Who and When to Screen: Multi-Round Active Screening for Network Recurrent Infectious Diseases Under Uncertainty

Han-Ching Ou
Harvard University
Cambridge, Massachusetts
hou@g.harvard.edu

Arunesh Sinha
Singapore Management University
Singapore
aruneshs@smu.edu.sg

Sze-Chuan Suen
University of Southern California
Los Angeles, California
ssuen@usc.edu

Andrew Perrault
Harvard University
Cambridge, Massachusetts
aperrault@g.harvard.edu

Alpan Raval
Wadhwani AI
Mumbai, India
alpan@wadhwaniai.org

Milind Tambe
Harvard University
Cambridge, Massachusetts
milind_tambe@harvard.edu

ABSTRACT

Controlling recurrent infectious diseases is a vital yet complicated problem in global health. During the long period of time from patients becoming infected to finally seeking treatment, their close contacts are exposed and vulnerable to the disease they carry. Active screening (or case finding) methods seek to actively discover undiagnosed cases by screening contacts of known infected people to reduce the spread of the disease. Existing practice of active screening methods often screen all contacts of an infected person, requiring a large budget. In cooperation with a research institute in India, we develop a model of the active screening problem and present a software agent, REMEDY. This agent assists maximizing effectiveness of active screening under real world budgetary constraints and limited contact information. Our contributions are: (1) A new approach to modeling multi-round network-based screening/contact tracing under uncertainty and proof of its NP-hardness; (2) Two novel algorithms, Full- and Fast-REMEDY. Full-REMEDY considers the effect of future actions and provides high solution quality, whereas Fast-REMEDY scales linearly in the size of the network; (3) Evaluation of Full- and Fast-REMEDY on several real-world datasets which emulate human contact to show that they control diseases better than the baselines. We also show that the software agent is robust to errors in estimates of disease parameters, and incomplete information of the contact network. Our software agent is currently under review before deployment as a means to improve the efficiency of district-wise active screening for tuberculosis in India.

KEYWORDS

Network theory (graphs), Network SIS model, Public healthcare

ACM Reference Format:
Han-Ching Ou, Arunesh Sinha, Sze-Chuan Suen, Andrew Perrault, Alpan Raval, and Milind Tambe . 2020. Who and When to Screen: Multi-Round Active Screening for Network Recurrent Infectious Diseases Under Uncertainty. In *Proc. of the 19th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2020), Auckland, New Zealand, May 9–13, 2020,* IFAAMAS, 9 pages.

1 INTRODUCTION

Contagious diseases, such as tuberculosis, influenza and sexually transmitted diseases (STDs) (e.g., gonorrhea and chlamydia) are critical public-health challenges that continue to threaten lives and impose significant economic burden on society. For example, the economic loss due to influenza in the USA alone is estimated to be $11.2 billion in 2015 [30]. While low-cost treatment programs are available, individuals ignore symptoms and delay care, increasing transmission risk. As a result, health agencies engage in active screening or contact tracing efforts as figure 1 shows, where individuals are asked to undergo diagnostic tests and offered treatment if tests are positive [7, 14]. However, active screening is expensive in developing countries. Even in the USA, Braxton et al. [6] state that "In 2012, 52% of state and local STD programs experienced budget cuts. This amounts to reductions in clinic hours, contact tracing, and screening for common STDs." In India, an estimated 1 million missing tuberculosis (TB) cases require an efficient method of active screening, particularly given limited health budgets [10]. Efficiently identifying and intervening for infectious cases is therefore of vital importance.

(a) Passive screening

(b) Active screening

Figure 1: Passive screening only treats patients who come to clinic voluntarily whereas active screening can treat patients in hard-to-reach tribal areas at a higher cost [40]

There is a huge body of literature on spread and control of recurrent diseases (no permanent immunity) [3, 16, 37, 42, 47]. However, these prior studies assume perfect observation of who is infected and who is not. Also, most of these methods focus on eradication of disease, which is not possible if the screening resources are limited. Thus, important real world characteristics such as partial observation and limited resources have not been adequately handled in prior work.

To address this shortcoming in active screening of recurrent diseases, we develop a model of the active screening problem and

present an adaptive software agent, REMEDY (**RE**current screening **M**ulti-round **E**fficient **DY**namic agent). We develop our model in cooperation with a research institute in India (name withheld for anonymity of authors), which partners with the Central Tuberculosis Division (CTD) of India to facilitate active screening for TB. REMEDY assists maximizing effectiveness of active screening under real world budgetary constraints and limited contact information. Such screening of TB patients currently takes place quarterly in over 50 districts scattered across 18 states of India. REMEDY is intended to assist health workers in India in their work in active screening in the field. Our software agent is currently under review before deployment as a means to improve the efficiency of district-wise active screening for tuberculosis in India, although REMEDY has applicability for active screening of other recurrent diseases.

REMEDY is based on three main contributions. Our *first contribution* is a model of the multiagent active screening problem (ACTS). We focus on spread of recurrent infectious diseases modeled using the well-known network SIS model in computational epidemiology [43], which is applicable for many diseases such as syphilis and typhoid. It is the foundation of more complex models that capture more disease dynamics (such as latent states, variation in birth/death rates, or multiple treatment states). The network SIS model is specified by a graph where nodes are individuals and edges indicate physical contact through which disease spread is probabilistic. ACTS models multi-agent interactions in that the nodes in the graph model individuals who interact with other individuals. The individuals can be either susceptible (S) or infected (I). The contribution of multiagent systems in computational epidemiology is well recognized in previous literature [38]. Our model further includes real-world constraints, namely that health workers are uncertain about the health state of individuals, have a small screening budget relative to the population size, and must engage in active screening over multiple rounds (time periods) due to recurrent of the disease. As a first result, we prove that the ACTS problem is NP-hard. To the best of our knowledge, no other model in the AI literature has considered multi-round active screening with partially observable health state for controlling disease spread.

Our second contribution is two novel algorithms, Full- and Fast-REMEDY. In the former, we consider the effect of both current and future screening actions to solve the ACTS problem. Full-REMEDY achieves scale-up via an innovative combination of : (i) easier to optimize upper-bound of the ACTS objective; (ii) a Frank-Wolfe Style gradient descent algorithm; (iii) compact representation of belief states to represent uncertainty. Fast-REMEDY works in a similar fashion as Full-REMEDY, but by optimizing just the current step actions runs almost two orders of magnitude faster than Full-REMEDY in practice. As our *third contribution*, we illustrate the benefits of Full- and Fast-REMEDY via extensive testing on seven different real-world human contact networks against various baselines across a range of realistic disease parameters. For the largest network of ~76,000 individuals we see improvements in performance of almost 40% over the prior best method which directly maps to thousands of fewer infections every six months.

REMEDY is developed to assist screening for infectious diseases under conditions where screening tests are expensive, budgets are limited, and information on the underlying social graph is available. As we also show, the performance improvements exhibited by Full- and Fast-REMEDY are robust to varying levels of missing information in the social graph and budget change, thus enabling the use of our agent to improve the current practice of real-world screening contexts.

2 RELATED WORK

Epidemic models continue to be widely used across biological, social, and computer sciences. Applications range widely, including influence propagation [22], rumor adoption [44], computer virus suppression [17], and of course, disease spread. The studies of disease spreading history can date back to as early as 1760 when Bernoulli proposed the first mathematical epidemic model for smallpox (Variola Major) [5]. In early 2000, studies [43] have found that graph-based epidemic propagation models provide a more realistic approach compared to fully mixed models of earlier literature. Under these graph-based models, non-recurrent and recurrent disease suppression and eradication have been studied using different approaches.

