Adaptive Influence Blocking: Minimizing the Negative Spread by Observation-based Policies

Qihao Shi[1,2], Can Wang[1,2,*], Deshi Ye[2], Jiawei Chen[1,2], Yan Feng[1,2], Chun Chen[2]
[1]*ZJU-LianlianPay Joint Research Center*
[2]*College of Computer Science, Zhejiang University*
Hang Zhou, China
{*shiqihao321,wcan,yedeshi,sleepyhunt,fengyan,chenc*}@*zju.edu.cn*

***Abstract*—Spread of negative influence (N-Inf) in a networked system seems to be inevitable, e.g., epidemic spread in human networks, rumors in an online social network and computer virus plaguing the Internet etc. The widespread of N-Inf might cause severe damage and hence the Influence Blocking (IB) problem is attracting ample research interest. The IB problem aims at minimizing the N-Inf spread by immunization, i.e. selecting k (budget size) immunization nodes (Imm-nodes) to prevent the N-Inf from spreading. However, existing works for IB problem are all formulated as a one-shot task: selecting all the k Imm-nodes at the very beginning of N-Inf spread. In real world, unforeseen events might occur and one-shot policies will lack reserved measures to handle these situations. A more reasonable policy is to adaptively invest the budget based on the observation of N-Inf spread along as the time goes by. With the adaptive policy, we can both reserve resources for handling unforeseen events and save unnecessary costs if the spread of N-Inf dies out quickly.**

Motivated by the above considerations, we propose a novel Adaptive Influence Blocking (AIB) problem. Given the intermediate observations of N-Inf spread, the AIB problem aims at selecting Imm-nodes adaptively. We design a k-R (k-nodes-per-Round) policy which selects k Imm-nodes for each round until the budget is exhausted, and an α-T (α-Tolerance) policy which selects a new Imm-node if the expected N-Inf spread exceeds a threshold α. Scalable algorithms with provable approximation guarantees and error bounds are implemented for these policies and significant improvements on time complexity are achieved. Experimental results on real-world datasets demonstrate the effectiveness and scalability of the proposed methods.

***Keywords*-influence blocking; adaptive policy; immunization**

I. Introduction

Controlling the spread of *negative influence* (N-Inf) in a networked system is an interesting research problem [1]–[3], which finds applications in multiple domains such as epidemiology [4], public health [5], cyber security [6] and social media [7]. As the problem of minimizing the N-Inf spread is studied by researchers from different areas, many different names have been proposed for it. For convenience, we term it as *Influence Blocking* (IB) problem in this paper.

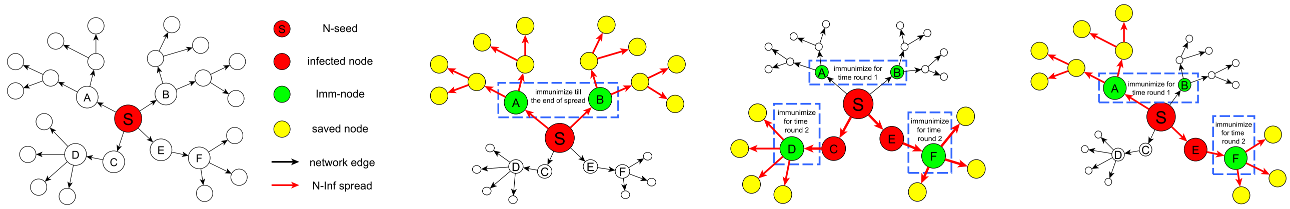
Assuming the seed nodes (N-seeds) that start to spread N-Inf are known, the IB problem aims to select k nodes to immunize. The immunization nodes (Imm-nodes) can be regarded as removed from the network till the end of N-Inf spread. Accordingly, the subgraphs connecting to the Imm-nodes are separated from the N-seeds and become less susceptive. In many applications, immunization is proven to be efficient for bringing an infectious disease under control, stifling a computer virus in its infancy or restricting a malicious rumor within a small circle [8]–[13].

Existing works [8]–[16] for IB problem adopt the one-shot formulation: selecting the k Imm-nodes in the beginning when the N-Inf (or N-seeds) are observed. During the subsequent process of N-Inf spread, these Imm-nodes are kept immunized. In practice, such one-shot policy exhibits two drawbacks. First, the N-Inf spread is highly stochastic and unforeseen events may happen [17], [18]. The one-shot policy leaves no reserved measures for handling emergencies and great damages are likely to occur. Second, if a node is expected to make little reduction on N-Inf spread, keeping it immunized is useless and will incur unnecessary cost. In fact, immunizing nodes are usually temporary measures, e.g., closing public places during a fierce epidemic, shutting down electronic communication channels during a social unrest and immunizing influential users when rumors sweeping through the social networks, etc. These temporary measures generally cannot last for too long a time [19].

Considering the two issues above, a more robust approach is to adaptively select and release Imm-nodes based on the observation of N-Inf spread along with the time advancing. In each time round, the available observation offers the evidence for estimating the future reduction on N-Inf spread of current Imm-nodes. Accordingly, we can release the Imm-nodes with low future reduction, and select new Imm-nodes. This adaptive policy enables a more flexible treatment of the N-inf spread by either reserving budgets for a spread more fierce than expected, or saving unnecessary cost if the N-Inf dies out quickly. We construct an example in Fig.1 to illustrate the benefits of adaptive policies more clearly.

The topology of the example network is shown in Fig.1(a). The infected nodes are colored by red and node S is the N-seed that starts the N-Inf spread at time round 0. Suppose S independently infects nodes A, B, C, D with probability of $\frac{1}{2}$ respectively in time round 1, and all other nodes (including

2375-026X/19/$31.00 ©2019 IEEE
DOI 10.1109/ICDE.2019.00135



(a) Original Network Topology (b) IB Instance (c) AIB Instance (d) AIB Instance

Figure 1. Example of Adaptive Influence Blocking

A,B,C,E) succeed to infect their neighbours in the next time round with probability of 1 once infected. For the IB problem where 2 Imm-nodes are allowed to be selected, the one-shot policy will select A and B as Imm-nodes and keep them immunized till the end of N-Inf spread. The expected number of saved nodes by immunizing A, B is $14 \cdot \frac{1}{2} = 7$, which is the largest possible with budget of 2. As shown in Fig.1(b), if the real spread of N-Inf follow the red bold arrows, 14 nodes (including A, B themselves) will be saved. However, if N-Inf spread as in Fig.1(c), i.e., infecting node C, E, keeping immunizing A, B till the end of spread (3 time rounds) is useless and wasteful. However, the adaptive policy may immunize A, B in time round 1, but release them and immunize D, F in time round 2, based on the observation that the N-Inf infect C, E at time round 1. The effectiveness of the adaptive policy is obvious, i.e., 9 nodes are definitely saved as illustrated in Fig.1(c). If the N-Inf spread follows the case in Fig.1(d), the adaptive policy will immunize A, B in time round 1, but release them and only immunize F in time round 2. Obviously, lower cost is incurred but better performance is achieved. The cases in Fig.1(b), 1(c) and 1(d) happen with the same probability, which demonstrate the necessity of invoking the adaptive policy.

In this paper, we present the first study on Adaptive Influence Blocking (AIB) problem. The AIB problem aims at minimizing the N-Inf spread by adaptively selecting Imm-nodes for current time round based on the observation of previous rounds. It seems untouchable to tackle AIB problem without any constraint. Thus we first introduce the k-R (k-Nodes-per-Round) setting which constrains to select k Imm-nodes for each time round $t \in [T]$, with k, T given in advance and $k \cdot T$ being the total budget. Note T can be viewed as a critical time that the temporary measures are demanded.[1] Under the k-R setting, we design the k-R policy that can approximately solve AIB problem with a $1 - e^{-(1-1/e)} - \varepsilon$ approximation ratio to the optimal policy. Though the k-R policy is effective, it always exhausts all the budget which is not flexible. Therefore, we further design an α-T (α-Tolerance) policy which balance a tradeoff between efficiency and effectiveness. More flexibly, the α-T policy selects Imm-nodes based on the current N-Inf spread results with a tolerance factor α. If the proportion between expected number of infected nodes and the total node number exceed α, α-T policy will select new Imm-nodes. The α-T policy can achieve a $(1-\alpha)(1-e^{-\frac{1}{1-\alpha}})$ approximation ratio which is dependent on the value of α.

For both the k-R policy and α-T policy, we greatly improve the scalability by incorporating the state-of-the-art reverse influence sampling (RIS) approach [20]–[22]. In each case, the RIS method needs to be carefully revised and we show that the proposed scalable algorithms can achieve near-linear running time with respect to the network size, greatly improving their counterpart policies. Based on RIS method, we also design two one-shot policies under k-R constraint, which select all Imm-nodes in the very beginning without any observation. These two one-shot policies exhibit interesting trade-off between efficiency and effectiveness and are used as baselines in our experiments.

To demonstrate the effectiveness of the proposed methods, we conduct experiments on real-world networks. Our experimental results show that the proposed policies are more effective than baselines, and the scalable implementations run in orders of magnitude faster than their counterparts while keeping the performance at the same level. The results also show some interesting findings, such as the α-T policy could save considerable budget with a price of less reduction on N-Inf spread. This may suggest that in practice one may need to consider whether spending the cost of collecting the feedbacks and making decision of investing more budget in each round till the end of N-Inf spread, or spending up the budget in the first few rounds. It opens new directions for further investigations.

To summarize, our major contributions are: (1) We propose a novel AIB problem which more effectively reduce N-Inf spread by adaptively invest the budget based on stepwise observations. (2) We design effective k-R policy and α-T policy for AIB problem and showing their trade-offs. (3) We develop highly scalable algorithms for the two policies. (4) We conduct experiments on real-world networks and demonstrate the effectiveness of the proposed algorithms.

Note that all proofs are included in the appendix.

II. Related Works

There is much literatures on preventing the N-Inf spread in a networked system and manipulating the network topology

[1]Counter measures such as efficient vaccine for epidemic or antivirus software will be developed after a certain critical time, and then blocking influence is not necessary.

is proven to be an effective technique [8]–[13]. A large amount of works try to select the optimal set of nodes/edges to immunize, so as to minimize the N-Inf spread.

Assuming the N-seeds are known, a part of works try to minimize the influence spread by finding the optimal Imm-node set. The work of [24]–[26] adopt similar techniques for selecting the Imm-node set. By constructing a specific tree structure rooted at the infected nodes, they select the nodes that can save the maximum number of descendants in the trees. The work of [8] select the Imm-nodes by a greedy algorithm and prove the approximation guarantees under Linear Threshold model. Wang et al. [19] further consider users' experience with the time period where users are willing to be immunized.

