IDENTIFYING INFECTION SOURCES IN A NETWORK

LUO WUQIONG

School of Electrical and Electronic Engineering

A thesis submitted to the Nanyang Technological University
in partial fulfilment of the requirement for the degree of
Doctor of Philosophy

2015

Identifying Infection Sources in a Network

by

LUO Wuqiong

submitted to the Nanyang Technological University
in partial fulfilment of the requirements for the degree of
Doctor of Philosophy
April, 2015

Abstract

Modern networks like social, communication and transportation networks have grown drastically in complexity. Such networks are susceptible to rapidly spreading "infection", which can have different meanings under different contexts, including a contagious disease, a computer virus or a rumor. Timely identification of the infection sources (the index cases of a contagious disease, the original servers that injected a computer virus into the Internet, or the rumor origins that started a rumor in a social network) is critical for limiting the damage caused by the infection. The infection sources identification problem has thus attracted considerable interest from the research community over the past few years.

We first consider the problem of identifying multiple infection sources in the Susceptible-Infected (SI) model under the maximum likelihood (ML) criterion. Assuming the number of sources is unknown a priori, we propose estimators to estimate both the infection sources and their infection regions (subsets of nodes infected by each source). We derive estimators for the infection sources and their infection regions based on approximations of the infection sequences count. We prove that if there are at most two infection sources in a geometric tree, our estimator identifies the true source or sources with probability going to one as the number of infected nodes increases. When there are more than two infection sources, and when the maximum possible number of infection sources is known, we propose an algorithm with quadratic complexity to estimate the actual number and identities of the infection sources.

The infection sources identification problem becomes more challenging when we have access only to a limited set of observations. We consider the problem of identifying an infection source for the SI model, in which not all infected nodes can be observed. We show that a Jordan center, i.e., a node with minimum distance to the set of observed infected nodes, serves as an optimal infection source estimator under the most likely infection path (MLIP) criterion. We also propose approximate source estimators for general networks.

We then extend the work to more complex infection spreading models: Susceptible-Infected-Recovered (SIR), Susceptible-Infected-Recovered-Infected (SIRI) and Susceptible-

Infected-Susceptible (SIS) models. Under the MLIP criterion, we show that a Jordan center serves as an optimal estimator for the infection source that is universally applicable for the SI, SIR, SIRI and SIS models.

Finally, we consider the problem of identifying multiple infection sources in the SI model under the MLIP criterion. We introduce the concept of a k-Jordan center set, and show that if an infection spreads from $k > 1$ sources in an infinite tree network according to the SI model, then the k-Jordan center set is an optimal estimator of the infection source set.

Supervisor: Tay Wee Peng
Title: Assistant Professor

Acknowledgments

I would like to express my sincere gratitude to my supervisor Prof. Tay Wee Peng for his enthusiasm, patience and encouragement. I am deeply grateful to him for giving me the freedom to explore my own research area, and at the same time providing guidance and support throughout the way. I have been fortunate to have him as my supervisor and mentor who arranged weekly tutorial sessions to help me build up the required mathematical background; provided encouragement and advices when my progress was slow; and taught me how to write a scientific paper by editing my draft line by line and pointing out the areas to improve.

I am also thankful to all the lovely friends I met in NTU, including Peng Shuai, Tang Jianhua, Zhang Yi, Cheng Chi, Francois, Hu Wuhua, Wang Yuan, Sun Meng, Che Yueling and Cai Shengming. Thanks for their accompany and they made my postgraduate life colorful. A very special thanks goes to Leng Mei, a co-author, friend, and partner. This thesis would not have been possible without numerous meaningful, inspiring and at the same time "painful" discussions with her. Last but not least, I would like to thank Leng Mei and Tang Peng, together, we build up something that we are truly proud of.

This thesis is dedicated to my beloved parents, grandparents, aunt Yu, uncle Hou Quan and my sister Yu Yan for their love, support and continuous caring.

THIS PAGE INTENTIONALLY LEFT BLANK

Contents

THIS PAGE INTENTIONALLY LEFT BLANK

List of Figures

THIS PAGE INTENTIONALLY LEFT BLANK

List of Tables

THIS PAGE INTENTIONALLY LEFT BLANK

Chapter 1

Introduction

Consider a new contagious disease that has crossed the species boundary and started spreading in a community. In order to quickly search for the cause of the disease, the index cases or the first individuals who contacted the disease have to be found. In another example, a malicious rumor about a terrorist attack spreading on a social network can cause unnecessary public panic and affect financial markets. Law enforcement agencies would want to identify the sources of this rumor. We can model all the above examples as an infection spreading in a network of nodes. In a population network, the infection is the disease that is transmitted between individuals. In the case of a rumor spreading in a social network, the infection is the rumor. The above examples show that identifying the infection sources in a network plays a critical role in many applications.

In this dissertation, we derive infection source estimators to infer the identities of the nodes that started an infection spreading, based only on a one-time observation of the infected nodes in the network. We characterize the optimal source estimators under the maximum likelihood (ML) criterion and the most likely infection path (MLIP) criterion, for specific network classes. More specifically, we derive an estimator under the ML criterion for identifying multiple infection sources under the Susceptible-Infected (SI) spreading model, when the number of infection sources is bounded but unknown a priori. The Jordan center of the infected node set is the node in the network with the smallest maximum distance to any observed infected node. For the MLIP criterion, we show that the Jordan center is an optimal single source estimator for:

- the SI model in an infinite tree network but with limited observations of the infected nodes,
- the Susceptible-Infected-Susceptible (SIS) model in an infinite regular tree network, and
- the Susceptible-Infected-Recovered-Infected (SIRI) model in an infinite tree network with heterogeneous node infection probabilities.

This shows that the Jordan center is a robust infection source estimator over a wide range of infection spreading models in tree networks. In addition, we generalize the Jordan center to the notion of a k-Jordan center set, and show that the k-Jordan center set is the optimal multiple-source estimator under the MLIP criterion for the SI model in an infinite tree network. In this dissertation, we also perform extensive simulation studies to verify the performance of our proposed estimators in general network topologies.

In the rest of this chapter, we first present the background and motivations for the infection sources identification problem, and then discuss the related works. Finally, we give an outline of the thesis contributions.

1.1 Background and Motivations

Modern networks like social, communication and transportation networks, have grown drastically in complexity. Online social networks like Facebook, Twitter and Google+ have grown immensely in popularity over the last ten years [1–5]. The increase in the size and complexity of such social networks have also been driven by increasing use of technologies like smart phones that facilitate more frequent and faster updates and interactions between members of the social network [6]. Physical social networks consisting of communities of individuals have also grown bigger and more complex due to urbanization and technological advancements in transportation [7]. As a result, the world has become more interconnected and an infection (for example, a contagious disease, a computer virus or a rumor) can propagate to a large number of nodes in the underlying network in a short period of time and cause significant damage [8–14]. In the following, we see some real-life examples of an infection spreading in a network.

- **SARS Outbreak in 2002**

In November 2002, the epidemic of Severe Acute Respiratory Syndrome (SARS) started in Guangdong Province, China. Within weeks, SARS spread and infected individuals in 37 countries [15]. During the SARS outbreak from November 2002 to July 2003, more than 8,000 individuals were infected, and more than 700 were killed [16].

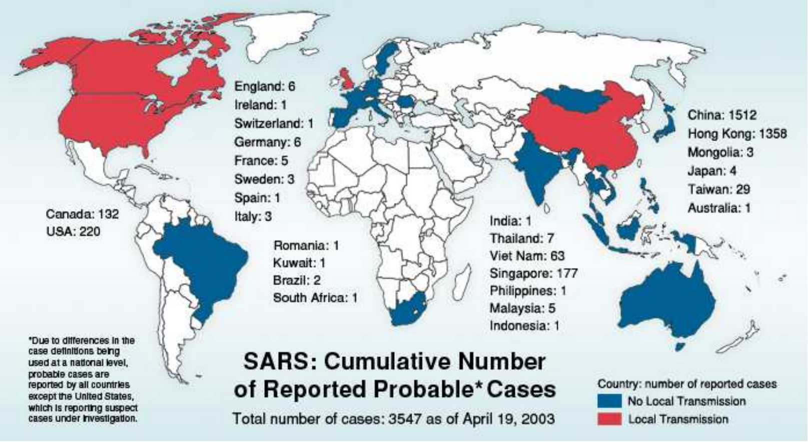

England: 6
Ireland: 1
Switzerland: 1
Germany: 6
France: 5
Sweden: 3
Spain: 1
Italy: 3
China: 1512
Hong Kong: 1358
Mongolia: 3
Japan: 4
Taiwan: 29
Australia: 1
Canada: 132
USA: 220
Romania: 1
Kuwait: 1
Brazil: 2
South Africa: 1
India: 1
Thailand: 7
Viet Nam: 63
Singapore: 177
Philippines: 1
Malaysia: 5
Indonesia: 1
*Due to differences in the case definitions being used at a national level, probable cases are reported by all countries except the United States, which is reporting suspect cases under investigation.
SARS: Cumulative Number of Reported Probable* Cases
Total number of cases: 3547 as of April 19, 2003
Country: number of reported cases
No Local Transmission
Local Transmission

Figure 1-1: Global cumulative distribution of SARS. Source: April 2003, Web, *Canadian Environmental Health Atlas*, http://www.ehatlas.ca/health-care/sars-severe-acute-respiratory-syndrome (accessed 9 August 2014).

- **False Report on Twitter: Explosions at White House**

On April 22, 2013, a false report was posted on Twitter by the hacked account of the Associated Press (AP), a trusted American news source, claiming that "Two Explosions in the White House and Barack Obama is injured". This message was sent to around 2 million followers of AP's and forwarded 1,181 times on Twitter [17].

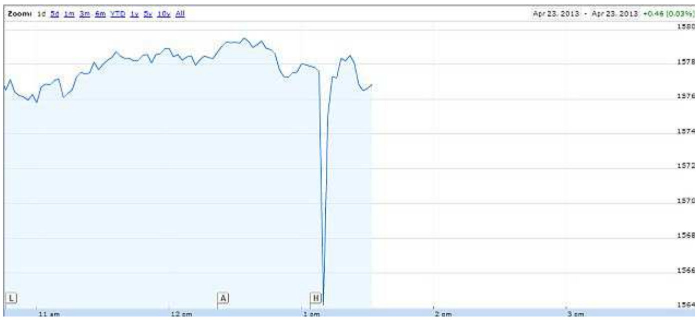


Figure 1-2: Standard & Poor's 500 index fell about 1 percent. Source: Nina Golgowski, 23 April 2013, Web, *Daily Mail*, http://www.dailymail.co.uk/news/article-2313652/AP-Twitter-hackers-break-news-White-House-explosions-injured-Obama.html (accessed 9 August 2014).

Immediately after the spreading of this false report, the Dow Jones Industrial Average plunged 100 points and the Standard & Poor's 500 index fell about 1 percent [17].

- **Panic Salt Buying in China**

After Japan's Fukushima nuclear accident on March 11, 2011, a rumor was posted on Sina Weibo, a Chinese micro-blogging platform, stating that "iodized salt can protect against radiation" and "China's sea salt supplies would be contaminated by Japan's nuclear crisis" [18]. This rumor was quickly spread over Sina Weibo, and results in panic salt-buying in some big cities of China.

Figure 1-3: Panic salt-buying. Source: Chen Yang, 18 March 2011, Photograph, *China Daily*, http://www.chinadaily.com.cn/cndy/2011-03/18/content˙12189705.htm (accessed 9 August 2014).

- **Arizona-Southern California Outages**

The largest power failure in California history started with the loss of a single transmission line On September 8, 2011 [19]. Instantaneous power flow redistributions led to large voltage deviations, resulting in cascading outages affecting parts of Arizona, Southern California, and Baja California, Mexico. This blackout left approximately 2.7 million customers without power [19].

Prompt identification and isolation of the infection sources is crucial in many practical applications in limiting the damage caused by the infection [8, 10], and dealing with the aftermath effectively. For example, quickly identifying the index cases of a contagious disease allows us to study the causes, and hence facilitate the search for antiviral drugs and efficacious therapies. Moreover, by inferring the set of individuals infected by each source, potential containment policies can be formulated to prevent further spreading of the disease due to new index cases [20, 21]. Similarly, identifying the servers or facilities in the computer network or power transmission network that are first infected also allows us to detect the latent points of weaknesses in the network so that preventive measures can be taken to enhance the protection at these points.

Despite the importance of the infection sources identification problem, it has not attracted significant research efforts until the pioneering work of [22] in 2009. The main reason is that the infection sources identification problem is very challenging. Typically, complete data about the infection spreading process, like the first times when the infection is detected at each node, is not available. Even when such detection times are available, the naive method of declaring the first detected node in the network as the sole infection source is often incorrect, as the infection may have a random dormant period, the length of which varies from node to node. For example, the spreading of a disease in a population with individuals having varying degrees of resistance, and hence exhibiting symptoms not necessarily in the order in which they are infected, presents such a problem. In this thesis, we consider this challenging combinatorial problem based only on a one-time observation of the infected nodes in the network. In the next section, we give a brief overview of works related to infection sources identification.