Non-Recurrent Diseases: A large portion of work related to active screening deals primarily with SIR or SEIR type diseases (with two extra state *Exposed* and *Recovered*), often referred to as the *Vaccination Problem* [3, 16, 37, 42, 47], where permanent immunization (entry into R state) can be viewed as removing nodes from the graph. Exploiting this idea, Saha et al. [34] and Tong et al. [39] focus on immunization ahead of an epidemic and suggest a heuristic method of removing a set of k nodes based on the eigenvalues of the adjacency matrix. Zhang and Prakash [47] consider the problem of selecting the best k nodes to immunize in a network after the disease has started to spread. Ren et al. [31] extend the problem to tackle network with graph structure uncertainty. These methods do not apply to our scenario as they assume that a single round of screening offers permanent immunity.

Recurrent Diseases: For diseases in which there is no permanent immunity, one-time screening (cure) is not enough and, further, it may not be reasonable to quarantine patients until the disease has died out. When the true state of the graph in every round is known (in other words, when the policymaker has *perfect observations*), given certain budget constraints, Drakopoulos et al. [12, 13] provide a theoretical lower bound on the expected time needed to eradicate the disease, which grows linearly in the number of nodes. The authors provide a policy to show that disease eradication is possible when the graph structure and budget have specific properties under such perfect observation. Scaman et al. [36] provide a scalable algorithm *maxcut minimization* and tighter theoretical bound of the eradication time based on the idea.

Our work differs from studies of recurrent diseases that assume perfect observations and seek to bound eradication time [12, 13, 36]. The impact of curing uncertainty in these previous works is analyzed in Hoffman and Caramanis [19] by providing non-constructive, algorithm-independent bounds, motivating our work. We focus on developing algorithms to minimize the disease spread. To the best of our knowledge, this complex setting has not been studied previously. Although both inherently a multiagent problem because nodes (agents) make decisions in response to those around them, this problem of minimizing disease spread is different from another well-studied multiagent problem of influence maximization

in general [9, 22, 26, 45, 46]. The influence maximization problem optimizes the selection of seeds or starting nodes for maximizing influence spread that usually has sub-modular property to exploit, as opposed to optimizing the selection of nodes on which to intervene to minimize disease spread.

3 DISEASE MODEL

We introduce the disease model for our problem, which is based on the well-known SIS model [1, 2]. An individual can either be in state S (a healthy individual *susceptible* to disease) or I (the individual is *infected*). SIS models capture the dynamics of recurrent diseases, where permanent immunity is not possible (e.g., TB, typhoid).

We adopt a discrete time SIS model for modeling the disease dynamics propagating on a graph. We summarize our notation in the supplement material. [1] Given a contact network $G(V, E)$, infection spreads via the edges in the network. There are $|V|$ individuals, and we use $\delta(v)$ to denote neighbors of node v in the network. Each individual (node) v in the network at time t is in state $\mathbf{s}_v(t) \in \{S, I\}$. Let $\mathbf{t}_v(t)$ denote the state vector that represents the true state of node v at time t where S is represented as $[1, 0]^\top$ and I as $[0, 1]^\top$. Given the initial state, an infected node infects its healthy neighbors with rate α independently and recovers with probability c. The latter term represents the probability that the node may visit a doctor on its own initiative. The health state transition probabilities of a node is then given by $P\left[s_v(t+1) = \{S, I\}\right] = \mathbf{T}_v^N(t)\mathbf{t}_v(t)$ where

$$\mathbf{T}_v^N(t) = \begin{array}{c} \\ S \\ I \end{array} \begin{array}{c} \begin{array}{cc} S & I \end{array} \\ \begin{bmatrix} 1-q_v & c \\ q_v & 1-c \end{bmatrix} \end{array}, \tag{1}$$

and $q_v = 1 - (1-\alpha)^{|\{u \in \delta(v) \mid \mathbf{s_u}(t) = I\}|}$. The columns denote the state of v at time t and the rows denote the state at $t+1$. The transition probabilities follow the disease dynamics described earlier. In particular, q_v captures the exact probability that node v becomes infected from its neighbors $\{u \in \delta(v) \mid \mathbf{s_u}(t) = I\}$ and c captures the probability that I individuals seek treatment voluntarily.

Given such transition probabilities and an initial state, if no intervention happens, the network state evolves by flipping biased coins for each node to determine their next true state in each round. The process is repeated until the terminal step T is reached.

4 THE ACTIVE SCREENING (ACTS) PROBLEM

Motivated by active screening/contact tracing campaigns that have been practiced since the 1980s [7] and applied in various forms/diseases [6], we propose the Active Screening (ACTS) Problem. Given the SIS model in the previous section, an active screening agent seeks to determine the best node sets $C_a(t) \subset V$ to actively screen and cure with a limited budget of $|C_a(t)| \leq k$ at each round t. The agent does not know the ground truth health state of all individuals. The agent knows the network structure $G(V, E)$, the infection probability α, and recovery probability c. In addition, the agent observes the *naturally cured* node set $C_n(t)$ at time t—because this set of patients come to the clinic voluntarily. Active screening starts after the agent acquires information about $C_n(t)$. Let $C_a(t)$ be the set of nodes that are actively screened at time t. A node $v \in C_a(t)$ becomes cured at time $t+1$. Thus, the transition matrix for a node $v \in C_a(t)$ is $P\left[s_v(t+1) = \{S, I\}\right] = \mathbf{T}_v^A(t)\mathbf{t_v}(t)$, where

$$\mathbf{T}_v^A(t) = \begin{array}{c} \\ S \\ I \end{array} \begin{array}{c} \begin{array}{cc} S & I \end{array} \\ \begin{bmatrix} 1 & 1 \\ 0 & 0 \end{bmatrix} \end{array}. \tag{2}$$

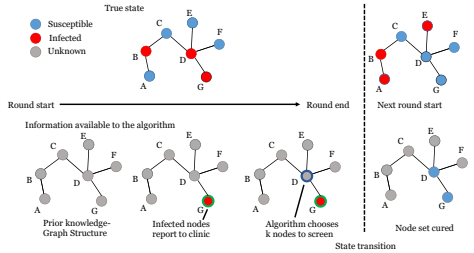


Figure 2: The procedure of the ACTS problem.

The action the agent takes at time t does not affect the transition matrix $\mathbf{T}_v^N(t)$ of the nodes not involved in active screening. Fig. 2 illustrates an example of the problem procedure. The upper part of the figure shows how the true state of the network evolves and the lower part of the figure shows the information available to the algorithm. In this example, there are seven nodes A~G. In each round, infected nodes (nodes B, D, and G in the example) flip a coin and report to the clinic with probability c. The algorithm acquires the information of the nodes that eventually report to the clinic and are about to be cured, which is {G} this round. Based on this information, the algorithm will choose a set of nodes, say {D}, to actively screen. These two sets of nodes are guaranteed to be in S state in the next round. After that, the state of the network transitions and the next round starts.

It is worth noting that although both the nodes that voluntarily report to the clinic and the nodes that are actively screened are guaranteed to be in S state in the next round, their neighbors may still be infected by them in the current round. In the example, node E is infected by node D even though node D was actively screened. This allows us to simplify the state transitions because curing and spreading infection occur at the same time.

