-weighted sum of log normalizers used to obtain node priorities for h.
Node ordering and conformal prediction		
$q^{\mathrm{R}}_v(x)$	Priority	$\mathbb{E}_{P_{\mathrm{R}}(\cdot\mid x)}[\mathbf{1}\{v \in Y\}/r_\beta(\lvert Y\rvert)]$.
$\pi_x,\ \pi_1(x)$	Node orders	A generic order and the order of JCSL's priorities.
$A,\ C_k(x),\ C(x)$	Prediction sets	A set under comparison, a size-k prefix, and the final output.
$H_x(k),\ b_x(k)$	Score maps	Probability under $P_0(\cdot \mid x)$ that $C_k(x)$ meets the recall target; $b_x(k) = H_x(k-1)$.
$e_j(A)$	Weight sum	Total product weight of size-j subsets of A.
$K_\pi(x, y)$	Position	First prefix meeting the recall target for y under π_x.
$S,\ S_{\mathrm{OA}}$	Scores	Generic nonconformity score; $H_x(K_{\pi_1}(x, y) - 1)$.
$U_i,\ U$	Random scalars	Independent uniform tie breakers across examples; the test draw U is reused at every prefix.
$\widetilde{S}_i$	Ordered pair	$(S_{\mathrm{OA}}(X_i, Y_i), U_i)$, compared lexicographically.
$q_\alpha,\ T_\alpha$	Thresholds	Scalar split-CP threshold (possibly $+\infty$); JCSL's lexicographic threshold (possibly $\pm\infty$).
$R_\alpha,\ \widehat{k}(x)$	Rank; position	Randomized calibration rank; final selected prefix size.

B Theoretical Results and Proofs

B.1 Conditioning at Fixed Count Probabilities

In derivations for one mixture component, we suppress the index h. For a fixed feasible distance hypothesis, let $\mathcal{A}_x = \{s : \eta_x(s) > 0,\ P_0(E_x^J \mid x, |Y| = s) > 0\}$. For $s \in \mathcal{A}_x$, the conditional distribution is

$$Q_x^J(y \mid s) = \frac{\mathbf{1}\{|y| = s\} E_x^J(y) \prod_{v \in y} w_v(x)}{Z_{x,s}^J}, \qquad Z_{x,s}^J = \sum_{\substack{B \subseteq V_x \\ |B| = s}} E_x^J(B) \prod_{v \in B} w_v(x), \tag{15}$$

where $E_x^J(y) := \mathbf{1}\{y \in E_x^J\}$ is the event indicator. The denominator sums the weights of all admissible size-s sets. Write $P_0^{\mathcal{A}} = P_0(\cdot \mid x, |Y| \in \mathcal{A}_x)$. Its count marginal is $\widetilde{\eta}_x(s) = \eta_x(s) / \sum_{t \in \mathcal{A}_x} \eta_x(t)$ for $s \in \mathcal{A}_x$ and zero otherwise. Define

$$\mathcal{R}_x = \{R : R(E_x^J) = 1,\ R(|Y| = s) = \widetilde{\eta}_x(s) \text{ for every } s\}.$$

The following result specializes the information-projection geometry of Csiszár (1975) to these support and count constraints.

Proposition 3 (Information projection with fixed feasible-count probabilities)**.** *The distribution Q_x^J is the unique minimizer of $\mathrm{KL}(R \| P_0^{\mathcal{A}})$ over $R \in \mathcal{R}_x$. For every $R \in \mathcal{R}_x$,*

$$\mathrm{KL}(R \| P_0^{\mathcal{A}}) = \mathrm{KL}(R \| Q_x^J) + \mathrm{KL}(Q_x^J \| P_0^{\mathcal{A}}), \tag{16}$$

with $\mathrm{KL}(Q_x^J \| P_0^{\mathcal{A}}) = \sum_{s \in \mathcal{A}_x} \widetilde{\eta}_x(s)[-\log P_0(E_x^J \mid x, |Y| = s)]$.

Proof. By feasibility of the chosen distance hypothesis, $\mathcal{A}_x$ is nonempty. Positive node weights imply that $P_0^{\mathcal{A}}$ is positive on every source set with count in $\mathcal{A}_x$. For $R \in \mathcal{R}_x$, the chain rule for KL divergence and the common count marginal yield

$$\begin{aligned} \mathrm{KL}(R \| P_0^{\mathcal{A}}) &= \sum_{s \in \mathcal{A}_x} \widetilde{\eta}_x(s)\, \mathrm{KL}\big(R(\cdot \mid s) \| P_0(\cdot \mid x, |Y| = s)\big) \\ &= \sum_{s \in \mathcal{A}_x} \widetilde{\eta}_x(s) \left[\mathrm{KL}\big(R(\cdot \mid s) \| Q_x^J(\cdot \mid s)\big) - \log P_0(E_x^J \mid x, |Y| = s)\right]. \end{aligned}$$

The second equality uses $R(E_x^J) = 1$. Its logarithmic term is constant over R, and the remaining KL terms are nonnegative. They vanish exactly when $R(\cdot \mid s) = Q_x^J(\cdot \mid s)$ for every $s \in \mathcal{A}_x$. Thus, Q_x^J is the unique minimizer. Subtracting the value at Q_x^J gives Equation 16.

B.2 Mixture Support

Proposition 4 (Mixture support)**.** *With normalized mixture components and weights as in Equation 7, P_{R} is a probability law, $P_{\mathrm{R}}(y \mid x) \geq \omega_{x,0} P_0(y \mid x)$, and P_{R} has exactly the support of P_0.*

Proof. Each $Q_{x,h}^J$ mixes normalized conditional distributions $Q_{x,h}^J(\cdot \mid s)$ with count probabilities $\widetilde{\eta}_{x,h}(s)$. The nonnegative mixture weights sum to one, so P_{R} is normalized. Its unconstrained term gives $P_{\mathrm{R}}(y \mid x) \geq \omega_{x,0} P_0(y \mid x)$. Conversely, $P_0(y \mid x) = 0$ implies $Q_{x,h}^J(y) = 0$ in every feasible mixture component because conditioning retains only source counts and sets already supported by P_0. Since $\omega_{x,0} > 0$, the two supports coincide.

B.3 Complexity of Constrained Inference

Theorem 5 (Complexity of constrained inference)**.** *Computing the full vector of normalizers over source counts for unrestricted cover constraints with graph-distance limit one is #P-hard even with unit node weights. For a fixed input x and a feasible distance hypothesis, let $m_x = |V_x|$ and let $s_{\max}$ be the largest represented count. Given at most m_x retained cover factors and an elimination order of induced width κ after substituting fixed source indicators, binary variable elimination computes all normalizers through degree $s_{\max}$ and the recall-weighted node priorities in $O(m_x 2^{\kappa+1} s_{\max}^2)$ arithmetic operations. Each intermediate factor stores at most $2^{\kappa+1}(s_{\max} + 1)$ coefficients.*

Proof. For hardness, take an arbitrary nonempty undirected graph $G = (V, E)$ and use all its nodes as candidates, with unit node weights. Assign every node the cover constraint with distance limit one,

$$\psi_u(y) = \mathbf{1}\left\{ \sum_{v \in N_G[u]} y_v \geq 1 \right\},$$

where $N_G[u]$ is the closed neighborhood of u. A source set satisfies all these constraints exactly when it is a dominating set of G. Consequently, if Z_s denotes the normalizer for these constraints at count s, then $\sum_{s=0}^{|V|} Z_s$ counts the dominating sets of G. This counting problem is #P-complete even on split graphs (Kijima et al., 2011); computing the full vector (Z_s) is therefore #P-hard.

For the upper bound, first substitute all fixed source indicators. Introduce a formal variable ζ that records source count: selecting a remaining node contributes $w_v\zeta$, while not selecting it contributes 1. Every factor entry is a polynomial truncated after degree $s_{\max}$. Truncation is exact for the requested coefficients because all degrees are nonnegative. Fixed sources contribute a multiplicative weight and a degree shift.

There are $O(m_x)$ initial factors (one unary factor per node and at most one retained cover factor per observed node) and at most one generated factor per eliminated variable. Each initial or generated factor is consumed once in a bucket or in the final scalar multiplication, so the total number of pairwise factor multiplications is $O(m_x)$. Given the supplied elimination order, induced width κ bounds each bucket's union scope by $\kappa + 1$ binary variables. Each multiplication over this scope uses at most $2^{\kappa+1}$ polynomial convolutions, each costing $O(s_{\max}^2)$ arithmetic operations. Summing out a binary indicator costs $O(2^{\kappa+1}s_{\max})$. The total cost is therefore $O(m_x 2^{\kappa+1} s_{\max}^2)$ arithmetic operations. The final coefficient of ζ^s is $Z_{x,s}^J$.

A factor with at most $\kappa + 1$ binary variables stores at most $2^{\kappa+1}(s_{\max} + 1)$ coefficients. This is the storage bound for each factor; retaining all messages uses $O(m_x 2^{\kappa+1} s_{\max})$ coefficients before the additional storage required for reverse-mode differentiation. Differentiating Φ_x in reverse mode gives all recall-weighted priorities at the same arithmetic complexity (Appendix B.4).

B.4 Derivative Identity and Count Conservation

For the component Q_x^J, hold E_x^J and η_x fixed. Then $\mathcal{A}_x$ and $\widetilde{\eta}_x$ remain unchanged as the positive node weights vary. Put $\theta_v = \log w_v$. Direct differentiation gives

$$\frac{\partial \log Z_{x,s}^J}{\partial \theta_v} = \frac{\sum_{y:\,|y|=s} E_x^J(y)\mathbf{1}\{v \in y\} \prod_{u \in y} w_u(x)}{Z_{x,s}^J} = Q_x^J(v \in Y \mid s).$$

Thus, the gradient of

$$\Phi_x(\boldsymbol{\theta}) = \sum_{s \in \mathcal{A}_x} \frac{\widetilde{\eta}_x(s)}{r_\beta(s)} \log Z_{x,s}^J(\exp \boldsymbol{\theta})$$

has coordinates

$$q_v^J(x) = \sum_{s \in \mathcal{A}_x} \widetilde{\eta}_x(s) \frac{Q_x^J(v \in Y \mid s)}{r_\beta(s)}. \tag{17}$$

Summing over $v \in V_x$ and using $\sum_{v \in V_x} \mathbf{1}\{v \in Y\} = s$ at each fixed count yields

$$\sum_{v \in V_x} q_v^J(x) = \sum_{s \in \mathcal{A}_x} \widetilde{\eta}_x(s) \frac{s}{r_\beta(s)}. \tag{18}$$

This identity also includes fixed source indicators: a node fixed to one has inclusion probability one at every feasible count. The priorities under P_{R} are weighted sums of the priorities under its mixture components.

