APPROXIMATION AND LEARNING IN NETWORK SECURITY GAMES WITH CONTAGION AND UNCERTAINTY

Anonymous authors
Paper under double-blind review

ABSTRACT

Network defense under contagious attacks couples resource allocation with learning, as protecting a node can prevent damage and reveal its otherwise unknown value. We study this problem under a budget constraint and feedback from defended nodes and attacked neighborhoods. For known values, we construct a threshold surrogate that is monotone submodular and quantitatively bounds the original worst-case loss. Enumerating all feasible seeds of size at most three gives a $(1 - 1/e)$ approximation, and a problem-specific maximum-coverage reduction establishes matching hardness for the surrogate with general defending requirements. For unknown values, directly substituting upper confidence bounds into this surrogate is not optimistic because its normalization also depends on the unknown values. We resolve this issue with a normalization fixed by known support bounds. With this oracle, PatchLearn admits an $O(\|\mathbf{d}\|_2\sqrt{BT\log(nT)})$ upper bound on surrogate approximation regret, where B bounds the number of defended nodes and $\mathbf{d}$ captures propagation and value scales. Its proof uses observations of selected nodes and requires no uniform observation frequency. Experiments on synthetic and real-world networks assess computational efficiency and show that PatchLearn achieves lower defense loss than baselines using the same UCB rule in the tested settings with heterogeneous node values.

1 INTRODUCTION

Modern networked systems, such as online information and content propagation networks Ju et al. (2022); Chen et al. (2024) and large-scale communication networks with dynamic traffic patterns Lv et al. (2024); Lu et al. (2025), are increasingly vulnerable to localized attacks whose impact propagates through underlying network structures. In such environments, targeting a single node can influence a much larger portion of the system through structural dependencies, which makes defensive decision making inherently challenging. Network security games provide a natural framework for modeling such problems, where a defender allocates limited resources across graph nodes to reduce potential damage while an attacker selects targets to maximize loss Schlenker et al. (2018); Kumar et al. (2010); Gan et al. (2015); Jain et al. (2011). These games have been widely applied in cybersecurity Wang et al. (2020); Bai et al. (2022), counter-terrorism Kiekintveld et al. (2009); Korzhyk et al. (2010), public safety Fang et al. (2015); Bai et al. (2025), and resource protection in complex graph structures. When attacks propagate through the network, the effect of a single attack extends beyond the initially targeted node and influences a larger region of the system. Prior work has studied such propagation effects in network security settings Aspnes et al. (2005); Bai et al. (2021), demonstrating their importance in modeling cascading failures and adversarial influence.

Despite their practical relevance, most existing works on network security games assume complete information, where node values are known in advance. This assumption is often unrealistic in practice, as node importance is uncertain and varies over time due to dynamic traffic demand, user behavior, and external conditions. For example, in communication networks, the load associated with a node can fluctuate significantly, and the impact of a disruption depends on these time-varying factors. Such quantities are not known a priori and must be learned through interactions over time, naturally leading to stochastic models with unknown distributions.

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

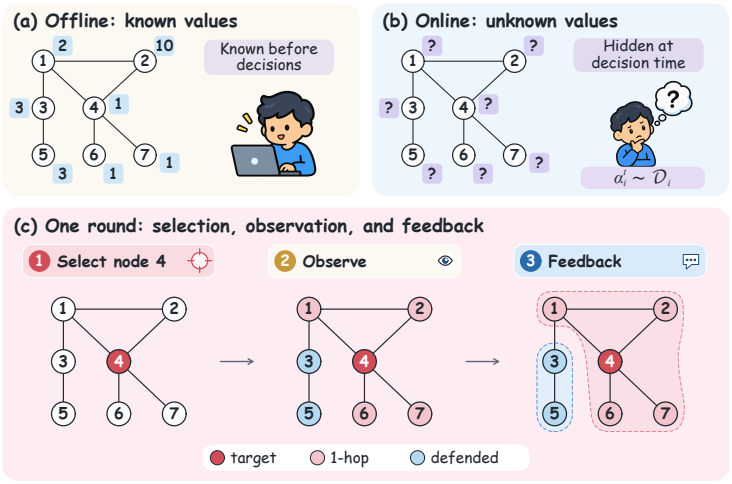

Figure 1: Motivating example (spread level $k = 1$). (a) Full-information setting, where node values are known prior to decision making. (b) Online setting, where $\alpha_i^t \sim \mathcal{D}_i$ and both the realizations and the underlying distributions are unknown.

Moreover, the observation process is inherently localized. After an attack on a node, its impact propagates through the network and exposes the states of nodes in its k-hop neighborhood, while the rest of the system remains unobserved. In communication networks, defensive actions such as deploying protection or monitoring mechanisms are often coupled with inspection capabilities. When sufficient resources are allocated to a node, monitoring modules can be activated to allow the defender to observe its current load. As a result, the defender must learn node values from structured partial feedback that combines attack-induced observations with resource-dependent inspection (Figure 1).

Recent works have started to relax the complete information assumption. Nguyen et al. Nguyen et al. (2013) study adversarial models under partial observability where the attacker lacks full information, and Gan et al. Gan et al. (2022) consider sequential interactions with Bayesian signaling. However, these works do not incorporate propagation together with multi-round learning under structured feedback. More broadly, while cascading processes and influence propagation have been widely studied Xing (2021); Zhou et al. (2024), integrating these dynamics into security games with unknown node values remains largely unexplored.

1.1 Our Results

We study a new multi-round network security game where node values and their underlying distributions are initially unknown and must be learned through repeated interactions under contagious attacks and structured partial feedback. Even under the full information setting where all node values are known, directly optimizing the defending result remains computationally challenging, and existing works only provide limited bi-criteria guarantees. To address this issue, we introduce a threshold-based surrogate reformulation and establish an explicit connection between the surrogate objective and the original defending result, showing that optimizing the surrogate suppresses excess worst-case loss above a target threshold.

Offline Results. We first consider the offline setting where node values are known. We prove that computing the optimal surrogate solution remains NP-hard. We then show that the surrogate objective is monotone and submodular, which enables efficient optimization under a knapsack constraint. Based on this property, we develop a greedy surrogate allocation algorithm that achieves a $(1 - 1/e)$ approximation guarantee in polynomial time.

Online Results. We next consider the online setting where node values are unknown and must be learned over time under structured partial feedback. We develop an online learning framework, PatchLearn, which combines upper confidence bound estimation with the offline surrogate oracle to jointly handle exploration and exploitation. We establish a sublinear ρ-regret bound of order

$O\big(\sqrt{T \log T}\big)$, where ρ is the approximation ratio of the offline oracle, implying that the average per-round performance converges to the offline benchmark over time.

Experiments. We conduct experiments on both synthetic and real-world networks. The results demonstrate that the proposed framework achieves strong defending performance with significantly improved computational efficiency and stable online learning behavior.

1.2 Related Work

Contagious attacks in security games have been extensively studied, driven by applications aiming to limit viral spread Tsai et al. (2012); Acemoglu et al. (2016); Bai et al. (2023; 2024). These works analyze computational hardness, externalities, and optimal allocations under contagion, typically assuming complete information in which both players know the graph and node parameters (values and defending requirements), allowing full-information optimization. A complementary direction considers boundedly rational attackers Shieh et al. (2012); Zhang et al. (2021); Zychowski & Mandziuk (2021), where attack probabilities depend on node payoffs. Although incorporating behavior, these models still rely on known payoffs and do not learn unknown node values through feedback.

Another line explores repeated interaction and dynamic defense. Multi-round security games integrate behavioral probability weighting under resource limits and show that regret matching can converge to correlated equilibria Wang et al. (2023); Abdallah & Hu (2023). However, these focus on equilibrium outcomes without modeling contagion or evaluating regret against offline baselines. Other works treat defense as a gradual rollout with delays across neighbors (Basilico et al., 2009; Yin et al., 2015), capturing temporal friction but assuming known parameters at planning time. In contrast, our model reveals node values only after actions are taken and evaluates performance via regret relative to an offline benchmark.

2 Problem formulation

Let $G = (V, E)$ be a graph with $n = |V|$. Each target u has a fixed attack region $\mathcal{N}(u) \subseteq V$ containing u. For undirected graphs, $\mathcal{N}(u) = N_k(u)$ is the closed k-hop neighborhood; for directed graphs, $\mathcal{N}(u) = N_k^+(u)$ follows outgoing edges. The analysis uses these fixed sets and does not require symmetry. This is a region-based contagion model: protection prevents loss at a defended node but does not remove that node from propagation paths.