Our objective is to maximize the health quality of each individual at each round (in contrast to past work, which primarily focuses on the cost of eradicating the disease entirely). The objective of the ACTS problem is:

$$\min_{C_a(0),\ldots,C_a(T)} \mathbb{E}\left[\sum_{t=0}^{T} \sum_{v \in V} \mathbb{1}_{\mathbf{s}_v(t)=I}\right]. \tag{3}$$

Problem Statement. *(ACTS Problem) Given a contact network $G(V, E)$, the disease and active screening model, find an active screening policy such that the expectation of $\sum_{t=0}^{T} \sum_{v \in V} \mathbb{1}_{\mathbf{s}_v(t)=I}$ is minimized.*

Even assuming we know the ground truth infected state for each node, ACTS is NP-hard. All proofs are in the supplemental material.

Theorem 4.1. *The ACTS Problem is NP-hard.*

We introduce REMEDY, a software agent for assisting to select nodes to actively screen in the ACTS problem. REMEDY, shown in Algorithm 1 has two components: (i) a marginal belief state update that we use for reasoning about the infected status of nodes, and (ii) an algorithm for selecting which nodes to actively screen based on the marginal belief state and an upper bound of the ACTS objective.

[1] Appendix Link: https://bit.ly/2H0uCyK

4.1 Belief State Update

Tracking the exact probability that a node is infected in ACTS requires storing $O(2^{|V|})$ values, which is computationally intractable for reasonably sized graphs. Thus, REMEDY maintains a belief state based on the *marginal* probability that each node is infected, requiring only $O(|V|)$ values for storage. To calculate the marginal infection probability for the next round, we have to consider all possible events of a node's neighbors are infected or not, which appears to require computing a sum with exponentially many terms. We prove in Lemma 4.2 that the sum may be written with a linear number of terms. However, the marginal belief state discards correlation between nodes and this may lead to underestimating the number of infected nodes. We address this issue in the next section by deriving an upper bound for the true ACTS in terms of the marginal belief state.

The marginal belief update is lines 1–7 and 9–15 of Alg. 1. At each round $t \in \{0, \ldots, T-1\}$, we acquire perfect information about the infected state of each I node that naturally recovers, i.e., the nodes that satisfy $s(t) = I$ and $s(t+1) = S$. The state of the remaining nodes is unknown.

Let $x_v(t) \in [0,1]$ be the probability that node v is in state I at time t, and let $\mathbf{b_v}(t) = [1 - x_v(t), x_v(t)]^\top$ be the marginal belief vector. For each node, we update an intermediate belief state $\bar{\mathbf{b}}_\mathbf{v}(t) = [1 - \bar{x}_v(t), \bar{x}_v(t)]^\top$ in which $\bar{x}_v(t) = 1$ for $v \in C_n(t)$ and $\bar{x}_v(t) = \frac{(1-c)x_v(t)}{(1-x_v(t))+(1-c)x_v(t)}$ for the remaining nodes $v \in V \setminus C_n(t)$. These update steps are in lines 1–7 of Algorithm 1. This intermediate belief state is then exploited by the action choice subroutine to select $C_a(t)$, the node set we actively cure (line 8). After that, we calculate the marginal belief state at the next round: $\mathbf{b_v}(t+1) = \mathbf{B}_v^N(t)\bar{\mathbf{b}}_\mathbf{v}(t)$ and $\mathbf{b_v}(t+1) = \mathbf{B}_v^A(t)\bar{\mathbf{b}}_\mathbf{v}(t)$ for $v \in V \setminus (C_n(t) \cup C_a(t))$ and $v \in C_n(t) \cup C_a(t)$ respectively where

$$\mathbf{B}_v^N(t) = \begin{array}{c} \\ S \\ I \end{array} \begin{array}{c} \begin{array}{cc} S & I \end{array} \\ \begin{bmatrix} 1-p_v & 0 \\ p_v & 1 \end{bmatrix} \end{array}, \mathbf{B}_v^A(t) = \begin{array}{c} \\ S \\ I \end{array} \begin{array}{c} \begin{array}{cc} S & I \end{array} \\ \begin{bmatrix} 1 & 1 \\ 0 & 0 \end{bmatrix} \end{array} \tag{4}$$

and $p_v = 1 - \prod_{u \in \delta(v)}(1 - \alpha \bar{x}_u(t))$. These steps are shown in lines 9-15 of Alg. 1. The transition matrix $\mathbf{B}^N$ does not contain parameter c because each node in the I state that did not naturally recover will remain in I state with probability 1. Note that p_v is not an approximation but the exact value calculated by listing all possible events of v's neighbor being infected or not which we show in Lemma 4.2.

Lemma 4.2. *The exact marginal probabilities of* $P[s(t+1) = I | s(t) = S]$ *can be calculated by* p_v.

4.2 Action Choice Algorithm

Possible approaches: We now turn our attention to selecting the set of nodes to actively screen, i.e., line 8 in Alg. 1. First, treating the Acts problem as a POMDP and applying state of the art reinforcement learning techniques is not feasible for the real world scenario we are aiming for [27]. This is due to the fact that the computation time scales poorly with the high dimension action choice, which is exponential in the budget for our problem. Even when we approximated the actual feasible action choice by choosing the nodes greedily one node at a time and estimating the reward function, the resulting approach performed poorly and did not scale up to 20 nodes, which is less than even the smallest graph in our dataset.

Algorithm 1 REMEDY

Input: $\mathbf{A}$, $\mathbf{b}(t)$, α, c, $C_n(t)$, t, T, k
Output: $C_a(t)$, $\mathbf{b}(t+1)$

1: **for** $v \in V$ **do**
2: **if** $v \in C_n(t)$ **then**
3: $\bar{\mathbf{b}}_\mathbf{v}(t) \leftarrow [0,1]^\top$
4: **else**
5: $\bar{\mathbf{b}}_\mathbf{v}(t) \leftarrow \frac{[(1-x_v(t)),(1-c)x_v(t)]^\top}{((1-x_v(t))+(1-c)x_v(t))}$
6: **end if**
7: **end for**
8: $C_a(t) \leftarrow$ ActionChoice$(\mathbf{A}, \bar{\mathbf{b}}(t), \alpha, c, C_n(t), t, T, k)$
9: **for** $v \in V$ **do**
10: **if** $v \in V \setminus C_n(t) \cup C_a(t)$ **then**
11: $\mathbf{b_v}(t+1) \leftarrow \mathbf{B}_v^N(t)\bar{\mathbf{b}}_\mathbf{v}(t)$
12: **else**
13: $\mathbf{b_v}(t+1) \leftarrow \mathbf{B}_v^A(t)\bar{\mathbf{b}}_\mathbf{v}(t)$
14: **end if**
15: **end for**
16: **return** $C_a(t)$, $\mathbf{b}(t+1)$

One fast yet naive approach to this problem is to select the node set with maximum marginal belief to be in I state. This approach can be computed in $O(|V|)$, but it does not take the network structure and future infection probabilities into account. For example, suppose we have a tree structure with a known infection state: the root is the only infected node. The belief-based approach will screen the root and spend the remainder of the budget on random nodes. This is suboptimal because the remaining budget could be spent on the children of the root to prevent the disease from spreading.