Without the assumption of knowing N-seeds, another part of works study a more global influence blocking strategy. It has been proved that whether an epidemic can survive in a network is deeply intertwined with the spectral radius [6]. Under this premise, sequent studies [12], [13] boil down the problem to the optimization of the spectral radius and propose efficient methods to evaluate the perturbation of the removal of each node/edge on the spectral radius. Other heuristic methods are also explored, such as using "betweenness centrality" [27] or "degree centrality" [16] for measuring the importance of nodes.

Another approach for preventing influence is to start a competing campaign in the network, which is formally defined in [14] as the eventual influence limitation (EIL) problem. The works in [28], [29] propose variants of competitive Linear Threshold model and develop approximation algorithms to solve the EIL problem. More recently, a scalable algorithm is developed in [15] for solving EIL problem which reduces the time complexity to near linear to the network size while retains the approximation guarantees.

III. Preliminaries

In this section, we first introduce the Linear Threshold (LT) Model [30] which describe the N-Inf spread process.

A. Linear Threshold (LT) Model.

Let $G = (V, E)$ be the social network where V, E are the node and edge set. In the LT model [30], each edge $(u, v) \in E$ is associated with a diffusion weight p_{uv} defined by function $p : E \to [0, 1]$. We require that $\sum_u p_{uv} \leq 1, \forall v \in V$. If $(u, v) \notin E$, define $p_{uv} = 0$. Meanwhile, each node v chooses a threshold θ_v uniformly at random from the interval $[0, 1]$, which represents the weighted fraction of v's in-neighbors that must be infected to let v become infected.

Given the random choices of node thresholds, and a N-seed set $S \in V$, the dynamics the LT model proceed in discrete time rounds 0, 1, 2,...., as follows. Initially, at round 0, nodes in S are infected. Then at any time round $t \geq 1$, nodes who have been infected in previous rounds remain infected, and a node v becomes infected if its threshold θ_v is surpassed by the total weights of its currently infected in-neighbors. That is, $\sum_{u:u \text{ is infected}} p_{uv} \geq \theta_v$. The process continues until no more infections are possible.

B. Live-Edge Graph

The LT model can be equivalently described as propagations in *live-edge graphs* [30]. Here we first describe the process of generating a random live-edge graph $L = (V, E_L)$. For each $v \in V$, we randomly pick only one live-edge (u, v) from its incoming edges with the weight p_{uv}, and with probability $1 - \sum_{u \in V} p_{uv}$, no live-edge is selected. All the picked live-edges consist the live-edge set E_L.

Given a Imm-node set B, all nodes in B are kept immunized till the end of N-Inf spread, which can be equivalently viewed as removed from the network. Then for a random generated live-edge graph $L = (V, E_L)$ and a N-seed set S, if there exists a path from any node in S to v through edges in E_L, with no nodes in B lying on the path, then node v becomes infected in live-edge graph L. Let $R_L(S|B)$ denote the set of nodes in L that becomes infected, and $\sigma(S|B)$ be the expected number of infected node that S can infect under LT mode given Imm-node set B. It is proved in [30] that $\sigma(S|B) = \mathbb{E}[|R_L(S|B)|]$, where the expectation is taken over the distribution of live-edge graphs.

The IB problem aims to select an Imm-node set B with at most size k, which can minimize $\sigma(S|B)$, or equivalently maximize the reduction on N-Inf spread $\sigma(S|\emptyset) - \sigma(S|B)$.

IV. Adaptive Influence Blocking (AIB) Problem

Assuming the observation of N-Inf spread is available after each time round, we can select the Imm-nodes in a sequential manner for each round, depending on the current observation and the remaining budget. The AIB problem is to find the optimal policy that accordingly selecting Imm-nodes round by round within budget, so as to maximize the reduction on N-Inf spread.

A. Problem Definition

For AIB problem under LT model, we denote $\phi^F : E \to [0, 1]$ as the function that represents the realization of N-Inf spread, i.e., a random live-edge graph for LT model where 1 represents the edge is live and 0 otherwise. For any N-seed set S and time round t, we are able to observe the status (infected or not) of all nodes that are t-hop neighbours of nodes in S, via live-edges under ϕ^F. We can also observe the corresponding status (live or not) of in-coming edges of these t-hop neighbour nodes. The union of such observations can be represented by a partial realization $\phi_t \subseteq \phi^F$.

We denote (v, t) as the *immunization pair (Imm-pair)* which means immunize node v at time round t. In a live-edge graph, if v becomes infected at time round t, then select Imm-pair (v, t) can prevent v together with all its decedents from being infected in this live-edge graph. Now we define our adaptive policy as $\pi : \phi_{t-1} \to (v, t)$, which is a function from the partial realization, or previous observation, to Imm-pair (v, t), determining which Imm-pair to selected in the next time round given ϕ_{t-1}.

Assume there is a known prior probability distribution $p(\phi^F) := P[\Phi = \phi^F]$ over realizations. Given a realization ϕ^F, we use $\mathcal{B}$ to denote the Imm-pair set selected by policy π under ϕ^F. Note here $\mathcal{B}$ is only constrained by a total budget, denoted as $k \cdot T$, which is written for convenience of presenting the policies below. Then the reduction on N-Inf spread by policy π is $f(\pi) = \sum_{\phi^F} p(\phi^F) \cdot (\sigma(S|\phi^F, \emptyset) - \sigma(S|\phi^F, \mathcal{B}))$, where $\sigma(\cdot)$ is the number of nodes that are infected under corresponding realization.

***Definition** 1 (AIB): The AIB problem aims to find a policy π for maximizing $f(\pi)$: $\pi = \arg\max_\pi f(\pi)$.*

In the following section, we will introduce the k-R policy and α-T policy. Here we first show the adaptive monotonicity and adaptive submodularity which are the theoretical basis of the proposed policies. For convenience, we define $g(\mathcal{B}|\phi^F) = \sigma(S|\phi^F, \emptyset) - \sigma(S|\phi^F, \mathcal{B})$ as the N-Inf reduction function for the realization ϕ^F and the Imm-pair set $\mathcal{B}$.

***Lemma** 1: For any pair (v, t), the N-Inf reduction function $g(\mathcal{B}|\phi^F)$ satisfies (1) adaptive monotonicity: $g(\mathcal{B} \cup \{(v, t)\}|\phi) - g(\mathcal{B}|\phi) \geq 0$ for any partial realization $\phi \subseteq \phi^F$, and (2) adaptive submodularity: $g(\mathcal{B} \cup \{(v, t)\}|\phi) - g(\mathcal{B}|\phi) \geq g(\mathcal{B} \cup \{(v, t)\}|\phi') - g(\mathcal{B}|\phi')$, for any $\phi \subseteq \phi' \subseteq \phi^F$.*

B. k nodes per Round (k-R) Policy: π^k

It is hard to derive a policy without any constraint. So we first consider the k-R setting that selects k Imm-pairs for each time round $t \in [T]$. The budget is divide into T equal-sized parts, given as $k \cdot T$. We then propose the k-R policy in Algorithm 1. We use $f(\mathcal{B}|\phi)$ to denote the expected reduction on N-Inf spread by Imm-pair set $\mathcal{B}$ under the partial realization ϕ. The Imm-pairs in each round are selected greedily with largest marginal gain (Lines 4-7). After k Imm-pairs are selected, we wait one time round and observe the new partial realization (Lines 9-10). The approximation guarantees and time complexity are established in the following theorem.

***Theorem** 1: Let π^* be the optimal policy under the k-R setting. For any $\varepsilon > 0$ and $\ell > 0$, with probability at least $1 - 1/n^\ell$, the k-R policy π^k satisfies:*

$$f(\pi^k) \geq (1 - e^{-(1-1/e)} - \varepsilon) f(\pi^*),$$

if the input $R \geq (8k^2 + 2k\varepsilon) n \log(kTn^{\ell+1})/\varepsilon^2$. The running time for T rounds is $O(k^3 \ell T n^2 m \log(knT)/\epsilon^2)$, where the time of sampling a random live-edge graph is $O(m)$.

Given total budget $k \cdot T$, we can tune the value of k and T to achieve larger reduction on N-Inf spread. Though the k-R policy is effective, it always exhausts all the budget, which is not flexible. Therefore, we introduce α-Tolerance (α-T) Policy which select the Imm-pair set more flexibly.

Algorithm 1: k-R Policy: π^k (G, R, k, T)

1 Initialize $\mathcal{B} = \{B_i\}$ with $B_i = \emptyset, \forall i \in [T]$;
2 Initialize node set $V' = \{v | v$ is reachable from S in $G\}$;
3 **for** $t = 1$ *to* T **do**
4   **while** $|B_t| < k$ **do**
5         $\forall v \in V'$, estimate $f(\mathcal{B} \cup \{(v, t)\}|\phi_{t-1})$ by sampling live edge graphs with nodes in V' for R times;
6         $(v, t) = \arg\max_{v \in V'} f(\mathcal{B} \cup \{(v, t)\}|\phi_{t-1})$;
7         $B_t = B_t \cup (v, t)$;
8   Wait one round and obtain partial realization ϕ_t;
9   **if** $\phi_{t-1} = \phi_t$ **then** break;
10   **else** $\{\phi_{t-1} = \phi_t$; $t = t + 1$; update V' based on ϕ_t;$\}$
11 **return** $\mathcal{B}$

C. α-Tolerance (α-T) Policy: π^α

The α-T policy is shown in Algorithm 2. Let H_0 be the set of nodes that N-seed set S can reach in G. The α-T policy use α as a tolerance threshold. If $\sigma(S|\phi, \mathcal{B}) > \alpha|H_0|$, or equivalently $f(\mathcal{B}|\phi) < (1 - \alpha)|H_0|$ in the current time round, we select new Imm-pairs (Lines 6-10). Otherwise, we wait for a time round and obtain the new partial realization (Lines 11-14). Note the input $K = k \cdot T$ for consistency.

