ATTRIBUTE-BALANCED FAIR INFLUENCE MAXIMIZATION VIA STRATEGIC GROUP COMPETITION

Anonymous authors
Paper under double-blind review

ABSTRACT

Fair influence maximization aims to select influential seed nodes while ensuring that information spreads fairly across different groups. Existing methods often define fairness over a single group partition or treat all groups as a flat collection. This becomes limiting when several categorical attributes are considered at the same time, because attributes with more groups may have greater influence on the selection process, and raw group utilities may not distinguish between persistent under-service and limited structural opportunity. We propose SAGE-IM, a multi-attribute fair influence maximization framework in which each attribute-group pair is modeled as a strategic agent. SAGE-IM combines opportunity-normalized deficits, cross-step regret, and hierarchical attribute balancing to adapt fairness pressure as the seed set grows. Group-specific teachers learn RR-based marginal utilities, and their preferences are distilled into a shared student through policy, ranking, and representation objectives. During sequential selection, the learned models are used to generate a compact candidate set, while the final seed choice is based on current global and group-specific reverse-reachable marginals. This design reduces reliance on learned rankings and keeps the final decision tied to the sampled diffusion structure. Experiments on eight processed real-world networks show that SAGE-IM achieves strong attribute-wise fairness and worst-group coverage while maintaining competitive influence spread. Sensitivity and ablation analyses further show how fairness dynamics, strategic coordination, attribute balancing, and active-agent selection shape the spread-fairness trade-off.

1 INTRODUCTION

Influence maximization (IM) seeks a seed set whose activation maximizes the expected diffusion of information or influence in a network (Kempe et al., 2003). Although classical IM is effective at maximizing aggregate reach, it does not account for how that reach is distributed across different groups. Consequently, optimizing only the total influence spread can disproportionately benefit well-connected or structurally advantaged groups while leaving other groups under-served (Fish et al., 2019; Tsang et al., 2019). This limitation motivates fair IM (FIM), which incorporates fairness considerations into the seed-selection process to promote a more equitable distribution of diffusion benefits across groups. The challenge becomes more pronounced when several sensitive attributes coexist: a seed set that appears balanced with respect to one attribute may still systematically under-serve groups defined by another. Moreover, treating all groups from multiple attributes within a single flat objective can implicitly assign greater importance to attributes containing more categories, since they contribute more groups to the optimization. Therefore, fairness in multi-attribute settings requires considering not only group-level disparities but also the balance of fairness across attributes.

We study multi-attribute fair IM (Feng et al., 2023) as a sequential decision problem in which fairness pressure must evolve with the current diffusion state. The central challenge is not merely to include several attributes, but to coordinate them without collapsing their distinct group structures. In addition, groups may have very different attainable diffusion opportunities. A group located in a weakly connected region should not be judged against the same raw marginal scale as a group embedded in a dense region. Fair seed selection must therefore distinguish persistent under-service from structural inaccessibility.

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.

We propose Strategic Attribute-Group Equity through Competitive Influence Maximization (SAGE-IM), which represents each attribute-group pair as a self-interested strategic agent and couples learned candidate generation with exact reverse-reachable (RR) (Borgs et al., 2014) marginal evaluation. The method first learns group-specific marginal-utility teachers from RR supervision. It then constructs an attribute-balanced deficit-regret competition in which agents are normalized by their attainable opportunity, underserved groups accumulate strategic pressure, and total strategic mass is allocated hierarchically across attributes and groups. The resulting competitive policy is distilled into a shared student through policy, ranking, and representation objectives. During sequential seed selection, learned models are used only to form a compact proposal pool; the final decision is computed from exact current RR marginals, so a learned misranking cannot by itself remove strong structural candidates.

SAGE-IM does not directly optimize fairness. Fairness is induced by four mechanisms that act during selection: within-attribute deficit, opportunity normalization, regret for missed attainable utility, and attribute-balanced strategic weighting. Main contributions are as follows:

- We formulate simultaneous multi-attribute fairness with one strategic agent per attribute-group pair and a hierarchical allocation that prevents attributes with more groups from receiving more strategic mass merely because of cardinality.
- We introduce an opportunity-normalized deficit-regret competition that combines current within-attribute service imbalance with persistent missed opportunity and dynamically concentrates computation on the worst-served and highest-priority agents.
- We learn group-specialized teachers and distill their strategically reconciled preferences into a shared student through policy, ranking, and representation supervision, while an influence-protecting gradient projection attenuates conflicts with the global influence objective.
- We retain exact RR marginals in the final decision loop and update every group tracker after each selected seed, providing a structural safeguard that is independent of learned proposal quality.

2 Related Work

IM was formalized as diffusion-aware seed selection by Kempe et al. (2003), but the classical objective optimizes aggregate reach rather than how that reach is distributed. Fair IM subsequently introduced explicit group constraints (Tsang et al. 2019), unifying optimization formulations (Farnadi, Babaki, and Gendreau 2020), and welfare-based objectives that trade efficiency against protection of disadvantaged groups (Rahmattalabi et al. 2021). Complementary approaches address fairness through representation learning, homophily-aware balancing, and randomized seed policies (Khajehnejad et al. 2020; Anwar, Saveski, and Roy 2021; Becker et al. 2022). Together, these lines establish that fairness can concern minimum group utility, balanced exposure, or equitable opportunity, and that the fairness objective can materially alter which nodes should be selected.

Recent work broadens the optimization machinery and the structural settings considered. Multi-objective and population-based methods search for spread-fairness trade-offs (Gong and Guo 2023; Zhao et al. 2024; Meena, Singh, and Biswas 2025), while multigroup welfare learning directly incorporates equity into learned decision rules (Zhang et al. 2025). Community-aware approaches exploit overlapping structure to improve scalability or local allocation objectives (Chen and Rezaeipanah 2025). In 2026, higher-order and sequential formulations further expanded the design space: FIMH considers hypergraph diffusion (Abhelakh, Mao, and Wang 2026), and deep reinforcement-learning methods use graph representations and fairness-aware rewards for sequential seed choice (Guo, Li, and Mo 2026; Saxena et al. 2026). Other studies examine temporal or adaptive fairness, robust optimization across plausible group partitions, randomized continuous optimization, and RR-assisted multi-objective search (Osmani 2026; Gao and Ito 2026; Zhang et al. 2026; Srinivas et al. 2026).

Despite this breadth, many formulations still link fairness to a fixed partition, a flat collection of groups, or a single reward or constraint that does not explicitly prevent attributes with different numbers of categories from exerting unequal influence on selection (Tsang et al., 2019; Farnadi et al., 2020; Rahmattalabi et al., 2021; Zhang et al., 2025). SAGE-IM targets a different operational

problem: simultaneous fairness across multiple observed categorical attributes. It represents each attribute-group pair as a strategic agent, allocates competition hierarchically across attributes and groups, normalizes group utility by currently attainable opportunity, and accumulates deficit-regret pressure as the seed set grows. Group-specialized teachers are distilled into a shared student for proposal generation, but the final decision does not rely on a learned score alone: current global and group-specific RR marginals are recomputed on the candidate union and drive the final competitive choice. This separation between learned proposal formation and RR-based set-conditioned selection is the main distinction from prior fairness-aware learned policies and population-based search.

3 Proposed Method

3.1 Problem Setting and Execution Order

Let $\mathcal{G} = (V, E)$ denote the fixed graph, where $N = |V|$ is the number of nodes. Let $a \in \{1, \ldots, A\}$ index the observed categorical node attributes. Attribute a has group index set $\mathcal{I}_a$; the $G = \sum_a |\mathcal{I}_a|$ observed attribute–group pairs are indexed by $g \in \{1, \ldots, G\}$, and $V_g \subseteq V$ denotes their nodes. A node may belong to one group per available attribute; missing values produce no membership for that attribute. There is no agent for an intersection of attributes. The proposed method forms the node set from edge endpoints and label-file identifiers, and removes self-loops and repeated neighbors. Group assignments are used throughout training and selection, including in features and RR roots. From the requested budgets $\mathcal{K}$, the method produces one ordered sequence $(v_1, \ldots, v_{K_{\max}})$, where $K_{\max} = max\mathcal{K}$ and $S_K = \{v_1, \ldots, v_K\}$.

The proposed method has six stages: construct global and group RR banks; build set-conditioned training examples; train a shared graph encoder and group-specific teachers; obtain strategic-game targets and train a student; select seeds with current RR-bank utilities; update all RR states and cross-step memory. Figure 1 depicts the two training and selection phases, and Algorithm 1 specifies their order. The optional depth bound, numerical thresholds, proposal counts, and other configured values appear in Appendix B; their roles are defined when introduced below. Input parsing is specified in Appendix C.1.

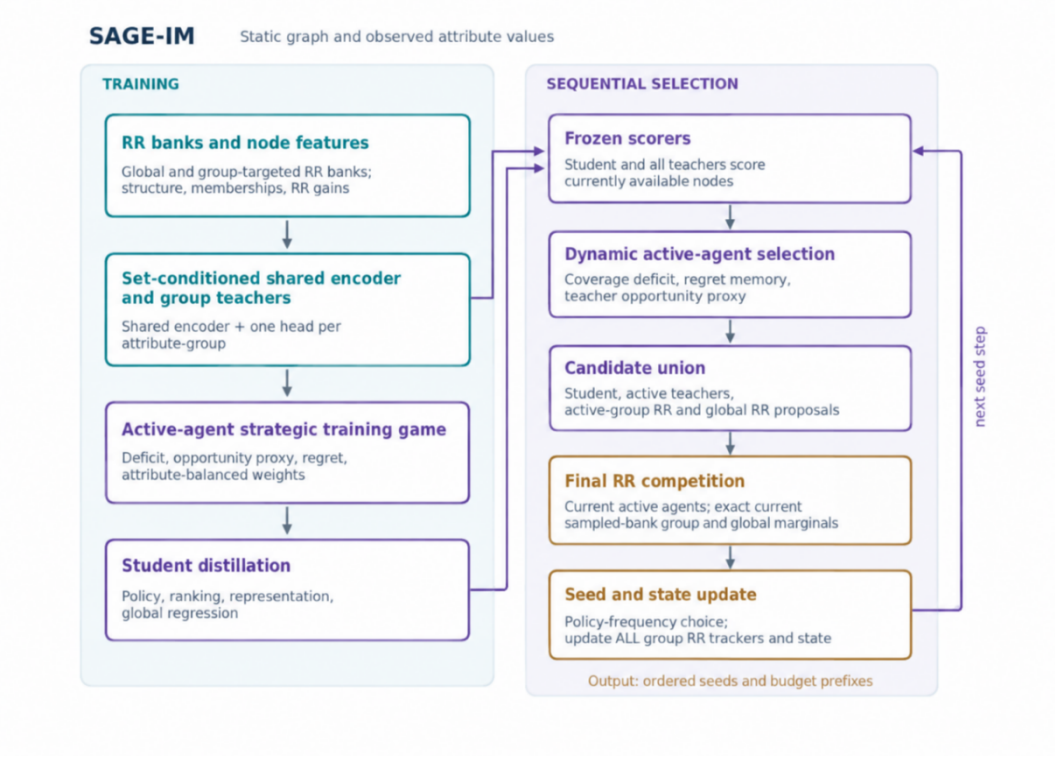

Figure 1: SAGE-IM. Training games use predicted group utilities; final selection games use current marginals of sampled RR banks over a finite proposal union. All group RR trackers are updated after every chosen seed.

3.2 GLOBAL AND GROUP RR OPPORTUNITIES

The selection model permits one independent transmission with probability β along each edge followed by immediate recovery. An RR set is generated by choosing a root and exploring live incoming arcs in reverse. The global bank $\mathcal{R}_0 = \{R_i^0\}_{i=1}^{M_0}$ samples roots uniformly over V; bank $\mathcal{R}_g = \{R_i^g\}_{i=1}^{M_g}$ samples roots uniformly over V_g. The graph, labels, and sampled banks remain fixed during training and selection. For bank $b \in \{0, 1, \ldots, G\}$, partial seed set S, and available candidate $v \notin S$, define sampled coverage and the uncovered marginal

$$\widehat{c}_b(S) = \frac{1}{M_b} \sum_{i=1}^{M_b} \mathbf{1}\{S \cap R_i^b \neq \varnothing\}, \qquad \widehat{\Delta}_b(v \mid S) = \frac{1}{M_b} \sum_{i=1}^{M_b} \mathbf{1}\{v \in R_i^b,\ S \cap R_i^b = \varnothing\}. \quad (1)$$

Here M_b is the realized bank size and $b = 0$ identifies the global bank. For sampling tolerance ϵ, failure parameter δ, and maximum budget $K_{\max}$, the proposed method calculates

$$\begin{aligned} L_{N,K} &= log\left(\textstyle\sum_{i=0}^{K} \binom{N}{i}\right), & M_{req} &= \left\lceil \frac{L_{N,K_{\max}} + log(2(G+1)/\delta)}{2\epsilon^2} \right\rceil, \\ M_0 &= min\left(M_{req}, M_{0,max}\right), & M_g &= min\left(M_{req}, M_{g,max}\right). \end{aligned} \quad (2)$$

under the configured capped mode; $M_{0,max}$ and $M_{g,max}$ are the global and per-group caps. An inverted index $\mathcal{J}_b(v) = \{i : v \in R_i^b\}$ and uncovered count $n_b(v; S) = \sum_{i \in \mathcal{J}_b(v)} \mathbf{1}\{S \cap R_i^b = \varnothing\}$ give $\widehat{\Delta}_b(v \mid S) = n_b(v; S)/M_b$. When a seed covers a previously uncovered RR set, its tracker marks that set and decrements the counts for all nodes in it. Appendix C.2 details sampling, caps, and incremental updates.

Put $V_0 = V$ and $F_b(S) = |V_b|^{-1} \sum_{u \in V_b} \Pr(u \text{ reached from } S)$ under the RR-generating one-attempt diffusion. For a fixed S and $v \notin S$, $\mathbb{E}\widehat{c}_b(S) = F_b(S)$ and $\mathbb{E}\widehat{\Delta}_b(v \mid S) = F_b(S \cup \{v\}) - F_b(S)$. For each realized bank, $\widehat{c}_b$ is monotone submodular and its tracker returns its exact finite-bank marginal.

The expectation concerns RR generation; it gives no uniform-accuracy statement for the capped banks. A proof appears in Appendix D.

3.3 SET-CONDITIONED EXAMPLES AND GROUP TEACHERS

Let d_v be the number of forward neighbors of node v, $m_{vg} = \mathbf{1}\{v \in V_g\}$, and $q_v = \sum_g m_{vg}$. Node v has the feature vector

$$x_v = \left[\frac{\log(1 + d_v) - \mu_d}{\sigma_d + \tau_d}, \left(\frac{m_{vg}}{\sqrt{\max(q_v, 1)}} \right)_{g=1}^{G}, \widehat{c}_0(\{v\}), (\widehat{c}_g(\{v\}))_{g=1}^{G} \right], \quad (3)$$

where μ_d, σ_d are the mean and standard deviation of $\log(1 + d_v)$ across nodes, τ_d guards the scale, and the last $G + 1$ entries are singleton RR-bank coverage. In particular, the normalized membership features use observed labels. Let X collect these vectors by row and let P average the incoming neighbors of each node, assigning a zero row to isolated nodes. A shared two-layer graph encoder computes

$$\begin{aligned} H^{(1)} &= Drop\left(GELU\left(LN_1\left(XW_s^{(1)} + PXW_n^{(1)} + b_s^{(1)} + b_n^{(1)}\right)\right)\right), \\ Z &= H^{(1)} + GELU\left(LN_2\left(H^{(1)}W_s^{(2)} + PH^{(1)}W_n^{(2)} + b_s^{(2)} + b_n^{(2)}\right)\right), \end{aligned} \quad (4)$$

where Z has node rows $z_v \in \mathbb{R}^H$, $W_s^{(\updownarrow)}, W_n^{(\updownarrow)}$ and $b_s^{(\updownarrow)}, b_n^{(\updownarrow)}$ are distinct learned self and neighbor maps and biases at layer $\updownarrow$, $LN_\updownarrow$ is layer normalization, $GELU$ is the activation, and $Drop$ acts only after the first layer during training. Its probability and width H are in Appendix B.

To train on partial decisions, the proposed method builds a base pool from the highest singleton global RR scores, highest singleton scores for each group, and uniformly sampled nodes. For each of

B states, it draws $|S_s|$ uniformly from $\{0, \ldots, min\,(K_{state}, N - C)\}$ and samples that many seeds uniformly without replacement. Here K_{state} is the maximum training-state size and C is the number of candidates per state. From the base pool outside S_s, it takes up to $\lfloor C/2 \rfloor$ candidates with the highest score $b(v) = \widehat{c}_0\,(\{v\}) + G^{-1} \sum_g \widehat{c}_g\,(\{v\})$; it samples the remaining candidates uniformly from unselected nodes outside that guided subset and shuffles them to form $\mathcal{C}_s$. Exact sampled-bank marginals $\widehat{\Delta}_b\,(v \mid S_s)$ are computed for every $v \in \mathcal{C}_s$ and every bank b. The base-pool counts are in Appendix B.

For state S, let $\bar{z}_S = |S|^{-1} \sum_{u \in S} z_u$ (zero when $S = \varnothing$), $r(S) = \left[1 - \widehat{c}_0(S), (1 - \widehat{c}_g(S))_{g=1}^{G}\right]$, and

$$\phi(v, S) = [z_v, \bar{z}_S, z_v \odot \bar{z}_S, r(S)]\,. \tag{5}$$

The product $\odot$ is coordinatewise. Each group has its own head $a_g = GELU\,(W_{g,1}\phi + b_{g,1})$, $e_g = GELU\,(W_{g,2}a_g + b_{g,2})$, and $\widehat{u}_g(v, S) = sigmoid\left(w_g^{\mathsf{T}} e_g + b_{g,o}\right)$. Its learned matrices, vectors, and biases are specific to g; $e_g \in \mathbb{R}^D$ is also a later distillation target, with head width D. The shared encoder and all heads are trained together on each group's own RR marginal by minimizing

$$\mathcal{L}_{teach} = \frac{1}{BCG} \sum_{s=1}^{B} \sum_{v \in \mathcal{C}_s} \sum_{g=1}^{G} \left(\widehat{u}_g\,(v, S_s) - \widehat{\Delta}_g\,(v \mid S_s)\right)^2. \tag{6}$$

After training, the encoder and teachers are evaluated without dropout. Appendix C.3 specifies the sampler and optimizer details.

3.4 Active agents and attribute-balanced competition

Write $c_g = \widehat{c}_g(S)$ for attained coverage and $\mathcal{I}_a$ for the groups of attribute a. Let $c_a^{\max} = \max_{h \in \mathcal{I}_a} c_h$ and $c_a^{\min} = \min_{h \in \mathcal{I}_a} c_h$. Its within-attribute deficit is

$$d_g = \begin{cases} \left(c_a^{\max} - c_g\right) / \left(c_a^{\max} - c_a^{\min}\right), & c_a^{\max} - c_a^{\min} > \tau, \\ c_a^{\max} - c_g, & c_a^{\max} - c_a^{\min} \le \tau, \end{cases} \qquad g \in \mathcal{I}_a. \tag{7}$$

Here τ is a numerical zero threshold; the proposed method retains tiny unnormalized gaps when the range is at most τ. Let $m_g \ge 0$ be cross-step regret memory, initially zero, and let U_{jg}^{proxy} be teacher g's utility for available action j in a preliminary matrix of J actions. At training this matrix contains the C candidates of one state, whereas at selection it contains *all* remaining nodes before proposal truncation. The opportunity-concentration proxy is

$$b_g^{proxy} = \max_j \max\left(U_{jg}^{proxy}, 0\right), \qquad \rho_g = \begin{cases} \max\left(0, 1 - \frac{1}{J} \sum_{j=1}^{J} \frac{\max\left(U_{jg}^{proxy}, 0\right)}{b_g^{proxy}}\right), & b_g^{proxy} > \tau, \\ 0, & \text{otherwise.} \end{cases} \tag{8}$$

For a nonnegative signal x, $\mathcal{N}_a\,(x_g) = x_g / \max_{h \in \mathcal{I}_a} x_h$ when this maximum exceeds τ, and zero otherwise. The active priority is

$$p_g^{act} = \mathcal{N}_a\,(m_g) + \lambda_{proxy} \mathcal{N}_a\,(\rho_g) + \lambda_{def} d_g. \tag{9}$$

The coefficients λ_{proxy} and λ_{def} control proxy and deficit contributions. For each attribute, the union of its top R groups under p_g^{act} and one minimum-coverage group forms the active set $\mathcal{A}$; R is the per-attribute activity cap before the worst group is added. Ties rotate with the state or seed-step index. When all attained coverages and memory entries are numerically zero, the active set contains all G agents regardless of the proxy. The training games have no cross-step memory; the final selector maintains m_g. All group RR trackers are updated even if their groups are inactive.

