Title	Infection Spreading and Source Identification: A Hide and Seek Game
Author(s)	Luo, Wuqiong; Tay, Wee Peng; Leng, Mei
Citation	Luo, W., Tay, W. P., & Leng, M. (2016). Infection Spreading and Source Identification: A Hide and Seek Game. IEEE Transactions on Signal Processing, 64(16), 4228-4243.
Date	2016
URL	http://hdl.handle.net/10220/43480
Rights	© 2016 IEEE. Personal use of this material is permitted. Permission from IEEE must be obtained for all other uses, in any current or future media, including reprinting/republishing this material for advertising or promotional purposes, creating new collective works, for resale or redistribution to servers or lists, or reuse of any copyrighted component of this work in other works. The published version is available at: [http://dx.doi.org/10.1109/TSP.2016.2558168].

Infection Spreading and Source Identification: A Hide and Seek Game

Wuqiong Luo, *Member, IEEE*, Wee Peng Tay, *Senior Member, IEEE* and Mei Leng, *Member, IEEE*

Abstract

The goal of an infection source node (e.g., a rumor or computer virus source) in a network is to spread its infection to as many nodes as possible, while remaining hidden from the network administrator. On the other hand, the network administrator aims to identify the source node based on knowledge of which nodes have been infected. We model the infection spreading and source identification problem as a strategic game, where the infection source and the network administrator are the two players. As the Jordan center estimator is a minimax source estimator that has been shown to be robust in recent works, we assume that the network administrator utilizes a source estimation strategy that can probe any nodes within a given radius of the Jordan center. Given any estimation strategy, we design a best-response infection strategy for the source. Given any infection strategy, we design a best-response estimation strategy for the network administrator. We derive conditions under which a Nash equilibrium of the strategic game exists. Simulations in both synthetic and real-world networks demonstrate that our proposed infection strategy infects more nodes while maintaining the same safety margin between the true source node and the Jordan center source estimator.

Index Terms

Infection source, rumor source, source identification, infection spreading, Jordan center, social network.

I. Introduction

With the increasing popularity of online social networks like Facebook, Twitter and Google+ [1]–[4], more and more people are getting news and information via social networks instead of traditional media outlets. According to a study from the Pew Research Center, about 30% of Americans now get news from Facebook [5]. Due to the interactive nature of online social networks, instead of passively consuming news, half of social networks users

A preliminary version of this paper has been presented at the 2015 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining. This work was supported in part by the Singapore Ministry of Education Academic Research Fund Tier 2 grants MOE2013-T2-2-006 and MOE2014-T2-1-028. W. Luo was with the Nanyang Technological University, Singapore, and is currently with Micron Semiconductor Asia. W. P. Tay is with the Nanyang Technological University, Singapore. M. Leng is with the Temasek Laboratories@NTU, Singapore. E-mail: wluo1@e.ntu. edu.sg, wptay@ntu.edu.sg, lengmei@ntu.edu.sg.

actively share or repost news stories, images or video, and 46% of them discuss news issues within their social circles [5]–[8]. As a result, a piece of information or a rumor posted by a social network user can be reposted by other users and spread quickly on the underlying social network and reach a large number of users in a short period of time [9]–[11]. We say that such users or nodes in the network are "infected". A widely spread rumor or misinformation can lead to reputation damage [12], political consequences [13], and economic damage [14]. The network administrator may want to identify the rumor source in order to catch the culprit, control the damage, and counter the rumor influence. Here, the term "network administrator" is used in a very broad sense to include anyone (e.g., regulatory authorities and researchers) who has been given access to data about the network topology and infected nodes.

Another example of an infection spreading is that of a malicious node in a computer network whose goal is to spread a virus throughout the network. The virus can be a spam bot that is not easily detected [15] (e.g., when the Mariposa botnet was dismantled in 2009, it had infected over 8 million computers [16]), and the network administrator is alerted to the virus infection only at a much later time. Motivated by these applications, many recent research works [17]–[22] have focused on the problem of identifying rumor or infection sources in a network under various spreading models. In all these works, the source is assumed to be "dumb", and whether a susceptible node becomes infected or not follows a stochastic process that is not controlled by the source. Under this simplified assumption, the works [17]–[22] show that source estimators can be constructed so that the true source can be identified with high probability to within a fixed number of hops.

In many applications, the source may wish to maintain anonymity while spreading the infection to as many users as possible. An example is the now defunct anonymous social networking app Secret [23], which allowed smart phone users to share information and repost a posting anonymously among his device contacts or Facebook friends. In February 2014, Secret was used to spread the false rumor that Evernote Corporation was going to be acquired, which prompted the CEO to subsequently issued a public denial [24]. Messaging services including Wickr [25] and FireChat [26] have been used in civil protests like those in Hong Kong in 2014 [27], [28]. Government authorities may trace the initiators of certain protest events even if the messages are encrypted through the use of source identification algorithms that do not rely on message content or metadata [17]–[21]. Therefore, in distributing information, civil protest leaders may design an infection strategy that carefully controls the rate of information spreading in order to obfuscate their identities. In the example of spam bot infection spreading, the perpetrator also wants an infection strategy that controls the rate of the virus spreading to avoid being caught by the authorities while spreading the virus to as many computers as possible. The recent work [29] introduces a messaging protocol, which guarantees obfuscation of the source under the assumption that the network administrator utilizes the maximum likelihood (ML) estimator to identify the source, and when the underlying network is an infinite regular tree. Moreover, simulations are provided in [29] to verify the performance of the messaging protocol on irregular trees and general networks.

With prior knowledge that the infection source may try to avoid detection, the network administrator needs to

adapt its estimation strategy to increase its chance of identifying the infection source. On the other hand, if the source has prior knowledge of the estimation strategy, it needs to further adapt its own infection strategy, and so on. The source and network administrator is thus playing a "hide and seek" game of infection spreading and source identification. This complex dynamic can be modeled as a strategic game with the source and network administrator as the two players of the game. To the best of our knowledge, studying infection spreading and source identification as a strategic game is novel since previous works like [17]–[22] focus only on the estimation strategy, while the work [29] focuses only on the infection strategy.

In this paper, we study best-response strategies for *both* the source and network administrator for trees from a game theoretic perspective, whereas [29] develops an *order-optimal* infection strategy for the source for infinite regular trees (which are special cases of expanding trees), and extends heuristically to more general networks. In [29], the network administrator is assumed to adopt the ML estimation strategy, whereas we assume that the network administrator is allowed to tune a Jordan center based estimation strategy (see Section II-A for justifications of our estimation strategy choice). In our current work, we assume that the network administrator becomes aware of the infection only when the number of infected nodes exceeds a given threshold, and only then it makes an observation of the infection status of the nodes. (The problem becomes trivial if the network administrator is constantly monitoring all the nodes in the network.) For example, a perpetrator who aims to manipulate the stock price of a company may start to spread a false rumor about the company on a social network. The regulatory authority do not have enough resources to monitor the whole network all the time and for all possible events. Therefore, it becomes aware of the false rumor only when the number of infected nodes becomes sufficiently large. Our work can also be applied to the case where the source has an estimate of when the network administrator discovers the infection. This can be the case in the previous stock price manipulation example when the perpetrator first spreads a rumor within his social network using private messages to collude with other users, and then all colluding users post the rumor publicly on the stock's initial public offering day in order to manipulate its price and profit from it.

Our main contributions are the following:

(i) We formulate a strategic game in which the network administrator and infection source are the players. The network administrator uses a source estimator in which it can probe any nodes within a given radius of a randomly chosen Jordan center of the observed infection graph. A larger probe or estimation radius ensures that the source can be identified but incurs a higher cost. The infection source uses an infection strategy in which the rate of infection over each edge in the network can be controlled in order to achieve a minimum safety margin to the Jordan centers. The source is rewarded for each infected node, and penalized if it is identified by the network administrator.

(ii) Given a safety margin for the infection source, we show that the best-response strategy for the network administrator is to use the Jordan centers as the source estimator or adopt an estimation radius equal to the safety margin. We derive conditions under which each of these strategies are optimal.

(iii) Given an estimation radius for the network administrator, we show that the optimal safety margin for the infection source when the underlying network is a tree, is either zero or one more than the estimation radius. We derive an infection strategy, called the Dominant Infection Strategy (DIS), which maximizes the number of infected nodes subject to a given safety margin.

(iv) We derive conditions under which a Nash equilibrium for the strategic game in (i) exists. We show that when a Nash equilibrium exists, the best response for the network administrator is to adopt the Jordan center estimator. This gives a game-theoretic interpretation to the Jordan center estimator, in addition to being a universally robust estimator (which we showed previously in [22]).

Our problem of finding the best-response infection strategy is related to the influence maximization problem, which aims to find a subset of influential nodes to maximize the expected number of nodes that are "influenced" or infected by the chosen subset [30], [31], and is shown by [32] to be a NP-hard optimization problem. Approximate solutions have been extensively investigated by various researchers [33]–[35]. The main difference between our work and the influence maximization problem is that the source node in our problem is fixed, and we seek an infection strategy, given any source node, to maximize the set of infected nodes, subject to a safety margin to the Jordan center of the infection graph.

The rest of this paper is organized as follows. In Section II, we present the system model, assumptions and provide a game-theoretic problem formulation. In Section III, we show the best-response estimation strategy for the network administrator given any infection strategy. In Section IV, we propose a best-response infection strategy for the source given any estimation strategy. In Section V, we derive conditions under which a Nash equilibrium of the strategic game exists. We present simulation results in Section VI to evaluate the effectiveness of the proposed strategies on various synthetic and real networks. Finally we conclude and summarize in Section VII.

II. Problem Formulation

In this section, we first describe our system model and assumptions, and then we provide a game-theoretic problem formulation for the infection spreading and source identification.

Consider an undirected graph $G(V, E)$ representing a social network, where V is the set of vertices or nodes, and E is the set of edges. Because of technical difficulties, our analysis and strategy design assume that G is a tree, as is commonly done in the literature [17]–[21], [29]. We will however apply our strategies heuristically to general networks in our simulations in Section VI.

We assume that there is a single source node $v^* \in V$ at time 0. An infection can pass from one node to another. For example in the case of an online social network, a user may post a rumor he sees in the posting of his friend using his own account. An infected node remains infected throughout, and has the capability of infecting its neighbors at a deterministic rate. For any edge $(i, j) \in E$, we let $\mu(i, j)$ to be the time it takes for an infected node i to infect its susceptible neighbor j, which we call the infection time associated with the edge (i, j). Let $\lambda(i, j) = 1/\mu(i, j)$ be the infection rate of (i, j). For any pair of nodes v and u in G, let $d(v, u)$ be the number

of hops in the shortest path between v and u, which is also called the *distance* between v and u. For any edge (i,j) with $d(v^*,i) = m$ and $d(v^*,j) = m+1$, we assume $\lambda(i,j)$ is uniformly upper bounded by a maximum infection rate $\bar{\lambda}_m > 0$. In examples like rumor spreading, $\bar{\lambda}_m$ is non-increasing in m as it becomes more difficult for an infected node further away from the source to infect another susceptible node. We assume that the network administrator observes one snapshot of all the infected nodes in the network, and tries to estimate the source at the time t_{obs} when the number of infected nodes first exceeds a threshold $n_{\text{obs}} > 1$. We call t_{obs} the observation time, and n_{obs} the observation threshold.

For any time $t > 0$, let

$$\bar{d}(t) = \max\left\{k : \sum_{m=0}^{k-1} \bar{\lambda}_m^{-1} \leq t\right\}. \tag{1}$$

Since the infection rate of each edge is upper bounded by its respective $\bar{\lambda}_m$, the maximum number of hops the infection can spread from the source in time t is $\bar{d}(t)$. We assume that the graph is sufficiently large so that $\bar{d}(t_{\text{obs}}) \leq \bar{d}(v^*, V)$, i.e., the network administrator observes the infection graph before the infection can spread to all nodes in the network.

For any pair of nodes v and u in G, let $\rho(v,u)$ to be the shortest path from v to u, and the infection time of $\rho(v,u)$ to be

$$\begin{aligned}\mu(v,u) &= \sum_{(i,j)\in\rho(v,u)} \mu(i,j) \\ &= \sum_{(i,j)\in\rho(v,u)} \frac{1}{\lambda(i,j)}.\end{aligned} \tag{2}$$

The collection of infection rates $\Lambda = \{\lambda(i,j) : (i,j) \in E\}$ is called an infection strategy for the source node. Given any infection strategy Λ, we denote the set of infected nodes at time t_{obs} as

$$V_{\text{I}} = \{u \in G : \mu(v^*,u) \leq t_{\text{obs}}\}. \tag{3}$$

We sometimes use $V_{\text{I}}(\Lambda)$ instead of V_{I} to indicate that the given set of infected nodes resulted from the infection strategy Λ. Let G_{I} to be the minimum connected subgraph of G that spans V_{I}, which we call the *infection graph* at time t_{obs}.

Throughout this paper, we let $|X|$ denote the *expected* number of nodes in the random set X conditioned on the infection graph, and $\mathbf{1}_A$ denote an indicator function with value 1 iff the clause A is true.