A node v has a known positive defending requirement θ_v. Allocating less than θ_v has no protective effect, so we select a defended set

$$S \in \mathcal{F} := \left\{ S \subseteq V : \sum_{v \in S} \theta_v \le R \right\}, \qquad r_v = \theta_v \mathbf{1}\{v \in S\}, \tag{1}$$

where $R \ge 0$ is the total budget. For a nonnegative value vector $\mathbf{a}$, define

$$L_u(S; \mathbf{a}) = \sum_{v \in \mathcal{N}(u) \setminus S} a_v, \qquad L(S; \mathbf{a}) = \max_{u \in V} L_u(S; \mathbf{a}).$$

The offline optimum is $W_{\text{opt}} := \min_{S \in \mathcal{F}} L(S; \boldsymbol{\alpha})$ for known values $\boldsymbol{\alpha}$.

Stochastic values and decision timing. At round t, node values $\alpha_v^t \in [0, b_v]$ have unknown means α_v and known finite support bounds b_v. Value vectors are independent across rounds and identically distributed; independence across nodes is not necessary. The defender chooses S_t from past observations. After seeing S_t, the attacker selects u_t, possibly using its own estimates and the interaction history, but before observing current-round values. Values are then realized and the defender observes

$$O_t = \mathcal{N}(u_t) \cup S_t.$$

Thus observation selection cannot depend on the current unobserved realizations. The attacker may observe its attacked neighborhood; defender-only monitoring is an explicit modeling assumption. The regret proof allows any such nonanticipating attack policy, not only a best response.

Three distinct performance quantities. The realized loss at the selected target is $L_{u_t}(S_t; \boldsymbol{\alpha}^t)$. Conditional on the history and both chosen actions, its expectation equals $L_{u_t}(S_t; \boldsymbol{\alpha})$ and is bounded by $L(S_t; \boldsymbol{\alpha})$. Our loss certificate concerns this worst-case loss at the mean vector. It does not identify $L(S_t; \boldsymbol{\alpha})$ with $\mathbb{E}[\max_u L_u(S_t; \boldsymbol{\alpha}^t)]$, which would allow the target to depend on current realizations. The online regret defined below concerns a surrogate evaluated at the mean vector. We report empirical loss and surrogate performance as separate measurements.

3 Threshold surrogates and offline optimization

3.1 Surrogate construction and loss certificate

For a threshold $W \geq 0$, write $A_u = \sum_{v \in \mathcal{N}(u)} \alpha_v$, $P_u(S) = \sum_{v \in \mathcal{N}(u) \cap S} \alpha_v$, and $\delta_u(W) = (A_u - W)_+$. The condition $L(S; \boldsymbol{\alpha}) \leq W$ is equivalent to $P_u(S) \geq \delta_u(W)$ for every u. Define

$$q_{u,W}(S; \boldsymbol{\alpha}) = \begin{cases} \min\{P_u(S)/\delta_u(W), 1\}, & \delta_u(W) > 0, \\ 1, & \delta_u(W) = 0, \end{cases} \qquad q_W(S; \boldsymbol{\alpha}) = \sum_{u \in V} q_{u,W}(S; \boldsymbol{\alpha}). \quad (2)$$

The zero-denominator convention counts an already satisfied constraint as satisfied. Hence $q_W = n$ is exactly a threshold certificate, while $n - q_W$ measures the aggregate deficit.

Lemma 3.1 (Loss certificate)**.** *For every feasible S and $W \geq 0$,*

$$L(S; \boldsymbol{\alpha}) \leq W + \delta_{\max}(W)\big(n - q_W(S; \boldsymbol{\alpha})\big), \quad \delta_{\max}(W) = \max_u \delta_u(W).$$

Proof. If $\delta_u = 0$, then $L_u \leq A_u \leq W$. Otherwise, an unsaturated term satisfies $L_u - W = \delta_u(1 - q_{u,W})$, and a saturated term implies $L_u \leq W$. Taking the maximum and bounding each nonnegative local deficit by their sum proves the claim. □

For normalized coverage $q_W/n = 1 - \varepsilon$, the certified excess loss is at most $n\delta_{\max}\varepsilon$. Near-full coverage is informative but is not an exact feasibility certificate.

3.2 Approximation and hardness

Theorem 3.2 (Structure and approximation)**.** *For fixed W and known $\boldsymbol{\alpha}$, $q_W(\cdot; \boldsymbol{\alpha})$ is nonnegative, monotone, and submodular. Algorithm 1 with full enumeration returns $\widehat{S} \in \mathcal{F}$ satisfying*

$$q_W(\widehat{S}; \boldsymbol{\alpha}) \geq (1 - 1/e) \max_{S \in \mathcal{F}} q_W(S; \boldsymbol{\alpha}). \quad (3)$$

It uses $O(n^5)$ objective evaluations. The same guarantee holds for the fixed-normalization surrogate in Section 4.

Proof. For $\delta_u(W) > 0$, $P_u(S)$ is nonnegative and modular, and $z \mapsto \min\{z/\delta_u(W), 1\}$ is nondecreasing and concave. Thus, for $A \subseteq B$ and $i \notin B$, adding i gives at least as much gain at A as at B. If $i \notin \mathcal{N}(u)$, both gains are zero. When $\delta_u(W) = 0$, the local term is constant at one. Summing proves nonnegativity, monotonicity, and submodularity. The same argument applies to $\bar{q}_W(\cdot; \mathbf{a})$ for fixed nonnegative $\mathbf{a}$, since $\bar{\delta}_u(W)$ is fixed.

For either surrogate, all costs are positive and (1) is a single knapsack constraint. Algorithm 1 enumerates every feasible seed of size at most three and completes it by maximum marginal gain per cost. Removing unaffordable items in advance is equivalent to inspecting and discarding them during greedy completion, because the remaining budget only decreases. Stopping at zero gain preserves the objective, and completing seeds of size one or two cannot reduce their values. The returned value is therefore at least that of the partial-enumeration algorithm in Theorem 1 of Sviridenko (2004), which gives (3).

There are $O(n^3)$ seeds, at most n additions per completion, and at most n marginal evaluations per addition, giving $O(n^5)$ objective evaluations. □

The ratio concerns surrogate maximization and does not imply a multiplicative approximation for L. Appendix A discusses implementation costs.

Algorithm 1 Patching-GSA$(f, \boldsymbol{\theta}, R, C)$

Require: Objective f, requirements $\boldsymbol{\theta}$, budget R, seed pool C

$\widehat{S} \leftarrow \varnothing$
for each $A \subseteq C$ with $|A| \leq 3$ and $\sum_{v \in A} \theta_v \leq R$ **do**
  $S \leftarrow A$; $F \leftarrow \{v \in V \setminus S : \theta_v + \sum_{i \in S} \theta_i \leq R\}$
  while $F \neq \varnothing$ **do**
    $v \leftarrow \arg\max_{i \in F} [f(S \cup \{i\}) - f(S)]/\theta_i$
    if $f(S \cup \{v\}) = f(S)$ **then**
      break
    $S \leftarrow S \cup \{v\}$
    $F \leftarrow \{v \in V \setminus S : \theta_v + \sum_{i \in S} \theta_i \leq R\}$
  if $f(S) > f(\widehat{S})$ **then**
    $\widehat{S} \leftarrow S$
return $\widehat{S}$

Full and Fast variants. Full Patching-GSA uses all affordable nodes as C. Fast Patching-GSA chooses a smaller pool of M affordable nodes ranked by singleton gains $f(\{v\}) - f(\varnothing)$; greedy augmentation still searches all affordable nodes. A practical pool rule is $M \leq \min\{100, \lceil\sqrt{n}\rceil\}$. Its objective-evaluation cost is $O(M^3 n^2)$. This restriction sacrifices the complete-enumeration guarantee; we do not assign the full version's approximation ratio to the fast version. These costs count objective evaluations, not constant-time arithmetic. Maintaining reverse incidence lists makes a marginal update local to regions containing the candidate node.