***Theorem** 2: Let π^* be the optimal policy. The expected number of savable nodes of π^α is bounded by $f(\pi^\alpha) \geq (1 - \alpha)(1 - e^{-\frac{1}{1-\alpha}}) f(\pi^*)$.*

We can see when $\alpha = 0$, the approximation of α-T policy becomes $1 - e^{-1}$. In this case, the budget are exhausted in the very beginning, and all Imm-pairs are selected for time round 1, since the condition in line 6 always holds. In contrast, when $\alpha = 1$, no Imm-pair will be selected. As the name implies, the tolerance threshold α indicates the intensity of the N-Inf spread that we can tolerate. Empirically tuning this parameter is practical for real-world scenarios, as we would not exhaust all the budget but invest the part that can reduce the N-Inf spread down to our tolerance threshold.

V. Scalable Implementations

In this section, we speed up the proposed policies by developing scalable implementations. Here we first introduce a useful tool of Reverse Reachable (RR) set.

For the LT model, an RR set R_v rooted at node $v \in V$ is the set of nodes that can reach v in a random generated live edge graph. A random RR set R is an RR set rooted at a node picked uniformly at random from V. Note for LT model, a random RR set is a simple path, since each node in a random generated live edge graph has at most one incoming edge. Given a N-seed set S and a set $\mathcal{R}$ of random generated RR sets, it is proved that $\sigma(S|\emptyset) = n \cdot \mathbb{E}[F_{\mathcal{R}}(S)]$, where n is total node number and $F_{\mathcal{R}}(S)$ is the fraction that $S \cap R \neq \emptyset, \forall R \in \mathcal{R}$. Thus the expected N-Inf spread of S can be estimated by generating enough number of RR sets. The RR sets have already been utilized for solving IB

Algorithm 2: α-T Policy: π^{α} (G,R,K)

Initialize $\mathcal{B} = \{B_i\}$ with $B_i = \emptyset, \forall i \in [T]$;
Initialize node set $V' = \{v | v$ is reachable from S in $G\}$;
Initialize current time round $t = 1$;
$H_0 = V'$;
while $K > 0$ **do**
  if $f(\mathcal{B}|\phi_{t-1}) \leq (1-\alpha)|H_0|$ **then**
    $\forall v \in V'$, estimate $f(\mathcal{B} \cup \{(v,t)\}|\phi_{t-1})$ by sampling live edge graphs for nodes in V' for R times;
    $(v,t) = \arg\max_{v \in V'} f(\mathcal{B} \cup \{(v,t)\}|\phi_{t-1})$;
    $\mathcal{B} = \mathcal{B} \cup \{(v,t)\}$;
    $K = K - 1$;
  else
    Wait one round and obtain partial realization ϕ_t;
    if $\phi_{t-1} = \phi_t$ **then** break;
    else $\{\phi_{t-1} = \phi_t$; $t = t+1$; update V' based on ϕ_t;$\}$
return $\mathcal{B}$

problem [15]. Here we revise it carefully for solving AIB problem under LT model.

A. Reverse Immunization Imm-Pair (RIP) Set

We define the *Reverse Immunization Pair (RIP) Set* M_v for node v, which is a set of Imm-pairs generated for corresponding RR set R_v. We first explain how we generate an RIP set and then present the relation between RIP set, RR set and the LT propagation process.

Under LT model, during the generation process of R_v, whenever we visit a new node u that would be added into R_v, we can simultaneously record the distance between u and v as d_{uv}. The generation of R_v ends when meeting a N-seed or no new node can be visited. If a N-seed s is met, for all visited node u, we add $(u, d_{sv} - d_{uv})$ into M_v. If no N-seed is met, $M_v = \emptyset$. See an example in Fig.2 where the RR set R_v for node v is $\{a, b, c, d\}$. The corresponding RIP set M_v is illustrated in Fig.2. If we use any Imm-pair in M_v, the node v can be definitely saved in this live-edge graph if the N-seed a start to spread N-Inf at time round 0.

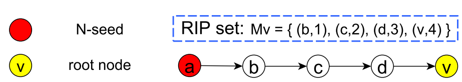


Figure 2. An RIP Set Example for LT Model.

Formally, given the Imm-pair set $\mathcal{B}$, N-seed set S, a random RR set R_v generated for node v and its corresponding RIP set M_v, we say M_v is covered, or equivalently R_v is *saved*, if (1) $S \cap R \neq \emptyset$, and (2) $\mathcal{B} \cup M_v \neq \emptyset$. Lemma 2 shows the connection between RIP set and the LT propagation.

***Lemma** 2: For any N-seed set S, Imm-pair set $\mathcal{B}$ and node v, the probability that v can be saved in a random LT spread process is equal to the probability that a random generated RIP set M_v is covered.*

Suppose we generate a set $\mathcal{M}$ of RIP sets. Let $F_{\mathcal{M}}(\mathcal{B})$ be the fraction of RIP sets in $\mathcal{M}$ that are covered by Imm-pairs in $\mathcal{B}$. Based on Lemma 2, we have $f(\mathcal{B}) = n \cdot \mathbb{E}[F_{\mathcal{M}}(\mathcal{B})]$, where n is the number of nodes. This implies that we can accurately estimate $f(\mathcal{B})$ by generating a large enough set $\mathcal{M}$. The optimal Imm-pair set can be found by seeking the optimal set of Imm-pairs that make the most number of RIP sets covered, which is a max-cover problem. Similar techniques are adopted for the influence maximization problem [20]–[23]. These works all have the same structure and we base our work on the IMM algorithm [22].

B. Scalable Implementation for k-R Policy

We base the implementation of k-R policy on the IMM algorithm, and name it as k-R-IMM which is shown in Algorithm 3. The idea of k-R-IMM is to run IMM (Lines 10-27) in each round t which selects k Imm-pairs for consisting B_t. The main differences include four points. First, we generate RIP pair set $\mathcal{M}$ instead of the set of RR sets in IMM, and the root of each RIP set is selected from V' (Lines 12-16 and 22-26). Second, the ImmSelection$(\mathcal{R}, k, t)$ in Lines 15 and 23 constructs B_t by searching the space $\{(v,t)|v \in V'\}$ with current time round t and greedily adding the Imm-pair (v,t) with largest marginal gain of $F_{\mathcal{M}}(\mathcal{B} \cup \{v,t\})$. Third, the new generated RIP sets may have been covered by the Imm-pairs selected for previous rounds. When we generate an RIP set for round t, we should check that if it is already covered. If so, this RIP set is invalid for round t and we need to withdraw it and generate a new RIP set (Lines 15-16 and 25-26). Fourth, we need to adjust the parameters (Lines 4-8) used in the algorithm carefully to guarantee the approximation stated in Theorem 3.

***Theorem** 3: For any $\varepsilon > 0$ and $\ell > 0$, with probability at least $1 - 1/n^{\ell}$, the k-R-IMM algorithm still ensure a $1 - e^{-(1-1/e)} - \varepsilon$ approximation ratio to the optimal policy. Meanwhile, the total running time for T rounds is $O(T(k+\ell)(n+m)\log(nT)/\epsilon^2)$.*

C. Scalable Implementation of α-T Policy

The implementation of α-T policy requires to estimate $f(\mathcal{B}|\phi)$ accurately in each time round based on the current N-Inf spread results. By using RIP sets, we can implement such estimation with provable error bound. We denote $K = k \cdot T$ as the total budget and $F_{\mathcal{M}}(\mathcal{B}|\phi)$ as the fraction of RIP sets in $\mathcal{M}$ that are covered by Imm-pairs in $\mathcal{B}$ where $\mathcal{M}$ is the set of RIP sets generated[2] based on the realization ϕ.

***Lemma** 3: Given ϕ, for any $\varepsilon > 0$ and $\ell > 0$, if we generate a set $\mathcal{M}$ of RIP sets with $|\mathcal{M}| \geq (2+\varepsilon)|H_0| \cdot \frac{\log(2Kn^{\ell+1})}{\varepsilon^2}$, then for any non-empty Imm-pair set $\mathcal{B}$, with at least $1 - 1/(Kn^{\ell+1})$ probability, we have $\big||H_0| \cdot F_{\mathcal{M}}(\mathcal{B}|\phi) - f(\mathcal{B}|\phi)\big| < \varepsilon f(\mathcal{B}|\phi)$.*

Based on the above lemma, we can construct a sizable set of RIP sets for each time round to estimate $f(\mathcal{B}|\phi)$.

[2]Given realization ϕ, the nodes infected at last time round are new N-seeds to start N-Inf spread. Then RIP sets are generated accordingly.

Algorithm 3: k-R-IMM($G, S, k, T, \varepsilon, \ell$)

1 Initialize $\mathcal{B} = \{B_i\}$ with $B_i = \emptyset, \forall i \in [T]$;
2 Initialize node set $V' = \{v|v$ is reachable from S in $G\}$;
3 *// Lines 4-8 set the parameters for ensuring Theorem 3*;
4 $\ell = \ell + \ln(2T)/\ln n$; $\mathcal{M} = \emptyset$; $LB = 1$;
5 $\varepsilon_0 = e^{1-1/e}\varepsilon/2$; $\varepsilon' = \sqrt{2}\varepsilon_0$;
6 $\alpha = \sqrt{\ell \ln n + \ln 2 + \ln T}$; $\beta = \sqrt{(1-1/e)(\ln\binom{n}{k}) + \alpha^2}$;
7 $\lambda' = (2 + 2\varepsilon'/3)(\ln\binom{n}{k} + \ell \ln n + \ln T + \ln \log_2 n)n/\varepsilon'^2$;
8 $\lambda^* = 2n((1-1/e)\alpha + \beta)^2/\varepsilon_0^2$;
9 **for** $t = 1$ *to* T **do**
10 **for** $i = 1$ *to* $\log_2(n-1)$ **do**
11 $x = n/2^i$;$\theta_i = \lambda'/x_i$;
12 **while** $|\mathcal{M}| < \theta_i$ **do**
13 Select a root node u from V' uniformly at random;
14 Generate a valid RIP set for u;
15 **if** M *is valid* **then** Insert it into $\mathcal{M}$;
16 **else** continue;
17 B_t=ImmSelection($\mathcal{M}, V', k, t$);
18 **if** $nF_{\mathcal{R}}(\mathcal{B} \cup B_t) \geq (1+\varepsilon')x$ **then**
19 $LB = nF_{\mathcal{R}}(\mathcal{B} \cup B_t)/(1+\varepsilon')$;
20 break;
21 $\theta = \lambda^*/LB$;
22 **while** $|\mathcal{M}| < \theta$ **do**
23 Select a root node u from V' uniformly at random;
24 Generate a RIP M set for u;
25 **if** M *is valid* **then** Insert it into $\mathcal{M}$;
26 **else** continue;
27 B_t=ImmSelection($\mathcal{M}, V', k, t$);
28 *// Lines 10-27 are the main body of IMM algorithm, using the adjusted parameterizations in Lines 4-8*;
29 $\mathcal{B} = \mathcal{B} \cup B_t$;
30 Wait one round and obtain partial realization ϕ_t;
31 **if** $\phi_{t-1} = \phi_t$ **then** break;
32 **else** $\{\phi_{t-1} = \phi_t$; $t = t+1$; update V' based on ϕ_t;$\}$
33 Return $\mathcal{B}$
34 **Function** ImmSelection($\mathcal{M}, V', k, t$)
35 Initialize a set $B_t = \emptyset$;
36 **while** $|B_t| < k$ **do**
37 $(v,t) = \arg\max_{v \in V'} F_{\mathcal{M}}(\mathcal{B} \cup \{(v,t)\})$;
38 $B_t = B_t \cup (v,t)$;
39 **return** B_t

Then we can decide whether to select another Imm-pair. The algorithm is shown in Algorithm 4 and named as α-T-RIP (α-T policy based on RIP set). A main difference between Algorithm 4 and k-R-IMM is that in each time round, when we are generating RIP sets, we do not need to check whether it is valid, since $F_{\mathcal{M}}(\cdot)$ is estimated with the Imm-pair set $\mathcal{B}$ consisting of all previous selected Imm-pairs. We establish the accumulative error bound in Theorem 4.