Another approach is to choose nodes based on the graph structure. For example, we can select the nodes that, if deleted (permanently actively screened), would reduce the largest eigenvalue of the graph the most [29]. This approach guarantees that the infection is eradicated in the long term if the largest eigenvalue can be reduced below $\frac{c}{\alpha}$ for sufficient budget k. However, structure based approaches perform poorly when there are many nodes with identical roles in the graph structure, e.g., in symmetric graphs. Here, belief information would be more useful because it takes into account current signals from local neighbors of nodes

Our approach: The key novelty of our software agent is that it brings together three key features: the use of belief states, a Frank-Wolfe style gradient-based algorithm for efficient reasoning about the structure of the graph, and use of an upper-bound of the *true* ACTS objective. Whereas algorithms for active screening have typically used discrete reasoning such as Markov chain (see Related work) and have not appealed to gradient-based approaches, it is the novel combination of this gradient-based approach with the use of belief states and upper bounds that is key in our work. Note that whereas marginal belief states avoid the exponential storage requirement of exact belief states, they typically underestimate the expected number of infected nodes as a result of the lost correlation information. We rely on using an upper bound on true number of infected nodes reduces this effect — thus we face the issue of

determining a suitable upper bound. Our desiderata for determining this upper bound are therefore: (i) encapsulate the observations and actions of past and future, (ii) provide a performance guarantee compensating the information lost from marginal belief state, and (iii) be minimizable in time polynomial in T, k and $|V|$.

We develop two different algorithms for action choice: Full-ActionChoice, which looks ahead through all future actions and Fast-ActionChoice, a less computationally intensive variant that considers only the current action, allowing it to exploit eigenvalue decomposition. We refer to REMEDY agent using Full and Fast-ActionChoice as Full and Fast-REMEDY.

We start with some preliminary notation. To encapsulate the effect of active-screening toward our objective function, we define the $|V|\times|V|$ diagonal action matrix $\mathbf{R}_a(t)$ at time t as $\mathbf{R}_a(t)_{v,v} = 1$ if and only if $v \in C_a(t)$, and 0 otherwise. For the current round, say t_0, we observe the nodes that are cured and need to decide the nodes to actively screen. We define the *naturally cured matrix* $\mathbf{R}_n(t_0)$ as $\mathbf{R}_n(t_0)_{v,v} = 1$ if and only if $v \in C_n(t_0)$, which encapsulates the knowledge we gain from natural recovery in the current round. Let vector $\mathbf{x}(t)$ represent $x_v(t)$ for all v. To bound $\mathbf{x}(t)$ across all rounds given the actions we take, let $\mathbf{M}' = \alpha\mathbf{A} + \mathbf{I}$, where $\mathbf{A}$ is the adjacency matrix and $\mathbf{I}$ is the identity matrix, define the *upper bound transition matrix* for the current round ($t = t_0$) as $\mathbf{M}_a(t_0) = (\mathbf{I} - \mathbf{R}_a(t_0) - \mathbf{R}_n(t_0))\mathbf{M}'$. And for future rounds ($t > t_0$), we define it as $\mathbf{M}_a(t) = (\mathbf{I} - \mathbf{R}_a(t))\mathbf{M}$ where $\mathbf{M} = \alpha\mathbf{A} + (1-c)\mathbf{I}$.

Theorem 4.3. *Let the current time be t_0. $\mathbf{M}_a$ is defined as above for t_0 and $t > t_0$. The ACTS objective (Eq. 3) is bounded above by:*

$$\mathbb{E}[\sum_{t=t_0}^{T}\sum_{v\in V}|s_v(t) = I|] \le F = \mathbb{1}^\top \sum_{t=t_0}^{T}\prod_{\tau=t_0}^{t}\mathbf{M}_a(\tau)\mathbf{x}(t_0) \tag{5}$$

$$\text{where } \prod_{\tau=t_0}^{t}\mathbf{M}_a(\tau) = \mathbf{M}_a(t)\mathbf{M}_a(t-1)...\mathbf{M}_a(t_0). \tag{6}$$

Given that the function F upper bounds our objective function, we next describe the method we use to select the action matrix $R_a(t)$ that minimizes F for every round. Distinct from previous literature, our objective takes into account the number of infected nodes at each round. We also have the flexibility to change the action we take based on the observation we make in each round. Such flexibility results a solution space of size $\binom{|V|}{k}^T$, making the bound challenging to optimize exactly, since it is nonconvex. Hence, we apply a Frank-Wolfe style method [15] to the continuous relaxation. The result is Full-ActionChoice (Alg. 2), a gradient-based algorithm that runs for L iterations, simultaneously updating the actions taken at each round. It begins with an arbitrary feasible point and performs three steps per iteration: (i) computes the gradient of the objective at the current point, (ii) optimizes the linear approximation to the objective over the true (not relaxed) feasible set, and (iii) steps toward it. After L iterations, we greedily round the solution, selecting the k nodes that have highest values. Each time we receive a new naturally cured set, we run Alg. 2 over all remaining rounds and output the action for the current time.

We describe Alg. 2 in more detail. We initialize to an arbitrary feasible point in Ψ, the convex hull of the binary valued $\mathbf{R}_a(t)$: we choose $\mathbf{R}_a^0(t) = \mathbf{0}$ for $t = t_0 \sim T$ in iteration $l = 0$ (line 1). In each iteration, we need to caluclate the gradient of F w.r.t. the action choice, which is the GradientOracle of line 4. We relax

Algorithm 2 Full-ActionChoice

Input: $\mathbf{A}$, $\bar{\mathbf{b}}(t_0)$, α, c, T, t_0, k
Output: $C_a(t_0)$

1: $\mathbf{R}_a^0(t) \leftarrow \mathbf{0} \quad \forall t$
2: **for** $l = 1...L$ **do**
3: **for** $t = t_0...T$ **do**
4: $\Delta(t) \leftarrow$ GradientOracle($\mathbf{R}_a^{l-1}$)
5: $\mathbf{R}_a^*(t) \leftarrow$ ProjectFeasible(Δ, k)
6: $\mathbf{R}_a^l(t) \leftarrow (1-\gamma_l)\mathbf{R}_a^{l-1}(t) + \gamma_l\mathbf{R}_a^*(t)$
7: **end for**
8: **end for**
9: $C_a(t_0) \leftarrow \arg\max_k \mathbf{R}_a^L(t_0)$
10: **return** $C_a(t_0)$

the optimization to the continuous problem by allowing $\mathbf{R}_a(t)_{v,v}$ to take real values between 0 and 1, which can be interpreted as the probability of choosing node v. By taking the derivative of F, the gradient w.r.t. action at each time t is

$$\frac{\partial F}{\partial \mathbf{R}_a(t)} = -\sum_{t'=t+1}^{T}\prod_{\tau=t'}^{t+1}\mathbf{M}_a^\top(\tau)\mathbb{1}\mathbf{x}^\top(t_0)\prod_{\tau=t-1}^{t_0}\mathbf{M}_a^\top(\tau), \tag{7}$$

The above gradient is a matrix $\Delta(t)$, where the diagonal elements $\Delta(t)_{v,v}$ represent the gradient w.r.t. the choice of node v to actively screen at time t.