A. Network Administrator

At the observation time t_{obs}, the network administrator observes the infection graph G_{I}, and tries to estimate the infection source. Although n_{obs} is known to the network administrator, since it does not know the starting time that the source begins its infection spreading, it does not know the amount of elapsed time t_{obs}. We assume that the network administrator can choose a subset of nodes to investigate, which we call the suspect set. It is important

for the network administrator to decide which subset of nodes to investigate in order to minimize the cost and maximize its chance of identifying the infection source. In the same spirit as [17]–[21], the network administrator is assumed to have limited knowledge of the underlying infection spreading process, and its estimation strategy can only depend on the observed infection graph G_{I}. In the following, we present the definition of the Jordan center and then introduce a class of estimation strategies based on the Jordan center.

Given any set $A \subset V$, denote the largest distance between v and any node $u \in A$ to be

$$\bar{d}(v, A) = \max_{u \in A} d(v, u).$$

For any infection strategy Λ, we call the largest distance $\bar{d}(v, V_{\mathrm{I}})$ between v and any infected node the *infection range* of v. We let

$$\mathrm{JC} = \{v : \bar{d}(v, V_{\mathrm{I}}) = \min_{u \in G} \bar{d}(u, V_{\mathrm{I}})\}, \tag{4}$$

to be the set of nodes with minimum infection range, which are known as the *Jordan centers* of G_{I} [36]. It is shown in [20] that if G is a tree, then $|\mathrm{JC}| \leq 2$.

When no prior knowledge of the infection source is available, any node in V_{I} is equally likely to be the source as infection rates over different edges can be heterogeneous. Therefore, a Jordan center is a minimax source estimator that minimizes the largest distance to any infected node. It has also been shown in [20]–[22] that the Jordan center is a robust source estimate. Another popular estimator is the ML estimator. However, [29] shows that it is possible to design an infection strategy so that the probability of the ML estimator being the true source is approximately $1/|V_{\mathrm{I}}|$, i.e., all the infected nodes are considered by the network administrator to be approximately equally likely to be the source. If t_{obs} is large, then the ML estimator performs badly. The Jordan center estimator does not have this problem since there are at most two Jordan centers in any tree. As such, we assume that the network administrator chooses the suspect set $V_{\mathrm{sp}}(d_a)$ to be the set of infected nodes within $d_a \geq 0$ hops from an arbitrarily chosen Jordan center u, i.e.,

$$V_{\mathrm{sp}}(d_a) = \{v \in V_{\mathrm{I}} : d(v, u) \leq d_a\}. \tag{5}$$

We call d_a the *estimation radius*. Note that $V_{\mathrm{sp}}(d_a)$ depends only on the observed infection graph V_{I}. The strategy of the network administrator is denoted using d_a.

We let d_s to be the distance between the actual source and the Jordan centers, i.e.,

$$d_s(\Lambda) = \min_{u \in \mathrm{JC}} d(v^*, u). \tag{6}$$

We call $d_s(\Lambda)$ the *safety margin* of the source achieved by the infection strategy Λ.

If $d_a \geq d_s(\Lambda)$, then the infection source is in the suspect set, and the network administrator has a non-negative probability of identifying the source. The network administrator obtains an expected gain $g_a(d_a, V_{\mathrm{I}})$, which we assume to be non-increasing in d_a. For example, if the network administrator only has access to the infection graph and has no additional prior information, its best strategy is to uniformly choose a node from the set of suspects as

the estimated source node. Its expected reward is then inversely proportional to the number of nodes in $V_{\text{sp}}(d_a)$. In another application, the network administrator may have side information that allows it to always correctly identify the source node if it is included in the suspect set. In this case, we let the expected reward to be $g_a(d_a, V_{\text{I}}) = g_a$. Some positive cost $c_a(V_{\text{sp}}(d_a))$ is incurred for probing nodes in $V_{\text{sp}}(d_a)$. We assume that $c_a(V_{\text{sp}}(d_a))$ is an increasing function of the estimation radius d_a. We let the utility function of the network administrator be

$$u_a(d_a, \Lambda) = -c_a(V_{\text{sp}}(d_a)) + g_a(d_a, V_{\text{I}})\mathbf{1}_{d_a \geq d_s(\Lambda)}. \tag{7}$$

The network administrator's utility function depends on Λ only through its infection graph V_{I}, which the administrator observes at time t_{obs}, and the safety margin $d_s(\Lambda)$. Although the network administrator's utility function depends on the safety margin of the source, it does not know a priori the safety margin chosen by the source. In this paper we perform a game theoretic analysis of the estimation strategy used.

B. Infection Source

Suppose that the network administrator uses an estimation radius of $d_a \geq 0$. We assume that the observation time t_{obs} is unknown to the infection source, but the source knows the infection threshold n_{obs} at which the network administrator will attempt to identify it. We show in Section IV that t_{obs} can be computed from n_{obs}. For each node that is infected, the source is rewarded with a gain g_s. A positive cost $c_s(d_a)$ is incurred if it falls within the suspect set of the network administrator. We assume that $c_s(d_a)$ is a non-decreasing function of the estimation radius d_a. The utility function of the source adopting the infection strategy Λ is given by

$$u_s(d_a, \Lambda) = g_s|V_{\text{I}}| - c_s(d_a)\mathbf{1}_{d_a \geq d_s(\Lambda)}. \tag{8}$$

C. Strategic Game

We model the infection spreading and source identification as a strategic game where the network administrator and the infection source are the two players. The utility functions of the two players are given in (7) and (8), respectively. Given any infection strategy Λ (or more specifically, the safety margin $d_s(\Lambda)$), the network administrator finds the *best-response estimation strategy* with estimation radius d_a^* that maximizes its utility function, i.e.,

$$d_a^* = \arg\max_{d_a} u_a(d_a, \Lambda).$$

On the other hand, given any estimation radius d_a, the infection source finds the *best-response infection strategy* Λ^* that maximizes its utility function, i.e.,

$$\Lambda^* = \arg\max_{\Lambda} u_s(d_a, \Lambda).$$

If there exists a pair (d_a^*, Λ^*) such that given Λ^*, the best-response estimation strategy has estimation radius d_a^*; and given d_a^*, the best-response infection strategy is Λ^*, then (d_a^*, Λ^*) is a Nash equilibrium of the strategic game [37].

In Sections III and IV, we find the best-response estimation strategy and the best-response infection strategy for the network administrator and the infection source, respectively. In Section V, we derive the Nash equilibrium of the strategic game.

III. Best-response Estimation Strategy for the Network Administrator

In this section, we derive the best-response estimation strategy for the network administrator. In this paper, we assume that the network administrator utilizes the Jordan center based estimation strategy, which is characterized by the estimation radius d_a. Given the infection strategy Λ with safety margin $d_s(\Lambda) = d_s$, the network administrator chooses an optimal estimation radius to maximize its utility function.

We first consider the case where $d_s = 0$. In this case, the inequality $d_a \geq d_s$ always holds. From (7), the network administrator's utility function becomes

$$u_a(d_a, \Lambda) = -c_a(V_{\text{sp}}(d_a)) + g_a(d_a, V_{\text{I}}),$$

which is decreasing in d_a. Therefore, the optimal estimation radius is given by $d_a = 0$.

Now suppose that $d_s > 0$. We claim that the estimation radius of a best-response estimation strategy is either 0 or d_s. To prove this claim, it suffices to show the following inequalities:

$$u_a(0, \Lambda) > u_a(d'_a, \Lambda), \quad \forall\ d'_a \in (0, d_s); \tag{9}$$

$$u_a(d_s, \Lambda) > u_a(d'_a, \Lambda), \quad \forall\ d'_a \geq d_s + 1. \tag{10}$$

We first show the inequality (9). Since $0 < d'_a < d_s$, the gains for both the estimation strategy with $d_a = 0$ and the estimation strategy with $d_a = d'_a$ are 0. The inequality (9) then holds because $c_a(V_{\text{sp}}(0)) < c_a(V_{\text{sp}}(d'_a))$. We next show the inequality (10). The inequality $d_a \geq d_s$ holds for both the estimation strategy with $d_a = d_s$ and the estimation strategy with $d_a = d'_a \geq d_s + 1$. Since $c_a(V_{\text{sp}}(d_s)) < c_a(V_{\text{sp}}(d'_a))$ and $g_a(d_s, V_{\text{I}}) \geq g_a(d'_a, V_{\text{I}})$, the inequality (10) holds. This completes the proof of the claim. The following theorem then follows immediately.

Theorem 1. *Suppose that the infection source adopts the infection strategy Λ with safety margin $d_s(\Lambda) = d_s$. If $d_s = 0$, the estimation radius d_a^* of the best-response estimation strategy is 0. If $d_s > 0$, the estimation radius of the best-response estimation strategy is given by*

$$d_a^* = \begin{cases} 0, & \text{if } g_a(d_s, V_{\text{I}}) < c_a(V_{\text{sp}}(d_s)) - c_a(V_{\text{sp}}(0)), \\ d_s, & \text{if } g_a(d_s, V_{\text{I}}) > c_a(V_{\text{sp}}(d_s)) - c_a(V_{\text{sp}}(0)), \\ 0 \text{ or } d_s, & \text{if } g_a(d_s, V_{\text{I}}) = c_a(V_{\text{sp}}(d_s)) - c_a(V_{\text{sp}}(0)). \end{cases} \tag{11}$$

We remind the reader that the quantities $g_a(d_s, V_{\text{I}})$ and $V_{\text{sp}}(d_s)$ in Theorem 1 depend on the infection strategy Λ only through the infection graph V_{I} observed by the network administrator. Therefore, given the safety margin d_s, the network administrator can formulate its best response using Theorem 1 without knowing the source utility function.

We observe that if $d_s > 0$ in Theorem 1, then using an estimation radius of $d_a^* = 0$ implies that the network administrator has zero probability of identifying the infection source. This happens when the reward of catching the infection source is significantly lower than the cost of probing more nodes. In practical systems, attempts should be made to keep the cost of probing each node in the network sufficiently small so that the infection source can be identified with positive probability. On the other hand, our result also points to the intuitive conclusion that for a source to escape identification with *probability one*, the infection observation time t_{obs} must be sufficiently long (cf. Theorem 2), and the source's safety margin must be chosen to be sufficiently large so that $c_a(V_{\text{sp}}(d_s)) - c_a(V_{\text{sp}}(0))$ is large and the first case in (11) holds.

IV. Best-response Infection Strategy for the Infection Source in a Tree

In this section, we derive a best-response infection strategy for the infection source for the case where the underlying graph G is a tree. We assume that the infection source knows the observation threshold n_{obs} but not the observation time t_{obs}. We first derive our infection strategy based on the observation time t_{obs}, and show how to compute t_{obs} from n_{obs}.

Given an estimation radius d_a and an observation time t, the source designs a best-response infection strategy that maximizes its utility function. We first introduce the notion of a *maximum infection strategy*. Let $\|\Lambda\|$ denote the infection size of the infection strategy Λ.

Definition 1. *Given any safety margin d_s and observation time t, we define the maximum infection strategy with safety margin d_s to be the infection strategy that maximizes the number of infected nodes at time t among all infection strategies that achieve the safety margin d_s. Let the set of maximum infection strategies with safety margin d_s be $\mathcal{M}(d_s, t)$.*

For each estimation radius d_a, we design a best-response infection strategy Λ^* for the infection source in three steps:

- Step 1: Given any safety margin d_s and any observation time t, we find a maximum infection strategy $\Lambda_{d_s,t} \in \mathcal{M}(d_s, t)$.
- Step 2: We search for the smallest t such that $\|\Lambda_{d_s,t}\| \geq n_{\text{obs}}$, and set $t_{\text{obs}} = t$.
- Step 3: Among all maximum infection strategies found in Step 2, we find one that maximizes the source's utility function as the best-response infection strategy, i.e., an infection strategy $\Lambda^* = \Lambda_{d_s^*, t_{\text{obs}}}$ where

$$d_s^* = \arg\max_{d_s} u_s(d_a, \Lambda_{d_s, t_{\text{obs}}}).$$

Note that under a given safety margin constraint d_s and observation time t, the source's utility $u_s(\cdot, \cdot)$ is invariant to which infection strategy is chosen from $\mathcal{M}(d_s, t)$. In the following, we first determine the range of values that the safety margin d_s can take for given set of maximum infection rates $\{\bar{\lambda}_m\}$, and observation time t, i.e., those values of d_s such that $\mathcal{M}(d_s, t) \neq \emptyset$. We call such a safety margin *feasible*. It is clear that $\mathcal{M}(0, t) = \{\Lambda_{\max}\}$, where $\Lambda_{\max}$

is the infection strategy in which each node at distance m from the source is infected at its respective maximum rate $\bar{\lambda}_{m-1}$. We next propose an algorithm to find an infection strategy in $\mathcal{M}(d_s, t)$, for all feasible $d_s > 0$.