***Theorem** 4: With probability of at least $1 - 1/n^{\ell}$, the expected number of savable nodes of π^{α} is bounded by*

$$f(\pi^{\alpha}) \geq \frac{1-\alpha}{1+\varepsilon}(1 - e^{-\frac{1-\varepsilon}{1-\alpha}})f(\pi^*) - \frac{2\varepsilon}{1+\varepsilon}|H_0|K.$$

Algorithm 4: α-T-RIP(G, S, K, ε)

Initialize $\mathcal{B} = \{B_i\}$ with $B_i = \emptyset, \forall i \in [T]$;
Initialize node set $V' = \{v|v$ is reachable from S in $G\}$;
while $k \geq 0$ **do**
Generate a set $\mathcal{M}$ following Lemma 3, with root nodes selected from V' uniformly at random;
if $F_{\mathcal{M}}(\mathcal{B}|\phi_{t-1}) \leq (1-\alpha)|H_0|$ *and* $K > 0$ **then**
$v = \arg\max_{u \in V} F_{\mathcal{M}}(\mathcal{B} \cup \{(u,t)\}|\phi_{t-1})$;
$\mathcal{B} = \mathcal{B} \cup \{(v,t)\}$;
$K = K - 1$;
else
Wait one round and obtain partial realization ϕ_t;
if $\phi_{t-1} = \phi_t$ **then** break;
else $\{\phi_{t-1} = \phi_t$; $t = t+1$; update V' based on ϕ_t;$\}$
return B

D. One-Shot Policies

Sometimes, collecting the feedbacks of N-Inf spread is not easy since the N-Inf spread usually occurs accidently and spread fast. Then we have to deal with the AIB problem under one-shot setting, i.e., selecting the Imm-pair set at the very beginning without any observation. To this end, in this section we propose two one-shot policies, which exhibit interesting trade-off between efficiency and scalability.

The two one-shot policies also select k Imm-pairs for each time round $t \in [T]$. We denote $h(\mathcal{B}) = \sigma(S|\emptyset) - \sigma(S|\mathcal{B})$ as the reduction function for the selected Imm-pair set $\mathcal{B}$. Then the objective function is

$$\mathcal{B} = \underset{\mathcal{B}:|B_t| \leq k, \forall t \in [T]}{\arg\max} \quad h(\mathcal{B}).$$

Define the candidate space as $\mathcal{V} = \{\mathcal{V}_1, ..., \mathcal{V}_T\}$ where $\mathcal{V}_t = \{(v,t)|v \in V'\}, \forall t \in [T]$. Similarly, we show the monotonicity and submodularity which are the theoretical basis of the proposed one-shot policies.

***Lemma** 4: The function $h(\mathcal{B})$ satisfies (1) monotonicity: for any $\mathcal{B} \subseteq \mathcal{B}' \subseteq \mathcal{V}$, $h(\mathcal{B}) \leq h(\mathcal{B}')$; and (2) submodularity: for any $\mathcal{B} \subseteq \mathcal{B}' \subseteq \mathcal{V}$ and any pair $(v,t) \in \mathcal{V} \backslash \mathcal{B}'$, $h(\mathcal{B} \cup \{(v,t)\}) - h(\mathcal{B}) \geq f(\mathcal{B}' \cup \{(v,t)\}) - f(\mathcal{B}')$.*

1) One-Shot Round-Polling (OS-RP) Algorithm: The idea of OS-RP policy is similar to k-R policy, i.e., select the Imm-pair round by round. For each time round t, we search the space $\mathcal{V}_t$. Only after we select k Imm-pairs for the current round, we go on to the selection for the next round. Utilizing the RIP set, the whole process of OS-RP policy is completely the same to k-R-IMM, only with one difference: V' is not changed for all the T selection iterations since we do not observe the N-Inf spread results. That is to say, lines 30-32 in Algorithm 3 are not needed. The resulting algorithm would have the same approximation guarantee and time complexity with k-R-IMM, as shown in Theorem 5.

***Theorem** 5: Let $\mathcal{B}^*$ be the optimal solution for maximizing $h(\cdot)$ under k-R setting. For any $\varepsilon > 0$ and $\ell > 0$, with* *probability at least $1-1/n^{\ell}$, the output $\mathcal{B}$ of OS-RP satisfies:*

$$h(\mathcal{B}) \geq (1 - e^{-(1-1/e)} - \varepsilon) h(\mathcal{B}^*),$$

The total running time is $O(T(k+\ell)(n+m)\log(nT)/\epsilon^2)$.

2) One-Shot Individual Polling (OS-IP) policy: The idea of OS-IP policy is that at each iteration, we searches (v, t) in the whole space of $\mathcal{V}$ and picks the one having the largest marginal gain on reduction of N-Inf spread, without replacement. If the budget for some round t exhausts, then $\mathcal{V}_t$ is removed from $\mathcal{V}$. By utilizing RIP sets and by similar modification based on IMM, we implement OS-IP and present it in Algorithm 5. Compared to Algorithm 3, Algorithm 5 has two major difference. First, the parameterizations in Algorithm 5 (Lines 4-7) are slightly different from that of Algorithm 3, which is made to ensure the approximation guarantees stated in Theorem 6 below. Second, the function ImmSelection$(\mathcal{M}, V', k, t)$ (in Lines 17 and 27 of Algorithm 3) is replaced by the function OS-IP-Selection$(\mathcal{M}, V', k, T)$, shown in Algorithm 5. OS-IP-Selection needs to search (v, t) from the whole space $\mathcal{V}$ while ImmSelection only search space $\mathcal{V}_t$. Such cross round searching incurs at least a factor of T spending on the running time of OS-IP than that of OS-RP. However, the space $\mathcal{V}$ with the constraint of k-R setting is a partitioned matroid, which indicates that selecting the Imm-pair set $\mathcal{B}$ to cover the maximum number of RIP sets in $\mathcal{M}$ is an instance of submodular maximization under partition matroid. Thus the approximation ratio can be improved to $\frac{1}{2} - \varepsilon$, larger than $1 - e^{-(1-1/e)} - \varepsilon \approx 0.46 - \varepsilon$. We conclude the theoretical guarantees along with time complexity in the following theorem.

***Theorem** 6: Let $\mathcal{B}^*$ be the optimal solution for maximizing $h(\cdot)$ under k-R setting. For any $\epsilon > 0$ and $\ell > 0$, with probability at least $1 - \frac{1}{n^{\ell}}$, the output $\mathcal{B}$ of OS-IP satisfies:*

$$f(\mathcal{B}) \geq (\frac{1}{2} - \varepsilon) f(\mathcal{B}^*).$$

The total running time is $O(T^2(k+\ell)(n+m)\log(n)/\epsilon^2)$.

Though OS-RP algorithm has a slightly lower approximation ratio, it gives a factor of T saving on the time complexity. It is because in each iteration, OS-IP needs to search a space T times larger than that of OS-RP. This shows the trade-off between efficiency and approximation ratio, that OS-IP has a better performance guarantee while OS-RP is more efficient.

Algorithm 5: OS-IP$(G, T, k, \varepsilon, \ell)$

1 Initialize $\mathcal{B} = \{B_i\}$ with $B_i = \emptyset, \forall i \in [T]$;
2 Initialize node set $V' = \{v | v$ is reachable from S in $G\}$;
3 *// Lines 4-7 set the parameters for ensuring Theorem 3*;
4 $\ell = \ell + \ln 2/\ln n$; $\mathcal{M} = \emptyset$; $LB = 1$; $\varepsilon' = \sqrt{2}\varepsilon$;
5 $\alpha = \sqrt{\ell \ln n + \ln 2}$; $\beta = \sqrt{1/2(T\ln\binom{n}{k}) + \alpha^2}$;
6 $\lambda' = (2 + 2\varepsilon'/3)(T\ln\binom{n}{k} + \ell \ln n + \ln \log_2 n) n/\varepsilon'^2$;
7 $\lambda^* = 2nT(\alpha/2 + \beta)^2/\varepsilon^2$;
8 Do the same to Lines 10-29 in Algorithm 3, with the function ImmSelection$(\mathcal{M}, V', k, t)$ in Lines 17 and 27 replaced by the function OS-IP-Selection$(\mathcal{M}, V', k, T)$ shown below;
9 Return $\mathcal{B}$
10 **Function** OS-IP-Selection$(\mathcal{M}, V', k, T)$
11 Initialize $\mathcal{B} = \{B_i\}$ with $B_i = \emptyset, \forall i \in [T]$;
12 **while** *true* **do**
13 $\quad (v, t) = \arg\max_{(v,t):v\in\mathcal{V}} F_{\mathcal{M}}(\mathcal{B} \cup \{(v, t)\})$,
14 $\quad\quad$ subject to: $|B_t| < k, \forall t \in [T]$;
15 $\quad$ **If** $(v, t) = null$ **then** break;
16 $\quad$ **Else** $\mathcal{B} = \mathcal{B} \cup (v, t)$;
17 Return $\mathcal{B}$

VI. Experiments

A. Experimental Settings

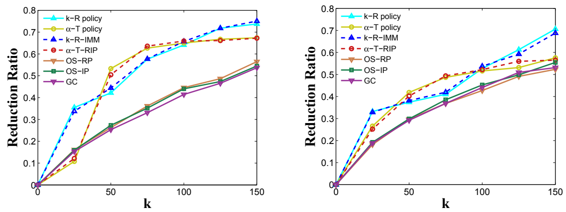
1) Datasets: We use two public available[3] real-world networks, NetHEPT and NetPHY, in our experiments. They are both collaboration networks extracted from the e-print arXiv (http://www.arXiv.org). The NetHEPT is extracted from the High Energy Physics - Theory section (form 1991 to 2003), and the NetPHY is extracted from Physics section. The nodes in both networks are authors and an edge between two nodes means the two authors coauthored at least one paper. We clean the dataset by removing the duplicated edges. Then the network of NetHEPT has 15223 nodes and 31387 edges, and the network of NetPHY has 37154 nodes and 174161 egdes. The edges in both NetHEPT and NetPHY are undirected. In the experiments, we change each undirected edge to a bi-directed edge.

