GRAPHBET: PROSPECTIVE FDR-CONTROLLED DISCOVERY FROM TEMPORAL GRAPH PREDICTIONS

Anonymous authors
Paper under double-blind review

ABSTRACT

Temporal graph models rank likely future interactions, but a high score does not establish that a nominated group attracts more subsequent activity than its available opportunities explain. We introduce GRAPHBET, which freezes a nominated destination set and compounds predictable likelihood ratios from later interactions relative to pre-event opportunity probabilities. Under the directional null that the set attracts no more probability than those opportunities permit, the resulting wealth is an e-process. We prove graph-wide validity for adaptively selected, dependent candidates with history-dependent null labels; e-value Benjamini–Hochberg (e-BH) controls the false discovery rate (FDR) at a common reporting time, and the accumulated union of earlier e-BH selections also controls FDR. We additionally establish safe aggregation of betting experts, robustness to bounded reference error, and finite-sample links from predictive accuracy and betting regret to detection. We observe no final-snapshot false discoveries in 300 controlled global-null streams or 200 exact-law benchmark replays. In a crossed study of three prespecified training fits and 20 shared held-out semi-synthetic Wiki replays, learned candidates raise mean conditional testing power from .335 to .418 under a common fixed bet, with positive evidence-growth gains in every fit. A prespecified six-expert mixture raises power to .452 and achieves the highest mean end-to-end recall among the evaluated pipelines, .084 versus .069 for degree–recency candidates with the fixed bet, with no final-snapshot false discoveries. Thus the opportunity law governs validity, candidate learning governs what is nominated, and betting calibration governs how efficiently evidence accumulates.

1 INTRODUCTION

Temporal graph models use timestamped interactions to predict which edge or event will occur next (Rossi et al., 2020; Huang et al., 2023; Gastinger et al., 2024). In many applications, however, ranking is only an intermediate step. After a model nominates a group of destinations, the scientific question is whether later interactions concentrate on that group more than the available opportunities would explain. A high neural score does not answer this question because it neither specifies what could have occurred instead nor quantifies post-nomination evidence.

Consider an editor for whom a model nominates five pages. If those pages account for 40% of the editor's available activity-weighted opportunities, a subsequent edit to the group is not especially surprising; the same event is much stronger evidence if the pages account for only 1%. We call this pre-event quantity the candidate set's *reference probability*. It can change as nodes enter, leave, or accumulate activity, and it must be computed before the next destination is observed. Dynamic risk sets and history-dependent intensities play related roles in relational-event models (Lerner & Lomi, 2020; Bianchi et al., 2024); a neural ranking score alone does not provide the required reference.

Three forms of adaptivity complicate valid discovery: history used to nominate a candidate cannot also support it; overlapping candidates share nodes, events, and model parameters; and an analyst may inspect the graph repeatedly. The analysis must therefore use only post-nomination outcomes and condition on every graph-wide quantity available before each destination is revealed (Ramdas et al., 2023; Wang et al., 2025a; Fischer et al., 2024; Tavyrikov et al., 2026).

GRAPHBET separates nomination from prospective verification. At registration, it freezes a source, relation, and destination set selected from completed history. Before each later event with that source

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.

and relation, an opportunity model assigns the set reference probability p_0, while a predictable graph model supplies $\widehat{p} \geq p_0$. A hit multiplies evidence by $\widehat{p}/p_0$ and a miss by $(1-\widehat{p})/(1-p_0)$. Under the null that the set's true conditional probability is at most p_0, this multiplier has conditional expectation at most one, yielding an e-process that remains valid at data-dependent reporting times (Grünwald et al., 2021; Shaer et al., 2023).

This construction separates validity from utility: the opportunity model defines the null, the temporal representation determines which enriched patterns are nominated, and the predictable bet determines how efficiently their enrichment becomes evidence. A safe mixture hedges across prespecified betting strengths without selecting one from evaluation outcomes.

Our contributions are as follows.

- We formulate a prospective enrichment hypothesis for temporal-graph candidates relative to dynamic, pre-event opportunity probabilities. Candidate identities may be selected from the evolving graph, but only interactions observed after registration contribute evidence.
- We establish a graph-wide e-process result for adaptively selected candidate identities and history-dependent null labels. The construction permits overlapping candidates and shared predictable neural models and yields FDR control for both a contemporaneous e-BH report and the accumulated union of candidates selected at preceding e-BH snapshots.
- We derive an exactly computable typed-stub opportunity law, sensitivity guarantees for prespecified one-sided reference error, and safe aggregation of betting experts. Finite-sample results connect stable destination forecasts to candidate enrichment and separate graph signal from betting regret in the time needed for detection.
- Controlled heterogeneous streams, exact-law replays, observational diagnostics, and a crossed fit-by-replay Wiki study evaluate the complete pipeline. Learned candidates consistently improve conditional power and evidence growth under a common fixed bet, and safe aggregation gives the strongest mean conditional power and end-to-end recall among the evaluated Wiki pipelines.

Building on e-processes and e-value multiple testing (Wang & Ramdas, 2022; Xu et al., 2021; Xu & Ramdas, 2024; Fischer et al., 2024), GRAPHBET makes their guarantees operational for temporal-graph predictions through prospective registration, a graph-wide filtration for shared predictors and histories, an explicit opportunity law, and random-null analysis for adaptively selected candidate identities.

2 RELATED WORK

Prediction and graph inference. Temporal Graph Networks (TGN) and recent benchmarks develop chronological prediction with recurrent memories and heterogeneous time features (Rossi et al., 2020; Huang et al., 2023; Gastinger et al., 2024). TimeHet combines absolute and pairwise-relative time, relation-wise aggregation, recurrent states, and graph-token alignment for prediction (Wang et al., 2026). Lagged-THG is a graph-only, pre-destination event-stream adaptation: it omits the Graph2Token and language-model components, uses only completed events, and only nominates sets and parameterizes future bets. The e-process construction and FDR guarantees do not derive from TimeHet. Our TGN-style comparator shares its state-update and prediction heads and is an architectural ablation, not an official TGN reimplementation. Conformal graph methods instead target missing-edge truth, prediction-set coverage, or individual anomalous edges under exchangeability or related assumptions (Marandon, 2024; Blanchard et al., 2024; Du et al., 2026; Davis et al., 2025; Wang et al., 2025b; Hu et al., 2026; Chen et al., 2026; Lunde, 2023; Luo et al., 2023). GRAPHBET tests whether a nominated set receives excess future interaction mass relative to a stated opportunity law.

Risk sets and sequential evidence. Relational-event models provide dynamic risk sets and history-dependent event intensities (Bianchi et al., 2024; Lerner & Lomi, 2020; Filippi-Mazzola & Wit, 2024). Safe likelihood ratios, Model-X and pairwise betting, and conformal martingales provide sequential evidence (Grünwald et al., 2021; Shaer et al., 2023; Saha & Ramdas, 2024), while graph martingales have detected global structural changes (Ho et al., 2026). The closest temporal-network use of

e-values averages snapshot-level e-values for one global community-structure hypothesis (Yanchenko et al., 2026). In contrast, GRAPHBET registers many destination sets, normalizes event-level factors by pre-destination opportunity mass, and tests simultaneous, adaptively selected enrichments.

Multiple testing with e-values. E-BH controls FDR under arbitrary dependence, while online, stopped, closure, and feedback variants address other forms of adaptivity (Wang & Ramdas, 2022; Xu et al., 2021; Xu & Ramdas, 2024; Wang et al., 2025a; Fischer et al., 2024; Tavyrikov et al., 2026; Xu et al., 2026; Lu et al., 2025). Our graph-wide filtration establishes global validity despite shared histories and predictors. Unlike e-BH applied to coordinatewise running maxima, which can be invalid, our accumulated procedure unions hypotheses actually selected at contemporaneous e-BH snapshots and uses their rejection counts at first entry (Tavyrikov et al., 2026). Fixed slots make registration and the multiplicity denominator explicit.

3 PROBLEM SETUP

Two-stage event filtration. Each event $Z_t = (C_t, V_t)$ is revealed in two stages: the source–relation context $C_t = (U_t, R_t)$ first, then destination V_t. Let $\mathcal{F}_{t-1}$ contain the complete graph-wide information after event $t-1$, including candidate decisions, algorithmic randomness, resolved delays, and current model parameters. Define the pre-destination field

$$\mathcal{H}_t = \mathcal{F}_{t-1} \vee \sigma(C_t, \xi_t),$$

where ξ_t is any declared randomization exposed before V_t. Let ω_t denote randomness revealed atomically with V_t and set $\mathcal{F}_t = \mathcal{H}_t \vee \sigma(V_t, \omega_t)$. Conditional on the opportunity state, internally generated ξ_t is independent of V_t. Risk sets, reference probabilities, neural predictions, and bets must be $\mathcal{H}_t$-measurable; realized factors are $\mathcal{F}_t$-measurable. Tied timestamps use a destination-blind precommitted order and the corresponding sequential conditional laws (Appendix G).

For context $c = (u, r)$, the $\mathcal{H}_t$-measurable risk set $\mathcal{R}_t(c)$ contains scientifically possible destinations, and predictable weights $a_t(v, C_t)$ encode their relative opportunities. The reference distribution is

$$q_{0,t}(v \mid \mathcal{H}_t) = \frac{a_t(v, C_t)\mathbf{1}\{v \in \mathcal{R}_t(C_t)\}}{\sum_{w \in \mathcal{R}_t(C_t)} a_t(w, C_t)}, \qquad a_t(v, C_t) \geq 0, \tag{1}$$

with a finite, positive denominator. Let $q_t(v \mid \mathcal{H}_t) = \mathbb{P}(V_t = v \mid \mathcal{H}_t)$ denote the true conditional destination law. Section 4.2 derives (1) from a design. An arbitrary fitted propensity is not an exact design law and instead requires sensitivity analysis.

Adaptive candidate slots. Fix a deterministic budget of K slots. Slot j is born at an extended $\mathcal{F}_t$-stopping time $B_j \in \mathbb{N} \cup \{\infty\}$; a finite birth freezes an $\mathcal{F}_{B_j}$-measurable context c_j, destination set A_j, and predictable retirement rule ρ_j giving stopping time $D_j > B_j$. Evidence begins strictly after birth. Define

$$I^{\circ}_{j,t} = \mathbf{1}\{B_j < t \leq D_j,\ C_t = c_j\}, \qquad p_{0,j,t} = \begin{cases} q_{0,t}(A_j \cap \mathcal{R}_t(c_j) \mid \mathcal{H}_t), & C_t = c_j, \\ 1/2, & C_t \neq c_j, \end{cases} \tag{2}$$

For fixed $\varepsilon \in (0, 1/2)$, define the update and outcome by

$$I_{j,t} = I^{\circ}_{j,t}\mathbf{1}\{p_{0,j,t} \in [\varepsilon, 1-\varepsilon]\}, \qquad Y_{j,t} = \mathbf{1}\{V_t \in A_j \cap \mathcal{R}_t(c_j)\}.$$

The selected candidate's directional null is

$$H_j: \quad \mathbb{P}(Y_{j,t} = 1 \mid \mathcal{H}_t) \leq p_{0,j,t} \quad \text{at every update with } I_{j,t} = 1. \tag{3}$$

Rejecting (3) means that the frozen set attracts more future destination probability than its opportunity mass permits, not merely that one edge was predicted correctly. In the editor–page example, the activity level, available pages, and reference mass may all change with the observed past; only the next destination's conditional mass is restricted.

Because A_j is selected from the completed birth record $h^{\mathrm{obs}}_{B_j}$, its null status can depend on history. For law P, let $\mathcal{H}_0(P)$ contain birth records (h, c, A, ρ) for which (3) holds on every P-reachable continuation, and define

$$N_j(P) = \mathbf{1}\{(h^{\mathrm{obs}}_{B_j}, c_j, A_j, \rho_j) \in \mathcal{H}_0(P)\}.$$

Thus $N_j(P)$ is the realized slot's null label: random before birth and $\mathcal{F}_{B_j}$-measurable afterward. This random label is essential for adaptive candidate selection.

4 THE GRAPHBET METHOD

4.1 METHOD OVERVIEW

GRAPHBET has six operations (Figure 1). A predictor first uses the completed graph to register a context and freeze a destination set; pre-birth events never support that candidate. Before each later matching destination is revealed, the risk-set engine computes the set's reference probability and a predictable rule chooses a one-sided bet. The destination then updates evidence before graph memory or model state changes. After registration closes, e-BH reports over the fixed family and selected discoveries may be retained. Candidate generation, opportunity modeling, and betting are modular:

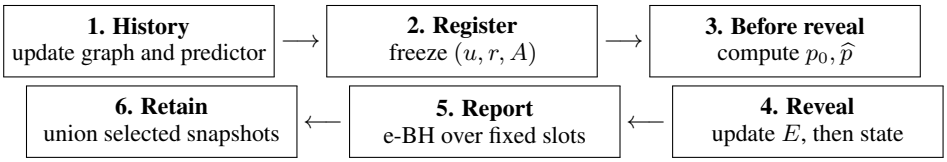

Figure 1: Information flow. The opportunity probability and bet precede the destination; a retained alert must first be selected by contemporaneous e-BH.

degree–recency, the TGN-style ablation, or Lagged-THG may nominate sets, while fixed, learned, or safely mixed bets share the same validity conditions.

4.2 A GRAPH-DERIVED OPPORTUNITY LAW

The primary exact regime is a dynamic typed-stub design. Immediately before the destination is drawn, eligible node v owns $d_t(v, r)$ opportunity stubs for relation r; compatibility, availability, capacity, and past updates are already known. Uniform selection of one compatible stub gives

$$q_{0,t}(v \mid \mathcal{H}_t) \quad \text{on } C_t = (u, r) = \frac{d_t(v, r)\mathbf{1}\{v \in \mathcal{R}_t(u, r)\}}{D_t(u, r)}, \qquad D_t(u, r) = \sum_{w \in \mathcal{R}_t(u,r)} d_t(w, r). \tag{4}$$

This covers uniform opportunity and typed activity matching. It is design-identified only when destination selection is randomized or enforced according to (4), or a fully specified and logged choice mechanism justifies (4); exposure and eligibility logs alone do not suffice. Otherwise, (4) defines a reference-relative null rather than observational truth. Appendix I distinguishes exact, externally bounded, and diagnostic uses. Appendix B also gives an exact sparse-normalizer implementation with sublinear queries; benchmark negative samples never enter the reference law.