Theorem 3.3 (Problem-specific hardness)**.** *Maximizing q_W is NP-hard even on undirected one-hop graphs with identical defending requirements. With general defending requirements, for every fixed $\varepsilon \in (0, 1/e)$, no polynomial-time algorithm achieves ratio $1 - 1/e + \varepsilon$ unless $P = NP$.*

Appendix B proves the first statement by reducing Vertex Cover and the tightness result through an incidence graph for maximum coverage. Replicated element nodes suppress the solution-independent contribution, preserving the gap of Feige (1998). Tightness applies to general costs, not every restricted graph family.

Threshold certification. An oracle returning $q_W = n$ certifies feasibility, but $q_W < n$ does not certify infeasibility. Initialize $U = \max_u A_u$ with the feasible empty set. Threshold search retains the best pair (U, S_U) and updates it only after checking $L(S_U; \boldsymbol{\alpha}) \leq U$. Failed queries guide the search without establishing lower bounds. Thus $W_{\text{opt}} \leq L(S_{\widehat{W}}; \boldsymbol{\alpha}) \leq \widehat{W}$, but $\widehat{W}$ need not be the minimum feasible threshold. Appendix C gives the procedure and its logarithmic query bound.

4 Learning with fixed normalization

4.1 Fixed normalization and online allocation

The offline denominator $\delta_u(W)$ depends on unknown means. Consider two nodes joined by an undirected edge with $k = 1$. Both attack regions contain both nodes. With values $(0.4, 0.4)$, defending the first node at $W = 0.2$ gives each term $q_{u,W} = 2/3$. Increasing the vector to $(0.5, 0.8)$ lowers each term to $5/11$, so q_W decreases from $4/3$ to $10/11$. Thus coordinatewise upper bounds on values need not upper-bound q_W.

Fix $W \geq 0$ before online interaction without access to the unknown mean vector, and use the known support vector $\mathbf{b}$ to define

$$\bar{\delta}_u(W) = \left(\sum_{v \in \mathcal{N}(u)} b_v - W \right)_+, \qquad \bar{q}_{u,W}(S; \mathbf{a}) = \begin{cases} \min\left\{ \dfrac{\sum_{v \in \mathcal{N}(u) \cap S} a_v}{\bar{\delta}_u(W)}, 1 \right\}, & \bar{\delta}_u(W) > 0, \\ 1, & \bar{\delta}_u(W) = 0. \end{cases}$$

Set $\bar{q}_W = \sum_u \bar{q}_{u,W}$. Unlike q_W, this objective is coordinatewise nondecreasing in $\mathbf{a}$ for every fixed S. The support bounds determine only the normalization, not the unknown numerator. Write

Algorithm 2 PatchLearn($\mathcal{N}, T, \boldsymbol{\theta}, R, W, \mathbf{b}$)

Require: Regions $\mathcal{N}$, horizon T, requirements $\boldsymbol{\theta}$, budget R, fixed W, known $\mathbf{b}$

Initialize $N_v(0) = 0$, $\widehat{\alpha}_v(0) = 0$ for all v; fix $\bar{\delta}_u(W)$
$C \leftarrow \{v \in V : \theta_v \le R\}$
for $t = 1, \dots, T$ **do**
  Compute $\mathbf{U}(t)$ using (5)
  $S_t \leftarrow$ Patching-GSA$\big(\bar{q}_W(\cdot; \mathbf{U}(t)), \boldsymbol{\theta}, R, C\big)$ ▷ Algorithm 1
  Deploy $r_v^t = \theta_v \mathbf{1}\{v \in S_t\}$; the attacker selects u_t
  Observe α_v^t for $v \in O_t = \mathcal{N}(u_t) \cup S_t$
  for $v \in O_t$ **do**
    $N_v(t) \leftarrow N_v(t-1) + 1$
    $\widehat{\alpha}_v(t) \leftarrow [N_v(t-1)\widehat{\alpha}_v(t-1) + \alpha_v^t]/N_v(t)$
  Carry forward counts and means for $v \notin O_t$

$\bar{\delta}_{\max}(W) = \max_u \bar{\delta}_u(W)$. Since $\bar{\delta}_u \ge \delta_u$, the argument of Lemma 3.1 gives

$$L(S; \boldsymbol{\alpha}) \le W + \bar{\delta}_{\max}(W)\big(n - \bar{q}_W(S; \boldsymbol{\alpha})\big). \tag{4}$$

Full $\bar{q}_W$ coverage is sufficient but need not be necessary for $L \le W$. Loose support bounds can make this certificate conservative. The fixed W is a deployment target or a value selected from independent calibration information; the learner does not run threshold search on the unknown mean vector.

Clipped upper confidence bounds. Let $N_v(t-1)$ count observations before round t, and let $\widehat{\alpha}_v(t-1)$ be their empirical mean. With $T \ge 2$ and $\beta = \log(2nT^3)$, use

$$U_v(t) = \begin{cases} b_v, & N_v(t-1) = 0, \\ \min\left\{ b_v, \widehat{\alpha}_v(t-1) + b_v \sqrt{\dfrac{\beta}{2N_v(t-1)}} \right\}, & N_v(t-1) > 0. \end{cases} \tag{5}$$

This initialization does not require free observations or an unreported exploration phase. Nodes with $b_v = 0$ are known exactly.

Algorithm 2 invokes Full Patching-GSA (Algorithm 1) at every round. The regret analysis also covers replacing this call with a general approximation oracle $\mathcal{A}$. Using a restricted seed pool instead gives the Fast variant subject to the optimization-error term in Appendix D.4.

4.2 Regret analysis

Definition 4.1 (Surrogate approximation regret)**.** For $S^\star \in \arg\max_{S \in \mathcal{F}} \bar{q}_W(S; \boldsymbol{\alpha})$ and $\rho \in (0, 1]$, define

$$\mathcal{R}_\rho(T) = T\rho \bar{q}_W(S^\star; \boldsymbol{\alpha}) - \sum_{t=1}^{T} \mathbb{E}[\bar{q}_W(S_t; \boldsymbol{\alpha})].$$

This quantity may be negative when the learner exceeds the ρ-scaled benchmark.

Let $V_R = \{v : \theta_v \le R\}$, $B = \max_{S \in \mathcal{F}} |S|$, and

$$c_v = \sum_{u : v \in \mathcal{N}(u),\, \bar{\delta}_u(W) > 0} \frac{1}{\bar{\delta}_u(W)}, \qquad d_v = b_v c_v \quad (v \in V_R). \tag{6}$$

Empty sums are zero. The vector $\mathbf{d}$ is restricted to affordable nodes.

Theorem 4.2 (Selected-node regret)**.** *Suppose the allocation step in Algorithm 2 uses an oracle $\mathcal{A}$ that returns a feasible ρ-approximate maximizer of $\bar{q}_W(\cdot; \mathbf{a})$ for every $\mathbf{a} \in [0, \mathbf{b}]$. Under the timing and feedback in Section 2, the learner satisfies*

$$\mathcal{R}_\rho(T) \le (1 + 2\sqrt{2\beta})\|\mathbf{d}\|_2 \sqrt{BT} + \frac{n}{T}. \tag{7}$$

In particular, Full Patching-GSA gives $\rho = 1 - 1/e$. No uniform observation-frequency assumption is required.

Proof. Observation decisions precede the current value realization. Each node's observed sequence therefore admits concentration at every sample count, even though the count is adaptive. Let $\mathcal{E}$ be the event that all empirical means satisfy their Hoeffding bounds at every reached sample count up to T. A union bound over at most nT node–count pairs gives

$$\Pr(\mathcal{E}^c) \le 2nTe^{-\beta} = T^{-2}.$$

Appendix D.1 justifies this step without conditioning on a data-dependent count. Clipping at $b_v \ge \alpha_v$ preserves optimism. On $\mathcal{E}$, whenever $N_v(t-1) > 0$,

$$0 \le U_v(t) - \alpha_v \le b_v \sqrt{\frac{2\beta}{N_v(t-1)}}.$$

For an unobserved node, the error is at most b_v.