A. Maximum Infection Strategy with Safety Margin Constraint

Given any observation time t, it turns out that not all values of d_s are feasible. To see this, consider an infection spreading along a linear network. Since the maximum number of hops the infection can spread from the source is $\bar{d}(t)$ (cf. (1)), the safety margin cannot be more than $\lfloor \bar{d}(t)/2 \rfloor$. The following theorem provides the achievable upper bound for the safety margin in a tree. The proof is in Appendix A.

Theorem 2. *Suppose the underlying graph G is a tree. For any given observation time t, the largest feasible safety margin is*

$$\bar{d}_s = \left\lfloor \frac{\bar{d}(t)}{2} \right\rfloor. \tag{12}$$

From Theorem 2, we see that the safety margin of any infection strategy can take values only from the set $[0, \bar{d}_s]$. Given a feasible safety margin d_s, we next show how to design a maximum infection strategy $\Lambda_{d_s,t} \in \mathcal{M}(d_s, t)$. In the rest of this section, we adopt the following notations: Let $G_{\mathrm{I},t}$ be the infection graph at observation time t generated by a given infection strategy, which will be clear from the context. Let T_u to be the subtree of $G_{\mathrm{I},t}$ rooted at node u with the first link in the path from u to the source node v^* removed. Let l_u to be a leaf node in T_u that has maximum distance from u, i.e.,

$$l_u \in \{i \in T_u : d(u, i) = \max_{j \in T_u} d(u, j)\}. \tag{13}$$

We start by defining a dominant path and showing an elementary result related to this definition.

Definition 2. *For any observation time t, a dominant path is a path between the source v^* and any node in the infection graph $G_{\mathrm{I},t}$ that has the maximum distance.*

Lemma 1. *Suppose that G is a tree. Consider a maximum infection strategy with a feasible safety margin d_s that results in an infection graph $G_{\mathrm{I},t}$ at time t. Then, the infection along each edge in any dominant path of $G_{\mathrm{I},t}$ has maximum infection rate $\bar{\lambda}_m$ if the endpoint of the edge closer to the source is at distance m from it.*

The proof of Lemma 1 is provided in Appendix B. To get a safety margin $d_s > 0$, the intuition is to construct one dominant path P_d starting at v^* so that the Jordan center is biased towards the leaf node at the other end of P_d, which in turn results in a safety margin d_s. We discuss how to select an optimal dominant path in Algorithm 1. For now, we assume that a dominant path P_d is given. Our proposed DIS(d_s, t) strategy, given in Strategy 1, is defined by a set of parameters, $\{\lambda_m : m \in [0, \bar{d}(t) - 1]\}$. Consider any infected node i on the path P_d. Suppose $d(v^*, i) = m$ and let j be the susceptible neighboring node of i on P_d. The node i infects j with rate $\bar{\lambda}_m$ and infects all its other susceptible neighbors with rate λ_m. On the other hand, for any infected node that is not on P_d, it infects all its susceptible neighbors with the same rate that it itself was previously infected.

Strategy 1 Dominant Infection Strategy DIS(d_s, t)

Inputs: $G = (V, E)$, source v^*, observation time t, required safety margin d_s, set of maximum infection rates $\{\bar{\lambda}_m\}$, infection rates $\{\lambda_{m,j}\}$ given in (22), and a dominant path P_d found by Algorithm 1 below.
Output: $\lambda(u, v)$, the infection rate for every $(u, v) \in G_{\mathrm{I},t}$.
for each $u \in V$ **do**
  if $u \in P_d$ **then**
    Let $m = d(v^*, i)$.
    If v is the susceptible neighbor of u on P_d, set $\lambda(u, v) = \bar{\lambda}_m$.
    For any other susceptible neighbor v of u not on P_d, set $\lambda(u, v) = \lambda^{\mathrm{DIS}}_{m,j}$ where $j = d(u, v)$, and pass the message $(m, 1)$ to v.
  else
    Let (a_u, b_u) be the message received by u. For any susceptible neighbor v of u, set $\lambda(u, v) = \lambda^{\mathrm{DIS}}_{a_u,b_u}$. Pass the message $(a_u, b_u + 1)$ to v.
  end if
end for
return $\{\lambda(u, v) : (u, v) \in G_{\mathrm{I},t}\}$

In the following discussion, we show that if the parameter λ_m is set to be that in (22), and the dominant path is selected by Algorithm 1, then DIS$(d_s, t) \in \mathcal{M}(d_s, t)$.

Suppose that v^* has k neighbors $v_1, v_2, \cdots, v_k$ in G_{I}, where $k \geq 2$. Without loss of generality, suppose that the labels are assigned so that $d(v, l_{v_1}) \geq d(v, l_{v_2}) \geq \ldots \geq d(v, l_{v_k})$. We have the following elementary result, the proof of which is provided in Appendix C.

Lemma 2. *Suppose that G is a tree. Given an infection graph $G_{\mathrm{I},t}$, if $d(v^*, l_{v_1}) > d(v^*, l_{v_2})$, then at least one end of any diameter of $G_{\mathrm{I},t}$ is in the subtree T_{v_1}.*

We now show how to find the optimal parameter λ_m in DIS(d_s, t) so that it achieves a safety margin d_s. From Lemma 1, we have

$$d(v^*, l_{v_1}) = \bar{d}(t). \tag{14}$$

Let l_{v_1} be a leaf node so that $\rho(v^*, l_{v_1}) = (u_0 = v^*, u_1, \cdots, u_{\bar{d}(t)} = l_{v_1})$ is the given dominant path P_d. For $0 \leq m \leq d(v^*, l_{v_1}) - 1$, define $\tilde{T}_{u_m}$ to be $T_{u_m} \backslash T_{u_{m+1}}$. Similar to the definition in (13), let $\tilde{l}_{u_m}$ be a leaf node in $\tilde{T}_{u_m}$ that has maximum distance from u_m. Figure 1 shows an illustration of $G_{\mathrm{I},t}$ and $\tilde{T}_{u_m}$.

For each $m \geq 1$, let

$$t_m = \sum_{k=0}^{m-1} \bar{\lambda}_k^{-1} \tag{15}$$

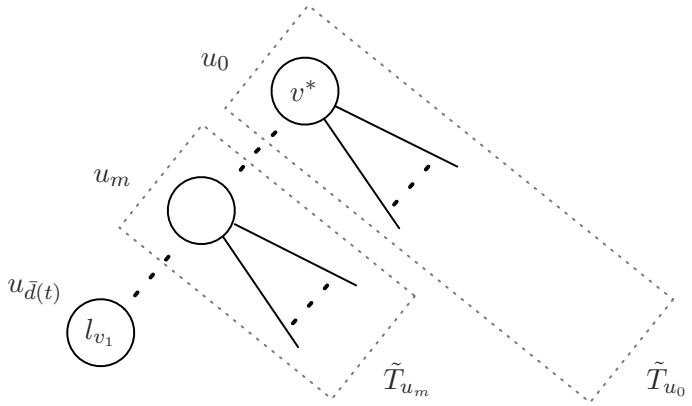


Fig. 1. Illustration of the infection graph G_{I}.

be the time taken to infect node u_m. Suppose the nodes in $\rho(u_m, \tilde{l}_{u_m})$ are infected at an *average* rate λ_m with

$$d(u_m, \tilde{l}_{u_m}) = \lfloor \lambda_m (t - t_m) \rfloor . \tag{16}$$

Let $\mathcal{D}(u_m)$ to be the set of longest paths such that one end of any path $D(u_m) \in \mathcal{D}(u_m)$ is l_{v_1} and the other end is in $\tilde{T}_{u_m}$. From (14) and (16), the number of vertices in $D(u_m)$ is

$$\begin{aligned} |D(u_m)| &= d(l_{v_1}, u_m) + d(u_m, \tilde{l}_{u_m}) + 1 \\ &= d(v^*, l_{v_1}) - d(v^*, u_m) + d(u_m, \tilde{l}_{u_m}) + 1 \\ &= \bar{d}(t) - m + \lfloor \lambda_m (t - t_m) \rfloor + 1. \end{aligned} \tag{17}$$

Let $\hat{v}(u_m)$ to be a node in the middle of $D(u_m)$, i.e., $\hat{v}(u_m) \in \arg\min_{v \in D(u_m)} \bar{d}(v, D(u_m))$. Since the infection is propagated at the maximum rates along P_d, we can always choose $\hat{v}(u_m) \in P_d$ with

$$\begin{aligned} & d(v^*, \hat{v}(u_m)) \\ &= d(v^*, l_{v_1}) - d(\hat{v}(u_m), l_{v_1}) \\ &= d(v^*, l_{v_1}) - \left(\left\lceil \frac{|D(u_m)|}{2} \right\rceil - 1 \right). \end{aligned} \tag{18}$$

In order to maximize the number of infected nodes, we maximize $|D(u_m)|$. Note that $|D(u_m)|$ is odd, because otherwise we can always increase λ_m so that we have one more node in $|D(u_m)|$, but (18) remains the same. Then we have

$$\begin{aligned} & d(v^*, \hat{v}(u_m)) \\ &= d(v^*, l_{v_1}) - \left(\frac{|D(u_m)|}{2} + \frac{1}{2} - 1 \right) \\ &= \frac{1}{2} \left(\bar{d}(t) + m - \lfloor \lambda_m (t - t_m) \rfloor \right). \end{aligned} \tag{19}$$

Since $d_s > 0$, we must have $d(v^*, l_{v_1}) > d(v^*, l_{v_2})$. From Lemma 2, we see that one end of any diameter is a l_{v_1}. Then any diameter is in the set of paths $\bigcup_{m=0}^{\bar{d}(t)-1} \mathcal{D}(u_m)$ and the set of Jordan centers is a subset of

$\{\hat{v}(u_m) : D(u_m) \in \bigcup_{m=0}^{\bar{d}(t)-1} \mathcal{D}(u_m)\}$. The safety margin requirement d_s is satisfied if the right hand side of (19) has value at least d_s for every $m \in [0, d_s]$, i.e.,

$$\lfloor \lambda_m(t - t_m) \rfloor \leq h_m \triangleq \bar{d}(t) - 2d_s + m. \tag{20}$$

To maximize the number of infected nodes at time t, we choose each λ_m to be as large as possible. Therefore, for $m \in [0, d_s]$, we choose λ_m to be the largest value so that equality holds in (20), i.e.,

$$\lambda_m = \frac{h_m}{t - t_m}. \tag{21}$$

We then find $\delta_{t,m} \geq 0$ for $j \in [0, h_m]$ such that

$$\sum_{j \in A_m} \left(\bar{\lambda}_{m+j} - \delta_{t,m}\right)^{-1} = t - t_m - \sum_{j \notin A_m} \bar{\lambda}_{m+j}^{-1},$$

where $A_m = \{j \leq h_m : \bar{\lambda}_{m+j} > \lambda_m\}$. Such a $\delta_{t,m}$ exists because $h_m \leq \bar{d}(t) - m$, the longest distance the infection can propagate from u_m. Finally, we let the infection at each node $v \in \tilde{T}_{u_m}$ such that $d(u_m, v) = j \leq h_m$ spread at rate $\bar{\lambda}_{m+j} - \delta_{t,m}$. If $m \in (d_s, \bar{d}(t))$, we choose $\bar{\lambda}_{m+j}$ to be the spreading rate for all $v \in \tilde{T}_{u_m}$ such that $d(u_m, v) = j$. Note that with this choice, $\hat{v}(u_m)$ is the Jordan center on P_d for all $m \in [0, d_s]$.

In summary, for a $v \in \tilde{T}_{u_m}$ such that $d(u_m, v) = j$, we let it infect its susceptible neighbors not on the dominant path with rate

$$\lambda_{m,j}^{\text{DIS}} = \begin{cases} \bar{\lambda}_{m+j} - \delta_{t,m}, & \text{if } j \in A_m, \text{ and } 0 \leq m \leq d_s, \\ \bar{\lambda}_{m+j}, & \text{otherwise.} \end{cases} \tag{22}$$

We have the following result. The proof is provided in Appendix D.

Lemma 3. *Suppose that* $\text{DIS}(d_s, t)$ *with safety margin* $d_s > 0$ *has the dominant path* P_d*. Then, it maximizes the number of infected nodes at time* t *amongst all infection strategies with safety margin* d_s *and dominant path* P_d*.*

In the following, we show how to find the optimal dominant path. Given any dominant path $P_d = (u_0 = v^*, \ldots, u_{\bar{d}(t)})$ as an input of $\text{DIS}(d_s, t)$ with safety margin $d_s > 0$, we have from (22), that the number of infected nodes is given by

$$\bar{d}(t) + \sum_{m=0}^{\bar{d}(t)-1} |\tilde{T}_{u_m}|. \tag{23}$$

To find an optimal dominant path so that the above sum is maximized, we use the procedure in Algorithm 1.

In Algorithm 1, since the weight we have assigned to each edge (u_m, u_{m+1}) in the maximal weight path found corresponds exactly to $|\tilde{T}_{u_m}|$ in (23), the algorithm gives us the optimal dominant path. In the first step of Algorithm 1, the weights $w(u, w)$ for all neighbors w of u in T_u can be found by performing another breadth-first search in the tree T_u. The time complexity of the first step is thus $O(n^2)$,[1] where n is the number of vertices within a distance

[1]A function is said to be $O(f(n))$ if it is upper bounded by $kf(n)$ for some constant $k > 0$ and for all n sufficiently large.