B.5 Proof of Theorem 1

For any distribution Q on nonempty source sets in V_x, define $q_v = \mathbb{E}_Q[\mathbf{1}\{v \in Y\}/r_\beta(|Y|)]$. JCSL uses $Q = P_{\mathrm{R}}(\cdot \mid x)$. For every $A \subseteq V_x$,

$$\mathbb{E}_Q \frac{|A \cap Y|}{r_\beta(|Y|)} = \sum_{v \in A} q_v.$$

At fixed size, exchanging an included node of lower q_v with an excluded node of higher q_v increases this sum. Hence the descending order of q_v maximizes the objective at every size. Ties may be resolved by any fixed rule.

B.6 Recall Probability under a Source Set Distribution

For any distribution Q on nonempty source sets in V_x, define $q_v = \mathbb{E}_Q[\mathbf{1}\{v \in Y\}/r_\beta(|Y|)]$ and $\mathcal{S}_Q = \{s : Q(|Y| = s) > 0\}$. JCSL uses $Q = P_{\mathrm{R}}(\cdot \mid x)$. Define

$$\delta_\beta = \max_{s \in \mathcal{S}_Q} \frac{r_\beta(s) - 1}{r_\beta(s)}, \qquad M_\beta = \max_{s \in \mathcal{S}_Q} \frac{s}{r_\beta(s)}. \tag{19}$$

δ_β bounds normalized recovery when the recall target is missed, and M_β bounds it overall. These bounds connect expected normalized recovery to the probability of meeting recall. The finite nonempty support ensures $0 \le \delta_\beta < 1 \le M_\beta$.

Corollary 6 (Recall probability bound under Q)**.** *Every prediction set $A \subseteq V_x$ satisfies*

$$Q\{\mathrm{rec}(A, Y) \ge 1 - \beta\} \ge \left[\frac{\sum_{v \in A} q_v - \delta_\beta}{M_\beta - \delta_\beta}\right]_+. \tag{20}$$

Here $[t]_+ = \max\{t, 0\}$. At every size, the top-q_v prefix maximizes this lower bound. If $Q(|Y| = 1) = 1$, then $\delta_\beta = 0$ and $M_\beta = 1$, so Equation 20 holds with equality.

Proof. Set $R_A = |A \cap Y|/r_\beta(|Y|)$ and $B_A = \mathbf{1}\{|A \cap Y| \ge r_\beta(|Y|)\}$. On failure, $R_A \le \delta_\beta$; on success, $R_A \le M_\beta$. Therefore

$$\sum_{v \in A} q_v = \mathbb{E}_Q R_A \le \delta_\beta(1 - \mathbb{E}_Q B_A) + M_\beta \mathbb{E}_Q B_A.$$

Rearranging and using nonnegativity of the success probability proves Equation 20. The denominator is positive and independent of A, so Theorem 1 maximizes this bound at every set size.

B.7 Proof of Theorem 2

For fixed fitted rules and a pair (x, y), put $K = K_{\pi_1}(x, y)$. The assumption $\varnothing \ne y \subseteq V_x$ ensures $1 \le K \le |V_x|$. Monotonicity of b_x and the shared tie variable give

$$\widehat{k}(x) \ge K \quad \Longleftrightarrow \quad (b_x(K), U) \le_{\mathrm{lex}} T_\alpha. \tag{21}$$

The equivalence holds on flat intervals of b_x and at both infinite thresholds.

Write $\widetilde{S}_i = (S_{\mathrm{OA}}(X_i, Y_i), U_i)$ for $i = 1, \ldots, n+1$. Independent continuous tie variables preserve exchangeability and make these pairs distinct almost surely. The rank of $\widetilde{S}_{n+1}$ among all $n+1$ pairs is consequently uniform on $\{1, \ldots, n+1\}$. Let R_α be the independently randomized calibration rank. For any fixed $r \in \{1, \ldots, n\}$, acceptance at or below the r-th calibration order statistic is equivalent to the rank of $\widetilde{S}_{n+1}$ among all $n+1$ pairs being at most r. Its probability is $r/(n+1)$. The same formula holds at $r = 0$ and $r = n+1$: thresholds $-\infty$ and $+\infty$ accept no and all score pairs, respectively. It follows that

$$\Pr\{\widetilde{S}_{n+1} \le_{\mathrm{lex}} T_\alpha\} = \frac{\mathbb{E}[R_\alpha]}{n+1} = 1 - \alpha.$$

Finally, $\widehat{k} \ge K$ is precisely the event of meeting the recall target. The same argument with the deterministic ceiling rank and the same continuous tie augmentation gives coverage $\lceil (n+1)(1-\alpha) \rceil/(n+1) \ge 1 - \alpha$.

C Method Implementation Details

C.1 Constraints from Observations

The benchmark implementation extracts candidates and constraints from the first observed snapshot. Write $\boldsymbol{z} = (z_v)_{v \in V}$ for this snapshot, with $z_v = 0, 1, 2$ encoding susceptible, infected, and recovered

nodes, respectively, and $V_x = \{v : z_v \neq 0\}$. In the fully observed SI/SIR simulations, affected nodes never return to the susceptible state. Thus all sources and all nodes on transmission paths up to this snapshot lie in V_x, so $G_x = G[V_x]$ retains those paths. Source set inference uses distances on G_x; the detector separately processes the full input sequence.

In the multi-source routine, an infected node has distance allowance h, whereas a recovered node has allowance $h-1$. In the synchronous simulator, a newly infected node can recover only in a later step, motivating this tighter allowance for recovered nodes. Accordingly,

$$N_{x,h}(u) = \begin{cases} \{v \in V_x : \mathrm{dist}_{G_x}(v,u) \leq h\}, & z_u = 1, \\ \{v \in V_x : \mathrm{dist}_{G_x}(v,u) \leq h-1\}, & z_u = 2. \end{cases}$$

At $h = 1$, each recovered node therefore yields a singleton constraint that fixes it as a source within that mixture component. These fixed indicators are substituted before selecting the remaining factors. Incompatible counts are excluded; if no supported count remains feasible, the constrained mixture component is omitted as in Appendix C.2. The routine for one source uses distance allowance h for all nodes in V_x and evaluates which individual candidates can reach the whole observed region. These routines instantiate the neighborhoods determined by observations in Equation 5.

C.2 Parameterization of Mixture Weights

The implementation assigns weights $\lambda_h \geq 0$ to the represented distances and $\lambda_0 \geq 0$ to the unconstrained mixture component P_0, with $\lambda_0 + \sum_{h \in \mathcal{H}} \lambda_h = 1$. For each distance, $\epsilon_h \in [0,1]$ allocates a share of its weight to P_0. The total unconstrained mass is

$$\rho = \lambda_0 + \sum_{h \in \mathcal{H}} \lambda_h \epsilon_h > 0. \tag{22}$$

The positive pseudocount for the unconstrained category (Appendix D.2) ensures $\lambda_0 > 0$ and hence $\rho > 0$. Let $Z^J_{x,h,s}$ be the normalizer at count s for hypothesis h. The component for h is retained when $\sum_{s: Z^J_{x,h,s} > 0} \eta_x(s) > 0$, defining $\mathcal{F}_x$. Set

$$\Lambda_x = \rho + \sum_{h \in \mathcal{F}_x} \lambda_h (1 - \epsilon_h), \qquad \omega_{x,0} = \frac{\rho}{\Lambda_x}, \qquad \omega_{x,h} = \frac{\lambda_h (1-\epsilon_h)}{\Lambda_x} \quad (h \in \mathcal{F}_x).$$

Infeasible constrained mixture components are omitted, while their contribution to ρ is retained. Since $0 < \rho \leq \Lambda_x \leq 1$, the weights satisfy Equation 7; Appendix D.2 details the fitting of $\lambda_0, \lambda_h, \epsilon_h$.

C.3 Evaluating the Probability of Meeting the Recall Target

For $A \subseteq V_x$, define the elementary symmetric polynomial $e_j(A) = \sum_{B \subseteq A,\, |B| = j} \prod_{v \in B} w_v(x)$, with $e_0(A) = 1$ and $e_j(A) = 0$ for $j < 0$ or $j > |A|$. Under P_0, conditional on s sources, exactly j lie in $C_k(x)$ with probability $e_j(C_k(x)) e_{s-j}(V_x \setminus C_k(x)) / e_s(V_x)$. Averaging over source counts s and summing over successful recovery counts j gives

$$H_x(k) = \sum_s \eta_x(s) \frac{\sum_{j=r_\beta(s)}^{s} e_j(C_k(x)) e_{s-j}(V_x \setminus C_k(x))}{e_s(V_x)}. \tag{23}$$

The sum ranges over counts with positive $\eta_x(s)$. For $v \notin A$, the recurrence $e_j(A \cup \{v\}) = e_j(A) + w_v(x) e_{j-1}(A)$ computes prefix and suffix tables along $\pi_1(x)$. These tables yield every value of H_x using the corrected node weights in each refined prefix. We evaluate these sums in `longdouble` arithmetic, convert H_x to `float64`, and clip it to $[0,1]$. A cumulative maximum enforces monotonicity after rounding. Calibration scores and test inversion use curves computed by this same numerical rule, preserving the equivalence in Equation 21.

C.4 Randomized Calibration Threshold

Calibration example i receives $U_i \sim \mathrm{Unif}[0,1]$; each test example receives an independent U from the same law. Calibration pairs are ordered first by $S_{\mathrm{OA}}(X_i, Y_i)$ and then by U_i. For n calibration

examples, randomize between the floor and ceiling of $(n+1)(1-\alpha)$, choosing the upper rank with probability equal to the fractional part of this quantity. This rank randomization is independent of the data and tie variables. The selected order statistic of the calibration score pairs is T_α. If the selected rank is 0, set $T_\alpha = -\infty$; if it is $n+1$, set $T_\alpha = +\infty$. For each test example, draw one tie variable and reuse it at all positions in Equation 12. This keeps the score pairs ordered along the nested sets, including where H_x is constant.