We then minimize this linear approximation over the true feasible set. Since the objective is linear and the only constraints are individual variable bounds and the budget constraint, we can optimize exactly by greedily selecting the k nodes with largest $\Delta(t)_{v,v}$ in line 5. We set the initial point $\mathbf{R}_a^l(t)$ of the next iteration in line 6, in which $\gamma_l = 2/(l+2)$ is the step size of Frank-Wolfe algorithm. Since Ψ is convex and $\mathbf{R}_a^l(t)$ is the convex combination of two feasible points, it is guaranteed that it will remain in the convex hull Ψ after the update. After L iterations, we output our action in the current round by greedily selecting k nodes of the relaxed $\mathbf{R}_a^L(t_0)$ of the final iteration, as line 9 shows.

The Full-REMEDY algorithm considers future actions simultaneously and has time complexity of $O(T^2|V|^\omega)$, where the exponent ω arises from complexity of matrix multiplication (best known ω is around 2.37). The algorithm used scales well to the budget k. However, calculating such solutions for a very large network—which is often the case for active screening—can be time consuming. To reduce time complexity, we further simplify the upper-bound function by assuming that no actions are taken in the future rounds and ignore their effect on the current decision making in Fast-ActionChoice (Alg. 3). By ignoring future actions, the action matrix $M_a(t)$ in Full-REMEDY is simplified to constant M. The contribution of actively screening each node can be written as the following vector form:

$$\mathbb{1}^\top\sum_{\tau=0}^{T-t_0-1}\mathbf{M}^\tau diag(\mathbf{M}_n\mathbf{x}(t_0)), \tag{8}$$

where $\mathbf{M}_n = (\mathbf{I} - \mathbf{R}_n(t_0))\mathbf{M}'$. Now, since $\mathbf{M}$ is the same for every future round, $\mathbf{M}$ can be decomposed as $\mathbf{Q}\mathbf{\Lambda}\mathbf{Q}^\top$ ahead of time, where $\mathbf{Q}$ is a matrix comprised of the eigenvectors of $\mathbf{M}$, and $\mathbf{\Lambda}$ a diagonal matrix comprised of the eigenvalues along the diagonal. Such a matrix can be approximated by calculating only the top m largest eigenvalues and their eigenvectors using the Lanczos algorithm [24] that has a complexity of $O(|E|)$ (assuming the large network

Algorithm 3 FAST-ACTIONCHOICE

Input: $\mathbf{A}$, $\bar{\mathbf{b}}(t_0)$, α, c, T, t_0, k
Output: $C_a(t_0)$

1: **if** $t_0 = 0$ **then**
2: $\quad \mathbf{M} \leftarrow \alpha\mathbf{A} + (1-c)\mathbf{I}$
3: $\quad \mathbf{Q}_m, \mathbf{\Lambda}_m \leftarrow$ LANCZOS$(\mathbf{M}, m)$
4: **end if**
5: **Scores** $\leftarrow \mathbb{1}^\top \mathbf{Q}_m (\sum_{\tau=0}^{T-t_0-1} \mathbf{\Lambda}_m^\tau) \mathbf{Q}_m^\top diag(\mathbf{M}_n \mathbf{x}(t_0))$
6: $C_a(t) \leftarrow k$ nodes with highest scores in vector **Scores**

is sparse), yielding the FAST-ACTIONCHOICE shown in Alg. 3. The approximate $\mathbf{M}$ is given by $\mathbf{Q}_m\mathbf{\Lambda}_m\mathbf{Q}_m^\top$, where these matrices are computed in line 3. In line 5, the well-known result $(\mathbf{Q}_m\mathbf{\Lambda}_m\mathbf{Q}_m^\top)^\tau = \mathbf{Q}_m\mathbf{\Lambda}_m^\tau\mathbf{Q}_m^\top$ is used to approximate $\mathbf{M}^\tau$. The time complexity of FAST-REMEDY is $O(|V|^2)$ assuming constant m.

5 EXPERIMENTS

We perform experiments comparing FAST- and FULL-REMEDY to baselines on a variety of real-world datasets. Table 1 lists the networks and their properties. Most of the networks were collected in human contact settings. The networks are carefully selected to have significant variation in size, degree, average shortest path length, assortativity and epidemic threshold (spectral radius).

Setting. Unless explicitly stated otherwise, we assume the budget k allows for screening and treatment of 20% of the total population $|V|$ per round. All results are averages over 30 runs.

In practice, active screening is performed only after conducting initial surveys on the prevalence and incidence of the disease. To simulate this, we run experiments in two stages.

Stage 1 (Survey Stage). This stage starts at $t = 0$ with 25% of individuals in I, selected uniformly at random, and ends at $t = 10$. No active screening is done and the disease evolves naturally. The initial belief $\mathbf{b}(\mathbf{0})$ for all nodes is assumed to be $[0.5, 0.5]^\top$ since we have no prior information. Beliefs are updated according to the belief update algorithm in the *Disease model* section. This belief update requires knowledge of α and c. There is a rich literature of how to estimate the disease parameters (α and c) in this stage and these methods have been tested on real-world scenarios [11, 23, 33]. Here, we assume that such parameters are known.

Such parameters can vary from disease to disease. For example, the transmission rate of Pertussis can be as high as 0.47 for certain age groups [18], and as low as 0.035 for Syphilis [33]. The cure rate also depends on how resourceful the target regions are. We initially assume $(\alpha, c) = (0.1, 0.1)$ and then evaluate a range of values.

Stage 2 (ACTS Stage). Here, we consider various screening algorithms. We perform active screening from $t = 11$ to $t = T = 20$ to represent 5 years of time (each round is 6 months [8]). Beliefs are updated according to the belief update scheme presented in *Disease Model and Background* section.

5.1 Metrics

We compare the outcomes of the following screening strategies compared to no intervention (**None**). In **None**, the evolution of the health states is based on disease dynamics only, with no active screening for all T rounds. The improvement over **None** is reported as the number of fewer infections as compared to **None**. Thus, the larger this number the better the performance of the algorithm.

(1a) RANDOM: Randomly select nodes for active screening.
(1b) MAXDEGREE: Successively choose nodes with the largest degree until the budget is reached.
(1c) EIGENVALUE: Greedily choose nodes that reduce the largest eigenvalue of $\mathbf{A}$ the most until the budget is reached.
(1d) MAXBELIEF: Choose nodes with the highest probability of being in the I state.
(1e) BELIEFCUTWIDTH: A modified version of the CutWidth method for a problem with known infection state [12, 36]. Since the original method requires known infection state, we modified it by using a sample from the marginal belief state as a substitute of the true state.

Unfortunately, the data sets from countries that have high infectious disease burden, for which the algorithms may be applied on the ground in the future, have restrictive terms of use. For privacy and security reasons, they cannot be shared externally. Instead, we present the result of testing these algorithms on the following realistic contact networks collected from diverse sources.

(2a) **Hospital** [41]: A dense contact network collected in a university hospital to study the path of disease spread.
(2b) **India** [4]: A human contact network collected from a rural village in India where active screening with limited budget may take place.
(2c) **Face-to-face** [21]: A network describing face-to-face contact in which influenza might spread through the close contact of individuals.
(2d) **Flu** [35]: A network of close proximity interactions in an American high school.
(2e) **Irvine** [28]: A friendship network collected from students in UC Irvine, used to study rumor modeled as epidemic spread.
(2f) **Escort** [32]: A sexual contact network between escorts and sex buyers in which STDs may be spread collected over six years.
(2g) **Epinion** [25]: A trust network of a general consumer review site. This dataset is adopted mainly to show the scalability of the algorithms.