2) Edge probabilities: Given a network $G(V, E)$, we set the probabilities on each edge according to the LT model as follows: for a given node $v \in V$, we draw a probability value $\widetilde{p}_{uv}$ for each edge $e = (u, v) \in E$ that is incoming into v, uniformly at random from the interval $[0, 1]$. In addition, we draw from the same interval a probability value $\overline{p}(v)$ representing no infection, i.e the probability that v's infected parents fail to infect it. Since the probabilities on the edges plus the probability of no infection must sum to 1, we then normalize each probability over the sum of all the probabilities, i.e., we obtain $p_{uv} = \widetilde{p}_{uv}/(\sum_{u\in V} \widetilde{p}_{uv} + \overline{p}(v))$.

3) Comparison algorithms.: For AIB problem, we propose k-R policy and α-T policy and their scalable implementations. Meanwhile, we design two one-shot policies OS-RP and OS-IP as baselines. We compare the 6 proposed algorithms with a non-adaptive IB method. All the algorithms are listed below, with $k \cdot T$ being the total budget.

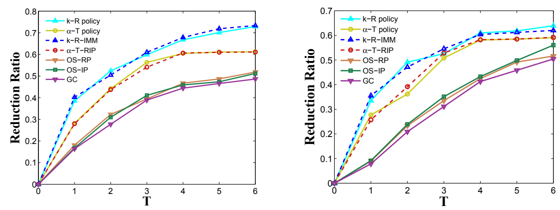
1) **k-R policy.** It is Algorithm 1, which selects k Imm-nodes for each round based on the current observation.
2) **α-T policy.** It is Algorithm 2, which selects new Imm-node if the expected spread of N-Inf exceeds a threshold α based on the current observation.
3) **k-R-IMM.** It is Algorithm 3, which is the scalable implementation of k-R policy.

[3]http://research.microsoft.com/en-us/people/weic/projects.aspx



(a) NetHEPT($\alpha = 0.25$) (b) NetPHY($\alpha = 0.55$)

Figure 3. The Results of Reduction Ratio with T=5.



(a) NetHEPT($\alpha = 0.25$) (b) NetPHY($\alpha = 0.55$)

Figure 4. The Results of Reduction Ratio with k=125.

4) **α-T-RIP.** It is Algorithm 4, which is the scalable implementation of α-T policy.
5) **OS-RP.** It is Algorithm 5, which selects k Imm-nodes round by round at the very beginning.
6) **OS-IP.** It is Algorithm 6, which selects k Imm-nodes cross rounds at the very beginning.
7) **GreedyCutting(GC).** It is a classic algorithm for edge blocking [8] under LT model. It estimates a score for each edge which represents the reduction on N-Inf spread by removing this edge. We can simply estimate the reduction on N-Inf spread of each node by summing up the out-going edges. We then immunize the top-k scored nodes till the end of N-Inf spread.

4) Parameters and Measurements.: For k-R policy and α-T policy and GreedyCutting, we sampling 1000 live-edge graphs to estimate the N-Inf spread in each time round. For k-R-IMM, α-T-RIP, OS-RP and OS-IP, we set $\varepsilon = 0.5$ and $l = 1$ by default. For each dataset, we use IMM algorithm to select the top-200 influential nodes and uniformly randomly choose 30 of them as the N-seeds to start the N-Inf spread. We measure the performance by the influence spread reduction ratio as

$$\text{reduction ratio} = \frac{\sigma(S|\emptyset) - \sigma(S|\mathcal{B})}{\sigma(S|\emptyset)}.$$

We construct 5000 live-edge graphs and evaluate the reduction ratio of all methods by averaging on these live-edge graphs. We run the experiments on a Linux server with 24 Core Intel E5 CPU and 256GB RAM.

B. Experimental Results

1) Reduction Ratio and Budget Cost: We first conduct experiments on the two dataset by fixing $T = 5$ and varying k from 25-150. We set $\alpha = 0.25$ for NetHEPT dataset and 0.55 for NetPHY dataset respectively. The results of reduction on N-Inf spread are shown in Fig.3.

Table I
THE BUDGET USED IN DIFFERENT METHODS

NetHEPT (T=5)	C1	C2	NetHEPT (k=125)
k=25,50,75	$k \cdot T$	$k \cdot T$	T=1,2,3
k=100,125,150	403 ± 3		T=4,5,6

NetPHY (T=5)	C1	C2	NetPHY (k=125)
k=25,50	$k \cdot T$	$k \cdot T$	T=1,2
k=75,100,125,150	320 ± 6		T=3,4,5,6

C1 denotes the α-T policy and α-T-RIP method.
C2 denotes the other comparisons.

In Fig.3, we can see the k-R policy and α-T policy outperform the three baselines in both the two datasets. Meanwhile, the k-R-IMM and α-T-RIP achieve comparable performance to their counter-parts. These all demonstrate the effectiveness of the proposed methods.

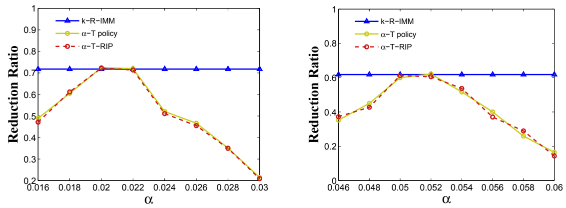
An interesting phenomenon is that in both Fig.3(a) and Fig.3(b), the two α-T methods outperform the two k-R methods when $50 \leq k \leq 100$. However, when $k < 50$ and $k > 100$, the two α-T methods are less effective. Specifically in Fig.3(a), we can see when $k < 25$, the two α-T methods even exhibit worse performance than the three baselines. It is due to the setting of tolerance threshold α. For a fixed α, a small budget may be exhausted very soon since the expected N-Inf spread can easily exceed threshold α. In contrast, a large budget will not bring great improvement for α-T methods since if the N-Inf is reduced down to the threshold by the current Imm-pair set $\mathcal{B}$, it would not easily exceed the threshold again, and thus no more budget will be invested. This makes the lower reduction ratio for α-T methods when k is very small or very large.

We also conduct experiments by fixing $k = 125$ and varying T from 1 to 6. The results of reduction on N-Inf spread are shown in Fig.4. Similarly, we can see the proposed k-R methods and α-T methods outperform the three baselines in both the two datasets. However, in both Fig.4(a) and Fig.4(b), the α-T methods perform completely worse than the two k-R methods. This is because, under our setting, the two α-T methods tends to spend less budget, with a price of worse performance. To show such trade-off, we list the budget used for all the methods in Table I. In Table I, we can see the two α-T methods indeed save considerable budget. This suggests that in practice one may need to consider whether spending more budget for further preventing the N-Inf spread, or saving the budget with an acceptable price of less reduction on N-Inf spread. It may open new directions for further investigations.

Note also the two one-shot policies OS-RP and OS-IP slightly outperform the GC method in all the cases. This indicates that if no observation is available, these two policies also bring considerable improvement on effectiveness.

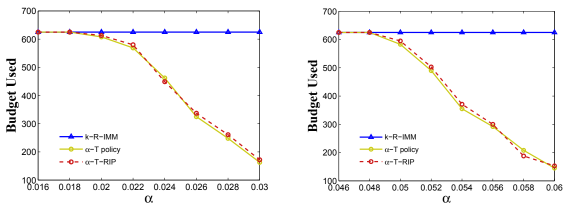
For further exploring the impact of the threshold α, we conduct experiments by varying α, as discussed below.

2) Impact of α: We fix $T = 5$, $k = 125$ and vary the tolerance threshold α to see its impact, shown in Fig.5.



(a) NetHEPT (b) NetPHY

Figure 5. The Results of Reduction Ratio Varying α with T=5 and k=125.



(a) NetHEPT (b) NetPHY

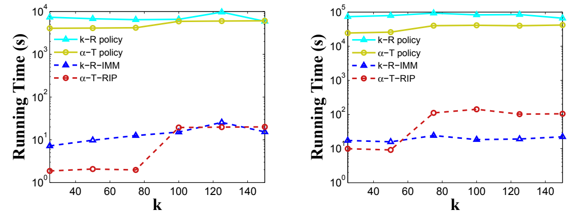
Figure 6. The Results of Budget Used Varying α with T=5 and k=125.

In Fig.5(a), we can see the two α-T methods both perform comparably with k-R-IMM only $\alpha \in [0.02, 0.22]$ for NetHEPT and $\alpha \in [0.05, 0.52]$ for NetPHY. When α becomes larger or smaller, the reduction ratio falls quickly. For the case that α is small, the expected N-Inf spread may easily exceeds α, which means we will exhaust the budget in the first few rounds. This is not effective since all the immunization are done within a short term which cannot significantly reduce the N-Inf spread. For the case that α is large, the expected N-Inf spread cannot easily exceed α, which means by investing a few budget (or no budget), we can make the N-Inf spread below the tolerance threshold α. Such compromise on N-Inf spread reduction brings high efficiency on saving the budget, as present in Fig.6.

In Fig.6, we can see only with small α, the α-T methods exhaust all the budget. With α being large, the budget used decrease very fast. Specifically, when $\alpha \in [0.2, 0.22]$ in Fig.6(a) and $\alpha \in [0.5, 0.52]$ in Fig.6(b), the two α-T methods both save considerable budget with the reduction ratio comparable with that of k-R-IMM method. This indicates that when we are adopting α-T policy, an appropriate α may bring effective reduction on N-Inf spread both with low cost.