4.3 PREDICTABLE CANDIDATE GENERATION AND BETTING

A temporal graph model reads $\mathcal{F}_{t-1}$, freezes a small destination set at birth, and on later matching events chooses $\beta_{j,t} \in [0, 1]$ from $\mathcal{H}_t$. To keep log factors bounded, define

$$(\eta_{j,t}, \widehat{p}_{j,t}) = \begin{cases} \left(\dfrac{1 - \varepsilon - p_{0,j,t}}{1 - p_{0,j,t}} \beta_{j,t},\ p_{0,j,t} + (1 - \varepsilon - p_{0,j,t})\beta_{j,t} \right), & I_{j,t} = 1, \\ (0, 1/2), & I_{j,t} = 0. \end{cases} \tag{6}$$

On an active update, $\widehat{p}_{j,t} = p_{0,j,t} + (1 - p_{0,j,t})\eta_{j,t} \in [p_{0,j,t}, 1 - \varepsilon]$. The event-level Bernoulli likelihood ratio is

$$L_{j,t} = \begin{cases} \left(\dfrac{\widehat{p}_{j,t}}{p_{0,j,t}} \right)^{Y_{j,t}} \left(\dfrac{1 - \widehat{p}_{j,t}}{1 - p_{0,j,t}} \right)^{1 - Y_{j,t}} = 1 + \eta_{j,t} \left(\dfrac{Y_{j,t}}{p_{0,j,t}} - 1 \right), & I_{j,t} = 1, \\ 1, & I_{j,t} = 0. \end{cases} \tag{7}$$

The one-sided constraint $\widehat{p}_{j,t} \geq p_{0,j,t}$ is essential: hits increase evidence and misses decrease it, while under (3) the conditional expected factor is at most one. A destination-specific neural factor requires the stronger exact law $q_t = q_{0,t}$ (Appendix D).

Starting from one, the candidate's evidence process is

$$E_j(t) = \prod_{s=1}^{t} L_{j,s}, \qquad E_j(0) = 1. \tag{8}$$

Formally, an e-value for a null class $\mathcal{P}_0$ is a nonnegative variable E satisfying $\sup_{P\in\mathcal{P}_0} \mathbb{E}_P(E) \le 1$, and an e-process remains an e-value at every stopping time. Consequently, for one null candidate, crossing $1/\alpha$ at any time has probability at most α. In testing-by-betting terminology, $E_j(t)$ is the candidate's wealth: it begins at one and compounds the return from each prospective bet. Theorem 2 proves this property for (8) despite adaptive graph updates and dependence among overlapping slots, because all bets are normalized in the same global pre-destination field.

Lagged-THG uses only states and neighborhoods from completed events. Relation-specific neighborhood aggregation, time features, and a 128-dimensional gated recurrent unit (GRU) feed separate heads for destination ranking and the later hit probability $\widetilde{q}_{j,t} = \widehat{\mathbb{P}}(Y_{j,t} = 1 \mid \mathcal{H}_t)$. Before each destination, GRAPHBET projects this probability to $\widehat{p}_{j,t} = \max\{p_{0,j,t}, \min(1-\varepsilon, \widetilde{q}_{j,t})\}$, derives $\beta_{j,t}$ through (6), and stores the factor before updating graph memory. Sensitivity paths reuse the same predictable raw $\beta_{j,t}$ and rebuild $\widehat{p}$ from their inflated reference; parameters and memory remain frozen within tied batches. Appendix H specifies the architecture, losses, and update order.

When one calibrated strength is unavailable, GRAPHBET splits initial wealth across prespecified fixed-β experts. Proposition 3 shows that this safe mixture remains an e-process and, on every path, loses at most $\log M$ in log wealth relative to the best of M experts. We use it for the strongest Wiki pipeline. Validation fixes all parameters before the primary test, and data-specific negative sampling never enters $p_{0,j,t}$ or $L_{j,t}$.

4.4 REPORTING DISCOVERIES

Fix a predeclared registration-closing stopping time σ. Thereafter births stop and every unfilled slot is permanently padded as a null dummy with wealth one. At any common global stopping time $\tau > \sigma$, apply e-BH at level α to $E_1(\tau), \dots, E_K(\tau)$. If $E_{(1)} \ge \cdots \ge E_{(K)}$, let

$$\widehat{k} = \max\left\{ k \in \{1, \dots, K\} : E_{(k)} \ge \frac{K}{\alpha k} \right\} \tag{9}$$

Set $\widehat{k} = 0$ and $\widehat{\mathcal{D}}(\tau) = \varnothing$ if (9)'s set is empty; otherwise reject slots with $E_j(\tau) \ge K/(\alpha\widehat{k})$, yielding $\widehat{\mathcal{D}}(\tau)$. For retained alerts define $\widehat{\mathcal{U}}(\tau) = \bigcup_{s=\sigma+1}^{\tau} \widehat{\mathcal{D}}(s)$. The results below control slot-level FDR for both sets. This union contains only actual contemporaneous e-BH selections; e-BH applied instead to coordinatewise running maxima can combine asynchronous peaks and is not generally valid (Tavyrikov et al., 2026). Padding fixes the denominator independently of random births but can be conservative; replacing K by the born count is not covered.

5 THEORY

Proposition 1 (Typed-stub conditional law)**.** *Conditional on $\mathcal{H}_t$, if the destination stub is uniform over the $D_t(u, r)$ compatible stubs, then the destination distribution is (4), even when stub counts, availability, contexts, and candidate births adapt to the complete past graph.*

The proof conditions on the realized global history and counts compatible stubs (Appendix B); observational use instead requires a design justification or the sensitivity formulation below.

Theorem 2 (Adaptive-birth global validity)**.** *Fix a born slot j. If birth is an $\mathcal{F}_t$-stopping time, registered metadata are $\mathcal{F}_{B_j}$-measurable, $I_{j,t}, p_{0,j,t}, \beta_{j,t}, \eta_{j,t}, \widehat{p}_{j,t}$ are $\mathcal{H}_t$-measurable, and $Y_{j,t}, L_{j,t}$ are $\mathcal{F}_t$-measurable, then $\{N_j(P)E_j(t) : t \ge B_j\}$ is a nonnegative supermartingale initialized at $N_j(P)$. Hence, for every almost surely finite global stopping time $\tau \ge B_j$,*

$$\mathbb{E}_P[N_j(P)E_j(\tau)] \le \mathbb{P}_P\{N_j(P) = 1\}. \tag{10}$$

If $N_j(P) = 1$ almost surely, E_j is a global e-process. No independence is required among slots, nodes, relations, candidate births, or predictable model updates.

The proof uses $\mathbb{E}_P(L_{j,t} \mid \mathcal{H}_t) \leq 1$ under (3). A dyad-local history is insufficient for a globally informed predictor; Appendix F gives a counterexample, and Corollary 11 in Appendix B gives the time-uniform Ville bound.

Proposition 3 (Safe expert aggregation)**.** *For wealth processes satisfying Theorem 2 for the same slot, null, and opportunity law, nonnegative $\mathcal{F}_{B_j}$-measurable weights summing to one give the valid mixture $E_j^{\mathrm{mix}} = \sum_\ell w_{j,\ell} E_{j,\ell}$, with pathwise bound*

$$\log E_j^{\mathrm{mix}}(t) \geq \max_{\ell : w_{j,\ell} > 0} \{\log E_{j,\ell}(t) + \log w_{j,\ell}\}.$$

Thus uniform mixing is within $\log M$ of the best of M experts.

Corollary 4 (FDR for adaptive candidate identities)**.** *If registration closes at $\sigma < \tau$, every born slot satisfies Theorem 2, and unfilled slots are permanently padded at one, then e-BH (9) at any common post-registration global stopping time satisfies*

$$\mathbb{E}_P\left[\frac{\sum_{j=1}^K N_j^\sigma(P)\mathbf{1}\{j \text{ rejected}\}}{|\widehat{\mathcal{D}}(\tau)| \vee 1}\right] \leq \alpha \tag{11}$$

for the padded null statuses defined in Appendix C.

Theorem 5 (FDR for accumulated snapshot discoveries)**.** *Under Corollary 4, suppose global time is discrete and slots are not reused after registration closes. For $\widehat{\mathcal{U}}(t) = \bigcup_{s=\sigma+1}^t \widehat{\mathcal{D}}(s)$ and every almost surely finite global stopping time $\tau > \sigma$,*

$$\mathbb{E}_P\left[\frac{\sum_{j=1}^K N_j^\sigma(P)\mathbf{1}\{j \in \widehat{\mathcal{U}}(\tau)\}}{|\widehat{\mathcal{U}}(\tau)| \vee 1}\right] \leq \alpha. \tag{12}$$

Thus alerts first selected by contemporaneous e-BH may be retained with slot-level FDR control.

Theorem 5 does not justify e-BH on coordinatewise running maxima, which may combine asynchronous peaks that were never jointly selected (Tavyrikov et al., 2026).

Proposition 6 (Sensitivity to opportunity-law error)**.** *Fix a predeclared predictable path $\Gamma_{j,t} \geq 1$, inflate the reference to $\widetilde{p}_{0,j,t}^\Gamma = \min\{\Gamma_{j,t} p_{0,j,t}, 1\}$, and recompute eligibility and the alternative probability from $\widetilde{p}_{0,j,t}^\Gamma$ using the same stored predictable $\beta_{j,t}$. If $\mathbb{P}(Y_{j,t} = 1 \mid \mathcal{H}_t) \leq \widetilde{p}_{0,j,t}^\Gamma$ at every resulting active update, then the null-weighted robust wealth is a nonnegative supermartingale; after permanent padding, snapshot and accumulated e-BH retain slot-level FDR control at level α for that Γ path.*

A sensitivity factor must come from design, external knowledge, or a declared analysis grid. A validation-derived anchor is diagnostic unless it is a genuine pointwise bound. Appendix D gives the exact factor construction and proof.

Theorem 7 (Graph signal minus neural prediction regret)**.** *Assume $t_{j,1}, \ldots, t_{j,m} < \infty$ almost surely, where $t_{j,k}$ is candidate j's kth active-update time. Write $L_k, p_{0,k}, \widehat{p}_k$, and $p_k = \mathbb{P}(Y = 1 \mid \mathcal{H})$ for the corresponding factor, reference, bet probability, and true conditional probability. If $p_{0,k}, \widehat{p}_k \in [\varepsilon, 1-\varepsilon]$ and $\widehat{p}_k \geq p_{0,k}$, define*

$$S_m = \sum_{k=1}^m \mathrm{KL}(\mathrm{Ber}(p_k) \,\|\, \mathrm{Ber}(p_{0,k}))\,, \qquad R_m = \sum_{k=1}^m \mathrm{KL}(\mathrm{Ber}(p_k) \,\|\, \mathrm{Ber}(\widehat{p}_k))\,. \tag{13}$$

Here $\mathrm{KL}(\cdot\|\cdot)$ denotes Kullback–Leibler (KL) divergence. Then $\sum_{k=1}^m \mathbb{E}(\log L_k \mid \mathcal{H}) = S_m - R_m$, and with probability at least $1-\delta$,

$$\log E_j(t_{j,m}) \geq S_m - R_m - c_\varepsilon \sqrt{2m \log(1/\delta)}, \qquad c_\varepsilon = \log \frac{1-\varepsilon}{\varepsilon}. \tag{14}$$

Proposition 8 (Prediction-to-signal transfer)**.** *Suppose the first m active updates share a finite risk set $\mathcal{R}$, the reference is uniform, and candidates have fixed size a, so $p_0 = a/|\mathcal{R}|$. Let π be the registration-time destination forecast and $\widehat{A} \in \arg\max_{|A|=a} \pi(A)$. For any size-$a$ comparator $A^\star$*

and future destination laws $q_1, \dots, q_m$, if the total variation (TV) distance satisfies $\mathrm{TV}(q_k, \pi) \le \delta_k$ *and* $q_k(A^\star) - 2\delta_k \ge p_0$*, then*

$$q_k(\widehat{A}) \ge q_k(A^\star) - 2\delta_k, \qquad S_m(\widehat{A}) \ge \sum_{k=1}^{m} \mathrm{KL}\big\{\mathrm{Ber}\big(q_k(A^\star) - 2\delta_k\big) \,\big\|\, \mathrm{Ber}(p_0)\big\}.$$

If $\mathrm{KL}(q_k \| \pi) \le \epsilon_k$*, one may take* $\delta_k = \sqrt{\epsilon_k/2}$*.*

Corollary 9 (Finite-sample active-update detection time)**.** *Let* $\nu_h = \inf\{m : E_j(t_{j,m}) \ge h\}$ *for* $h > 1$*. If the kth active update occurs almost surely whenever* $\nu_h \ge k$*, and* $\mathbb{E}(\log L_k \mid \mathcal{H}_{t_{j,k}}) \ge \gamma > 0$ *before crossing, then with*

$$m_\delta(h, \gamma) = \left\lceil \frac{2 \log h}{\gamma} + \frac{2 c_\varepsilon^2 \log(1/\delta)}{\gamma^2} \right\rceil, \qquad \mathbb{P}\{\nu_h \le m_\delta(h, \gamma)\} \ge 1 - \delta, \qquad \mathbb{E}(\nu_h) \le \frac{\log h + c_\varepsilon}{\gamma}. \tag{14b}$$

For a finite lifetime with M_j *eligible active updates and* $\nu_h^{\mathrm{op}} = \inf\{k \le M_j : E_j(t_{j,k}) \ge h\}$*,*

$$\mathbb{P}\{\nu_h^{\mathrm{op}} > m_\delta(h, \gamma)\} \le \delta + \mathbb{P}\{M_j < m_\delta(h, \gamma)\}. \tag{14c}$$

Thus a matching-opportunity horizon alone is insufficient unless it supplies the required active updates. A directly checkable sufficient condition for the drift assumption is $p_k - p_{0,k} \ge \Delta > 0$ *and* $\mathrm{KL}\{\mathrm{Ber}(p_k) \| \, \mathrm{Ber}(\widehat{p}_k)\} \le r < 2\Delta^2$ *before crossing, which gives* $\gamma = 2\Delta^2 - r$*.*

Theorem 7 separates enrichment signal from betting regret; Proposition 8 makes the signal explicit under forecast stability, and Corollary 9 converts positive drift into detection time. At a common report, one slot crosses e-BH at K/α, or k slots cross jointly at $K/(\alpha k)$. Appendix E provides full proofs and the sharper finite-lifetime drift condition.

6 Experiments

Four studies examine complementary questions: controlled heterogeneous streams assess FDR and power under a known opportunity law; exact-law benchmark replays check the sequential implementation on realistic event skeletons; observational replays diagnose a proposed activity reference; and a crossed fit-by-replay Wiki study separates candidate generation from betting. All discovery, false discovery proportion (FDP), conditional testing power, and end-to-end metrics use the same predeclared final snapshot $\widehat{\mathcal{D}}(T)$. Appendices H–J give full mechanisms, seeds, metrics, leakage checks, and compute details.