Coordinatewise monotonicity and the oracle guarantee imply

$$\rho\bar{q}_W(S^\star;\boldsymbol{\alpha}) \le \rho\bar{q}_W(S^\star;\mathbf{U}(t)) \le \bar{q}_W(S_t;\mathbf{U}(t)).$$

For each u with $\bar{\delta}_u > 0$, the truncation $x \mapsto \min\{x/\bar{\delta}_u, 1\}$ is $1/\bar{\delta}_u$-Lipschitz. Terms with zero denominator are constant. Thus, for $\mathbf{a}' \ge \mathbf{a}$,

$$\bar{q}_W(S;\mathbf{a}') - \bar{q}_W(S;\mathbf{a}) \le \sum_{\substack{u\in V\\ \bar{\delta}_u>0}} \frac{1}{\bar{\delta}_u} \sum_{v\in\mathcal{N}(u)\cap S} (a'_v - a_v) = \sum_{v\in S} c_v(a'_v - a_v).$$

Only selected nodes enter this error bound. Combining the two inequalities gives

$$\rho\bar{q}_W(S^\star;\boldsymbol{\alpha}) - \bar{q}_W(S_t;\boldsymbol{\alpha}) \le \bar{q}_W(S_t;\mathbf{U}(t)) - \bar{q}_W(S_t;\boldsymbol{\alpha}) \le \sum_{v\in S_t} c_v(U_v(t) - \alpha_v). \tag{8}$$

Every node on the right is observed at that round. Let $K_v(T) = \sum_{t=1}^{T} \mathbf{1}\{v \in S_t\}$. Before its jth selection, node v has at least $j-1$ observations, irrespective of the attack sequence. Its first selection contributes at most d_v because $U_v \le b_v$. For $K_v(T) \ge 1$, the subsequent confidence errors satisfy

$$\sum_{t:v\in S_t} c_v(U_v(t) - \alpha_v) \le d_v \left(1 + \sqrt{2\beta} \sum_{j=1}^{K_v(T)-1} j^{-1/2}\right) \le (1 + 2\sqrt{2\beta}) d_v \sqrt{K_v(T)}. \tag{9}$$

The contribution is zero if $K_v(T) = 0$. Since $\sum_{v\in V_R} K_v(T) \le BT$, Cauchy–Schwarz gives

$$\sum_{v\in V_R} d_v \sqrt{K_v(T)} \le \|\mathbf{d}\|_2 \sqrt{\sum_{v\in V_R} K_v(T)} \le \|\mathbf{d}\|_2 \sqrt{BT}. \tag{10}$$

Summing (8) and applying (9)–(10) bounds the regret on $\mathcal{E}$. Outside $\mathcal{E}$, each round contributes at most n, so the additional expected contribution is at most $nT\Pr(\mathcal{E}^c) \le n/T$. This proves (7). Extra attack feedback can improve the estimates but is not needed for the bound. For directed graphs, c_v counts attack regions containing v, as required by the sensitivity calculation above. □

What the guarantee implies for loss. Summing (4) gives

$$\frac{1}{T}\sum_{t=1}^{T} \mathbb{E}[L(S_t;\boldsymbol{\alpha})] \le W + \bar{\delta}_{\max}(W) \left(n - \rho\bar{q}_W(S^\star;\boldsymbol{\alpha}) + \frac{\mathcal{R}_\rho(T)}{T}\right). \tag{11}$$

Substituting (7) into (11) gives a vanishing estimation penalty of order $\sqrt{\log(nT)/T}$ for a fixed instance. The approximation and coverage residual $n - \rho\bar{q}_W(S^\star;\boldsymbol{\alpha})$ may remain positive. If $\bar{q}_W(S^\star;\boldsymbol{\alpha}) = n$ and $\rho = 1$, the bound approaches W; otherwise convergence to the original offline loss optimum is not implied. Fast oracles require an additional optimization-error term, made explicit in Appendix D.4.

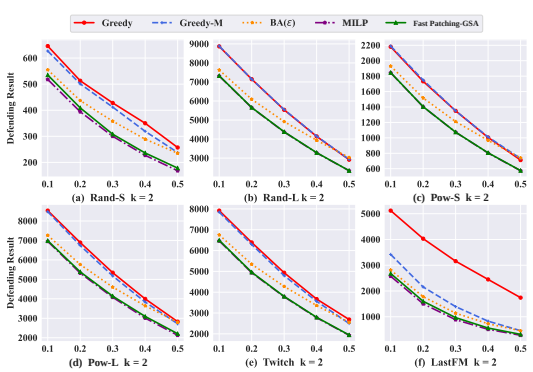

Figure 2: Offline defense loss versus resource ratio η at $k = 2$, using Fast Patching-GSA.

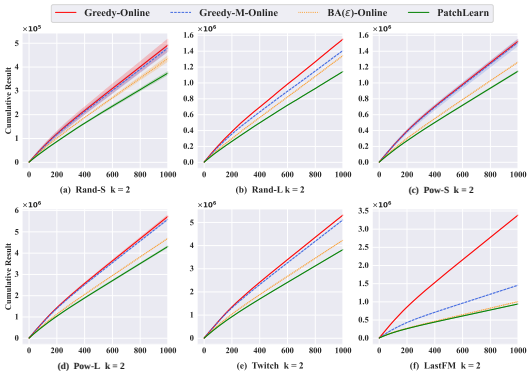

Figure 3: Cumulative defense loss of the mean-normalized plug-in learner and ϵ-greedy baselines at $\eta = 0.4$ and $k = 2$.

5 Experiments

5.1 Experimental setup

Table 1 lists the benchmark datasets and their topology sizes. The suite comprises random graphs (Rand-S/L), power-law graphs (Pow-S/L), and Twitch and LastFM topologies. We evaluate defense quality on the original loss, the computational cost of full enumeration, and learning under partial feedback. The resource ratio is $\eta = R/\sum_v \theta_v$. In this benchmark suite, $k = 2$ unless indicated otherwise; offline comparisons use $\eta \in \{0.1, 0.2, 0.3, 0.4, 0.5\}$ and online comparisons use $\eta = 0.4$, $T = 1000$. Appendix E gives construction details, additional results, and metric definitions.

Baselines and variants. Offline baselines are Greedy, Greedy-M, the bi-criteria approximation BA(ϵ) of Bai et al. (2023), and a mixed-integer linear program (MILP) for the original loss. Greedy prioritizes node values; Greedy-M prioritizes high-value nodes in a currently high-loss attack region. Large-instance Patching-GSA results use the fast variant. We separately compare Full and Fast on small instances. Online UCB baselines use their respective allocation rules with common confidence estimation and the same observation rule. The ϵ-greedy comparisons and mean-normalized plug-in learner are reported separately from the fixed-normalization setting; their loss curves assess empirical performance under that normalization.

5.2 Results and analysis

Offline defense quality and runtime. Fast Patching-GSA closely tracks the MILP comparator in Figure 2. These curves measure original loss, whereas Theorem 3.2 concerns the surrogate value obtained by Full. Table 2 isolates the effect of restricting the seed pool. Full matches MILP loss on three of four instances; Fast differs from Full by 1, 1, 4, and 0 loss units. The comparison supports a practical quality–runtime tradeoff, not preservation of the full approximation guarantee. With the same greedy completion and tie-breaking rule, the larger seed family cannot decrease the best surrogate value at a fixed threshold, but this alone does not order the original losses after separate threshold searches. Appendix E.5 extends this comparison to both $k = 1$ and $k = 2$.

Online defense performance. Figure 3 compares cumulative defense loss over $T = 1000$ rounds. The mean-normalized plug-in version of PatchLearn has the lowest final cumulative loss on all six datasets. BA(ϵ)-Online is the closest competitor, while the gap to Greedy-Online is especially large on LastFM. These comparisons use ϵ-greedy baselines, so the observed differences reflect both allocation and exploration. They evaluate practical defense performance under the plug-in normalization rather than the fixed-normalization regret bound in Theorem 4.2. The following comparisons use a common UCB rule to examine allocation under matched exploration.

Table 1: Datasets

Dataset	Nodes	Edges
Rand-S	200	1068
Rand-L	2000	100030
Pow-S	500	3926
Pow-L	2000	15922
Twitch	1912	31299
LastFM	7624	27806

Table 2: Full versus Fast on individual small instances. Entries give original loss / runtime (s).

Instance	MILP	Full	Fast
Rand-S-50	41 / 0.0304	41 / 4.2249	42 / 0.1495
Pow-S-50	99 / 0.0144	99 / 2.1115	100 / 0.0003
Rand-S-100	82 / 0.1292	83 / 79.0942	87 / 0.0860
Pow-S-100	157 / 0.0543	157 / 54.4544	157 / 0.0008

Table 3: Cumulative defense loss under heterogeneous values with UCB baselines.