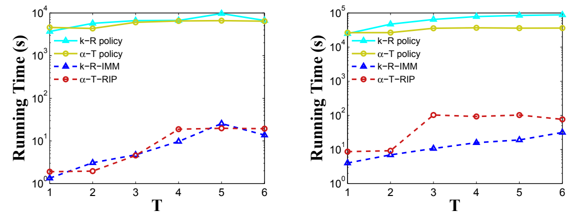
3) Scalability: The theoretical analyses in this paper show that the proposed implementations of the two policies achieve significant improvement on time complexity. In this section, we show that our experimental results confirm such improvement. Fig.7, Fig.8 and Fig.9 are the running time results of the same sets of experiments in Fig.3, Fig.4 and Fig.5 respectively. We can see in all the cases, the implementations k-R-IMM and α-T-RIP run faster than their counter-parts with at least up to three orders of magnitude. Such high scalability makes the proposed methods more practical in real-world scenarios.

An interesting phenomenon is that the two α-T-RIP methods exhibit a step change in all the cases, with k, T and



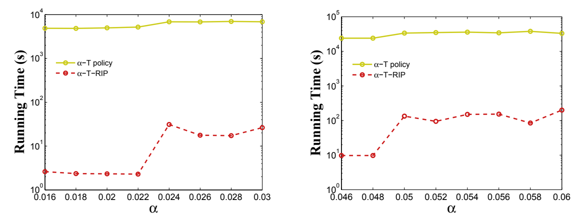
(a) NetHEPT (b) NetPHY

Figure 7. The Running Time varying k with T=5.



(a) NetHEPT (b) NetPHY

Figure 8. The Running Time varying T with k=125.



(a) NetHEPT (b) NetPHY

Figure 9. The Running Time varying α with T=5 and k=125.

α being larger in Fig.7, Fig.8 and Fig.9 respectively. The rational is that, the N-Inf usually spread fiercely in the first few time rounds, and has a long tail that it spread for many time rounds but infect a few nodes in each round. Thus, after we invest some budget to make the expected N-Inf spread below the threshold α, it cannot easily exceed the threshold α again, in general. However, with budget not exhausted, we still need to check in each time round whether the expected N-Inf may exceed the threshold α. Still due to the long tail of N-Inf spread, the accumulated time incurred of such check becomes large. It causes a consideration that whether we should keep collecting the feedbacks for decisions of investing more budget in each time round until the N-Inf spread ends, or using up the budget in the first few rounds. It leaves an interesting future work of finding a more effective allocation of the budget.

VII. Conclusion and Future Work

Motivated by the realistic demand of adaptive policies for influence blocking, in this paper we present the first study of Adaptive Influence Blocking (AIB) problem. Given the observations of N-Inf spread results in each time round, the AIB problem aims at selecting Imm-nodes adaptively. We design a k-R policy and an α-T policy together with scalable implementations. The experimental results demonstrate that the proposed policies are more effective than baselines, and the scalable implementations run in orders of magnitude faster than their counterparts while keeping the performance at the same level. An interesting trade-off among reduction on N-Inf spread, the incurred budget cost and the running time are observed. It may open new directions of investigations such as finding more effective adaptive policy. Besides, finding adaptive policies are interesting research directions for network applications with dynamic social relation or node attributes [35]–[37].

ACKNOWLEDGEMENT

This work is supported by the National Natural Science Foundation of China (Grant No: U1866602, 11671355) and by a Discovery Grant from the National Science and Engineering Research Council of Canada. It is also partially supported by ByteDance.

REFERENCES

[1] Y.-Y. Liu, J.-J. Slotine, and A.-L. Barabási, "Controllability of complex networks," *Nature*, vol. 473, no. 7346, p. 167, 2011.

[2] T. W. Valente, "Network interventions," *Science*, vol. 337, no. 6090, pp. 49–53, 2012.

[3] S. Eubank, H. Guclu, V. S. Kumar, M. V. Marathe, A. Srinivasan, Z. Toroczkai, and N. Wang, "Modelling disease outbreaks in realistic urban social networks," *Nature*, vol. 429, no. 6988, pp. 180–184, 2004.

[4] H. W. Hethcote, "The mathematics of infectious diseases," *SIAM review*, vol. 42, no. 4, pp. 599–653, 2000.

[5] J. Medlock and A. P. Galvani, "Optimizing influenza vaccine distribution," *Science*, vol. 325, no. 5948, pp. 1705–1708, 2009.

[6] D. Chakrabarti, Y. Wang, C. Wang, J. Leskovec, and C. Faloutsos, "Epidemic thresholds in real networks," *ACM TISSEC*, vol. 10, no. 4, p. 1, 2008.

[7] J. Leskovec, L. Backstrom, and J. Kleinberg, "Meme-tracking and the dynamics of the news cycle," in *Proceedings of the 15th ACM SIGKDD*. ACM, 2009, pp. 497–506.

[8] E. B. Khalil, B. Dilkina, and L. Song, "Scalable diffusion-aware optimization of network topology," in *Proceedings of the 20th ACM SIGKDD*. ACM, 2014, pp. 1226–1235.

[9] M. Kimura, K. Saito, and H. Motoda, "Minimizing the spread of contamination by blocking links in a network." in *AAAI*, vol. 8, 2008, pp. 1175–1180.

[10] C. J. Kuhlman, G. Tuli, S. Swarup, M. V. Marathe, and S. Ravi, "Blocking simple and complex contagion by edge removal," in *ICDM,*. IEEE, 2013, pp. 399–408.

[11] F. Morone and H. A. Makse, "Influence maximization in complex networks through optimal percolation," *Nature*, vol. 524, no. 7563, p. 65, 2015.

[12] H. Tong, B. A. Prakash, T. Eliassi-Rad, M. Faloutsos, and C. Faloutsos, "Gelling, and melting, large graphs by edge manipulation," in *Proceedings of the 21st ACM CIKM*. ACM, 2012, pp. 245–254.

[13] S. Saha, A. Adiga, B. A. Prakash, and A. K. S. Vullikanti, "Approximation algorithms for reducing the spectral radius to control epidemic spread," in *Proceedings of the 2015 SDM*. SIAM, 2015, pp. 568–576.

[14] C. Budak, D. Agrawal, and A. El Abbadi, "Limiting the spread of misinformation in social networks," in *Proceedings of the 20th WWW*. ACM, 2011, pp. 665–674.

[15] G. A. Tong, W. Wu, L. Guo, D. Li, C. Liu, B. Liu, and D. Z. Du, "An efficient randomized algorithm for rumor blocking in online social networks," in *INFOCOM, IEEE*, 2017, pp. 1–9.

[16] C. Gao, J. Liu, and N. Zhong, "Network immunization and virus propagation in email networks: experimental evaluation and analysis," *KIS*, vol. 27, no. 2, pp. 253–279, 2011.

[17] W. Chen, T. Lin, Z. Tan, M. Zhao, and X. Zhou, "Robust influence maximization," in *ACM SIGKDD*, 2016, pp. 795–804.

[18] N. Ohsaka and Y. Yoshida, "Portfolio optimization for influence spread," in *International Conference on World Wide Web*, 2017, pp. 977–985.

[19] B. Wang, G. Chen, L. Fu, L. Song, and X. Wang, "Drimux: Dynamic rumor influence minimization with user experience in social networks," *IEEE TKDE*, vol. 29, no. 10, pp. 2168–2181, 2017.

[20] C. Borgs, M. Brautbar, J. Chayes, and B. Lucier, "Maximizing social influence in nearly optimal time," in *Proceedings of the Twenty-Fifth Annual ACM-SIAM Symposium on Discrete Algorithms*. SIAM, 2014, pp. 946–957.

[21] Y. Tang, X. Xiao, and Y. Shi, "Influence maximization: Near-optimal time complexity meets practical efficiency," in *Proceedings of the 2014 SIGMOD*. ACM, 2014, pp. 75–86.

[22] Y. Tang, Y. Shi, and X. Xiao, "Influence maximization in near-linear time: a martingale approach," in *Proceedings of the 2015 SIGMOD*. ACM, 2015, pp. 1539–1554.

[23] Q. Shi, C. Wang, J. Chen, Y. Feng, and C. Chen, "Location driven influence maximization: Online spread via offline deployment," *KBS*, vol. 166, pp. 30–41, 2019.

[24] C. Song, W. Hsu, and M. L. Lee, "Node immunization over infectious period," in *Proceedings of the 24th CIKM*. ACM, 2015, pp. 831–840.

[25] Y. Zhang and B. A. Prakash, "Data-aware vaccine allocation over large networks," *ACM TKDD*, vol. 10, no. 2, p. 20, 2015.

[26] ——, "Scalable vaccine distribution in large graphs given uncertain data," in *Proceedings of the 23rd CIKM*. ACM, 2014, pp. 1719–1728.

[27] C. M. Schneider, T. Mihaljev, S. Havlin, and H. J. Herrmann, "Suppressing epidemics with a limited amount of immunization units," *Physical Review E*, vol. 84, no. 6, p. 061911, 2011.

[28] X. He, G. Song, W. Chen, and Q. Jiang, "Influence blocking maximization in social networks under the competitive linear threshold model," in *Proceedings of the 2012 SDM*. SIAM, 2012, pp. 463–474.

[29] A. Borodin, Y. Filmus, and J. Oren, "Threshold models for competitive influence in social networks," in *International Workshop on Internet and Network Economics*. Springer, 2010, pp. 539–550.

[30] D. Kempe, J. Kleinberg, and É. Tardos, "Maximizing the spread of influence through a social network," in *Proceedings of the ninth ACM SIGKDD*. ACM, 2003, pp. 137–146.

[31] D. Golovin and A. Krause, "Adaptive submodularity: Theory and applications in active learning and stochastic optimization," *Journal of AI Research*, vol. 42, pp. 427–486, 2011.

[32] G. L. Nemhauser, L. A. Wolsey, and M. L. Fisher, "An analysis of approximations for maximizing submodular set functionsłi," *Mathematical Programming*, vol. 14, no. 1, pp. 265–294, 1978.

[33] P. R. Goundan and A. S. Schulz, "Revisiting the greedy approach to submodular set function maximization," *Optimization online*, pp. 1–25, 2007.

[34] M. L. Fisher, G. L. Nemhauser, and L. A. Wolsey, *An analysis of approximations for maximizing submodular set functionsłII*. Springer Berlin Heidelberg, 1978.