Algorithm 1 Bellman-Ford Dominant Path Finding

1: Perform a breadth-first search starting at v^*, and for each edge (u, w) where $d(v^*, w) = d(v^*, u) + 1 \leq \bar{d}(t)$, assign the following weight:

$$w(u, w) = |\{v \in T_u \backslash T_w : d(v, u) \leq h(u)\}|,$$

with

$$h(u) = \begin{cases} \bar{d}(t) - 2d_s + d(v^*, u), & \text{if } d(v^*, u) \leq d_s, \\ \bar{d}(t) - d(v^*, u), & \text{otherwise.} \end{cases}$$

2: Use the Bellman-Ford algorithm [38] to find a maximal weighted path $(u_0, \ldots, u_{\bar{d}(t)})$ starting at $u_0 = v^*$. The maximal weighted path is output as the dominant path, and its weight added to $\bar{d}(t)$ is output as $\|\text{DIS}(d_s, t)\|$.

$\bar{d}(t)$ of v^* [38]. The Bellman-Ford algorithm in the second step also has time complexity $O(n^2)$. Therefore, the overall time complexity of Algorithm 1 is $O(n^2)$.

Lemma 3 and Algorithm 1 then lead to the following result.

Theorem 3. *Suppose G is a tree. For any observation time t and feasible safety margin d_s,* $\text{DIS}(d_s, t) \in \mathcal{M}(d_s, t)$ *if the dominant path is found by Algorithm 1.*

Since $\|\text{DIS}(d_s, t)\|$ is non-decreasing in t, we can now perform a binary search procedure in Algorithm 2 to determine the smallest t such that $\|\text{DIS}(d_s, t)\| \geq n_{\text{obs}}$. Note that it suffices to perform the binary search over the times $\{t_m : m \geq 1\}$ defined in (15), because for any $t \in [t_m, t_{m+1})$, $m \geq 1$, we have $\|\text{DIS}(d_s, t)\| = \|\text{DIS}(d_s, t_m)\|$ as the right hand side of (20) remains unchanged. Let

$$x_0 = \min\{m : |\{v : d(v^*, v) \leq \bar{d}(t_m)\}| \geq n_{\text{obs}}\}, \tag{24}$$

$$y_0 = \min\{m : |\{v : d(v^*, v) \leq \bar{d}(t_m) - 2d_s\}| \geq n_{\text{obs}}\}. \tag{25}$$

We initialize the search to be over $[t_{x_0}, t_{y_0}]$. Because not all vertices within distance $\bar{d}(t_{\text{obs}})$ are infected by $\text{DIS}(d_s, t_{\text{obs}})$, while all nodes within distance $\bar{d}(t_{\text{obs}}) - 2d_s$ are infected, we have $\|\text{DIS}(d_s, t_{x_0})\| \leq n_{\text{obs}} \leq \|\text{DIS}(d_s, t_{y_0})\|$. The binary search takes at most $O(\log n_{\text{obs}})$ search steps. Assuming that the tree G has bounded degree β, in each search step the computation of $\|\text{DIS}(d_s, t_m)\|$ using Algorithm 1 takes at most $O(\beta^{4d_s} n_{\text{obs}}^2)$ time complexity. Therefore the overall time complexity to find $\text{DIS}(d_s, t_{\text{obs}})$ is $O(\beta^{4d_s} n_{\text{obs}}^2 \log n_{\text{obs}})$.

B. Homogeneous Infection Rate Bounds

If $\bar{\lambda}_m = \bar{\lambda}$ for all $m \geq 0$, then $\bar{d}(t) = \lfloor \bar{\lambda} t \rfloor$, and it can be shown from (22) that for all m and j,

$$\lambda_{m,j}^{\text{DIS}} = \bar{\lambda} \cdot \min\left\{1, \frac{\lfloor \bar{\lambda} t \rfloor - 2d_s + m}{\bar{\lambda} t - m}\right\}, \tag{26}$$

i.e., the same rate is used to infect the vertices in the subtree $\tilde{T}_{u_m}$. In this case, the $\text{DIS}(d_s, t)$ strategy need not pass additional distance information along with the infection.

Algorithm 2 Binary Search for t_{obs}

Initialize $x = x_0$ using (24), and $y = y_0$ using (25).
while $x < y - 1$ **do**
  Set $m = \lceil (x + y)/2 \rceil$.
  if $\|\text{DIS}(d_s, t_m)\| \leq n_{\text{obs}}$ **then**
    Set $x = m$.
  else
    Set $y = m$.
  end if
end while
return Output $t_{\text{obs}} = t_y$.

C. Infinite Regular Trees

In the following, we consider the special case where the underlying network is an infinite r-regular tree, every node has $r > 2$ neighboring nodes, and $\bar{\lambda}_m = 1$ for all $m \geq 0$. The fastest infection strategy $\Lambda_{\max}$ is the one that sets all infection rates to be the upper bound 1. Then the number of nodes infected by the fastest infection strategy by time t can be shown to be given by

$$|V_{\text{I}}(\Lambda_{\max})| = \frac{r(r-1)^t - 2}{r-2}.$$

This infection strategy has safety margin 0. Now suppose that the source wishes to achieve a safety margin $d_s \leq \bar{d}_s$ in (12), the set of infected nodes by time t by our proposed $\text{DIS}(d_s, t)$ strategy can be shown to be all nodes with distance not greater than $t - d_s$ from the Jordan center as shown in Fig. 2. Then, the number of nodes infected by time t is

$$|V_{\text{I}}(\text{DIS}(d_s, t))| = \frac{r(r-1)^{t-d_s} - 2}{r-2}.$$

When d_s increases, the radius $t - d_s$ decreases, and the number of nodes infected by the maximum infection strategy decreases. This is the necessary trade-off between the faster infection spreading speed and the larger safety margin of the source.

The paper [29] proposes a messaging protocol called adaptive diffusion (AD) under the assumption that the network administrator utilizes a ML estimation strategy. AD is a stochastic infection strategy, where its safety margin falls in the range $[1, \lfloor t/2 \rfloor]$ with probability one. Given any safety margin $d_s \in [1, \lfloor t/2 \rfloor]$, with probability one, the set of infected nodes by time t by AD can be shown to be all nodes with distance not greater than $\lfloor t/2 \rfloor$ from the Jordan center as shown in Fig. 2. Then, the number of nodes $|V_{\text{I}}(\text{AD})|$ infected by AD by time t satisfies the following bound with probability one:

$$|V_{\text{I}}(\text{AD})| \leq \frac{r(r-1)^{t-\lfloor t/2 \rfloor} - 2}{r-2} \leq |V_{\text{I}}(\text{DIS}(d_s, t))|.$$

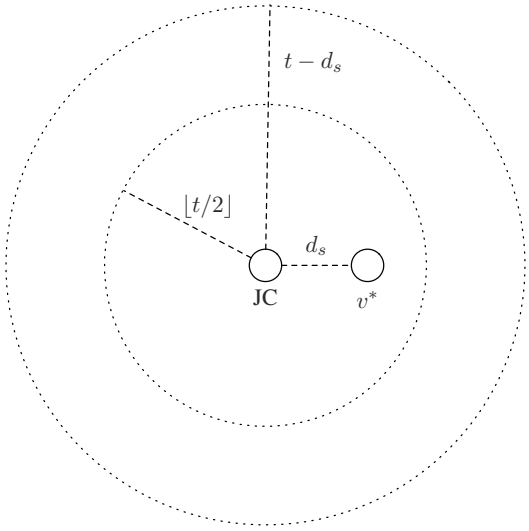


Fig. 2. Illustration of the set of infected nodes by DIS and AD, where JC denotes the Jordan center of the set of infected nodes at time t. The set of infected nodes by DIS are all the nodes within the circle with radius $t - d_s$, while the set of infected nodes by AD are all the nodes within the circle with radius $\lfloor t/2 \rfloor$. Since d_s is upper bounded by $\lfloor t/2 \rfloor$, we have $t - d_s \geq \lfloor t/2 \rfloor$.

Therefore, our proposed DIS strategy infects at least as many nodes as the AD strategy almost surely.

D. Best-response Infection Strategy for the Infection Source

Theorem 3 shows how to find a maximum infection strategy for a feasible d_s. We next identify one that maximizes the utility function of the infection source. We first present the following relationship between maximum infection strategies of different safety margins.

Lemma 4. *Suppose G is a tree. For any two safety margins d_s and d'_s, where $0 \leq d_s < d'_s \leq \bar{d}_s$, any strategy $\Lambda_{d_s} \in \mathcal{M}(d_s, t_{\text{obs}})$ infects more nodes than any strategy $\Lambda_{d'_s} \in \mathcal{M}(d'_s, t_{\text{obs}})$, i.e., $|V_{\text{I}}(\Lambda_{d_s})| > |V_{\text{I}}(\Lambda_{d'_s})|$.*

Proof: From (21), the average infection rate λ_m is a non-increasing function of the safety margin. Since $d_s < d'_s$, the infection rate λ_m in Λ_{d_s} is larger than or equal to that in $\Lambda_{d'_s}$ for any $0 \leq m \leq \bar{d}(t) - 1$. However, equality does not hold for all m, because otherwise, Λ_{d_s} and $\Lambda_{d'_s}$ lead to the same infection graph, which in turn implies that $d_s = d'_s$, a contradiction. As a result, we have $|V_{\text{I}}(\Lambda_{d_s})| > |V_{\text{I}}(\Lambda_{d'_s})|$, which completes the proof of Lemma 4. ■

We now derive the best-response infection strategy based on Lemma 4. Let $\Lambda_{d_s} \in \mathcal{M}(d_s, t_{\text{obs}})$ be any maximum infection strategy.

Assume that the network administrator uses the estimation radius d_a. We first consider the case where $d_a \geq \bar{d}_s$. Since no infection strategies have safety margins greater than $\bar{d}_s$, the inequality $d_a \geq d_s$ always holds. As a result, the cost $c_s(d_a)$ is always incurred. Therefore, to maximize its utility function, the infection source maximizes its reward $g_s|V_{\text{I}}|$ by maximizing the number of infected nodes $|V_{\text{I}}|$. Lemma 4 then leads to the conclusion that $\Lambda_0 = \Lambda_{\max}$ is a best-response infection strategy.

Next, consider the case where $d_a < \bar{d}_s$. We claim that a best-response infection strategy is either Λ_0 or Λ_{d_a+1}. Following Theorem 3, it suffices to prove the claim by showing the following inequalities:

$$u_s(d_a, \Lambda_0) > u_s(d_a, \Lambda_{d'_s}), \quad \forall\ d'_s \in (0, d_a]; \tag{27}$$

$$u_s(d_a, \Lambda_{d_a+1}) > u_s(d_a, \Lambda_{d'_s}), \quad \forall\ d'_s \in (d_a+1, \bar{d}_s]. \tag{28}$$

We first show the inequality (27). When $0 < d'_s \leq d_a$, the source incurs a cost of $c_s(d_a)$ for both Λ_0 and $\Lambda_{d'_s}$. In addition, from Lemma 4, we have $|V_{\rm I}(\Lambda_0)| > |V_{\rm I}(\Lambda_{d'_s})|$, which in turn shows that the inequality (27) holds. We next show the inequality (28). When $d_a + 1 < d'_s \leq \bar{d}_s$, the source does not incur a cost for both Λ_{d_a+1} and $\Lambda_{d'_s}$. From Lemma 4, we have $|V_{\rm I}(\Lambda_{d_a+1})| > |V_{\rm I}(\Lambda_{d'_s})|$, which shows that the inequality (28) holds. This completes the proof for the claim. The following theorem now follows immediately.

Theorem 4. *Suppose that G is a tree. Then, for any estimation radius $d_a \geq 0$, a best-response infection strategy for the infection source is given by*

$$\Lambda^* = \begin{cases} \Lambda_0, & \text{if } c_s(d_a) < g_s(|V_{\rm I}(\Lambda_0)| - |V_{\rm I}(\Lambda_{d_a+1})|), \\ \Lambda_{d_a+1}, & \text{if } c_s(d_a) > g_s(|V_{\rm I}(\Lambda_0)| - |V_{\rm I}(\Lambda_{d_a+1})|), \\ \Lambda_0 \text{ or } \Lambda_{d_a+1}, & \text{if } c_s(d_a) = g_s(|V_{\rm I}(\Lambda_0)| - |V_{\rm I}(\Lambda_{d_a+1})|), \end{cases}$$

where $\Lambda_d \in \mathcal{M}(d, t_{\rm obs})$ for all $d \geq 0$.

V. Nash Equilibrium in a Tree

In this section, we derive conditions under which a Nash equilibrium for the strategic game played by the network administrator and the infection source exists. We also derive explicitly their respective strategies at these Nash equilibria.