1.2 Related Work

The infection sources identification problem is first studied in [22, 23], which considers the single infection source identification problem in the Susceptible-Infected (SI) model [24–28]. In the SI model, a susceptible node become infected probabilistically, while an infected node retains the infection forever once it is infected. This simple infection spreading model has been widely used in modeling viral epidemics [29–34]. Based only on the knowledge of which nodes are infected and the underlying network structure, a ML estimator based on the linear extensions count of a poset or the number of infection sequences (cf. Section 3.1) is derived to identify the most likely infection source in regular trees. It is shown in [23] that finding a single infection source is a #P-complete problem even in the case where the infection is relatively simple, with infection from an infected node being equally likely to be transmitted to any of its neighbors at each time step. An algorithm for evaluating the single source estimator is proposed in [23], and it is shown to have complexity[1] $O(n)$ for tree networks, where n is the total number of infected nodes. Furthermore, it is shown that this estimator performs well in a very general class of tree networks known as the geometric trees (cf. Section 3.2.4), and identifies the infection source with probability going to one as n increases.

Subsequently, [35] studies the single infection source identification problem in the SI model with additional a priori knowledge of the set of suspect nodes. Based on one snapshot observation of the set of infected nodes, [35] constructs a maximum a posteriori (MAP) infection source estimator to identify the infection source from a set of suspect nodes. It is shown in [35] that, for regular tree networks, the MAP estimator identifies the infection source with probability going to one as n increases and the node degree grows sufficiently large, where n is the number of infected nodes. Moreover, it is shown in [35] that the correct detection probability of the MAP estimator

[1] A function $f(n) = O(g(n))$ if $f(n) \leq cg(n)$ for some constant c and for all n sufficiently large.

monotonically increases with the node degree and decreases with n .

In a lot of applications, it is difficult, if not impossible, to observe all infected nodes in the network. The work [36] considers the single infection source identification problem in the SI model when only a fraction of infected nodes can be observed. It is assumed in [36] that for each of these observed nodes, the infection time of that node, and from which neighboring node the infection comes from is known. An algorithm to find the ML estimator in tree networks is proposed in [36], with a complexity of $O(n)$, where n is the number of nodes in the tree network.

All the works listed above adopt the SI model due to its simplicity. Then the reference [37] considers the single source identification problem in the Susceptible-Infected-Recovered (SIR) model [38–43], where an infected node can recover from an infection with a given probability at each time step, upon which it gains immunity from further infections. The knowledge assumed in [37] is one snapshot observation of the set of all infected nodes, where susceptible nodes and recovered nodes can not be distinguished. Unlike in the SI model considered in [23,35,36], after the introduction of recovery state in the SIR model, the ML estimator is difficult to obtain. Instead, [37] introduces a different statistical criterion where the source node associated with the most likely sample path is proposed as the infection source estimator.[2] It is shown in [37] that, for infinite tree network, the estimator is a Jordan center of the set of infected nodes, i.e., a node that has minimum distance to any infected nodes. When only a subset of infected nodes are observed and the underlying network is an infinite tree, the reference [44] shows that the most likely sample path based estimator is a Jordan center of the set of observed infected nodes under the heterogeneous SIR model. The reference [45] proposes most likely sample path based estimators to identify multiple sources under the SIR model. When the underlying network is a regular tree, the distances between the estimators and the real sources are shown in [45] to be upper bounded by a constant with a high probability.

[2] We adopt the most likely sample path based approach in Chapters 4-6, where we call it the most likely infection path criterion (cf. Section 2.3).

1.3 Our Contributions

1.3.1 Multiple Infection Sources Identification in the SI Model under the ML Criterion

In many applications, there may be more than one infection source in the network. For example, an infectious disease may be brought into a country through multiple individuals. Multiple individuals may collude in spreading a rumor or malicious piece of information in a social network. In Chapter 3, we consider the identification of multiple infection sources when the number of infection sources is unknown a priori. We adopt the same SI infection spreading model as in [23]. We show that unlike the single source identification problem, the multiple sources identification problem is much more complex and cannot be solved exactly even for regular trees. Our main contributions are the following.

(i) For the case of a tree network, and when it is known that there are two infection sources, we derive an estimator for the infection sources based on the infection sequences count. The estimator can be calculated in $O(n^2)$ time complexity, where n is the number of infected nodes.

(ii) When there are at most two infection sources that are at least two hops apart, we derive an estimator for the class of geometric trees based on approximations of the estimator in (i), and we show that our estimator correctly estimates the number of infection sources and correctly identifies the source nodes, with probability going to one as the number of infected nodes increases.

(iii) We derive an estimator for the infection regions of every infection source under a simplifying technical condition.

(iv) For general graphs, when there are at most $k_{\max}$ infection sources, we provide an estimation procedure for the infection sources and infection regions. Simulations suggest that on average, our estimators are within a few hops of the true infection sources in the infection graph.[3]

(v) We test our estimators on real data. The first test is based on real contact tracing data of a patient cluster during the SARS outbreak in Singapore in 2003. Our estimator correctly identifies the number of index cases for the cluster to be one and successfully finds this index case. The second test considers the Arizona-Southern California cascading power outages in 2011. Our estimator correctly identifies the number of outage sources for the main affected power network to be two, and the distance between our estimators and the real sources are within 1 hop. These tests suggest that our estimator has reasonable performance in some applications even though we have adopted a simplistic infection model.

1.3.2 Single Infection Source Identification in the SI Model with Limited Observations under the MLIP Criterion

The assumption of complete observations of the set of infected nodes may not be feasible in a lot of practical scenarios. For example, given the large size of various online social networks like Facebook, it is impossible for a law enforcement agency to analyze all the profiles that post a particular rumor or link to it. Monitoring of network traffic to identify potential terrorist plots can only be done at select nodes in the network. Similarly, an user of Google+ may post a rumor on her profile, and choose to make her post public, which can be seen by any Google+ user, or to restrict access of her post to only a select group of friends. The Google+ user who restricts access to her posting will appear to be uninfected to an observer not amongst the select group of friends. The observer will thus only be able to observe a limited subset of all the infected nodes in the network. Finding the infection source is thus a very challenging problem, and we are often limited to knowing just the topology of the network, and a subset of nodes that are infected. In Chapter 4, we study the single

[3] In general, we do not know the whole underlying network, but rather the subgraph of infected nodes. For example, in the case of a contagious disease spreading in a population, we only perform contact tracing on the patients to construct the connections among them. From our simulation studies, the infection graph typically has an average diameter of more than 27 hops even though the underlying network's diameter is much smaller.

infection source identification problem in the SI model with limited observations of the set of infected nodes. We adopt the same most likely infection path criterion as that in [37].

The reference [36] considers the source identification problem when only a fraction of infected nodes can be observed. However, [36] assumes that for each of these observed nodes, we know the infection time of that node, and from which neighboring node the infection comes from. In an online social network, the posting time of a rumor gives us information about the infection time, but it is often difficult to determine which neighbor the rumor is obtained from, unless the user explicitly references the person she obtained the rumor from. In a physical social network scenario like a disease spreading, the infection times are often not available or inaccurate due to varying degrees of immunity amongst the populace. Therefore, in this thesis, we do not make either of the assumptions used by [36] so that our proposed methods can be used in more general applications that have limited information. Our main contributions are the following.

(i) For tree networks, we derive an estimator for the source node associated with the most likely infection path that yields the observed subset of infected nodes. We propose an algorithm with time complexity $O(n)$ to find the proposed estimator, where n is the number of nodes in the subtree spanning the set of observed infected nodes.

(ii) For general networks, we propose an approximate estimator for the source node associated with the most likely infection path. We then convert the problem into a Mixed Integer Quadratically Constrained Quadratic Program (MIQCQP), which can be solved using standard optimization toolboxes. However, since the MIQCQP has high complexity, we also propose a heuristic algorithm with time complexity $O(n^3)$ to find the proposed estimator, where n is the size of the network.

(iii) We verify the performance of our estimators on various synthetic tree networks, small-world networks, the western states power grid network of the United

States, and part of the Facebook network. In our simulation results, our estimator performs better than the distance, closeness, and betweenness centrality based estimators.

1.3.3 Single Infection Source Identification in the SI, SIR, SIRI and SIS Models under the MLIP Criterion

The SI and SIR models have been widely adopted in the literature due to their simplicity, but these models do not adequately reflect many practical situations in which an infected node recovers and becomes infected again at some future time through either a relapse or reinfection. If an individual recovers from a disease such as bovine tuberculosis or human herpes virus, he may later experience a relapse and exhibit infection symptoms again [46–49]. The spread of such diseases are often modeled using a Susceptible-Infected-Recovered-Infected (SIRI) model [47–49]. On the other hand, if an individual recovers from a disease such as gonorrhea [50], he does not acquire any immunity from his previous infection and may later become reinfected with the same disease. These types of diseases are often modeled using a Susceptible-Infected-Susceptible (SIS) model [29, 51, 52]. A further example of SIRI and SIS type of infection spreading is rumor spreading in an online social network, as monitored by an external agency that does not have access to the full database of the social network. An individual in the network may post a rumor, remove it, and repost the rumor subsequently. If the external agency only has access to a limited set of the most recent postings of each user (for example, due to storage constraints), then trying to identify the source of the rumor based purely on the time-stamps of the rumor posts will lead to an erroneous result.

To the best of our knowledge, finding infection sources under the SIS and SIRI models have not been investigated. Moreover, all the existing works assume that the underlying infection spreading model is known. This knowledge may be difficult to obtain in practice. For example, when a new type of infectious disease breaks out, the spreading characteristics of the disease is usually unclear before its epidemiology is determined. Therefore, it would be highly desirable if a source estimator can be shown to be robust, under a reasonable non-trivial statistical criterion, to the underlying spreading mechanism. Indeed, it is unclear that such an estimator even exists for the SI, SIR, SIRI and SIS models.

In Chapter 5, we adopt the MLIP criterion of [37] to find the optimal infection source estimator. Our main contributions are the following:

(i) For an infection spreading from a single source under the SIRI model, and over an infinite heterogenous tree network, we show that the Jordan center of the infected node set is an optimal infection source estimator.

(ii) We show that if an infection spreads according to the SIS model over an infinite homogeneous regular tree, then the Jordan center is again the optimal infection source estimator.

Since the SI and SIR models are special cases of the SIRI or SIS models, our contributions in items (i) and (ii) show that the Jordan center is robust for the SI, SIR, SIRI and SIS models in regular tree networks. Our simulation results suggest that Jordan center source estimator outperform many other source estimators, regardless of which of the four considered infection spreading models is used. This is somewhat surprising since the SI, SIR, SIRI, and SIS spreading mechanisms are quite different from each other.

We note that finding optimal source estimators is in general NP-hard, and proving the optimality of an estimator is also in general very challenging, with similar results in the current literature restricted to tree networks and the SI or SIR spreading models [23, 37]. Our work is thus a small step towards finding optimal source estimators for the more general SIRI and SIS models.

1.3.4 Multiple Infection Sources Identification in the SI Model under the MLIP Criterion

We propose a heuristic procedure in Chapter 3 to determine multiple infection sources in the SI model based on the single source ML estimator for regular trees, but we have not shown it to be an optimal ML estimator. In Chapter 6, we consider the multiple infection sources identification problem in the SI model under the most likely infection path (MLIP) criterion of [37]. We introduce the concept of a k-Jordan center set, and show that if an infection spreads from $k > 1$ sources in an infinite tree network according to the SI model, then the k-Jordan center set is associated with the most likely infection path that yield the observation, therefore a MLIP-optimal estimator of the infection source set. We heuristically extend the proposed estimator to general graphs, and propose a heuristic algorithm to find the k-Jordan center set. To verify the performance of our estimator, we perform extensive simulations on random trees, part of the Facebook network, and the western states power grid network of the United States. In our simulation results, our estimator consistently achieve the lowest average error distance compared to the distance, closeness, and betweenness centrality based heuristics.

Parts of this thesis have appeared in [53–55], which studied the problem of estimating multiple infection sources in the SI model under the ML criterion; in [56, 57], which investigated in the problem of estimating a single infection source in the SI model with limited observations under the MLIP criterion; in [58], which studied the infection source identification problem in the SIS model under the MLIP criterion; and in [59], which studied the infection sources identification problem in the SI, SIR, SIRI and SIS models under the MLIP criterion.