For a candidate v_j in a game's action set, let U_{jg} and U_{j0} be its group and global utilities. In a *training* game, $U_{jg} = \widehat{u}_g\,(v_j, S_s)$ and $U_{j0} = \widehat{\Delta}_0\,(v_j \mid S_s)$. In a *selection* game, both sources are sampled-bank marginals: $U_{jg} = \widehat{\Delta}_g\,(v_j \mid S)$ for active g and $U_{j0} = \widehat{\Delta}_0\,(v_j \mid S)$. In either game, $\widetilde{U}_{jb} = U_{jb} / \max_i U_{ib}$ when that maximum exceeds τ, and zero otherwise, for $b \in \mathcal{A} \cup \{0\}$; inactive groups have zero normalized utility and zero weight. Thus an attainable opportunity of a group is scaled by its best action *within the current candidate set*, and no teacher prediction replaces a group RR marginal in final game scoring.

Within each game call, the achieved coverage and d_g stay fixed, while cumulative within-game regret is reset to $R_{g,0} = 0$. At round $t \geq 1$, let $\mathcal{A}_a = \mathcal{A} \cap \mathcal{I}_a$, $p_{g,t} = R_{g,t-1}/t + \lambda_{def} d_g$, and $P_{a,t} = \max_{g \in \mathcal{A}_a} p_{g,t}$. With softmax scale η, conditional floor $f_a = min(\kappa, \xi/|\mathcal{A}_a|)$, and $s_{g|a,t} = exp(\eta p_{g,t}) / \sum_{h \in \mathcal{A}_a} \exp(\eta p_{h,t})$, the base mass is

$$\mu_{a,t} = softmax_{a'}(\eta P_{a',t})_a, \qquad \alpha_{g,t}^{base} = \mu_{a,t}\{(1 - |\mathcal{A}_a| f_a) s_{g|a,t} + f_a\}, \quad g \in \mathcal{A}_a. \tag{10}$$

Here $\mu_{a,t} = exp(\eta P_{a,t}) / \sum_{a'} \exp(\eta P_{a',t})$ allocates mass to attributes; κ, ξ limit the floor of each active group's *conditional* mass. If the preceding seed step supplied positive weight on the present active set, restrict and renormalize it to $\bar{\alpha}^{prev}$, then set $\alpha_{g,t} = t\alpha_{g,t}^{base}/(t+1) + \bar{\alpha}_g^{prev}/(t+1)$; otherwise use $\alpha_{g,t}^{base}$. Weights of inactive groups are zero. This warm start decays within a game call, rather than updating its source after each inner round. The action and its regret update are

$$j_t = arg\max_j \left[\sum_{g \in \mathcal{A}} \alpha_{g,t} \widetilde{U}_{jg} + \lambda_{\text{inf}} \widetilde{U}_{j0} \right], \qquad R_{g,t} = R_{g,t-1} + max\left(0, \max_j \widetilde{U}_{jg} - \widetilde{U}_{j_t g}\right), \tag{11}$$

where λ_{inf} weights global normalized opportunity and the action tie goes to the first candidate. After a minimum of $T_{\min}$ rounds, the game stops if $\max_{g \in \mathcal{A}} R_{g,t}/t \leq \tau_{regret}$, and otherwise runs up to $T_{\max}$ rounds; $T_{\min}, T_{\max}$ and τ_{regret} are its round bounds and stopping tolerance. For T executed rounds, the empirical action frequency is $\pi_j = \sum_{t=1}^{T} \mathbf{1}\{j_t = j\}/(T + \tau_{count})$; $q_j = \sum_{g \in \mathcal{A}} \alpha_{g,T} \widetilde{U}_{jg} + \lambda_{\text{inf}} \widetilde{U}_{j0}$ is the last-round score, and τ_{count} guards division. The floor and warm start have distinct mass implications:

With exact probability normalization and $0 \leq f_a \leq 1/|\mathcal{A}_a|$, $\sum_{g \in \mathcal{A}_a} \alpha_{g,t}^{base} = \mu_{a,t}$ and $\alpha_{g,t}^{base} \geq \mu_{a,t} f_a$. Warm starting preserves total mass one but may change each attribute's mass. The proposed method renormalizes with small denominator guards in floating-point arithmetic.

The proof is in Appendix D; this is an identity of the weights, not a fairness guarantee.

3.5 Student distillation and influence-gradient adjustment

The training game is run independently for each state. It supplies action-frequency targets $\pi_{s,j}$, final scores $q_{s,j}$, and final agent weights $\alpha_{s,g}$; inactive groups have zero weight. The representation target is $e_{s,j}^* = \sum_g \alpha_{s,g} e_g(v_j, S_s)$, where e_g is teacher g's second hidden activation. A separate student graph encoder is initialized with the trained teacher encoder's weights and then trained with its own parameters. Given its own conditioned input $\phi_{s,j}^{stu}$ from Eq. (5), its head computes

$$\begin{array}{ll} a_{s,j}^{stu} & = GELU\left(W_{S,1}\phi_{s,j}^{stu} + b_{S,1}\right), \quad e_{s,j}^{stu} = GELU\left(W_{S,2}a_{s,j}^{stu} + b_{S,2}\right), \\ \updownarrow_{s,j} & = w_P^{\mathsf{T}} e_{s,j}^{stu} + b_P, \qquad \widehat{u}_{s,j}^0 = sigmoid\left(w_0^{\mathsf{T}} e_{s,j}^{stu} + b_0\right), \end{array} \tag{12}$$

where $W_{S,1}, W_{S,2}, w_P, w_0$ and their biases are learned student parameters, $\updownarrow_{s,j}$ is the policy logit, and $\widehat{u}_{s,j}^0$ predicts the global RR marginal. The latter provides an auxiliary training objective rather than a final decision score. The student learns the global regression $\mathcal{L}_I = (BC)^{-1} \sum_{s,j} \left(\widehat{u}_{s,j}^0 - \widehat{\Delta}_0(v_j \mid S_s)\right)^2$ and distillation

$$\begin{array}{rl} \mathcal{L}_D & = \lambda_P \mathcal{L}_P + \lambda_R \mathcal{L}_R + \lambda_E \mathcal{L}_E, \\ \mathcal{L}_P & = -B^{-1} \sum_{s,j} \pi_{s,j} logsoftmax(\updownarrow_{s,\cdot})_j, \\ \mathcal{L}_R & = |\mathcal{P}|^{-1} \sum_{(s,h,l) \in \mathcal{P}} softplus\left[-(\updownarrow_{s,h} - \updownarrow_{s,l})\right], \\ \mathcal{L}_E & = \omega_{mse} MSE(e^{stu}, e^*) + \omega_{\cos}\left[1 - mean_{s,j} \cos\left(e_{s,j}^{stu}, e_{s,j}^*\right)\right]. \end{array} \tag{13}$$

Here $\mathcal{P}$ contains every within-state unordered pair oriented so that $q_{s,h} > q_{s,l}$ with gap at least τ_{rank}; the ranking loss is zero if $\mathcal{P}$ is empty. The terms $\lambda_P, \lambda_R, \lambda_E$ weigh policy, ranking, and representation distillation, while $\omega_{mse}, \omega_{\cos}$ weigh the two representation losses; B is the number of training states. The head width and all coefficients are reported in Appendix B.

For each student parameter block p, write $g_{I,p} = \nabla_p \mathcal{L}_I$ and $g_{D,p} = \nabla_p \mathcal{L}_D$. Over the blocks $\mathcal{O}$ where both gradients exist, compute $d_{ID} = \sum_{p \in \mathcal{O}} \langle g_{I,p}, g_{D,p} \rangle$ and $Q_I = \sum_{p \in \mathcal{O}} \| g_{I,p} \|^2$. If $d_{ID} < 0$, replace $g_{D,p}$ by $g_{D,p} - d_{ID} g_{I,p}/(Q_I + \tau_{proj})$; otherwise leave it. Here τ_{proj} is a denominator guard.

Missing block gradients are set to zero *after* defining $\mathcal{O}$, and this conditional adjustment is applied to every parameter block. The optimizer receives the norm-clipped $g_{I,p} + \lambda_D g_{D,p}^{adj}$, where λ_D weights distillation, and applies an Adam update. On shared blocks the negative inner product is attenuated to $d_{ID}\tau_{proj}/\left(Q_I + \tau_{proj}\right)$; this algebra does not guarantee improved spread after optimization. Appendix D gives the proof and Appendix C.5 gives details.

3.6 RR-BASED SEQUENTIAL SEED SELECTION

After training, the teacher and student encoders and heads run in inference mode. At step k, with $S = S_{k-1}$, both the student and *every* teacher score all nodes in $V \backslash S$ using their own embeddings, the pooled embedding of S, and current remaining RR fractions. Teacher scores over these available nodes determine the proxy and active set before the proposal pool is built. Write $Top_P(f; V\backslash S)$ for the up-to-P remaining nodes with highest score f. The deduplicated candidate union is

$$\begin{aligned} \mathcal{C}(S) = \quad & Top_{P_S}\left(\updownarrow^{stu}; V\backslash S\right) \cup Top_{P_0}\left(\widehat{\Delta}_0; V\backslash S\right) \\ & \cup \bigcup\nolimits_{g\in\mathcal{A}} \left[Top_{P_T}\left(\widehat{u}_g; V\backslash S\right) \cup Top_{P_G}\left(\widehat{\Delta}_g; V\backslash S\right)\right]. \end{aligned} \tag{14}$$

Here P_S is the student proposal cap, P_T the per-active-teacher cap, P_G the per-active-group RR cap, and P_0 the global RR cap. Each $\widehat{\Delta}_b$ is the *current uncovered* marginal from the corresponding tracker. The union is sorted by node identifier after duplicates are removed; it need not contain every remaining node. The final game uses $\widehat{\Delta}_g(v \mid S)$ and $\widehat{\Delta}_0(v \mid S)$ over this union, with current group coverage $\widehat{c}_g(S)$ and the preceding game's weights as an optional warm start. It chooses a candidate with the largest π_j; numerical ties are resolved by q_j, and any remaining tie by candidate order.

The winning node becomes v_k. The global tracker and *all* group trackers mark its newly covered RR sets, including inactive groups. For final-game average regret $\bar{R}_g = R_{g,T}/T$, persistent memory is updated by

$$m_g^+ = \begin{cases} \mu m_g + (1-\mu)\bar{R}_g, & g\in\mathcal{A}, \\ \mu m_g, & g\notin\mathcal{A}, \end{cases} \tag{15}$$

where μ retains earlier memory across seed steps. The final agent weights become the next game's optional warm start. This outer memory is different from the within-game regret $R_{g,t}$, which is reset at every game call. Requested budgets use prefixes of the same sequence; the independent diffusion evaluator does not feed back into these decisions. Parameter values are in Appendix B.

Algorithm 1: SAGE-IM training and selection
Input: graph, observed attributes, RR-model probability β, budgets $\mathcal{K}$.
Output: ordered seeds and prefixes S_K for $K\in\mathcal{K}$.
1. Form V_g for each observed pair (a, g); sample $\mathcal{R}_0, \mathcal{R}_g$ and build their inverted indices.
2. Form features x_v; sample $(S_s, \mathcal{C}_s)$ and targets $\widehat{\Delta}_b\left(v \mid S_s\right)$ for all banks.
3. Train the shared encoder and one teacher per group on group RR marginals.
4. In each state, select active groups and run the teacher-utility game; save π, q, α.
5. Form weighted teacher representations; train a separate student with policy, ranking, representation, and global-RR losses and the conditional gradient adjustment.
6. Initialize $S_0 = \varnothing$, zero memory m, uncovered RR trackers, and no warm-start weights.
7. **For** $k = 1, \ldots, K_{\max}$: score all available nodes; activate groups using teacher utilities, coverage, and memory; form $\mathcal{C}\left(S_{k-1}\right)$; compute its current RR marginals; run the final game; choose the modal action (break ties by final score); update m, the warm start, and every RR tracker.
8. **Return** the one ordered sequence and its prefixes S_K for $K\in\mathcal{K}$.

4 EVALUATION AND RESULTS

Data and comparison protocol. We evaluate on eight processed graphs. Table 1 reports their graph size and attribute structure. High-School-2013 is a contact network (Mastrandrea, Fournet, and Barrat 2015); Od is derived from learning-analytics records (Kuzilek, Hlosta, and Zdrahal 2017); Ego-Facebook uses a social-circle graph (McAuley and Leskovec 2012); MovieLens uses a constructed user-similarity graph (Harper and Konstan 2015); DBLP uses a processed collaboration graph (Yang and Leskovec 2012). Pokec, AH, and APS complete the binary-attribute cases. Appendix E describes construction and exact group sizes.

Table 1: Processed graph sizes and group cardinalities. Group counts follow attribute order.

Dataset	Nodes	Edges	Attributes	Groups per attribute
High-School-2013	327	5,818	2	9, 3
Od	2,302	18,484	9	2, 13, 5, 11, 3, 2, 4, 4, 2
Ego-Facebook	4,039	88,234	8	2, 2, 2, 2, 2, 2, 2, 2
MovieLens	6,040	152,225	3	2, 7, 21
DBLP	5,000	27,322	1	15
Pokec	5,000	48,454	1	2
AH	1,997	8,523	1	2
APS	1,281	3,064	1	2

We compare against nine baselines: Log (Ali et al., 2021), Sqrt (Ali et al., 2021), MOEIM (Cunegatti et al., 2024), S3D-OT (Chowdhary et al., 2024), Tsang (Tsang et al., 2019), DeepIM (Ling et al., 2023), FAC (Feng et al., 2023), FPS (Feng et al., 2023), and CrossWalk (Khajehnejad et al., 2022). Our *primary* budget sweeps use evaluation transmission probability $p_{eval} = 0.01$ at $K = 4, 8, \ldots, 40$. This choice reduces the ceiling effect in coverage-based fairness measures and leaves more room to distinguish methods on entropy, worst-group coverage, and dispersion. At larger transmission probabilities, many reported entropy values approach one, which can conceal differences relevant to a fairness-focused comparison. A near-one entropy alone does not establish near-one worst-group coverage.

As a supplementary propagation check, the seed sets selected by proposed method are evaluated at $p_{eval} \in \{0.03, 0.05\}$ for $K \in \{20, 40\}$. Appendix B lists the known training, selection, and default-evaluation settings.

Measures. The independent evaluator in the proposed method uses one-attempt SIR diffusion model (Kermack et al., 1927) and computes normalized mean reached nodes $F(K)$ and the reached fraction $C_g(K)$ of each group. For R_{MC} realizations, reached total $I^{(r)}$, and reached group count $n_g^{(r)}$ in realization r,

$$F(K) = \frac{1}{R_{MC} N} \sum_{r=1}^{R_{MC}} I^{(r)}\left(S_K\right), \qquad C_g(K) = \frac{1}{R_{MC}\left|V_g\right|} \sum_{r=1}^{R_{MC}} n_g^{(r)}\left(S_K\right). \tag{16}$$

For an attribute with $m_a = |\mathcal{I}_a|$ groups, normalize its group coverage vector as $p_{ag} = C_g / \sum_{h \in \mathcal{I}_a} C_h$. Its entropy is $H_a = -\sum_{g: p_{ag} > 0} p_{ag} \log p_{ag} / log m_a$; the reported macro score is $H = A^{-1} \sum_a H_a$. The evaluator sets $H_a = 1$ for a single group and $H_a = 0$ when a multi-group attribute has zero total coverage. It also computes worst-group coverage and normalized within-attribute L1 dispersion as detailed in Appendix C.6. None of these fairness measures enters the training or selection loss directly. Entropy near one can coexist with low $F(K)$ because it describes balance of group coverage fractions, not their total level.

Primary fairness analysis at $\mathbf{p_{eval} = 0.01}$. The eight budget-sweep figures in Appendix H place normalized spread on the horizontal axis and show entropy, worst-group coverage, and normalized L1 dispersion on separate vertical axes. Higher is better for entropy and worst-group coverage; lower is better for dispersion. Figure 2 highlights two worst-group trade-offs under the primary protocol. High-School-2013 and Od end with high entropy and worst-group coverage and low dispersion relative to the plotted alternatives. DBLP attains markedly higher worst-group coverage but does not attain the rightmost spread endpoint. On Ego-Facebook, CrossWalk has a smaller endpoint dispersion, while SAGE-IM has greater spread, entropy, and worst-group coverage. These plots illustrate why the three fairness measures are analyzed jointly. The trajectories are evaluated at 0.01. Appendix H examines every graph.

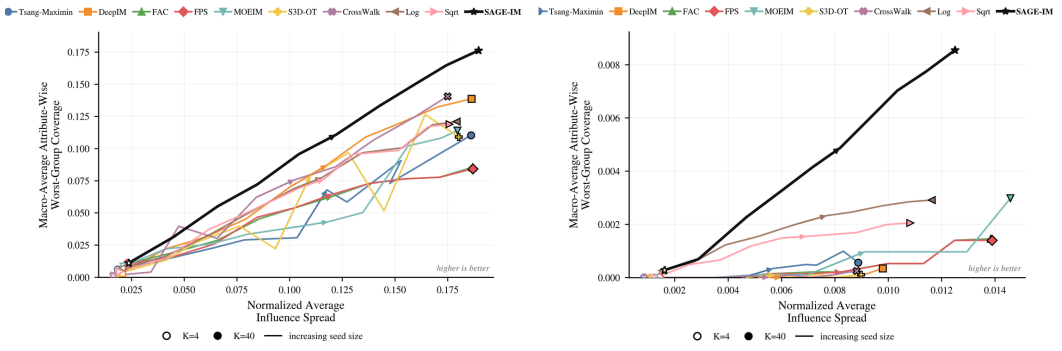

Figure 2: Primary $p_{eval} = 0.01$ budget sweeps: macro-average attribute-wise worst-group coverage versus normalized spread for High-School-2013 (left) and DBLP (right). Each path varies K from 4 to 40; full three-measure plots for all graphs appear in Appendix H.

5 Conclusion

SAGE-IM addresses multi-attribute fair influence maximization by combining sequential deficit-regret competition, hierarchical attribute balancing, group-specific teachers, student distillation, and reverse-reachable (RR) based seed selection. Unlike approaches that treat all groups as a single flat collection, SAGE-IM coordinates attribute-group agents while preventing attributes with more categories from receiving disproportionate strategic influence. Learned teacher and student models are used to construct candidate proposals, whereas the final selection decision relies on current global and group-specific RR marginals, maintaining a direct connection to the sampled diffusion structure. Experiments across eight processed networks show that SAGE-IM can improve attribute-wise fairness and worst-group coverage while maintaining competitive influence spread. The results also demonstrate that fairness should not be assessed using a single measure. Entropy, worst-group coverage, and normalized dispersion can exhibit different behaviors, particularly when propagation probabilities increase and entropy approaches its upper bound. The sensitivity analysis further reveals a clear interaction between influence and fairness: strengthening the global-influence contribution can increase spread while reducing minimum group coverage, whereas weakening the deficit mechanism can substantially reduce fairness on some datasets. The structural ablations indicate that competitive coordination, fairness dynamics, learned proposal mechanisms, attribute balancing, and dynamic agent selection affect the resulting spread-fairness trade-off in different ways. Future work can investigate stronger statistical guarantees, improved scalability, alternative agent-selection strategies, and extensions to partially observed or richer group structures.