6.1 Controlled temporal heterogeneous graphs

The controlled streams contain three node types and relations, dynamic typed opportunities, preferential source activity, and a latent two-state regime. For selected source–relation contexts, the destination distribution is tilted toward a hidden five-node block $A_c^\star$:

$$q_{\theta,t}(v \mid c) \propto q_{0,t}(v \mid \mathcal{H}_t) \exp\{\theta_c \mathbf{1}(v \in A_c^\star)\}, \qquad \theta_c \in \{0, \log(1.5), \log(2), \log(4)\}. \tag{15}$$

Here $\theta_c = 0$ is null. After 20 matching events, a predictor freezes its top typed block; disjoint libraries make it nonnull exactly when it equals $A_c^\star$ and $\theta_c > 0$, giving analytic labels.

With $|\mathcal{V}| = 500$, size-five candidates, $K = 200$, and 250 active updates, neither degree–recency nor the hidden-block log-growth oracle makes a false discovery in 300 global-null streams or 150 mixed streams at $\theta = \log(4)$. Their nomination/end-to-end recalls are .027/.025 and .351/.330, respectively; conditional testing powers are .935 (116 eligible replications) and .941 (all 150). Thus the strong-signal end-to-end gap arises mainly at nomination. Table 1 gives the full prespecified power curve; Appendix H.4 reports uncertainty intervals and secondary diagnostics.

6.2 Real temporal interactions

We use version-2 `tgbl-wiki` and `tgbl-review` (Huang et al., 2023). A Temporal Graph Network (TGN)-style architectural ablation and Lagged-THG nominate five destinations; mean

Table 1: Controlled-stream final-snapshot power curve at $\alpha = 0.10$, with 150 mixed streams per signal and 10% enriched contexts. Test is conditional testing power and n_+ is its contributing-replication count.

Multiplier	Candidate and bet	Nom.	E2E	Test	n_+
1.5	Degree/recency, fixed	.0129	.0000	.0000	72
1.5	Hidden-block, log-growth oracle	.3509	.0001	.0003	150
2	Degree/recency, fixed	.0168	.0006	.0233	86
2	Hidden-block, log-growth oracle	.3509	.0567	.1601	150
4	Degree/recency, fixed	.0266	.0248	.9353	116
4	Hidden-block, log-growth oracle	.3509	.3305	.9406	150

reciprocal rank (MRR) measures tuning-set prediction, not inferential validity. Sequential analyses use $K = 5000$, $\alpha = .10$, a prespecified sensitivity grid, and validation-derived diagnostic anchor Γ_{a}. Because both datasets have one observed relation and no node-type field, they assess temporal representation and opportunity calibration rather than the full heterogeneous architecture. We use three training seeds per architecture and 100 exact-null replays per event skeleton for Lagged-THG.

A separate strong-signal Wiki study crosses three prespecified fits, four candidate–bet arms, and 20 held-out replay streams shared across arms, giving 240 fit–arm–replay cells. Crossed bootstrap intervals resample fit and replay identifiers separately. From the fixed-candidate records, an equal-weight mixture over six prespecified fixed betting fractions is evaluated without selecting or reweighting an expert from test outcomes.

Table 2: Observational diagnostics over three seeds. MRR is mean (sample SD); Born and $R_{1/a/1.5}$ are mean final-snapshot counts at $\Gamma = 1, \Gamma_{\mathrm{a}}, 1.5$. NC is the number of exact-law replays with any final rejection out of 100; n/a means not evaluated.

Data	Predictor	Tuning MRR	Born	Γ_{a}	NC	$R_{1/a/1.5}$
Wiki-v2	TGN-style	.8323 (.0019)	614	124.05	n/a	449.7/0.0/453.0
Wiki-v2	Lagged-THG	.8376 (.0015)	614	135.45	0/100	441.0/1.3/449.7
Review-v2	TGN-style	.1410 (.0723)	5000	4.35	n/a	0.0/0.0/0.0
Review-v2	Lagged-THG	.1711 (.0815)	5000	4.20	0/100	0.0/0.0/0.0

Wiki yields hundreds of reference-relative rejections at $\Gamma = 1$, whereas the diagnostic anchors 124.05 (TGN-style) and 135.45 (Lagged-THG) reduce their means to 0.0 and 1.3, revealing substantial mismatch in the simple activity reference; Review yields none at $\Gamma = 1$ or its anchor. Neither set of 100 exact-law Lagged-THG replays contains a final rejection (two-sided 95% Clopper–Pearson interval [.000,.036] for each). Thus exact-law replays check implementation under a known generating law, while observational sensitivity paths assess reference credibility. MRR comparisons are descriptive because architectures use disjoint initialization seeds.

Table 3: Wiki strong-signal component study at $\Gamma = 1$: final-snapshot means over three prespecified fits and 20 shared replays per fit. Nom., E2E, Test, Disc., and FDP are nomination recall, end-to-end recall, conditional power among born nonnull slots, e-BH discoveries, and false discovery proportion. Identical degree-candidate copies are collapsed across fit labels.

Candidate	Bet	Nom.	E2E	Test	Disc.	FDP
Degree/recency	Fixed	.208	.069	.335	4.00	.000
Degree/recency	Safe mixture	.208	.075	.366	4.35	.000
Lagged-THG	Fixed	.187	.077	.418	4.53	.008
Lagged-THG	Safe mixture	.187	**.084**	**.452**	**4.85**	.000

Holding the bet fixed, Lagged-THG candidates increase conditional power by .083 [.030,.140] and expected log growth per nonnull update by .059 [.042,.077], with both gains positive in every fit. The corresponding end-to-end contrast is .008 [$-.003$,.021]: positive in mean, although its crossed interval includes zero because nomination recall is .021 lower. Safe aggregation raises conditional power by .031 [.007,.056] and .034 [.007,.059], and end-to-end recall by .0060 [.0017,.0112] and

Table 4: Signal–regret audit on analytically nonnull Wiki updates, averaged over the crossed fit–replay design. Expected growth is $(S_m - R_m)/m$; realized growth is cumulative log factor divided by m. Identical model-free degree/fixed copies are collapsed.

Candidate	Bet	S_m/m	R_m/m	Expected	Realized
Degree/recency	Fixed	.501	.241	.261	.270
Degree/recency	Neural	.501	1.087	$-.586$	$-.544$
Lagged-THG	Fixed	.579	.260	.320	.327
Lagged-THG	Neural	.579	1.032	$-.453$	$-.422$

.0063 [.0020,.0109], for degree–recency and Lagged-THG candidates, respectively. The Lagged-THG/safe-mixture pipeline therefore attains the highest observed means, .452 conditional power and .084 end-to-end recall, with zero final-snapshot false discoveries. Relative to degree–recency/fixed, these values represent 35% higher conditional power and 21% higher end-to-end recall. Appendix I gives full crossed arm estimates and fit-level mixture results.

6.3 Where the gains arise

Table 4 connects the empirical gains to Theorem 7. Holding the bet fixed, Lagged-THG raises enrichment signal from .501 to .579 and increases expected and realized log growth by .059 [.042,.077] and .057 [.039,.074] per nonnull update. The transferred hit-probability head instead incurs enough regret to make expected growth negative for both candidate families. The safe mixture avoids selecting a fragile betting strength from evaluation outcomes while remaining within the pathwise $\log 6$ bound of the best prespecified expert. Thus the learned representation contributes stronger enrichment among correctly nominated sets, and safe aggregation converts that signal into discoveries. Appendix I reports the full paired contrasts.

7 Discussion

Link prediction ranks likely destinations; GraphBet asks the distinct prospective question of whether a frozen set subsequently receives more activity than its opportunities explain. This separation is modular: new temporal representations can improve candidate quality and predictable betting rules can improve detection efficiency without changing the evidence construction or graph-wide FDR argument. The crossed Wiki results show both sides—learned candidates carry more enrichment signal under a common bet, while direct transfer of an inadequately calibrated probability head can erase that advantage. Safe aggregation provides a principled alternative to choosing a betting strength from evaluation outcomes.

Predictive accuracy alone cannot justify the null comparison. The Wiki diagnostic shows that high ranking accuracy can coexist with severe mismatch of a simple activity reference. In applications, opportunity probabilities can be grounded in logged choice sets, randomized exposure, enforced assignment, or externally justified pointwise bounds; prespecified sensitivity paths then quantify bounded departures. The controlled study already exercises typed multi-relational streams, while combining learned heterogeneous candidates with defensible multi-relational opportunity mechanisms is a natural next step. Because FDR targets registered slots, deployments with duplicate scientific targets should enforce unique registration or a predeclared grouping rule.

8 Conclusion

GraphBet turns adaptive temporal-graph nominations into opportunity-adjusted sequential evidence. We establish graph-wide e-process validity, snapshot and retained FDR control, reference-error robustness, and finite-sample prediction-to-detection links. We observe no final-snapshot false discoveries in the tested exact-law nulls. Across three prespecified fits, learned candidates improve conditional power and evidence growth under a common fixed bet; safe aggregation is strongest, with .452 conditional power and .084 end-to-end recall versus .335 and .069 for degree–recency/fixed. This decomposition separates the scientific null, target nomination, and conversion of signal into discoveries.

Reproducibility statement

Section 3 states the filtration and null; Appendices A–E give the measurability checklist and proofs. Appendices H–J specify the data-generating mechanisms, parameter grid, preprocessing, leakage checks, logging schema, aggregation, and compute profile. Anonymous code accompanies the submission.

AI use statement

In accordance with the conference guidelines, we acknowledge the use of Large Language Models (LLMs) solely for language polishing and improving the clarity and readability of this manuscript. All research contributions, including the methodology, experiments, analysis, and scientific insights, were independently developed by the authors. LLMs did not contribute to the conceptual development, experimental validation, or interpretation of the results.

References

Federica Bianchi, Edoardo Filippi-Mazzola, Alessandro Lomi, and Ernst C Wit. Relational event modeling. *Annual Review of Statistics and Its Application*, 11, 2024.

Gilles Blanchard, Guillermo Durand, Ariane Marandon-Carlhian, and Romain Périer. FDR control and FDP bounds for conformal link prediction, 2024. URL https://arxiv.org/abs/2404.02542.

Xudong Chen, Shengbo Gong, Lu Cheng, and Wei Jin. Temporal graph prototype-conditioned conformal prediction for fraud detection. In *Proceedings of the 32nd ACM SIGKDD Conference on Knowledge Discovery and Data Mining V. 2*, pp. 484–495, 2026.

Ed Davis, Ian Gallagher, Daniel Lawson, and Patrick Rubin-Delanchy. Valid conformal prediction for dynamic GNNs. In Y. Yue, A. Garg, N. Peng, F. Sha, and R. Yu (eds.), *International Conference on Learning Representations*, volume 2025, pp. 77514–77538, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/file/c0ea9a95e243818f62004e73d57831ca-Paper-Conference.pdf.

Wenqin Du, Wanteng Ma, Dong Xia, Yuan Zhang, and Wen Zhou. Conformal network link prediction with false discovery rate control under unstructured missingness. *Journal of Computational and Graphical Statistics*, 2026. doi: 10.1080/10618600.2026.2719794. URL https://doi.org/10.1080/10618600.2026.2719794.

Edoardo Filippi-Mazzola and Ernst C Wit. Modeling non-linear effects with neural networks in relational event models. *Social networks*, 79:25–33, 2024.

Lasse Fischer, Ziyu Xu, and Aaditya Ramdas. An online generalization of the (e-) benjamini-hochberg procedure. *arXiv preprint arXiv:2407.20683*, 2024.

Julia Gastinger, Shenyang Huang, Mikhail Galkin, Erfan Loghmani, Ali Parviz, Farimah Poursafaei, Jacob Danovitch, Emanuele Rossi, Ioannis Koutis, Heiner Stuckenschmidt, et al. TGB 2.0: A benchmark for learning on temporal knowledge graphs and heterogeneous graphs. *Advances in neural information processing systems*, 37:140199–140229, 2024.

Peter Grünwald, Alexander Ly, Muriel Perez-Ortiz, and Judith Ter Schure. The safe logrank test: Error control under optional stopping, continuation and prior misspecification. In *Survival Prediction-Algorithms, Challenges and Applications*, pp. 107–117. PMLR, 2021.

Shen-Shyang Ho, Tarun Teja Kairamkonda, and Izhar Ali. Detecting and explaining structural changes in an evolving graph using a martingale. *Pattern Recognition*, 169:111855, 2026.

Cheng Hu, Cong Cao, Fangfang Yuan, Diandian Guo, Pin Xu, Yu Liu, and Yanbing Liu. Conformal event prediction with temporal knowledge graph. In Maria Liakata, Viviane P. Moreira, Jiajun Zhang, and David Jurgens (eds.), *Findings of the Association for Computational Linguistics: ACL 2026*, pp. 5233–5248, San Diego, California, United States, July 2026. Association for Computational Linguistics. ISBN 979-8-89176-395-1. doi: 10.18653/v1/2026.findings-acl.258. URL https://aclanthology.org/2026.findings-acl.258/.

Shenyang Huang, Farimah Poursafaei, Jacob Danovitch, Matthias Fey, Weihua Hu, Emanuele Rossi, Jure Leskovec, Michael Bronstein, Guillaume Rabusseau, and Reihaneh Rabbany. Temporal graph benchmark for machine learning on temporal graphs. *Advances in Neural Information Processing Systems*, 36:2056–2073, 2023.

Jürgen Lerner and Alessandro Lomi. Reliability of relational event model estimates under sampling: How to fit a relational event model to 360 million dyadic events. *Network Science*, 8(1):97–135, 2020.

Lin Lu, Yuyang Huo, Haojie Ren, Zhaojun Wang, and Changliang Zou. Feedback-enhanced online multiple testing with applications to conformal selection. *arXiv preprint arXiv:2509.03297*, 2025.

Robert Lunde. On the validity of conformal prediction for network data under non-uniform sampling, 2023. URL https://arxiv.org/abs/2306.07252.

Rui Luo, Buddhika Nettasinghe, and Vikram Krishnamurthy. Anomalous edge detection in edge exchangeable social network models. In Harris Papadopoulos, Khuong An Nguyen, Henrik Boström, and Lars Carlsson (eds.), *Proceedings of the Twelfth Symposium on Conformal and Probabilistic Prediction with Applications*, volume 204 of *Proceedings of Machine Learning Research*, pp. 287–310. PMLR, 13–15 Sep 2023. URL https://proceedings.mlr.press/v204/luo23a.html.

Ariane Marandon. Conformal link prediction for false discovery rate control. *TEST*, 33(4): 1062–1083, 2024. doi: 10.1007/s11749-024-00934-w. URL https://doi.org/10.1007/s11749-024-00934-w.

Aaditya Ramdas, Peter Grünwald, Vladimir Vovk, and Glenn Shafer. Game-theoretic statistics and safe anytime-valid inference. *Statistical Science*, 38(4):576–601, 2023.