For a task with $\eta_x(1) = 1$, positive node weights give $H_x(0) = 0$ and $H_x(k) > 0$ for $k \geq 1$. A calibration score is therefore zero exactly when $K_{\pi_1} = 1$. If the selected rank lies among these zero scores, then $T_\alpha = (0, u_*)$: Equation 12 returns the singleton containing the first node when $U \leq u_*$ and the empty set otherwise.

C.5 Procedure

Given the frozen detector, $\mathcal{D}_{\mathrm{fit}}$, $\mathcal{D}_{\mathrm{cal}}$, and tolerances α, β, JCSL proceeds as follows. Steps 2–5 apply the same fitted mapping to every calibration and test input.

Step	Operation
1. Fit	Using $\mathcal{D}_{\mathrm{fit}}$, select the score correction and estimate the source count probabilities and parameters of the distance mixture (Appendix D.2). Fix these choices, the snapshot rule, and the width budget before calibration.
2. Reference	For input x, extract V_x and G_x from the selected snapshot. Compute $w_v(x)$ and form η_x by restricting and renormalizing the fitted count probabilities over $1 \leq s \leq \lvert V_x \rvert$. These weights and count probabilities define the reference distribution P_0 by Equations 3–4.
3. Constraints	Construct the neighborhoods and factors for each h (Equation 5). For multiple sources, substitute fixed source indicators and greedily retain constraints within the width budget. For a fixed single source, evaluate candidate feasibility directly (Appendix C.1).
4. Priorities	Compute the normalizers for feasible counts and node contributions for each retained hypothesis, using exact variable elimination and log-weight derivatives for multiple sources (Appendix B.4). Omit hypotheses with no feasible supported count. Combine the remaining hypotheses' node priorities with those of P_0 using Appendix C.2 to obtain $q_v^{\mathrm{R}}(x)$ (Equation 8).
5. Score curve	Sort candidates by decreasing $q_v^{\mathrm{R}}(x)$ with fixed tie-breaking to obtain $\pi_1(x)$. Compute all $H_x(k)$ under P_0 along this order using Equation 23.
6. Calibrate	Apply steps 2–5 to each calibration input. Use its labels to compute K_{π_1} and S_{OA} (Equations 10–11), and draw an independent U_i. Obtain T_α from the lexicographically ordered score pairs using the independent randomized rank in Appendix C.4.
7. Predict	Apply steps 2–5 to a test input x. Draw one independent U and reuse it at every prefix. Return $C(x)$ by Equation 12, including $C_0(x) = \varnothing$ when no position is accepted.

D Experimental Details and Additional Results

D.1 Data, Splits, and Comparators

We use the processed network versions described by Sha et al. (2021), treated as unweighted, undirected graphs with each edge counted once. Table 5 reports graph and candidate sizes. Source nodes are drawn uniformly without replacement. For experiments with random source counts, we sample source counts uniformly from 1 to 15 and retain simulations with more than 16 recorded iterations. Each simulation draws its recovery probability γ uniformly from $[0.1, 0.4]$. Following Jian et al. (2026), slow, medium, and fast regimes draw the reproduction number R_0 uniformly from $[1, 15]$, $[11, 25]$, and $[21, 35]$, respectively, and set the per-edge transmission probability to $R_0\gamma/\lambda_{\max}$, where $\lambda_{\max}$ is the largest eigenvalue of the adjacency matrix. highSchool experiments with fixed counts use one, seven, or ten sources. Transmission probability is 0.25; recovery probabilities are 0 (SI) and 0.15 (SIR).

Detector-based methods use the released SD-STGCN checkpoints of Jian et al. (2026), fixed throughout evaluation. Each of the 50 repeats per condition uses a separately simulated dataset of 8,000 examples with a distinct seed. Within each repeat, we uniformly sample 7,600 examples without replacement for fitting and calibration, leaving 400 for testing. Detector-based methods share that repeat's data and this split. JCSL randomly partitions the 7,600 examples into 1,520 for fitting and

Table 5: Sizes of the processed benchmark graphs and candidate sets from the first snapshot. Candidate statistics pool 20,000 test examples per condition (50 repeats, 400 examples each); P10 and P90 are the 10th and 90th percentiles. Each undirected edge is counted once.

Network	$\lvert V\rvert$	$\lvert E\rvert$	Setting	Mean $\lvert V_x\rvert$	P10	P90
Random source counts (1–15), SIR						
highSchool	774	7,992	Slow	20.89	5	41
			Medium	35.50	8	68
			Fast	49.47	11	96
bkFratB	58	967	Slow	15.79	4	29
			Medium	23.75	7	40
			Fast	29.58	10	46
SFHH	403	9,565	Slow	19.15	5	37
			Medium	31.29	7	60
			Fast	42.68	9	81
Fixed source counts						
highSchool	774	7,992	SI, 1 source	37.37	8	66
			SI, 7 sources	42.00	32	52
			SI, 10 sources	59.27	48	71
			SIR, 1 source	36.85	8	65
			SIR, 7 sources	42.13	32	52
			SIR, 10 sources	59.37	48	71

6,080 for calibration. ArbiTree follows the adaptation for partial recall of Jian et al. (2026; 2025), using 1,000 examples for tree fitting and 6,600 for calibration. setCP and FavMac use all 7,600 for calibration, so JCSL calibrates on 1,520 fewer examples than these baselines. ADiT-DSI follows the setup of Jian et al. (2026; 2025) for SI with one source, with $m_l = m_p = 20$ simulated trajectories for discrepancy estimation and Monte Carlo p-value evaluation, respectively. FavMac-complement applies FavMac's violation control to excluded nodes, using missed-source fraction as cost and the number excluded as value. As shown by Jian et al. (2025), setCP-min coincides with conformal risk control (Angelopoulos et al., 2024) for the same detector-threshold family and recall-event loss.

The shared detector, SD-STGCN (Sha et al., 2021), processes 16 consecutive snapshots of node states and the aggregated contact graph. In every dataset, each true source set is nonempty and lies within the candidate set, so no example is removed.

D.2 Fitting the Reference Distribution and Mixture Weights

We instantiate Equation 3 with $g_v(x) = a\, d_v(x)$, where $d_v(x)$ is the fraction of v's neighbors in G that are infected or recovered in the first observed snapshot. All three graphs have positive node degrees. A single coefficient a, shared across nodes and inputs, is selected separately for each condition and repeat from $\{0, 2, 4, 8, 16, 32\}$ using two equal halves of the fitting split: a temporary conformal threshold is estimated on one half, and the coefficient minimizing mean prediction set size on the other half is selected, with ties favoring the smaller coefficient. Selection uses P_0; the coefficient is then fixed for JCSL. Detector scores are clipped to $[10^{-8}, 1 - 10^{-8}]$. For each input, we subtract the largest log weight before exponentiating; this common scaling cancels when normalizing within each source count. We differentiate with respect to the centered log weights, holding the centering constant fixed, and add forced-source contributions using $r_\beta(s)$ at the total count s. For the reported targets, inference and evaluation share the integer recovery requirement $\lceil (1-\beta)s - 10^{-12} \rceil$.

Source count probabilities are empirical frequencies on the fitting split, restricted to $s \le |V_x|$ and renormalized for each input; the restricted counts have positive total frequency for every input in our experiments. For fitting distance weights, $E_h^{\mathrm{fit}}(x, y)$ requires each infected node to lie within distance h of y and each recovered node within $h - 1$, measured in G_x. This criterion uses all constraints before selection. In tasks with one source, it is stricter at recovered nodes than the common-radius criterion used for inference. Satisfying this fitting criterion therefore satisfies the retained inference constraints. Define

$$h_{\min}(x, y) = \min\{h \in \{1, 2, 3, 4\} : E_h^{\mathrm{fit}}(x, y) = 1\}. \tag{24}$$

Here $\min \varnothing = +\infty$ denotes overflow, assigned to the unconstrained category with weight λ_0. Finite $h_{\min}$ values observed in the fitting split form the active distance set. Adding one pseudocount to each active distance and the unconstrained category, then normalizing, gives λ_h and λ_0 in Appendix C.2.

Among $n_{\text{fit}} = 1{,}520$ independent fitting simulations, let f_h count failures of E_h^{fit}, equivalently overflow or $h_{\min}(x, y) > h$. For each fixed h, we set ϵ_h to the pointwise one-sided 99% Clopper–Pearson upper bound on the failure probability: the 0.99 quantile of $\text{Beta}(f_h + 1, n_{\text{fit}} - f_h)$ when $f_h < n_{\text{fit}}$, and one otherwise, with a numerical floor of 10^{-8}. After substituting forced source indicators, we rank the remaining unsatisfied constraints in decreasing order of the negative logarithm of their satisfaction probability under a mixture of product-weight distributions conditional on the remaining source count. The count mixture weights inherit η_x, with counts reduced by the number of fixed sources and renormalized over feasible values, allowing zero free sources. Constraints are accepted greedily using a deterministic min-fill elimination order, provided its verified induced width is at most 12. For each retained distance h, we check count conservation (Equation 18) and positive finite normalizers.

D.3 Statistical Analysis

We compare mean set sizes over 50 paired repeats, with methods sharing test examples within each repeat. For each comparison, d_r denotes JCSL's mean set size minus the comparator's on repeat r. A one-sided sign-flip test of the mean of d_r, which assumes a symmetric null distribution, uses 100,000 Monte Carlo sign vectors and $p = (1 + c)/100{,}001$, where c counts sign-flipped means not exceeding the observed mean. The upper endpoint of a two-sided 90% BCa interval from 10,000 resamples of the repeat-level d_r supplies the nominal pointwise one-sided 95% upper bound. We adjust p-values by Holm's method across 76 main study comparisons and, separately, 60 grid comparisons. In every main and grid comparison, JCSL has the smaller mean set in each of the 50 paired repeats.

D.4 Complete Main and Recall/Confidence Results

Tables 6–8 report mean $\pm$ sample standard deviation across repeats for every method and condition. Across the 15 main conditions, JCSL's mean inclusion rate is 0.8999 at nominal confidence 0.90. Tables 9–10 give the complete recall/confidence grid, reusing the datasets, splits, and baseline results from the 50 repeats for slow propagation on highSchool.