The results are shown in Table 1. We begin with initial observations and provide a more detailed analysis in the following section. In most cases, although the baselines behave differently for each data set, both versions of REMEDY make substantial improvements over them, and, as expected, FULL-REMEDY exhibits better performance than FAST-REMEDY. In **Irvine**, the largest network for which all the algorithms are able to complete running within a 24-hour period, FAST-REMEDY and FULL-REMEDY outperformed MAXDEGREE, the next best competitor, by 16.29% and 36.23% respectively. FAST-REMEDY also outperformed its next best competitor (MAXDEGREE) on **Epinion**, the largest network, by 37.44%. We further examined the performance of REMEDY for a range of α and c values (see Fig. 4). FAST- and FULL-REMEDY continue to perform better than their closest competitors.

Specifically, Fig. 3 shows the average number of infected nodes in each round on the **India** network. The values shown in Table 1 are the accumulation of the difference between NONE and each algorithm. FULL-REMEDY steadily outperforms the other algorithms

Network	$\|V\|$	$\frac{1}{\lambda_A^*}$	d	ρ_L	ρ_D	Number of reduced infections						
						Random	Max-Degree	Eigenvalue	Max-Belief	BeliefCutWidth	Fast-REMEDY	Full-REMEDY
Hospital [41]	75	0.027	15.19	1.60	-0.18	144	150	151	150	147	**156**	**160**
India [4]	202	0.095	3.43	3.11	0.02	605	470	420	636	754	**890**	**901**
Face-to-face [21]	410	0.042	6.74	3.63	0.23	809	843	745	1057	1100	**1297**	**1409**
Flu [35]	788	0.003	150.12	1.62	0.05	1336	1421	1431	1438	1396	**1443**	**1446**
Irvine [28]	1899	0.021	7.29	3.06	-0.18	4630	5741	3692	4957	5623	**6676**	**7821**
Escorts [32]	16730	0.032	2.33	4.20	-0.03	27400	30167	TLE	29493	TLE	**46549**	TLE
Epinion [25]	75879	0.004	2.67	4.40	-0.04	187369	228174	TLE	207565	TLE	**285280**	TLE

Table 1: Improvement over None in terms of the number of reduced infections (the larger the better). All computations are carried out with $\alpha = 0.1, c = 0.1$. Here, TLE signifies that the 24 hour limit was exceeded.

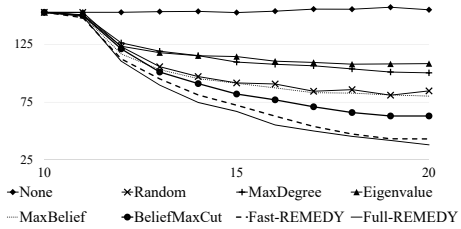
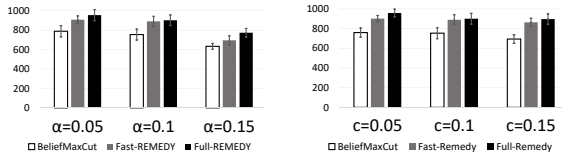


Figure 3: The average number of infected nodes (y-axis) vs. time (x-axis) of the ACTS stage in the India network.



(a) c=0.1 **(b) a=0.1**

Figure 4: Improvement over None (y-axis) under different parameter settings for India network.

in each round and keeps decreasing the infected node number and Fast-REMEDY follows slightly behind it. The other algorithms, however, reach steady state and stop decreasing earlier.

Fig. 5 gives the running time of all the algorithms in different networks, sorted by size. Fast-REMEDY is about two orders of magnitude faster compared to Full-REMEDY, and takes about two hours on the largest network. All the algorithms that select a fixed set of nodes (Eigenvalue and MaxDegree) in every round are timed for the first round only for fairness of comparison. On **Escort** and **Epinion**, Eigenvalue, BeliefMaxCut and Full-REMEDY exceed 24 hours of computation time. It appears that only algorithms with complexity of $O(|V|^2)$ or less terminate within the time limit.

5.2 Policy Analysis: Cure, Prevent, and Miss

To analyze the performance differences between algorithms, we introduce metrics that decompose the effect of active screening. As Figure 6 shows, screening a node has one of the following three effects:

- **Cure:** screening a node that is currently in the I state causes that node to transition to the S state.
- **Prevent:** screening a node in the S state prevents that node from entering the I state if it would have otherwise.

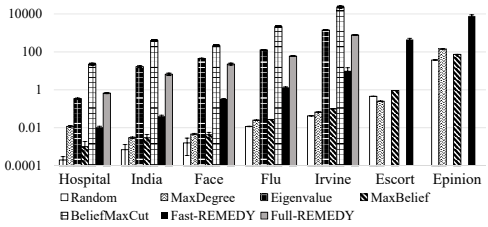


Figure 5: Computation time (y-axis, in seconds) in different contact networks in logarithmic scale.

- **Miss:** screening a node in the S state has no effect if that node would not have transitioned to the I state.

We analyze the performance of the algorithms by measuring how many of their active screening actions result in each effect. Figure 7 shows the frequency of each effect for each algorithm, averaged over the rounds in the ACTS stage.

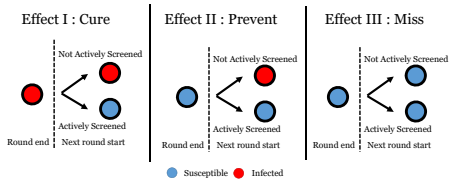
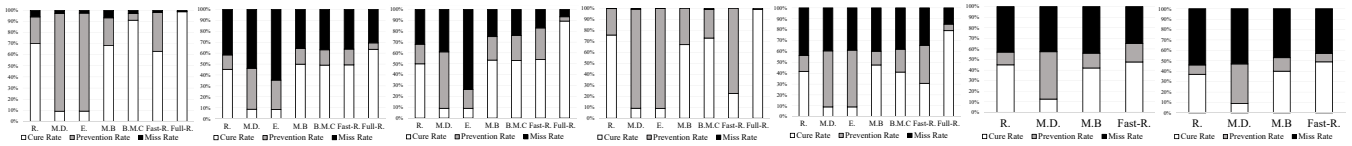


Figure 6: Three possible effects of active screening.

Active screening actions that cure infected nodes or prevent susceptible nodes from becoming infected will generally decrease the amount of infection in the network, whereas a miss has no effect. Therefore, we expect that algorithms with higher combined cure and prevention rates should have higher performance. This is indeed the case as the height of combined cure and prevention in Figure 7 is strongly correlated to the performance in Table 1 with only a few exceptions. The success of Fast- and Full-REMEDY can be explained primarily along these lines. Due to space constraints, a more detail analysis of the performance variation of each algorithm on each network based on their properties and three active screening effects is provided in the supplemental material.

6 DEPLOYMENT CONSIDERATIONS

In this section, we consider the effect of practical constraints on deployment of REMEDY in real-world active screening situations. Our ultimate aim is to be able to use the algorithms in settings such as active screening for tuberculosis in India.



(a) Hospital (b) India (c) Face-to-face (d) Flu (e) Irvine (f) Escort (g) Epinion

Figure 7: Frequency of each active screening effect, averaged over the ACTS stage. (R.: RANDOM; M.D.: MAXDEGREE; E.:EIGENVALUE; M.B.: MAXBELIEF; B.M.C.: BELIEFMAXCUT; Fasr-R:FAST-REMEDY; Full-R.:FULL-REMEDY)

Active screening implementations usually choose to screen whole districts where disease infections are severe. Recent experimental attempts that involved screening every first degree contact of reported patients (these are designated as *Naturally Cured* in this paper)[20]. To address the challenge of deploying this work as a next step in active screening implementations, it is essential to take into account realistic barriers such as limited budget or missing information about the contact structure.