Emanuele Rossi, Ben Chamberlain, Fabrizio Frasca, Davide Eynard, Federico Monti, and Michael Bronstein. Temporal graph networks for deep learning on dynamic graphs. *arXiv preprint arXiv:2006.10637*, 2020.

Aytijhya Saha and Aaditya Ramdas. Testing exchangeability by pairwise betting. In *International Conference on Artificial Intelligence and Statistics*, pp. 4915–4923. PMLR, 2024.

Shalev Shaer, Gal Maman, and Yaniv Romano. Model-X sequential testing for conditional independence via testing by betting. In *International Conference on Artificial Intelligence and Statistics*, pp. 2054–2086. PMLR, 2023.

Yury Tavyrikov, Jelle J Goeman, and Rianne de Heide. Carefree multiple testing with e-processes. *Electronic Journal of Statistics*, 20(2):3178–3189, 2026.

Hongjian Wang, Sanjit Dandapanthula, and Aaditya Ramdas. Anytime-valid FDR control with the stopped e-BH procedure. *Statistics & Probability Letters*, pp. 110512, 2025a.

Ruodu Wang and Aaditya Ramdas. False discovery rate control with e-values. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 84(3):822–852, 2022.

Tingting Wang, Feng Ding, Yingzhou Li, Shuo Yu, Xiang Zhou, Ruolin Li, and Xin Ye. Temporal heterogeneous graph tokenization. In *2026 IEEE International Conference on Data Mining*, 2026. Accepted for the research track; proceedings metadata not yet assigned as of 30 August 2026.

Tuo Wang, Jian Kang, Yujun Yan, Adithya Kulkarni, and Dawei Zhou. Non-exchangeable conformal prediction for temporal graph neural networks. In *Proceedings of the 31st ACM SIGKDD Conference on Knowledge Discovery and Data Mining V.2*, KDD '25, pp. 3031–3042, New York, NY, USA, 2025b. Association for Computing Machinery. ISBN 9798400714542. doi: 10.1145/3711896.3737064. URL https://doi.org/10.1145/3711896.3737064.

Ziyu Xu and Aaditya Ramdas. Online multiple testing with e-values. In *International Conference on Artificial Intelligence and Statistics*, pp. 3997–4005. PMLR, 2024.

Ziyu Xu, Ruodu Wang, and Aaditya Ramdas. A unified framework for bandit multiple testing. *Advances in Neural Information Processing Systems*, 34:16833–16845, 2021.

Ziyu Xu, Lasse Fischer, and Aaditya Ramdas. Improving online FDR procedures via online analogs of e-closure and compound e-values. *arXiv preprint arXiv:2603.24792*, 2026.

Eric Yanchenko, Jonathan P. Williams, and Ryan Martin. Hypothesis testing for community structure in temporal networks using e-values. *Network Science*, 14:e14, 2026. doi: 10.1017/nws.2026.10036. URL https://doi.org/10.1017/nws.2026.10036.

A Notation and measurability checklist

Table 5 summarizes the information constraint required by the theory; its guarantee applies only when every row is satisfied.

Table 5: Information allowed at each stage of event t.

Object	Must be measurable with respect to	Excluded information
Candidate birth and frozen set	$\mathcal{F}_{B_j}$	Any event after birth
Risk set and baseline weights	$\mathcal{H}_t$	Destination V_t and later tied events
Neural state and stake	$\mathcal{H}_t$	Current destination, target label, future graph
Betting factor	$\mathcal{F}_t$	Later events
Model update	$\mathcal{F}_t$ after factor logging	Unresolved or future outcomes
Reporting time	Global $\mathcal{F}_t$ stopping time	Future events

A.1 Operational interpretation

The two-stage filtration is an implementation contract, not only proof notation. At the start of event t, $\mathcal{F}_{t-1}$ contains the complete graph, node entry and exit, previous candidate births and retirements, resolved delays, algorithmic randomness already exposed, and current model parameters. The context C_t and any declared pre-destination randomization ξ_t then enlarge this information to $\mathcal{H}_t$. Risk sets, opportunity probabilities, candidate summaries, neural predictions, and bet strengths must all be computed from this field. The destination V_t, its hit indicator, and any auxiliary randomization revealed with it belong only to $\mathcal{F}_t$. This separation prevents the current destination, or a destination-derived state update, from leaking into its own factor while still permitting arbitrary dependence on the complete earlier graph.

Operationally, each candidate follows a fixed sequence. First, registration uses only a completed event or completed tied batch to allocate the lowest available slot, freeze (c_j, A_j, ρ_j), and set B_j; event B_j never contributes evidence. Second, on each later matching context, the risk-set engine computes $p_{0,j,t}$ before the destination is known. Third, a fixed rule or frozen neural head chooses a one-sided alternative and the revealed hit or miss multiplies wealth through (7). Only after every applicable factor has been stored may graph memory and model state change. Fourth, births close at the prespecified time σ, unused slots become permanent null dummies with wealth one, and every report applies e-BH to the same fixed family. A retained alert enters the accumulated list only after appearing in an actual contemporaneous e-BH selection.

The opportunity model and learned model play different scientific roles. The opportunity law defines the null comparison and must come from a randomized or enforced design, complete logging of the choice mechanism, or an explicitly reference-relative analysis. Candidate and betting models may be highly adaptive because they affect which alternatives are useful rather than the null normalization, provided their outputs are predictable. Better destination ranking can therefore improve the enrichment of frozen candidates, and better betting calibration can convert that enrichment into evidence more efficiently, but neither repairs an unjustified opportunity law. This distinction is why benchmark negative samples are used for predictive training and evaluation but never enter $p_{0,j,t}$ or a betting factor.

For a tied batch $\mathcal{B}_b$, augment the pre-batch field with all contexts and an independently randomized order before any destination is revealed. Destinations are then processed in that order using the correct conditional opportunity law given earlier marks in the same batch. Neural parameters remain frozen, every factor is recorded sequentially, and training occurs only after the batch closes.

B Proofs for the opportunity law and e-process

B.1 Proof of Proposition 1

Proof. Conditional on $\mathcal{H}_t$, the compatible receiver stubs form a finite disjoint union with $d_t(v, r)$ stubs owned by node v and total size $D_t(u, r)$. Uniform selection assigns probability $1/D_t(u, r)$ to each stub. Summing over the stubs owned by v gives $d_t(v, r)/D_t(u, r)$ for eligible v, and zero otherwise, which is (4). All adaptively updated quantities are fixed after conditioning on $\mathcal{H}_t$, so their dependence on the earlier graph does not change the counting argument. □

B.2 Exact normalizer complexity

Let $\mathcal{U}_t(a, r)$ be the currently available destinations with a type compatible with source type a and relation r, including restrictions common to that type pair. Suppose source-specific restrictions form a sparse set $X_t(u, r) \subseteq \mathcal{U}_t(\text{type}(u), r)$, $\mathcal{R}_t(u, r) = \mathcal{U}_t(\text{type}(u), r) \setminus X_t(u, r)$, and $S_t(a, r) = \sum_{v \in \mathcal{U}_t(a,r)} d_t(v, r)$. Then

$$p_{0,j,t} = \frac{\sum_{v \in A_j \cap \mathcal{R}_t(c_j)} d_t(v, r_j)}{S_t(\text{type}(u_j), r_j) - \sum_{v \in X_t(u_j, r_j)} d_t(v, r_j)}. \tag{5}$$

Proposition 10. *Suppose $\mathcal{R}_t(u, r) = \mathcal{U}_t(\text{type}(u), r) \setminus X_t(u, r)$ under the definitions preceding (5), and $|X_t(u, r)| \le x$. Maintain $S_t(a, r)$ and a keyed value $d_t(v, r)$ for each node. A node or weight change costs $O(\log |\mathcal{V}_t|)$, and (5) costs $O((|A_j| + x) \log |\mathcal{V}_t|)$ without enumerating $\mathcal{R}_t(u, r)$.*

Proof. The maintained type–relation total supplies the first denominator term. Querying and subtracting the at most x excluded weights supplies the exact denominator. Querying the weights of candidate nodes still present in the risk set supplies the numerator. Balanced trees or Fenwick indices give the stated update and lookup costs. The ratio is algebraically identical to (4), so the computation introduces no approximation. □

B.3 Proof of Theorem 2

Proof. Write $M_{j,t} = L_{j,t}$. When $I_{j,t} = 0$, $M_{j,t} = 1$. On $N_j(P)$ and $I_{j,t} = 1$, let $p_{j,t} = \mathbb{P}_P(Y_{j,t} = 1 \mid \mathcal{H}_t) \le p_{0,j,t}$. Conditional on $\mathcal{H}_t$,

$$\mathbb{E}_P(L_{j,t} \mid \mathcal{H}_t) = p_{j,t} \frac{\widehat{p}_{j,t}}{p_{0,j,t}} + (1 - p_{j,t}) \frac{1 - \widehat{p}_{j,t}}{1 - p_{0,j,t}}. \tag{B.1}$$

The right side is affine in $p_{j,t}$, with slope $(\widehat{p}_{j,t} - p_{0,j,t})/\{p_{0,j,t}(1 - p_{0,j,t})\} \ge 0$. Its maximum over $p_{j,t} \le p_{0,j,t}$ is therefore attained at $p_{0,j,t}$, where it equals one. Thus

$$N_j(P)\mathbb{E}_P(M_{j,t} \mid \mathcal{H}_t) \le N_j(P). \tag{B.2}$$

For $t > B_j$, the null-status event and wealth accumulated through time $t - 1$ are measurable before the current mark. The tower property gives

$$\mathbb{E}_P[N_j(P)E_j(t) \mid \mathcal{F}_{t-1}] \le N_j(P)E_j(t - 1). \tag{B.3}$$

Thus the null-weighted wealth, started at the random time B_j, is a nonnegative supermartingale with initial value $N_j(P)$. Optional sampling in the shifted filtration gives (10) for bounded $\tau \ge B_j$. For an almost surely finite unbounded τ, apply the bounded result to $\tau \wedge (B_j + m)$ and use Fatou's lemma. Factors after expiration are one. Shared events and model parameters do not enter the proof beyond $\mathcal{H}_t$-measurability. □

Corollary 11 (Time-uniform candidate evidence)**.** *Under Theorem 2, for every $\delta \in (0, 1)$,*

$$\mathbb{P}_P\left\{ N_j(P) = 1, \sup_{t \ge B_j} E_j(t) \ge \delta^{-1} \right\} \le \delta\, \mathbb{P}_P\{N_j(P) = 1\} \le \delta.$$

B.4 Proofs of Corollary 11 and Proposition 3

Proof. On the event $N_j(P) = 1$, the null-weighted process in Theorem 2 equals E_j; off that event it is zero. Ville's inequality for a nonnegative supermartingale started at $N_j(P)$ gives

$$\mathbb{P}_P\left\{ N_j(P) = 1, \sup_{t \geq B_j} E_j(t) \geq \delta^{-1} \right\} \leq \delta \mathbb{E}_P\{N_j(P)\},$$

which proves Corollary 11.

For Proposition 3, the weights are fixed at birth and hence remain measurable before every later update. A nonnegative birth-measurable weighted sum of the expert null-weighted supermartingales is again a nonnegative supermartingale, with initial value $N_j(P) \sum_\ell w_{j,\ell} = N_j(P)$. Optional sampling and the proof of Corollary 4 therefore apply without change. Finally, $E_j^{\mathrm{mix}}(t) \geq w_{j,\ell} E_{j,\ell}(t)$ for every ℓ; taking logs and maximizing proves the pathwise bound. □

C Proof of false discovery control

Proof of Corollary 4. For completeness, define the padded null status and wealth by

$$N_j^\sigma(P) = \begin{cases} N_j(P), & B_j \leq \sigma, \\ 1, & B_j > \sigma, \end{cases} \qquad E_j(t) = 1 \text{ on } \{B_j > \sigma\}.$$

For a born slot, $\{B_j \leq \sigma\} \in \mathcal{F}_{B_j}$ in discrete time. Applying the proof of (10) after multiplying its initial value by this event gives

$$\mathbb{E}_P[\mathbf{1}\{B_j \leq \sigma\} N_j(P) E_j(\tau)] \leq \mathbb{P}_P\{B_j \leq \sigma, N_j(P) = 1\}.$$

Consequently, for each slot the padded process $S_j^\sigma(t) = N_j^\sigma(P) E_j(t)$, $t \geq \sigma$, is a nonnegative supermartingale: it is constant for a padded slot, and for a born slot this follows from Theorem 2 after multiplication by the birth-measurable event $\{B_j \leq \sigma\}$. In particular,

$$\mathbb{E}_P\{N_j^\sigma(P) E_j(\tau)\} \leq \mathbb{P}_P\{B_j > \sigma\} + \mathbb{P}_P\{B_j \leq \sigma, N_j(P) = 1\} = \mathbb{E}_P\{N_j^\sigma(P)\} \leq 1.$$

Let $R = |\widehat{\mathcal{D}}(\tau)|$ and let $\mathrm{FDP}(\tau)$ denote the false discovery proportion (FDP). For every rejected slot, the e-BH step-up rule implies $E_j(\tau) \geq K/(\alpha(R \vee 1))$. Pointwise,

$$\mathrm{FDP}(\tau) = \frac{\sum_j N_j^\sigma(P) \mathbf{1}\{j \text{ rejected}\}}{R \vee 1} \leq \frac{\alpha}{K} \sum_{j=1}^K N_j^\sigma(P) E_j(\tau). \tag{C.1}$$

Taking expectations and applying the preceding slotwise inequality yields

$$\mathbb{E}_P\{\mathrm{FDP}(\tau)\} \leq \frac{\alpha}{K} \sum_{j=1}^K \mathbb{E}_P\{N_j^\sigma(P)\} \leq \alpha.$$

No assumption on cross-slot dependence is used. The stopping-time closure σ makes the padding decision part of $\mathcal{F}_\sigma$, and no padded slot can subsequently be activated. Permanently unfilled slots are therefore the null dummies specified in Corollary 4; their wealth remains one. □

Proof of Theorem 5. For slot j, define its first post-registration entry time $\zeta_j = \inf\{t > \sigma : j \in \widehat{\mathcal{D}}(t)\}$, with $\zeta_j = \infty$ if it never enters. Because each contemporaneous e-BH set is determined by the global history, ζ_j is a global stopping time. On $\{\zeta_j \leq \tau\}$, let $R_j = |\widehat{\mathcal{D}}(\zeta_j)|$. E-BH self-consistency and $\widehat{\mathcal{D}}(\zeta_j) \subseteq \widehat{\mathcal{U}}(\tau)$ give

$$\frac{1}{|\widehat{\mathcal{U}}(\tau)|} \leq \frac{1}{R_j} \leq \frac{\alpha}{K} E_j(\zeta_j). \tag{C.2}$$

Therefore,

$$\frac{\sum_{j=1}^K N_j^\sigma(P) \mathbf{1}\{j \in \widehat{\mathcal{U}}(\tau)\}}{|\widehat{\mathcal{U}}(\tau)| \vee 1} \leq \frac{\alpha}{K} \sum_{j=1}^K N_j^\sigma(P) E_j(\zeta_j) \mathbf{1}\{\zeta_j \leq \tau\}. \tag{C.3}$$

The stopped time $\zeta_j \wedge \tau$ is almost surely finite. Since $S_j^\sigma(t) = N_j^\sigma(P)E_j(t)$ is a nonnegative supermartingale after registration closes,

$$\mathbb{E}_P\big[N_j^\sigma(P)E_j(\zeta_j)\mathbf{1}\{\zeta_j \le \tau\}\big] \le \mathbb{E}_P\big[S_j^\sigma(\zeta_j \wedge \tau)\big] \le \mathbb{E}_P\big[S_j^\sigma(\sigma)\big] \le \mathbb{E}_P\{N_j^\sigma(P)\} \le 1.$$

Taking expectations in (C.3) and summing over the K slots proves (12). Nondecreasingness follows immediately from the union definition. The argument uses the contemporaneous rejection count R_j at the slot's first actual e-BH entry; it therefore does not justify e-BH applied to coordinatewise running maxima. □

Multiplicity scope and alternatives. The guarantees above control false discoveries among the K registered or padded slots. They do not automatically control a post hoc list of deduplicated scientific targets when several slots can encode the same target; such a deployment should enforce unique-target registration or prespecify a grouping rule with its own error-control justification. Permanent padding keeps the multiplicity denominator independent of random births but may be conservative. Replacing K by the realized number of births is not covered by the present proof. When no scientifically credible deterministic slot budget is available, online e-FDR procedures provide a natural alternative multiplicity layer (Xu & Ramdas, 2024; Fischer et al., 2024), but that extension is not evaluated here.