In all 50 repeats of each fixed-count setting, mean size lies strictly between 0 and $r_\beta(s)$. The score structure then implies that each output is empty or the prefix of size $r_\beta(s)$ (Appendix C.4), so 1.9–7.2% of outputs are empty. Enlarging outputs to include the first $r_\beta(s)$ candidates gives sizes of 1, 5, and 7 for source counts 1, 7, and 10 while preserving the recall lower bound. Relative to the best main study baseline means over all 50 repeats, the reductions remain 81.47–95.06%.

When its calibrated threshold equals the maximum score 1, setCP-rec's non-strict comparison includes every node and returns the full graph; Appendix D.5 reports results restricted to V_x.

Table 6: SIR with random source counts at $(\alpha, \beta) = (0.10, 0.30)$: mean $\pm$ sample SD over 50 paired repeats. Bold size values mark minima. Reductions use the smallest baseline mean.

Method	Set size ↓ (top) Inclusion target: 0.90 (bottom) Slow	Medium	Fast
highSchool			
setCP-rec	16.475 ± 0.594 0.9014 ± 0.0127	26.677 ± 0.999 0.8968 ± 0.0151	36.778 ± 1.339 0.9002 ± 0.0156
setCP-pre	18.733 ± 0.513 0.9027 ± 0.0159	32.233 ± 0.856 0.8971 ± 0.0131	47.076 ± 1.447 0.9028 ± 0.0145
setCP-min	15.867 ± 0.420 0.9012 ± 0.0143	26.259 ± 0.709 0.8965 ± 0.0151	37.516 ± 1.290 0.9010 ± 0.0170
ArbiTree	741.123 ± 8.669 0.9744 ± 0.0094	760.532 ± 4.197 0.9909 ± 0.0052	760.964 ± 4.516 0.9889 ± 0.0060
FavMac-comp.	14.834 ± 0.484 0.9017 ± 0.0158	25.245 ± 0.911 0.8995 ± 0.0175	35.450 ± 1.219 0.9024 ± 0.0175
JCSL (ours)	$\mathbf{10.327 \pm 0.328}$ 0.8929 ± 0.0177	$\mathbf{10.478 \pm 0.599}$ 0.8996 ± 0.0161	$\mathbf{10.264 \pm 0.393}$ 0.8975 ± 0.0156
Size reduction ↑	**30.39%**	**58.50%**	**71.05%**
bkFratB			
setCP-rec	13.380 ± 0.393 0.8963 ± 0.0186	19.513 ± 0.456 0.9010 ± 0.0144	24.144 ± 0.520 0.9001 ± 0.0141
setCP-pre	16.092 ± 0.433 0.9002 ± 0.0178	25.153 ± 0.721 0.8978 ± 0.0155	32.188 ± 0.968 0.8986 ± 0.0166
setCP-min	13.966 ± 0.366 0.8984 ± 0.0179	21.591 ± 0.568 0.9008 ± 0.0161	26.866 ± 0.735 0.9004 ± 0.0188
ArbiTree	56.551 ± 0.553 0.9869 ± 0.0089	57.342 ± 0.315 0.9932 ± 0.0060	57.745 ± 0.268 0.9967 ± 0.0044
FavMac-comp.	12.142 ± 0.288 0.9013 ± 0.0163	18.748 ± 0.401 0.9048 ± 0.0141	23.587 ± 0.486 0.8988 ± 0.0156
JCSL (ours)	$\mathbf{11.723 \pm 0.251}$ 0.9012 ± 0.0175	$\mathbf{17.281 \pm 0.327}$ 0.9046 ± 0.0136	$\mathbf{21.031 \pm 0.365}$ 0.9024 ± 0.0133
Size reduction ↑	**3.45%**	**7.83%**	**10.84%**

Table 7: Complete SIR results with random source counts on SFHH. Reporting follows Table 6.

Method	Set size ↓ (top) Inclusion target: 0.90 (bottom) Slow	Medium	Fast
SFHH			
setCP-rec	14.772 ± 0.481 0.9017 ± 0.0148	22.797 ± 0.722 0.9005 ± 0.0140	30.449 ± 0.901 0.9007 ± 0.0159
setCP-pre	16.133 ± 0.470 0.8989 ± 0.0176	25.695 ± 0.700 0.8989 ± 0.0142	36.749 ± 1.088 0.9010 ± 0.0146
setCP-min	14.081 ± 0.406 0.8991 ± 0.0158	21.968 ± 0.613 0.8982 ± 0.0155	30.507 ± 0.839 0.9024 ± 0.0155
ArbiTree	383.197 ± 4.737 0.9705 ± 0.0111	390.193 ± 3.610 0.9778 ± 0.0096	394.438 ± 2.766 0.9863 ± 0.0068
FavMac-comp.	13.182 ± 0.384 0.8998 ± 0.0143	21.391 ± 0.630 0.8996 ± 0.0142	29.229 ± 0.813 0.9010 ± 0.0132
JCSL (ours)	$\mathbf{11.594 \pm 0.270}$ 0.8992 ± 0.0150	$\mathbf{14.981 \pm 0.411}$ 0.8986 ± 0.0167	$\mathbf{16.756 \pm 0.426}$ 0.9027 ± 0.0180
Size reduction ↑	**12.05%**	**29.97%**	**42.67%**

Table 8: Complete highSchool results with fixed source counts. Entries are mean ± sample SD over 50 paired repeats. Here $\alpha = 0.10$; $\beta = 0$ for one source and 0.30 otherwise. Size reductions follow Table 6. N/A marks incompatible tasks; randomization permits empty sets.

	Set size ↓ (top) Inclusion target: 0.90 (bottom)		
Method	$\lvert Y\rvert = 1$	$\lvert Y\rvert = 7$	$\lvert Y\rvert = 10$
SI			
setCP-rec	23.301 ± 0.486 0.9009 ± 0.0144	28.984 ± 0.402 0.8986 ± 0.0159	40.077 ± 0.459 0.9004 ± 0.0137
setCP-pre	25.444 ± 0.585 0.8960 ± 0.0143	37.512 ± 0.366 0.8981 ± 0.0166	51.610 ± 0.625 0.8978 ± 0.0176
setCP-min	20.250 ± 0.455 0.8951 ± 0.0141	30.244 ± 0.360 0.8984 ± 0.0152	41.427 ± 0.403 0.8986 ± 0.0145
ArbiTree	626.839 ± 17.442 0.9008 ± 0.0180	770.656 ± 0.796 0.9999 ± 0.0006	770.998 ± 0.719 0.9998 ± 0.0007
FavMac-comp.	20.712 ± 0.469 0.8971 ± 0.0146	29.010 ± 0.395 0.8986 ± 0.0158	40.110 ± 0.458 0.9003 ± 0.0143
ADiT-DSI	24.224 ± 0.633 0.9864 ± 0.0047	N/A	N/A
JCSL (ours)	$\mathbf{0.941 \pm 0.013}$ 0.9005 ± 0.0161	$\mathbf{4.702 \pm 0.063}$ 0.9015 ± 0.0157	$\mathbf{6.870 \pm 0.094}$ 0.9022 ± 0.0146
Size reduction ↑	**95.35%**	**83.78%**	**82.86%**
SIR			
setCP-rec	22.835 ± 0.759 0.9038 ± 0.0123	28.193 ± 0.354 0.9002 ± 0.0169	37.776 ± 0.429 0.8992 ± 0.0148
setCP-pre	23.635 ± 0.495 0.8998 ± 0.0139	34.378 ± 0.397 0.9005 ± 0.0155	49.481 ± 0.555 0.8959 ± 0.0154
setCP-min	19.170 ± 0.602 0.9019 ± 0.0160	29.039 ± 0.395 0.9032 ± 0.0122	41.320 ± 0.587 0.8982 ± 0.0183
ArbiTree	608.586 ± 12.676 0.9009 ± 0.0134	769.221 ± 0.881 0.9999 ± 0.0006	769.996 ± 1.255 1.0000 ± 0.0004
FavMac-comp.	19.697 ± 0.642 0.9024 ± 0.0161	28.214 ± 0.343 0.9004 ± 0.0172	37.793 ± 0.428 0.8994 ± 0.0150
ADiT-DSI	N/A	N/A	N/A
JCSL (ours)	$\mathbf{0.942 \pm 0.012}$ 0.8962 ± 0.0140	$\mathbf{4.639 \pm 0.076}$ 0.8982 ± 0.0171	$\mathbf{6.728 \pm 0.080}$ 0.9016 ± 0.0166
Size reduction ↑	**95.09%**	**83.54%**	**82.19%**

Table 9: Recall/confidence grid on highSchool SIR slow. Entries are mean ± sample SD over 50 paired repeats. Targets are 0.95, 0.90, and 0.85 for $\alpha = 0.05$, 0.10, and 0.15. Bold values mark the smallest mean size. Size reductions follow Table 6.

	Set size ↓ (top) Inclusion target: $1 - \alpha$ (bottom)		
Method	$\alpha = 0.05$	$\alpha = 0.10$	$\alpha = 0.15$
$\beta = 0.10$			
setCP-rec	774.000 ± 0.000 1.0000 ± 0.0000	774.000 ± 0.000 1.0000 ± 0.0000	774.000 ± 0.000 1.0000 ± 0.0000
setCP-pre	25.275 ± 0.664 0.9495 ± 0.0092	22.732 ± 0.578 0.9000 ± 0.0148	20.774 ± 0.540 0.8497 ± 0.0179
setCP-min	19.610 ± 0.619 0.9500 ± 0.0101	18.650 ± 0.560 0.9005 ± 0.0165	17.657 ± 0.509 0.8503 ± 0.0193
ArbiTree	770.929 ± 1.040 0.9931 ± 0.0048	764.419 ± 3.899 0.9826 ± 0.0071	750.847 ± 9.686 0.9631 ± 0.0098
FavMac-comp.	19.804 ± 0.675 0.9485 ± 0.0096	19.239 ± 0.649 0.8982 ± 0.0122	18.787 ± 0.626 0.8481 ± 0.0174
JCSL (ours)	$\mathbf{16.224 \pm 0.526}$ 0.9470 ± 0.0126	$\mathbf{15.051 \pm 0.499}$ 0.8939 ± 0.0136	$\mathbf{14.184 \pm 0.464}$ 0.8427 ± 0.0162
Size reduction ↑	**17.27%**	**19.30%**	**19.67%**
$\beta = 0.30$			
setCP-rec	774.000 ± 0.000 1.0000 ± 0.0000	16.475 ± 0.594 0.9014 ± 0.0127	15.198 ± 0.536 0.8518 ± 0.0181
setCP-pre	21.467 ± 0.587 0.9512 ± 0.0094	18.733 ± 0.513 0.9027 ± 0.0159	16.808 ± 0.460 0.8526 ± 0.0187
setCP-min	17.297 ± 0.491 0.9500 ± 0.0102	15.867 ± 0.420 0.9012 ± 0.0143	14.711 ± 0.380 0.8510 ± 0.0175
ArbiTree	764.359 ± 3.355 0.9938 ± 0.0039	741.123 ± 8.669 0.9744 ± 0.0094	706.218 ± 14.602 0.9403 ± 0.0166
FavMac-comp.	16.069 ± 0.536 0.9511 ± 0.0109	14.834 ± 0.484 0.9017 ± 0.0158	14.008 ± 0.454 0.8502 ± 0.0201
JCSL (ours)	$\mathbf{11.320 \pm 0.397}$ 0.9449 ± 0.0113	$\mathbf{10.327 \pm 0.328}$ 0.8929 ± 0.0177	$\mathbf{9.610 \pm 0.298}$ 0.8427 ± 0.0192
Size reduction ↑	**29.55%**	**30.39%**	**31.40%**

Table 10: Recall/confidence grid continued for larger recall tolerances. Reporting follows Table 9.