AI Use Statement

Generative AI used for grammar check and proofreading.

REFERENCES

Abhelakh, Z., T. Mao, and H. Wang. 2026. "Fair Influence Maximization in Hypergraphs." arXiv:2606.14189. https://arxiv.org/abs/2606.14189.

Ali, J., Babaei, M., Chakraborty, A., Mirzasoleiman, B., Gummadi, K. P., & Singla, A. (2021). On the fairness of time-critical influence maximization in social networks. IEEE Transactions on Knowledge and Data Engineering, 35(3), 2875-2886.

Anwar, Muhammad S., Martin Saveski, and Deb Roy. 2021. "Balanced Influence Maximization in the Presence of Homophily." In *WSDM*, 175–83. https://doi.org/10.1145/3437963.3441787.

Becker, Ruben, Gianlorenzo D'Angelo, Sajjad Ghobadi, and Hugo Gilbert. 2022. "Fairness in Influence Maximization Through Randomization." *Journal of Artificial Intelligence Research* 73: 1251–83. https://doi.org/10.1613/jair.1.13367.

Borgs, C., Brautbar, M., Chayes, J., & Lucier, B. (2014, January). Maximizing social influence in nearly optimal time. In Proceedings of the twenty-fifth annual ACM-SIAM symposium on Discrete algorithms (pp. 946-957). Society for Industrial and Applied Mathematics.

Chen, L., and A. Rezaeipanah. 2025. "SFIMCO: Scalable Fair Influence Maximization Based on Overlapping Communities and Optimization Algorithms." *Neurocomputing* 629: 129687. https://doi.org/10.1016/j.neucom.2025.129687.

Chowdhary, Shreya, Giulia De Pasquale, Nicolas Lanzetti, Ana-Andreea Stoica, and Florian Dörfler. 2024. "Fairness in Social Influence Maximization via Optimal Transport." In *Advances in Neural Information Processing Systems*.

Cunegatti, E., Custode, L., & Iacca, G. (2024, July). Many-objective evolutionary influence maximization: Balancing spread, budget, fairness, and time. In Proceedings of the genetic and evolutionary computation conference companion (pp. 655-658).

Farnadi, Golnoosh, Behrouz Babaki, and Michel Gendreau. 2020. "A Unifying Framework for Fairness-Aware Influence Maximization." In *The Web Conference Companion*, 714–22. https://doi.org/10.1145/3366424.3383555.

Feng, Y., Patel, A., Cautis, B., & Vahabi, H. (2023, August). Influence maximization with fairness at scale. In Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining (pp. 4046-4055).

Fish, B., Bashardoust, A., Boyd, D., Friedler, S., Scheidegger, C., & Venkatasubramanian, S. (2019, May). Gaps in information access in social networks?. In The world wide web conference (pp. 480-490).

Gao, T., and T. Ito. 2026. "Robust Fair Influence Maximization under Multiple Community Partitions." Proceedings of the ACM on Management of Data 4 (1): Article 78, 1–25. https://doi.org/10.1145/3786692.

Gong, H., and C. Guo. 2023. "Influence Maximization Considering Fairness: A Multi-Objective Optimization Approach with Prior Knowledge." *Expert Systems with Applications* 214: 119138. https://doi.org/10.1016/j.eswa.2022.119138.

Guo, P., H. Li, and X. Mo. 2026. "Higher-Order Learning for Fairness: A Reinforcement Learning Framework for Equitable and Efficient Influence Maximization." Journal of King Saud University - Computer and Information Sciences 38: Article 403. https://doi.org/10.1007/s44443-026-00798-0.

Harper, F. Maxwell, and Joseph A. Konstan. 2015. "The MovieLens Datasets: History and Context." *ACM Transactions on Interactive Intelligent Systems* 5 (4): 19. https://doi.org/10.1145/2827872.

Kempe, David, Jon Kleinberg, and Éva Tardos. 2003. "Maximizing the Spread of Influence Through a Social Network." In *KDD*, 137–46. https://doi.org/10.1145/956750.956769.

Kermack, W. O., & McKendrick, A. G. (1927). A contribution to the mathematical theory of epidemics. Proceedings of the royal society of london. Series A, Containing papers of a mathematical and physical character, 115(772), 700-721.

Khajehnejad, M., A. A. Rezaei, M. Babaei, J. Hoffmann, M. Jalili, and A. Weller. 2020. "Adversarial Graph Embeddings for Fair Influence Maximization over Social Networks." In *IJCAI*, 4306–12. https://doi.org/10.24963/ijcai.2020/594.

Khajehnejad, A., Khajehnejad, M., Babaei, M., Gummadi, K. P., Weller, A., & Mirzasoleiman, B. (2022, June). Crosswalk: Fairness-enhanced node representation learning. In Proceedings of the AAAI Conference on Artificial Intelligence (Vol. 36, No. 11, pp. 11963-11970).

Kuzilek, Jakub, Martin Hlosta, and Zdenek Zdrahal. 2017. "Open University Learning Analytics Dataset." *Scientific Data* 4: 170171. https://doi.org/10.1038/sdata.2017.171.

Lee, Eun, Fariba Karimi, Claudia Wagner, Hang-Hyun Jo, Markus Strohmaier, and Mirta Galesic. 2019. "Homophily and Minority-Group Size Explain Perception Biases in Social Networks." *Nature Human Behaviour* 3: 1078–87. https://doi.org/10.1038/s41562-019-0677-4.

Mastrandrea, Rossana, Julie Fournet, and Alain Barrat. 2015. “Contact Patterns in a High School: A Comparison Between Data Collected Using Wearable Sensors, Contact Diaries and Friendship Surveys.” *PLOS ONE* 10 (9): e0136497. https://doi.org/10.1371/journal.pone.0136497.

Leskovec, J., & Mcauley, J. (2012). “Learning to discover social circles in ego networks”. Advances in neural information processing systems, 25.

Ling, C., Jiang, J., Wang, J., Thai, M. T., Xue, R., Song, J., ... & Zhao, L. (2023, July). Deep graph representation learning and optimization for influence maximization. In International conference on machine learning (pp. 21350-21361). PMLR.

Meena, S. K., K. Singh, and B. Biswas. 2025. “Fairness-Aware Influence Maximization: A Novel Learning Automata-Based Approach.” Expert Systems with Applications 280: 127445. https://doi.org/10.1016/j.eswa.2025.127445.

Osmani, A. 2026. “Adaptive Fairness-Aware Influence Maximization via Multi-Objective Optimization and Step-Wise Entropy.” Information Sciences 731: 122895. https://doi.org/10.1016/j.ins.2025.122895.

Rahmattalabi, Aida, Shahin Jabbari, Himabindu Lakkaraju, Phebe Vayanos, Max Izenberg, Ryan Brown, Eric Rice, and Milind Tambe. 2021. “Fair Influence Maximization: A Welfare Optimization Approach.” In *AAAI*, 11630–38. https://doi.org/10.1609/aaai.v35i13.17383.

Saxena, A., H. K. Yadav, B. Rutten, and S. S. Jha. 2026. “DQ4FairIM: Fairness-Aware Influence Maximization Using Deep Reinforcement Learning.” IEEE Transactions on Computational Social Systems. Advance online publication. https://doi.org/10.1109/TCSS.2026.3705520.

Srinivas, A. J., P. Potikas, W. Andreopoulos, and K. Potika. 2026. “Fair Influence Maximization with Reverse Influence Sampling Boosted Multi-Objective Genetic Algorithm.” In 2026 IEEE 12th International Conference on Big Data Computing Service and Machine Learning Applications (BigDataService), 138–145. IEEE. https://doi.org/10.1109/BigDataService70481.2026.00027.

Tsang, Alan, Bryan Wilder, Eric Rice, Milind Tambe, and Yair Zick. 2019. “Group-Fairness in Influence Maximization.” In *IJCAI*, 5997–6005. https://doi.org/10.24963/ijcai.2019/831.

Yang, Jaewon, and Jure Leskovec. 2012. “Defining and Evaluating Network Communities Based on Ground-Truth.” In *KDD Workshop on Mining Data Semantics*. https://doi.org/10.1145/2350190.2350193.

Zhang, Y., S. Chen, L. Du, Z. Zhang, and W. Yang. 2026. “Fairness-Aware Influence Maximization with Randomized Strategies: A Stochastic Frank-Wolfe Framework.” ACM Transactions on Knowledge Discovery from Data. Advance online publication. https://doi.org/10.1145/3827606.

Zhang, Z., D. Li, Y. Wang, W. Chen, and Y. Zhu. 2025. “Fairness-Constrained Multigroup Influence Maximization.” *Knowledge and Information Systems* 67: 3487–3511. https://doi.org/10.1007/s10115-024-02314-0.

Zhao, Z., W. Li, J. Ma, J. Jiang, and Q. Bai. 2024. “Fair Influence Maximization in Social Networks: A Group-Fairness-Aware Multi-Objective Grey Wolf Optimizer.” In *IEEE International Conference on Agents*, 88–93. https://doi.org/10.1109/ICA63002.2024.00026.

A Appendix/Supplementary: Complete Specification of the Proposed Method

This section expands the proposed method's equations, training and selection sequence, and numerical implementation.

A.1 Global and group-specific RR opportunities

We use a one-attempt diffusion model with transmission probability β on each edge and immediate recovery. To estimate the opportunities of the whole graph and its groups, we construct a global bank $\mathcal{R}_0 = \{R_i^0\}_{i=1}^{M_0}$ whose roots are uniform over V and a bank $\mathcal{R}_g = \{R_i^g\}_{i=1}^{M_g}$ for each agent g whose roots are uniform over V_g. An RR set is obtained by exploring incoming arcs from its root and retaining each tested arc with probability β. For a partial seed set S, the sampled-bank coverage fraction is

$$\widehat{c}_b(S) = \frac{1}{M_b} \sum_{i=1}^{M_b} \mathbf{1}\{S \cap R_i^b \neq \varnothing\}, \qquad b \in \{0, 1, \ldots, G\}. \tag{17}$$

The global fraction $\widehat{c}_0(S)$ estimates normalized spread; $\widehat{c}_g(S)$ estimates the reached fraction of V_g. The uncovered RR marginal of candidate v is

$$\widehat{\Delta}_b(v \mid S) = \frac{1}{M_b} \sum_{i=1}^{M_b} \mathbf{1}\{v \in R_i^b\} \mathbf{1}\{S \cap R_i^b = \varnothing\}. \tag{18}$$

We maintain an inverted index from nodes to RR sets, a covered-set indicator for every bank, and the number of uncovered RR sets containing each node. Thus Eq. (18) is computed exactly with respect to the sampled banks and can be updated after each selected seed. The bank-size rule and its cap are specified in Supplementary Section C.2.

Put $V_0 = V$ and let V_g be the node set of agent g. Let $F_b(S) = |V_b|^{-1} \sum_{u \in V_b} \Pr(u \text{ is reached from } S)$ under the one-attempt diffusion model used for RR generation. Then, for every fixed $S \subseteq V$ and $v \notin S$,

$$\mathbb{E}_{\mathcal{R}_b} \widehat{c}_b(S) = F_b(S), \qquad \mathbb{E}_{\mathcal{R}_b} \widehat{\Delta}_b(v \mid S) = F_b(S \cup \{v\}) - F_b(S). \tag{19}$$

The expectation is over the root and transmission trials of the bank. It does not imply a finite-bank uniform-accuracy guarantee under the configured sample cap. The proof is in Supplementary Section D.

For each realized bank, $\widehat{c}_b$ is monotone and submodular. For $S \subseteq T \subseteq V$ and $v \notin T$,

$$\widehat{c}_b(S \cup \{v\}) - \widehat{c}_b(S) = \widehat{\Delta}_b(v \mid S) \geq \widehat{\Delta}_b(v \mid T) \geq 0. \tag{20}$$

The incremental tracker maintains exactly these marginals after every seed insertion.

This property belongs to the sampled coverage function; the competitive selector is not a greedy maximizer of that function. The proof and tracker invariant are in Supplementary Section D.

A.2 Set-conditioned group utility learning

The teachers need a candidate representation that describes both where a node lies in the graph and which seed set has already been chosen. Let d_v be the number of forward neighbors of v, $m_{vg} = \mathbf{1}\{v \in V_g\}$, and $q_v = \sum_g m_{vg}$. We concatenate normalized log degree, a row-normalized multi-hot membership vector, and the global and group singleton RR fractions:

$$x_v = \left[\frac{\log(1 + d_v) - \mu_d}{\sigma_d + \tau_d}, \left(\frac{m_{vg}}{\sqrt{\max(q_v, 1)}} \right)_{g=1}^{G}, \widehat{c}_0(\{v\}), (\widehat{c}_g(\{v\}))_{g=1}^{G} \right] \in \mathbb{R}^{2G+2}. \tag{21}$$

Here μ_d and σ_d are the mean and standard deviation of $\log(1 + d_v)$ over V, and τ_d stabilizes degree scaling. The multi-hot normalization accounts for differing numbers of available attribute values.

Let P be the row-normalized incoming-neighbor aggregation matrix, with $P_{vu} = 1/\left|N^-(v)\right|$ for $(u, v) \in E$ when $N^-(v) \neq \varnothing$ and zero otherwise. Our graph encoder uses separate self and neighbor affine maps in two layers:

$$\begin{aligned} h_v^{(1)} &= Drop_{p_{drop}}\left(GELU\left[LN_1\left(W_{self}^{(1)} x_v + b_{self}^{(1)} + W_{neigh}^{(1)}(PX)_v + b_{neigh}^{(1)}\right)\right]\right), \\ h_v^{(2)} &= GELU\left[LN_2\left(W_{self}^{(2)} h_v^{(1)} + b_{self}^{(2)} + W_{neigh}^{(2)}\left(PH^{(1)}\right)_v + b_{neigh}^{(2)}\right)\right], \qquad z_v = h_v^{(1)} + h_v^{(2)}. \end{aligned} \tag{22}$$

Here LN is layer normalization, $GELU$ is the activation, and $W_{self}^{(\updownarrow)}, W_{neigh}^{(\updownarrow)}$ and their b terms are distinct learned weights and biases at layer $\updownarrow$. The quantity p_{drop} is the first-layer dropout probability. Each $z_v \in \mathbb{R}^H$.

We generate training states S_s and candidate collections $\mathcal{C}_s$ as detailed in Supplementary Section C.3. Each candidate v has global and group targets $\widehat{\Delta}_0\left(v \mid S_s\right)$ and $\{\widehat{\Delta}_g\left(v \mid S_s\right)\}_{g=1}^G$. For any state S, define its pooled embedding by $\bar{z}_S = |S|^{-1} \sum_{u \in S} z_u$, with $\bar{z}_\varnothing = 0$, and let

$$r(S) = \left[\, 1 - \widehat{c}_0(S),\ (1 - \widehat{c}_g(S))_{g=1}^G \,\right], \qquad \phi(v, S) = [\, z_v, \bar{z}_S, z_v \odot \bar{z}_S, r(S) \,] \in \mathbb{R}^{3H+G+1}. \tag{23}$$

The vector $r(S)$ gives the remaining RR coverage in the global and group banks; $\odot$ is elementwise multiplication.

A shared encoder feeds one teacher head for each attribute–group agent. For teacher g, the two hidden layers and its scalar output are

$$\begin{aligned} a_g(v, S) &= GELU\left(W_{g,1}\phi(v, S) + b_{g,1}\right), \\ e_g(v, S) &= GELU\left(W_{g,2} a_g(v, S) + b_{g,2}\right), \\ \widehat{u}_g(v, S) &= sigmoid\left(w_g^\intercal e_g(v, S) + b_{g,o}\right). \end{aligned} \tag{24}$$

The vectors $a_g, e_g \in \mathbb{R}^D$ are hidden activations; the W matrices, w_g, and b terms are teacher-specific learned weights and biases. The output target is the teacher's own uncovered group marginal. We jointly train the encoder and all heads with

$$\mathcal{L}_{teach} = \frac{1}{BCG} \sum_{s=1}^{B} \sum_{v \in \mathcal{C}_s} \sum_{g=1}^{G} (\widehat{u}_g\left(v, S_s\right) - \widehat{\Delta}_g\left(v \mid S_s\right))^2, \tag{25}$$

where B is the number of states and $C = |\mathcal{C}_s|$ is the number of candidates per state.

A.3 Attribute-balanced strategic competition

For a current seed state, let $c_g = \widehat{c}_g(S)$ be the coverage attained by agent g. SAGE-IM compares groups within their own attributes. For $g \in \mathcal{I}_a$, define the within-attribute coverage deficit

$$d_g = \begin{cases} \frac{\max_{h \in \mathcal{I}_a} c_h - c_g}{\max_{h \in \mathcal{I}_a} c_h - \min_{h \in \mathcal{I}_a} c_h}, & \max_{h \in \mathcal{I}_a} c_h - \min_{h \in \mathcal{I}_a} c_h > \tau, \\ \max_{h \in \mathcal{I}_a} c_h - c_g, & \max_{h \in \mathcal{I}_a} c_h - \min_{h \in \mathcal{I}_a} c_h \leq \tau. \end{cases} \tag{26}$$

The threshold τ prevents division by a numerically negligible range; the proposed method retains the original, unnormalized gaps in that case. We keep a nonnegative memory m_g of regret from earlier seed steps. To decide which agents participate at the present step, we also consider how concentrated each teacher's available utilities are. For a matrix U_{jg} on J considered actions, let $b_g^U = \max_j \max\left(U_{jg}, 0\right)$ and

$$\rho_g = \begin{cases} \max\left(0, 1 - \frac{1}{J} \sum_{j=1}^{J} \frac{\max(U_{jg}, 0)}{b_g^U}\right), & b_g^U > \tau, \\ 0, & \text{otherwise.} \end{cases} \tag{27}$$

We normalize each nonnegative signal within its attribute: $\mathcal{N}_a\left(x_g\right) = x_g / \max_{h \in \mathcal{I}_a} x_h$ if the denominator exceeds τ, and zero otherwise. The priority for active-agent selection is

$$p_g^{act} = \mathcal{N}_a\left(m_g\right) + \lambda_{proxy} \mathcal{N}_a\left(\rho_g\right) + \lambda_{def} d_g. \tag{28}$$

For every attribute, we activate its top R agents under Eq. (28) and one current worst-covered agent, taking their union as $\mathcal{A}$. When all coverages and regret memories are zero, we activate all agents.

Ties are resolved by rotating the group order. Inactive agents are not discarded: their RR trackers remain current, enabling their subsequent reentry.

We run the same competitive procedure on training states and during final selection, with a crucial difference in utility sources. On a training state, the group utility of candidate j is $U_{jg} = \widehat{u}_g(v_j, S)$, while its global utility is $U_{j0} = \widehat{\Delta}_0(v_j \mid S)$. At selection time, $U_{jg} = \widehat{\Delta}_g(v_j \mid S)$ for active agents, and the global utility remains $U_{j0} = \widehat{\Delta}_0(v_j \mid S)$. We normalize each active group's utility against its best candidate:

$$\widetilde{U}_{jg} = \begin{cases} U_{jg}/\max_i U_{ig}, & g \in \mathcal{A},\ \max_i U_{ig} > \tau, \\ 0, & \text{otherwise}, \end{cases} \qquad \widetilde{U}_{j0} = \begin{cases} U_{j0}/\max_i U_{i0}, & \max_i U_{i0} > \tau, \\ 0, & \text{otherwise}. \end{cases} \tag{29}$$

This normalization places group opportunities on comparable scales without equating their attained coverage.