[35] C. Jiawei, F. Yan, E. Martin, Z. Sheng, C. Chun, and C. Wang, "Modeling users' exposure with social knowledge influence and consumption influence for recommendation," in *Proceedings of the 27th CIKM*. ACM, 2018, pp. 953–962.

[36] J. Chen, C. Wang, M. Ester, Q. Shi, Y. Feng, and C. Chen, "Social recommendation with missing not at random data," in *ICDM*. IEEE, 2018, pp. 29–38.

[37] S. Zhou, H. Yang, X. Wang, J. Bu, M. Ester, P. Yu, J. Zhang, and C. Wang, "Prre: Personalized relation ranking embedding for attributed networks," in *Proceedings of the 27th CIKM*. ACM, 2018, pp. 823–832.

Appendix
Proofs of Lemmas and Theorems

A. Proof of Lemma 1

Proof: Note a realization can be viewed as a random live edge graph. Then the adaptive monotonicity trivially follows since adding a new Imm-pair at any time will never lead the infected nodes increase in this realization, which gives $g(\mathcal{B} \cup \{(v,t)\}|\phi) - g(\mathcal{B}|\phi) \geq 0$ hold for any partial realization $\phi \subseteq \phi^F$. For the submodularity, consider a certain node u which will be infected under realization ϕ^F with no Imm-pair selected. Note we have $\phi \subseteq \phi' \subseteq \phi^F$. Suppose u can be saved by $\mathcal{B} \cup \{(v,t)\}$ but not by $\mathcal{B}$ under partial realization ϕ'. This implies that u is saved by $\mathcal{B}\cup\{(v,t)\}$ but not by $\mathcal{B}$ under partial realization ϕ too, since the adaptive monotonicity of $g(\cdot)$. In addition, if u is saved by $\mathcal{B}\cup\{(v,t)\}$ but not by $\mathcal{B}$ under partial realization ϕ, it may not be saved by $\mathcal{B}\cup\{(v,t)\}$ under partial realization ϕ' since its infection may be inevitable, i.e., it may already be infected or the live edge path from a source to u can no longer be blocked by $\mathcal{B} \cup \{(v,t)\}$. The above two statements together give $g(\mathcal{B} \cup \{(v,t)\}|\phi) - g(\mathcal{B}|\phi) \geq g(\mathcal{B} \cup \{(v,t)\}|\phi') - g(\mathcal{B}|\phi')$ which proves the adaptive submodularity. ∎

B. Proof of Theorem 1

Proof: When $R = (8k^2 + 2k\varepsilon)n \log(kTn^{\ell+1})/\varepsilon^2$, we can bound the performance of the selection in each time round by the factor ε. Let $\mathcal{B} = \{(v,t)\}$ be a random Imm-pair set with fixed t, and $p = \mathbb{E}[f(\mathcal{B})]/n$ and $\delta = \varepsilon f(\mathcal{B}^*)/2knp$. Then by Chernoff bounds, we have

$$\begin{aligned}
&\Pr\left[\left|\hat{f}(\mathcal{B}) - f(\mathcal{B})\right| \geq \frac{\varepsilon}{2k} f(\mathcal{B}^*)\right] \\
&= \Pr\left[\left|R\hat{f}(\mathcal{B})/n - Rp\right| \geq R\delta p\right] \\
&< 2\exp\left(-\frac{\delta^2 pR}{2+\delta}\right) = 2\exp\left(-\frac{\varepsilon^2 f(\mathcal{B}^*)R}{8k^2 n + 2k\varepsilon n}\right) \\
&\leq 2\exp\left(-\frac{\varepsilon^2 f(\mathcal{B}^*)}{(8k^2T^2 + 2kT\varepsilon)n}\frac{R}{f(\mathcal{B}^*)}\right) \leq \frac{1}{n^\ell nkT}.
\end{aligned}$$

Since each iteration of the greedy selection inspects at most n sets and the number of iteration executed is k. So the algorithm inspects at most kn sets. By the union bound, we can see with probability of $1 - 1/(n^\ell T)$, each set inspected by the greedy selection satisfies: $\left|\hat{f}(\mathcal{B}) - f(\mathcal{B})\right| < \frac{\varepsilon}{2k} f(\mathcal{B}^*)$ where $\hat{f}(\mathcal{B})$ is the estimated value of $f(\mathcal{B})$. Note in each iteration, we select the node with largest marginal gain on reduction on N-Inf spread, which needs to estimate the increase of $f(\cdot)$ by adding a new Imm-pair. Thus the error incurred by each selection is at most $\frac{\varepsilon}{k} f(\mathcal{B}^*)$ and the total error for k selections is $\varepsilon f(\mathcal{B}^*)$. Therefore, the selected Imm-pair set in each round satisfies $f(\mathcal{B}) \geq (1 - 1/e - \varepsilon) f(\mathcal{B}^*)$.

From the general result of Theorem 5.2 in [31] and the adaptive monotonicity and adaptive submodularity proved in Lemma 1, we know that if in each round we find an α approximation of the optimal solution of this round, then the greedy adaptive policy is an $1 - e^{-\alpha}$ approximation of the optimal greedy policy π^*. Replace ε by ε_0 and by union bound, we have the final output $\mathcal{B}$ of k-R policy satisfies $f(\mathcal{B}) \geq (1 - e^{-(1-1/e-\varepsilon_0)}) f(\mathcal{B}^*)$ with probability of at least $1 - 1/n^\ell$. Let $\varepsilon_0 = e^{1-1/e}\varepsilon/2$, it is easy to verify that $1 - e^{-(1-1/e-\varepsilon_0)} \geq 1 - e^{-(1-1/e)} - \varepsilon$ in this case. Finally, the total running time of T rounds is $O(TknRm) = O(k^3 \ell T n^2 m \log(knT)/\varepsilon^2)$. ∎

C. Proof of Theorem 2

Proof: Assume two partial realizations ϕ and ϕ' satisfying $\phi \subseteq \phi'$. Let $\mathcal{B}$ and $\mathcal{B}'$ be Imm-pair sets that selected under partial realization ϕ and ϕ' respectively, satisfying $\mathcal{B} \subseteq \mathcal{B}'$. Based on Lemma 1, we have (1) $g(\mathcal{B}\cup\{(v,t)\}|\phi) - g(\mathcal{B}|\phi) \geq g(\mathcal{B}\cup\{(v,t)\}|\phi') - g(\mathcal{B}|\phi')$, and (2) $g(\mathcal{B}\cup\{(v,t)\}|\phi) - g(\mathcal{B}|\phi) \geq g(\mathcal{B}'\cup\{(v,t)\}|\phi) - f(\mathcal{B}'|\phi)$ hold for any $\mathcal{B} \subseteq \mathcal{B}' \subseteq \mathcal{V}$ and any pair (v,t) with $v \in V'$.

Let π^μ_t be the level-t-truncation of policy π^μ obtained by running until it terminates or until time round t. We assume that the i^{th} selection is executed at time round t_i by selecting $(v_i, t_i{+}1)$ and ϕ_t is the partial realization at time round t. Let $\mathcal{B}_i$ be the Imm-pair set after the i^{th} selection, H_i be the node set in which the nodes' infection probabilities are reduced by $\mathcal{B}_i$. We use $f_H(\pi|\phi)$ to denote the expected reduction on N-Inf spread of π under realization ϕ. Then we have

$$\begin{aligned}
&f(\pi^\alpha_{t_{i+1}}|\phi_{t_i}) - f(\pi^\alpha_{t_i}|\phi_{t_i}) \\
&= \mathbb{E}[g(\mathcal{B}_i \cup \{(v_{i+1}, t_i + 1)\}|\phi_{t_i}) - g(\mathcal{B}_i|\phi_{t_i})] \\
&\geq \max\nolimits_v \mathbb{E}[g_{H_0\setminus H_i}(\mathcal{B}_i \cup \{(v, t_i + 1)\}|\phi_{t_i}) - g_{H_0\setminus H_i}(\mathcal{B}_i|\phi_{t_i})] \\
&\geq \frac{1}{K}(f_{H_0\setminus H_i}(\pi^*|\phi_{t_i}) - f_{H_0\setminus H_i}(\pi^\alpha_{t_i}|\phi_{t_i})),
\end{aligned}$$

The first inequality follows from that π^α selects the Imm-pair that maximizes the entire marginal gain, and the second inequality follows from the properties of submodular maximization. Let $\Delta_i = f(\pi^*) - f(\pi^\alpha_i)/(1-\alpha)$. We have

$$\begin{aligned}
(1-\alpha)(\Delta_i - \Delta_{i+1}) &= f(\pi^\alpha_{t_{i+1}}) - f(\pi^\alpha_{t_i}) \\
&= \mathbb{E}[f(\pi^\alpha_{t_{i+1}}|\phi_{t_i}) - f(\pi^\alpha_{t_i}|\phi_{t_i})] \\
&\geq \mathbb{E}[\frac{1}{K}(f_{H_0\setminus H_i}(\pi^*|\phi_{t_i}) - f_{H_0\setminus H_i}(\pi^\alpha_{t_i}|\phi_{t_i}))],
\end{aligned}$$

$$\begin{aligned}
&f(\pi^*|\phi_{t_i}) - f(\pi^\alpha|\phi_{t_i})/(1-\alpha) \\
&\quad = (f_{H_0\setminus H_i}(\pi^*|\phi_{t_i}) - f_{H_0\setminus H_i}(\pi^\alpha|\phi_{t_i})/(1-\alpha)) \\
&\quad\quad + (f_{H_i}(\pi^*|\phi_{t_i}) - f_{H_i}(\pi^\alpha|\phi_{t_i})/(1-\alpha)) \\
&\quad \leq (f_{H_0\setminus H_i}(\pi^*;\phi_{t_i}) - f_{H_0\setminus H_i}(\pi^\alpha|\phi_{t_i})) \\
&\quad\quad + (f_{H_i}(\pi^*|\phi_{t_i}) - f_{H_i}(\pi^\alpha|\phi_{t_i})/(1-\alpha))
\end{aligned}$$

According to Algorithm 2, we update the time round when the condition is satisfied, which means $f(\mathcal{B}|\phi) \geq (1-\alpha)|H_0|$. Thus $f_{H_i}(\pi^*|\phi_{t_i}) - f_{H_i}(\pi^\alpha|\phi_{t_i})/(1-\alpha) \leq 0$, and then we have $f(\pi^*|\phi_{t_i}) - f(\pi^\alpha|\phi_{t_i})/(1-\alpha) \leq f_{H_0\setminus H_i}(\pi^*;\phi) - f_{H_0\setminus H_i}(\pi^\alpha|\phi)$. Thus $(1-\alpha)(\Delta_i - \Delta_{i+1}) \geq \mathbb{E}[\frac{1}{K}(f(\pi_i^*|\phi_{t_i}) - f(\pi_i^\alpha|\phi_{t_i})/(1-\alpha)] = \frac{1}{k}\Delta_i$.