Method	**Set size ↓ (top) Inclusion target: $1-\alpha$ (bottom)** $\alpha = 0.05$	$\alpha = 0.10$	$\alpha = 0.15$
$\beta = 0.50$			
setCP-rec	12.008 ± 0.421 0.9499 ± 0.0118	10.023 ± 0.371 0.9012 ± 0.0161	8.980 ± 0.324 0.8510 ± 0.0207
setCP-pre	16.591 ± 0.462 0.9509 ± 0.0112	13.703 ± 0.429 0.9001 ± 0.0181	11.794 ± 0.374 0.8491 ± 0.0205
setCP-min	13.829 ± 0.364 0.9493 ± 0.0112	12.091 ± 0.342 0.8991 ± 0.0167	10.740 ± 0.332 0.8499 ± 0.0201
ArbiTree	749.636 ± 6.242 0.9839 ± 0.0058	709.473 ± 12.165 0.9485 ± 0.0128	665.451 ± 14.314 0.9044 ± 0.0183
FavMac-comp.	10.143 ± 0.325 0.9492 ± 0.0107	8.941 ± 0.297 0.8982 ± 0.0169	8.137 ± 0.252 0.8466 ± 0.0190
JCSL (ours)	$\mathbf{7.510 \pm 0.241}$ 0.9489 ± 0.0113	$\mathbf{6.732 \pm 0.204}$ 0.8980 ± 0.0146	$\mathbf{6.304 \pm 0.193}$ 0.8448 ± 0.0167
Size reduction ↑	**25.95%**	**24.71%**	**22.52%**
$\beta = 0.70$			
setCP-rec	8.895 ± 0.363 0.9506 ± 0.0120	6.609 ± 0.257 0.9013 ± 0.0158	5.426 ± 0.207 0.8510 ± 0.0203
setCP-pre	13.040 ± 0.444 0.9506 ± 0.0117	9.776 ± 0.356 0.9010 ± 0.0154	7.877 ± 0.293 0.8511 ± 0.0192
setCP-min	10.312 ± 0.354 0.9512 ± 0.0105	8.175 ± 0.313 0.9023 ± 0.0160	6.793 ± 0.263 0.8515 ± 0.0207
ArbiTree	728.098 ± 10.385 0.9688 ± 0.0090	678.280 ± 13.620 0.9244 ± 0.0161	632.222 ± 14.899 0.8793 ± 0.0187
FavMac-comp.	6.313 ± 0.239 0.9495 ± 0.0129	4.925 ± 0.150 0.9003 ± 0.0169	4.290 ± 0.123 0.8490 ± 0.0174
JCSL (ours)	$\mathbf{4.807 \pm 0.135}$ 0.9501 ± 0.0109	$\mathbf{4.233 \pm 0.119}$ 0.8988 ± 0.0145	$\mathbf{3.790 \pm 0.096}$ 0.8460 ± 0.0169
Size reduction ↑	**23.86%**	**14.06%**	**11.66%**

D.5 Baseline Comparison with Matched Candidate Ranges

Protocol. To isolate the effect of the candidate range, we intersect the final outputs of ArbiTree, setCP-rec, setCP-pre, and setCP-min with the observed candidate set: $C^{\cap}(x) = C(x) \cap V_x$. Because $Y \subseteq V_X$, $C^{\cap}(X) \cap Y = C(X) \cap Y$: this operation removes only nodes that are not sources and preserves recall on every test example. It leaves each baseline's fitting, scores, and calibration unchanged. The FavMac-complement adaptation removes a calibrated exclusion set from V_x, so its prediction sets already lie in V_x and remain unchanged under this restriction. We reuse all 15 main conditions and all 12 recall/confidence settings, with the original 50 paired repeats, detector outputs, and data splits. JCSL results are held fixed.

Results. Tables 11 and 12 report the restricted baselines alongside JCSL. Candidate restriction substantially reduces ArbiTree's sets, yet its mean size remains 3.50–52.25 nodes larger than JCSL across the main conditions and 4.51–13.63 larger across the grid. JCSL has the smallest mean set size in all 108 comparisons with the four restricted baselines. Using the paired procedure in Appendix D.3, with Holm correction separately over the 60 main and 48 grid comparisons, all size differences favor JCSL with adjusted $p \leq 0.0006$. Thus the size advantage persists after matching candidate ranges.

Table 11: Comparison with matched candidate ranges for all 15 main conditions ($\alpha = 0.10$; $\beta = 0$ for one source and 0.30 otherwise). Entries are mean ± SD over 50 paired repeats. Baseline outputs are intersected with V_x; JCSL is unchanged. Bold values mark the smallest mean size. Original baseline values appear in the preceding tables of complete results.

	Set size ↓ (top) Inclusion target: 0.90 (bottom)				
Condition	setCP-rec	setCP-pre	setCP-min	ArbiTree	JCSL
highSchool	16.47 ± 0.59	16.46 ± 0.47	15.87 ± 0.42	19.09 ± 0.69	$\mathbf{10.33 \pm 0.33}$
slow	0.9014 ± 0.0127	0.9027 ± 0.0159	0.9012 ± 0.0143	0.9744 ± 0.0094	0.8929 ± 0.0177
highSchool	26.68 ± 1.00	27.88 ± 0.82	26.26 ± 0.71	34.23 ± 1.42	$\mathbf{10.48 \pm 0.60}$
medium	0.8968 ± 0.0151	0.8971 ± 0.0131	0.8965 ± 0.0151	0.9909 ± 0.0052	0.8996 ± 0.0161
highSchool	36.78 ± 1.34	40.50 ± 1.39	37.52 ± 1.29	47.57 ± 1.59	$\mathbf{10.26 \pm 0.39}$
fast	0.9002 ± 0.0156	0.9028 ± 0.0145	0.9010 ± 0.0170	0.9889 ± 0.0060	0.8975 ± 0.0156
bkFratB	13.38 ± 0.39	14.16 ± 0.37	13.97 ± 0.37	15.22 ± 0.46	$\mathbf{11.72 \pm 0.25}$
slow	0.8963 ± 0.0186	0.9002 ± 0.0178	0.8984 ± 0.0179	0.9869 ± 0.0089	0.9012 ± 0.0175
bkFratB	19.51 ± 0.46	21.99 ± 0.58	21.59 ± 0.57	23.45 ± 0.50	$\mathbf{17.28 \pm 0.33}$
medium	0.9010 ± 0.0144	0.8978 ± 0.0155	0.9008 ± 0.0161	0.9932 ± 0.0060	0.9046 ± 0.0136
bkFratB	24.14 ± 0.52	27.30 ± 0.75	26.87 ± 0.73	29.46 ± 0.62	$\mathbf{21.03 \pm 0.36}$
fast	0.9001 ± 0.0141	0.8986 ± 0.0166	0.9004 ± 0.0188	0.9967 ± 0.0044	0.9024 ± 0.0133
SFHH	14.77 ± 0.48	14.87 ± 0.42	14.08 ± 0.41	17.20 ± 0.57	$\mathbf{11.59 \pm 0.27}$
slow	0.9017 ± 0.0148	0.8989 ± 0.0176	0.8991 ± 0.0158	0.9705 ± 0.0111	0.8992 ± 0.0150
SFHH	22.80 ± 0.72	23.86 ± 0.66	21.97 ± 0.61	28.97 ± 0.98	$\mathbf{14.98 \pm 0.41}$
medium	0.9005 ± 0.0140	0.8989 ± 0.0142	0.8982 ± 0.0155	0.9778 ± 0.0096	0.8986 ± 0.0167
SFHH	30.45 ± 0.90	33.11 ± 0.99	30.51 ± 0.84	40.68 ± 1.19	$\mathbf{16.76 \pm 0.43}$
fast	0.9007 ± 0.0159	0.9010 ± 0.0146	0.9024 ± 0.0155	0.9863 ± 0.0068	0.9027 ± 0.0180
SI	23.30 ± 0.49	22.99 ± 0.54	20.25 ± 0.46	30.84 ± 1.31	$\mathbf{0.94 \pm 0.01}$
$\lvert Y\rvert = 1$	0.9009 ± 0.0144	0.8960 ± 0.0143	0.8951 ± 0.0141	0.9008 ± 0.0180	0.9005 ± 0.0161
SI	28.98 ± 0.40	36.39 ± 0.37	30.24 ± 0.36	41.79 ± 0.43	$\mathbf{4.70 \pm 0.06}$
$\lvert Y\rvert = 7$	0.8986 ± 0.0159	0.8981 ± 0.0166	0.8984 ± 0.0152	0.9999 ± 0.0006	0.9015 ± 0.0157
SI	40.08 ± 0.46	50.81 ± 0.60	41.43 ± 0.40	58.96 ± 0.47	$\mathbf{6.87 \pm 0.09}$
$\lvert Y\rvert = 10$	0.9004 ± 0.0137	0.8978 ± 0.0176	0.8986 ± 0.0145	0.9998 ± 0.0007	0.9022 ± 0.0146
SIR	22.83 ± 0.76	21.71 ± 0.50	19.17 ± 0.60	29.46 ± 1.31	$\mathbf{0.94 \pm 0.01}$
$\lvert Y\rvert = 1$	0.9038 ± 0.0123	0.8998 ± 0.0139	0.9019 ± 0.0160	0.9009 ± 0.0134	0.8962 ± 0.0140
SIR	28.19 ± 0.35	33.88 ± 0.39	29.04 ± 0.40	41.83 ± 0.40	$\mathbf{4.64 \pm 0.08}$
$\lvert Y\rvert = 7$	0.9002 ± 0.0169	0.9005 ± 0.0155	0.9032 ± 0.0122	0.9999 ± 0.0006	0.8982 ± 0.0171
SIR	37.78 ± 0.43	48.44 ± 0.51	41.32 ± 0.59	58.98 ± 0.48	$\mathbf{6.73 \pm 0.08}$
$\lvert Y\rvert = 10$	0.8992 ± 0.0148	0.8959 ± 0.0154	0.8982 ± 0.0183	1.0000 ± 0.0004	0.9016 ± 0.0166

Table 12: Comparison with matched candidate ranges for all 12 recall/confidence settings under slow propagation on highSchool. Entries are mean ± SD over 50 paired repeats. Targets are 0.95, 0.90, and 0.85 for $\alpha = 0.05$, 0.10, and 0.15. Baseline outputs are intersected with V_x; JCSL is unchanged. Bold values mark the smallest mean size. Original baseline values appear in the preceding tables of complete results.