D Robust and full-mark factors

Expanded construction for Proposition 6. Fix a predeclared predictable sensitivity path $\Gamma = (\Gamma_{j,t})$ with finite $\mathcal{H}_t$-measurable $\Gamma_{j,t} \ge 1$, and define

$$\widetilde{p}^{\Gamma}_{0,j,t} = \min\{\Gamma_{j,t}p_{0,j,t}, 1\}, \qquad I^{\Gamma}_{j,t} = I^{\circ}_{j,t}\mathbf{1}\{\widetilde{p}^{\Gamma}_{0,j,t} \in [\varepsilon, 1-\varepsilon]\}. \tag{D.1a}$$

The cutoff is applied to the inflated probability rather than inherited from $I_{j,t}$. Define

$$H_j^{\Gamma}: \quad p_{j,t} := \mathbb{P}(Y_{j,t} = 1 \mid \mathcal{H}_t) \le \widetilde{p}^{\Gamma}_{0,j,t} \quad \text{whenever } I^{\Gamma}_{j,t} = 1, \tag{D.1b}$$

and let $N_j^{\Gamma}(P)$ be the corresponding $\mathcal{F}_{B_j}$-measurable realized-slot null label. Using the same predictable raw $\beta_{j,t}$, set

$$(\eta^{\Gamma}_{j,t}, \widehat{p}^{\Gamma}_{j,t}) = \begin{cases} \left(\dfrac{1-\varepsilon-\widetilde{p}^{\Gamma}_{0,j,t}}{1-\widetilde{p}^{\Gamma}_{0,j,t}}\beta_{j,t},\ \widetilde{p}^{\Gamma}_{0,j,t} + (1-\varepsilon-\widetilde{p}^{\Gamma}_{0,j,t})\beta_{j,t}\right), & I^{\Gamma}_{j,t} = 1, \\ (0, 1/2), & I^{\Gamma}_{j,t} = 0, \end{cases} \tag{D.1c}$$

$$L^{\Gamma}_{j,t} = \begin{cases} 1 + \eta^{\Gamma}_{j,t}\left(Y_{j,t}/\widetilde{p}^{\Gamma}_{0,j,t} - 1\right), & I^{\Gamma}_{j,t} = 1, \\ 1, & I^{\Gamma}_{j,t} = 0, \end{cases} \qquad E^{\Gamma}_j(t) = \prod_{s=1}^{t} L^{\Gamma}_{j,s}. \tag{D.1d}$$

Under H_j^{Γ}, $\{N_j^{\Gamma}(P)E_j^{\Gamma}(t) : t \ge B_j\}$ is a nonnegative supermartingale and satisfies Theorem 2's stopping-time conclusion. After permanent padding, Corollary 4 and Theorem 5 apply separately to each predeclared Γ path. If $q_t(v \mid \mathcal{H}_t) \le \Gamma_t q_{0,t}(v \mid \mathcal{H}_t)$, any nonnegative full-mark factor normalized under $q_{0,t}$, divided by Γ_t, is likewise valid.

Operationally, the nominal replay stores $(p_{0,j,t}, \beta_{j,t}, Y_{j,t})$ before any outcome-dependent update. For every predeclared Γ path, the sensitivity replay inflates the stored reference, reevaluates eligibility using $I^{\Gamma}_{j,t}$, rebuilds $\widehat{p}^{\Gamma}_{j,t}$ from the same predictable raw $\beta_{j,t}$, and recomputes the factor. It does not inherit nominal eligibility, reinterpret the neural hit-probability output as a bet fraction, or reproject that output separately for each Γ. Thus all sensitivity paths use the same observed destinations and predictable betting decisions while changing only the stated reference bound.

D.1 Proof of Proposition 6

Proof. For the predeclared Γ path, $I^{\Gamma}_{j,t}$, $\widetilde{p}^{\Gamma}_{0,j,t}$, $\beta_{j,t}$, $\eta^{\Gamma}_{j,t}$, and $\widehat{p}^{\Gamma}_{j,t}$ are $\mathcal{H}_t$-measurable. On $N_j^{\Gamma}(P)$ and $I^{\Gamma}_{j,t} = 1$, equation (D.1c) gives $\widehat{p}^{\Gamma}_{j,t} \in [\widetilde{p}^{\Gamma}_{0,j,t}, 1-\varepsilon]$. Conditional on $\mathcal{H}_t$,

$$\mathbb{E}_P(L^{\Gamma}_{j,t} \mid \mathcal{H}_t) = p_{j,t}\frac{\widehat{p}^{\Gamma}_{j,t}}{\widetilde{p}^{\Gamma}_{0,j,t}} + (1-p_{j,t})\frac{1-\widehat{p}^{\Gamma}_{j,t}}{1-\widetilde{p}^{\Gamma}_{0,j,t}}.$$

The right side is affine and nondecreasing in $p_{j,t}$; under H_j^Γ, its maximum is attained at $p_{j,t} = \widetilde{p}_{0,j,t}^\Gamma$, where it equals one. If $I_{j,t}^\Gamma = 0$, then $L_{j,t}^\Gamma = 1$. The proof of Theorem 2 therefore gives the claimed null-weighted supermartingale property and optional-sampling bound after replacing (N_j, E_j) by (N_j^Γ, E_j^Γ). Applying the padding and self-consistency proof of Corollary 4 with $(N_j^{\Gamma,\sigma}, E_j^\Gamma)$ gives snapshot FDR control, and applying the unchanged first-entry proof of Theorem 5 gives accumulated-snapshot FDR control. The same raw $\beta_{j,t}$, computed predictably from the nominal hit-probability head or fixed by design before the destination is revealed, may be reused for every Γ; the robust path rebuilds $\widehat{p}_{j,t}^\Gamma$ from $\widetilde{p}_{0,j,t}^\Gamma$ and does not reproject the head output.

For an $\mathcal{H}_t$-measurable full-mark factor $B_t(v) \geq 0$ satisfying $\sum_v q_{0,t}(v \mid \mathcal{H}_t)B_t(v) \leq 1$, the conditional pointwise ratio bound gives

$$\mathbb{E}_q\{B_t(V_t)/\Gamma_t \mid \mathcal{H}_t\} = \sum_v q_t(v \mid \mathcal{H}_t)B_t(v)/\Gamma_t \leq \sum_v q_{0,t}(v \mid \mathcal{H}_t)B_t(v) \leq 1.$$

Multiplication and optional sampling proceed as before. □

D.2 Exact-law neural destination betting

Under the stronger null $q_t(\cdot \mid \mathcal{H}_t) = q_{0,t}(\cdot \mid \mathcal{H}_t)$, a temporal graph model may output any predictable distribution $\pi_{j,t}$ supported where $q_{0,t}(\cdot \mid \mathcal{H}_t) > 0$. For $\eta_{j,t} \in [0,1]$ and $q_{0,t}(v \mid \mathcal{H}_t) > 0$,

$$B_{j,t}(v) = (1 - \eta_{j,t}) + \eta_{j,t}\frac{\pi_{j,t}(v)}{q_{0,t}(v \mid \mathcal{H}_t)} \tag{D.1}$$

and set $B_{j,t}(v) = 1 - \eta_{j,t}$ when $q_{0,t}(v \mid \mathcal{H}_t) = 0$. This factor has conditional expectation one and can exploit the full destination ranking, but it is not valid for the weaker subset-mass null (3) without additional shape restrictions. The main method therefore uses (7).

E Proof of the neural-regret power bound

Proof of Theorem 7. For active update k, set $X_k = \log L_k$. Direct calculation gives

$$\begin{aligned}\mathbb{E}(X_k \mid \mathcal{H}_{t_{j,k}}) &= p_k \log\frac{\widehat{p}_k}{p_{0,k}} + (1-p_k)\log\frac{1-\widehat{p}_k}{1-p_{0,k}} \\ &= \mathrm{KL}\{\mathrm{Ber}(p_k)\|\,\mathrm{Ber}(p_{0,k})\} - \mathrm{KL}\{\mathrm{Ber}(p_k)\|\,\mathrm{Ber}(\widehat{p}_k)\}.\end{aligned} \tag{E.1}$$

Summing (E.1) proves the expectation identity. To respect the two-stage filtration, define at every global time after birth

$$X_{j,t} = \log L_{j,t}, \qquad \mu_{j,t} = \mathbb{E}(X_{j,t} \mid \mathcal{H}_t).$$

Both quantities are zero when $I_{j,t} = 0$, and $X_{j,t} \in [-c_\varepsilon, c_\varepsilon]$ when $I_{j,t} = 1$. For fixed $\lambda > 0$, let

$$\mathcal{M}_j(t;\lambda) = \exp\left[-\lambda\sum_{s=B_j+1}^{t}(X_{j,s} - \mu_{j,s}) - \frac{\lambda^2 c_\varepsilon^2}{2}\sum_{s=B_j+1}^{t} I_{j,s}\right].$$

Because $I_{j,t}$ is $\mathcal{H}_t$-measurable, the conditional Hoeffding lemma gives

$$\mathbb{E}\left[\exp\left\{-\lambda(X_{j,t} - \mu_{j,t}) - \frac{\lambda^2 c_\varepsilon^2}{2}I_{j,t}\right\}\middle|\mathcal{H}_t\right] \leq 1.$$

The tower property from $\mathcal{H}_t$ to $\mathcal{F}_{t-1}$ therefore makes $\mathcal{M}_j(t;\lambda)$ a nonnegative $\mathcal{F}_t$-supermartingale. Evaluating it at $t_{j,m}$ and optimizing λ gives

$$\mathbb{P}\left[\sum_{k=1}^{m}\{X_k - \mathbb{E}(X_k \mid \mathcal{H}_{t_{j,k}})\} \leq -c_\varepsilon\sqrt{2m\log(1/\delta)}\right] \leq \delta,$$

which proves (14). If some $c > 0$ satisfies $\mathbb{P}\{(S_m - R_m)/m \geq c\} \to 1$, choose $\delta_m = \exp(-\sqrt{m})$; then the deviation term is $O(m^{3/4})$ and the failure probability vanishes. Hence wealth crosses every fixed threshold with probability tending to one.

For the simultaneous bound, Ville's inequality gives $\sup_t \mathcal{M}_j(t;\lambda) \le 1/\delta$ with probability at least $1-\delta$. Evaluating this event at every finite $t_{j,n}$, using $\sum_{s=B_j+1}^{t_{j,n}} I_{j,s} = n$, gives simultaneously for every $n \ge 1$,

$$\log E_j(t_{j,n}) \ge S_n - R_n - \frac{\lambda c_\varepsilon^2 n}{2} - \frac{\log(1/\delta)}{\lambda}. \tag{14a}$$

□

E.1 Proof of Proposition 8

Proof. For any subset $A \subseteq \mathcal{R}$, the definition of total variation gives $|q_k(A) - \pi(A)| \le \delta_k$. Since $\widehat{A}$ maximizes $\pi(A)$ over size-a sets,

$$q_k(\widehat{A}) \ge \pi(\widehat{A}) - \delta_k \ge \pi(A^\star) - \delta_k \ge q_k(A^\star) - 2\delta_k.$$

For fixed p_0, the derivative of $x \mapsto \mathrm{KL}\{\mathrm{Ber}(x) \| \mathrm{Ber}(p_0)\}$ is $\log\{x(1-p_0)/(p_0(1-x))\}$, which is nonnegative for $x \ge p_0$. The assumed lower bound therefore gives the stated inequality for each update, and summing gives the bound on $S_m(\widehat{A})$. Finally, Pinsker's inequality gives $\mathrm{TV}(q_k, \pi) \le \sqrt{\mathrm{KL}(q_k \| \pi)/2}$. □

E.2 Proof of Corollary 9

Proof. Stop the factor sequence after its first crossing of h and set $\lambda = \gamma/c_\varepsilon^2$ in (14a). On the event in (14a), if no crossing has occurred by active update m, the drift condition gives

$$\log E_j(t_{j,m}) \ge \gamma m/2 - c_\varepsilon^2 \log(1/\delta)/\gamma.$$

At $m = m_\delta(h,\gamma)$, the right side is at least $\log h$, contradicting the absence of a crossing. This proves the probability bound in (14b). For the expectation bound, let $\nu_{h,n} = \nu_h \wedge n$. Conditional drift before crossing, including the crossing update, is at least γ, while a crossing overshoots $\log h$ by at most c_ε. Hence

$$\gamma \mathbb{E}(\nu_{h,n}) \le \mathbb{E}\{\log E_j(t_{j,\nu_{h,n}})\} \le \log h + c_\varepsilon.$$

Monotone convergence proves the expectation bound. For the finite-lifetime statement, let $\mathcal{G}_\delta$ be the simultaneous event in (14a), which has probability at least $1-\delta$. On $\mathcal{G}_\delta \cap \{M_j \ge m_\delta(h,\gamma)\}$, the same argument forces a crossing by $m_\delta(h,\gamma)$. Therefore $\{\nu_h^{\mathrm{op}} > m_\delta(h,\gamma)\} \subseteq \mathcal{G}_\delta^c \cup \{M_j < m_\delta(h,\gamma)\}$, and the union bound proves (14c). Finally, Pinsker's inequality gives $\mathrm{KL}\{\mathrm{Ber}(p_k) \| \mathrm{Ber}(p_{0,k})\} \ge 2(p_k - p_{0,k})^2$; subtracting $r_k \le r$ yields $\mu_k \ge 2\Delta^2 - r = \gamma$. □

F Why a local graph filtration can fail

Example 12. Let $W \sim \mathrm{Ber}(1/2)$ be revealed by an event in relation type 1. A later relation-2 event has destination $V = W$. A dyad-local history that omits the relation-1 event assigns the uniform baseline $q_0(V=0) = q_0(V=1) = 1/2$. A globally informed learner sees W and uses $B(V;W) = 2\mathbf{1}\{V = W\}$. A procedure that evaluates normalization under the local uniform law while allowing the factor to depend on globally observed W incorrectly obtains $\sum_v q_0(v)B(v;W) = 1$. The factor is not predictable in the local filtration; in the correct global pre-destination field, $V = W$ deterministically and its conditional expectation is two. The error is not graph dependence by itself, but combining a globally informed bet with a null law conditioned on less information.