Graph	Values	Greedy-UCB	Greedy-M-UCB	BA-UCB	PatchLearn
Twitch	Lognormal	141.94	144.95	139.23	**114.83**
Twitch	Degree-correlated	264.05	279.39	226.08	**196.95**
LastFM	Lognormal	73.17	55.62	68.74	**40.03**
LastFM	Degree-correlated	114.89	91.10	75.34	**65.16**

Heterogeneous values and matched exploration. On connected samples from Twitch and LastFM, we replace uniform values with lognormal and degree-correlated node values, normalized to have mean 5.5 across nodes. Table 3 compares allocation rules with UCB exploration. PatchLearn achieves lower reported cumulative loss in all four settings, with improvements over the strongest baseline of 17.5%, 12.9%, 28.0%, and 13.5%, respectively. The common exploration rule makes these comparisons more informative about allocation than comparisons against ϵ-greedy alone. These runs evaluate the mean-normalized plug-in variant; they are empirical evidence about allocation under heterogeneous values, not a test of the fixed-normalization theorem.

Fixed-normalization evaluation. A separate evaluation on Rand-S also yields lower average defense loss for the fixed-normalization learner than for the UCB baselines. Appendix E.3 presents the experimental setup and analyzes the results.

Threshold sensitivity and feasibility. The Rand-S sensitivity study shows that near-full surrogate coverage can still violate the requested loss threshold. Appendix E.4 reports the full results. These results certify the returned defenses without establishing the minimum feasible threshold.

Small residual thresholds. In the sensitivity experiment, integer node values and integer W make every positive offline residual $\delta_u(W)$ at least one. The zero-residual convention in (2) handles already satisfied regions separately. This integer lower bound does not extend to noninteger online means or to arbitrary support bounds. For the fixed-normalization learner, a small positive $\bar{\delta}_u(W)$ increases the sensitivity coefficients in (6) and therefore weakens the regret constant. It does not change the stated dependence on T when the graph, supports, and threshold are fixed. Threshold choice consequently affects both the loss target and the informativeness of the guarantee.

6 Conclusion

We connect a threshold-based defense certificate with a learning objective that supports valid optimism. The offline surrogate admits a $(1 - 1/e)$ approximation with matching hardness under general costs. With the guaranteed oracle, fixed normalization gives a sublinear upper bound on surrogate approximation regret using only observations of selected defenses. Experiments illustrate the quality–runtime tradeoff between Full and Fast and show the benefit of the proposed allocation rule under matched UCB exploration.

In the future, we plan to develop more efficient allocation oracles with provable guarantees and investigate adaptive threshold mechanisms while maintaining valid confidence bounds. We also aim to extend the analysis to settings with time-varying node values and costly monitoring, enabling the joint optimization of information acquisition and defense allocation.

AI USE STATEMENT

Generative AI tools were used for language polishing and LaTeX assistance. All scientific claims, experimental results, mathematical arguments, and the final manuscript were reviewed and verified by the authors.

REFERENCES

Mustafa Abdallah and Qin Hu. Should I regret more? A regret-based multi-round learning with behavioral human players in a multi-target security game. In *ECC*, pp. 1–6. IEEE, 2023.

Daron Acemoglu, Azarakhsh Malekian, and Asuman E. Ozdaglar. Network security and contagion. *J. Econ. Theory*, 166:536–585, 2016.

James Aspnes, Kevin L. Chang, and Aleksandr Yampolskiy. Inoculation strategies for victims of viruses and the sum-of-squares partition problem. In *SODA*, pp. 43–52. SIAM, 2005.

Rufan Bai, Haoxing Lin, Xinyu Yang, Xiaowei Wu, Minming Li, and Weijia Jia. Defending against contagious attacks on a network with resource reallocation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, 2021.

Rufan Bai, Haoxing Lin, Xinyu Yang, Xiaowei Wu, Minming Li, and Weijia Jia. Mixed strategies for security games with general defending requirements. In *IJCAI*, pp. 46–52. ijcai.org, 2022.

Rufan Bai, Haoxing Lin, Xinyu Yang, Xiaowei Wu, Minming Li, and Weijia Jia. Stackelberg security games with contagious attacks on a network: Reallocation to the rescue. *J. Artif. Intell. Res.*, 77:487–515, 2023.

Rufan Bai, Chao Xu, Chenyang Xu, and Ruilong Zhang. Resource-limited network security games with general contagious attacks. In *COCOON (2)*, volume 15162 of *Lecture Notes in Computer Science*, pp. 79–91. Springer, 2024.

Rufan Bai, Haoxing Lin, Xiaowei Wu, Minming Li, and Weijia Jia. On the computation of mixed strategies for security games with general defending requirements. *Artif. Intell.*, 341:104297, 2025.

Nicola Basilico, Nicola Gatti, and Francesco Amigoni. Leader-follower strategies for robotic patrolling in environments with arbitrary topologies. In *AAMAS (1)*, pp. 57–64. IFAAMAS, 2009.

Hao Chen, Yuanchen Bei, Qijie Shen, Yue Xu, Sheng Zhou, Wenbing Huang, Feiran Huang, Senzhang Wang, and Xiao Huang. Macro graph neural networks for online billion-scale recommender systems. In *WWW*, pp. 3598–3608. ACM, 2024.

Fei Fang, Peter Stone, and Milind Tambe. When security games go green: Designing defender strategies to prevent poaching and illegal fishing. In *IJCAI*, pp. 2589–2595. AAAI Press, 2015.

Uriel Feige. A threshold of ln n for approximating set cover. *Journal of the ACM*, 45(4):634–652, 1998. doi: 10.1145/285055.285059.

Jiarui Gan, Bo An, and Yevgeniy Vorobeychik. Security games with protection externalities. In *AAAI*, pp. 914–920. AAAI Press, 2015.

Jiarui Gan, Rupak Majumdar, Goran Radanovic, and Adish Singla. Bayesian persuasion in sequential decision-making. In *AAAI*, pp. 5025–5033. AAAI Press, 2022.

Manish Jain, Dmytro Korzhyk, Ondrej Vanek, Vincent Conitzer, Michal Pechoucek, and Milind Tambe. A double oracle algorithm for zero-sum security games on graphs. In *AAMAS*, pp. 327–334, 2011.

Chunhua Ju, Chenyu Wang, Yihao Jiang, Fuguang Bao, Huajian Zhou, and Chonghuan Xu. Exploring a multi-layer coupled network propagation model based on information diffusion and bounded trust. *International Journal of Public Health*, 67:1604887, 2022.

Christopher Kiekintveld, Manish Jain, Jason Tsai, James Pita, Fernando Ordóñez, and Milind Tambe. Computing optimal randomized resource allocations for massive security games. In *AAMAS (1)*, pp. 689–696. IFAAMAS, 2009.

Dmytro Korzhyk, Vincent Conitzer, and Ronald Parr. Complexity of computing optimal stackelberg strategies in security resource allocation games. In *AAAI*. AAAI Press, 2010.

V. S. Anil Kumar, Rajmohan Rajaraman, Zhifeng Sun, and Ravi Sundaram. Existence theorems and approximation algorithms for generalized network security games. In *ICDCS*, pp. 348–357. IEEE Computer Society, 2010.

Yuhuan Lu, Zhen Zhang, Rufan Bai, Han Liu, and Wei Wang. Dual-view interaction-aware lane change prediction for autonomous driving. In *AAAI*, pp. 28222–28230. AAAI Press, 2025.

Wenkai Lv, Pengfei Yang, Tianyang Zheng, Chengmin Lin, Zhenyi Wang, Minwen Deng, and Quan Wang. Graph-reinforcement-learning-based dependency-aware microservice deployment in edge computing. *IEEE Internet Things J.*, 11(1):1604–1615, 2024.

Thanh Hong Nguyen, Rong Yang, Amos Azaria, Sarit Kraus, and Milind Tambe. Analyzing the effectiveness of adversary modeling in security games. In *AAAI*. AAAI Press, 2013.

Aaron Schlenker, Omkar Thakoor, Haifeng Xu, Fei Fang, Milind Tambe, Long Tran-Thanh, Phebe Vayanos, and Yevgeniy Vorobeychik. Deceiving cyber adversaries: A game theoretic approach. In *AAMAS*, pp. 892–900, 2018.