	Set size ↓ (top) Inclusion target: $1 - \alpha$ (bottom)				
(α, β)	setCP-rec	setCP-pre	setCP-min	ArbiTree	JCSL
$(0.05, 0.10)$	20.89 ± 0.74 1.0000 ± 0.0000	19.73 ± 0.64 0.9495 ± 0.0092	19.61 ± 0.62 0.9500 ± 0.0101	20.73 ± 0.72 0.9931 ± 0.0048	$\mathbf{16.22 \pm 0.53}$ 0.9470 ± 0.0126
$(0.10, 0.10)$	20.89 ± 0.74 1.0000 ± 0.0000	18.68 ± 0.56 0.9000 ± 0.0148	18.65 ± 0.56 0.9005 ± 0.0165	20.41 ± 0.70 0.9826 ± 0.0071	$\mathbf{15.05 \pm 0.50}$ 0.8939 ± 0.0136
$(0.15, 0.10)$	20.89 ± 0.74 1.0000 ± 0.0000	17.67 ± 0.51 0.8497 ± 0.0179	17.66 ± 0.51 0.8503 ± 0.0193	19.66 ± 0.77 0.9631 ± 0.0098	$\mathbf{14.18 \pm 0.46}$ 0.8427 ± 0.0162
$(0.05, 0.30)$	20.89 ± 0.74 1.0000 ± 0.0000	18.04 ± 0.54 0.9512 ± 0.0094	17.30 ± 0.49 0.9500 ± 0.0102	20.30 ± 0.72 0.9938 ± 0.0039	$\mathbf{11.32 \pm 0.40}$ 0.9449 ± 0.0113
$(0.10, 0.30)$	16.47 ± 0.59 0.9014 ± 0.0127	16.46 ± 0.47 0.9027 ± 0.0159	15.87 ± 0.42 0.9012 ± 0.0143	19.09 ± 0.69 0.9744 ± 0.0094	$\mathbf{10.33 \pm 0.33}$ 0.8929 ± 0.0177
$(0.15, 0.30)$	15.20 ± 0.54 0.8518 ± 0.0181	15.19 ± 0.42 0.8526 ± 0.0187	14.71 ± 0.38 0.8510 ± 0.0175	17.41 ± 0.64 0.9403 ± 0.0166	$\mathbf{9.61 \pm 0.30}$ 0.8427 ± 0.0192
$(0.05, 0.50)$	12.01 ± 0.42 0.9499 ± 0.0118	15.04 ± 0.42 0.9509 ± 0.0112	13.83 ± 0.36 0.9493 ± 0.0112	19.51 ± 0.67 0.9839 ± 0.0058	$\mathbf{7.51 \pm 0.24}$ 0.9489 ± 0.0113
$(0.10, 0.50)$	10.02 ± 0.37 0.9012 ± 0.0161	12.90 ± 0.39 0.9001 ± 0.0181	12.09 ± 0.34 0.8991 ± 0.0167	17.53 ± 0.62 0.9485 ± 0.0128	$\mathbf{6.73 \pm 0.20}$ 0.8980 ± 0.0146
$(0.15, 0.50)$	8.98 ± 0.32 0.8510 ± 0.0207	11.33 ± 0.35 0.8491 ± 0.0205	10.74 ± 0.33 0.8499 ± 0.0201	15.56 ± 0.62 0.9044 ± 0.0183	$\mathbf{6.30 \pm 0.19}$ 0.8448 ± 0.0167
$(0.05, 0.70)$	8.90 ± 0.36 0.9506 ± 0.0120	12.36 ± 0.40 0.9506 ± 0.0117	10.31 ± 0.35 0.9512 ± 0.0105	18.44 ± 0.63 0.9688 ± 0.0090	$\mathbf{4.81 \pm 0.14}$ 0.9501 ± 0.0109
$(0.10, 0.70)$	6.61 ± 0.26 0.9013 ± 0.0158	9.55 ± 0.34 0.9010 ± 0.0154	8.17 ± 0.31 0.9023 ± 0.0160	16.14 ± 0.62 0.9244 ± 0.0161	$\mathbf{4.23 \pm 0.12}$ 0.8988 ± 0.0145
$(0.15, 0.70)$	5.43 ± 0.21 0.8510 ± 0.0203	7.79 ± 0.29 0.8511 ± 0.0192	6.79 ± 0.26 0.8515 ± 0.0207	14.26 ± 0.63 0.8793 ± 0.0187	$\mathbf{3.79 \pm 0.10}$ 0.8460 ± 0.0169

D.6 Baselines with Shared Score Correction

The main comparison retains the baseline scoring rules in Section 5.1. These additional controls incorporate JCSL's fitted score correction into setCP-rec, setCP-pre, setCP-min, and FavMac-complement. Corrected probabilities apply the sigmoid to clipped detector log odds plus $g_v(x)$ (Equation 3), using the coefficient a fitted on the 1,520 fitting examples. Both raw and corrected versions use the same V_x, 6,080 calibration examples, and 400 test examples, over repeats 40–49 of all 15 conditions. The main comparison uses 50 repeats and 7,600 calibration examples for these baselines; Table 13 therefore recomputes both versions. For each condition, each column reports the baseline with the smallest mean size across the ten repeats.

Table 13: Shared score correction over ten paired repeats (40–49) per condition. All methods use the same 6,080 calibration examples, 400 test examples, and candidate sets in each repeat. Raw and corrected columns each select the baseline with the smallest mean size across all ten repeats among setCP-rec, setCP-pre, setCP-min, and FavMac-comp. Set sizes are mean ± SD and inclusion rates are means. Superscripts identify the selected method: R, setCP-rec; M, setCP-min; F, FavMac-comp. Reduction compares JCSL with the corrected baseline.

	Set size ↓ (top) Inclusion target: 0.90 (bottom)			
Condition	Raw baseline	Corrected baseline	JCSL	**Reduction ↑**
highSchool SIR, random counts				
	$15.08 \pm 0.42^{\mathrm{F}}$	$14.07 \pm 0.36^{\mathrm{M}}$	10.52 ± 0.33	
Slow	0.9120	0.8970	0.8900	25.25%
	$24.84 \pm 0.66^{\mathrm{F}}$	$19.19 \pm 0.87^{\mathrm{F}}$	10.44 ± 0.57	
Medium	0.9050	0.9030	0.9030	45.60%
	$35.00 \pm 1.32^{\mathrm{F}}$	$20.55 \pm 1.15^{\mathrm{F}}$	10.13 ± 0.21	
Fast	0.9020	0.8933	0.8993	50.70%
bkFratB SIR, random counts				
	$12.02 \pm 0.24^{\mathrm{F}}$	$11.99 \pm 0.27^{\mathrm{F}}$	11.61 ± 0.24	
Slow	0.9013	0.9035	0.9083	3.19%
	$18.80 \pm 0.42^{\mathrm{F}}$	$18.14 \pm 0.43^{\mathrm{F}}$	17.39 ± 0.32	
Medium	0.9030	0.9068	0.9055	4.14%
	$23.74 \pm 0.62^{\mathrm{F}}$	$21.94 \pm 0.63^{\mathrm{F}}$	21.13 ± 0.46	
Fast	0.8943	0.9000	0.9073	3.71%
SFHH SIR, random counts				
	$13.12 \pm 0.32^{\mathrm{F}}$	$13.05 \pm 0.32^{\mathrm{F}}$	11.58 ± 0.28	
Slow	0.8998	0.8985	0.9043	11.27%
	$21.38 \pm 0.47^{\mathrm{F}}$	$19.24 \pm 0.47^{\mathrm{F}}$	15.05 ± 0.38	
Medium	0.9073	0.9028	0.8975	21.77%
	$29.23 \pm 0.65^{\mathrm{F}}$	$22.87 \pm 0.53^{\mathrm{F}}$	16.89 ± 0.40	
Fast	0.8898	0.8950	0.8995	26.14%
highSchool SI, fixed counts				
	$20.33 \pm 0.29^{\mathrm{M}}$	$3.32 \pm 0.15^{\mathrm{F}}$	0.94 ± 0.01	
1 source	0.8998	0.9025	0.9010	71.70%
	$29.06 \pm 0.33^{\mathrm{R}}$	$12.24 \pm 0.14^{\mathrm{F}}$	4.70 ± 0.05	
7 sources	0.8978	0.9013	0.8983	61.61%
	$40.12 \pm 0.34^{\mathrm{R}}$	$18.02 \pm 0.20^{\mathrm{F}}$	6.84 ± 0.15	
10 sources	0.9010	0.9008	0.8990	62.05%
highSchool SIR, fixed counts				
	$19.33 \pm 0.82^{\mathrm{M}}$	$3.58 \pm 0.17^{\mathrm{F}}$	0.94 ± 0.01	
1 source	0.9030	0.8943	0.8873	73.79%
	$28.16 \pm 0.40^{\mathrm{R}}$	$11.02 \pm 0.18^{\mathrm{F}}$	4.65 ± 0.04	
7 sources	0.9000	0.9045	0.8958	57.84%
	$37.66 \pm 0.57^{\mathrm{F}}$	$16.08 \pm 0.25^{\mathrm{F}}$	6.72 ± 0.12	
10 sources	0.9090	0.9018	0.9055	58.23%

JCSL retains smaller mean sets in every condition, reducing size by 3.19–73.79% relative to the best corrected baseline. It is smaller in all 150 condition–repeat pairs; exact one-sided sign-flip tests over the ten paired repeats give Holm-adjusted $p < 0.015$ in every condition. These comparisons assess the complete framework beyond the shared score correction. The conditioning ablation in Appendix D.7 isolates the effect of propagation conditioning. With the enlargement in Appendix D.4 (sizes 1, 5, and 7), the reductions in the six fixed-count conditions remain 54.65–72.04%.