It follows that $\Delta_{i+1} \leq (1 - \frac{1}{(1-\alpha)K})\Delta_i$. Hence we have $\Delta_K \leq (1-\frac{1}{(1-\alpha)K})\Delta_0 \leq e^{-\frac{1}{1-\alpha}}\Delta_0$, and thus $f(\pi^*) - f(\pi^\alpha_{t_K})/(1-\alpha) \leq e^{-\frac{1}{1-\alpha}}\Delta_0 = e^{-\frac{1}{1-\alpha}} f(\pi^*)$. Hence $f(\pi^\alpha) \geq (1-\alpha)(1-e^{-\frac{1}{1-\alpha}})f(\pi^*)$. ■

D. Proof of Lemma 2

Proof: Suppose we generate an RR set R_v and a corresponding RIP set M_v for v on a live edge graph g constructed from G. Let ρ_1 be the probability that M_v is covered, and ρ_2 be the probability that v can be saved in a random LT process. Then, ρ_1 equals the probability that after generating a random live-edge graph, the following events happen: (1) there exists a live-edge path that can reach v from a node s in S, (2) on every such path, there lies a node u that the distance between u and s is t, and (3) $(u,t) \in M_v$. Meanwhile, ρ_2 equals the probability that from any node $s \in S$ that can reach v through a live-edge path, there lies a node u which is a t-hop neighbour of s in the live-edge graph and $(u,t) \in M_v$. It follows that $\rho_1 = \rho_2$. ■

E. Proof of Theorem 3

Proof: The process of k-R-IMM in each time round is essentially the same as IMM with parameters ε_0 and ℓ'. Thus, following the result in [22], we know that for each round, with probability at least $1 - 2/n^{\ell'} = 1 - 1/(n^\ell T)$, the constructed $\mathcal{B}$ is a $(1-1/e-\varepsilon_0)$ approximation of the optimal Imm-pair set for this round. Then following the similar arguments as in Theorem 1, we know that across all T rounds, with probability at least $1 - 1/n\ell$, $f(\pi^k) \geq (1 - e^{-(1-1/e-\varepsilon_0)})f(\pi^*) \geq (1 - e^{-(1-1/e)} - \varepsilon)f(\pi^*)$. ■

F. Proof of Lemma 3

Proof: Let p be the probability that a random RIP set M is covered by $\mathcal{B}$. Then we have $p = \mathbb{E}[F_\mathcal{M}(\mathcal{B}|\phi)/\theta] = f(\mathcal{B}|\phi)/|H_0|$. Let $\delta = \varepsilon f(\mathcal{B}|\phi)/|H_0|p$. Then by Chernoff bounds, we have

$$\begin{aligned}
&\Pr\left[\left||H_0|F_\mathcal{M}(\mathcal{B}|\phi) - f(\mathcal{B}|\phi)\right| \geq \varepsilon f(\mathcal{B}|\phi)\right] \\
&= \Pr\left[\left|\theta \cdot F_\mathcal{M}(\mathcal{B}|\phi) - p\theta\right| \geq \frac{\varepsilon f(\mathcal{B}|\phi)}{|H_0|p}\theta p\right] \\
&< 2\exp\left(-\frac{\delta^2 p\theta}{2+\delta}\right) \leq 2\exp\left(-\frac{\varepsilon^2 f^2(\mathcal{B}|\phi)\theta}{2|H_0|^2 p + \varepsilon f(\mathcal{B}|\phi)|H_0|}\right) \\
&\leq 2\exp\left(-\frac{\varepsilon^2 f(\mathcal{B}|\phi)}{(2+\varepsilon)|H_0|}\theta\right) \leq \frac{1}{N}.
\end{aligned}$$

The final inequality holds since $f(\mathcal{B}|\phi) \geq 1$. Therefore, the claim follows. ■

G. Proof of Theorem 4

Proof: Similar to the proof of Theorem 2, we use $F_\mathcal{M}(\pi|\phi)$ to denote the number of RIP sets in $\mathcal{M}$ that covered by the Imm-pair set selected by π under realization ϕ. We first prove the error bound of each selection of new Imm-pair. By Lemma 3, we have

$$\begin{aligned}
&f(\pi^\alpha_{r+1}|\phi) - f(\pi^\alpha_r|\phi) + f(\pi^\alpha_r|\phi) \\
&\geq \frac{\hat{f}(\pi^\alpha_{r+1}|\phi) - F_\mathcal{M}(\pi^\alpha_r|\phi) + F_\mathcal{M}(\pi^\alpha_r|\phi)}{1+\varepsilon} \\
&= |H_0|\frac{\max_z[F_\mathcal{M}(\mathcal{B}\cup\{z\}|\phi) - F_\mathcal{M}(\mathcal{B}|\phi)] + F_\mathcal{M}(\mathcal{B}|\phi)}{1+\varepsilon} \\
&\geq |H_0|\frac{F_\mathcal{M}(\mathcal{B}\cup\{u^*\}|\phi) - F_\mathcal{M}(\mathcal{B}|\phi) + F_\mathcal{M}(\mathcal{B}|\phi)}{1+\varepsilon} \\
&\geq \frac{1-\varepsilon}{1+\varepsilon}(f(\mathcal{B}\cup\{u^*\}|\phi) - f(\mathcal{B}|\phi) + f(\mathcal{B}|\phi))
\end{aligned}$$

Accordingly, we have

$$\begin{aligned}
&f(\pi^\alpha_{r+1}|\phi) - f(\pi^\alpha_r|\phi) \\
&\geq \frac{1-\varepsilon}{1+\varepsilon}(f(\mathcal{B}\cup\{u^*\}|\phi) - f(\mathcal{B}|\phi)) - \frac{2\varepsilon}{1+\varepsilon}|H_0| \\
&\geq \frac{1-\varepsilon}{1+\varepsilon}\frac{1}{K}(f_{H_0\setminus H_i}(\pi^*|\phi) - f_{H_0\setminus H_i}(\pi^\alpha_r|\phi)) - \frac{2\varepsilon}{1+\varepsilon}|H_0|
\end{aligned}$$

Then let $\Delta_i = f(\pi^*) - f(\pi^\alpha_i)\cdot\frac{1+\varepsilon}{1-\alpha}$. We have $\frac{1-\alpha}{1+\varepsilon}(\Delta_i - \Delta_{i+1}) = f(\pi^\alpha_{i+1}) - f(\pi^\alpha_i) \geq \mathbb{E}[\frac{1-\varepsilon}{1+\varepsilon}\frac{1}{K}(f_{H_0\setminus H_i}(\pi^*|\phi) - f_{H_0\setminus H_i}(\pi^\alpha_r|\phi)) - \frac{2\varepsilon}{1+\varepsilon}|H_0|]$. According to Algorithm 4, we update the time round when the condition is un-satisfied, which means $f(\mathcal{B}|\phi) \geq \frac{1-\alpha}{1+\varepsilon}|H_0|$. Thus $f_{H_i}(\pi^*|\phi_r) - f_{H_i}(\pi^\alpha|\phi_r)\cdot\frac{1+\varepsilon}{1-\alpha} \leq 0$, and $f(\pi^*|\phi_r) - f(\pi^\alpha|\phi_r)\cdot\frac{1+\varepsilon}{1-\alpha} \leq f_{H_0\setminus H_i}(\pi^*|\phi) - f_{H_0\setminus H_i}(\pi^\alpha|\phi)$. Thus $\frac{1-\alpha}{1+\varepsilon}(\Delta_i - \Delta_{i+1}) \geq \mathbb{E}[\frac{1}{K}(f(\pi^*_i|\phi_i) - f(\pi^\alpha_i|\phi_i)\cdot\frac{1+\varepsilon}{1-\alpha} - \frac{2\varepsilon}{1+\varepsilon}|H_0|] = \frac{1-\varepsilon}{1+\varepsilon}\frac{1}{K}\Delta_i - \frac{2\varepsilon}{1+\varepsilon}|H_0|$.

Thus we have $\Delta_{i+1} \leq (1 - \frac{1-\varepsilon}{1-\alpha}\frac{1}{k})\Delta_i + \frac{2\varepsilon}{1-\alpha}|H_0| \implies \Delta_K \leq (1-\frac{1-\varepsilon}{1-\alpha}\frac{1}{K})\Delta_0 + \frac{2\varepsilon}{1-\alpha}|H_0|k \leq e^{-\frac{1-\varepsilon}{1-\alpha}}\Delta_0 + \frac{2\varepsilon}{1-\alpha}|H_0|K \implies f(\pi^*) - f(\pi^\alpha_{r_K})\cdot\frac{1+\varepsilon}{1-\alpha} \leq e^{-\frac{1-\varepsilon}{1-\alpha}}f(\pi^*) + \frac{2\varepsilon}{1-\alpha}|H_0|K \implies f(\pi^\alpha) \geq \frac{1-\alpha}{1+\varepsilon}(1-e^{-\frac{1-\varepsilon}{1-\alpha}})f(\pi^*) - \frac{2\varepsilon}{1+\varepsilon}|H_0|K$. ■

H. Proof of Lemma 4

Proof: Consider a certain node u which can be saved by $\mathcal{B}'\cup\{(v,t)\}$ but not by $\mathcal{B}'$. This implies (1) u is not saved by $\mathcal{B}$ either, and (2) the live path to u must be blocked by (v,t). Hence, u is saved by $\mathcal{B}\cup\{(v,t)\}$ but not by $\mathcal{B}$, which gives the submodularity. Note a nonnegative linear combination of monotone and submodular functions is also monotone and submodular. We can derive $h(\mathcal{B}\cup\{(v,t)\}) - h(\mathcal{B}) \geq h(\mathcal{B}'\cup\{(v,t)\}) - h(\mathcal{B}')$. ■

I. Proof of Theorem 5 and Theorem 6

We omit the detailed proofs here since Theorem 5 is a direct application of the double submodular property established in [32], [33], and Theorem 6 is a direct application of the monotone and submodular partition matroid optimization [34], respectively.