Eric Shieh, Bo An, Rong Yang, Milind Tambe, Craig Baldwin, Joseph DiRenzo, Ben Maule, and Garrett Meyer. PROTECT: a deployed game theoretic system to protect the ports of the united states. In *AAMAS*, pp. 13–20. IFAAMAS, 2012.

Maxim Sviridenko. A note on maximizing a submodular set function subject to a knapsack constraint. *Operations Research Letters*, 32(1):41–43, 2004.

Jason Tsai, Thanh Hong Nguyen, and Milind Tambe. Security games for controlling contagion. In *AAAI*. AAAI Press, 2012.

Hao Wang, Guangjie Han, Weizhe Lai, Yun Hou, and Chuan Lin. A multi-round game-based source location privacy protection scheme with AUV enabled in underwater acoustic sensor networks. *IEEE Trans. Veh. Technol.*, 72(6):7728–7742, 2023.

Kai Wang, Andrew Perrault, Aditya Mate, and Milind Tambe. Scalable game-focused learning of adversary models: Data-to-decisions in network security games. In *AAMAS*, pp. 1449–1457. International Foundation for Autonomous Agents and Multiagent Systems, 2020.

Liudong Xing. Cascading failures in internet of things: Review and perspectives on reliability and resilience. *IEEE Internet Things J.*, 8(1):44–64, 2021.

Yue Yin, Haifeng Xu, Jiarui Gan, Bo An, and Albert Xin Jiang. Computing optimal mixed strategies for security games with dynamic payoffs. In *IJCAI*, pp. 681–688. AAAI Press, 2015.

Jing Zhang, Yan Wang, and Jun Zhuang. Modeling multi-target defender-attacker games with quantal response attack strategies. *Reliab. Eng. Syst. Saf.*, 205:107165, 2021.

Mingyang Zhou, Weiji Cao, Hao Liao, and Rui Mao. Motif-oriented influence maximization for viral marketing in large-scale social networks. In *NeurIPS*, 2024.

Adam Zychowski and Jacek Mandziuk. Learning attacker's bounded rationality model in security games. In *ICONIP (5)*, volume 1516 of *Communications in Computer and Information Science*, pp. 530–539. Springer, 2021.

A OFFLINE IMPLEMENTATION DETAILS

The proof of Theorem 3.2 appears in the main text. Its structural argument requires neither symmetric attack regions nor a particular graph representation.

The $O(n^5)$ bound counts objective evaluations. If a fresh evaluation costs C_q, a direct implementation takes $O(n^5 C_q)$ time in addition to preprocessing. Reverse incidence lists $\{u : v \in \mathcal{N}(u)\}$ permit incremental marginal computations. Restricting only the seed pool to M nodes gives $O(M^3 n^2)$ evaluations but removes the complete-enumeration premise of the approximation theorem.

B PROOF OF PROBLEM-SPECIFIC HARDNESS

B.1 NP-HARDNESS WITH UNIFORM COSTS

Take a Vertex Cover decision instance $H = (U, F)$ with budget K. Subdivide each edge $e = \{i, j\}$ by introducing z_e and replacing the edge with $\{i, z_e\}$ and $\{z_e, j\}$. Set $k = 1$, $W = 1$, $R = K$, and all defending requirements to one. Original nodes have value one; subdivision nodes have value zero.

The closed one-hop neighborhood of z_e has total value two, so $\delta_{z_e} = 1$ and

$$q_{z_e,W}(S) = \min\{\mathbf{1}\{i \in S\} + \mathbf{1}\{j \in S\}, 1\}.$$

Each original node has total neighborhood value one and hence contributes the constant one. Selecting a subdivision node has no effect on the objective because its value is zero. Thus

$$q_W(S) = |U| + |\{e \in F : e \text{ has an endpoint in } S \cap U\}|.$$

A feasible set attains $|U| + |F|$ if and only if H has a vertex cover of size at most K. This proves NP-hardness of surrogate maximization even for uniform costs and one-hop undirected propagation.

B.2 A GAP-PRESERVING REDUCTION FOR TIGHTNESS

Let a maximum-coverage instance consist of a nonempty universe $\mathcal{U}$, sets $F_1, \ldots, F_m$, and a cardinality budget $K \geq 1$. For any fixed $\varepsilon \in (0, 1/e)$, the maximum-coverage hardness result of Feige (1998) rules out distinguishing instances in which K sets cover all elements from instances in which at most $(1 - 1/e + \varepsilon/2)|\mathcal{U}|$ elements can be covered, unless $P = NP$.

Write $d_e = |\{i : e \in F_i\}|$ and $D = \max\{2, \max_e d_e\}$. Construct an undirected graph as follows.

1. Introduce one set node s_i for each F_i, with value one and defending requirement one.
2. For each e, introduce $D - d_e$ padding nodes, each with value one and requirement $K + 1$. These padding nodes are shared by all copies associated with this element.
3. Introduce M element nodes $z_{e,1}, \ldots, z_{e,M}$ for each e, each with value zero and requirement $K+1$. Connect every copy to the set nodes containing e and to the padding nodes associated with e. There are no other edges.
4. Set $R = K$, $k = 1$, and $W = D - 1$.

Only set nodes are affordable. Every element node has neighborhood value D, hence residual threshold one. Its surrogate term is exactly the indicator that a selected set covers its element. Every set or padding node has neighborhood value one, since its other neighbors have value zero. Because $W \geq 1$, all these nodes contribute the constant one. If $C = m + \sum_e (D - d_e)$, then

$$q_W(S) = C + M \left| \bigcup_{i : s_i \in S} F_i \right|.$$

The constant C does not grow with M; introducing new padding nodes for every copy would not have this property.

Let $r = 1 - 1/e + \varepsilon$ and $\gamma = 1 - 1/e + \varepsilon/2$. Choose an integer $M > 2C/\varepsilon$, which is polynomial in the input size for fixed ε. In a yes instance the optimum is $C + M|\mathcal{U}|$, so an r-approximation must

Algorithm 3 Threshold search retaining a verified feasible pair

Require: Known $\boldsymbol{\alpha}$, budget R, resolution $\tau > 0$, offline oracle $\mathcal{A}$

$\ell \leftarrow 0$; $U \leftarrow \max_u \sum_{v \in \mathcal{N}(u)} \alpha_v$; $S_U \leftarrow \varnothing$
while $U - \ell > \tau$ **do**
  $W \leftarrow (\ell + U)/2$
  $S \leftarrow \mathcal{A}(q_W(\cdot; \boldsymbol{\alpha}), \boldsymbol{\theta}, R)$
  if $L(S; \boldsymbol{\alpha}) \leq W$ **then**
    $(U, S_U) \leftarrow (W, S)$
  else
    $\ell \leftarrow W$ ▷ Search marker, not an infeasibility claim
return (U, S_U)

return at least $r(C + M|\mathcal{U}|)$. In a no instance every feasible value is at most $C + M\gamma|\mathcal{U}|$. Their separation is

$$r(C + M|\mathcal{U}|) - (C + M\gamma|\mathcal{U}|) = \tfrac{\varepsilon}{2} M|\mathcal{U}| - (1 - r)C > 0.$$

Such an approximation would distinguish the two cases. This proves the claimed tightness for the structured surrogate with general defending requirements, rather than appealing to generic submodular hardness alone.

C Threshold certification and exact comparators

C.1 A search with a one-sided certificate

Let $\tau > 0$ be a desired search resolution and initialize $U = \max_u A_u$, $S_U = \varnothing$, and a search marker $\ell = 0$. The marker is not asserted to be a lower bound on the optimum. At every update, evaluate the original loss directly and retain a pair with $L(S_U; \boldsymbol{\alpha}) \leq U$. One possible heuristic is Algorithm 3.

Every stored upper endpoint is feasible by construction, including initialization. The interval length halves at each query, so the procedure terminates after $O(\log(\max_u A_u/\tau))$ queries when the initial interval exceeds τ. Since oracle failures may skip feasible thresholds, the result is not an approximation to W_{opt} with additive error τ. Nor is monotonicity of the oracle's success pattern assumed. For integer node values, a discrete search can use unit resolution; this does not apply automatically to stochastic mean vectors, which can be noninteger.

If coverage is checked numerically rather than by direct loss, a tolerance must be propagated. Accepting $q_W \geq n - \zeta$ yields only $L \leq W + \delta_{\max}\zeta$. The direct loss test makes the meaning of numerical acceptance explicit.