D.7 Ablation Protocol

The ablation uses the last ten repeats (40–49) in each of the 15 main conditions. All ten-repeat analyses in this appendix use these repeats and the same full-JCSL runs. All variants share the fitting, calibration, and test examples and the randomization variables. Each variant is recalibrated.

No conditioning. We use P_0 in place of P_{R} when computing node priorities. We recompute the node order and the P_0 score curve along that order. This retains the fitted score correction, count probabilities, and set scoring rule while removing propagation conditioning. The complementary *no score correction* variant sets $g_v(x) = 0$, retains propagation conditioning and the fitted count and distance weights, and recomputes P_0, P_{R}, the node order, and the OA score curve before recalibration. This changes the probabilities used in both inference and scoring. Table 14 compares the two interventions on the same repeats. Removing conditioning increases mean size in every condition. Removing the correction increases size in all nine conditions with random source counts and changes it by less than 2% on average in the fixed-count groups.

Table 14: Effects of joint conditioning and score correction over ten paired repeats (40–49) per condition. Entries are mean paired percentage size changes and inclusion rates for full → ablated JCSL. Each group averages three conditions and ten repeats equally; positive changes mean larger sets. No conditioning retains the fitted score correction; $g_v = 0$ retains joint conditioning.

Size change (%) ↓ (top) Inclusion target: 0.90 (bottom)		
Condition group	No conditioning	No score correction ($g_v = 0$)
Random highSchool	$+47.86$ $0.8974 \to 0.8996$	$+39.01$ $0.8974 \to 0.8968$
Random bkFratB	$+1.02$ $0.9070 \to 0.9073$	$+2.77$ $0.9070 \to 0.9008$
Random SFHH	$+14.65$ $0.9004 \to 0.9013$	$+9.84$ $0.9004 \to 0.8939$
Fixed counts, SI	$+175.31$ $0.8994 \to 0.8997$	-1.51 $0.8994 \to 0.8988$
Fixed counts, SIR	$+165.69$ $0.8962 \to 0.9018$	-1.46 $0.8962 \to 0.8957$

Unary only. We set the elimination width cap to zero, preserving the other fitted choices. After substituting forced source indicators, retained factors involve at most one free candidate variable. This removes propagation constraints involving multiple free candidates. The intervention applies when source counts are random or fixed above one; inference for one source is unchanged and is excluded from this aggregate.

Prefix fraction score. We keep the full JCSL order and replace its score curve by $k/|V_x|$ at every prefix length k. The same calibration and inversion procedure then selects a cutoff on this curve. This compares OA with rank-only prefix scoring while preserving the refined order and candidate set.

Table 3 averages paired percentage changes in each repeat's mean test set size over applicable conditions and repeats. Inclusion rates use the same subset for each variant. Appendix E.1 reports a variant with uniform count weights.

Table 15 reports absolute means over the same ten repeats, using the smallest baseline mean as reference. Relative to the unconditioned variant, propagation conditioning reduces mean size by 60.46–65.63% in each fixed-count condition and by up to 37.66% with random source counts,

Table 15: Absolute ablation results over ten paired repeats (40–49) per condition. Entries are means. The variant without conditioning retains P_0 and is recalibrated. The listed baseline has the smallest mean size among the evaluated baselines on these same repeats.

	Set size ↓ (top)	**Inclusion target: 0.90 (bottom)**		
Condition	Baseline	Baseline result	No conditioning	Full JCSL
highSchool SIR, random counts				
Slow	FavMac-comp.	15.07 0.9113	13.39 0.8983	10.52 0.8900
Medium	FavMac-comp.	24.84 0.9050	16.23 0.9048	10.44 0.9030
Fast	FavMac-comp.	35.04 0.9028	16.25 0.8958	10.13 0.8993
bkFratB SIR, random counts				
Slow	FavMac-comp.	12.02 0.9010	11.66 0.9075	11.61 0.9083
Medium	FavMac-comp.	18.80 0.9030	17.61 0.9065	17.39 0.9055
Fast	FavMac-comp.	23.73 0.8940	21.42 0.9078	21.13 0.9073
SFHH SIR, random counts				
Slow	FavMac-comp.	13.11 0.8995	12.32 0.9033	11.58 0.9043
Medium	FavMac-comp.	21.38 0.9075	17.47 0.9023	15.05 0.8975
Fast	FavMac-comp.	29.21 0.8900	20.51 0.8985	16.89 0.8995
highSchool SI, fixed counts				
1 source	setCP-min	20.31 0.8995	2.58 0.8998	0.94 0.9010
7 sources	setCP-rec	29.08 0.8995	12.89 0.9030	4.70 0.8983
10 sources	setCP-rec	40.09 0.8998	18.99 0.8963	6.84 0.8990
highSchool SIR, fixed counts				
1 source	setCP-min	19.34 0.9045	2.73 0.9038	0.94 0.8873
7 sources	setCP-rec	28.17 0.9015	11.76 0.8975	4.65 0.8958
10 sources	setCP-rec	37.65 0.9083	16.99 0.9040	6.72 0.9055

measured as ratios of ten-repeat means. On SI with ten sources, it reduces mean size from 18.99 to 6.84 (63.99%), demonstrating gains from propagation constraints even though no nodes recover.

D.8 Alternative Node Orders

We compare JCSL's priorities with raw detector ordering and one-step graph diffusion while retaining its order-adaptive scoring rule. Both controls use the same candidate set V_x, fitted reference distribution P_0, and recall target as JCSL. For each order, we recompute the P_0 score curve along its prefixes and calibrate a separate threshold. The comparison uses repeats 40–49 of all 15 main conditions, with shared fitting, calibration, and test examples and randomization variables. Thus the controls evaluate the full priority construction, including score correction and propagation conditioning.

Ordering rules. *Raw detector + OA* sorts candidates by their original detector scores $p_v(x)$. *Graph diffusion + OA* sorts candidates by an interpolation of their detector scores and neighbor averages on the original graph. The interpolation weight is selected from $\{0, 0.25, 0.50, 0.75, 1\}$ using only the

1,520 fitting examples, minimizing the mean prefix length needed to attain the recall target; ties favor the smaller weight.

Results. Table 16 reports set sizes and inclusion rates. Both controls produce larger mean sets than JCSL in every condition. Since they share JCSL's count probabilities through P_0, count information alone does not explain these differences. Graph diffusion improves on raw detector ordering in all 15 conditions, but JCSL's priorities yield further gains under the same scoring rule. For example, on highSchool SIR fast, raw ordering, graph diffusion, and JCSL give mean sizes of 29.68, 19.79, and 10.13, respectively. The paired size difference (control minus JCSL) has a positive lower endpoint in all 30 pointwise 95% bootstrap intervals, computed from 10,000 resamples of the ten paired repeats. These comparisons support the efficiency of JCSL's priority construction beyond raw scores and local graph smoothing.

Table 16: Alternative node orders with order-adaptive (OA) scoring. Entries average ten paired repeats (40–49), with sample SD for set sizes. Each order uses the same fitted P_0 but is scored and calibrated separately. We use $\alpha = 0.10$, with $\beta = 0.30$ for multiple sources and $\beta = 0$ for one. Bold values mark the smallest mean size.

	Set size ↓ (top) Inclusion target: 0.90 (bottom)		
Condition	Raw detector + OA	Graph diffusion + OA	Full JCSL
highSchool SIR, random counts			
	14.324 ± 0.382	14.020 ± 0.404	$\mathbf{10.520 \pm 0.334}$
Slow	0.9035	0.9035	0.8900
	22.297 ± 0.584	17.753 ± 0.386	$\mathbf{10.438 \pm 0.566}$
Medium	0.8983	0.8983	0.9030
	29.680 ± 0.729	19.786 ± 0.481	$\mathbf{10.133 \pm 0.213}$
Fast	0.8965	0.8975	0.8993
bkFratB SIR, random counts			
	11.781 ± 0.228	11.723 ± 0.250	$\mathbf{11.609 \pm 0.242}$
Slow	0.9028	0.9078	0.9083
	18.282 ± 0.388	17.728 ± 0.385	$\mathbf{17.386 \pm 0.320}$
Medium	0.8995	0.9030	0.9055
	22.970 ± 0.445	21.593 ± 0.410	$\mathbf{21.130 \pm 0.464}$
Fast	0.8968	0.9080	0.9073
SFHH SIR, random counts			
	12.638 ± 0.292	12.515 ± 0.286	$\mathbf{11.581 \pm 0.279}$
Slow	0.9033	0.9058	0.9043
	19.808 ± 0.447	18.062 ± 0.431	$\mathbf{15.052 \pm 0.383}$
Medium	0.8935	0.9000	0.8975
	26.494 ± 0.612	22.377 ± 0.498	$\mathbf{16.888 \pm 0.396}$
Fast	0.8955	0.8930	0.8995
highSchool SI, fixed counts			
	14.380 ± 0.606	4.088 ± 0.343	$\mathbf{0.941 \pm 0.012}$
1 source	0.9035	0.8985	0.9010
	26.445 ± 0.551	13.150 ± 0.232	$\mathbf{4.699 \pm 0.054}$
7 sources	0.8895	0.9020	0.8983
	37.686 ± 0.538	18.762 ± 0.278	$\mathbf{6.839 \pm 0.148}$
10 sources	0.8965	0.8955	0.8990
highSchool SIR, fixed counts			
	13.003 ± 0.521	5.024 ± 0.259	$\mathbf{0.938 \pm 0.010}$
1 source	0.9043	0.8953	0.8873
	25.513 ± 0.849	14.149 ± 0.275	$\mathbf{4.648 \pm 0.036}$
7 sources	0.8920	0.8968	0.8958
	35.440 ± 0.894	19.763 ± 0.308	$\mathbf{6.717 \pm 0.116}$
10 sources	0.8988	0.9035	0.9055

Recall at matched budgets. We also compare the same saved node orders at budgets $k \in \{1, 5, 10\}$, retaining the first $\min\{k, |V_x|\}$ candidates of each order. Table 17 reports mean source recall and the fraction meeting the recall target over the same ten paired repeats. JCSL yields higher mean recall than both controls in every condition at all three budgets. Target attainment is also higher whenever the budget allows the target to be reached; all orders have zero success when fewer than $r_\beta(s)$ nodes can be selected. These diagnostics measure ordering quality before conformal calibration.