6.1 Impact of Budget

Determining the improvement an intervention can achieve with various budgets is critical when informing health policy. We therefore study the improvement possible over different budget values for two realistically modeled diseases: Influenza and Syphilis.

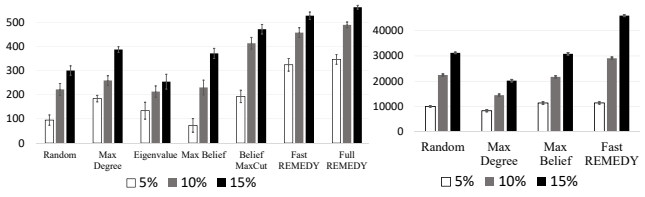
Influenza. For Influenza, we use parameters estimated by previous literature through a continuous survey administered in a student residence hall community [11]. The transition rate is estimated to be $\alpha = 0.024$ and the self-cure rate is estimated to be $c = 0.3$. We test the algorithms on the **Face-to-face** network, since this network is used to study the dynamics of SIS-type epidemic spread in its original paper [21].

Fig. 8 (a) shows that both FAST-REMEDY and FULL-REMEDY outperform other baselines under realistic settings. The difference grows larger as the budget increases. According to Prakash et al. [29], such a network requires at least $k/|V| \geq \alpha\lambda_A = 57\%$ for random screening to fully eradicate the disease. However, the epidemic dies out at the end of the 20th round (in all runs) when FULL-REMEDY is deployed with a budget of only $k/|V| = 15\%$.

Syphilis. We use the syphilis parameters derived by Saad-Roy et al. [33]. The natural cure rate is estimated to be $c = 0.01$ and transmission rate $\alpha = 0.035$. The network is the **Escort** network with 16730 nodes, an STD contact network. Because the network is large, we show only the algorithms that do not exceed running time due to time complexity, which are Random, MaxDegree, MaxBelief and FAST-REMEDY. Fig. 8 (b) shows that FAST-REMEDY achieves significantly better results than all other baselines. On average, it saves 1140, 2900, and 4600 people from becoming infected every six months for 5%, 10% and 15% budgets, respectively.

6.2 Impact of Structure Uncertainty

In realistic settings, it is quite possible that the contact network is not known precisely. To simulate this, we randomly remove edges from the graph and then provide the graph with missing edges as input to the algorithms. All the algorithms make decisions based on this graph with missing edges without knowing such fact while the disease spread happens along the true network with all edges.



(a) Influenza (b) STD

Figure 8: Improvement over None (y-axis) with specific disease parameters under different budget constraints (corresponding to 5%, 10%, 15% of total population).

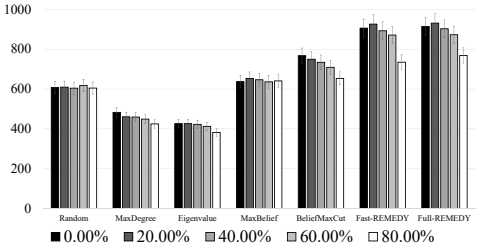


Figure 9: The improvement over None for different percentage of edge information lost in the India network.

Both version of REMEDY still significantly outperform other baselines even when the percentage of edges randomly removed is as high as 80% (Fig. 9). In other words, it is able to outperform the other implementations with only 20% of the contacts are known.

7 CONCLUSION

This paper presents the REMEDY agent for a novel active screening multiagent problem (ACTS) that takes into account real world constraints such as uncertain health states and limited intervention resources. No previous work has addressed the challenge of such uncertainty raised from the emerging application active screening of recurrent diseases. Active screening provides a powerful yet expensive means to control disease spread in the public health domain that passive screening cannot achieve due to its latency of cure. The agent is developed to assist our collaborator in India to decide who and when health workers should invest their limited resources and improve the current practice approach. We introduced two variant of algorithms the agent used, FULL-REMEDY and FAST-REMEDY and examined them on various real human contact networks and realistic disease parameters to show their superior performance over any past approach.

ACKNOWLEDGMENTS

This work was supported by the Army Research Office (MURI W911NF1810208).

REFERENCES

[1] R. M. Anderson and R. M. May. 1992. *Infectious diseases of humans: dynamics and control.* Oxford University, Oxford University.

[2] N. TJ Bailey. 1975. *The mathematical theory of infectious diseases and its applications.* Charles Griffin & Company Ltd, Charles Griffin & Company Ltd.

[3] F. G. Ball, E. S. Knock, and P. D. O'Neill. 2015. Stochastic epidemic models featuring contact tracing with delays. *Mathematical biosciences* 266 (2015), 23–35.

[4] A. Banerjee, A. G. Chandrasekhar, E. Duflo, and M. O. Jackson. 2013. The diffusion of microfinance. *Science* 341, 6144 (2013), 1236498.

[5] D. Bernoulli and S. Blower. 2004. An attempt at a new analysis of the mortality caused by smallpox and of the advantages of inoculation to prevent it. *Reviews in medical virology* 14, 5 (2004), 275–288.

[6] J. Braxton, D. W. Davis, B. Emerson, E. W. Flagg, J. Grey, L. Grier, A. Harvey, S. Kidd, J. Kim, K. Kreisel, et al. 2017. Sexually transmitted disease surveillance. *CDC* (2017).

[7] D. Cadman, L. Chambers, W. Feldman, and D. Sackett. 1984. Assessing the effectiveness of community screening programs. *Jama* 251, 12 (1984), 1580–1585.

[8] CDC. 2011. Tuberculosis: General Information. *MMWR. Recommendations and reports: Morbidity and mortality weekly report.* (2011). https://www.cdc.gov/tb/publications/factsheets/general/tb.pdf

[9] W. Chen, Y. Wang, and S. Yang. 2009. Efficient influence maximization in social networks. In *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 199–208.

[10] P. Chinnakali, P. Thekkur, G. Ramaswamy, K. Selvaraj, et al. 2016. Active screening for tuberculosis among slum dwellers in selected urban slums of Puducherry, South India. *Annals of Tropical Medicine and Public Health* 9, 4 (2016), 295.

[11] W. Dong, K. Heller, and A. S. Pentland. 2012. Modeling infection with multi-agent dynamics. In *International Conference on Social Computing, Behavioral-Cultural Modeling, and Prediction*. Springer, 172–179.

[12] Kimon Drakopoulos, Asuman Ozdaglar, and John N Tsitsiklis. 2014. An efficient curing policy for epidemics on graphs. *IEEE Transactions on Network Science and Engineering* 1, 2 (2014), 67–75.

[13] Kimon Drakopoulos, Asuman Ozdaglar, and John N Tsitsiklis. 2016. When is a network epidemic hard to eliminate? *Mathematics of Operations Research* (2016).

[14] K. TD Eames and M. J Keeling. 2003. Contact tracing and disease control. *Proc. R. Soc. Lond., B, Biol. Sci.* 270 (2003), 2565.

[15] M. Frank and P. Wolfe. 1956. An algorithm for quadratic programming. *Naval research logistics quarterly* 3, 1-2 (1956), 95–110.