C.2 MILP for the original loss

The exact comparator for a given known vector solves

$$\begin{aligned} \min_{z, \mathbf{x}} \quad & z \\ \text{subject to} \quad & \sum_{v \in \mathcal{N}(u)} \alpha_v (1 - x_v) \leq z \qquad (u \in V), \\ & \sum_v \theta_v x_v \leq R, \qquad x_v \in \{0, 1\}, \quad z \geq 0. \end{aligned}$$

A feasible incumbent alone is an upper bound; exactness requires an optimality certificate. Time-limited runs should not be interpreted as certified optima without their solver gaps.

For exact surrogate maximization at fixed W, introduce $y_u \in [0, 1]$, maximize $\sum_u y_u$, and impose $\delta_u y_u \leq \sum_{v \in \mathcal{N}(u)} \alpha_v x_v$ for positive residual thresholds, together with $y_u = 1$ for zero thresholds and the same resource constraint. Replacing δ_u by $\bar{\delta}_u$ gives an exact comparator for the online surrogate when evaluated at a known mean vector. The two surrogate comparators and the original-loss MILP solve different optimization problems.

D Additional details for the online guarantee

D.1 Predictable observations and simultaneous confidence

Let $\mathcal{H}_{t-1}$ contain the observations and random choices preceding round t. Conditional on $\mathcal{H}_{t-1}$ and the round's pre-realization actions, the current value vector is a fresh sample. Because S_t and u_t are chosen before that vector is revealed, the decision to observe node v cannot select it based on its current value. Each node's observations admit bounded-sample concentration because observation selection precedes the current draw. The event below concerns every sample count actually reached; it does not condition on that count, which could depend on earlier observed values. Dependence between different nodes within a round does not affect this marginal argument.

For the concentration argument only, extend each node's observed sequence beyond the horizon with fresh draws from its marginal distribution. Nonanticipating selection ensures that this extended sequence is i.i.d.; no such extra samples are supplied to the learner. Let $\widehat{\alpha}_{v,m}$ denote the average of its first m values. For $b_v > 0$ and $m \in \{1, \dots, T\}$, Hoeffding's inequality gives

$$\Pr\left(|\widehat{\alpha}_{v,m} - \alpha_v| > b_v \sqrt{\frac{\beta}{2m}} \right) \le 2e^{-\beta}.$$

A union bound over at most nT node–count pairs, with $\beta = \log(2nT^3)$, yields an event $\mathcal{E}$ of probability at least $1 - T^{-2}$. On this event, whenever $N_v(t-1) > 0$,

$$0 \le U_v(t) - \alpha_v \le b_v \sqrt{\frac{2\beta}{N_v(t-1)}}.$$

Clipping preserves the lower inequality because $b_v \ge \alpha_v$. For an unobserved node, $U_v(t) = b_v$ and $0 \le U_v(t) - \alpha_v \le b_v$.

D.2 Connection to the main-text proof

The proof of Theorem 4.2 combines the confidence event above with coordinatewise optimism and the local sensitivity calculation. Equation (8) then involves only selected nodes. Because $S_t \subseteq O_t$, the counting argument in (9) holds for every attack sequence allowed by the model. Equation (10) completes the pathwise bound, and the failure event contributes at most n/T in expectation. If no node is affordable or all coefficients vanish, the same argument gives a zero estimation term.

D.3 An interpretable graph-dependent bound

If $b_v \le b_{\max}$, $\bar{\delta}_{\min} = \min_{u:\bar{\delta}_u > 0} \bar{\delta}_u$, and

$$H = \max_{v \in V_R} |\{u : v \in \mathcal{N}(u),\ \bar{\delta}_u > 0\}|,$$

then $\|\mathbf{d}\|_2 \le b_{\max} H \sqrt{|V_R|} / \bar{\delta}_{\min}$. Consequently the leading regret term is

$$O\left(\frac{b_{\max} H}{\bar{\delta}_{\min}} \sqrt{|V_R| BT \log(nT)} \right).$$

This expression assumes $V_R \neq \varnothing$ and at least one positive denominator. If $V_R = \varnothing$, the only feasible defense is empty and approximation regret is nonpositive. If all denominators vanish, the surrogate is constant. For undirected closed k-hop neighborhoods, reverse incidence is symmetric and $H \le \max_u |N_k(u)|$. For directed graphs, the appropriate quantity is the number of attack regions containing a node, not its outgoing neighborhood size. Small positive denominators enlarge the bound; they do not alter its dependence on T for a fixed problem instance.

D.4 Approximate implementations and oracle error

Suppose an implementation only ensures

$$\bar{q}_W(S_t; \mathbf{U}(t)) \ge \rho \max_{S \in \mathcal{F}} \bar{q}_W(S; \mathbf{U}(t)) - \xi_t, \qquad \xi_t \ge 0.$$

The proof adds ξ_t to the one-round bound and yields

$$\mathcal{R}_\rho(T) \le (1 + 2\sqrt{2\beta})\|\mathbf{d}\|_2\sqrt{BT} + \frac{n}{T} + \sum_{t=1}^{T} \mathbb{E}[\xi_t].$$

For Full, $\rho = 1 - 1/e$ and $\xi_t = 0$. For the unrestricted use of Fast, a sublinear bound on $\sum_t \mathbb{E}[\xi_t]$ is not established. A small observed loss gap on several instances is not a proof of such a bound. This is the precise distinction between the analyzed learner and its practical accelerated implementation.

D.5 Loss certificate and its interpretation

If $\bar{\delta}_u = 0$, then $\sum_{v \in \mathcal{N}(u)} b_v \le W$, so $L_u(S; \boldsymbol{\alpha}) \le W$. If $\bar{\delta}_u > 0$, then $A_u \le W + \bar{\delta}_u$. For a nonsaturated term,

$$L_u(S; \boldsymbol{\alpha}) = A_u - P_u(S) \le W + \bar{\delta}_u - P_u(S) = W + \bar{\delta}_u(1 - \bar{q}_{u,W}(S; \boldsymbol{\alpha})).$$

For a saturated term, $P_u(S) \ge \bar{\delta}_u$ implies $L_u(S; \boldsymbol{\alpha}) \le W$. Maximizing over u and bounding by the sum of local deficits proves (4). Taking expectations, summing, and substituting Definition 4.1 gives (11) exactly.

Because the chosen target is nonanticipating, the same bound applies to the time average of expected realized losses at the selected targets; multiplying by T gives a cumulative-loss bound. It does not bound the expectation of a hindsight maximum over realized losses by simply replacing random values with their means. An $o(T)$ upper bound implies $[\mathcal{R}_\rho(T)]_+/T \to 0$, where $[x]_+ = \max\{x, 0\}$. This takes the positive part after summing and taking expectations; it does not bound the sum of positive one-round shortfalls. The signed ratio $\mathcal{R}_\rho(T)/T$ need not converge to zero.

E Additional experimental details

E.1 Benchmark topologies and value scales

Table 1 lists the original topology sizes. Rand-S and Rand-L use Erdős–Rényi graphs with edge probability 0.05. Pow-S and Pow-L use the power-law cluster generator with $(n, m, p) = (500, 8, 0.1)$ and $(2000, 8, 0.1)$, respectively. Original static node values and defending requirements are sampled uniformly from the integers 1 through 10. We use the labels Pow-S and Pow-L consistently; labels Power-S and Power-L in earlier records refer to these same datasets.

Our formulation supports either undirected neighborhoods or directed outgoing propagation sets without changing the algorithms. The graph representation must be held fixed across methods within an experiment. The heterogeneous-value comparisons use connected samples of the real-network topologies rather than claiming full-graph runs at the sizes in Table 1.

For heterogeneous values, draw $z_v \sim \mathrm{LogNormal}(0, 1)$ for the heavy-tailed model. For the degree-correlated model, set

$$z_v = (1 + \log(1 + d_v^{\mathrm{graph}}))\epsilon_v, \qquad \log \epsilon_v \sim \mathcal{N}(-\sigma_d^2/2, \sigma_d^2), \qquad \sigma_d = 0.15,$$

where d_v^{graph} denotes graph degree, distinct from the coefficient in (6). In both cases set $\alpha_v = 5.5 z_v / (n^{-1} \sum_i z_i)$, matching the expected node-value scale of the discrete uniform distribution on $\{1, \ldots, 10\}$. The lognormal draw describes heterogeneity in node parameters and should not be confused with the bounded current-round outcome assumption in the regret theorem. Applying that theorem requires a specified bounded observation distribution and known supports for each node.