Table 17: Ordering diagnostics at three shared candidate budgets. Every order retains its first $\min\{k, |V_x|\}$ candidates, giving identical set sizes for each example. Entries are means over the same ten paired repeats (40–49) in all 15 main conditions, with $\beta = 0$ for tasks with one source and 0.30 otherwise. Settings with fixed source counts use highSchool. Raw and Diff. denote raw detector and one-step graph diffusion ordering. Panel (a) reports average source recall; panel (b) reports the fraction attaining recall $\geq 1 - \beta$. These diagnostics evaluate ordering before calibration.

(a) Mean source recall ↑

Condition	$k = 1$			$k = 5$			$k = 10$		
	JCSL	Raw	Diff.	JCSL	Raw	Diff.	JCSL	Raw	Diff.
highSchool slow	0.2037	0.1742	0.1635	0.6452	0.5351	0.5134	0.8665	0.7316	0.7306
highSchool medium	0.2203	0.1732	0.1753	0.6674	0.4785	0.5164	0.8916	0.6333	0.7015
highSchool fast	0.2256	0.1673	0.1814	0.6760	0.4388	0.5095	0.9069	0.5768	0.6879
bkFratB slow	0.1916	0.1839	0.1808	0.5563	0.5520	0.5518	0.7704	0.7687	0.7694
bkFratB medium	0.1840	0.1668	0.1626	0.4845	0.4609	0.4675	0.6619	0.6427	0.6533
bkFratB fast	0.1828	0.1603	0.1501	0.4568	0.4156	0.4320	0.6167	0.5728	0.6024
SFHH slow	0.1985	0.1764	0.1660	0.6003	0.5586	0.5481	0.8018	0.7673	0.7647
SFHH medium	0.2085	0.1711	0.1651	0.5900	0.4960	0.4982	0.7614	0.6670	0.6821
SFHH fast	0.2182	0.1701	0.1695	0.6044	0.4567	0.4784	0.7618	0.6090	0.6471
SI, 1 source	0.9583	0.1733	0.4110	1.0000	0.4538	0.8940	1.0000	0.6260	0.9750
SI, 7 sources	0.1413	0.0713	0.1158	0.7080	0.2441	0.5115	0.9938	0.3978	0.7690
SI, 10 sources	0.0984	0.0551	0.0787	0.4952	0.1898	0.3637	0.9349	0.3085	0.6172
SIR, 1 source	0.9460	0.2868	0.3743	0.9998	0.5660	0.8908	1.0000	0.6895	0.9663
SIR, 7 sources	0.1422	0.1193	0.1126	0.7090	0.3300	0.4619	0.9951	0.4648	0.7199
SIR, 10 sources	0.0993	0.0905	0.0785	0.4963	0.2790	0.3364	0.9459	0.3906	0.5679

(b) Rate of meeting the recall target ↑

Condition	$k = 1$			$k = 5$			$k = 10$		
	JCSL	Raw	Diff.	JCSL	Raw	Diff.	JCSL	Raw	Diff.
highSchool slow	0.0508	0.0360	0.0428	0.4138	0.2448	0.2380	0.7978	0.5268	0.5220
highSchool medium	0.0660	0.0365	0.0513	0.4450	0.1835	0.2398	0.8700	0.3615	0.4790
highSchool fast	0.0700	0.0353	0.0525	0.4618	0.1483	0.2260	0.8945	0.2855	0.4440
bkFratB slow	0.0548	0.0443	0.0445	0.2785	0.2713	0.2715	0.5980	0.5938	0.5915
bkFratB medium	0.0538	0.0375	0.0393	0.1965	0.1700	0.1760	0.4015	0.3800	0.3925
bkFratB fast	0.0573	0.0355	0.0370	0.1740	0.1308	0.1458	0.3345	0.2750	0.3175
SFHH slow	0.0460	0.0325	0.0323	0.3418	0.2678	0.2630	0.6620	0.5965	0.5863
SFHH medium	0.0580	0.0373	0.0405	0.3460	0.1998	0.2130	0.5858	0.4100	0.4455
SFHH fast	0.0668	0.0370	0.0413	0.3708	0.1595	0.1918	0.5930	0.3323	0.3850
SI, 1 source	0.9583	0.1733	0.4110	1.0000	0.4538	0.8940	1.0000	0.6260	0.9750
SI, 7 sources	0.0000	0.0000	0.0000	0.9565	0.0033	0.1283	0.9998	0.0905	0.8365
SI, 10 sources	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.9950	0.0065	0.3893
SIR, 1 source	0.9460	0.2868	0.3743	0.9998	0.5660	0.8908	1.0000	0.6895	0.9663
SIR, 7 sources	0.0000	0.0000	0.0000	0.9640	0.0140	0.0610	1.0000	0.1623	0.7263
SIR, 10 sources	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.9988	0.0260	0.2323

D.9 Runtime Protocol

For the methods in Table 18, we time five serial executions on main study repeat 40 after one untimed warmup for each method and condition. Each execution starts a new process and recomputes the method's fitted objects and predictions. Repeated executions return identical prediction sets for each method and condition. All single-source timings use an Intel Xeon Platinum 8358P CPU at 2.60 GHz; multi-source JCSL additionally uses an NVIDIA RTX 4090. Threads used by numerical libraries are

Table 18: Runtime in seconds, median [Q1,Q3] of five serial runs after a warmup. Test time covers 400 examples. The number after the slash gives the CPU worker count. Shared detector inference is excluded; method-specific fitting and test preprocessing are included.

Method	Hardware / workers	Fit + calibration	Test / 400
highSchool: SI, one source			
setCP-rec	CPU / 1	0.278 [0.267, 0.301]	0.094 [0.091, 0.101]
setCP-pre	CPU / 1	0.263 [0.257, 0.273]	0.104 [0.098, 0.107]
ArbiTree	CPU / 5	4.529 [4.488, 4.550]	69.038 [68.839, 69.672]
ADiT-DSI	CPU / 1	–	911.701 [908.467, 911.914]
FavMac-comp.	CPU / 1	0.235 [0.227, 0.237]	0.009 [0.009, 0.009]
JCSL	CPU / 5	8.839 [8.707, 8.848]	0.319 [0.315, 0.329]
highSchool: SIR, ten sources			
JCSL	4090 + CPU / 5	665.286 [664.123, 666.364]	46.232 [45.729, 46.600]

fixed to one per process. ArbiTree and single-source JCSL each use five CPU workers. Multi-source JCSL uses five CPU workers to prepare inference problems and processes them in GPU batches of up to 256. On the GPU, problems with at most 13 unsatisfied constraints use an equivalent dynamic program over constraint-satisfaction states; the others use variable elimination.

Shared data loading, initialization of the timing process, detector training, detector inference, metric evaluation, and output writing are excluded. Fitting and calibration include each method's parameter selection, transformations, structural computation, and threshold calculation. Test inference includes structure construction, method worker startup, planning, device transfers, and final set generation for all 400 examples. GPU phases are synchronized at their boundaries. ADiT-DSI has no fitting stage on the calibration pool; its graph simulations are included in test time.

On the representative task with one source, JCSL test time lies between that of the inexpensive methods using only scores and those of ArbiTree and ADiT-DSI. The multi-source measurement quantifies the additional cost of structured inference on a different task and device configuration.

E Sensitivity to Count Probabilities

E.1 Uniform Count Probabilities

We assess whether prediction set efficiency depends on the particular count probabilities in P_0 (Section 3.1). On the same ten repeats as the ablations, *Uniform count law* replaces the fitted probabilities of supported source counts with equal weights. All other parameters, data, seeds, and calibration rules are shared.

Table 19: Count law sensitivity on the main splits. Mean paired size changes (%) are relative to full JCSL (positive: larger sets). N counts matched evaluations over three speeds (ten per condition).

		Size change (%) ↓	**Inclusion target: 0.90**	
Condition group	N	**Size change (%) ↓**	Full incl.	Uniform incl.
Random highSchool	30	−0.11	0.8974	0.8975
Random bkFratB	30	+0.00	0.9070	0.9081
Random SFHH	30	−0.01	0.9004	0.9005

Uniform count weights apply to the nine conditions with random source counts. Their mean paired size change is at most 0.11% in magnitude across the three networks (Table 19).

E.2 Nonuniform Source Counts

The main benchmark samples source counts uniformly before retaining simulations lasting more than 16 recorded iterations. We compare fitted and uniform count probabilities on samples reweighted to

favor fewer sources, using the condition with medium propagation speed on each network and the first five ablation repeats (40–44).

Each example with s sources is independently retained with probability $\exp[-0.18(s-1)]$, so that examples with one source are retained about 12 times as often as those with 15 sources. The same rule applies to the original fitting, calibration, and test pools. Under the original exchangeability assumption, the retained calibration and test examples remain exchangeable. Both variants are recalibrated on the retained calibration pool and share retained examples, refitted distance weights, score correction, count support, and calibration randomness.

Table 20: Count law sensitivity after count reweighting (five repeats per network at medium propagation speed). Variant entries are means. N_{test} totals retained test examples. Change is the mean paired percentage size change from empirical to uniform weights (negative: smaller sets). The minimum attainable exact two-sided sign-flip p-value is 0.0625.

		Set size ↓ / Inclusion target: 0.90			
Network	N_{test}	Empirical	Uniform	**Change (%) ↓**	p
highSchool	745	$6.778/0.9119$	$6.461/0.9043$	-4.67	0.0625
bkFratB	772	$11.930/0.9096$	$11.874/0.8952$	-0.45	0.3750
SFHH	710	$9.582/0.9027$	$9.211/0.9104$	-3.80	0.0625

Fitted and uniform count probabilities differ in mean set size by 0.45–4.67% (Table 20), although the count law changes node orderings for 43.9–60.5% of test examples.