1.4 Thesis Outline

In Chapter 2, we present four infection spreading models and discuss the formulation of the infection sources identification problem under two statistical criteria. In Chapter 3, we study the multiple infection sources identification in the SI model under the ML criterion. In Chapter 4, we consider the single infection source identification in the SI model with limited observation under the MLIP criterion. We investigate the single infection source identification in the SI, SIR, SIRI and SIS models under the MLIP criterion in Chapter 5. In Chapter 6, we consider the multiple infection sources identification under the MLIP criterion. Finally, we conclude and discuss some future research directions in Chapter 7.

Chapter 2

The Basic Model

In this chapter, we introduce four infection spreading models considered in this thesis, and discuss formulation of the infection sources identification problem under two statistical criteria.

2.1 Infection Spreading Models

We model the underlying network over which an infection spreads as an undirected graph $G = (V, E)$, where V is the set of nodes and E is the set of edges. If there is an edge connecting two nodes, we say that they are neighbors. In a computer network, the graph G models the interconnections between computers in the network. In the example of a population or a social network, V is the set of individuals, while an edge in E represents a relationship between two individuals. We define an infection to be a property that a node in G possesses, and can be transmitted to a neighboring node. At time 0, suppose that there are $k \geq 1$ nodes $S^* \subset V$ possessing the infection, which we call the *infection sources*. Then the infection spreads from the infection sources to their neighboring nodes according to certain infection spreading model. In this thesis, we consider the following four infection spreading models.

- **SI model**: In the SI model, each node takes on one of 3 possible states: *susceptible* (**s**), *infected* (**i**) and *non-susceptible* (**n**). At any time, if a node is infected, we say that it is in state **i**. The set of uninfected nodes that have infected neighbors are in state **s**, and are called susceptible nodes. In the SI model, an infected node remains infected forever, and a susceptible node becomes infected probabilistically. All other nodes are in state **n**, and are called non-susceptible nodes. A non-susceptible node has probability zero of becoming infected in the future.

- **SIR model**: In the SIR model, the possible node states are *susceptible* (**s**), *infected* (**i**), *non-susceptible* (**n**), and *recovered* (**r**). The only difference with the SI model is that an infected node in state **i** may recover to state **r**. A recovered node then stays in the recovered state **r** forever. In other words, a recovered node will never become infected again.
- **SIRI model**: The possible nodes states in the SIRI model are the same as for the SIR model. The difference from the SIR model is that a recovered node (in state **r**) may become infected again at a future time. This infection relapse is spontaneous, and can take place even if the node does not have any infected neighbors. Here, we reserve the state **s** for those nodes that have infected neighbors and have never been infected before.
- **SIS model**: In the SIS model, the possible node states are *susceptible* (**s**), *infected* (**i**) and *non-susceptible* (**n**). This model describes a more complicated spreading process where once an infected node recovers from the infection, it immediately becomes a susceptible node (if it has at least one infected neighbor) or non-susceptible node (if it does not have any infected neighbor). There is therefore no *recovered* state in this model.

In this thesis, we consider both *continuous time* and *discrete time* infection spreading models. In the continuous time SI model, we adopt the same infection spreading process as that in [23], where the time taken for an infected node to infect a susceptible neighbor is exponentially distributed with rate 1. In the discrete time SI, SIR, SIRI and SIS models, We assume that the infection spreading process is a discrete time Markov process with probability measure $\mathbb{P}$. For any node $v \in V$, we let

$p_{\mathbf{s}}(v)$, $p_{\mathbf{i}}(v)$ and $p_{\mathbf{r}}(v)$ be the probability for v to become infected in the next time slot conditioned on v being susceptible, infected, or recovered in current time slot, respectively.

At some time t, we observe the set (or a subset) of nodes that are currently infected. The observed set of infected nodes is denoted as $V_{\mathbf{i}}$ and is assumed to be non-empty.[1] We assume that the elapsed time t is unknown. Based only on knowledge of $V_{\mathbf{i}}$ and topology of the underlying graph G, we want to estimate the infection sources under certain statistical criterion. In the following, we present two statistical criteria considered in this thesis.

2.2 Infection Sources Identification under the ML Criterion

In the continuous time SI model, the set of k infection sources must be a subset of V_i. We do not assume any a prior knowledge of the infection sources. Therefore, we assume a uniform prior probability of the set of infection sources among all sets of k nodes in V_i. This leads to the definition of the Maximum Likelihood (ML) estimator that maximizes the conditional probability of observing the infected set $V_{\mathbf{i}}$, given by

$$\hat{S}_{ML} \in \arg\max_{\substack{S \subset V \\ |S|=k}} P(V_{\mathbf{i}} \mid S^* = S),$$

where $|S|$ is the number of elements in set S and $P(V_{\mathbf{i}} \mid S^* = S)$ is the probability of observing $V_{\mathbf{i}}$ conditioned on S being the infection sources. The reference [23] constructs a single source ML estimator in the SI model, when the underlying network G is a regular tree and the set of all infected nodes can be observed. Based on this single source ML estimator, we consider the problem of identifying multiple infection sources in the SI model in Chapter 3.

[1] As $V_{\mathbf{i}}$ denotes the observed set of infected nodes at time t, its notation should include t. However, in order to avoid cluttered expressions, we drop t in our notations.

2.3 Infection Sources Identification under the MLIP Criterion

In the discrete time SI, SIR, SIRI and SIS models, we let $\mathbf{X}(u,t)$ be a random variable denoting the state of a node u in time slot t. Let $\mathbf{X}^t = \{\mathbf{X}(u,\tau) : u \in V, 1 \leq \tau \leq t\}$ be the collection of the states of all nodes in V from time 1 to t. A realization $X^t = \{X(u,\tau) : u \in V, 1 \leq \tau \leq t\}$ of $\mathbf{X}^t$ is an *infection path*. We say that an infection path X^t is *consistent* with $V_{\mathbf{i}}$ if all $u \in V_{\mathbf{i}}$ is observed to be infected while no other nodes in V is observed to be infected at time t in X^t. Conditioned on S being the set of infection sources, we let $\mathcal{X}_S$ be the set of all possible infection paths consistent with $V_{\mathbf{i}}$, and $\mathcal{T}_S$ be the set of the corresponding feasible elapsed times. The ML estimator is given by

$$\hat{S}_{ML} \in \arg\max \sum_{\substack{S\subset V,|S|=k\\ t\in\mathcal{T}_S, X^t\in\mathcal{X}_S}} \mathbb{P}(\mathbf{X}^t = X^t \mid S^* = S).$$

For the ML estimator, we need to consider the probability of all possible infection paths, however, it was shown in [23] that finding the single source ML estimator in the SI model with complete observations in a general graph network is a #P-complete problem. Obtaining the ML estimator is even more challenging in the SI model with limited observations, or in more complex infection spreading models including the SIR, SIRI and SIS models. It would be highly desirable if we can construct source estimators for these cases under a reasonable non-trivial statistical criterion. The reference [37] proposed to estimate the infection sources as the nodes associated with a *most likely infection path* out of all possible infection paths that are consistent with $V_{\mathbf{i}}$, given by,

$$\hat{S}_{MLIP} \in \arg \max_{\substack{S\subset V,|S|=k\\ t\in\mathcal{T}_S, X^t\in\mathcal{X}_S}} \mathbb{P}(\mathbf{X}^t = X^t \mid S^* = S). \tag{2.1}$$

In order to derive insights into an optimal sources estimator for (2.1), we first consider the single source identification in the SI model with limited observations in Chapter 4. We then investigate the single source identification in the SI, SIR, SIRI and SIS models in Chapter 5. Finally, we consider the multiple sources identification in the SI model in Chapter 6.

THIS PAGE INTENTIONALLY LEFT BLANK

Chapter 3

Multiple Infection Sources Identification in the SI Model under the ML Criterion

In a lot of applications, there might be more than one infection source in a network, including an infectious disease spreading in a community and a rumor spreading in a social network (cf. Section 1.3.1). In this chapter, We consider the problem of identifying multiple infection sources in a network when the number of sources is unknown a priori.

3.1 Problem Formulation

In this section, we describe our model and assumptions, introduce some notations, and present some preliminary results. We assume the infection spreads according to the SI model as discussed in Section 2.1. We adopt the same infection spreading process as in [23], where the time taken for an infected node to infect a susceptible neighbor is exponentially distributed with rate 1. All infections are independent of each other. Therefore, if a susceptible node has more than one infected neighbors and subsequently becomes infected, its infection is transmitted by one of its infected neighbors, chosen uniformly at random. For mathematical convenience, we also assume that G is large so that boundary effects can be ignored in our analysis.

Suppose that at time 0, there are $k \geq 1$ nodes in the infected node set $S^* = \{s_1, \ldots, s_k\} \subset V$. These are the **infection sources** from which all other nodes get infected. Suppose that after the infection process has run for some time, we observe the set of all n infected nodes $V_{\mathbf{i}}$. Typically, n is much larger than k. These nodes form an **infection graph** $G_{\mathbf{i}} = (V_{\mathbf{i}}, E_{\mathbf{i}})$, which is a subgraph of G. Let $\mathcal{A}^* = \cup_{i=1}^k A_i$ be a partition of the infected nodes $V_{\mathbf{i}}$ so that $A_i \cap A_j = \emptyset$ for $i \neq j$, with each partition A_i being connected in $G_{\mathbf{i}}$, and consisting of the nodes whose infection can be traced back to the source node s_i. The set A_i is called the **infection region** of s_i, and we say that $\mathcal{A}^*$ is the **infection partition**. Given $G_{\mathbf{i}}$, our objective is to infer the sources of infection S^* and to estimate $\mathcal{A}^*$. In addition, if we do not have prior knowledge of the number of infection sources k, we also aim to infer the number of infection sources. Without loss of generality, we assume that $G_{\mathbf{i}}$ is connected, otherwise the same estimation procedure can be performed on each of the components of the graph. We also assume that there are at most $k_{\max}$ infection sources, i.e., the number of infection sources $k \leq k_{\max}$. The length of the shortest path between a pair of nodes u and v is denoted as $d(u, v)$. From a a practical point of view, if two infection sources are close to each other, we can ignore either one of them and treat the infection as spreading from a single source. Therefore, we are interested in cases where the infection sources are separated by a minimum distance. These assumptions are summarized in the following.

Assumption 3.1. The number of infection sources is at most $k_{\max}$, and the infection graph $G_{\mathbf{i}}$ is connected.

Assumption 3.2. For all $s_i, s_j \in S^*$, the length of the shortest path between them $d(s_i, s_j) \geq \tau$, where τ is a constant greater than 1.

Assumption 3.3. Every node in G has bounded degree, with d_* being the maximum node degree.

Suppose that our priors for S^* and $\mathcal{A}^*$ are uniform over all possible realizations, and let $\mathbb{P}$ be the probability measure of the infection process. We seek S and $\mathcal{A}$ that maximize the posterior probability of S^* and $\mathcal{A}^*$ given $G_{\mathbf{i}}$,

$$\mathbb{P}(S^* = S, \mathcal{A}^* = \mathcal{A} \mid G_{\mathbf{i}}) \propto P(G_{\mathbf{i}} \mid S) P(\mathcal{A} \mid S, G_{\mathbf{i}}), \tag{3.1}$$

where $P(G_{\mathbf{i}} \mid S)$ is the probability of observing $G_{\mathbf{i}}$ if S is the set of infection sources, and $P(\mathcal{A} \mid S, G_{\mathbf{i}})$ is the probability that $\mathcal{A}^* = \mathcal{A}$ conditioned on S being the infection source set and the infection graph being $G_{\mathbf{i}}$.

For any source set S, let an **infection sequence** $\sigma = (\sigma_1, \ldots, \sigma_{n-k})$ be a sequence of the nodes in $G_{\mathbf{i}}$, excluding the the k source nodes in S, arranged in ascending order of their infection times (note that with probability one, no two infection times are the same). For any sequence to be an infection sequence, a necessary and sufficient condition is that any infected node σ_i, $i = 1, \ldots, n-k$, has a neighbor in $S \cup \{\sigma_1, \ldots, \sigma_{i-1}\}$. We call this the *infection sequence property.* An example is shown in Figure 3-1, where the shaded nodes are the infected nodes which form the infection graph $G_{\mathbf{i}}$. Infection sources are $S = \{s_1, s_2\}$. The sequence (u_2, u_4) is an infection sequence, but (u_4, u_2) is not. The probability of the infection sequence $\sigma = (u_2, u_4)$ is then given by $P(\sigma \mid S) = \frac{2}{4} \times \frac{1}{4} = \frac{1}{8}$. The first fraction $\frac{2}{4}$ is obtained by observing that when only s_1 and s_2 are infected, there are four edges (s_1, u_2), (s_1, u_3), (s_2, u_2), and (s_2, u_5) for the infection to spread. The infection is equally likely to spread along any of these four edges, out of which two results in the infection of node u_2. After u_2 is infected, there are 4 edges over which the infection can spread and this corresponds to the fraction $\frac{1}{4}$.