Theorem 5. *Suppose G is a tree, and $\bar{d}_s$ in* (12) *is greater than 0. Then, the strategic game of infection spreading and source identification* (7)-(8) *has the following properties:*

(a) Let $\Lambda_0 \in \mathcal{M}(0, t_{\rm obs})$. The strategy pair $(0, \Lambda_0)$ is a Nash equilibrium iff $u_s(0, \Lambda_1) \leq u_s(0, \Lambda_0)$, i.e., $c_s(0) \leq g_s\left(|V_{\rm I}(\Lambda_0)| - |V_{\rm I}(\Lambda_1)|\right)$ for any $\Lambda_1 \in \mathcal{M}(1, t_{\rm obs})$.

(b) For each $\Lambda_1 \in \mathcal{M}(1, t_{\rm obs})$, the strategy pair $(0, \Lambda_1)$ is a Nash equilibrium iff $u_s(0, \Lambda_0) \leq u_s(0, \Lambda_1)$ and $u_a(1, \Lambda_1) \leq u_a(0, \Lambda_1)$, i.e., $c_s(0) \geq g_s\left(|V_{\rm I}(\Lambda_0)| - |V_{\rm I}(\Lambda_1)|\right)$ and $g_a(1, V_{\rm I}(\Lambda_1)) \leq c_a(V_{\rm sp}(1)) - c_a(V_{\rm sp}(0))$.

(c) No other pure strategy Nash equilibria exist.

Furthermore, if $g_a(d_a, V_{\rm I}(\Lambda)) = g_a(d_a)$ is non-increasing in d_a for all infection strategies Λ, and $c_a(V_{\rm sp}(d_a)) = c_a(d_a)$ is non-decreasing in d_a for all infection strategies Λ, then the sum utility of the two players is maximized at the strategy pairs $(0, \Lambda_0)$ or $(0, \Lambda_1)$ for all $\Lambda_1 \in \mathcal{M}(1, t_{\rm obs})$.

The proof of Theorem 5 is provided in Appendix E. From Theorem 5, when a Nash equilibrium exists, whether a infection source is identified with positive probability or not depends on the relative gains and costs of the two

players. It is interesting to note that if a Nash equilibrium exists, then the strategy of the network administrator in equilibrium has $d_a = 0$, which corresponds to the Jordan center estimator. This shows that under the technical conditions given in Theorem 5, the natural infection source estimator to use is the Jordan center estimator, instead of probing a neighborhood set of the Jordan centers.

VI. Simulation Results

In this section, we present simulation results to evaluate the performance of our proposed infection and estimation strategies. We first compare DIS with AD, and then show the behavior of the best-response infection and estimation strategies under different gains and costs. For simplicity, our simulations are performed assuming that $g_a(\cdot,\cdot) = g_a$, $c_a(V_{\text{sp}}(d_a)) = c_a|V_{\text{sp}}(d_a)|$, and $c_s(d_a) = c_s$.

A. Extension to General Networks

Although the paths along which the infection spreads from the source node forms a tree that is a subgraph of the given graph G, finding the best underlying tree over which to perform the infection spreading is a NP-hard problem (similar to the procedure used in Algorithm 1, this is equivalent to the longest path problem in a weighted graph, which is known to be NP-hard [38]). To adapt our proposed DIS strategy for general networks, we adopt a heuristic: we first find a breadth-first search tree rooted at the infection source and then apply the DIS strategy on this tree. In the following, we show simulation results to verify the performance of the DIS strategy in general networks.

B. Number of Infected Nodes

We first evaluate the effectiveness of our proposed DIS algorithm in infecting nodes. We perform simulations on four kinds of networks: random trees where each node has a degree uniformly drawn from the set $\{2, 3\}$, scale-free networks [39] with 5000 nodes, the western states power grid network of the United States [40] containing 4941 nodes, and a part of the Facebook network with 4039 nodes [41].

The benchmark we compare against is the AD infection strategy proposed in [29], where AD is shown to be order-optimal for the source for infinite regular trees (with heuristic extensions to more general networks). We let t to be even and $\bar{\lambda}_m = 1$ for $m \geq 0$ in the simulations in order not to conflate the effect of the rate bounds with the other factors. Given any observation time t_{obs}, the safety margin $d_s(\text{AD})$ resulting from the AD falls in the range $[1, t_{\text{obs}}/2]$ with probability one, and the set of infected nodes at time t_{obs} is

$$V_{\text{I}}(\text{AD}) = \{u \in G : d(\tilde{v}, u) \leq t_{\text{obs}}/2\},$$

where $\tilde{v}$ is picked uniformly at random from the set of nodes in G with distance $d_s(\text{AD})$ from the infection source v^*.

We let the observation time t_{obs} to be 14, 14, 6 and 6 for random trees, the power grid network, scale-free networks and the Facebook network, respectively. The observation times for scale-free networks and the Facebook network

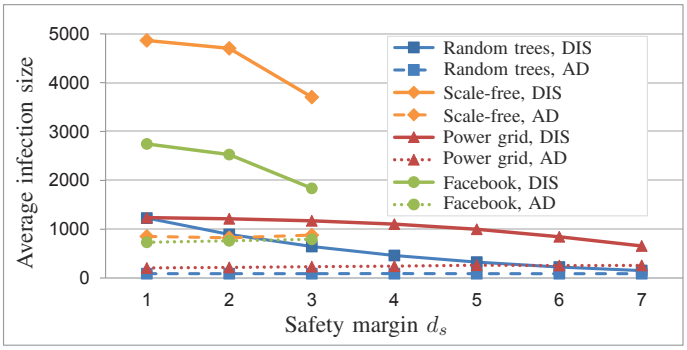


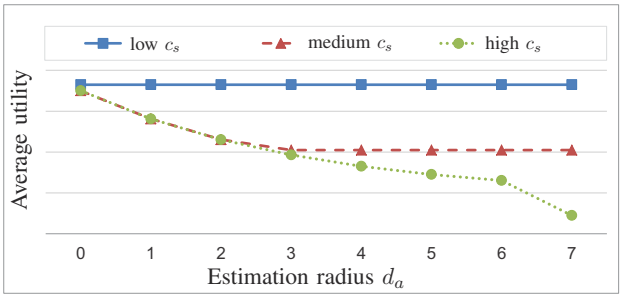
Fig. 3. Average numbers of infected nodes for DIS and AD for different networks.

are chosen to be relatively small because these networks are highly connected and the average distances between each pair of nodes in the scale-free network and the Facebook network are only 4.6 and 5.5 hops, respectively. The safety margin requirement d_s is set to be $1, 2, \ldots, t_{\text{obs}}/2$, respectively. We run 1000 simulation runs for each kind of network and each value of d_s. Fig. 3 shows the average number of infected nodes for both DIS and AD. As expected, we see that there is a trade off for DIS between the number of infected nodes and the safety margin. We also see that DIS consistently infects more nodes than AD.

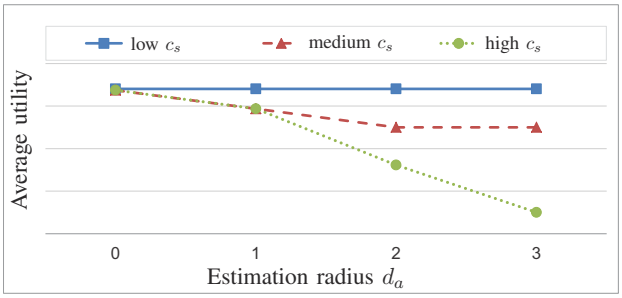
C. Best-response Infection Strategy

We then evaluate the proposed best-response infection strategy on random trees and the Facebook network. The observation time t_{obs} for each network is chosen as in Section VI-B. For each value of $d_a \in [0, \bar{d}_s]$, where $\bar{d}_s = t_{\text{obs}}/2$, the best-response infection strategy is given by Theorem 4. We fix the gain g_s for all cases and vary the cost c_s to make it low, medium and high, compared to g_s. Specifically, we set c_s to be $400g_s$, $1200g_s$ and $2000g_s$ for random trees, and $50g_s$, $500g_s$ and $1500g_s$ for Facebook network. Let DIS_{d_s} denote the DIS strategy with safety margin constraint d_s. We run 1000 simulation runs for each setting and plot the average utility of the best-response infection strategies for the infection source in Fig. 4. We observe similar trends from Fig. 4 for both random trees and the Facebook network, even though the DIS strategy was derived for tree networks.

- When the cost c_s is low compared to the gain g_s, the infection source always chooses $\text{DIS}(0, t_{\text{obs}})$ as its infection strategy. As a result, the infection source is identified by the network administrator. However, it maximizes its reward by infecting the most number of nodes.
- When the cost c_s is high compared to the gain g_s, the infection source chooses $\text{DIS}(d_a + 1, t_{\text{obs}})$ for $d_a < \bar{d}_s$ to ensure that the network administrator does not identify it, and chooses $\text{DIS}(0, t_{\text{obs}})$ for $d_a = \bar{d}_s$ as it can not find any infection strategy with a safety margin greater than $\bar{d}_s$ (cf. Theorem 2).
- When the cost c_s is medium compared to the gain g_s, the infection source chooses $\text{DIS}(d_a + 1, t_{\text{obs}})$ when d_a is small. As d_a increases, the infection source switches to $\text{DIS}(0, t_{\text{obs}})$ as its infection strategy.



(a) Random trees.



(b) Facebook network.

Fig. 4. Average utility of the best-response infection strategies for the infection source. When c_s is low, $\text{DIS}(0, t_{\text{obs}})$ is the best-response infection strategy for all d_a. When c_s is high, $\text{DIS}(d_a+1, t_{\text{obs}})$ are the best-response infection strategies for $d_a < \bar{d}_s$, and $\text{DIS}(0, t_{\text{obs}})$ is the best-response infection strategy for $d_a = \bar{d}_s$. When c_s is medium, $\text{DIS}(d_a+1, t_{\text{obs}})$ are the best-response infection strategies for $d_a \leq 2$ for random trees and $d_a \leq 1$ for Facebook network, respectively, and $\text{DIS}(0, t_{\text{obs}})$ is the best-response infection strategy for other values of d_a.

D. Best-response Estimation Strategy

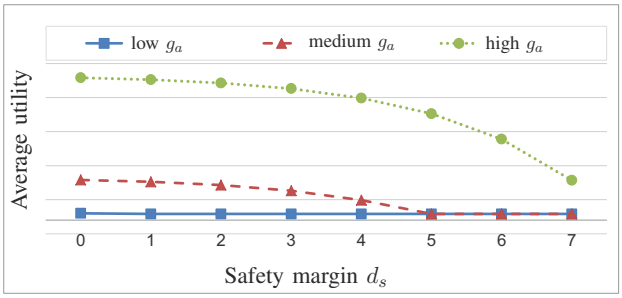
Lastly, we evaluate the proposed best-response estimation strategy on random trees and Facebook network. Given any infection strategy Λ with safety margin $d_s(\Lambda) = d_s$, where $d_s \in [1, \bar{d}_s]$ and $\bar{d}_s = t_{\text{obs}}/2$, the best-response estimation strategy is given by Theorem 1. We choose n_{obs} so that it corresponds to the same observation time t_{obs} used for each network in Section VI-B. We fix the cost c_a for all cases and vary the gain g_a to make it low, medium and high, compared to c_a. Specifically, we set g_a to be c_a, $50c_a$ and $200c_a$ for random trees, and c_a, $500c_a$ and $2000c_a$ for Facebook network. We run 1000 simulation runs for each setting and plot the average utility of the best-response estimation strategies for the network administrator in Fig. 5. We observe the following from Fig. 5.

- When the gain g_a is low compared to the cost c_a, the network administrator always chooses d_a to be 0 to minimize the cost. As a result, the infection source gets caught only when $d_s = 0$.
- When the gain g_a is high compared to the cost c_a, the network administrator always chooses d_a to be d_s. As a result, the overall cost increases with d_s as more nodes need to be investigated, which in turn decreases the utility of the network administrator. Moreover, the infection source always gets caught in this case.
- When the gain g_a is medium compared to the cost c_a, the network administrator chooses d_a to be d_s when d_s is small and the gain of identifying the infection source is higher than the cost of investigating more nodes. When d_s increases to a point that the increase in cost of investigating $|V_{\text{sp}}(d_a)| - 1$ more nodes exceeds the gain of identifying the infection source, the network administrator chooses d_a to be 0.

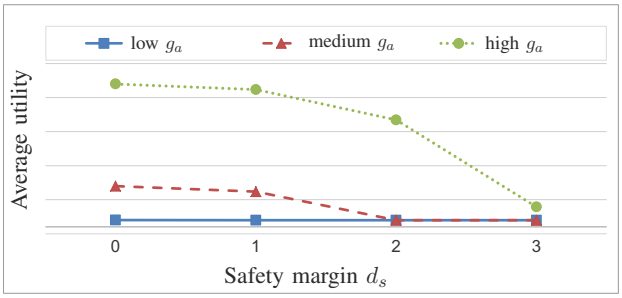
E. Incomplete Observations

In this paper, we have assumed that the network administrator can observe all the infected nodes. In this subsection, we evaluate the robustness of the Jordan center based estimation strategy and DIS infection strategy when only a subset of the infected nodes are observed by the network administrator.

Let α be the percentage of infected nodes that are randomly observed by the network administrator, and let $\text{JC}(\alpha)$ be the Jordan center of the set of observed infected nodes. Note that $\text{JC}(\alpha)$ may be different from the Jordan



(a) Random trees.