[16] A. Ganesh, L. Massoulié, and D. Towsley. 2005. The effect of network topology on the spread of epidemics. In *Proceedings IEEE 24th Annual Joint Conference of the IEEE Computer and Communications Societies.*, Vol. 2. IEEE, 1455–1466.

[17] Michele Garetto, Weibo Gong, and Don Towsley. 2003. Modeling malware spreading dynamics. In *IEEE INFOCOM 2003. Twenty-second Annual Joint Conference of the IEEE Computer and Communications Societies (IEEE Cat. No. 03CH37428)*, Vol. 3. IEEE, 1869–1879.

[18] H. W. Hethcote. 1997. An age-structured model for pertussis transmission. *Mathematical biosciences* 145, 2 (1997), 89–136.

[19] Jessica Hoffmann and Constantine Caramanis. 2018. The Cost of Uncertainty in Curing Epidemics. *Proceedings of the ACM on Measurement and Analysis of Computing Systems* 2, 2 (2018), 31.

[20] King K. Holmes, Stefano Bertozzi, Barry R. Bloom, and Prabhat. Jha. 2017. *Disease Control Priorities, (Volume 6): Major Infectious Diseases.* The World Bank.

[21] L. Isella, J. Stehlé, A. Barrat, C. Cattuto, J.-F. Pinton, and W. Van den Broeck. 2011. What's in a crowd? Analysis of face-to-face behavioral networks. *J. Theor. Biol.* (2011), 166.

[22] D. Kempe, J. Kleinberg, and E. Tardos. 2003. Maximizing the spread of influence through a social network. In *Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 137–146.

[23] C. Kirkeby, T. Halasa, M. Gussmann, N. Toft, and K. Græsbøll. 2017. Methods for estimating disease transmission rates: Evaluating the precision of Poisson regression and two novel methods. *Sci. Rep.* 7 (2017), 9496.

[24] C. Lanczos. 1950. *An iteration method for the solution of the eigenvalue problem of linear differential and integral operators.* United States Gov. Press Office Los Angeles, CA.

[25] Jure Leskovec and Andrej Krevl. 2014. Stanford Large Network Dataset Collection. http://snap.stanford.edu/data. (2014).

[26] Mahsa Maghami and Gita Sukthankar. 2012. Identifying influential agents for advertising in multi-agent markets. In *Proceedings of the 11th International Conference on Autonomous Agents and Multiagent Systems-Volume 2*. International Foundation for Autonomous Agents and Multiagent Systems, 687–694.

[27] Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Andrei A Rusu, Joel Veness, Marc G Bellemare, Alex Graves, Martin Riedmiller, Andreas K Fidjeland, Georg Ostrovski, et al. 2015. Human-level control through deep reinforcement learning. *Nature* 518, 7540 (2015), 529.

[28] P. Panzarasa, T. Opsahl, and K. M. Carley. 2009. Patterns and dynamics of users' behavior and interaction: Network analysis of an online community. *J. Assoc. Inf. Sci. Technol.* (2009), 911.

[29] B. A. Prakash, D. Chakrabarti, N. C. Valler, M. Faloutsos, and C. Faloutsos. 2012. Threshold conditions for arbitrary cascade models on arbitrary networks. *Knowledge and information systems* 33, 3 (2012), 549–575.

[30] Wayan CWS Putri, David J Muscatello, Melissa S Stockwell, and Anthony T Newall. 2018. Economic burden of seasonal influenza in the United States. *Vaccine* 36, 27 (2018).

[31] Yizhi Ren, Mengjin Jiang, Ye Yao, Ting Wu, Zhen Wang, Mengkun Li, and Kim-Kwang Raymond Choo. 2018. Node Immunization in Networks with Uncertainty. IEEE, 1392–1397.

[32] L. EC Rocha, F. Liljeros, and P. Holme. 2010. Information dynamics shape the sexual networks of Internet-mediated prostitution. *Proceedings of the National Academy of Sciences* 107, 13 (2010), 5706–5711.

[33] CM Saad-Roy, Z. Shuai, and P. van den Driessche. 2016. A mathematical model of syphilis transmission in an MSM population. *Mathematical biosciences* 277 (2016), 59–70.

[34] S. Saha, A. Adiga, B. A. Prakash, and A. K. S. Vullikanti. 2015. Approximation algorithms for reducing the spectral radius to control epidemic spread. In *Proceedings of the 2015 SIAM International Conference on Data Mining*. SIAM, 568–576.

[35] M. Salathé, M. Kazandjieva, J. W. Lee, P. Levis, M. W. Feldman, and J. H. Jones. 2010. A high-resolution human contact network for infectious disease transmission. *Proceedings of the National Academy of Sciences* 107, 51 (2010), 22020–22025.

[36] K. Scaman, A. Kalogeratos, and N. Vayatis. 2016. Suppressing epidemics in networks using priority planning. *IEEE Transactions on Network Science and Engineering* (2016).

[37] C. Sun and Y.-H. Hsieh. 2010. Global analysis of an SEIR model with varying population size and vaccination. *Applied Mathematical Modelling* 34, 10 (2010), 2685–2697.

[38] S. Swarup, S. G. Eubank, and M. V. Marathe. 2014. Computational epidemiology as a challenge domain for multiagent systems. In *Proceedings of the 2014 international conference on Autonomous agents and multi-agent systems*. Citeseer, 1173–1176.

[39] H. Tong, B. A. Prakash, T. Eliassi-Rad, M. Faloutsos, and C. Faloutsos. 2012. Gelling, and melting, large graphs by edge manipulation. In *Proceedings of the 21st ACM international conference on Information and knowledge management*. ACM, 245–254.

[40] International Union Against Tuberculosis and Lung Disease. 2018. Community-based active case finding in India can test and treat more people with TB in hard-to-reach tribal areas. *Union news report.* (2018). https://www.theunion.org/

[41] P. Vanhems et al. 2013. Estimating potential infection transmission routes in hospital wards using wearable proximity sensors. *PloS one* 8 (2013), 73970.

[42] N. Wang. 2005. *Modeling and analysis of massive social networks.* Ph.D. Dissertation. UMD.

[43] Y. Wang, C. Chakrabarti, D.and Wang, and C. Faloutsos. 2003. Epidemic spreading in real networks: An eigenvalue viewpoint. In *22nd International Symposium on Reliable Distributed Systems, 2003. Proceedings.* IEEE, 25–34.

[44] Mieneke W Weenig and Cees J Midden. 1991. Communication network influences on information diffusion and persuasion. *Journal of personality and social psychology* (1991).

[45] B. Wilder, L. Onasch-Vera, J. Hudson, J. Luna, N. Wilson, R. Petering, D. Woo, M. Tambe, and E. Rice. 2018. End-to-end influence maximization in the field. In *Proceedings of the 17th International Conference on Autonomous Agents and MultiAgent Systems*. International Foundation for Autonomous Agents and Multiagent Systems, 1414–1422.

[46] A. Yadav, E. Kamar, B. Grosz, and M. Tambe. 2016. Healer: Pomdp planning for scheduling interventions among homeless youth. In *Proceedings of the 2016 International Conference on Autonomous Agents & Multiagent Systems*. International Foundation for Autonomous Agents and Multiagent Systems, 1504–1506.

[47] Y. Zhang and B. A. Prakash. 2015. Data-aware vaccine allocation over large networks. *ACM Transactions on Knowledge Discovery from Data (TKDD)* 10, 2 (2015), 20.