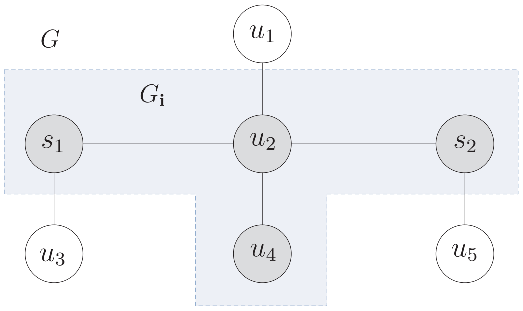


Figure 3-1: Example of an infection sequence.

Let $\Omega(G_{\mathbf{i}}, S)$ be the set of infection sequences for an infection graph $G_{\mathbf{i}}$ and source set S, and let $C(S \mid G_{\mathbf{i}}) = |\Omega(G_{\mathbf{i}}, S)|$ be the number of infection sequences. We have

$$P(G_{\mathbf{i}} \mid S) = \sum_{\sigma \in \Omega(G_{\mathbf{i}}, S)} P(\sigma \mid S), \tag{3.2}$$

where $P(\sigma \mid S)$ is the probability of obtaining the infection sequence σ conditioned on S being the infection sources.

Evaluating the expression (3.2) and maximizing (3.1) for a general $G_{\mathbf{i}}$ is a computationally hard problem as it involves combinatorial quantities. As shown in [23], if G is a regular tree and $|S|= 1$, $P(G_{\mathbf{i}} \mid S)$ is proportional to $|\Omega(G_{\mathbf{i}}, S)|$, which is equivalent to the number of linear extensions of a poset. It is known that evaluating the linear extensions count is a #P-complete problem [60]. As such, we will make a series of approximations to simplify the problem, and present numerical results in Section 3.4 to verify our algorithms. The first approximation we make is to evaluate the estimators

$$\hat{S} = \arg \max_{\substack{S \subset V_{\mathbf{i}} \\ |S| \le k_{\max}}} P(G_{\mathbf{i}} \mid S), \tag{3.3}$$

$$\hat{\mathcal{A}}(\hat{S}) = \arg\max_{\mathcal{A}} P(\mathcal{A} \mid \hat{S}, G_{\mathbf{i}}), \tag{3.4}$$

instead of the exact maximum a posteriori (MAP) estimators for (3.1). Even with this approximation, the optimal estimators are difficult to compute exactly, and may not be unique in general. Therefore, our goal is to design algorithms that are approximately optimal but computationally efficient. In Section 3.2, we make further approximations and design algorithms to evaluate the estimators $\hat{S}$ and $\hat{\mathcal{A}}(\hat{S})$ when G is a tree. In Section 3.3, we consider the case when G is a general graph. For the reader's convenience, we summarize some notations commonly used in this chapter in Table 3.1. Several notations have been introduced previously, while we formally define the remaining ones in the sequel where they first appear.

Table 3.1: Summary of some notations used in Chapter 3.

Symbol	**Definition**
G	underlying network
$d(u,v)$	length of the shortest path between u and v
$\mathcal{N}_G(u)$	set of neighbors of u in G
$\deg_G(u)$	number of neighbors of node u in G
$G_\mathbf{i}$	infection graph with n infected nodes
S^*	infection sources
$\mathcal{A}^*$	infection partition of an infection graph $G_\mathbf{i}$
A_i	infection region of an infection source s_i
$\Omega(G_\mathbf{i}, S)$	set of infection sequences for an infection graph $G_\mathbf{i}$ and source set S
$C(S \mid G_\mathbf{i})$	$= \lvert\Omega(G_\mathbf{i}, S)\rvert$

Symbol	**Definition** (defined implicitly w.r.t. $G_\mathbf{i}$)
$\rho(u,v)$	path between u and v in the infection graph $G_\mathbf{i}$
$T_v(S)$	tree in $G_\mathbf{i}$, rooted at v w.r.t. source set S
$T_M(S)$	$= \cup_{v\in M} T_v(S)$, where M is a subset of nodes
$I_i(\xi; S)$	$= \sum_{j\leq i} \lvert T_{\xi_j}(S)\rvert$, where ξ is a sequence of nodes
$I_i^*(s_1, s_2)$	total number of nodes in the i biggest trees in $\{T_u(s_1, s_2) : u \in \rho(s_1, s_2)\}$

3.2 Identifying Infection Sources and Regions for Trees

In this section, we consider the problem of estimating the infection sources and regions when the underlying network G is a tree. We first derive an estimator for the infection partition in (3.4), given any source node set S and $G_\mathbf{i}$. Then, we derive an estimator based on the number of infection sequences. Next, we consider the case where there are two infection sources, propose approximations that allow us to compute the estimator with reasonable complexity, and show that our proposed estimator works well in an asymptotically large geometric tree under some simplifying assumptions. In most practical applications, the number of infection sources is not known a priori. We present a heuristic algorithm for general trees to estimate the infection sources when the number of infection sources is unknown, but bounded by $k_{\max}$.

3.2.1 Infection Partition with Multiple Sources

In this section, we derive an approximate infection partition estimator for (3.4) given any infection source set S. This estimator is exact under a simplifying technical condition given in Theorem 3.1 below, the proof of which is provided in Section 3.5.

Theorem 3.1. Suppose that G is a tree with infection sources S, and R is the subgraph of $G_{\mathbf{i}}$ consisting of all paths between any pair of nodes in S. If any two paths in R do not intersect except possibly at nodes in S, then the optimal estimator $\hat{\mathcal{A}}(S)$ for the infection partition is a Voronoi partition of the graph $G_{\mathbf{i}}$, where the centers of the partitions are the infection sources S.

A Voronoi partition may not produce the optimal estimator for the infection partition in a general infection graph. However, it is intuitively appealing as nodes closer to a particular source are more likely to be infected by that source. For simplicity, we will henceforth use the Voronoi partition of the infection graph $G_{\mathbf{i}}$ as an estimator for $\mathcal{A}^*$, and present simulation results in Section 3.4 to verify its performance. We will also see in Section 3.2.5 that this approximation allows us to design an infection source estimation algorithm with low complexity.

3.2.2 Estimation of Infection Sources

We now consider the problem of estimating the set of infection sources S^*. When $|S^*|= 1$, our estimation problem reduces to that in [23], which considers only the single source infection problem. In the following, we introduce some notations, and briefly review some relevant results from [23].

A path between any two nodes u and v in the tree $G_{\mathbf{i}}$ is denoted as $\rho(u, v)$. For any set of nodes S in $G_{\mathbf{i}}$, consider the connected subgraph $R \subset G_{\mathbf{i}}$ consisting of all paths between any pair of nodes in S. Treat this subgraph as a "super" node, with the tree $G_{\mathbf{i}}$ rooted at this "super" node. For any node $v \in G_{\mathbf{i}} \backslash R$, we define $T_v(S)$ to be the tree rooted at v with the path from v to R removed. For $v \in R$, we define $T_v(S)$ to be the tree rooted at v so that all edges between v and its neighbors in R

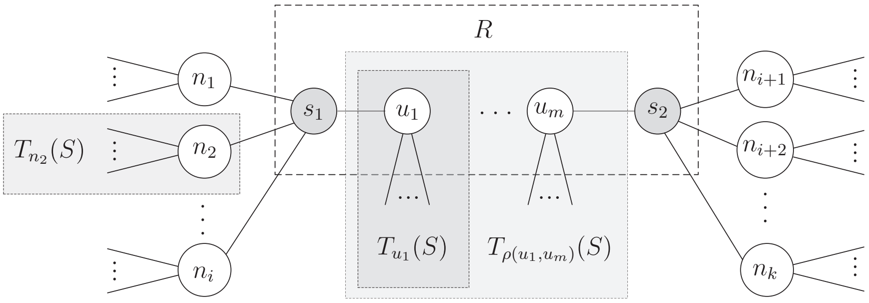


Figure 3-2: A sample infection graph with $S = \{s_1, s_2\}$.

are removed.[1] We say that $T_v(S)$ is the tree rooted at v with respect to (w.r.t.) S. For any subset of nodes $M \subset G_{\mathbf{i}}$, we let $T_M(S) = \cup_{v \in M} T_v(S)$. An illustration of these definitions is shown in Figure 3-2. If $S = \{s_1, \ldots, s_k\}$, we will sometimes use the notation $T_v(s_1, \ldots, s_k)$ instead.

Recall that $C(S \mid G_{\mathbf{i}})$ is the number of infection sequences if S is the infection source set. If there is a single infection source node $S = \{s\}$, and G is a regular tree where each node has the same degree, it is shown in [23] that the MAP estimator for the infection source is obtained by evaluating $\hat{S} = \arg\max_{v \in G_{\mathbf{i}}} C(v \mid G_{\mathbf{i}})$, which seeks to maximize $C(v \mid G_{\mathbf{i}})$ over all nodes. Therefore, it has been suggested that $C(v \mid G_{\mathbf{i}})$ can be used as the infection source estimator for general trees. The following result is provided in [23].

Lemma 3.1. Suppose that $G_{\mathbf{i}}$ is a tree. For any node $s \in G_{\mathbf{i}}$, we have

$$C(s \mid G_{\mathbf{i}}) = n! \prod_{u \in G_{\mathbf{i}}} |T_u(s)|^{-1}. \tag{3.5}$$

We observe that each term $|T_u(s)|$ in the product on the right hand side (R.H.S.) of (3.5) is the number of nodes in the subtree $T_u(s)$ (and which appears when we account for the number of permutations of these nodes). We can think of the terms in the product being ordered according to the infection spreading sequence, i.e., each time we reach a particular node u, we include terms corresponding to the nodes u

[1] As $T_v(S)$ is defined on $G_{\mathbf{i}}$, its notation should include $G_{\mathbf{i}}$. However, in order to avoid cluttered expressions, we drop $G_{\mathbf{i}}$ in our notations. Confusion will be avoided through the context in which these trees are referenced.

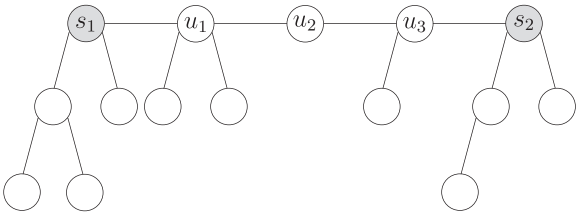


Figure 3-3: A sample infection graph with $S = \{s_1, s_2\}$. Given an infection sequence $\sigma = (u_3, u_1, u_2) \in \Omega(\rho(s_1, s_2), \{s_1, s_2\})$, we can find the corresponding reverse infection sequence $\xi = (u_2, u_1, u_3)$. We have $I_1(\xi; s_1, s_2) = |T_{u_2}(s_1, s_2)| = 1$, $I_2(\xi; s_1, s_2) = |T_{u_2}(s_1, s_2)| + |T_{u_1}(s_1, s_2)| = 4$, $I_3(\xi; s_1, s_2) = |T_{u_2}(s_1, s_2)| + |T_{u_1}(s_1, s_2)| + |T_{u_3}(s_1, s_2)| = 6$.

can potentially infect. This interpretation is useful in helping us understand the characterization in Lemma 3.2 for the case where there are two infection sources.

To compute $C(v \mid G_{\mathbf{i}})$, an $O(n)$ algorithm based on Lemma 3.1 was provided in [23]. We call this algorithm the Single Source Estimation (SSE) algorithm. We refer the reader to [23] for details about the implementation of the algorithm. Although finding $\hat{S}$ by maximizing $C(s \mid G_{\mathbf{i}})$ is exact only for regular trees, it was shown in [23] that this estimator has good performance for other classes of trees. In particular, if G is a geometric tree (cf. Section 3.2.4), then the probability, conditioned on $S^* = \{s\}$, of correctly identifying s using $C(s \mid G_{\mathbf{i}})$ goes to one as $n \to \infty$. Inspired by this result, we propose estimators based on quantities related to $C(S \mid G_{\mathbf{i}})$ for cases where $|S^*| > 1$. In the following, we first discuss the case where $|S^*| = 2$, and extend the results to the general case where $|S^*|$ is unknown in Section 3.2.5. We then numerically compare our proposed algorithms with a modified SSE algorithm adapted for finding multiple sources in Section 3.4.

3.2.3 Two Infection Sources

In this section, we assume that there are two infection sources $S = \{s_1, s_2\}$. Given two nodes u and v in $G_{\mathbf{i}}$, suppose that $|\rho(u, v)| = m$. For any permutation $\xi = (\xi_1, \ldots, \xi_m)$ of the nodes in $\rho(u, v)$, let

$$I_i(\xi; s_1, s_2) = \sum_{j \le i} |T_{\xi_j}(s_1, s_2)| \tag{3.6}$$

be the total number of nodes in the trees rooted at the first i nodes in the permutation ξ. We have the following characterization for $C(s_1, s_2 \mid G_{\mathbf{i}})$, whose proof is given in Section 3.5.