Let $\mathcal{A}_a = \mathcal{A} \cap \mathcal{I}_a$. The game's cumulative regret starts at $R_{g,0} = 0$; the achieved coverages c_g and deficits d_g stay fixed throughout this game call. In round t, we assign

$$p_{g,t} = \frac{R_{g,t-1}}{t} + \lambda_{def} d_g, \qquad P_{a,t} = \max_{g \in \mathcal{A}_a} p_{g,t}. \tag{30}$$

Attributes compete according to their largest active priority, while groups compete within their attribute:

$$\begin{array}{lll} \mu_{a,t} & = \frac{\exp(\eta P_{a,t})}{\sum_{a'=1}^{A} \exp(\eta P_{a',t})}, & s_{g|a,t} = \frac{\exp(\eta p_{g,t})}{\sum_{h \in \mathcal{A}_a} \exp(\eta p_{h,t})}, \\ f_a & = min\left(\kappa, \frac{\xi}{|\mathcal{A}_a|}\right), & w_{g|a,t} = (1 - |\mathcal{A}_a| f_a)\, s_{g|a,t} + f_a, \\ \alpha_{g,t}^{base} & = \mu_{a,t} w_{g|a,t}, & g \in \mathcal{A}_a. \end{array} \tag{31}$$

The floor f_a applies to active groups' conditional weights before multiplication by attribute mass. Inactive agents receive zero weight. The attribute-level softmax allocates base mass from the maximum active priority, so an attribute receives no multiplicative mass merely from having more groups. The proposed method subtracts the maximum before exponentiation, divides softmax numerators by their sum plus a denominator guard, and subsequently renormalizes local and full agent-weight vectors.

When the preceding seed step provides weights α^{prev} with positive mass on the current active set, we restrict and renormalize them to $\bar{\alpha}^{prev}$ and form

$$\alpha_{g,t} = \left(1 - \frac{1}{t+1}\right) \alpha_{g,t}^{base} + \frac{1}{t+1} \bar{\alpha}_g^{prev}. \tag{32}$$

Otherwise $\alpha_{g,t} = \alpha_{g,t}^{base}$. After normalization, the game's chosen action is

$$j_t = arg\max_j \left[\sum_{g \in \mathcal{A}} \alpha_{g,t} \widetilde{U}_{jg} + \lambda_{\text{inf}} \widetilde{U}_{j0} \right]. \tag{33}$$

Suppose every attribute has at least one active group, $0 \le f_a \le 1/|\mathcal{A}_a|$, and probability normalization is exact. Before the optional warm start, $\sum_{g \in \mathcal{A}_a} \alpha_{g,t}^{base} = \mu_{a,t}$ and $\alpha_{g,t}^{base} \ge \mu_{a,t} f_a$. If warm-start weights are available, put $\nu_t = 1/(t+1)$. Then

$$\sum_{g \in \mathcal{A}_a} \alpha_{g,t} = (1 - \nu_t)\, \mu_{a,t} + \nu_t \sum_{g \in \mathcal{A}_a} \bar{\alpha}_g^{prev}, \qquad \sum_{g \in \mathcal{A}} \alpha_{g,t} = 1. \tag{34}$$

Thus the attribute-level softmax allocates base mass; the warm start may change that mass in a later seed step. In particular, f_a is a floor on conditional active-group weights, not a global lower bound on final agent weights. Supplementary Section D proves the identities. The active agents then accumulate regret relative to their best available normalized action:

$$R_{g,t} = R_{g,t-1} + max\left(0, \max_j \widetilde{U}_{jg} - \widetilde{U}_{j_t g}\right), \qquad g \in \mathcal{A}. \tag{35}$$

Inactive regrets do not change within the game. The loop stops when its minimum round count has been met and $\max_{g \in \mathcal{A}} R_{g,t}/t$ falls below the tolerance, or when the maximum round count is

reached. With T rounds, we define $\pi_j = (T + \tau_{count})^{-1} \sum_{t=1}^{T} \mathbf{1}\{j_t = j\}$, where τ_{count} is a small denominator guard. The final candidate score q_j is the expression in Eq. (33) evaluated using the last round's weights. Thus π is an empirical frequency of the game's chosen actions, while q retains the last-round ordering.

A.4 STUDENT DISTILLATION

The training game produces $\pi_{s,j}$, $q_{s,j}$, and $\alpha_{s,g}$. For each candidate, we use its strategically weighted teacher representation as a student target:

$$e^*_{s,j} = \sum_{g=1}^{G} \alpha_{s,g} e_g\left(v_j, S_s\right). \tag{36}$$

We initialize a separate student graph encoder from the trained teacher encoder. On its own version of $\phi\left(v_j, S_s\right)$, computed from the student encoder, the student head performs

$$\begin{array}{lll} a^{stu}_{s,j} & = GELU\left(W_{S,1}\phi\left(v_j, S_s\right) + b_{S,1}\right), & \\ e^{stu}_{s,j} & = GELU\left(W_{S,2}a^{stu}_{s,j} + b_{S,2}\right), & \\ \updownarrow_{s,j} & = w_P^{\mathsf{T}} e^{stu}_{s,j} + b_P, \qquad \widehat{u}^0_{s,j} = sigmoid\left(w_0^{\mathsf{T}} e^{stu}_{s,j} + b_0\right). & \end{array} \tag{37}$$

Here the W, w, b terms are learned student weights and biases, $e^{stu}_{s,j} \in \mathbb{R}^D$ is the intermediate representation, $\updownarrow_{s,j}$ is the policy logit, and $\widehat{u}^0_{s,j}$ is the global RR-marginal prediction. The latter is a training signal; final seeds are not chosen by this output head.

The student learns from four terms. The global regression and policy losses are

$$\begin{array}{ll} \mathcal{L}_I & = \frac{1}{BC}\sum_{s,j}(\widehat{u}^0_{s,j} - \widehat{\Delta}_0\left(v_j \mid S_s\right))^2, \\ \mathcal{L}_P & = -\frac{1}{B}\sum_s \sum_{j=1}^{C} \pi_{s,j} logsoftmax\left(\updownarrow_{s,\cdot}\right)_j. \end{array} \tag{38}$$

Let $\mathcal{P}$ contain every within-state pair (s, h, l) for which $q_{s,h} > q_{s,l}$ and $|q_{s,h} - q_{s,l}| \geq \tau_{rank}$. The ranking and representation terms are

$$\begin{array}{ll} \mathcal{L}_R & = \frac{1}{|\mathcal{P}|}\sum_{(s,h,l)\in\mathcal{P}} softplus\left(-\left(\updownarrow_{s,h} - \updownarrow_{s,l}\right)\right), \\ \mathcal{L}_E & = \omega_{mse} MSE\left(e^{stu}, e^*\right) + \omega_{\cos}\left[1 - mean_{s,j}\cos\left(e^{stu}_{s,j}, e^*_{s,j}\right)\right]. \end{array} \tag{39}$$

When $\mathcal{P}$ is empty, $\mathcal{L}_R = 0$. We combine the distillation terms as

$$\mathcal{L}_D = \lambda_P \mathcal{L}_P + \lambda_R \mathcal{L}_R + \lambda_E \mathcal{L}_E. \tag{40}$$

The student uses a conditional gradient adjustment when distillation conflicts with global RR regression. Index its encoder and head parameter blocks by p, and write $g_{I,p} = \nabla_p \mathcal{L}_I$ and $g_{D,p} = \nabla_p \mathcal{L}_D$. Let $\mathcal{O}$ contain blocks for which both automatic gradients are returned. Only these blocks contribute to the conflict statistic and its denominator:

$$d_{ID} = \sum_{p\in\mathcal{O}} \langle g_{I,p}, g_{D,p}\rangle, \qquad Q_I = \sum_{p\in\mathcal{O}} \parallel g_{I,p} \parallel_2^2. \tag{41}$$

In the following update, a missing block gradient is set to zero *after* forming $\mathcal{O}$:

$$g^{adj}_{D,p} = \begin{cases} g_{D,p} - \frac{d_{ID}}{Q_I + \tau_{proj}} g_{I,p}, & d_{ID} < 0, \\ g_{D,p}, & d_{ID} \geq 0, \end{cases} \qquad g_p = g_{I,p} + \lambda_D g^{adj}_{D,p}. \tag{42}$$

The adjustment in Eq. (42) is applied to every parameter block, including a block for which the distillation gradient was absent. The combined gradient is clipped before an Adam step.

If $d_{ID} < 0$, then $Q_I > 0$ and the adjusted gradients obey

$$\sum_{p\in\mathcal{O}} \langle g_{I,p}, g^{adj}_{D,p}\rangle = d_{ID}\frac{\tau_{proj}}{Q_I + \tau_{proj}}. \tag{43}$$

The conflict on shared blocks is attenuated but need not become nonnegative. This identity is about the computed gradients, not about a guaranteed improvement in influence after clipping and the optimizer update. The proof appears in Supplementary Section D.

A.5 SEQUENTIAL SEED SELECTION

After training, we evaluate the teacher and student models in inference mode. Starting from $S_0 = \varnothing$, we grow one ordered list up to the largest requested budget. At step k, all unselected nodes receive a student logit and a prediction from every teacher. Current RR group coverages, the persistent regret memory, and the teacher utility proxy determine the active set using Eqs. (26)–(28). We construct the candidate union from the top P_S student logits, the top P_T predictions for each active teacher, the top P_G exact group-bank marginals for each active agent, and the top P_0 global-bank marginals. Duplicate candidates are removed. Writing $Top_p(f; V\backslash S)$ for up to p available nodes ranked by score f, this union is

$$\begin{aligned} \mathcal{C}(S) = \quad & Top_{P_S}\left(\updownarrow^{stu}; V\backslash S\right) \cup Top_{P_0}\left(\widehat{\Delta}_0; V\backslash S\right) \\ & \cup \bigcup_{g\in\mathcal{A}} \left[Top_{P_T}\left(\widehat{u}_g; V\backslash S\right) \cup Top_{P_G}\left(\widehat{\Delta}_g; V\backslash S\right)\right]. \end{aligned} \tag{44}$$

The scores $\widehat{\Delta}_b$ in Eq. (44) are current uncovered marginals, and all sources are restricted to unselected nodes.

On that union, we compute current uncovered RR marginals for the global bank and active group banks and run the competition in Eqs. (29)–(35). We add the candidate with the largest action frequency π_j to S_k. If frequencies are numerically tied, the final score q_j breaks the tie. We then update the global RR tracker and *all* group trackers. The game's weights become the next step's warm start. For a game's final average regret $\bar{R}_g = R_{g,T}/T$, the cross-step memory becomes

$$m_g^{next} = \begin{cases} \mu\, m_g + (1-\mu)\bar{R}_g, & g\in\mathcal{A}, \\ \mu\, m_g, & g\notin\mathcal{A}. \end{cases} \tag{45}$$

Consequently, an unselected group can become worst-covered and reenter a later competition. Algorithm 1 summarizes the complete training and selection sequence. The returned seed sets for the requested budgets are prefixes of the same ordered list.

B PARAMETER SETTINGS

We report the values used by the supplied implementation here; symbols in Section 3 refer to these settings.

Table 2: Main SAGE-IM parameter settings.

Setting	Value
Setting	Value
Model transmission probability β	0.01
Immediate-recovery probability γ	1
RR sample-size tolerance ϵ	0.05
RR sample-size failure parameter δ	0.05
Per-bank failure allocation	$\delta/(G+1)$
Maximum global-bank size $M_{0,max}$	$15,000$
Maximum bank size for each group $M_{g,max}$	$8,000$
seed budgets $\mathcal{K}$	$\{4, 8, \ldots, 40\}$
Encoder, training states, and teachers	
Graph-encoder hidden width H	128
Number of graph aggregation layers	2
First-layer dropout p_{drop}	0.15
Teacher/student head width D	64
Number of training states B	350
Candidates per state C	$\min(12, N)$
Maximum sampled training-state size	$\min(20, N-C)$
Global singleton base-pool count	$\min(500, N)$
Per-group singleton base-pool count	$\min(250, N)$
Uniform random base-pool count	$\min(500, N)$

Setting	Value
Guided candidates per training state	at most $\lfloor C/2 \rfloor$
Teacher training epochs	180
Student training epochs	180
Teacher learning rate	10^{-3}
Student learning rate	10^{-3}
Adam weight decay	10^{-5}
Gradient-norm clip for teacher and student	5
Active agents and competitive game	
Top agents per attribute R	1
Proxy priority coefficient λ_{proxy}	0.25
Deficit coefficient λ_{def}	2.0
Across-step regret retention μ	0.85
Hierarchical softmax scale η	5.0
Conditional active-group floor κ	0.03
Floor cap numerator ξ	0.50
Global normalized-utility coefficient λ_{inf}	0.10
Minimum game rounds $T_{\min}$	60
Maximum game rounds $T_{\max}$	600
Maximum-average-regret stop tolerance τ_{regret}	0.002
Student loss and candidate proposals	
Policy-loss weight λ_P	1.0
Ranking-loss weight λ_R	0.5
Representation-loss weight λ_E	0.2
Projected-distillation gradient weight λ_D	1.0
Representation MSE coefficient ω_{mse}	0.5
Representation cosine coefficient $\omega_{\cos}$	0.5
Student proposals P_S	20
Teacher proposals per active group P_T	15
Exact RR proposals per active group P_G	15
Global RR proposals P_0	10
Numerical safeguards	
Degree-standardization constant τ_d	10^{-8}
Zero-opportunity/normalization threshold τ	10^{-12}
Projection denominator τ_{proj}	10^{-12}
Minimum ranking-score difference τ_{rank}	10^{-9}
Minimum warm-start mass	10^{-15}
Worst-covered tie: relative tolerance	10^{-10}
Worst-covered tie: absolute tolerance	10^{-12}
Action-count denominator τ_{count}	10^{-12}
Evaluation zero-total threshold τ_{zero}	10^{-15}

B.1 Independent final evaluation

The final evaluator rereads the selected seeds and simulates a one-attempt SIR process. It uses $\beta_{eval} = 0.01$, $\gamma_{eval} = 1$, a maximum of 30 transmission steps, and 300 Monte Carlo realizations. Additional definitions of reported measures appear in Supplementary Section C.6.

The primary budget-sweep figures use the default $p_{eval} = 0.01$ for $K = 4, 8, \ldots, 40$. In the supplied supplementary tables, the seed sets selected with $\beta_{model} = 0.01$ are held fixed at $K = 20, 40$ and reevaluated at $p_{eval} = 0.03$ and 0.05.

B.2 Sensitivity and structural-ablation configurations

Table 3: The appended experiment block tests the following one-at-a-time values while keeping the other settings at their defaults:

Varied parameter	Tested values
η	0.5, 5.0, 20.0
λ_{def}	0.0, 2.0, 8.0
λ_{inf}	0.0, 0.10, 1.0
μ	0.0, 0.85, 0.99
R	1, 1, 3 before deduplication/capping

For R, repeated values are removed and each value is capped by the largest group count of any attribute. Sensitivity is evaluated at the largest seed budget. The enabled structural variants are Full Method, Influence-Only, No Competitive Game, No Fairness Dynamics, No Deep Teacher-Student, No Distillation, No Influence Protection, No Attribute Balancing, and All Agents (No Dynamic Selection). We conduct a structural ablation study to examine the contribution of the main components of SAGE-IM. The Full Method corresponds to the complete SAGE-IM configuration. Influence-Only removes the fairness mechanisms, group teachers, and distilled student and selects seeds greedily using only global RR marginals. No Competitive Game retains the group-level fairness information and RR utilities but replaces the regret-based strategic competition with a fixed attribute-balanced aggregation rule. No Fairness Dynamics preserves the competitive game while disabling both the deficit term and the cross-step regret dynamics, thereby removing the adaptive fairness pressure that evolves during sequential selection.

To assess the role of learned proposal mechanisms, No Deep Teacher-Student removes both the group-teacher and student proposal models, leaving exact RR-based candidate generation together with the strategic competition. No Distillation retains the group-specific teachers but removes the distilled student and its proposal source. No Influence Protection removes the explicit global-influence contribution by setting the global normalized-utility weight to zero and disabling global-RR proposals. No Attribute Balancing removes the hierarchical allocation of strategic mass across attributes and instead lets all active group agents compete in a single flat pool, allowing attributes with more groups to exert proportionally greater influence. Finally, All Agents (No Dynamic Selection) disables the dynamic active-agent mechanism and keeps every attribute-group agent active at every seed-selection step.

C Supplementary Method and Evaluation Details

C.1 Input alignment and graph construction

The implementation reads an edge file and a node-label file. The graph is assembled from unique neighbors. Self-loops are discarded. The graph used to construct features and RR banks is held fixed during training and selection. Group labels of nodes are directly used in group membership features, RR-bank roots, group targets, and the competitive selection rule.

C.2 RR-bank count and incremental tracking

For a largest requested seed budget $K_{\max}$, let

$$L\left(N, K_{\max}\right) = log\left(\sum_{i=0}^{K_{\max}} \binom{N}{i}\right), \qquad M_{req} = \left\lceil \frac{L\left(N, K_{\max}\right) + log\left(2(G+1)/\delta\right)}{2\epsilon^2} \right\rceil. \quad (46)$$

Here N is the node count, G is the number of attribute–group agents, and ϵ, δ are the configured sample-size tolerance and failure parameter. The calculation divides δ among the global bank and the G group banks. With the code's default capped configuration, the actual sizes are

$$M_0 = min\left(M_{req}, M_{0,max}\right), \qquad M_g = min\left(M_{req}, M_{g,max}\right) \quad \text{for every } g. \quad (47)$$

The maxima and the switch controlling whether to apply the caps are given in Table 2. The calculated M_{req} therefore need not be the number of samples used. In particular, no uncapped uniform-accuracy statement is implied for a capped run.

A bank samples a root uniformly from its specified root pool with replacement, then traverses reverse neighbors. The root is initially visited. For each unvisited reverse neighbor encountered during

breadth-first traversal, a Bernoulli transmission trial determines whether it is added to the visited set and queue. The default model explores with no RR depth limit. Distinct banks are generated independently with separate seeded random-number generators.

The inverted index $\mathcal{J}_b(v) = \{i : v \in R_i^b\}$ maps node v to its containing RR-set identifiers in bank b. For each bank, let $z_i^b(S) = \mathbf{1}\{S \cap R_i^b \neq \varnothing\}$ and define

$$n_b(v; S) = \sum_{i \in \mathcal{J}_b(v)} \left(1 - z_i^b(S)\right), \qquad \widehat{\Delta}_b(v \mid S) = n_b(v; S)/M_b. \tag{48}$$

When a new seed w is selected, the tracker visits only sets $i \in \mathcal{J}_b(w)$ that are still uncovered. It marks each such set covered and decrements $n_b(v; S)$ for every node $v \in R_i^b$. Already covered sets cause no further decrement. This procedure implements Eqs. (17) and (18) for the fixed sampled banks, including for banks of groups that are currently inactive. The term exact RR marginal throughout the study means this count divided by the bank size, not an exact diffusion expectation.

C.3 Training states, candidate pool, and network targets

The training-state generator first builds a common base pool: the leading nodes by global singleton RR fraction, the leading nodes by singleton fraction in each group bank, and a uniform sample of nodes. Their union is sorted and deduplicated. The counts are specified in Table 2. It then draws B independent state sizes uniformly from $\{0, \ldots, min\,(K_{state}, N - C)\}$ and, conditional on each size, samples a seed set S_s uniformly without replacement. Here K_{state} is the configured maximum state size and C is the candidate count. This construction leaves at least C nodes outside the sampled state.