G Tied timestamps and delayed observations

For a batch of events sharing timestamp b, the implementation performs:

1. construct all contexts from the graph ending before batch b, draw an independent permutation of the batch indices, and record both before any destination is revealed;
2. follow that order and, before each destination, compute its risk set and baseline from the conditional opportunity law given the earlier marks in the same batch; under stub sampling without replacement, this simply removes the previously used stubs;

3. keep neural parameters and graph memory frozen, compute the predictable stake, reveal the next destination, persist its factor, and cache the event before continuing; and
4. after the batch closes, replay the cached events in the persisted order to update graph memory; during training only, perform one optimization update after this replay, while validation and test make no gradient update. Candidate births are suppressed during the batch. After all memory updates, inspect the distinct contexts that appeared, ordered by first occurrence in the persisted permutation; eligible births use the complete post-batch history, receive the batch-end index as B_j, and receive no evidence from that batch.

This construction is an ordinary sequential filtration conditional on the recorded permutation, so the one-step proof applies. Multiplying marginal factors after revealing a dependent batch jointly is not generally valid. An alternative is a single factor normalized under a fully specified joint batch law. The randomized order is used only to expose the conditional law; it does not create temporal priority among tied events.

If an event is recorded after a delay, it enters $\mathcal{F}_t$ only when observed. A delay preserves the construction only when the opportunity law is defined relative to the observation filtration and unobserved delayed events do not alter current risk sets or weights, or when their effect is integrated out under a correct observation model. Outcome-dependent verification additionally requires a known selection mechanism or a separate sensitivity model; an unobserved selective delay cannot be repaired by relabeling missing outcomes as negative.

H Complete simulation specification

H.1 Data-generating process

Set $n_1 = \lfloor 0.40n \rfloor$, $n_2 = \lfloor 0.35n \rfloor$, and $n_3 = n - n_1 - n_2$, with consecutive identifiers within type. The directed compatibility map is $r_1 : 1 \to 2$, $r_2 : 1 \to 3$, and $r_3 : 2 \to 3$; type-3 nodes are receivers only. Initialize every compatible receiver with one stub. For each relation, partition initial receivers by sorted identifier into disjoint full blocks of size five, leaving an incomplete final block outside the candidate library. If $\mathcal{C}$ is the set of compatible source–relation contexts, designate $m_\star = \lfloor 0.10|\mathcal{C}| + 1/2 \rfloor$ contexts as enriched; this is 58 of the 575 contexts when $n = 500$. Selection uses the smallest digests produced by the 256-bit Secure Hash Algorithm (SHA-256) from strings encoded in the 8-bit Unicode Transformation Format (UTF-8), `GraphBet-ICLR27-v1|enrich|u|r`, interpreted as big-endian integers; throughout this appendix, tokens such as `n`, `u`, `v`, `b`, and `first-node-id` are replaced by their base-10 integer values, while relations use the American Standard Code for Information Interchange (ASCII) labels `r1`, `r2`, and `r3`. Vertical bars are literal, with no whitespace or zero padding. Digest ties are broken lexicographically by (u, r). Within each context, number the full blocks $b = 1, 2, \ldots$ by increasing smallest receiver identifier. The hidden block is the one with the smallest digest of the UTF-8 string `GraphBet-ICLR27-v1|block|u|r|b`; digest ties use the smaller integer b. These choices are fixed across evaluation replications and unavailable to the predictor.

Each replication uses a NumPy PCG64 generator initialized from the declared seed. The graph generator consumes random draws only in the order below, so the degree–recency and oracle procedures receive the same stream. Set $G_0 = 1$. Let $x_{u,r,t} = \sum_{s<t} \exp\{-(t-s)/200\} \mathbf{1}\{(U_s, R_s) = (u, r)\}$. At event t:

1. Draw $H_t \sim \mathrm{Ber}(0.05)$ independently conditional on the past, set $G_t = (-1)^{H_t} G_{t-1}$, and draw an active source with probability proportional to $1 + \sum_r x_{u,r,t}$.
2. Among relations compatible with the source type, draw $R_t = r$ with probability proportional to $\exp\{0.5 x_{u,r,t} + \zeta_r G_t\}$, where $(\zeta_{r_1}, \zeta_{r_2}, \zeta_{r_3}) = (0.6, -0.6, 0)$. Reveal $C_t = (U_t, R_t)$.
3. Form $\mathcal{R}_t(C_t)$ and draw V_t from (15), using $\theta_c = 0$ for an unenriched context. Within the hidden block and its complement, mass remains proportional to the pre-event stub law (4).
4. Record all active candidate factors, set $d_{t+1}(V_t, R_t) = d_t(V_t, R_t) + 1$, leave every other relation-specific receiver counter unchanged, and update the predictor history. All predictor parameters remain fixed throughout each evaluation run.

The primary stream contains $T = 250{,}000$ events at $n = 500$. The feasible baseline fixes the degree–recency parameters before evaluation; no controlled stream is used to fit a neural predictor. In the controlled study, the evidence horizon H counts active updates with $I_{j,t} = 1$. In the real and semi-synthetic replays, by contrast, H counts matching source–relation opportunities, including opportunities whose factor is one; Section H.3 gives the operational definition.

Before evaluation, an oracle feasibility envelope is run only on pilot seeds 40–49. It crosses multipliers $\{2, 3, 4\}$ with evidence horizons $\{50, 100, 250, 500\}$, using $T = 1000H$ so that a longer horizon has enough matching events. The oracle knows each hidden block and uses the exact conditional log-growth stake. A configuration is feasible when final-time conditional power is at least 0.80 and end-to-end recall is at least 0.20 at $\alpha = 0.10$; the first feasible configuration in increasing H, then increasing multiplier, is frozen. This rule selected $H = 250$, $T = 250{,}000$, and multiplier four. The oracle envelope is a design feasibility check, not an evaluation result for any fitted predictor.

In each evaluation stream, the first 30% of events are a history warm-up used only to initialize graph memory, receiver totals, and eligibility; they cause no gradient update or tuning. The final 70% form the sequential evaluation segment. Completed warm-up events count toward the 20-event candidate eligibility requirement. Subject to a ten-event cooldown, a free slot ranks candidate blocks by the predictor's mean member score and freezes the highest block for its fixed evidence horizon. Candidate births stop after the first quarter of the evaluation segment.

Because blocks within a context are disjoint and have positive reference mass, write

$$Z_{c,t} = 1 + \{\exp(\theta_c) - 1\} q_{0,t}(A_c^\star \mid \mathcal{H}_t).$$

If $\theta_c > 0$, then $q_{\theta,t}(A_c^\star \mid c) = \exp(\theta_c) q_{0,t}(A_c^\star \mid \mathcal{H}_t)/Z_{c,t} > q_{0,t}(A_c^\star \mid \mathcal{H}_t)$. For every other library block A, $q_{\theta,t}(A \mid c) = q_{0,t}(A \mid \mathcal{H}_t)/Z_{c,t} \le q_{0,t}(A \mid \mathcal{H}_t)$; when $\theta_c = 0$, equality holds for every block. Thus a born slot is nonnull exactly when $\theta_c > 0$ and $A_j = A_c^\star$, for every reachable continuation. The simulator logs the exact $p_{j,t} = \sum_{v \in A_j} q_{\theta,t}(v \mid c_j)$ to verify this classification but does not define null truth from a realized path.

H.2 Factor and predictor grid

Table 6: Controlled-study configuration and evaluation grid.

Component	Values
Primary	$n = 500$, $T = 250{,}000$, $\lvert A\rvert = 5$, active-update horizon 250, enriched fraction 0.10, $\varepsilon = 0.01$
Power signal	$\theta \in \{0, \log(1.5), \log(2), \log(4)\}$
Predictor	degree/recency; hidden-block oracle
Stake	fixed $\beta = 0.25$; conditional log-growth oracle
Outer rule	fixed-slot e-BH, $K = 200$, $\alpha = 0.10$
Evaluation seeds	null: 10000–10299; power: 20000–20149 per signal

H.3 Neural architecture and candidate lifecycle

Both neural predictors use 128-dimensional node memory and output embeddings. Lagged-THG adapts the temporal heterogeneous encoder design of TimeHet (Wang et al., 2026). To satisfy the pre-destination filtration required by GraphBet, it replaces snapshot-level processing by event-level states constructed from completed events, freezes state within tied timestamps, scores admissible destinations before revelation, and updates recurrent memory only after the corresponding betting factor has been stored. The TGN-style ablation uses two temporal-attention layers, two heads, and the 20 most recent neighbors. Lagged-THG uses 32-dimensional relation embeddings, 32-dimensional absolute-time and recency encodings, a 128-dimensional relation-wise mean aggregator, and softmax relation fusion. The destination and hit-probability heads are two-hidden-layer multilayer perceptrons (MLPs) with widths 128 and 64, rectified linear unit (ReLU) activations, and the selected dropout. Node identifiers are not features.

Let $s_v(t^-)$ denote the node state constructed from completed events, e_r a relation embedding, and $e_{\mathrm{abs}}(\tau_t)$ and $e_{\mathrm{rel}}(u,r,v,t)$ the absolute-time and recency features. The destination ranker has score

$$g_\theta(u,r,v,t) = \mathrm{MLP}_\theta\big[s_u(t^-) \,\|\, s_v(t^-) \,\|\, e_r \,\|\, e_{\mathrm{abs}}(\tau_t) \,\|\, e_{\mathrm{rel}}(u,r,v,t)\big]\,.$$

Membership of A_j is frozen, while its summary $\bar{s}_{A_j}(t^-) = |A_j|^{-1}\sum_{v\in A_j} s_v(t^-)$ evolves using only completed events. At every matching opportunity, let $p^\dagger_{0,j,t} = \min\{1-10^{-6}, \max(10^{-6}, p_{0,j,t})\}$. The hit-probability head, its one-sided probability projection, and the resulting raw bet fraction are

$$\begin{aligned}
\widetilde{q}_{j,t} &= \mathrm{sigmoid}\Big\{f_\psi\Big[s_{u_j}(t^-) \,\|\, \bar{s}_{A_j}(t^-) \,\|\, \mathrm{logit}(p^\dagger_{0,j,t}) \,\|\, \mathrm{age}_{j,t}\Big]\Big\}\,,\\
\widehat{p}_{j,t} &= \max\{p_{0,j,t}, \min(1-\varepsilon, \widetilde{q}_{j,t})\},\\
\beta_{j,t} &= \begin{cases} (\widehat{p}_{j,t} - p_{0,j,t})/(1-\varepsilon-p_{0,j,t}), & p_{0,j,t} < 1-\varepsilon,\\ 0, & p_{0,j,t} \ge 1-\varepsilon. \end{cases}
\end{aligned}$$

Here $p^\dagger_{0,j,t}$ is used only as a finite neural input; all opportunity probabilities, eligibility decisions, factors, and sensitivity reconstructions use the unmodified $p_{0,j,t}$.

The implementation initializes every 128-dimensional recurrent memory h_v at zero. It measures time in units of the median positive inter-event gap in the training split, fixed before validation; within each stream the origin τ_0 is its first timestamp and $z = (\tau-\tau_0)/\,\mathrm{median}_{\mathrm{train}}(\Delta\tau)$. The absolute-time encoding is $[\sin(\omega_k z), \cos(\omega_k z)]_{k=0}^{15}$, where $\omega_k = 2\pi 10^{-4k/15}$. For normalized pair recency Δ, the recency encoding is $[\exp\{-\,\mathrm{softplus}(\lambda_k)\Delta\}]_{k=1}^{32}$; an unseen pair has the zero vector, and the learned rates are initialized at 32 logarithmically spaced inverse lags from 10^{-4} to one.

For Lagged-THG, let $\mathcal{N}^{20}_{x,r,t}$ be the 20 most recent pre-event neighbors of node x through relation r. Define

$$z_{x,r,t} = \frac{1}{|\mathcal{N}^{20}_{x,r,t}|}\sum_{y\in\mathcal{N}^{20}_{x,r,t}} \phi[h_y \,\|\, e_r \,\|\, e_{\mathrm{rel}}(x,r,y,t)]\,,$$

with $z_{x,r,t} = 0$ for an empty set, and

$$a_{x,r,t} = \frac{\exp\{w_a^\top \tanh(W_a z_{x,r,t})\}}{\sum_{r'}\exp\{w_a^\top \tanh(W_a z_{x,r',t})\}}, \qquad s_x(t^-) = \mathrm{LayerNorm}\left[h_x + W_o\sum_r a_{x,r,t} z_{x,r,t}\right].$$

Here ϕ is a 128-output ReLU MLP, and the sum includes relations incident to the node type. The TGN-style ablation uses the same memory, encodings, and prediction heads but replaces these two displays by two residual temporal attention layers with two heads; their query is the current node memory and their keys and values concatenate neighbor memory, relation embedding, absolute time, and recency over the same 20-neighbor sets.