Lemma 3.2. Suppose that $G_{\mathbf{i}}$ is a tree. Consider any two nodes s_1 and s_2 in $G_{\mathbf{i}}$, and suppose that $\rho(s_1, s_2) = (s_1, u_1, \ldots, u_m, s_2)$. We have

$$C(s_1, s_2 \mid G_{\mathbf{i}}) = (n-2)! \cdot q(u_1, u_m; s_1, s_2) \cdot \prod_{u \in G_{\mathbf{i}} \setminus \rho(s_1, s_2)} |T_u(s_1, s_2)|^{-1}, \tag{3.7}$$

where for $1 \leq i \leq j \leq m$, $q(u_i, u_j; s_1, s_2)$ satisfies the following recursive relationship

$$q(u_i, u_j; s_1, s_2) = |T_{\rho(u_i, u_j)}(s_1, s_2)|^{-1}(q(u_{i+1}, u_j; s_1, s_2) + q(u_i, u_{j-1}; s_1, s_2)) \text{ for } i < j, \tag{3.8}$$

with $q(v, v; s_1, s_2) = |T_v(s_1, s_2)|^{-1}$ for all $v \in \rho(u_1, u_m)$. Furthermore, we have

$$q(u_1, u_m; s_1, s_2) = \sum_{\xi \in \Gamma(u_1, u_m)} \prod_{i=1}^{m} I_i(\xi; s_1, s_2)^{-1}, \tag{3.9}$$

and $\Gamma(u_1, u_m)$ is the set of all permutations $\xi = (\xi_1, \ldots, \xi_m)$ of nodes in $\rho(u_1, u_m)$ such that $(\xi_m, \ldots, \xi_1)$ is an infection sequence starting from s_1 and s_2 and resulting in $\rho(s_1, s_2)$.

The characterization for $C(s_1, s_2 \mid G_{\mathbf{i}})$ is similar to that for the single source case in (3.5), except for the additional $q(u_1, u_m; s_1, s_2)$ term. We first clarify the meaning of $\Gamma(u_1, u_m)$. Given any infection sequence σ that starts with $\{s_1, s_2\}$ and results in $\rho(s_1, s_2)$, i.e., $\sigma = (\sigma_1, \ldots, \sigma_m) \in \Omega(\rho(s_1, s_2), \{s_1, s_2\})$, we can find a permutation $\xi = (\xi_1, \ldots, \xi_m)$ of nodes in $\rho(u_1, u_m)$ such that $\xi_i = \sigma_{m-i+1}$ for $i = 1, \ldots, m$. In other words, ξ can be interpreted as the *reverse* infection sequence corresponding to σ. Then $\Gamma(u_1, u_m)$ is the set of all such reverse infection sequences corresponding to $\Omega(\rho(s_1, s_2), \{s_1, s_2\})$. We show an illustration of these definitions in Figure 3-3. Each term $|T_u(s)|$ in the product in the R.H.S. of (3.5) can be interpreted as the number of nodes that can be infected via u once u has been infected. Similarly, the sum in

(3.9) is over all possible reverse infection sequences ξ of the nodes in $\rho(u_1, u_m)$, and each term $I_i(\xi; s_1, s_2)$ in the product within the sum is the number of nodes that can be infected once $\xi_{i+1}, \ldots, \xi_m$ have been infected.

By utilizing Lemma 3.2, we can compute $C(u, v \mid G_{\mathbf{i}})$ for any two nodes u and v in $G_{\mathbf{i}}$ by evaluating $|T_w(u, v)|$ for all nodes $w \in G_{\mathbf{i}}$, and the quantity $q(u_1, u_m; u, v)$, where $\rho(u, v) = (u, u_1, \ldots, u_m, v)$. With Assumption 3.3, Algorithm 3.1 allows us to compute $f_w(u) = |T_w(u)|$ and $g_w(u) = \prod_{v \in T_w(u)} |T_v(u)|$ for all neighbors u of w, and for all $w \in G_{\mathbf{i}}$ in $O(n)$ time complexity. To do this, we first choose any node $r \in G_{\mathbf{i}}$, and consider $G_{\mathbf{i}}$ as a directed tree with r as the root node, and with edges in $G_{\mathbf{i}}$ pointing away from r. The neighborhood $\mathcal{N}_{G_{\mathbf{i}}}(w)$ of a node w is the set of all neighbors of w in $G_{\mathbf{i}}$. Let $\mathrm{pa}(w)$ and $\mathrm{ch}(w)$ be the parent and the set of children of w in the directed tree $G_{\mathbf{i}}$, respectively. Starting from the leaf nodes, let each non-root node $w \in G_{\mathbf{i}}$ pass two messages containing $f_w(\mathrm{pa}(w))$ and $g_w(\mathrm{pa}(w))$ to its parent. Each node stores the values of these two messages from each of its children, and computes its two messages to be passed to its parent. When r has received all messages from its children, a reverse sweep down the tree is done so that at the end of the algorithm, every node $w \in G_{\mathbf{i}}$ has stored the values $\{f_u(w), g_u(w) : u \in \mathcal{N}_{G_{\mathbf{i}}}(w)\}$. The algorithm is formally described in Algorithm 3.1. The last product term on the R.H.S. of (3.7) can then be computed using

$$g(s_1, s_2) = \prod_{w \in \rho(s_1, s_2)} \prod_{x \in \mathcal{N}_{G_{\mathbf{i}}}(w) \backslash \rho(s_1, s_2)} g_x(w), \tag{3.10}$$

and taking its reciprocal.

To compute $C(s_1, s_2 \mid G_{\mathbf{i}})$ in (3.7), we still need to compute $q(u_1, u_m; s_1, s_2)$. The recurrence (3.8) allows us to compute $q(u_1, u_m; s_1, s_2)$ for all $s_1, s_2 \in G_{\mathbf{i}}$ in $O(n^2 d_*^2)$ complexity, where d_* is the maximum node degree. The computation proceeds by first considering each pair of neighbors (u, v). Both nodes have at most d_* neighbors each, so that we need to evaluate $q(u, v; s_1, s_2)$ for all $s_1 \in \mathcal{N}_{G_{\mathbf{i}}}(u) \backslash \rho(u, v)$ and $s_2 \in \mathcal{N}_{G_{\mathbf{i}}}(v) \backslash \rho(u, v)$. This requires $O(d_*^2)$ computations. The computed values and $T_{\rho(u,v)}(s_1, s_2)$ are stored in a hash table. In the next step, we repeat the same pro-

Algorithm 3.1 Tree Sizes and Products Computation

```
**Inputs**: $G_{\mathbf{i}}$
Choose any node $r \in G_{\mathbf{i}}$ as the root node.
**for** $w \in G_{\mathbf{i}}$ **do**
    Store received messages $f_x(w)$ and $g_x(w)$, for each $x \in \mathrm{ch}(w)$.
    **if** $w$ is a leaf **then**
        $f_w(\mathrm{pa}(w)) = 1$
        $g_w(\mathrm{pa}(w)) = 1$
    **else**
        $f_w(\mathrm{pa}(w)) = \sum_{x \in \mathrm{ch}(w)} f_x(w) + 1$
        $g_w(\mathrm{pa}(w)) = f_w(\mathrm{pa}(w)) \cdot \prod_{x \in \mathrm{ch}(w)} g_x(w)$
    **end if**
    Store $f_{\mathrm{pa}(w)}(w) = n - f_w(\mathrm{pa}(w))$.
    Pass $f_w(\mathrm{pa}(w))$ and $g_w(\mathrm{pa}(w))$ to $\mathrm{pa}(w)$.
**end for**
Set $g_{\mathrm{pa}(r)}(r) = 1$.
**for** $w \in G_{\mathbf{i}}$ **do**
    Store received message $g_{\mathrm{pa}(w)}(w)$ from $\mathrm{pa}(w)$.
    **if** $w$ is not a leaf **then**
        **for** $x \in \mathrm{ch}(w)$ **do**
            $g_w(x) = f_w(x) \cdot g_{\mathrm{pa}(w)}(w) \cdot \prod_{y \in \mathrm{ch}(w) \setminus \{x\}} g_y(w)$
            Pass $g_w(x)$ to $x$.
        **end for**
    **end if**
**end for**
```

cedure for node pairs that are two hops apart, and so on until we have considered every pair of nodes in $G_{\mathbf{i}}$. Note that for a path $(u_1, \ldots, u_m)$ and s_1, s_2 neighbors of u_1 and u_m respectively, $q(u_1, u_m; s_1, s_2)$ can be computed in constant time from (3.8) as $q(u_2, u_m; s_1, s_2) = q(u_2, u_m; u_1, s_2)$ and $q(u_1, u_{m-1}; s_1, s_2) = q(u_1, u_{m-1}; s_1, u_m)$. A similar remark applies for the computation of $|T_{\rho(u_1,u_m)}(s_1, s_2)|$. In addition, each lookup of the hash table takes $O(1)$ complexity since $G_{\mathbf{i}}$ is known and collision-free hashing can be used. Therefore, the overall complexity is $O(n^2 d_*^2)$. The algorithm to compute the infection sources estimator is formally given in Algorithm 3.2. We call this the Two Source Estimation (TSE) algorithm, and it forms the basis of our algorithm for multiple sources estimation in the sequel.

Algorithm 3.2 Two Source Estimation (TSE)

```
Input: $G_{\mathbf{i}}$
Let $(s_1^*, s_2^*)$ be the maximizer of $C(\cdot, \cdot \mid G_{\mathbf{i}})$. Set $C^* = 0$.
for $d = 1$ to diameter of $G_{\mathbf{i}}$ do
    for each $s_1 \in G_{\mathbf{i}}$ do
        for each $s_2$ such that $d(s_1, s_2) = d$ do
            Let $\rho(s_1, s_2) = (s_1, u_1, \ldots, u_{d-1}, s_2)$.
            if $d = 1$ then
                $q(u_1, u_{d-1}; s_1, s_2) = 1$.
            else if $d = 2$ then
                Store $q(u_1, u_1; s_1, s_2) = |T_{u_1}(s_1, s_2)|^{-1}$ and $|T_{u_1}(s_1, s_2)|$.
            else
                Look up $|T_{\rho(u_1,u_{d-2})}(s_1, u_{d-1})|$, $q(u_2, u_{d-1}; u_1, s_2)$, and $q(u_1, u_{d-2}; s_1, u_{d-1})$.
                Store
                    $$|T_{\rho(u_1,u_{d-1})}(s_1, s_2)| = |T_{\rho(u_1,u_{d-2})}(s_1, u_{d-1})| \cdot |T_{u_{d-1}}(s_1, s_2)|.$$
                Store
                    $$q(u_1, u_{d-1}; s_1, s_2) = \frac{q(u_2, u_{d-1}; u_1, s_2) + q(u_1, u_{d-2}; s_1, u_{d-1})}{|T_{\rho(u_1,u_{d-1})}(s_1, s_2)|}.$$
            end if
            Compute $g(s_1, s_2)$ from (3.10).
            $C(s_1, s_2 \mid G_{\mathbf{i}}) = (n-2)!\, q(u_1, u_{d-1}; s_1, s_2)/g(s_1, s_2)$.
            Update $(s_1^*, s_2^*)$ and $C^*$ if $C(s_1, s_2 \mid G_{\mathbf{i}}) > C^*$.
        end for
    end for
end for
```

3.2.4 Geometric Trees with Two Sources

In this section, we study the special case of geometric trees, propose an approximate estimator for geometric trees, and provide theoretical analysis for its performance. First, we give the definition of geometric trees and prove some of its key properties. Then, we derive a lower bound for $C(S \mid G_{\mathbf{i}})$, and propose an estimator based on this lower bound. We show that our proposed estimator is asymptotically correct, i.e., it identifies the actual infection sources with probability (conditioned on the infection sources) going to one as the infection graph $G_{\mathbf{i}}$ becomes large. For mathematical convenience, instead of letting the number of infected nodes n grow large, we let the time t from the start of the infection process to our observation time become large.

The geometric tree network is defined in [23] w.r.t. a single infection source. In the following, we extend this definition to the case where there are two sources. Let $S^* = \{s_1, s_2\}$ be the infection sources, and let $T'_u(s_1, s_2)$ be defined in the graph G in the same way as $T_u(s_1, s_2)$ is defined for $G_{\mathbf{i}}$. Let $\mathcal{N}_G(\rho(s_1, s_2))$ be the set of nodes that have a neighboring node in $\rho(s_1, s_2)$. For each node u, let $n(u, r)$ be the number of nodes in $T'_u(s_1, s_2)$ that are at a distance r from u. We say that G is a geometric tree if for all $u \in \mathcal{N}(\rho(s_1, s_2))$, we have

$$br^{\alpha} \leq n(u, r) \leq cr^{\alpha}, \tag{3.11}$$

where α, b, and c are fixed positive constants with $b \leq c$. The condition (3.11) implies that all trees defined w.r.t. the infection sources are growing polynomially fast at about the same rate. As we have assumed that the infection rates are homogeneous for every node, the resulting infection graph $G_{\mathbf{i}}$ will also be approximately regular with high probability. We have the following properties for a geometric tree, whose proofs are in Section 3.5..