For each S_s, the generator intersects the base pool with the available nodes. It takes up to $\lfloor C/2 \rfloor$ of these nodes with the largest score

$$b(v) = \widehat{c}_0\left(\{v\}\right) + \frac{1}{G}\sum_{g=1}^{G} \widehat{c}_g\left(\{v\}\right). \tag{49}$$

It draws the remaining candidates uniformly without replacement from available nodes outside this guided subset and shuffles the resulting collection. The global and every group RR marginal are computed at that state for every sampled candidate. The group marginals supervise the group teachers, while the global marginals supervise the student's global prediction and provide the global utility in the training game. State-level remaining fractions from all banks and the pooled seed embedding form the state-conditioned input in Eq. (23). The group coverages $\widehat{c}_g\left(S_s\right)$ supply the game's attained-coverage state.

The teacher encoder and all heads are optimized together with Adam using the loss in Eq. (25). Each head consists of two affine layers followed respectively by GELU, and a separate affine sigmoid output. The second hidden activation is the representation supplied to the student target. Full-batch training, optimization settings, and model widths appear in Table 2. After training, the encoder and heads switch to inference mode before generating training-game utilities and targets.

C.4 Active agents and game implementation

In a training state, outer regret memory is absent and is therefore treated as zero. The active-agent selector receives the teacher's group-utility matrix for that state's candidates and uses the state index as its rotating tie offset. At sequential selection step k, it receives persistent outer regret memory, current coverage from all group trackers, and teacher predictions over *all* still-available nodes. The latter predictions provide the proxy in Eq. (27) before candidate proposals are assembled. Thus an inactive group still has a teacher prediction at this preliminary scoring stage.

For each attribute, its group deficit is the gap from the highest group coverage divided by the largest such gap when that gap exceeds τ. For a smaller gap, the unnormalized gap is retained. Nonnegative outer regret memory and proxy scores are separately divided by their attribute-wise maxima when those maxima are nonzero. Ranking by Eq. (28) selects the configured number of highest-priority groups per attribute, and then adds one of the current minimum-coverage groups from the same attribute. The union is sorted. For the latter choice, numerical closeness to the minimum uses the relative and absolute tolerances reported in Table 2. The priority ordering uses a rotation of the

attribute-local group indices to resolve exact ties. When both all attained coverages and all outer regret signals are numerically zero, every group is active irrespective of the proxy.

For clarity, the memory m_g in Eq. (28) is updated *between* seed choices by Eq. (45). In contrast, $R_{g,t}$ is reset to zero at the beginning of *each* competitive game and accumulates regret across that game's rounds. The achieved group coverages stay fixed during all rounds of a single game. The per-round priority uses the previous cumulative regret divided by the current round number, followed by the fixed coverage-deficit term. Each attribute receives softmax mass from its maximum active priority, and its active groups receive an internal softmax with a conditional floor. When applicable, the previous seed-step weights are restricted to the currently active groups, renormalized, and mixed according to Eq. (32). The resulting weights are renormalized. A group's final weight need not equal the within-attribute floor.

In the training game, the candidate matrix of group utilities comes from teachers, and the global utilities come from the uncovered global RR bank. In the selection game, utilities of active groups and the global utility come from their uncovered RR banks. The game chooses the first maximum of the score in Eq. (33) in each round, increases the corresponding action count, and updates active regrets. The returned policy is the normalized count vector. Its reported final score uses the weights from the last round. The stopping criterion, as well as both round bounds, is listed in Table 2.

C.5 Distillation and selection details

The training game is run independently for each sampled state. It returns the action-count policy, the last-round score vector, and the final active-agent weights. Because inactive agents have zero weights, the representation target in Eq. (36) sums to the contributions of active teachers. Every unordered candidate pair with sufficiently different final scores enters the ranking set in Eq. (39), oriented from higher to lower score.

The student has its own graph encoder initialized from the trained teacher encoder. Its two-layer GELU head produces an intermediate representation, a scalar policy logit, and a sigmoid global-marginal prediction. The policy and global outputs have separate final affine maps. The influence loss uses the global RR targets; policy, rank, and representation terms use the teacher-game outputs. The joint influence and distillation gradient is assembled across the student's encoder and head parameters as in Eq. (42). A missing gradient for a parameter is treated as zero; the vector dot product and squared influence norm use parameter blocks with both gradients present. When the dot product is negative, the proposed method projects the distillation component before summing and clipping the gradients. Adam then applies the update. This is a gradient operation during student training and does not itself establish a monotonic influence guarantee.

At each selection step the frozen teacher and student encoders compute their embeddings from the fixed graph and node features. The proposed method scores every unselected node with the student and *every* teacher before active agents are chosen. For each proposal source, it selects the highest scores from available nodes and limits its contribution by its own configured count. The student, each active teacher, each active group's exact RR tracker, and the global exact RR tracker contribute proposals; the union is sorted and deduplicated. Exact current marginals for the global bank and active groups are then computed on this union for the final game. The policy frequencies are compared with NumPy's closeness rule; if tied, the last-round final score chooses the winner. The chosen seed is removed from availability. Every group bank and the global bank are updated even when some groups did not participate in this game. The same ordered list is sliced at each requested budget; no separate training or seed-selection run is made for each budget.

C.6 Independent evaluation and reported measures

The final evaluator loads the graph and labels independently from the seed-selection procedure. For each Monte Carlo realization, the evaluator samples one independent Bernoulli live mask for every edge of its graph. It reuses that mask for every evaluated method and budget in the realization. Starting from a seed set, it explores reached nodes for at most the configured number of steps. Table 2 gives the evaluator's transmission probability, immediate recovery, horizon, and realization count. The selection RR sets have no depth limit, while this evaluation has a finite step horizon. The main budget-sweep plots use the evaluator's default $p_{eval} = .01$. For the supplementary $K \in \{20, 40\}$

tables, seed sets selected at the model probability .01 are kept fixed and evaluated at $p_{eval} = .03$ and .05.

For a budget K, let $I^{(r)}(S_K)$ be the number of reached nodes in realization r, R the number of realizations, and $n_g^{(r)}(S_K)$ the reached nodes belonging to group g. The reported normalized influence and group coverage are

$$F(K) = \frac{1}{RN}\sum_{r=1}^{R} I^{(r)}(S_K), \qquad C_g(K) = \frac{1}{R\,|V_g|}\sum_{r=1}^{R} n_g^{(r)}(S_K). \tag{50}$$

For attribute a with $m_a = |\mathcal{I}_a|$, let $\bar{C}_a = m_a^{-1}\sum_{g\in\mathcal{I}_a} C_g$ and $w_a = \min_{g\in\mathcal{I}_a} C_g$. The evaluator reports the normalized entropy of the coverage profile,

$$H_a = \begin{cases} 1, & m_a \le 1, \\ 0, & m_a > 1 \text{ and } \sum_{g\in\mathcal{I}_a} C_g \le \tau_{zero}, \\ -\dfrac{\sum_{g\in\mathcal{I}_a : p_g>0} p_g \log p_g}{\log m_a}, & \text{otherwise}, \end{cases} \qquad p_g = \frac{C_g}{\sum_{h\in\mathcal{I}_a} C_h}. \tag{51}$$

It reports the average $\frac{1}{A}\sum_a H_a$, each attribute's worst group w_a, and the global worst group $\min_a w_a$. This study computes the average of attribute worsts, as distinct from that global minimum.

That appended block also reports normalized within-attribute L1 coverage dispersion:

$$\begin{aligned} D_a &= \tfrac{1}{m_a}\textstyle\sum_{g\in\mathcal{I}_a}\left|C_g - \bar{C}_a\right|, \quad D_{a,max} = \tfrac{2\lfloor m_a/2\rfloor\lceil m_a/2\rceil}{m_a^2}, \\ ND_a &= \begin{cases} 0, & m_a \le 1, \\ D_a/D_{a,max}, & m_a > 1. \end{cases} \end{aligned} \tag{52}$$

Its macro score is $A^{-1}\sum_a ND_a$. These entropy, worst-coverage, and dispersion measures are calculated *after* seeds have been selected.

D Proofs of Properties Used in the Method

D.1 RR interpretation and finite-bank structure

Proof of Proposition A.1. Fix a bank b and a root $u \in V_b$. For the model with one independent transmission opportunity per arc and immediate recovery, sample an arc-live graph L by retaining each arc with its transmission probability β. The reverse breadth-first search used by the proposed method exposes the arcs needed to determine the nodes from which u is reachable. Skipping an arc leading to an already visited reverse-search node cannot change this reached-node set. Thus the sampled RR set is distributed as $R(u, L) = \{v : v \rightsquigarrow u \text{ in } L\}$. For any fixed seed set S,

$$\mathbf{1}\{S \cap R(u, L) \neq \varnothing\} = \mathbf{1}\{u \in Reach_L(S)\}. \tag{53}$$

The proposed method draws u uniformly from V_b and generates each RR sample with fresh transmission trials. Taking the expectation of Eq. (53) over u and L yields

$$\mathbb{E}\left[\mathbf{1}\{S \cap R(u, L) \neq \varnothing\}\right] = \frac{1}{|V_b|}\sum_{u\in V_b} \Pr\left(u \in Reach_L(S)\right) = F_b(S). \tag{54}$$

Averaging the M_b RR indicators does not change this expectation, giving the first identity. Applying the same identity to $S \cup \{v\}$ and subtracting that for S, then using the finite-bank marginal identity of Lemma A.2, proves the second. This is a statement about expectation over RR generation, irrespective of whether M_b equals the uncapped calculated sample count. □

Proof of Lemma A.2. For a fixed RR set R_i^b, let $f_i(S) = \mathbf{1}\{S \cap R_i^b \neq \varnothing\}$. If $v \notin S$, then

$$f_i\left(S \cup \{v\}\right) - f_i(S) = \mathbf{1}\{v \in R_i^b\}\mathbf{1}\{S \cap R_i^b = \varnothing\}. \tag{55}$$

Summing this equality over i and dividing by M_b gives Eq. (18). If $S \subseteq T$, the indicator that R_i^b is uncovered by T is no larger than the corresponding indicator for S. Consequently every summand in $\widehat{\Delta}_b(v \mid T)$ is no larger than the corresponding summand in $\widehat{\Delta}_b(v \mid S)$. The summands are nonnegative, which establishes monotonicity and diminishing returns.

It remains to connect this identity to the incremental implementation. Denote the set of RR indices containing v by $\mathcal{J}_b(v)$. At the empty seed set, the tracker initializes its count to $n_b(v;\varnothing) = |\mathcal{J}_b(v)|$. Suppose inductively that the tracker count at S equals $\sum_{i\in\mathcal{J}_b(v)} \mathbf{1}\{S\cap R_i^b = \varnothing\}$. When the proposed method inserts seed w, the only RR sets whose coverage changes are previously uncovered sets in $\mathcal{J}_b(w)$. For each such set, it marks the set covered and subtracts one from the count of each node in that set. Hence the resulting count is exactly the number of sets in $\mathcal{J}_b(v)$ still uncovered by $S\cup\{w\}$. By induction, dividing this count by M_b yields Eq. (18) at every selection step. The same induction applies to the global bank and each group bank, including inactive groups whose trackers are updated after every seed. □

D.2 Hierarchical weighting

Proof of Proposition A.3. Fix a game round t and attribute a, and write $n_a = |\mathcal{A}_a|$. The local softmax weights satisfy $s_{g|a,t} \geq 0$ and $\sum_{g\in\mathcal{A}_a} s_{g|a,t} = 1$. Because $0 \leq n_a f_a \leq 1$, the conditional weights in Eq. (31) satisfy

$$w_{g|a,t} = (1 - n_a f_a)\, s_{g|a,t} + f_a \geq f_a, \qquad \sum_{g\in\mathcal{A}_a} w_{g|a,t} = (1 - n_a f_a) + n_a f_a = 1. \tag{56}$$

Multiplying by the attribute mass gives $\sum_{g\in\mathcal{A}_a} \alpha_{g,t}^{base} = \mu_{a,t}$ and $\alpha_{g,t}^{base} \geq \mu_{a,t} f_a$. The attribute softmax sums to one, so the base agent weights do too. When a warm start is available, the previous weights have been set to zero outside $\mathcal{A}$ and renormalized to unit mass on $\mathcal{A}$. Summing the convex combination in Eq. (32) over the groups of attribute a gives Eq. (34); summing over all attributes gives unit total mass. The proposed method also normalizes intermediate softmax outputs and the resulting agent-weight vectors to control floating-point roundoff. □

The second identity is deliberately conditional on the availability of previous active weight mass. When there is none, the method uses α^{base} directly. Furthermore, the warm start can transfer attribute mass relative to the current round's base softmax. A final agent weight should therefore not be described as having an unconditional lower bound equal to f_a.

D.3 Shared-block gradient adjustment

Proof of Lemma A.4. Assume $d_{ID} < 0$. If $Q_I = 0$, every influence gradient on a shared block is zero and the sum defining d_{ID} is zero, a contradiction. Hence $Q_I > 0$. For every shared block $p\in\mathcal{O}$, Eq. (42) subtracts $d_{ID} g_{I,p} / (Q_I + \tau_{proj})$ from $g_{D,p}$. Taking the inner product with $g_{I,p}$ and summing over $\mathcal{O}$ yields

$$\sum_{p\in\mathcal{O}} \langle g_{I,p}, g_{D,p}^{adj} \rangle = d_{ID} - \frac{d_{ID}}{Q_I + \tau_{proj}} \sum_{p\in\mathcal{O}} \| g_{I,p} \|_2^2 = d_{ID} \frac{\tau_{proj}}{Q_I + \tau_{proj}}. \tag{57}$$

Since the numerical guard is positive, this quantity remains negative but has a smaller absolute value than d_{ID}. The proposed method applies the same subtraction to an influence-only block even if its distillation gradient was absent. Such a block did not contribute to Q_I. Accordingly the displayed identity is restricted to shared blocks and does not claim orthogonality for the full parameter vector. The subsequent combined-gradient clipping and Adam step are outside this algebraic identity. □

E Dataset Construction and Group Statistics

The numbers below describe the processed experimental graphs, not their complete raw source archives. All processed label tables have a usable categorical value for every included attribute and node. An explicit unknown or merged category is treated as an ordinary group. SAGE-IM uses those values in group membership features, rooted RR banks, teacher targets, and the game; no label-free claim applies to this algorithm.

High-School-2013. Repeated proximity contacts recorded by wearable sensors are aggregated into a static graph with 327 nodes and 5,818 unique edges (Mastrandrea, Fournet, and Barrat 2015). Student class has nine categories. The processed gender field has three categories; seven nodes are in an explicit category.

Od. The processed graph derives from the largest module-presentation cohort of the Open University Learning Analytics Dataset (Kuzilek, Hlosta, and Zdrahal 2017). Student clicks are aggregated by VLE site and transformed by $\log(1 + x)$; cosine similarity connects up to ten neighbors per student above a 0.05 similarity threshold. The largest component yields 2,302 nodes and 18,484 edges.

Ego-Facebook. The combined SNAP Facebook graph has 4,039 nodes and 88,234 friendship edges (McAuley and Leskovec 2012). Eight binary profile indicators form the attributes; their minority fractions range from 14.51% to 20.35%.

MovieLens. The MovieLens-1M data contain 6,040 users (Harper and Konstan 2015). Ratings of at least four define positive user–movie interactions; a binary matrix and cosine similarities create a user-similarity graph with at most 30 neighbors per user above threshold 0.05. The resulting graph has 152,225 edges. Gender, age, and occupation have 2, 7, and 21 encoded groups, respectively.

DBLP. A processed 5,000-node, 27,322-edge coauthorship subgraph of SNAP DBLP has one categorical community label with 15 groups (Yang and Leskovec 2012). The source archive contains overlapping venue-based communities, whereas this processed benchmark assigns one label to each retained node. The largest encoded group contains 76.08% of nodes and the smallest 0.38%.

Pokec, AH, and APS. The processed Pokec subgraph has 5,000 nodes and 48,454 edges with a binary gender-derived attribute; the encoded second category merges female and unspecified profiles. The Add Health school friendship network has 1,997 nodes and 8,523 edges and a binary white/non-white race attribute (Chowdhary et al. 2024). The APS network's largest processed component has 1,281 nodes and 3,064 edges; its two groups correspond to research topics in classical and quantum statistical mechanics (Lee et al. 2019; Chowdhary et al. 2024).

Table 4: Exact sizes of all encoded groups in the processed datasets (percent of dataset nodes).

Dataset	Attribute	Group sizes (percent)
Dataset	Attribute	Group sizes (percent)
High-School-2013	A1	44 (13.46%), 40 (12.23%), 39 (11.93%), 38 (11.62%), 36 (11.01%), 34 (10.40%), 34 (10.40%), 33 (10.09%), 29 (8.87%)
High-School-2013	A2	175 (53.52%), 145 (44.34%), 7 (2.14%)
Od	A1	1,718 (74.63%), 584 (25.37%)
Od	A2	336 (14.60%), 255 (11.08%), 201 (8.73%), 194 (8.43%), 185 (8.04%), 183 (7.95%), 162 (7.04%), 159 (6.91%), 158 (6.86%), 146 (6.34%), 121 (5.26%), 116 (5.04%), 86 (3.74%)
Od	A3	1,081 (46.96%), 638 (27.72%), 500 (21.72%), 63 (2.74%), 20 (0.87%)
Od	A4	239 (10.38%), 233 (10.12%), 226 (9.82%), 226 (9.82%), 221 (9.60%), 220 (9.56%), 217 (9.43%), 199 (8.64%), 199 (8.64%), 182 (7.91%), 140 (6.08%)
Od	A5	1,583 (68.77%), 685 (29.76%), 34 (1.48%)
Od	A6	2,116 (91.92%), 186 (8.08%)
Od	A7	890 (38.66%), 708 (30.76%), 398 (17.29%), 306 (13.29%)
Od	A8	603 (26.19%), 597 (25.93%), 575 (24.98%), 527 (22.89%)
Od	A9	2,087 (90.66%), 215 (9.34%)
Ego-Facebook	A1	3,217 (79.65%), 822 (20.35%)
Ego-Facebook	A2	3,292 (81.51%), 747 (18.49%)
Ego-Facebook	A3	3,301 (81.73%), 738 (18.27%)
Ego-Facebook	A4	3,302 (81.75%), 737 (18.25%)
Ego-Facebook	A5	3,363 (83.26%), 676 (16.74%)
Ego-Facebook	A6	3,420 (84.67%), 619 (15.33%)
Ego-Facebook	A7	3,448 (85.37%), 591 (14.63%)
Ego-Facebook	A8	3,453 (85.49%), 586 (14.51%)
MovieLens	A1	4,331 (71.71%), 1,709 (28.29%)

Dataset	Attribute	Group sizes (percent)
MovieLens	A2	2,096 (34.70%), 1,193 (19.75%), 1,103 (18.26%), 550 (9.11%), 496 (8.21%), 380 (6.29%), 222 (3.68%)
MovieLens	A3	759 (12.57%), 711 (11.77%), 679 (11.24%), 528 (8.74%), 502 (8.31%), 388 (6.42%), 302 (5.00%), 281 (4.65%), 267 (4.42%), 241 (3.99%), 236 (3.91%), 195 (3.23%), 173 (2.86%), 144 (2.38%), 142 (2.35%), 129 (2.14%), 112 (1.85%), 92 (1.52%), 72 (1.19%), 70 (1.16%), 17 (0.28%)
DBLP	A1	3,804 (76.08%), 333 (6.66%), 160 (3.20%), 136 (2.72%), 120 (2.40%), 83 (1.66%), 66 (1.32%), 63 (1.26%), 53 (1.06%), 48 (0.96%), 33 (0.66%), 31 (0.62%), 27 (0.54%), 24 (0.48%), 19 (0.38%)
Pokec	A1	2,733 (54.66%), 2,267 (45.34%)
AH	A1	1,306 (65.40%), 691 (34.60%)
APS	A1	874 (68.23%), 407 (31.77%)

F Complete Normalized Spread Comparison Tables

Supplementary fixed-seed evaluation at higher $\mathbf{p_{eval}}$. Table 5 compares SAGE-IM with the best competing score independently for each metric at $K = 40, p_{eval} = 0.05$. SAGE-IM exceeds the best baseline in normalized spread on seven of the eight graphs; AH is the exception (0.050 versus 0.055). MovieLens has the largest spread difference in this column (0.890 versus 0.843). Macro entropy is greater on five graphs, equal at the displayed precision on MovieLens, AH, and APS, and lower than the best baseline on none in this column. The full 8×4 configurations and all methods appear in Appendix F and G.