After the factor for an untied event (u,r,v,τ) is stored, write $a_t = e_{\mathrm{abs}}(\tau)$ and $r_{uv,t} = e_{\mathrm{rel}}(u,r,v,\tau)$. A shared 128-dimensional message MLP forms

$$\begin{aligned}
m_u &= \mathrm{MLP}_{\mathrm{msg}}[s_u \| s_v \| e_r \| a_t \| r_{uv,t} \| e_{\mathrm{src}}],\\
m_v &= \mathrm{MLP}_{\mathrm{msg}}[s_v \| s_u \| e_r \| a_t \| r_{uv,t} \| e_{\mathrm{dst}}].
\end{aligned}$$

where the two 8-dimensional role embeddings distinguish sender and receiver. Set $h_u \leftarrow \mathrm{GRU}(m_u, h_u)$ and $h_v \leftarrow \mathrm{GRU}(m_v, h_v)$; all other memories are unchanged. Every MLP not otherwise specified has hidden layers of widths 128 and 64, ReLU activation, selected dropout, and a task-specific linear output layer: ϕ and $\mathrm{MLP}_{\mathrm{msg}}$ output 128 coordinates, while the destination and hit-probability heads output one. These definitions also fix the state updates used during real-data training, validation, and replay.

Training uses float32 AdamW, weight decay 10^{-5}, event batch size 512, and gradient-norm clipping at one. Dataset-specific real-data grids and schedules are in Appendix I. Training and ranker-selection validation score the events in an optimization batch against the graph state at the beginning of that batch, then replay their messages in persisted order. This frozen-batch convention is used only for optimization and hyperparameter selection; final reported prediction metrics, calibration, and test use the stricter event-level pre-destination updates in Appendix G. First fit the encoder and ranker by next-destination loss. Freeze them, replay the training streams chronologically to generate pseudo-candidates, and fit the hit-probability head by Bernoulli log loss for $Y_{j,t}$. At evaluation, its sigmoid

output $\widetilde{q}_{j,t}$ is projected to $[p_{0,j,t}, 1-\varepsilon]$ before conversion to $\beta_{j,t}$, as specified in Section 4.3. Replay applies the same 20-event eligibility threshold, ten-event cooldown, post-event birth rule, lowest-free-slot assignment, and matching-opportunity horizon as evaluation; registration closes after the first quarter of each replay segment. With the ranker hyperparameters fixed, the hit-probability head is trained to at most epoch 30 with patience five and its checkpoint is selected by tuning-portion pseudo-candidate Bernoulli log loss. Ranker MRR ties within 10^{-6} use, in order, the smaller parameter count and the lexicographically smaller pair (learning rate, dropout). We enable deterministic PyTorch algorithms.

For an untied event t, after persisting all factors and updating graph state, the simulator checks only the just-completed context C_t for a birth using $\mathcal{F}_t$, so at most one slot is born per event. That context is eligible if it has at least 20 completed matched events, has no active slot, has completed ten matched events since its last retirement, and registration remains open. Its blocks are ranked using the updated state; score ties are broken by the SHA-256 block digest defined above. The winner is assigned to the lowest unused slot with $B_j = t$, so event t never contributes evidence. The operational horizon H counts the next H matching source–relation events, including events whose factor is one because the reference probability falls outside the active range. This predictable lifetime rule is distinct from the active-update index used in Theorem 7. Tied events instead follow Appendix G; their births occur only at batch closure.

All reported controlled evaluations use the complete, disjoint seed ranges in Table 6; interrupted replications retain their identifiers and are resumed or rerun rather than deleted according to their outcomes.

The degree–recency ranker assigns

$$g^{\mathrm{DR}}_{\lambda,h}(u,r,v,t) = \log\{1 + d_t(v,r)\} + \lambda \exp\{-\Delta_t(u,r,v)/h\},$$

where the second term is zero if (u,r,v) has not appeared. The controlled baseline fixes $\lambda = 1$, $h = 100$, and $\beta = 0.25$ before evaluation, nominates the block with largest mean member score, and uses (6)–(7). The realistic semi-synthetic baseline instead fixes $\lambda = 1$, $\beta = 0.10$, and takes h from the already selected reference law; it introduces no additional fitted parameters.

The log-growth oracle uses the exact conditional candidate-set probability $p_{j,t}$ from (15), subject to the same directional and clipping constraints:

$$\widehat{p}^{\,\mathrm{or}}_{j,t} = \begin{cases} \min\{1-\varepsilon, \max(p_{j,t}, p_{0,j,t})\}, & I_{j,t} = 1, \\ 1/2, & I_{j,t} = 0. \end{cases}$$

It records factor one off-update and for a depleted candidate. It is an infeasible one-sided log-growth oracle: on active updates it maximizes the conditional expected log factor over $[p_{0,j,t}, 1-\varepsilon]$, but it need not maximize finite-horizon rejection probability. Thus the log-growth oracle, exact GRAPHBET, and correctly bounded robust GRAPHBET are inferential procedures.

H.4 EVALUATION MEASURES AND CONTROLLED RESULTS

Let $\widehat{\mathcal{D}}_m(t)$ be method m's contemporaneous e-BH set and $\widehat{\mathcal{U}}_m(t) = \bigcup_{s=\sigma+1}^{t} \widehat{\mathcal{D}}_m(s)$ its accumulated post-registration snapshot set. At the prespecified final event T, write $\mathcal{S}^{\mathrm{snap}}_m(T) = \widehat{\mathcal{D}}_m(T)$ and $\mathcal{S}^{\mathrm{acc}}_m(T) = \widehat{\mathcal{U}}_m(T)$. For reporting rule $r \in \{\mathrm{snap}, \mathrm{acc}\}$, define

$$\mathrm{FDP}^r_m(T) = \frac{\sum_{j=1}^{K} N^\sigma_j \mathbf{1}\{j \in \mathcal{S}^r_m(T)\}}{|\mathcal{S}^r_m(T)| \vee 1},$$

$$\mathrm{TestPower}^r_m(T) = \frac{|\mathcal{S}^r_m(T) \cap \mathcal{J}_1(T)|}{|\mathcal{J}_1(T)|} \quad \text{when } |\mathcal{J}_1(T)| > 0.$$

where $\mathcal{J}_1(T)$ is the set of analytically nonnull born slots; testing power is undefined in a replication with $|\mathcal{J}_1(T)| = 0$. Its primary Monte Carlo estimand is the macro-average over contributing replications, with their count n_+ reported. Corollary 4 controls the expected snapshot FDP and Theorem 5 controls the expected accumulated FDP. Both guarantees concern slots; merging repeated slots into scientific targets requires unique registration or an additional grouping rule.

Let $\mathcal{C}_\star^{\mathrm{elig}}(T)$ be the enriched contexts that reach the 20-matched-event eligibility threshold before registration closes, and let $\mathcal{N}_m(T)$ contain those contexts for which method m registers the hidden target. For rule r, let $\mathcal{Q}_m^r(T) \subseteq \mathcal{N}_m(T)$ contain contexts whose corresponding nonnull slot belongs to $\mathcal{S}_m^r(T)$. We report

$$\mathrm{NomRecall}_m(T) = \frac{|\mathcal{N}_m(T)|}{|\mathcal{C}_\star^{\mathrm{elig}}(T)|}, \qquad \mathrm{E2ERecall}_m^r(T) = \frac{|\mathcal{Q}_m^r(T)|}{|\mathcal{C}_\star^{\mathrm{elig}}(T)|}.$$

Both ratios are set to zero when no enriched context is eligible. Nomination and end-to-end recall are context-level performance metrics, not the slot-level object of the FDR guarantee. All numerical results in this paper use $r = \mathrm{snap}$. In the semi-synthetic replays, each enriched context contributes at most one nonnull born slot, so final true-slot counts and discovered-context counts agree. For a persistent delay analysis, the first-entry time must be right-censored at a common prespecified administrative horizon and summarized by a survival estimand such as restricted mean time to discovery; assigning a different $H_j(T) + 1$ to each nondetection can reverse comparisons and is not used here.

Under the global null, where FDR equals $\mathbb{P}(R > 0)$, we use a two-sided exact 95% Clopper–Pearson interval. Controlled-stream means use a deterministic 10,000-resample percentile bootstrap with seed 20260918; undefined conditional quantities are excluded and reported with their contributing-replication counts. Wiki semi-synthetic component-arm intervals use 10,000 crossed bootstrap replicates that independently resample the three fit identifiers and the 20 shared replay identifiers while preserving paired arm comparisons; identical model-free degree/fixed copies are collapsed before resampling. These are descriptive crossed intervals for the prespecified design. As a secondary controlled-stream diagnostic, pooled born-slot power divides total true discoveries by total born nonnull slots and uses a replication-cluster bootstrap.

At the final report, both procedures have 0/300 global-null rejections; the two-sided 95% Clopper–Pearson interval for $\Pr(R > 0)$, which equals FDR under the global null, is [.000,.012]. At multiplier four, degree–recency 95% bootstrap intervals for nomination recall, end-to-end recall, and conditional testing power are [.023,.030], [.021,.028], and [.897,.968], respectively; the corresponding oracle intervals are [.342,.360], [.321,.340], and [.931,.950]. The conditional-power effective sample sizes are 116 and 150. The pooled born-slot powers, reported as secondary diagnostics, are .935 [.901,.966] and .942 [.933,.951].

I Real-data protocol

I.1 Data and preprocessing

The evaluation uses `py-tgb==2.3.0` with loader keys `tgbl-wiki` and `tgbl-review`, with the package registry required to report dataset version 2. The data audit records the package commit or wheel hash, download URL, archive checksum, event and node counts, timestamp range, tied-timestamp count, official split masks, and negative-sample checksum. Version-1 samples and locally processed files are excluded unless byte- and row-level comparisons establish equivalence.

Events are grouped by timestamp, with the original row identifier retained only as an event identifier. Within a tied batch, the procedure draws and persists a seeded, destination-blind permutation in Appendix G before supplying any destination to scoring code. The primary inductive registry contains nodes seen before the current event and one $\star$ bucket with mass κ_{new}. An event mapped to $\star$ cannot fall in a known-node candidate, and its identity is added only after the factor is stored. Candidate identities therefore remain strictly inductive.

For a known destination v, the baseline receiver mass is

$$d_t(v) = \kappa + \sum_{s<t:V_s=v} \exp\{-(\tau_t - \tau_s)/h\}, \qquad d_t(\star) = \kappa_{\mathrm{new}} > 0, \tag{I.1}$$

where τ is measured in days. The parameters $(\kappa, \kappa_{\mathrm{new}}, h)$ are selected by mark log loss on the tuning portion of validation and then fixed before calibration and test.

I.2 Reference-law calibration and sensitivity anchor

Justification hierarchy. A formal observational interpretation requires a reference law identified by randomized exposure, enforced eligibility or capacity, or complete prospective logging of the destination choice set. When external design knowledge instead supplies a predictable one-sided error bound, Proposition 6 supports a sensitivity-valid interpretation at that Γ. Without either, we label discoveries reference-relative. Held-out calibration can falsify a proposed law and quantify mismatch, but cannot prove that unmeasured preferences are absent or that its bound transports to test.

Practitioner workflow. Equation (I.1) is one example, not a required model family. Before outcomes are inspected, an analyst should document the unit of exposure, the eligible destination set, and the assignment mechanism using randomization, scheduler rules, invitation or impression logs, or enforced capacity constraints. These records define predictable masses; pre-test fitting may refine them, and a chronological holdout should try to falsify the resulting law. Domain knowledge must supply any claimed pointwise error bound; a data-derived Γ remains diagnostic. Without exposure records or an external bound, conclusions should be labeled reference-relative.

The official validation interval is split chronologically: its first 80% is used for all hyperparameter and checkpoint selection, and its final 20% is a calibration interval that is never used to select a model. After selection, the opportunity law and rankers are frozen and replayed prospectively through the calibration interval with the test-time eligibility, birth, and retirement rules. For every pseudo-candidate matching opportunity we record $(Y_{j,t}, p_{0,j,t})$, including probabilities outside the nominal active range. Calibration is summarized for the full interval, the fixed probability bins $(0, .01], (.01, .05], (.05, .10], (.10, .25], (.25, .5], (.5, 1)$, and source-history bins 20–49, 50–199, and at least 200 completed events; adjacent empty bins are omitted rather than redefined.

For stratum g, let $O_g = \sum_{(j,t)\in g} Y_{j,t}$, $P_g = \sum_{(j,t)\in g} p_{0,j,t}$, and compute O_g/P_g and the number of matching opportunities. A one-sided 95% moving-block-bootstrap upper bound U_g is computed from 2,000 resamples of consecutive tied-timestamp batches, with block length $\lceil B^{1/3} \rceil$ for B calibration batches and a simultaneous max statistic over the predeclared strata. For neural candidate families, the maximum also ranges over the three training seeds. Before test, set

$$\Gamma_{\mathrm{a}} = 0.05 \left\lceil 20 \max\{1, \max_g U_g\} \right\rceil . \tag{I.2}$$

The observational table reports $\Gamma = 1$, Γ_{a}, and the predeclared sensitivity path $\{1.1, 1.25, 1.5\}$. The anchor is a transparent validation-based diagnostic, not a theorem that the pointwise bound transports to test. Robust FDR interpretation at Γ_{a} remains conditional on that transportability assumption or on an external design argument; if neither is defensible, the corresponding rejection count is labeled reference-relative rather than verified.

Table 7: Audit trail for Γ_{a}. Each row gives the largest simultaneous upper bound across the predeclared strata and three training seeds; the bootstrap uses 2,000 resamples.

Data	Predictor	Worst stratum	Seed	Opps.	O_g/P_g	U_g / Γ_{a}
Wiki	TGN-style	$p_0 \in (0, .01]$	940001	151	107.32	124.03 / 124.05
Wiki	Lagged-THG	$p_0 \in (0, .01]$	940002	186	118.99	135.41 / 135.45
Review	TGN-style	history 20–49	950005	2354	2.17	4.34 / 4.35
Review	Lagged-THG	history 50–199	950006	920	1.92	4.19 / 4.20

Across the three Wiki seeds, the sample standard deviations of $R_{1/a/1.5}$ are 1.5/0.0/1.0 for TGN-style and 6.9/0.6/8.5 for Lagged-THG; all three Review counts have zero standard deviation. Rejection counts need not be monotone in Γ: inflation changes both hit and miss factors, can activate a nominal $p_0 < \varepsilon$ once $\Gamma p_0 \geq \varepsilon$, and replaces factors by one when $\widetilde{p}_0^{\Gamma} > 1 - \varepsilon$. Consequently, terminal e-values and joint e-BH sets are not nested. Each prespecified Γ is therefore a separate valid sensitivity analysis, and no value is selected after observing outcomes.