Lemma 3.3. Suppose that G is a geometric tree with two infection sources $S^* = \{s_1, s_2\}$. Let α, b and c be fixed positive constants satisfying (3.11) for the geometric tree G. Let t be the time from the start of the infection process to our observation time. For any $\epsilon \in (0, 1)$, let $\mathcal{E}_t$ be the event that all nodes within distance $t(1-t^{-1/2+\epsilon})$ of either source nodes are infected, and no nodes greater than distance $t(1+t^{-1/2+\epsilon})$ of either source nodes are infected. Then, there exists t_0 such that for all $t \geq t_0$, $\mathbb{P}(\mathcal{E}_t) \geq 1 - \epsilon$. Furthermore, conditioned on $\mathcal{E}_t$, we have for all $u \in \mathcal{N}_G(s_1) \cup \mathcal{N}_G(s_2)$ or $u = \rho(s_1, s_2) \backslash S^*$,

$$N_{\min}(t) \leq |T_u(s_1, s_2)| \leq N_{\max}(t), \tag{3.12}$$

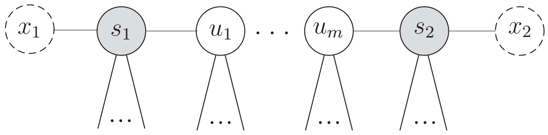


Figure 3-4: Addition of virtual nodes x_1 and x_2.

where

$$N_{\min}(t) = \frac{b}{1+\alpha}\left(t - t^{\frac{1}{2}+\epsilon} - d(s_1, s_2) - 2\right)^{\alpha+1}, \tag{3.13}$$

and

$$N_{\max}(t) = \frac{c}{1+\alpha}\left(t + t^{\frac{1}{2}+\epsilon}\right)^{\alpha+1}. \tag{3.14}$$

In addition, for $t \geq t_0$, we have

$$\frac{N_{\min}(t)}{N_{\max}(t)} \geq \frac{b}{c}(1-\epsilon).$$

The infection sequences count in (3.7) is not amendable to analysis. In the following, we seek an approximation to simplify our analysis. For $s_1, s_2 \in G_{\mathbf{i}}$, suppose that $\rho(s_1, s_2) = (s_1, u_1, \ldots, u_m, s_2)$, with $p = |\rho(s_1, s_2)| = m + 2$. Instead of computing $C(s_1, s_2 \mid G_{\mathbf{i}})$, we consider a new infection graph $G'_{\mathbf{i}}$ with two "virtual" nodes x_i, $i = 1, 2$ added, where x_i is attached to s_i (see Figure 3-4). We now consider the infection sequence count $C(x_1, x_2 \mid G'_{\mathbf{i}}) \geq C(s_1, s_2 \mid G_{\mathbf{i}})$. Since the trees rooted at x_i are single node trees, we have

$$\begin{aligned} C(x_1, x_2 \mid G'_{\mathbf{i}}) &= C(s_1, x_2 \mid G'_{\mathbf{i}}) + C(x_1, s_2 \mid G'_{\mathbf{i}}) \\ &\leq 2(n-1)C(s_1, s_2 \mid G_{\mathbf{i}}), \end{aligned}$$

where the last inequality follows because if s_1 and x_2 are sources, then s_2 can be inserted in any of at most $n-1$ positions in an infection sequence from $\Omega(G_{\mathbf{i}}, \{s_1, s_2\})$, so that $C(s_1, x_2 \mid G'_{\mathbf{i}}) \leq (n-1)C(s_1, s_2 \mid G_{\mathbf{i}})$. A similar argument holds for $C(x_1, s_2 \mid G'_{\mathbf{i}}) \leq (n-1)C(s_1, s_2 \mid G_{\mathbf{i}})$.

Let $\xi^* = (\xi_1^*, \ldots, \xi_p^*)$ be a permutation of the nodes in $\rho(s_1, s_2)$ such that $|T_{\xi_i^*}(s_1, s_2)| \geq |T_{\xi_j^*}(s_1, s_2)|$ for all $1 \leq i \leq j \leq p$, i.e., the nodes in ξ^* are arranged in descending order of the size of the sub-trees rooted at them. Let $I_i^*(s_1, s_2) = I_i(\xi^*; s_1, s_2)$ (cf. the definition in (3.6)) be the total number of nodes in the i biggest sub-trees in $\{T_u(s_1, s_2) : u \in \rho(s_1, s_2)\}$. From Lemma 3.2, we have

$$C(x_1, x_2 \mid G'_{\mathbf{i}}) \geq n! \cdot 2^{p-1} \prod_{i=1}^{p} I_i^*(s_1, s_2)^{-1} \prod_{u \in G_{\mathbf{i}} \setminus \rho(s_1, s_2)} |T_u(s_1, s_2)|^{-1}, \tag{3.15}$$

where the inequality holds because $|\Gamma(s_1, s_2)| = 2^{p-1}$, and each term in the sum on the R.H.S. of (3.9) is lower bounded by $\prod_{i=1}^{p} I_i^*(s_1, s_2)^{-1}$. We use the lower bound in (3.15) as a proxy for $C(s_1, s_2 \mid G_{\mathbf{i}})$. However, we have used a very loose lower bound in (3.15), so we propose the estimator

$$\tilde{S} = \arg \max_{s_1, s_2 \in G_{\mathbf{i}}} \tilde{C}(s_1, s_2 \mid G_{\mathbf{i}}), \tag{3.16}$$

where

$$\tilde{C}(s_1, s_2 \mid G_{\mathbf{i}}) = n! \cdot Q(s_1, s_2) \prod_{u \in G_{\mathbf{i}} \setminus \rho(s_1, s_2)} |T_u(s_1, s_2)|^{-1}, \tag{3.17}$$

$$Q(s_1, s_2) = [2(1 + \delta)]^{p-1} \prod_{i=1}^{p} I_i^*(s_1, s_2)^{-1},$$

and δ is a fixed positive constant, to be chosen based on prior knowledge about the graph G. Algorithm 3.2 can be modified to find the maximizer for $\tilde{C}(\cdot, \cdot \mid G_{\mathbf{i}})$. We call this the geometric tree TSE algorithm. The following result provides a way to choose δ, and shows that our proposed estimator $\tilde{S}$ is asymptotically correct in a geometric tree. A proof is provided in Section 3.5..

Theorem 3.2. Suppose that G is a geometric tree with two infection sources $S^* = \{s_1^*, s_2^*\}$. Let $d_{\min}$ and $d_{\max}$ be constants such that $\deg_G(s_i) \in [d_{\min}, d_{\max}]$ for $i = 1, 2$. Let b and c be fixed positive constants satisfying (3.11) for the geometric tree G.

Suppose that

$$d_{\min} \geq \frac{3}{2} + \frac{c}{b}\sqrt{2d_{\max}}. \tag{3.18}$$

Then, for any δ in the non-empty interval

$$\left(\frac{cd_{\max}}{b(d_{\min}-1)} - 1, \frac{b(d_{\min}-2)}{2c} - 1\right), \tag{3.19}$$

we have

$$\lim_{t\to\infty} \mathbb{P}(\tilde{S} = S^* \mid S^*) = 1.$$

Theorem 3.2 implies that if we know the constants governing the regularity condition (3.11) for G, we can choose a δ so that our estimator $\tilde{S}$ gives the true infection sources with high probability if the infection graph $G_{\mathbf{i}}$ is large. The class of geometric trees as defined by (3.11) can be used to model various scenarios in practice, e.g., a tree spanning a wireless sensor network with nodes randomly scattered. However, the assumption (3.11) may also be overly strong for other applications. In Section 3.4, we perform numerical studies to gain insights into the performance of our proposed estimator for different classes of tree networks.

3.2.5 Unknown Number of Infection Sources

In most practical applications, the number of infection sources is not known a priori. However, typically we may be able to guess the maximum number of infection sources $k_{\max}$, or we can choose a reasonable value of $k_{\max}$ depending on the size of the infection graph $G_{\mathbf{i}}$. In this section, we present a *heuristic* algorithm that allows us to estimate the infection sources with a given $k_{\max}$.

We first consider the instructive case where $k_{\max} = 2$ and G is a geometric tree. In this case, the number of infection sources can be either one or two. Suppose we run the geometric tree TSE algorithm on $G_{\mathbf{i}}$. We have the following result, whose proof is in Section 3.5.

Theorem 3.3. Suppose that there is a single infection source s and G is a geometric tree with (3.11) holding for all nodes u that are neighbors of s. Suppose that s has degree $\deg_G(s) \in [d_{\min}, d_{\max}]$, where $d_{\min}$ and $d_{\max}$ are positive constants satisfying (3.18). Then, for any δ in the interval (3.19), the geometric tree TSE algorithm estimates as sources s and one of its neighbors with probability (conditioned on s being the infection source) going to 1 as $t \to \infty$.

Theorem 3.3 implies that when there exists only one source, the geometric tree TSE algorithm finds two neighboring nodes, one of which is the true source. From Theorem 3.2 and Assumption 3.2, if there are two sources, our algorithm identifies the two source nodes, which are at least two hops from each other, with high probability. Therefore, by checking the distance between the two nodes identified by the geometric tree TSE algorithm, we can estimate the number of source nodes in the infection graph. This observation together with Theorem 3.1 suggest the following heuristic.

(i) Randomly choose $k_{\max}$ nodes satisfying Assumption 3.2 as the infection sources and find a Voronoi partition for $G_{\mathbf{i}}$. Use the SSE algorithm to find a source node for each infection region. Repeat these steps until for every region, the distance between estimated source nodes between iterations is below a fixed threshold or a maximum number of iterations is reached. We call this the Infection Partition (IP) Algorithm (see Algorithm 3.3).

(ii) For any two regions in the partition obtained from step (i) that are connected by an edge in $G_{\mathbf{i}}$, run the TSE algorithm in the combined region to find two source estimates. If the two estimates have distance less than τ, we decrement the number of source nodes, and repeat step (i).

(iii) The above two steps are repeated until no two pairs of regions in the Voronoi partition can be combined. The formal algorithm is given as the Multiple Sources Estimation and Partitioning (MSEP) algorithm in Algorithm 3.4.

To compute the complexity of the MSEP algorithm, we note that since the IP algorithm is based on the SSE algorithm, it has complexity $O(n)$. For each value of

Algorithm 3.3 Infection Partitioning (IP)

```
Inputs: An infection source set $S^{(0)} = \{s_i^{(0)} : i = 1, \ldots, m\}$ in $G_{\mathbf{i}}$.
Iterations:
for $l = 1$ to MaxIter do
    Run the Voronoi partitioning algorithm with centers in $S^{(l-1)}$ to obtain the infection partition $\mathcal{A}^{(l)} = \cup_{i=1}^{m} A_i^{(l)}$.
    for $i = 1$ to $m$ do
        Run SSE algorithm in $A_i^{(l)}$ to obtain
            $$s_i^{(l)} = \arg \max_{s \in A_i^{(l)}} C(s \mid A_i^{(l)}).$$
    end for
    $S^{(l)} := \{s_i^{(l)} : i = 1, \ldots, m\}$
    if $\max_{1 \le i \le m} d(s_i^{(l)}, s_i^{(l-1)}) \le \eta$ for some fixed small positive $\eta$ then
        break
    end if
end for
return $(S^{(l)}, \mathcal{A}^{(l)})$
```

Algorithm 3.4 Multiple Sources Estimation and Partitioning (MSEP)

```
Inputs: $G_{\mathbf{i}}$ and $k_{\max}$.
Initialization:
$k := k_{\max}$ and choose $S := \{s_1, \ldots, s_k\}$ randomly in $G_{\mathbf{i}}$.
Iterations:
while $k > 1$ do
    $(S, \mathcal{A})$ = Algorithm IP$(S)$
    $S' := S$
    for all regions $A_i$ and $A_j$ in the partition $\mathcal{A}$ such that there exists an edge $(u, v)$ in $G_{\mathbf{i}}$ with $u \in A_i$ and $v \in A_j$ do
        Set $(u, v)$ = Algorithm TSE$(A_i \cup A_j)$.
        if $d(u, v) < \tau$ then
            Merge $A_i$ and $A_j$, set $s_i = u$ and discard $s_j$
            $k := k - 1$
            break
        end if
    end for
    if $S = S'$ then
        break
    end if
end while
return $(S, \mathcal{A})$
```

$k = 1, \ldots, k_{\max}$ in the MSEP algorithm, there are $O(k^2)$ pairs of neighboring regions in the infection partition. For each pair of region, the TSE algorithm makes $O(n^2)$ computations. Summing over all $k = 1, \ldots, k_{\max}$, the time complexity of the MSEP algorithm can be shown to be $O(k_{\max}^3 n^2)$. On the other hand, to compute $C(S \mid G_{\mathbf{i}})$ for $|S^*|= k_{\max}$ would require $O(n^{k_{\max}})$ computations.