(b) Facebook network.

Fig. 5. Average utility of the best-response estimation strategies for the network administrator. When g_a is low, d_a is chosen to be 0 for all d_s. When g_a is high, d_a is chosen to be d_s for all d_s. When g_a is medium, d_a is set to be d_s for $d_s \leq 4$ for random trees and $d_s \leq 1$ for Facebook network, respectively, and d_a is set to be 0 for other values of d_s.

center of all infected nodes JC. Therefore, the distance $d(v^*, \text{JC}(\alpha))$ can differ from d_s. The network administrator can identify the infection source when $d(v^*, \text{JC}(\alpha)) \leq d_a$.

We perform simulations on random trees, scale-free networks, the power grid network and the Facebook network. We set $g_s = 1$ and c_s to be 1200, 6000, 1600 and 3000 for random trees, scale-free networks, the power grid network and the Facebook network, respectively. For the network administrator, we let the gain g_a to be medium compared to c_a. Specifically, we set $c_a = 1$ and g_a to be 50, 1500, 200 and 500 for random trees, scale-free networks, the power grid network and the Facebook network, respectively. The observation threshold n_{obs} for each network is chosen to correspond to the same observation time t_{obs} used in Section VI-B. For each kind of network, we run 1000 simulations for each value of $d_s \in [0, \bar{d}_s]$, $d_a \in \{0, 1, \cdots, d_s + 1\}$ and $\alpha \in \{1, 10, 50\}$, where $\bar{d}_s = t_{\text{obs}}/2$. For each simulation run, we randomly pick α percent of infected nodes as observed nodes, compute the *realized* utility values of the network administrator and infection source, and average them over the simulation runs. In the realized utilities we compute, the network administrator obtains a gain g_a while the infection source incurs a cost c_s, only when $d(v^*, \text{JC}(\alpha)) \leq d_a$. The average utilities of the infection source and the network administrator are shown in Fig. 6 and Fig. 7, respectively.

Consider the utility of the infection source in Fig. 6. The best-response of the infection source is still choosing d_s to be either 0 or $d_a + 1$. This implies that the result of Theorem 4 is robust for the tested networks even though only a subset of infected nodes can be observed.

Fig. 7 shows the utility of the network administrator. For random trees, the best-response of the network administrator is still choosing d_a to be either 0 or d_s, which verifies Theorem 1 in the case where only partial observations are available. On the other hand, for general networks with incomplete observations of the set of infected nodes, it becomes more difficult for the network administrator to correctly identify the infection source. The network administrator needs to increase d_a in order to have a higher chance of identifying the infection source. However, for dense networks, the cost of probing more nodes can increase very quickly as d_a increases. As a result, for scale-free networks and the Facebook network, the network administrator tends to choose d_a to be 0

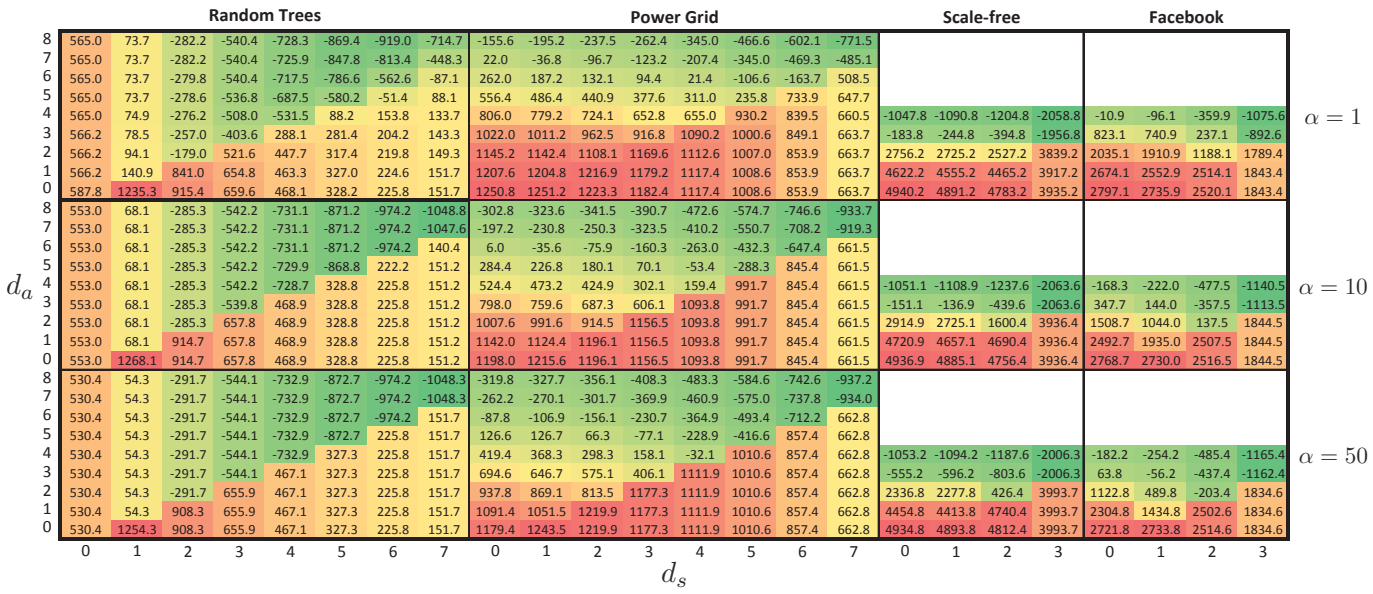


Fig. 6. Average utility of the infection source for various observation percentage α. The color scale of the heat map is calibrated for each kind of network respectively.

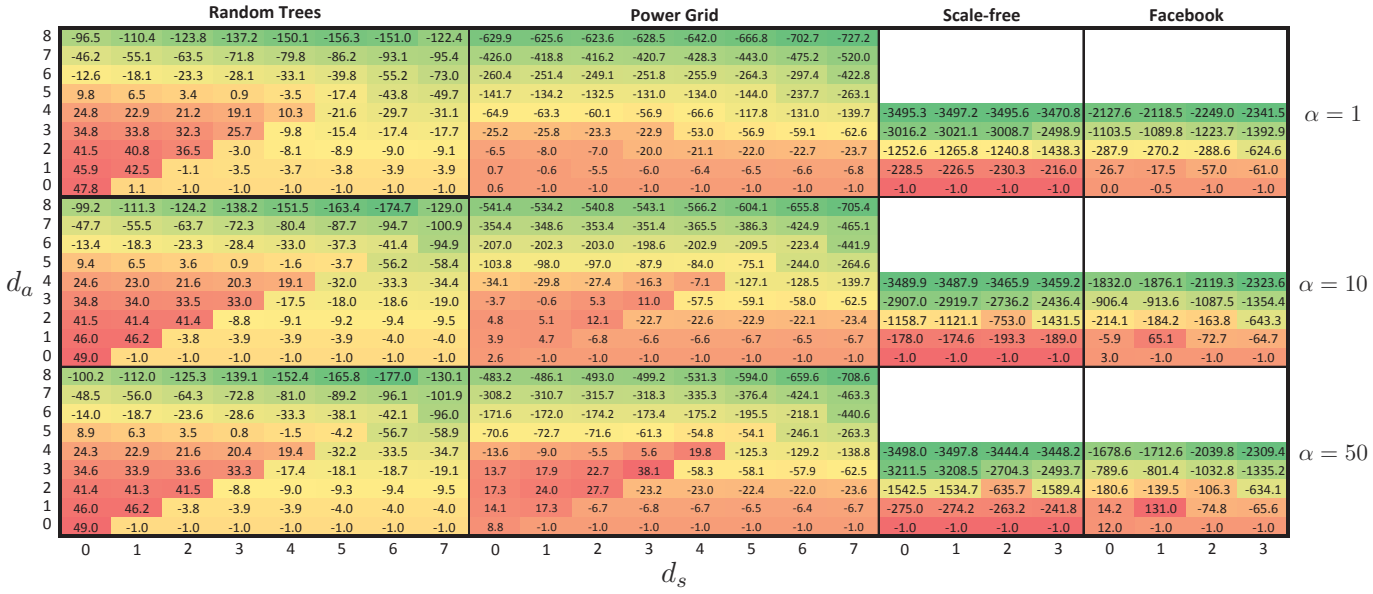


Fig. 7. Average utility of the network administrator for various observation percentage α. The color scale of the heat map is calibrated for each kind of network respectively.

to minimize the cost instead. In practice when the network administrator cannot observe all node status, in order to reduce its probing cost, it needs to formulate an estimation strategy that incorporates other side information. For example, in trying to identify the source of a computer virus, part of the cost of examining every node in the suspect set can be reduced by only examining known weak points in the network or by performing a forensic analysis of the virus code to reduce the suspect set size.

VII. Conclusion

We have formulated the problems of maximizing infection spreading and source identification in a network as a strategic game. Conditioned on the strategy of the other player, we proposed best-response strategies for both the infection source and the network administrator in a tree network. We also derived conditions under which a Nash equilibrium exists. In all Nash equilibria, the Jordan center estimator is the equilibrium estimation strategy for the network administrator. We showed that the sum utility of both players is maximized at one of these Nash equilibria.

In this work, we have assumed that the underlying network is a tree. Obtaining theoretical results for general networks seems unlikely due to difficulties in designing an optimal infection strategy in a loopy graph. Future work includes designing best-response strategies for the network administrator under a more general class of estimation strategies that may not be based on the Jordan center. It would also be of interest to study the best-response infection and estimation strategies when infection rates are stochastic and not fully controllable by the infection source, or when additional side information is available to the network administrator. We have also adopted a simple game theoretic formulation in this paper where the network administrator makes a one-shot observation of the network. It would be of interest to consider cases where the network administrator can observe the evolution of the network [42], [43] by formulating a multi-stage game.

Appendix A
Proof of Theorem 2

We prove Theorem 2 in two steps. We first show that there exists at least one infection strategy that can achieve $d_s = \bar{d}_s$. We then show that there is no infection strategy that results in $d_s > \bar{d}_s$.

Step 1: We only need to find one infection strategy that has safety margin $d_s = \bar{d}_s$. Let $u \in V$ be a node with $d(v^*, u) = \bar{d}(t)$, and let D be the path from v^* to u. Consider the following infection strategy: set the infection rate of each edge in D to be the respective maximum infection rate, and set the infection rates of other edges not in D to be 0. We then have $d_s = \lfloor \bar{d}(t)/2 \rfloor = \bar{d}_s$.

Step 2: Assume $d_s > \bar{d}_s$, i.e., $d_s \geq \bar{d}_s + 1$. Consider any infection strategy Λ and a Jordan center u such that $d(v^*, u) = d_s$ and let $D = (l_1, \ldots, u, \ldots, l_2)$ be a diameter of G_t containing u, where l_1 and l_2 are leaf nodes, with $d(l_1, v^*) \leq d(l_2, v^*)$. We first show that $d(v^*, u) \leq d(l_2, u)$. It can be shown that $d(l_1, u)$ and $d(l_2, u)$ differs in value by at most 1 [21]. If $d(l_1, u) = d(l_2, u) + 1$, consider the neighbouring node u' of u on the path $\rho(u, l_1)$. From [21], we obtain that u' is a Jordan center with $d(v^*, u') = d_s - 1$, a contradiction. Therefore, we have $d(l_1, u) \leq d(l_2, u)$. It is easy to see that $d(v^*, u) \leq d(l_2, u)$ because otherwise, the path with $\rho(v^*, u)$ concatenated with $\rho(u, l_2)$ has length greater than that of D, a contradiction. We then have

$$\begin{aligned} d(v^*, l_2) &= d(v^*, u) + d(u, l_2) \\ &\geq 2d(v^*, u) \end{aligned}$$

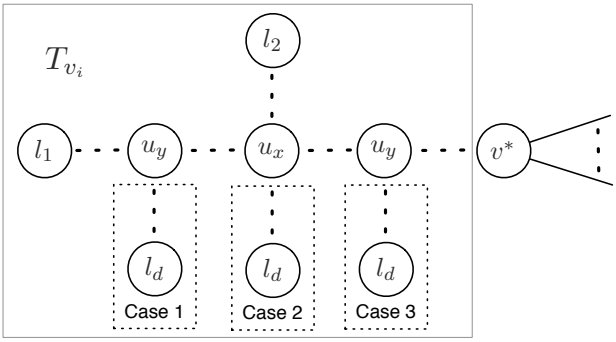


Fig. 8. Illustration of part of the infection graph G_{I}.

$$\geq 2(\bar{d}_s + 1)$$
$$\geq \bar{d}(t) + 1,$$

a contradiction since the infection can travel at most $\bar{d}(t)$ hops in time t. This shows that no infection strategy results in $d_s > \bar{d}_s$. The proof for Theorem 2 is now complete.

APPENDIX B
PROOF OF LEMMA 1

We call the leaf node l_d of a dominant path a *dominant leaf*. We prove Lemma 1 in two steps. We first show that any diameter D of $G_{\mathrm{I},t}$ contains at least one dominant leaf. We then show that the infection rate associated with each edge in any dominant path is its upper bound $\bar{\lambda}_m$.