Table 5: Results at $K = 40$, evaluation probability $p = 0.05$ from the comparison tables. Each cell is SAGE-IM / strongest competing value across the nine baselines; larger is better.

Dataset	Spread: SAGE	Spread: best	Entropy: SAGE	Entropy: best
High-School-2013	0.790	0.768	0.999	0.998
Od	0.140	0.127	0.998	0.996
Ego-Facebook	0.568	0.561	0.983	0.979
MovieLens	0.890	0.843	0.999	0.999
DBLP	0.145	0.143	0.987	0.971
Pokec	0.340	0.338	0.999	0.998
AH	0.050	0.055	0.999	0.999
APS	0.065	0.061	0.999	0.999

Across all 32 displayed dataset–budget–probability configurations, the spread advantage over the strongest baseline is positive in 25, tied in one, and negative in six; the analogous entropy counts are 22, six, and four. Figure 3 in the appendix visualizes these signed differences. The largest displayed spread gain is $+0.056$ (MovieLens, $K = 40, p_{eval} = .03$), while the largest loss is -0.005 (AH, $K = 40, p_{eval} = .05$). The largest entropy gain is $+0.052$ (DBLP, $K = 20, p_{eval} = .03$), while the largest loss is -0.002 (AH, $K = 20, p_{eval} = .03$).

Among the 160 displayed entropy entries at $p_{eval} = .05$ (ten methods, eight graphs, and two budgets), 97 are at least .990; this holds for every displayed method and budget on High-School-2013, MovieLens, and Pokec. None of the Ego-Facebook or DBLP entries reaches .990. At $p_{eval} = .03$, 70 of 160 entries reach that threshold. Hence the elevated probabilities are useful as a fixed-seed spread check, but their entropy values can have less power to distinguish fairness on particular graphs. At $K = 20, p_{eval} = .03$, SAGE-IM trails the best spread baseline on High-School-2013 (.388 versus .391), DBLP (.036 versus .040), AH (.016 versus .020), and APS (.026 versus .030); DBLP nevertheless has higher entropy (.978 versus .926). At $K = 40, p_{eval} = .05$, MovieLens spread is .890 versus .843 while both methods' entropy is .999 at the displayed precision.

Columns in each dataset are $(K, p_{eval}) = (20, .03), (20, .05), (40, .03), (40, .05)$. Each method's seed sets selected under $\beta_{model} = .01$.

Table 6: Normalized spread, High-School-2013 and Od. Higher is better.

	High-School-2013				Od			
Method	$20, .03$	$20, .05$	$40, .03$	$40, .05$	$20, .03$	$20, .05$	$40, .03$	$40, .05$
Log	0.391	0.738	0.482	0.768	0.025	0.085	0.044	0.124
Sqrt	0.391	0.738	0.483	0.768	0.025	0.083	0.045	0.123
MOEIM	0.390	0.713	0.470	0.729	0.031	0.091	0.054	0.120
S3D-OT	0.381	0.732	0.483	0.764	0.016	0.050	0.034	0.087
Tsang	0.389	0.720	0.470	0.747	0.018	0.058	0.035	0.127
DeepIM	0.383	0.722	0.477	0.743	0.018	0.054	0.035	0.093
FAC	0.388	0.724	0.471	0.744	0.019	0.049	0.038	0.088
FPS	0.388	0.723	0.471	0.744	0.020	0.049	0.038	0.088
CrossWalk	0.353	0.736	0.476	0.765	0.018	0.059	0.036	0.098
SAGE-IM	0.388	0.749	0.480	0.790	0.032	0.095	0.055	0.140

Table 7: Normalized spread, Ego-Facebook and MovieLens. Higher is better.

	Ego-Facebook				MovieLens			
Method	$20, .03$	$20, .05$	$40, .03$	$40, .05$	$20, .03$	$20, .05$	$40, .03$	$40, .05$
Log	0.376	0.553	0.381	0.561	0.570	0.841	0.583	0.841
Sqrt	0.376	0.553	0.382	0.560	0.570	0.841	0.583	0.841
MOEIM	0.363	0.534	0.363	0.547	0.576	0.842	0.576	0.842
S3D-OT	0.355	0.527	0.374	0.554	0.575	0.841	0.582	0.841
Tsang	0.368	0.534	0.364	0.550	0.575	0.842	0.581	0.842
DeepIM	0.351	0.521	0.361	0.537	0.576	0.841	0.581	0.843
FAC	0.358	0.498	0.358	0.534	0.573	0.842	0.579	0.843
FPS	0.357	0.524	0.358	0.534	0.573	0.842	0.579	0.843
CrossWalk	0.351	0.523	0.368	0.537	0.576	0.841	0.582	0.840
SAGE-IM	0.382	0.560	0.393	0.568	0.610	0.863	0.639	0.890

Table 8: Normalized spread, DBLP and Pokec. Higher is better.

	DBLP				Pokec			
Method	$20, .03$	$20, .05$	$40, .03$	$40, .05$	$20, .03$	$20, .05$	$40, .03$	$40, .05$
Log	0.032	0.128	0.046	0.137	0.109	0.331	0.123	0.338
Sqrt	0.031	0.127	0.041	0.136	0.109	0.331	0.123	0.338
MOEIM	0.040	0.126	0.051	0.143	0.112	0.316	0.121	0.321
S3D-OT	0.013	0.117	0.016	0.132	0.100	0.320	0.110	0.329
Tsang	0.013	0.113	0.029	0.134	0.100	0.315	0.116	0.323
DeepIM	0.008	0.101	0.021	0.130	0.093	0.317	0.112	0.329
FAC	0.031	0.124	0.048	0.132	0.105	0.312	0.119	0.321
FPS	0.030	0.130	0.044	0.138	0.106	0.316	0.119	0.322
CrossWalk	0.011	0.067	0.018	0.122	0.030	0.263	0.060	0.318
SAGE-IM	0.036	0.131	0.053	0.145	0.113	0.332	0.126	0.340

Table 9: Normalized spread, AH and APS. Higher is better.

	AH				APS			
Method	$20, .03$	$20, .05$	$40, .03$	$40, .05$	$20, .03$	$20, .05$	$40, .03$	$40, .05$
Log	0.014	0.020	0.026	0.036	0.021	0.033	0.040	0.055
Sqrt	0.014	0.019	0.027	0.036	0.021	0.032	0.040	0.054

MOEIM	0.020	0.029	0.030	0.055	0.030	0.037	0.047	0.060
S3D-OT	0.014	0.020	0.026	0.040	0.019	0.019	0.036	0.046
Tsang	0.013	0.019	0.028	0.049	0.018	0.023	0.045	0.061
DeepIM	0.014	0.019	0.027	0.037	0.023	0.032	0.044	0.057
FAC	0.015	0.023	0.030	0.043	0.018	0.020	0.036	0.038
FPS	0.015	0.023	0.030	0.043	0.018	0.020	0.036	0.040
CrossWalk	0.011	0.014	0.024	0.031	0.017	0.022	0.036	0.044
SAGE-IM	0.016	0.029	0.031	0.050	0.026	0.041	0.048	0.065

G Complete Coverage-Entropy Comparison Tables

Columns in each dataset are $(K, p_{eval}) = (20, .03), (20, .05), (40, .03), (40, .05)$.

Table 10: Macro-average normalized coverage entropy, High-School-2013 and Od. Higher is better.

	High-School-2013				Od			
Method	$20, .03$	$20, .05$	$40, .03$	$40, .05$	$20, .03$	$20, .05$	$40, .03$	$40, .05$
Log	0.991	0.993	0.990	0.990	0.983	0.993	0.985	0.994
Sqrt	0.991	0.995	0.991	0.990	0.982	0.991	0.982	0.994
MOEIM	0.993	0.998	0.994	0.998	0.978	0.991	0.990	0.991
S3D-OT	0.994	0.998	0.991	0.991	0.954	0.993	0.972	0.996
Tsang	0.990	0.998	0.991	0.992	0.956	0.993	0.971	0.993
DeepIM	0.994	0.998	0.990	0.998	0.958	0.993	0.984	0.994
FAC	0.995	0.998	0.995	0.998	0.961	0.982	0.981	0.991
FPS	0.995	0.998	0.995	0.998	0.964	0.982	0.980	0.990
CrossWalk	0.995	0.998	0.995	0.998	0.983	0.987	0.975	0.991
SAGE-IM	0.998	0.999	0.997	0.999	0.991	0.997	0.996	0.998

Table 11: Macro-average normalized coverage entropy, Ego-Facebook and MovieLens. Higher is better.

	Ego-Facebook				MovieLens			
Method	$20, .03$	$20, .05$	$40, .03$	$40, .05$	$20, .03$	$20, .05$	$40, .03$	$40, .05$
Log	0.951	0.978	0.953	0.979	0.999	0.999	0.999	0.999
Sqrt	0.951	0.978	0.951	0.979	0.999	0.999	0.999	0.999
MOEIM	0.946	0.974	0.946	0.977	0.998	0.999	0.998	0.999
S3D-OT	0.940	0.972	0.950	0.979	0.999	0.999	0.999	0.999
Tsang	0.949	0.974	0.946	0.978	0.999	0.999	0.999	0.999
DeepIM	0.936	0.969	0.942	0.974	0.998	0.999	0.998	0.999
FAC	0.942	0.961	0.943	0.974	0.998	0.999	0.999	0.999
FPS	0.942	0.970	0.942	0.974	0.998	0.999	0.999	0.999
CrossWalk	0.936	0.970	0.945	0.973	0.999	0.999	0.999	0.999
SAGE-IM	0.957	0.981	0.961	0.983	0.999	0.998	0.999	0.999

Table 12: Macro-average normalized coverage entropy, DBLP and Pokec. Higher is better.

	DBLP				Pokec			
Method	$20, .03$	$20, .05$	$40, .03$	$40, .05$	$20, .03$	$20, .05$	$40, .03$	$40, .05$
Log	0.890	0.963	0.950	0.971	0.996	0.998	0.996	0.998
Sqrt	0.872	0.958	0.892	0.958	0.996	0.998	0.996	0.998
MOEIM	0.904	0.932	0.947	0.959	0.997	0.998	0.997	0.998
S3D-OT	0.913	0.928	0.910	0.941	0.997	0.998	0.995	0.998
Tsang	0.694	0.935	0.839	0.953	0.997	0.998	0.996	0.998
DeepIM	0.790	0.927	0.877	0.942	0.998	0.998	0.998	0.998

FAC	0.926	0.955	0.921	0.954	0.997	0.998	0.996	0.998
FPS	0.889	0.953	0.895	0.954	0.996	0.998	0.996	0.998
CrossWalk	0.880	0.940	0.868	0.961	0.994	0.998	0.996	0.998
SAGE-IM	0.978	0.986	0.991	0.987	0.997	0.998	0.997	0.999

Table 13: Macro-average normalized coverage entropy, AH and APS. Higher is better.

	AH				APS			
Method	20, .03	20, .05	40, .03	40, .05	20, .03	20, .05	40, .03	40, .05
Log	0.964	0.982	0.971	0.984	0.926	0.980	0.918	0.963
Sqrt	0.939	0.970	0.951	0.974	0.784	0.928	0.854	0.907
MOEIM	0.990	0.995	0.989	0.997	0.782	0.757	0.832	0.788
S3D-OT	0.989	0.992	0.979	0.999	0.979	0.970	0.981	0.992
Tsang	0.995	0.997	0.927	0.999	0.889	0.919	0.988	0.938
DeepIM	0.981	0.990	0.940	0.999	0.950	0.918	0.954	0.931
FAC	0.979	0.994	0.987	0.999	0.937	0.849	0.983	0.998
FPS	0.936	0.993	0.987	0.999	0.937	0.849	0.988	0.999
CrossWalk	0.917	0.988	0.977	0.973	0.982	0.976	0.975	0.999
SAGE-IM	0.993	0.999	0.995	0.999	0.999	0.999	0.997	0.999

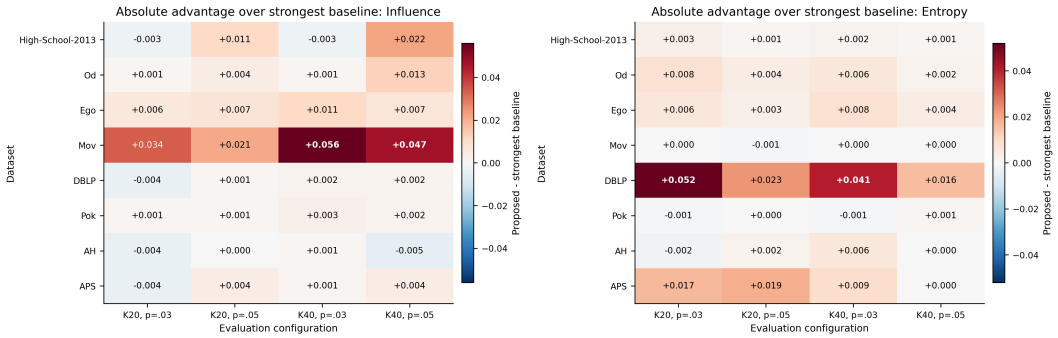

Figure 3: Supplementary fixed-seed reevaluations at $p_{eval} \in \{.03, .05\}$: SAGE-IM minus the strongest baseline for each configuration in normalized spread (left) and macro-average coverage entropy (right). Positive cells favor SAGE-IM.

H Budget–Trade-off Figures

The following eight primary-protocol plots use $p_{eval} = .01$ and display macro-average normalized entropy, macro-average attribute-wise worst-group coverage, and macro-average normalized L1 dispersion against normalized influence spread. Markers distinguish budgets $K = 4, 8, \ldots, 40$, with trajectories connecting increasing K. The tables in Appendix F and G are separate fixed-seed reevaluations at $p_{eval} \in \{.03, .05\}$.

Reading the trajectories. Each colored path follows one method as the seed budget grows; an open marker denotes its first displayed budget and the terminal marker its last. The three panels share the same horizontal spread measure but use different vertical measures. In particular, an upward change in worst-group coverage need not imply lower dispersion, and entropy can approach one even when absolute coverage is small.

High-School-2013 (Figure 4). SAGE-IM's entropy rises from its comparatively low first point and approaches one by the final budget. Its worst-group coverage increases throughout the sweep and has the highest plotted endpoint. Its dispersion first increases over a small range of budgets and then falls substantially; the terminal value is below the terminal values of the other shown methods. Early

budgets therefore do not have the same ranking as the final budget: the initial SAGE-IM entropy is lower than several alternatives, while its final balance and minimum coverage are higher. Its spread endpoint is close to the rightmost endpoints of the plot.

Od (Figure 5). The SAGE-IM curve rises rapidly in entropy from its first seed budget, while its worst-group coverage grows steadily and ends above the other displayed endpoints. Its L1 dispersion is small over the sweep and remains below the alternatives at the largest budget. The endpoint spread is high but not the rightmost: MOEIM extends farther horizontally.

Ego-Facebook (Figure 6). The SAGE-IM trajectory ends at the highest displayed entropy and worst-group coverage while extending to the largest spread among the plotted endpoints. CrossWalk instead has a lower dispersion endpoint than SAGE-IM, but its entropy is much lower and its spread endpoint is farther left. Consequently the three fairness measures do not induce a single ranking: high worst-group coverage and entropy do not make SAGE-IM the dispersion minimum on this graph. This is especially pertinent for the eight binary attributes, whose macro averages can obscure per-attribute variation (Figure 12).

MovieLens (Figure 7). Most entropy trajectories cluster near the upper bound, so their visible separation is narrow even when the methods have different absolute group coverage. SAGE-IM ends near the top of entropy and has the highest worst-group endpoint in the supplied plot. Its dispersion stays below several competing paths through the later budgets, while some alternatives have comparable small values at intermediate budgets. The SAGE-IM spread endpoint is close to the furthest-right displayed trajectory. The near-ceiling entropy is best interpreted together with worst-group coverage and dispersion.

DBLP (Figure 8). This panel makes the different objectives particularly visible. SAGE-IM finishes with substantially higher entropy and worst-group coverage than the displayed competitors, whereas MOEIM reaches a greater spread endpoint. SAGE-IM's dispersion also decreases after an early increase and is low at the final budget, but it is not the minimum endpoint among all methods. The plotted difference in minimum coverage is more pronounced than the differences between some entropy trajectories; the supplementary table likewise shows that entropy and spread can favor different methods, although its $p_{eval} \in \{.03, .05\}$ values cannot be compared directly with this $p_{eval} = .01$ trajectory.

Pokec (Figure 9). Entropy of most methods, including SAGE-IM, is close to one for much of the trajectory. SAGE-IM has low final dispersion, yet other methods can have even smaller plotted values. MOEIM extends farther horizontally and has a higher worst-group endpoint than SAGE-IM in this figure. Thus near-perfect entropy does not establish a best worst-group outcome.

AH (Figure 10). At later budgets SAGE-IM's entropy is near one and its worst-group coverage is close to the largest endpoints, while its L1 dispersion stays low. FPS has a slightly higher plotted worst-group endpoint and MOEIM extends farther on the spread axis. Several methods have comparably low or lower final dispersion.

APS (Figure 11). SAGE-IM maintains entropy near one over the displayed budgets, while other methods such as S3D-OT attain similarly high entropy. Its worst-group coverage rises with budget, but S3D-OT's final value is higher. Its final L1 dispersion is lower than several competitors' values, although S3D-OT and some others are lower still. The final spread endpoint is among the farther-right points without being the unique furthest-right.

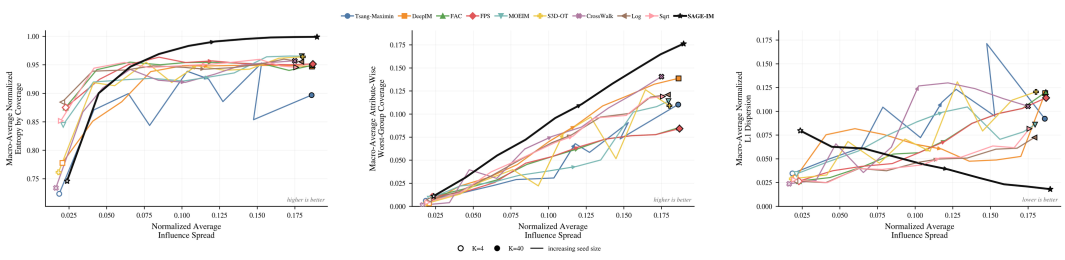

Figure 4: Primary $p_{eval} = .01$ budget–trade-off plots for High-School-2013. Left: coverage entropy; middle: worst-group coverage; right: normalized L1 dispersion.

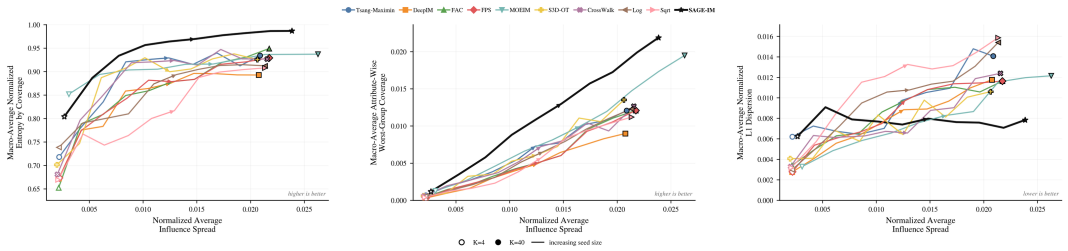

Figure 5: Budget–trade-off plots for Od: entropy, worst-group coverage, dispersion (left to right).