I.3 Training, validation, and sequential test

1. We select $(\kappa, \kappa_{\mathrm{new}}, h)$ in (I.1) by mark log loss on the first 80% tuning portion of validation over $\kappa, \kappa_{\mathrm{new}} \in \{0.1, 1, 10\}$; use $h \in \{1/24, 1, 7, 30\}$ days for Wiki and

$h \in \{30, 180, 730, 1825\}$ days for Review. For the TGN-style ablation and Lagged-THG search learning rate $\{10^{-4}, 3 \cdot 10^{-4}, 10^{-3}\}$ and dropout $\{0, 0.2\}$, with weight decay 10^{-5}, event batch size 512, and $\varepsilon = 0.01$. Wiki uses the full training and tuning portions with the 3/10/30 successive-halving schedule and patience five. To control the much larger Review cost, Review compares only $(3 \cdot 10^{-4}, 0)$ and $(10^{-3}, 0.2)$ on the tie-safe first 5% chronological prefix of both the training portion and the tuning portion, training each configuration for three epochs. Selection MRR uses the official version-2 negatives with the frozen-batch state-update convention in Appendix H; after selection, the tuning MRR values in Table 2 reevaluate that same model-selection interval with strict event-level pre-destination updates. They are not held-out calibration MRR. The architecture, float32 AdamW optimizer, gradient clipping, selection-tie rules, and two-stage ranker–hit-probability fitting protocol are otherwise those in Appendix H. Training pairs each observed destination with $\min\{100, |\mathcal{R}_t| - 1\}$ distinct negatives sampled uniformly without replacement from the other members of the pre-event risk set using a seeded generator; binary logistic loss gives the positive and the aggregate negative terms equal weight. If this number is zero, the event contributes no ranker loss but still updates memory. For each model, a separate NumPy PCG64 negative generator is initialized at the model seed plus 17 and consumed in chronological event order. Benchmark negatives are not used for fitting; the version-2 validation negatives are used only to compute the official MRR for model selection. The selected hyperparameter tuple is then fixed and weights are retrained from three initialization seeds per architecture: Wiki TGN-style uses 940001, 940003, and 940005; Wiki Lagged-THG uses 940002, 940004, and 940006; Review TGN-style uses 950001, 950003, and 950005; and Review Lagged-THG uses 950002, 950004, and 950006. Each Wiki final fit uses the selected epoch count; every Review final fit uses three epochs. All final fits use the complete training portion. The fitted hit-probability head retains the 30-epoch cap and patience five on Wiki and uses three epochs on Review. Each seed has its own negative generator at seed plus 17, candidate family, betting process, and discovery set. No seed is selected using calibration or test outcomes. All three seeds enter the observational summaries and the crossed Wiki semi-synthetic analysis; the first listed Lagged-THG seed is used only for each separate 100-replay exact-law implementation check. Deterministic PyTorch algorithms are enabled. A source has a ten-matched-event cooldown after slot retirement. Every selected rule and network weight is frozen before the held-out calibration interval and test. The unseen bucket uses a learned type-specific 128-dimensional cold-start state, zero recency features, and its reference mass κ_{new}; this state is frozen before calibration and test. Once an unseen identity is revealed after factor persistence, it receives zero recurrent memory and then the ordinary event update.

The degree–recency semi-synthetic comparator replaces $d_t(v, r)$ in the controlled score by $d_t(v)$ from (I.1), measures $\Delta_t(u, v)$ in days, and fixes $\lambda = 1$, $\beta = 0.10$, and $h_{\mathrm{DR}} = h$ before replay. This baseline is not applied to observed test destinations and receives no validation-derived Γ_{a}.

2. Test-time graph memory is initialized from completed training and validation events; initialization excludes test edges, full-graph neighborhoods, and future normalizers.

3. Before each test destination, the procedure constructs the registry and receiver totals and updates active stakes. Wiki scores every known page. Review scores the predictable union of the 10,000 products with largest pre-event baseline mass and products previously reviewed by that source; $p_0(A)$ is still normalized against the full risk set. If a tied batch is present, use Appendix G.

4. For an untied event, once all existing-slot factors for event t have been persisted and the event has then been added to memory, only its source U_t is checked for a birth; hence at most one slot is born per event and event t supplies no factor to that slot. Before the registration cutoff, an eligible source with at least 20 past events, a completed ten-event cooldown, and no active slot freezes its top-five destination set. The slot expires after 20 further matching source–relation events, seven days for Wiki, or 365 days for Review, whichever dataset-specific limit arrives first. Births are assigned sequentially to unused slots and cease after slot 5000 or the first quarter of the test segment, whichever occurs first; all remaining slots are then padded at one. Tied batches use the birth schedule in Appendix G.

5. The factor log is persisted before the current event is added to memory. Current-event text, rating, and other edge messages are unavailable to the predictor; they are logged after persistence but are not model inputs in this study. Each matching opportunity stores raw $\beta_{j,t}$, $p_{0,j,t}$, $Y_{j,t}$, its matching-opportunity index, and the factors for all predeclared Γ paths; path-specific eligibility is reconstructed from $\widetilde{p}^{\Gamma}_{0,j,t}$. All paths and final e-BH sets are reconstructed from this single neural pass. No test-time gradient updates are made.
6. Index test events locally by $t = 1, \ldots, T_{\text{test}}$. The single predeclared report is the final event T_{test}, and seed 20260918 defines tied-batch permutations. No reporting rule is selected after examining test outcomes.

Each negative-control replay begins from the frozen state at the end of validation. It retains only the test split boundary, static node and relation types known by that boundary, test timestamps, observed contexts C_t, and tied-batch membership. Wiki uses the complete test skeleton; Review uses the predeclared first $\min\{100{,}000, T_{\text{test}}\}$ consecutive test events. This computational restriction is applied only to replay studies, not to the observational test pass. Tied-batch permutations are regenerated from the replay seed. No observed test destination, destination-derived availability change, degree, registry entry, message, rating, neighborhood, or model state is retained; the benchmarks provide no external test-time availability table. Destinations are drawn from the declared baseline. If the $\star$ bucket is drawn, a fresh replay-only destination identifier of the compatible type is created after factor persistence. Receiver weights, the inductive registry, graph memory, candidate births, and neural states are then recomputed from the replayed history. The 100 replay seeds are 20261000–20261099 and use frozen hyperparameters. Only the first listed Lagged-THG training seed is evaluated on all 100 replays; the TGN-style ablation and degree–recency are not rerun in this implementation-calibration study. For each replay seed, `SeedSequence(seed).spawn(2)` initializes separate NumPy PCG64 streams for baseline destination draws and tied-batch permutations; all remaining replay operations are deterministic. This assesses implementation calibration under the exact replay law but does not establish that the observational baseline is true.

I.4 Semi-synthetic realistic replay

The Wiki semi-synthetic study provides a fully labeled evaluation of the complete discovery pipeline on a real Wiki event skeleton. It retains the validation-end state, complete Wiki test skeleton, observed contexts and timestamps, tied-batch membership, static types, and inductive registry rules of the exact-null replay while regenerating every destination-derived quantity. Its prespecified strong-signal regime complements the controlled power curve above. The design was frozen on the non-evaluation pilot seed 20261990; all reported results use the held-out evaluation seeds 20262300–20262319.

Before outcomes are drawn, 10% of eligible contexts are selected by the smallest SHA-256 digests of the dataset key, context, and replay seed. Each selected context receives one hidden validation-known destination, sampled by a separate seeded generator from the 20 destinations with largest pre-event reference-law mass, with probabilities renormalized within that pool. This uses neither observed test destinations nor neural scores. The hidden draw and labels are unavailable to both predictors. Destinations are generated prospectively from

$$q^{\text{ss}}_{\theta,t}(v \mid c) \propto q_{0,t}(v \mid \mathcal{H}_t) \exp\{\theta \mathbf{1}(v = v^{\star}_c)\}. \tag{I.3}$$

The frozen stress-test signal is $\theta = \log(64)$. Because the target is rare under the reference law, this multiplier should not be read as a 64-fold event probability. The stress-test family uses $K = 200$, a 250-matching-opportunity horizon, a 30-day lifetime, and $\varepsilon = 10^{-3}$; these settings apply only to this semi-synthetic study and do not alter the $K = 5000$ observational or exact-null protocols above. After each generated destination, receiver weights, registries, graph memory, candidate births, and stakes are updated exactly as in the observational pass; no observed test destination or destination-derived feature is retained.

The 20 replay seeds are crossed with the three prespecified Lagged-THG fits, trained with seeds 940002, 940004, and 940006, and four candidate–bet component arms: degree–recency or Lagged-THG candidates paired with either fixed $\beta = 0.10$ or the fit-specific neural hit-probability head. This $3 \times 4 \times 20$ design yields 240 executed cells. Test order, hidden targets, and generated destinations are shared by replay seed, so fixed-bet candidate contrasts isolate nomination and neural-minus-fixed contrasts isolate the transferred betting rule conditional on a candidate family. The model-free

Table 8: Wiki strong-signal candidate–bet arm estimates over three prespecified training fits and 20 shared replays per fit. Brackets contain 95% crossed-bootstrap intervals over fit and replay identifiers. Identical model-free degree/fixed copies are collapsed before resampling.

Candidate	Bet	Nomination recall	E2E recall	Testing power	FDP
Degree/recency	Fixed	.208 [.188,.228]	.069 [.059,.079]	.335 [.291,.378]	.000 [.000,.000]
Degree/recency	Neural	.208 [.189,.228]	.027 [.019,.036]	.138 [.096,.181]	.000 [.000,.000]
Lagged-THG	Fixed	.187 [.167,.208]	.077 [.067,.088]	.418 [.377,.459]	.008 [.000,.025]
Lagged-THG	Neural	.187 [.167,.207]	.032 [.022,.044]	.184 [.124,.254]	.000 [.000,.000]

Table 9: Paired candidate–bet contrasts in the Wiki study. Each entry is the stated left arm minus right arm with a 95% crossed-bootstrap interval.

Paired contrast	E2E	Testing power	Expected growth/update
Lagged-THG − degree (fixed bet)	.008 [−.003,.021]	.083 [.030,.140]	.059 [.042,.077]
Neural − fixed bet (degree candidates)	−.042 [−.058,−.026]	−.197 [−.266,−.129]	−.847 [−1.001,−.660]
Neural − fixed bet (Lagged-THG candidates)	−.045 [−.058,−.032]	−.234 [−.293,−.171]	−.772 [−.962,−.532]

degree/fixed copies passed an exact identity check and are collapsed to 20 unique runs. Crossed intervals resample training-fit and replay identifiers as separate factors. We additionally construct, from the fixed-candidate records, an equal-weight mixture over the six prespecified betting fractions $\{.03125, .0625, .125, .25, .5, 1\}$; no test outcome is used to select or reweight an expert. A born slot is analytically nonnull when its frozen five-node candidate contains $v_c^\star$, and final end-to-end discovery occurs only when the corresponding nonnull slot belongs to $\widehat{\mathcal{D}}(T)$. Every fit–replay cell produces at least one born nonnull slot. The separate 100-seed exact-law replays assess implementation behavior when the reference law is true.

Under the common fixed bet, the paired Lagged-THG-minus-degree differences [95% crossed interval] are $-.02069$ $[-.02931, -.01207]$ for nomination recall, $.00833$ $[-.00345, .02069]$ for end-to-end recall, and $.08333$ $[.03039, .13975]$ for conditional testing power. Expected and realized log growth per nonnull update increase by $.05893$ $[.04185, .07650]$ and $.05666$ $[.03909, .07381]$, respectively. Safe mixing raises mean conditional power and end-to-end recall for both candidate families and produces the highest mean aggregate performance with Lagged-THG candidates. Because the replay law is known, we compute the exact candidate mass p_k before every nonnull update and report S_m/m, R_m/m, their difference, and realized log-wealth growth per update. These quantities audit Theorem 7 but are never supplied to a predictor or betting rule.

The prespecified safe mixture raises the mean for both candidate families without selecting a betting strength from evaluation outcomes. Relative to the fixed bet, its conditional-power gains [95% crossed interval] are .03093 [.00740,.05646] for degree–recency candidates and .03388 [.00747,.05938] for Lagged-THG candidates; the corresponding end-to-end gains are .00603 [.00172,.01121] and .00632 [.00201,.01092]. For Lagged-THG candidates, safe-mixture conditional power ranges from .443 to .459 and end-to-end recall from .082 to .085 across the three training fits. The aggregate values are .452 conditional power, .084 end-to-end recall, and zero observed final-snapshot FDP. Each mixture satisfies the pathwise $\log 6$ regret bound in Proposition 3.

J Implementation and leakage safeguards

The auditable event loader preserves immutable identifiers, timestamps, endpoints and types, splits, tied batches, and message hashes. The risk-set engine reads only serialized pre-destination state and returns eligibility, receiver totals, and $p_0(A)$; exhaustive and indexed probabilities agree to 10^{-12} on 100 small random graphs. The simulator records the latent law, hidden blocks, and null status, and checks empirical marks against (4) and (15) with simultaneous multinomial intervals.

Before neural fitting, fixed-stake evidence is checked under the global null and the local-filtration construction is required to fail on Appendix F. Predictors see only pre-event state; suffix deletion, current-edge removal, and tied-batch tests must leave scores unchanged before revelation. Lagged-THG adds absolute time, candidate recency, relation aggregation, and recurrent state but no graph–language features. The testing module applies (9) to all K slots, including unused slots at wealth one, and matches hand-calculated rejection sets.

Null calibration precedes reference-law calibration, real-data negative controls, and semi-synthetic power. Each replay retains its identifier, status, isolated graph and PCG64 state, and a compressed Parquet log containing every matching opportunity; global events for other source–relation contexts are implicit. Matching-opportunity rows record event, batch, slot, source, relation, frozen candidate, seeds, history, p_0, raw β, outcome, matching-opportunity index, and factors; semi-synthetic rows add target, nonnull, and exact replay-probability fields. Manifests checksum the test order, destinations, records, ranker, validation-end state, and applicable reference, stake, and Γ reports. Up to eight isolated workers run on two machines. Reference probabilities, wealth, and e-BH use float64; neural computation uses float32. Each seed's validation-end states are checkpointed, cloned for replay, and used to reconstruct Γ offline.

Computational profile. The formal runs used 32 GB virtual GPUs backed by RTX 4080-class hardware and 12-core Xeon hosts. During monitoring, two concurrent ranker fits used about 13.4 GiB aggregate device memory; four concurrent neural Wiki replays used about 1.6 GiB aggregate, and one complete 23,621-event replay took approximately 5–6 minutes under that contention. These are system-level observations rather than hardware-independent benchmarks. Reference updates and queries have the exact costs stated after Equation (5); neural risk-set scoring at candidate birth is the dominant learned component and can be microbatched.

Scope and extensions. GRAPHBET is designed for applications with reconstructible risk sets and scientifically defensible opportunity probabilities, obtained for example from logged choice sets, randomized exposure, or a domain mechanism. The Γ-sensitivity analysis quantifies prespecified one-sided departures from that reference. The present implementation uses exact normalizers and a fixed slot budget, and the real benchmarks each contain one observed interaction relation. Natural extensions include corrected sampled normalizers, grouped scientific targets, and large multi-relational deployments; these change the implementation layer while retaining the prospective testing principle.