Step 1: Show that any diameter D of $G_{\mathrm{I},t}$ contains at least one dominant leaf.

Let l_1 and l_2 to be the two end nodes of D. Let v_i and v_j to be two different neighboring nodes of v^*. We consider two possible scenarios: l_1 and l_2 are in different subtrees T_{v_i} and T_{v_j}, respectively; l_1 and l_2 are in the same subtree T_{v_i}.

SCENARIO 1: l_1 and l_2 are in different subtrees T_{v_i} and T_{v_j}, respectively.

Suppose D does not contain any dominant leaf. Then we can find a dominant path $\rho(v^*, l_d)$ such that $l_d \notin T_{v_i}$ (if $l_d \in T_{v_i}$, we have $l_d \notin T_{v_j}$ and just exchange the notations i and j). Consider the path $D' = \{l_1, \cdots, v^*, \cdots, l_d\}$, we have

$$\begin{aligned} |D'| &= d(v^*, l_1) + d(v^*, l_d) + 1 \\ &> d(v^*, l_1) + d(v^*, l_2) + 1 \\ &= |D|. \end{aligned}$$

Thus, we find a path D' that has longer distance than the diameter D, a contradiction. So D must contain at least one dominant leaf.

SCENARIO 2: l_1 and l_2 are in the same subtrees T_{v_i}.

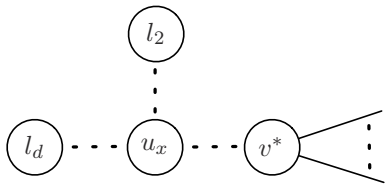


Fig. 9. Illustration of the diameter containing the dominant leaf l_d.

Suppose D does not contain any dominant leaf. Consider a dominant leaf l_d. If $l_d \notin T_{v_i}$, following the same argument as in Scenario 1, we can find a path $D' = \{l_1, \cdots, v^*, \cdots, l_d\}$ that has longer distance than D. We now consider the case where $l_d \in T_{v_i}$. Let u_x to be the first node on which the two paths $\rho(l_1, v^*)$ and $\rho(l_2, v^*)$ intersects, where x is the depth of u_x and $1 \leq x \leq \min\{d(v^*, l_1) - 1, d(v^*, l_2) - 1\}$. Then let u_y to be the first node on which the two paths $\rho(l_1, v^*)$ and $\rho(l_d, v^*)$ intersects, where y is the depth of u_y and $0 \leq y \leq d(v^*, l_1) - 1$. Figure 8 shows all three possible cases: $y > x, y = x$ and $y < x$. Consider the path $D' = \{l_d, \cdots, u_x, \cdots, l_2\}$. For case 1 and case 2 in Figure 8, we have $d(u_x, l_d) > d(u_x, l_1)$ because $d(v^*, l_d) > d(v^*, l_1)$. Similarly, for case 3 in Figure 8, we have $d(u_x, l_d) > d(u_y, l_d) > d(u_y, l_1)$. As a result, for all three cases, we have

$$\begin{aligned} |D'| &= d(u_x, l_d) + d(u_x, l_2) + 1 \\ &> d(u_x, l_1) + d(u_x, l_2) + 1 \\ &= |D|. \end{aligned}$$

We find a path D' that has longer distance than the diameter D, a contradiction. We can now conclude that any diameter D of $G_{\mathrm{I},t}$ contains at least one dominant leaf.

Step 2: Show that the infection rate associated with each edge in any dominant path is its upper bound $\bar{\lambda}_m$.

Consider any dominant path $P_d = \rho(v^*, l_d)$ and suppose the infection rate of some edges in P_d are less than $\bar{\lambda}_m$. Let $D = \{l_d, \cdots, u_x, \cdots, l_2\}$ to be the diameter containing l_d as shown in Figure 9, where $x \geq 0$ is the depth of u_x. Consider a Jordan center u such that $d(v^*, u) = d_s$. It is easy to see that u is at the middle of the diameter. Since $d(v^*, l_d) > d(v^*, l_2)$, we have $d(u_x, l_d) > d(u_x, l_2)$, which in turn implies that $u \in \rho(u_x, l_d)$. If we increase the infection rates of all edges in P_d to their maximum rates, the length of P_d will increase and u will move further away from v^*, i.e., the safety margin will increase as well. We can then increase the infection rates of some edges in the path $\rho(u_x, l_2)$ to increase the length of $\rho(u_x, l_2)$. As a result, u will move closer to v^* and the safety margin can reduce back to its original value. In this case, we find another infection strategy that results in more infected nodes subject to the same safety margin d_s, a contradiction. We can now conclude that the infection rate associated with each edge in any dominant path is its maximum rate. This completes the proof of Lemma 1.

APPENDIX C
PROOF OF LEMMA 2

Fix any diameter and let it be D. We prove Lemma 2 by contradiction. Suppose neither end of D is in the subtree T_{v_1}, then there are two possible cases: (1) both ends of D are in the subtree T_{v_i}, where $2 \leq i \leq k$; (2) the two ends of D are in the subtree T_{v_i} and T_{v_j}, respectively, where $2 \leq i, j \leq k$ and $i \neq j$.

We first consider case (1) and let l_1 and l_2 to be the two ends of D. Consider two paths $\rho(l_1, v^*)$ and $\rho(l_2, v^*)$ and let the first node on which these two paths intersect as w. We can find a path $D' = \{l_{v_1}, \cdots, v^*, \cdots, w, \cdots, l_2\}$ that has longer distance than D. We have

$$\begin{aligned}
&|\{l_{v_1}, \cdots, v^*, \cdots, w\}| \\
&> |\{l_{v_1}, \cdots, v^*\}| \\
&= d(v^*, l_{v_1}) + 1 \\
&> d(v^*, l_{v_i}) + 1 \\
&> d(l_1, u) + 1 \\
&= |\{l_1, \cdots, u\}|.
\end{aligned}$$

Then we have

$$\begin{aligned}
|D'| &= |\{l_{v_1}, \cdots, v^*, \cdots, u\}| + d(u, l_2) \\
&> |\{l_1, \cdots, u\}| + d(u, l_2) \\
&= |D|.
\end{aligned}$$

We find a path that has greater length than the diameter, which contradicts with the definition of diameter. This completes the proof for case (1).

We then consider case (2). Fix i and j, and let l_i and l_j denote the two ends of the diameter in subtree T_{v_i} and T_{v_j}, respectively. Since $d(v^*, l_{v_1}) > d(v^*, l_{v_i})$, the length of the path $D' = \{l_{v_1}, \cdots, v^*, \cdots, l_{v_j}\}$ is greater than the length of diameter $D = \{l_{v_i}, \cdots, v^*, \cdots, l_{v_j}\}$, which contradicts with the definition of diameter. This completes the proof for case (2), and the proof for Lemma 2 is now complete.

APPENDIX D
PROOF OF LEMMA 3

Consider any infection strategy Λ with safety margin d_s, a dominant path P_d, and that maximizes the number of infected nodes. We use the same notations in the discussion preceding Lemma 3, with $P_d = (u_0, \ldots, u_{\bar{d}(t)})$. In addition, let $T_u(G)$ be the subtree of G rooted at node u with the first link in the path from u to v^* removed. Let $\tilde{T}_{u_m}(G)$ be $T_{u_m}(G) \backslash T_{u_{m+1}}(G)$.

Suppose there exists a $m \in [0, d_s]$ such that for a path $\rho(u_m, \tilde{l}_{u_m})$, we have $d(u_m, T_{u_m}(G)) \geq \lambda'_m(t - t_m)$, and $\lambda'_m > \lambda_m$, where

$$\lambda'_m = d(u_m, \tilde{l}_{u_m}) \left(\sum_{(i,j) \in \rho(u_m, \tilde{l}_{u_m})} \lambda(i,j)^{-1} \right)^{-1},$$

i.e., λ'_m is the average infection rate along the path $\rho(u_m, \tilde{l}_{u_m})$. (Recall that λ_m is the average infection rate used by DIS(d_s, t) for $\tilde{T}_{u_m}(G)$.) We have

$$\left\lfloor \lambda'_m(t - t_m) \right\rfloor = d(u_m, \tilde{l}_{u_m}),$$

since otherwise, we can infect more nodes, contradicting the assumption that Λ maximizes the number of infected nodes. We also have $\hat{v}(u_m)$ is a Jordan center. By replacing λ_m with λ'_m in (19), we have $d(v^*, \hat{v}(u_m)) < d_s$ since $\lambda'_m > \lambda_m$ implies that the inequality in (20) is reversed when λ_m is replaced by λ'_m. This contradicts the assumption that Λ has safety margin d_s.

On the other hand, if $d(u_m, \tilde{T}_{u_m}(G)) < \lambda'_m(t - t_m)$, i.e., all the nodes in $\tilde{T}_{u_m}$ are infected by Λ before time t, then we can choose a $\lambda''_m \in (\lambda'_m, \lambda_m)$, infect all the nodes in $\tilde{T}_{u_m}$ by time t, and repeat the above argument using λ''_m in place of λ'_m. Therefore, no other strategy can infect more nodes than DIS(d_s, t), and the proof is complete.

APPENDIX E
PROOF OF THEOREM 5

We first prove the properties (a)-(c) in sequence. We then prove the sum utility optimality claim.

Proof of Theorem 5(a)**.**

Following the definition of the Nash equilibrium, it suffices to show that

$$u_a(0, \Lambda_0) \geq u_a(d_a, \Lambda_0), \ \forall d_a > 0, \tag{29}$$

$$u_s(0, \Lambda_0) \geq u_s(0, \Lambda), \ \forall \Lambda. \tag{30}$$

The inequality (29) follows from Theorem 1. To show (30), let Λ be any infection strategy, and d_s be its safety margin. If $d_s = 0$, we obtain from Lemma 4

$$\begin{aligned} u_s(0, \Lambda_0) &= g_s |V_{\mathrm{I}}(\Lambda_0)| - c_s(0) \\ &\geq g_s |V_{\mathrm{I}}(\Lambda)| - c_s(0) \\ &= u_s(0, \Lambda). \end{aligned}$$

If $d_s > 0$, from the assumption of Theorem 5(a) and Lemma 4, we have for any $\Lambda_1 \in \mathcal{M}(1)$,

$$\begin{aligned} u_s(0, \Lambda_0) &\geq u_s(0, \Lambda_1) \\ &= g_s |V_{\mathrm{I}}(\Lambda_1)| \\ &\geq g_s |V_{\mathrm{I}}(\Lambda)| \\ &= u_s(0, \Lambda). \end{aligned}$$

The proof of Theorem 5(a) is now complete.

Proof of Theorem 5(b)**.**

It again suffices to show that for $\Lambda_1 \in \mathcal{M}(1, t_{\text{obs}})$ satisfying the assumptions of Theorem 5(b), we have

$$u_a(0, \Lambda_1) \geq u_a(d_a, \Lambda_1),\ \forall d_a > 0, \tag{31}$$

$$u_s(0, \Lambda_1) \geq u_s(0, \Lambda),\ \forall \Lambda. \tag{32}$$

From the second assumption of Theorem 5(b), for any $d_a > 0$, we have

$$\begin{aligned} u_a(0, \Lambda_1) &\geq u_a(1, \Lambda_1) \\ &= -c_a(V_{\text{sp}}(1)) + g_a(1, V_{\text{I}}(\Lambda_1)) \\ &\geq -c_a(V_{\text{sp}}(d_a)) + g_a(d_a, V_{\text{I}}(\Lambda_1)) \\ &= u_a(d_a, \Lambda_1), \end{aligned}$$

and inequality (31) holds.

We now show (32). Let Λ be any infection strategy and d_s be its safety margin. If $d_s = 0$, from the first assumption of Theorem 5(b), and Definition 1, we have

$$\begin{aligned} u_s(0, \Lambda_1) &\geq u_s(0, \Lambda_0) \\ &= g_s|V_{\text{I}}(\Lambda_0)| - c_s(0) \\ &\geq g_s|V_{\text{I}}(\Lambda)| - c_s(0) \\ &= u_s(0, \Lambda). \end{aligned}$$

If $d_s > 0$, from Lemma 4, we have

$$\begin{aligned} u_s(0, \Lambda_1) &= g_s|V_{\text{I}}(\Lambda_1)| \\ &\geq g_s|V_{\text{I}}(\Lambda)| \\ &= u_s(0, \Lambda). \end{aligned}$$

We have now shown that (32) holds and the proof of Theorem 5(b) is complete.

Proof of Theorem 5(c)**.**

Consider any strategy pair $(d_a, \Lambda) \neq (0, \Lambda_0)$ or $(0, \Lambda_1)$ for all $\Lambda_1 \in \mathcal{M}(1, t_{\text{obs}})$. Let d_s be the safety margin of Λ. If $d_a = 0$ and $d_s = 0$, then

$$\begin{aligned} u_s(0, \Lambda) &= g_s|V_{\text{I}}(\Lambda)| - c_s(0) \\ &< g_s|V_{\text{I}}(\Lambda_0)| - c_s(0) \\ &= u_s(0, \Lambda_0), \end{aligned}$$

so (d_a, Λ) is not a Nash equilibrium. If $d_a = 0$ and $d_s \geq 1$, we have $u_s(0, \Lambda) = g_s|V_{\text{I}}(\Lambda)| < g_s|V_{\text{I}}(\Lambda_1)|$ for any $\Lambda_1 \in \mathcal{M}(1, t_{\text{obs}})$. This again implies that (d_a, Λ) is not a Nash equilibrium.