E.2 Baselines and exactness

The Greedy rule allocates to affordable unprotected nodes in decreasing value order. Greedy-M uses a currently high-loss region and prioritizes a high-value unprotected node within it. BA(ϵ) uses an LP-based bi-criteria defense rule (Bai et al., 2023). The common UCB comparisons change the value estimates used by each allocation rule while retaining the same observation model. Since ϵ in BA controls its bi-criteria rule, it is distinct from the exploration probability of the ϵ-greedy online wrappers.

Table 4: Original running-time record in seconds. Full and Fast are distinguished explicitly; the large-instance implementation is Fast.

	Rand-S	Rand-L	Pow-S	Pow-L	Twitch	LastFM
MILP	1655	3600	1.39	27.77	44.31	1195
BA(ϵ)	0.04	3.59	0.13	0.84	1.16	3.06
Fast Patching-GSA	0.01	0.95	0.11	0.76	2.33	1.71

Table 5: Fixed-normalization online setting on Rand-S ($n = 200$, $T = 1000$, $\eta = 0.4$, $k = 2$). Average loss with reported uncertainty across held-out seeds 5–9.

	PatchLearn	Greedy-M-UCB	Greedy-UCB
Average loss	$\mathbf{28.193 \pm 0.093}$	28.599 ± 0.132	29.138 ± 0.206

For practical deployment, each rule must check affordability before adding a node and stop when no candidate fits. A static MILP result is called exact only when the solver's optimality criterion is met. The large-instance timing record in Table 4 includes a 3600-second Rand-L run; its terminal optimality gap is not provided in the summarized record, so we treat that large-instance MILP curve as a comparator rather than claim a verified optimum on every dataset.

The small-instance runtimes in Table 2 yield speedups ranging from approximately 28 to $68{,}000$ for Fast relative to Full. These ratios reflect the recorded configurations and very small individual timings; they are not a uniform scaling law. Reporting the underlying times is more informative than rounding all cases to a single speedup range.

E.3 Fixed-normalization evaluation

We evaluate the fixed-normalization learner on Rand-S with $n = 200$, $k = 2$, $\eta = 0.4$, and $T = 1000$. Node values lie in $[0, 1]$, and the threshold is fixed before interaction. PatchLearn and the UCB baselines use the same partial-feedback rule. Table 5 reports average defense loss with the recorded uncertainty across held-out seeds 5–9.

PatchLearn achieves an average loss of 28.193, compared with 28.599 for Greedy-M-UCB and 29.138 for Greedy-UCB. The absolute reductions are 0.406 and 0.945, corresponding to relative improvements of 1.42% and 3.24%. The smaller gap to Greedy-M-UCB suggests that using the currently high-loss region already provides a competitive allocation rule in this setting. PatchLearn nevertheless achieves the lowest reported average loss under the common UCB exploration rule.

This experiment provides empirical loss evidence for the fixed-normalization learner. Its values in $[0, 1]$ differ in scale from the heterogeneous-value experiments in Table 3, so the loss magnitudes should be compared within each setting. The evaluation uses a single horizon and measures defense loss rather than surrogate approximation regret. It therefore does not establish the regret rate in Theorem 4.2.

E.4 Threshold sensitivity

Table 6 reports the Rand-S sensitivity experiment. Each row recomputes the defended set at the stated threshold. Here $\widehat{W} = 236$ is the certified threshold returned by the search, not a proof of the globally minimum threshold.

The observed transition establishes feasibility for the returned strategies at and above 236. A lower-coverage result below this value does not establish that no other defense could achieve full coverage. Larger thresholds permit greater loss and need not improve the original objective even when coverage stays at one. Integer values and integer W ensure positive offline residual thresholds are at least one in this sensitivity test. That lower bound does not automatically apply to noninteger online estimates or mean vectors.

Table 6: Threshold sensitivity on Rand-S. Coverage is q_W/n; loss is the original worst-case loss.

Threshold	W	Coverage	Loss
$\widehat{W}-10$	226	0.999727	246
$\widehat{W}-5$	231	0.999893	242
$\widehat{W}-1$	235	0.999991	236
$\widehat{W}$	236	1.000000	236
$\widehat{W}+1$	237	1.000000	237
$\widehat{W}+5$	241	1.000000	241
$\widehat{W}+10$	246	1.000000	246

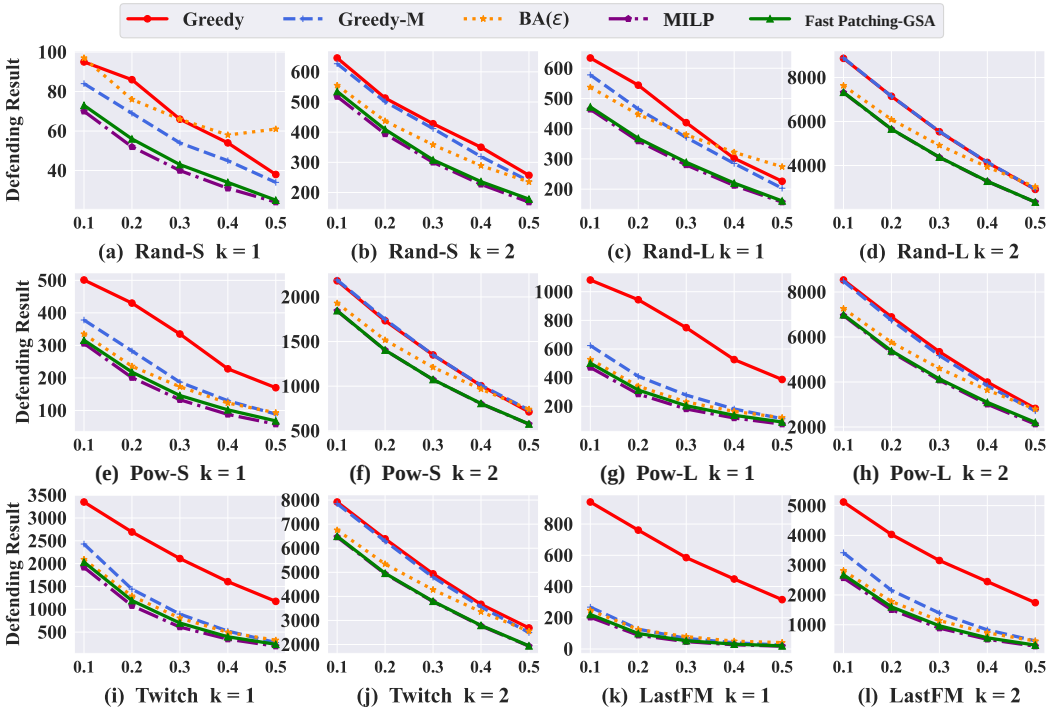

Figure 4: Complete offline defense-loss comparisons for six datasets at propagation radii $k = 1$ and $k = 2$. The horizontal axis is the resource ratio η; "Defending Result" denotes the original worst-case loss, with lower values indicating better defense. The proposed method uses Fast Patching-GSA. The $k = 2$ panels also appear in Figure 2.

E.5 Offline defense across propagation radii

Figure 4 shows how propagation radius affects defense quality over $\eta \in \{0.1, 0.2, 0.3, 0.4, 0.5\}$. At matched resource ratios, the reported losses are larger for $k = 2$ than for $k = 1$ on all six datasets. This is consistent with the larger attack regions exposing more unprotected value. Increasing the budget generally reduces loss, although BA(ϵ) on Rand-S at $k = 1$ increases slightly between $\eta = 0.4$ and 0.5. Thus the empirical outputs of a heuristic need not decrease with every budget increase, even though the optimal loss is nonincreasing in the budget.

Fast Patching-GSA remains close to the MILP comparator across both propagation radii. The contrast between Greedy and Greedy-M depends on graph structure. At $k = 1$, Greedy-M substantially improves on Greedy on Pow-S, Pow-L, Twitch, and LastFM. At $k = 2$, their curves nearly coincide on Rand-L, Pow-S, Pow-L, and Twitch, while Fast retains lower loss. These comparisons show that the benefit of allocation rules depends on propagation as well as resources.