3.3 Identifying Infection Sources and Regions for General Graphs

In this section, we generalize the MSEP algorithm to identify multiple infection sources in general graphs G. In [23], the SSE algorithm is extended to general graphs when it is known that there is only a single infection source in the network using a heuristic. The algorithm first chooses a node s of $G_{\mathbf{i}}$ as the root node, and generates a spanning tree $T_{\text{bfs}}(s, G_{\mathbf{i}})$ of $G_{\mathbf{i}}$ rooted at s using the breadth-first-search (BFS) procedure. The SSE algorithm is then applied on this spanning tree to compute $C(s \mid T_{\text{bfs}}(s, G_{\mathbf{i}}))$. In addition, the infection sequences count is weighted by the likelihood of the BFS tree. This is repeated using every node in $G_{\mathbf{i}}$ as the root node, and the node $\hat{s}$ with the maximum weighted infection sequences count is chosen as the source estimator, i.e.,

$$\hat{s} = \arg\max_{v \in G_{\mathbf{i}}} P(\sigma_v \mid v) C(s \mid T_{\text{bfs}}(v, G_{\mathbf{i}})),$$

where σ_v is the sequence of nodes that corresponds to an infection spreading from v along the BFS tree. It can be shown that this algorithm has complexity $O(n^2)$. For further details, the reader is referred to [23]. We call this algorithm the SSE-BFS algorithm in this thesis.

We adapt the MSEP algorithm for general graphs using the same BFS heuristic. Specifically, we replace the SSE algorithm in line 6 of the IP algorihm with the SSE-BFS algorithm. In addition, in line 9, we run the TSE algorithm on $T_{\text{bfs}}(s_i, A_i) \cup T_{\text{bfs}}(s_j, A_j)$, where the two BFS trees are connected by randomly selecting an edge

(u, v) in $G_{\mathbf{i}}$ with $u \in T_{\text{bfs}}(s_i, A_i)$ and $v \in T_{\text{bfs}}(s_j, A_j)$. We call this modified algorithm the MSEP-BFS algorithm. Since the worst case complexity for the SSE-BFS algorithm is $O(n^2)$, the complexity of the MSEP-BFS algorithm can be shown to be $O(k_{\max}^3 n^2)$, which is the same complexity as the MSEP algorithm. To verify the effectiveness of the MSEP-BFS algorithm, we conduct simulations on both synthetic and real world networks in Section 3.4.

3.4 Simulation Results and Tests

In this section, we present results from simulations and tests on real data to verify our proposed algorithms. We first consider geometric tree networks and regular tree networks with various numbers of infection sources, and then we present results on small-world networks, scale-free networks and a real world power grid network. We also apply our algorithms to the contact tracing data obtained during the SARS outbreak in Singapore in 2003 [9] and the Arizona-Southern California cascading power outages in 2011 [19].

3.4.1 Synthetic Networks

We first perform simulations on geometric trees, regular trees, and small-world networks. The number of infection sources $|S^*|$ are chosen to be 1, 2, or 3, and we set $k_{\max} = 3$. For each type of network and each number of infection sources, we perform 1000 simulation runs with 500 infected nodes. We randomly choose infection sources satisfying Assumption 3.2 and obtain the infection graph by simulating the infection spreading process using the SI model. Finally, the MSEP or MSEP-BFS algorithm for tree networks and small-world networks respectively, is applied to the infection graph to estimate the number and locations of the infection sources. The estimation results for the number of infection sources $|\hat{S}|$ in different scenarios are shown in Figure 3-5. It can be seen that our algorithm correctly finds the number of infection sources more than 93% of the time for geometric trees, and more than 71% of the time for regular trees. The accuracy of about 69.2% for small-world networks is worse than that for

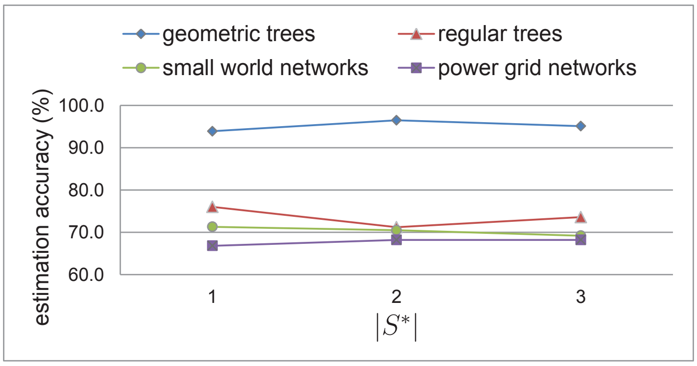


Figure 3-5: Estimating the number of infection source nodes.

the tree networks, as the infection tree for a small-world network has to be estimated using the BFS heuristics, thus additional errors are introduced into the procedure.

Table 3.2: Performance comparisons.

Simulation settings		Average	Average error distance Δ					Average minimum
network topology	$\lvert S^*\rvert$	diameter	MSEP/MSEP-BFS		nSSE			infection region
		of $G_\mathbf{i}$	$\eta = 0$	$\eta =$ diameter	$\eta = 0$	$\eta =$ diameter	known $\lvert S^*\rvert$	covering percentage (%)
geometric trees	2	63.7	0.61	1.72	9.65	30.16	12.85	97.06
	3	66.2	0.91	2.42	7.69	29.95	14.84	89.77
regular trees	2	40.5	0.84	6.07	4.50	17.70	6.13	73.82
	3	43.7	0.94	6.24	3.39	17.47	6.59	65.95
small-world networks	2	35.5	2.95	8.19	5.40	17.13	8.28	76.62
	3	40.9	2.58	8.18	4.99	18.56	10.37	60.69
power grid network	2	27.3	3.65	7.39	5.50	14.66	7.89	70.29
	3	30.8	2.85	8.47	4.71	14.75	8.89	59.95

When there are more than one infection sources, we compare the performance of the MSEP algorithm with a naive estimator based on the SSE algorithm. We call this the nSSE algorithm. Specifically, in the estimator for tree networks, we first compute $C(u \mid G_\mathbf{i})$ for all nodes $u \in G_\mathbf{i}$, and choose the $|S^*|$ nodes with the largest counts as the source nodes. In non-tree networks, we use the SSE-BFS algorithm. Since the nSSE algorithm can not estimate $|S^*|$, we consider two variants. In the first variant, we assume the nSSE algorithm has prior knowledge of $|S^*|$. In the second variant, we guess $|S^*|$ by choosing uniformly from $\{1, \ldots, k_{\max}\}$.

To quantify the performance of each algorithm, we first match the estimated source nodes $\hat{S} = \{\hat{s}_i : i = 1, \ldots, |\hat{S}|\}$ with the actual sources S^* so that the sum of the error distances between each estimated source and its match is minimized. Let this matching be denoted by the function π, which matches each actual source s_i to $\hat{s}_{\pi(i)}$. If we have incorrectly estimated the number of infection sources, i.e., $|\hat{S}|\neq |S^*|$, we add a penalty term to this sum. The average error distance is then given by

$$\Delta = \frac{1}{|S^*|}\left(\sum_{i=1}^{\min(|S^*|,|\hat{S}|)} d(\hat{s}_{\pi(i)}, s_i) + \eta\left||\hat{S}| - |S^*|\right|\right),$$

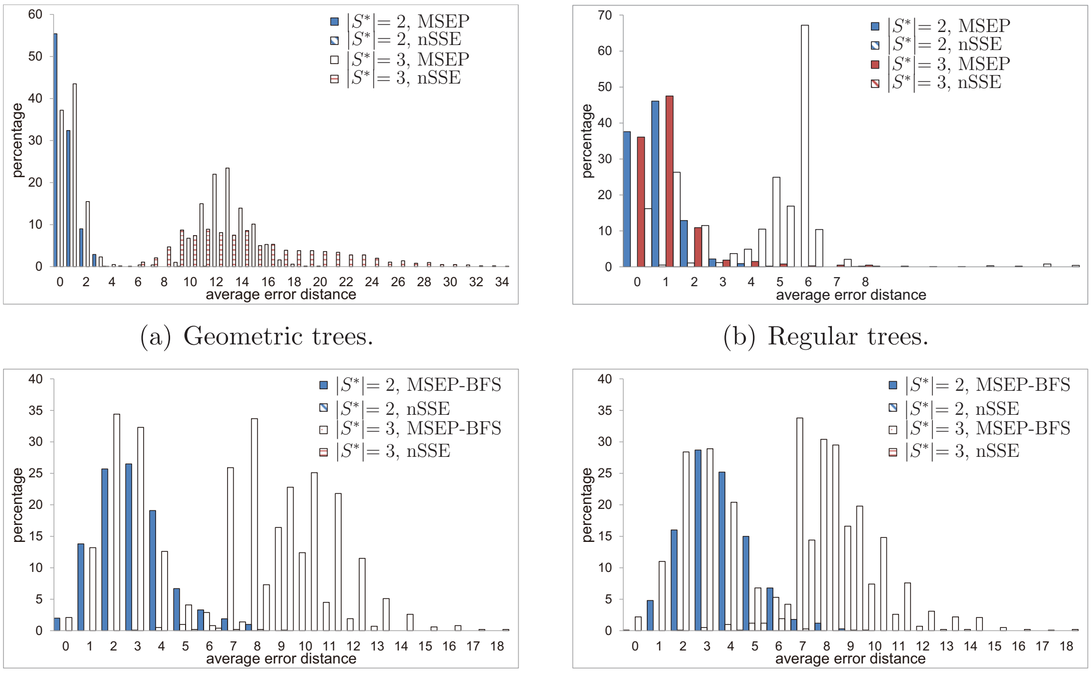
where η is a penalty weight for incorrectly estimating the number of infection sources. For different applications, we may assign different values to η depending on how important it is to estimate correctly the number of infection sources. In our simulations, we consider the cases where $\eta = 0$, and where η is the diameter of the infection graph. The average error distances for the different types of networks are provided in Table 3.2. Clearly, the MSEP/MSEP-BFS algorithm outperforms the nSSE algorithm, even when the nSSE algorithm has prior knowledge of the number of sources. When $|S^*|$ is known a priori, the performance of the nSSE algorithm deteriorates with increasing $|S^*|$. This is to be expected as the SSE algorithm assumes that the node with the largest infection sequence count is the only source, and this node tends to be close to the distance center [61] of the infection graph. The histogram of the average error distances when $\eta = 0$ are shown in Figure 3-6.

The MSEP/MSEP-BFS algorithm also estimates the infection region of each source. To evaluate its accuracy, we first perform the matching process described previously. Let the true infection region of s_i be $A_{n,i}$ and the estimated infection region of $\hat{s}_{\pi(i)}$ be $\hat{A}_{n,i}$, where we set $\hat{A}_{n,i} = \emptyset$, if we have underestimated the number of sources and s_i is unmatched. We define the correct infection region covering percentage for s_i as the ratio between $|\hat{A}_{n,i} \cap A_{n,i}|$ and $|A_{n,i}|$, and we compute the minimum (or worst case) infection region covering percentage as

$$\min_{i\in\{1,\cdots,|S^*|\}} \frac{|\hat{A}_{n,i} \cap A_{n,i}|}{|A_{n,i}|}.$$

This is then averaged over all simulation runs. We find that the average minimum infection region covering percentage is more than 59% for all networks, as shown in Table 3.2.

We now consider the scale-free networks which are typically dense and have small



(a) Geometric trees.

(b) Regular trees.

(c) Small-world networks.

(d) US power grid network.

Figure 3-6: Histogram of the average error distances for various networks. We assume $\eta = 0$ and that the nSSE algorithm has prior knowledge of the number of infection sources.

diameters [62]. When multiple infection sources are randomly chosen in such networks, a pair of chosen source nodes are likely to be neighboring nodes. This violates the assumption for the proposed MSEP-BFS algorithm. We consider the case where the number of infection sources is known to be 2, and test the proposed TSE algorithm on scale-free networks. For each simulation run, we randomly choose two infection sources and simulate the infection spreading process until there are 500 infected nodes. We perform 1000 simulation runs and show the simulation results in Figure 3-7. We see that the TSE algorithm outperforms the nSSE algorithm for the scale-free networks.