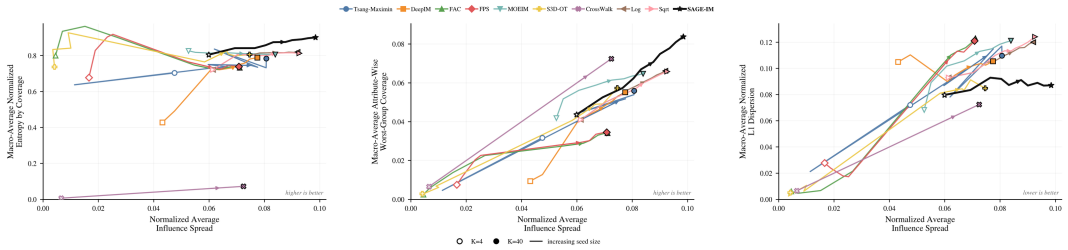

Figure 6: Budget–trade-off plots for Ego-Facebook: entropy, worst-group coverage, dispersion.

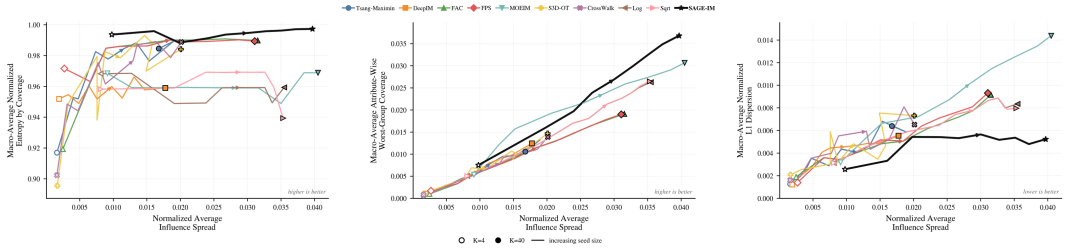

Figure 7: Budget–trade-off plots for MovieLens: entropy, worst-group coverage, dispersion.

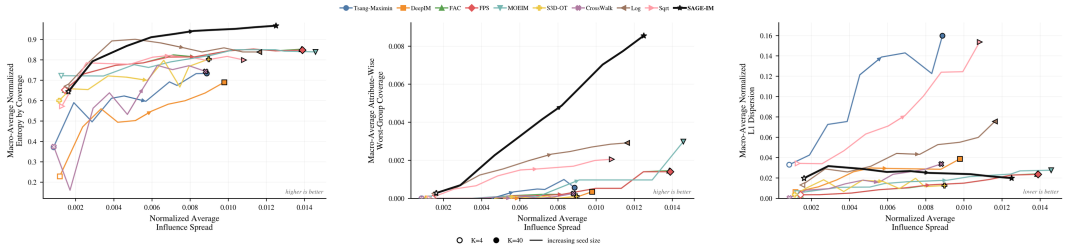

Figure 8: Budget–trade-off plots for DBLP: entropy, worst-group coverage, dispersion.

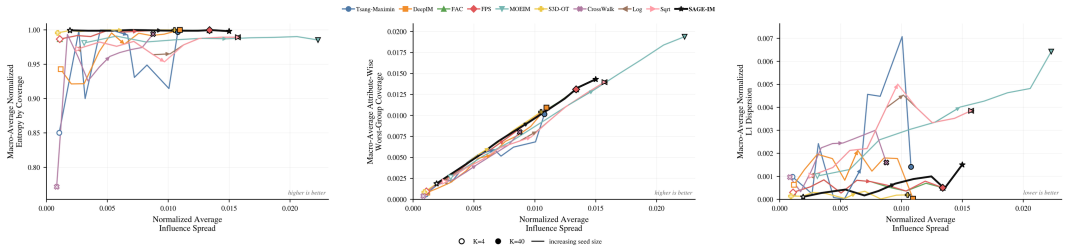

Figure 9: Budget–trade-off plots for Pokec: entropy, worst-group coverage, dispersion.

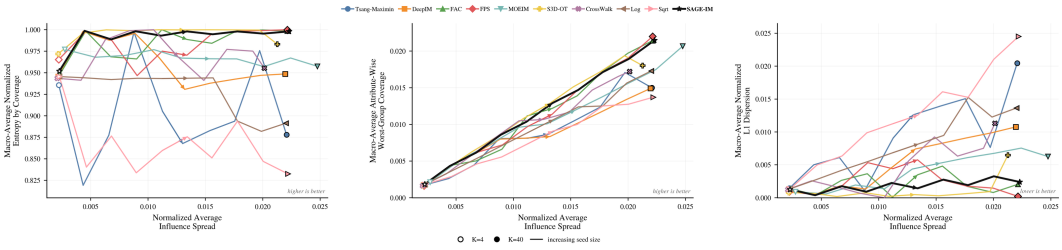

Figure 10: Budget–trade-off plots for AH: entropy, worst-group coverage, dispersion.

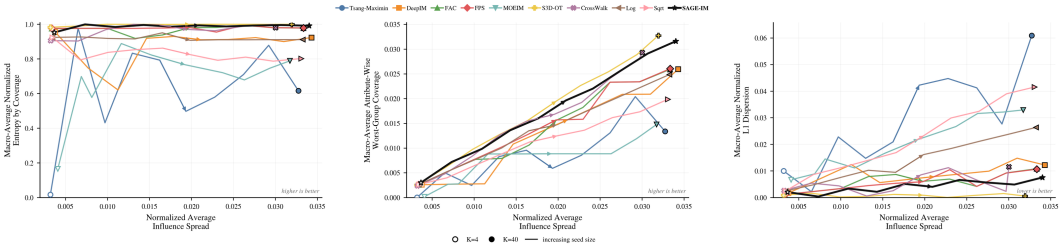

Figure 11: Budget–trade-off plots for APS: entropy, worst-group coverage, dispersion.

I Active-Agent, Sensitivity, and Ablation Figures

Selection diagnostics and component changes. The four-dataset heatmap in Appendix I resolves macro entropy by attribute: Od's third attribute and several Ego-Facebook attributes remain visibly less balanced at small budgets despite increasing entropy with K. The active-agent fraction falls as the seed budget increases on all four plotted graphs; the plot records active fractions, not end-to-end runtime. In the one-at-a-time sensitivity plots, suppressing the deficit weight substantially reduces entropy and worst-group coverage on High-School-2013 and Od. Raising the global-influence weight can increase spread while reducing the worst-group score, notably on MovieLens. The structural ablations are also mixed: keeping all agents active gives slightly higher spread than the full selection on these four graphs, while the influence-only variant has lower worst-group coverage. Several variants change multiple switches at once, so their curves are comparisons of selection configurations, not isolated estimates of one component's causal effect. The reruns reuse learned models and RR banks; full figure-by-figure interpretations appear in Appendix I.

The following four-dataset panels contain High-School-2013, Od, Ego-Facebook, and MovieLens. Their main evaluation uses $p_{eval} = .01$. The proposed method evaluates one-at-a-time selection-time changes to η, λ_{def}, λ_{inf}, μ, and top-R activity.

Attribute-level balance (Figure 12). The heatmap displays SAGE-IM's normalized entropy separately for each attribute, with one column per seed budget. The High-School-2013 and MovieLens rows become nearly uniform in color at larger budgets, reflecting entropy close to one. Od contains a more heterogeneous start: its third attribute has especially low entropy at the first displayed budget, and its fifth attribute also starts lower than many rows; both improve over the sweep. On Ego-Facebook, the last two attribute rows remain visibly lower than the others even at the largest budget. Macro averaging can therefore hide the attributes with weaker balance; a near-one macro score does not establish a uniformly near-one score for every attribute.

Active agents (Figure 13). The blue fraction is the mean share of group agents selected for competition over the prefix, and the red fraction is its complement. On all four graphs, the mean active fraction decreases with budget. Its decline is steepest during the early part of each sequence, then becomes gradual. The retained share is visibly dataset-dependent: the Ego-Facebook curve stays above one half at the final budget, whereas MovieLens falls below one fifth. The figure quantifies participation of group agents in the selection game.

One-at-a-time sensitivity: spread (Figure 14). Each colored segment varies one parameter at $K = 40$ while holding the other selection settings at their full-method values; repeated nominal top-R settings are deduplicated and the tested R is capped by the number of groups in the attribute. The vertical scales differ sharply by graph and are in some cases tight. Memory and game-softmax changes produce comparatively small spread changes in the displayed range. Setting deficit weight to zero lowers spread visibly on High-School-2013 and Od, but the MovieLens zero-deficit point has higher spread than the default-deficit point. Increasing global-influence weight gives MovieLens a higher spread while its outcome on Od is nonmonotone.

One-at-a-time sensitivity: entropy (Figure 15). For High-School-2013 and Od, removing the deficit contribution produces a conspicuous entropy drop relative to the nonzero deficit settings. Ego-Facebook also drops, though by less, and MovieLens remains close to its entropy ceiling across most of the sweep. In Od and Ego-Facebook, raising the global-influence weight to the largest displayed setting reduces entropy. The effect is far less visible on MovieLens because the measure is already close to one.

One-at-a-time sensitivity: minimum coverage (Figure 16). Zero deficit weight also lowers the worst-group score on High-School-2013, Od, and Ego-Facebook. On MovieLens, increasing the number of active groups per attribute gives a higher plotted worst-group score than the smallest R. Increasing the global-influence weight to the largest tested setting sharply lowers worst-group coverage on Od and MovieLens, even where spread can rise.

Structural variants: spread (Figure 17). The spread curves increase with budget on each graph. Their separation is small relative to the full vertical range on High-School-2013 and Ego-Facebook. The all-agents variant finishes slightly above the full method on all four displayed graphs, most visibly on MovieLens. The influence-only curve is below the full method on Od and Ego-Facebook, while the no-fairness-dynamics variant slightly exceeds it on Od and Ego-Facebook at the final budget.

Structural variants: entropy (Figure 18). On High-School-2013, Od, and Ego-Facebook, the influence-only variant has lower entropy than the full method at the final budget. The no-competitive-game and no-fairness-dynamics trajectories can also fall below the full method. MovieLens remains close to the entropy ceiling for several configurations, making their relative differences small on an absolute scale.

Structural variants: worst-group coverage (Figure 19). The influence-only configuration has conspicuously lower worst-group coverage on Od, Ego-Facebook, and MovieLens than the full method at the final budget. Removing fairness dynamics or the competitive game can also lower the minimum coverage on some graphs. Conversely, the all-agents setting exceeds the full method on High-School-2013, and no attribute balancing exceeds it on Od. These exceptions are relevant to interpreting the mechanism: the structural variants change the action sequence and sometimes several switches simultaneously, while the trained encoders and RR banks are reused.

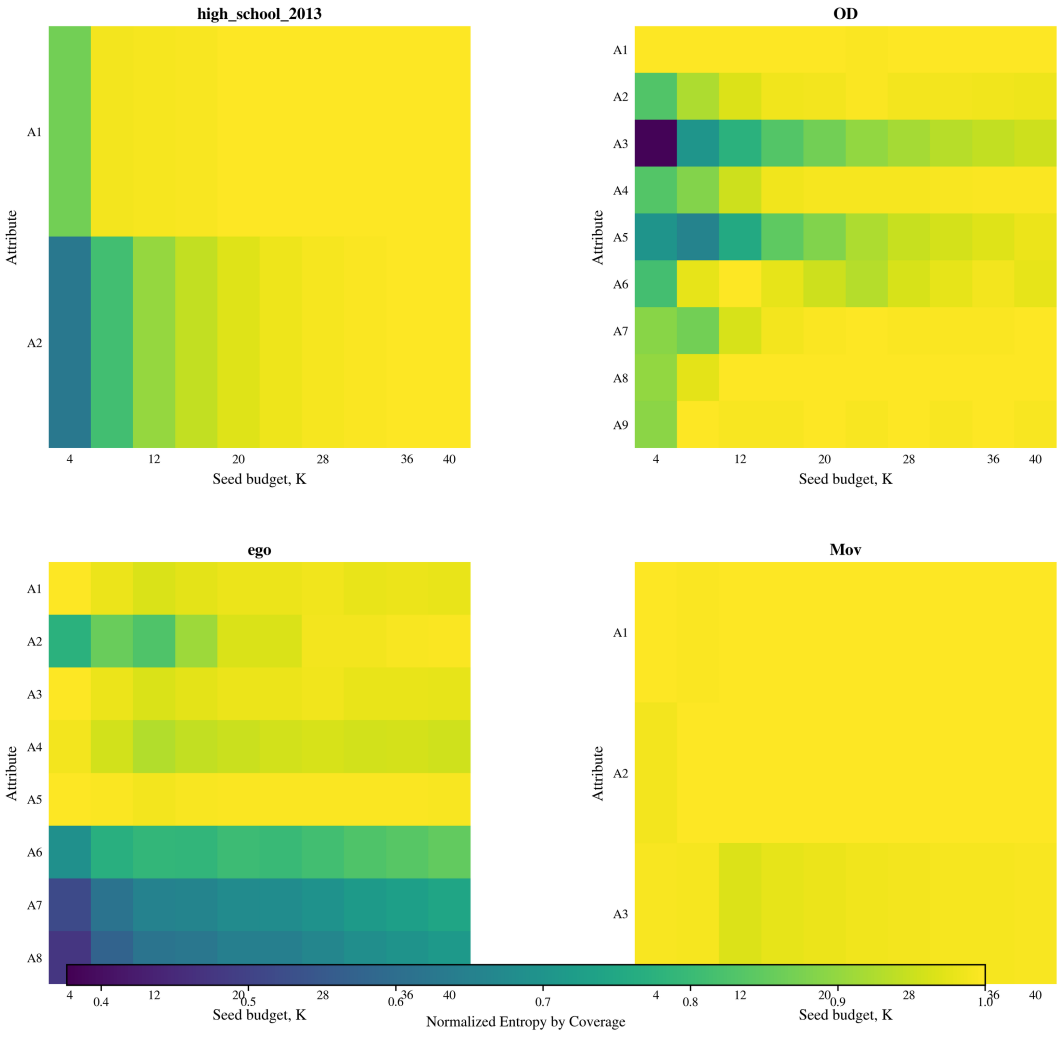

Figure 12: Attribute-wise normalized coverage entropy on four benchmark graphs; rows are attributes and columns are seed budgets.

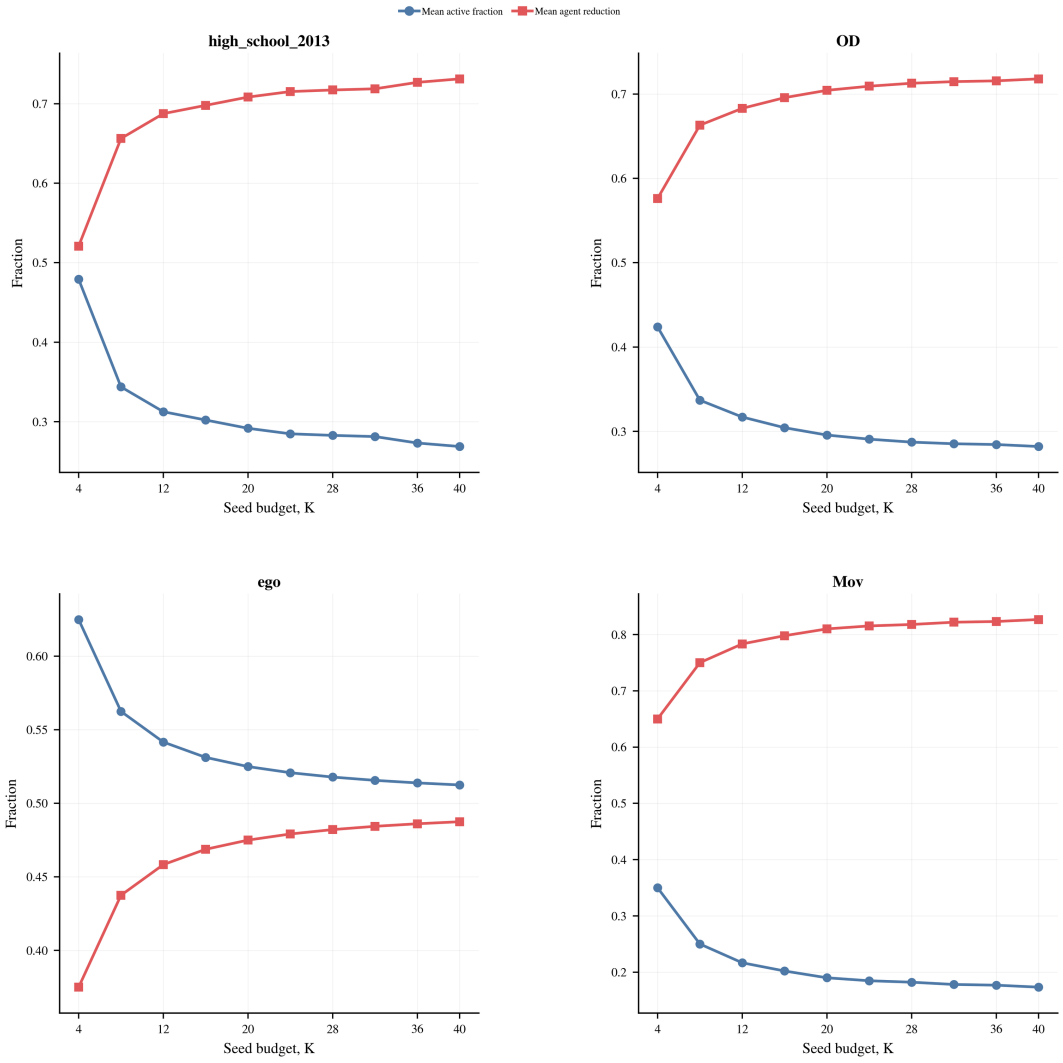

Figure 13: Mean active-agent fraction and its complement (agent reduction) on four graphs.

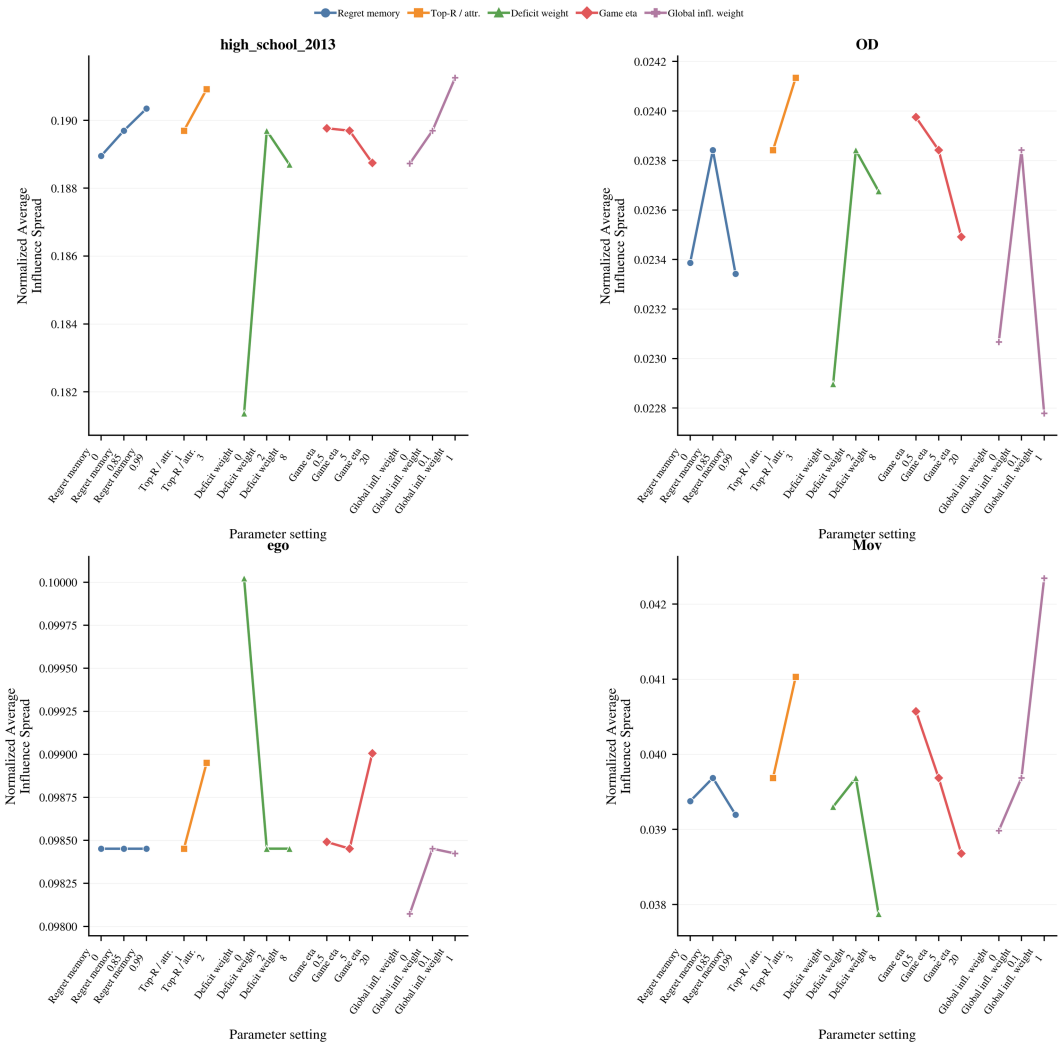

Figure 14: One-at-a-time selection sensitivity at $K = 40$: normalized spread.