Now suppose that $d_a > 0$. From Theorem 4, it suffices to show that (d_a, Λ_0) and (d_a, Λ_{d_a+1}) for all $\Lambda_{d_a+1} \in \mathcal{M}(d_a+1, t_{\text{obs}})$ are not Nash equilibria. We have

$$\begin{aligned} u_a(d_a, \Lambda_0) &= -c_a(V_{\text{sp}}(d_a)) + g_a(d_a, V_{\text{I}}(\Lambda_0) \\ &< -c_a(V_{\text{sp}}(0)) + g_a(0, V_{\text{I}}(\Lambda_0)) \\ &= u_a(0, \Lambda_0), \end{aligned}$$

where the inequality follows from the assumption that $\bar{d_s} > 0$. For each $\Lambda_{d_a+1} \in \mathcal{M}(d_a+1, t_{\text{obs}})$, we have

$$\begin{aligned} u_a(d_a, \Lambda_{d_a+1}) &= -c_a(V_{\text{sp}}(d_a+1)) \\ &< -c_a(V_{\text{sp}}(0)) \\ &= u_a(0, \Lambda_{d_a+1}). \end{aligned}$$

This completes the proof of Theorem 5(c).

Proof of sum utility optimality.

It suffices to show that for any strategy pair (d_a, Λ) and every $\Lambda_1 \in \mathcal{M}(1, t_{\text{obs}})$, we have

$$u_a(d_a, \Lambda) + u_s(d_a, \Lambda) \leq \max\{u_a(0, \Lambda_0) + u_s(0, \Lambda_0),\ u_a(0, \Lambda_1) + u_s(0, \Lambda_1)\}. \tag{33}$$

Let d_s be the safety margin of Λ. We first consider the case where $d_a < d_s$. Following (7), (8), Definition 1 and Lemma 4, we have for any $\Lambda_1 \in \mathcal{M}(1, t_{\text{obs}})$,

$$\begin{aligned} u_a(d_a, \Lambda) + u_s(d_a, \Lambda) &= -c_a(d_a) + g_s|V_{\text{I}}(\Lambda)| \\ &\leq -c_a(0) + g_s|V_{\text{I}}(\Lambda_1)| \\ &= u_a(0, \Lambda_1) + u_s(0, \Lambda_1). \end{aligned}$$

This implies that (33) holds for all $\Lambda_1 \in \mathcal{M}(1, t_{\text{obs}})$.

Suppose now that $d_a \geq d_s$. Following (7), (8), Definition 1 and Lemma 4, we have

$$\begin{aligned} u_a(d_a, \Lambda) + u_s(d_a, \Lambda) &= -c_a(d_a) + g_a(d_a) + g_s|V_{\text{I}}(\Lambda)| - c_s(d_a) \\ &\leq -c_a(0) + g_a(0) + g_s|V_{\text{I}}(\Lambda_0)| - c_s(0) \\ &= u_a(0, \Lambda_0) + u_s(0, \Lambda_0). \end{aligned}$$

This implies that (33) holds. The proof of Theorem 5 is now complete.

REFERENCES

[1] B. Viswanath, A. Mislove, M. Cha, and K. P. Gummadi, "On the evolution of user interaction in Facebook," in *Proc. 2nd ACM Workshop on Online Social Networks*, 2009.

[2] R. Kumar, J. Novak, and A. Tomkins, "Structure and evolution of online social networks," in *Link Mining: Models, Algorithms, and Applications*. Springer New York, 2010, pp. 337–357.

[3] M. Cha, H. Haddadi, F. Benevenuto, and K. P. Gummadi, "Measuring user influence in Twitter: the million follower fallacy," in *Proc. 4th International AAAI Conference on Weblogs and Social Media*, 2010.

[4] V. Gundotra. (2012, December) Google+: communities and photos. Google Official Blog.

[5] Pew Research Center. (2014, September) How social media is reshaping news. [Online]. Available: http://goo.gl/xeIoXi

[6] W. P. Tay, "The value of feedback in decentralized detection," *IEEE Trans. Inf. Theory*, vol. 58, no. 12, pp. 7226 – 7239, Dec. 2012.

[7] J. Ho, W. P. Tay, T. Q. S. Quek, and E. K. P. Chong, "Robust decentralized detection and social learning in tandem networks," *IEEE Trans. Signal Process.*, vol. 63, no. 19, pp. 5019 – 5032, Oct. 2015.

[8] W. P. Tay, "Whose opinion to follow in multihypothesis social learning? A large deviations perspective," *IEEE J. Sel. Topics Signal Process.*, vol. 9, no. 2, pp. 344 – 359, Mar. 2015.

[9] L. Han, S. Han, Q. Deng, J. Yu, and Y. He, "Source tracing and pursuing of network virus," in *Proc. 8th IEEE International Conference on Computer and Information Technology Workshops*, 2008.

[10] J. Weng, E.-P. Lim, J. Jiang, and Q. He, "Twitterrank: finding topic-sensitive influential twitterers," in *Proc. 3rd ACM International Conference on Web Search and Data Mining*, 2010.

[11] E. Bakshy, J. M. Hofman, W. A. Mason, and D. J. Watts, "Everyone's an influencer: quantifying influence on Twitter," in *Proc. 4th ACM International Conference on Web Search and Data Mining*, 2011.

[12] The Huffington Post. (2013, June) Jackie Chan addresses death hoax, proves he's alive with Facebook post. [Online]. Available: http://goo.gl/dO0ZQ9

[13] R. K. Garrett, "Troubling consequences of online political rumoring," *Human Communication Research*, vol. 37, pp. 255–274, 2011.

[14] Daily Mail. (2013, April) 'Syrian hackers' break into Associated Press' Twitter account and 'break news' that explosions at White House have injured Obama - sending DOW Jones plunging 100 points. [Online]. Available: http://goo.gl/NSliQP

[15] D. Dittrich and S. Dietrich, "Discovery techniques for P2P botnets," Stevens Institute of Technology, Tech. Rep., September 2008.

[16] M. Thompson. (2009, October) Mariposa botnet analysis. Defence Intelligence. [Online]. Available: http://www.defintel.com/docs/Mariposa_Analysis.pdf

[17] D. Shah and T. Zaman, "Rumors in a network: Who's the culprit?" *IEEE Trans. Inf. Theory*, vol. 57, no. 8, pp. 5163–5181, 2011.

[18] W. Luo, W. P. Tay, and M. Leng, "Identifying infection sources and regions in large networks," *IEEE Trans. Signal Process.*, vol. 61, no. 11, pp. 2850–2865, 2013.

[19] W. Dong, W. Zhang, and C. W. Tan, "Rooting out the rumor culprit from suspects," *arXiv:1301.6312*, 2013. [Online]. Available: http://arxiv.org/abs/1301.6312

[20] K. Zhu and L. Ying, "Information source detection in the SIR model: a sample path based approach," in *Information Theory and Applications Workshop*, 2013.

[21] W. Luo, W. P. Tay, and M. Leng, "How to identify an infection source with limited observations," *IEEE J. Sel. Top. Sign. Proces.*, vol. 8, no. 4, pp. 586–597, 2014.

[22] ——, "On the universality of Jordan centers for estimating infection sources in tree networks," *arXiv:1411.2370*, 2014. [Online]. Available: http://arxiv.org/abs/1411.2370

[23] Secret. [Online]. Available: https://en.wikipedia.org/wiki/Secret_(app)

[24] The Wall Street Journal. (2014, Feburary) Evernote denies 'Secret' acquisition rumor. [Online]. Available: http://goo.gl/6GiuAl

[25] Wickr. [Online]. Available: https://www.wickr.com/

[26] FireChat. [Online]. Available: http://opengarden.com/firechat/

[27] Wikipedia. 2014 Hong Kong protests. [Online]. Available: https://en.wikipedia.org/wiki/2014_Hong_Kong_protests

[28] P. Shadbolt. (2014, Oct.) Firechat in Hong Kong: How an app tapped its way into the protests. CNN. [Online]. Available: http://edition.cnn.com/2014/10/16/tech/mobile/tomorrow-transformed-firechat/

[29] G. Fanti, P. Kairouz, S. Oh, and P. Viswanath, "Spy vs. spy: rumor source obfuscation," *arXiv:1412.8439*, 2014. [Online]. Available: http://arxiv.org/abs/1412.8439

[30] P. Domingos and M. Richardson, "Mining the network value of customers," in *Proc. 7th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2001.

[31] M. Richardson and P. Domingos, "Mining knowledge-sharing sites for viral marketing," in *Proc. 8th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2002.

[32] D. Kempe, J. Kleinberg, and E. Tardos, "Maximizing the spread of influence through a social network," in *Proc. 9th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2003.

[33] J. Leskovec, A. Krause, C. Guestrin, C. Faloutsos, J. VanBriesen, and N. Glance, "Cost-effective outbreak detection in networks," in *Proc. 13th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2007.

[34] W. Chen, Y. Wang, and S. Yang, "Efficient influence maximization in social networks," in *Proc. 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2009.

[35] W. Chen, C. Wang, and Y. Wang, "Scalable influence maximization for prevalent viral marketing in large-scale social networks," in *Proc. 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2010.

[36] S. Wasserman, K. Faust, and D. Iacobucci, *Social Network Analysis: Methods and Applications (Structural Analysis in the Social Sciences)*. Cambridge University Press, 1994.

[37] M. J. Osborne and A. Rubinstein, *A course in game theory*. The MIT Press, 1994.

[38] T. H. Cormen, C. Stein, R. L. Rivest, and C. E. Leiserson, *Introduction to Algorithms*, 2nd ed. McGraw-Hill Higher Education, 2001.

[39] A. L. Barabasi and R. Albert, "Emergence of scaling in random networks," *Science*, vol. 286, no. 5439, pp. 509–512, 1999.

[40] D. J. Watts and S. H. Strogatz, "Collective dynamics of 'small-world' networks." *Nature*, vol. 393, no. 6684, pp. 440–442, 1998.

[41] J. McAuley and J. Leskovec, "Learning to discover social circles in ego networks," in *NIPS*, 2012.

[42] C. Jiang, Y. Chen, and K. Liu, "Evolutionary dynamics of information diffusion over social networks," *IEEE Trans. Signal Process.*, vol. 62, no. 17, pp. 4573–4586, 2014.

[43] ——, "Graphical evolutionary game for information diffusion over social networks," *IEEE J. Sel. Top. Sign. Proces.*, vol. 8, no. 4, pp. 524–536, 2014.

Wuqiong Luo (S'12 M'15) received the B.Eng. degree in Electrical and Electronic Engineering with First Class Honours from Nanyang Technological University, Singapore in 2010. He obtained the PhD degree in Electrical and Electronic Engineering from Nanyang Technological University, Singapore in 2015. He is currently a Senior Engineer at Micron Semiconductor Asia. His current research interest are in statistical learning using big data.

Dr. Luo was coawarded the Best Student Paper Award at the 46th Asilomar Conference on Signals, Systems, and Computers.

Wee Peng Tay (S'06 M'08 SM'14) received the B.S. degree in Electrical Engineering and Mathematics, and the M.S. degree in Electrical Engineering from Stanford University, Stanford, CA, USA, in 2002. He received the Ph.D. degree in Electrical Engineering and Computer Science from the Massachusetts Institute of Technology, Cambridge, MA, USA, in 2008. He is currently an Assistant Professor in the School of Electrical and Electronic Engineering at Nanyang Technological University, Singapore. His research interests include distributed detection and estimation, distributed signal processing, sensor networks, social networks, information theory, and applied probability.

Dr. Tay received the Singapore Technologies Scholarship in 1998, the Stanford University President's Award in 1999, the Frederick Emmons Terman Engineering Scholastic Award in 2002, and the Tan Chin Tuan Exchange Fellowship in 2015. He is the coauthor of the best student paper award at the Asilomar conference on Signals, Systems, and Computers in 2012. He is currently an Associate Editor for the IEEE Transactions on Signal Processing, serves on the MLSP TC of the IEEE Signal Processing Society, and is the chair of DSNIG in IEEE MMTC. He has also served as a technical program committee member for various international conferences.

Mei Leng (S'07 M'10) received the B.Eng. degree from the University of Electronic Science and Technology of China (UESTC), Chengdu, China, in 2005 and the Ph.D. degree from The University of Hong Kong in 2011.

She is currently a Research Scientist in Temasek Laboratories@NTU, Nanyang Technological University, Singapore. Her research interests include navigation and tracking algorithms, distributed algorithms and machine learning with applications to wireless sensor networks and wireless communication systems. Her current research focus on both theoretical and experimental works on outdoor localization and tracking.