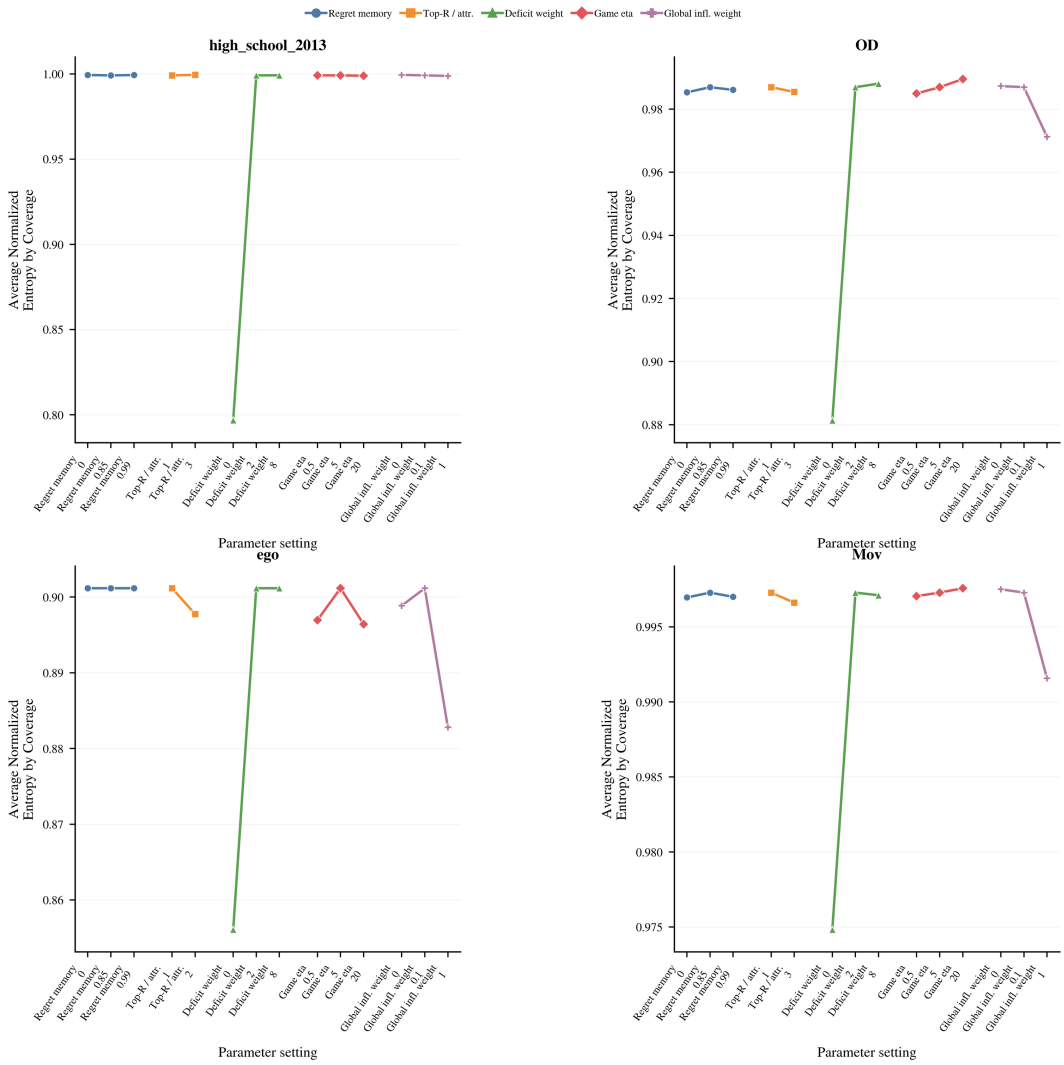

Figure 15: One-at-a-time selection sensitivity at $K = 40$: macro coverage entropy.

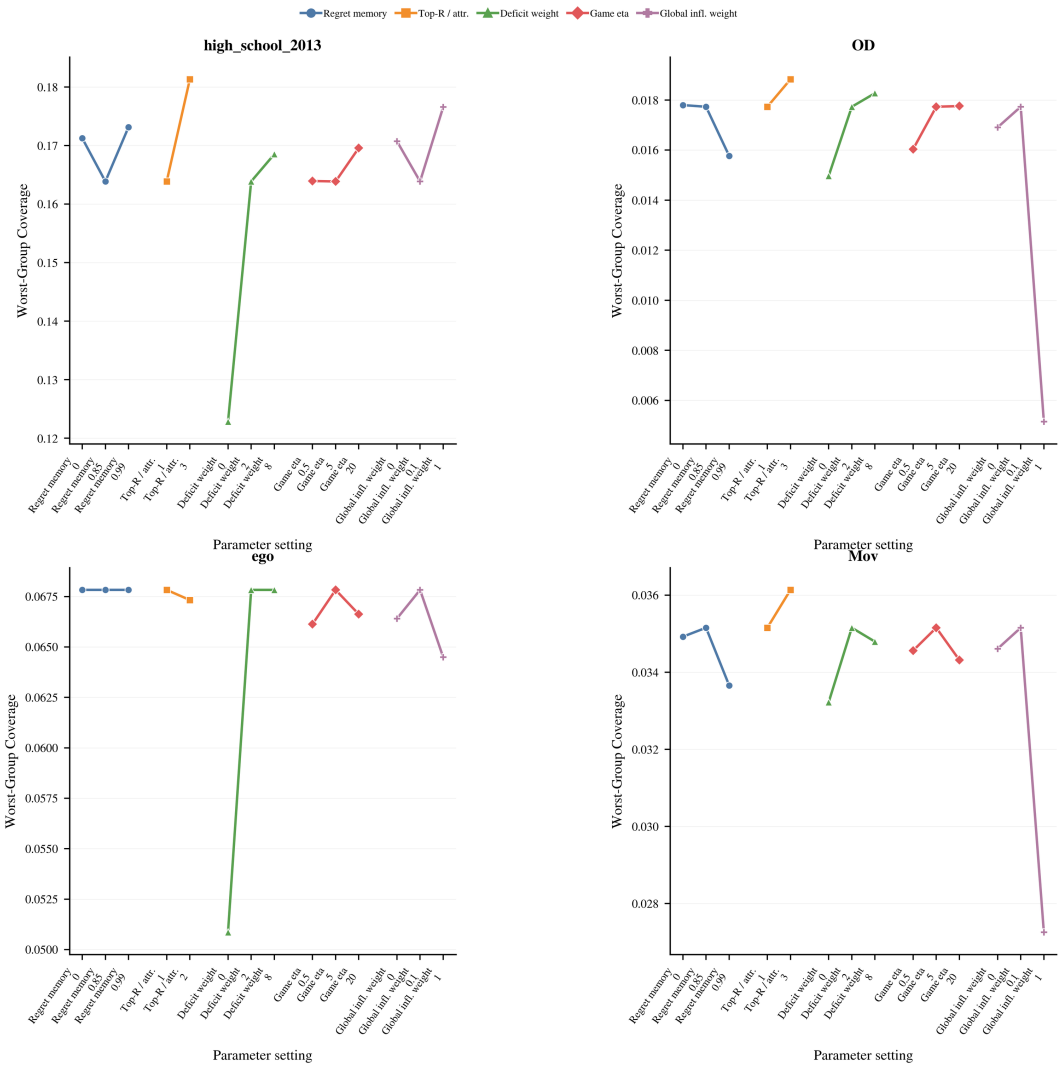

Figure 16: One-at-a-time selection sensitivity at $K = 40$: macro worst-group coverage.

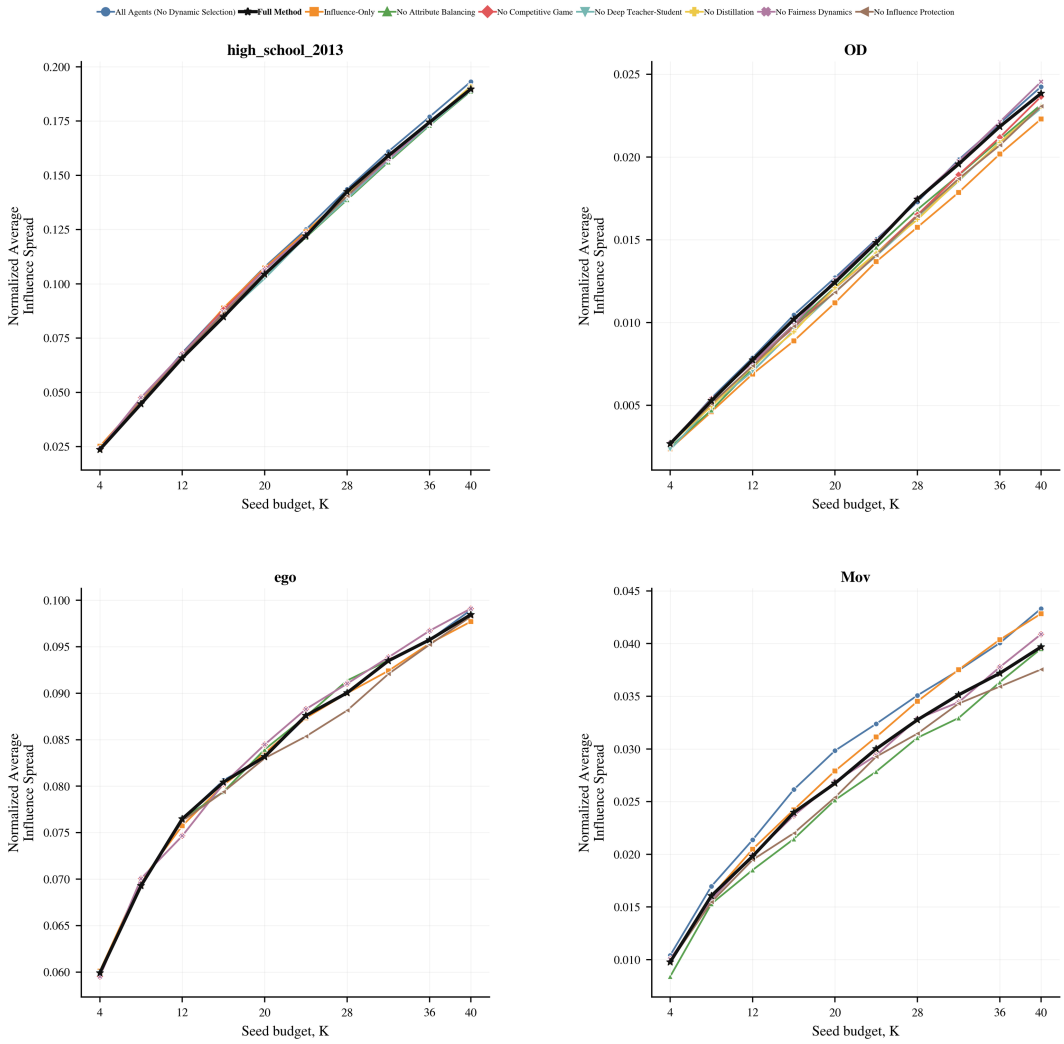

Figure 17: Selection-stage structural variants: normalized spread.

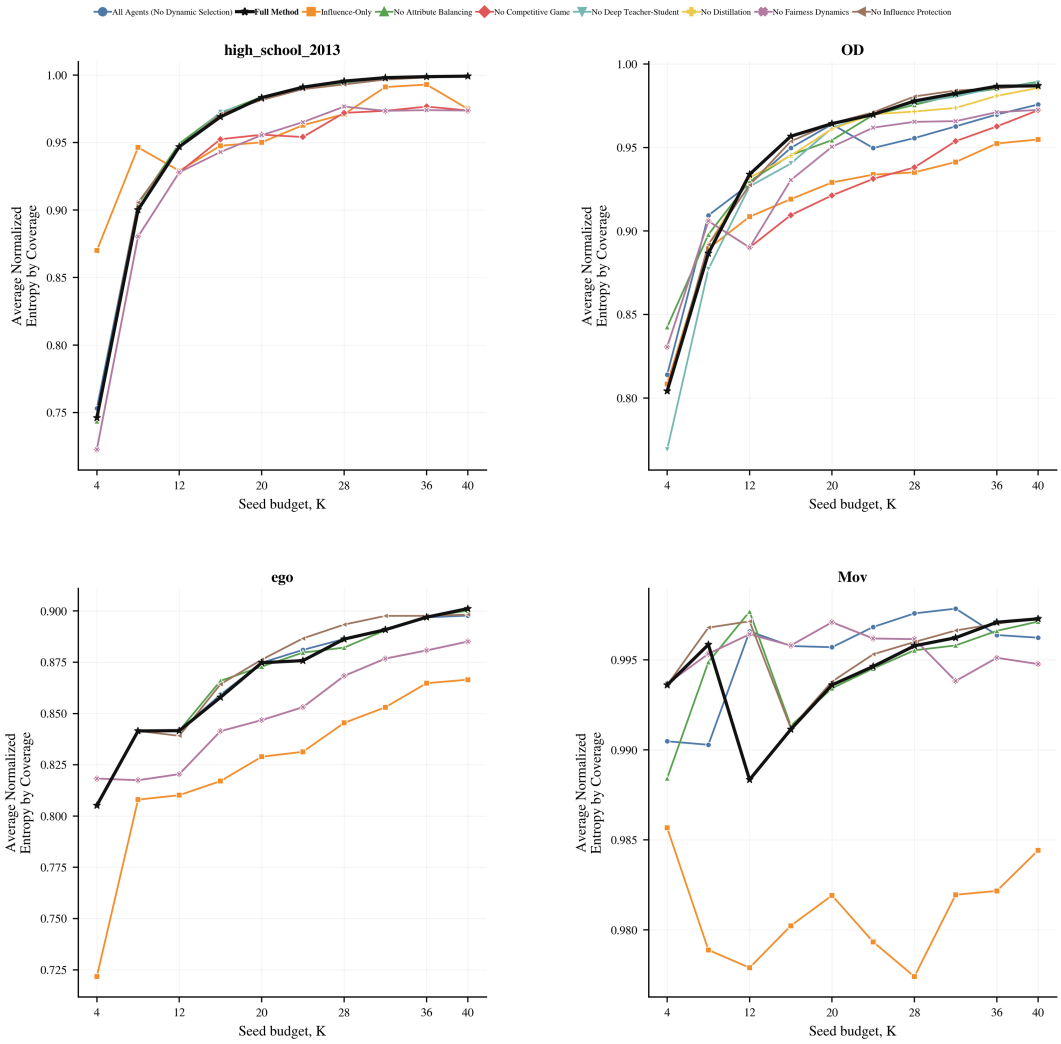

Figure 18: Selection-stage structural variants: macro coverage entropy.

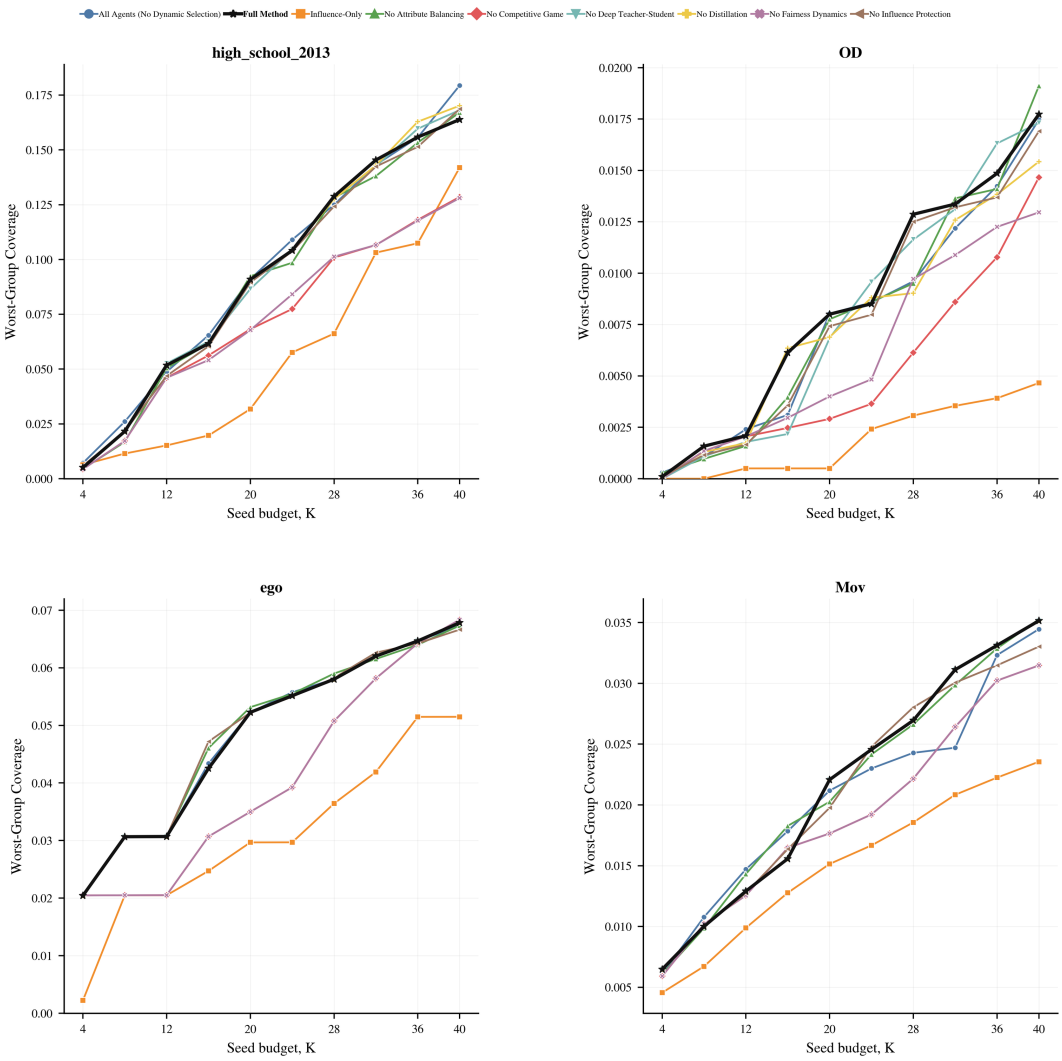

Figure 19: Selection-stage structural variants: macro worst-group coverage.