3.4.2 Real World Networks

We verify the performance of the MSEP-BFS algorithm on the western states power grid network of the United States [63]. We simulate the infection spreading process on the power grid network, which contains 4941 nodes. For each simulation run, 1, 2 or 3

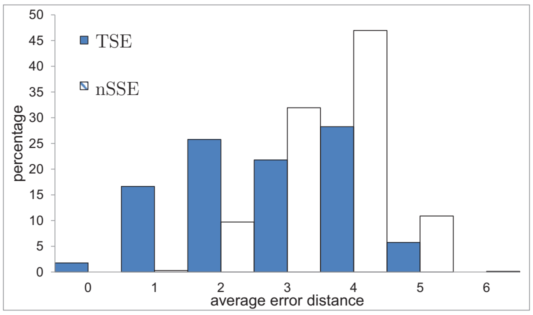


Figure 3-7: Histogram of the average error distances for scale-free networks when the number of infection sources is known to be 2.

infection sources are randomly chosen from the power grid network under Assumption 3.2, and the spreading process is simulated so that a total of 500 nodes are infected. For each value of $|S^*|$, 1000 simulation runs are performed. The simulation results are shown in Figures 3-5 and 3-6(d), and Table 3.2. We see that the MSEP-BFS algorithm outperforms the nSSE algorithm in every scenario. The average infection region covering percentage is above 59%.

3.4.3 Tests on Real Data

In order to get some insights in the performance of the MSEP-BFS algorithm in real infection spreads, we conduct two tests on real infection spreads data. We first apply the MSEP-BFS algorithm to to a network of nodes that represent the individuals who were infected with the SARS virus during an epidemic in Singapore in the year 2003. The data is collected using contact tracing of patients [9], where an edge between two nodes indicate that there is some form of interaction or relationship between the individuals (e.g., they are family members, classmates, colleagues, or commuters who shared the same public transport system). A part of the SARS infection network corresponding to a cluster of 193 patients is shown in Figure 3-8. We test the MSEP-BFS algorithm on the network in Figure 3-8, assuming that there are at most $k_{\max} = 3$ infection sources. It turns out that the MSEP-BFS algorithm correctly estimates the number of infection sources to be one, and correctly identifies the real infection source.

We next consider the Arizona-Southern California cascading power outages in

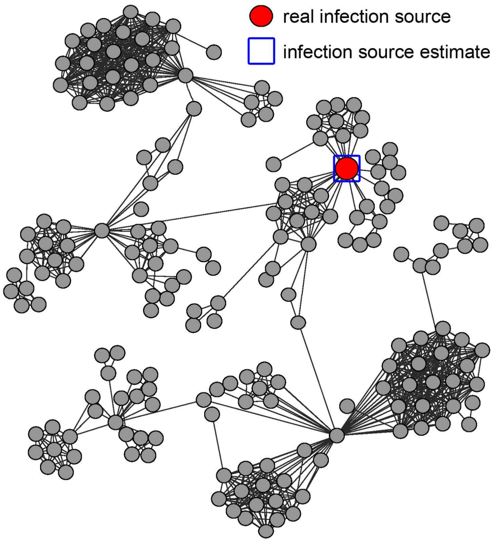


Figure 3-8: Illustration of a cluster of the SARS infection network with a single source.

2011 [19]. The affected power network is represented by a graph where a node represents a key facility (substation or generating plant) affected by an outage, and an edge between two nodes indicate that there is a transmission line between these two facilities. The cascading outage starts with the loss of a single transmission line. However, as indicated in [19], this transmission line alone would not cause a cascading outage. After the loss of this transmission line, instantaneous power flow redistributions led to large voltage deviations, resulting in the nuclear units at San Onofre Nuclear Generating Station being taken off the power grid. The failures of these two key facilities together serve as the main causes of the subsequent cascading outages, so these two facilities are considered as the two infection sources. The main affected power network containing 48 facilities is shown in Figure 3-9. We test the MSEP-BFS algorithm on the network in Figure 3-9, and assume that there are at most $k_{\max} = 3$ infection sources. We can see that the MSEP-BFS algorithm correctly estimates the number of infection sources to be two. We also found one of the sources correctly, and one estimate 1 hop away from the real source.

3.5 Proofs

In this section, we provides proofs of some of the results in this chapter.

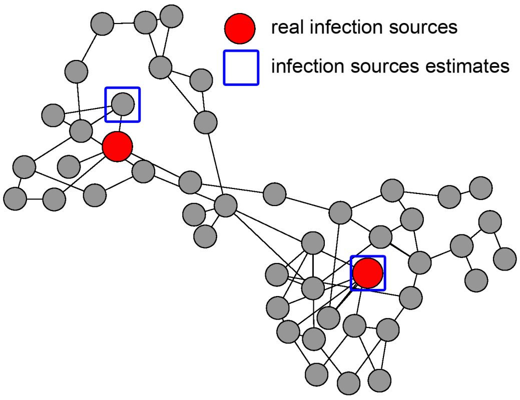


Figure 3-9: Illustration of the main affected power network with two infection sources.

Proof of Theorem 3.1

Let nodes that are infected by source s_i be colored with color i, with $i = 1, \ldots, k$. Then a partition $\mathcal{A}$ corresponds to a coloring of the graph R, and to quantify the probability of a partition, it is sufficient to consider only infection sequences in the graph R. We have

$$P(\mathcal{A} \mid S, G_{\mathbf{i}}) = \sum_{\sigma \in \Omega(R,S,\mathcal{A})} P(\sigma \mid S), \tag{3.20}$$

where

$$\Omega(R, S, \mathcal{A}) = \{\sigma \in \Omega(R, S) : \sigma \cap A_{n,i} \text{ is an infection sequence, for all } i = 1, \ldots, k.\},$$

and $\sigma \cap A_{n,i}$ is the subsequence of σ containing only nodes that are in $A_{n,i}$.

Let $h = |R| - k$, and consider an infection sequence $\sigma = (\sigma_1, \ldots, \sigma_h) \in \Omega(R, S, \mathcal{A})$. Let the set of edges connecting susceptible nodes to infected nodes be called the susceptible edge set. We have assumed that the infection times of susceptible nodes are independent and identically exponentially distributed. Therefore, given the infection sequence $\sigma_1, \ldots, \sigma_{l-1}$, the next edge along which the infection is spread is chosen uniformly at random from the susceptible edge set at time index $l - 1$. Since R is a tree where all nodes except those in S have degree 2, after infection of a new node, the susceptible edge set size remains the same except in the case where the infected node is the last node to be infected amongst those on a path connecting two infection sources. In that case, the susceptible edge set size reduces by 2. Let J_σ be the set of indices of the last infected nodes on every path connecting infection sources. Letting $n_l = 1$ if $l \notin J_\sigma$ and 2 otherwise, we then have

$$\begin{aligned} P(\sigma \mid S) &= \prod_{l=1}^{h} n_l p_l(\sigma \mid R, S) \\ &= 2^p \prod_{l=1}^{h} p_l(\sigma \mid R, S) \end{aligned} \tag{3.21}$$

where p is the number of paths connecting infection sources, and

$$p_l(\sigma \mid R, S) = \left(\sum_{s \in S} \deg_R(s) - 2 \sum_{j \in J_\sigma} \mathbf{1}_{\{j<l\}} \right)^{-1}. \tag{3.22}$$

Choose two sources s_i and s_j and let m be the number of nodes in the path $\rho(s_i, s_j)$ connecting s_i and s_j, excluding the source nodes. Suppose that $r > \lceil m/2 \rceil$ nodes in this path have color i. Construct a new coloring $\mathcal{A}'_n$ so that $\lceil m/2 \rceil$ nodes in $\rho(s_i, s_j)$ closest to s_i have color i and the rest have color j. The rest of the nodes in $\mathcal{A}'_n$ have the same colors as that in $\mathcal{A}$. Each infection sequence $\sigma \in \Omega(R, S, \mathcal{A})$ corresponds to an infection sequence $\sigma' \in \Omega(R, S, \mathcal{A}'_n)$, where the last $x = r - \lceil m/2 \rceil$ color-i nodes in σ become the last x color-j nodes in σ'. From (3.22), we have $p_l(\sigma \mid R, S) = p_l(\sigma' \mid R, S)$ for all l. Since $\binom{m}{\lceil m/2 \rceil} \geq \binom{m}{r}$, we have $|\Omega(R, S, \mathcal{A}'_n)| \geq |\Omega(R, S, \mathcal{A})|$, therefore (3.20) yields $P(\mathcal{A}'_n \mid S, G_\mathbf{i}) \geq P(\mathcal{A} \mid S, G_\mathbf{i})$.

The same argument can be repeated a finite number of times for all paths in R connecting infection sources. This shows that the estimator $\hat{\mathcal{A}}(S)$ is a Voronoi partition of $G_\mathbf{i}$, and the proof is complete.

Proof of Lemma 3.2

To simplify notations, we write $T_u(s_1, s_2)$ as T_u, with the implicit understanding that all trees are defined w.r.t. $\{s_1, s_2\}$. The number of infection sequences can be found by counting the number of ways to form such a sequence. The $n - 2$ slots in a sequence are occupied by nodes from $T_{s_i}\backslash\{s_i\}$, $i = 1, 2$, and $T_{\rho(u_1,u_m)}$. Therefore, we have

$$\begin{aligned}
C(s_1, s_2 \mid G_{\mathbf{i}}) &= (n-2)! \prod_{i=1}^{2} \frac{C(s_i \mid T_{s_i})}{(|T_{s_i}|-1)!} \cdot \frac{R(u_1, u_m)}{|T_{\rho(u_1,u_m)}|!} \\
&= \frac{(n-2)!}{|T_{\rho(u_1,u_m)}|!} \cdot R(u_1, u_m) \cdot \prod_{\substack{v \in T_{s_i}, i=1,2 \\ v \neq s_1, s_2}} |T_v|^{-1},
\end{aligned}$$

where $R(u_i, u_j)$ for $i \leq j$ is the number of ways of permuting the nodes in $T_{\rho(u_i,u_j)}$ such that the infection sequence property is maintained, and the last equality follows from Lemma 3.1. To simplify the notations, for $1 \leq i \leq j \leq m$, let

$$J(u_i, u_j) = \prod_{v \in T_{\rho(u_i,u_j)} \backslash \rho(u_i,u_j)} |T_v|^{-1}.$$

For example, from Lemma 3.1, we have $C(u_i \mid T_{u_i}) = (|T_{u_i}|-1)!\, J(u_i, u_i)$. In the following, we show that for $1 \leq i \leq j \leq m$,

$$R(u_i, u_j) = |T_{\rho(u_i,u_j)}|! \cdot q(u_i, u_j; s_1, s_2) \cdot J(u_i, u_j). \tag{3.23}$$

The proof proceeds by induction on $j - i$. If $j = i$, we have $R(u_i, u_i) = C(u_i \mid T_{u_i})$ and the claim follows from Lemma 3.1. Suppose that the claim (3.23) holds for all nodes u_k and u_p such that $p - k < j - i$. The number of permutations $R(u_i, u_i)$ can be computed by considering a sequence with $m = |T_{\rho(u_i,u_j)}|$ slots. The first slot can be filled with either u_i or u_j. Therefore, we have

$$\begin{aligned}
&R(u_i, u_j) \\
=&(m-1)! \left(\frac{C(u_i \mid T_{u_i})}{(|T_{u_i}|-1)!} \frac{R(u_{i+1}, u_j)}{|T_{\rho(u_{i+1},u_j)}|!} + \frac{C(u_j \mid T_{u_j})}{(|T_{u_j}|-1)!} \frac{R(u_i, u_{j-1})}{|T_{\rho(u_i,u_{j-1})}|!} \right) \\
=&(m-1)!\, (J(u_i, u_i) q(u_{i+1}, u_j; s_1, s_2) J(u_{i+1}, u_j) \\
&\quad + J(u_j, u_j) q(u_i, u_{j-1}; s_1, s_2) J(u_{i+1}, u_j)) \\
=&(m-1)!\, (q(u_{i+1}, u_j; s_1, s_2) + q(u_i, u_{j-1}; s_1, s_2))\, J(u_i, u_j),
\end{aligned}$$

where the penultimate equality follows from the inductive hypothesis and Lemma 3.1, and the last equality follows by noting that $J(u_i, u_i)J(u_{i+1}, u_j) = J(u_j, u_j)J(u_{i+1}, u_j) = J(u_i, u_j)$. The claim (3.23) now follows from (3.8). Finally, (3.9) follows by an inductive argument using (3.8), which we omit. The proof is now complete.

Proof of Lemma 3.3

The proof follows easily from Theorems 5 and 6 of [23]. Consider the infection spreading along a path in $G_{\mathbf{i}}$. Let $\Pi(t)$ be the counting process of the number of infected nodes in this path. The process $\Pi(t)$ consists of exponentially distributed arrivals with rate 1, and at most one arrival with rate 2 if the path is between the two infection sources. Let $\Pi_1(t)$ be a unit rate Poisson process corresponding to the rate 1 arrivals. Then $\Pi_1(t) \leq \Pi(t) \leq \Pi_1(t) + 1$. From Theorem 6 of [23], we have for any positive $\gamma < 0.2$,

$$\begin{aligned}
&\mathbb{P}(\Pi(t) \leq t(1-\gamma)) \leq \mathbb{P}(\Pi_1(t) \leq t(1-\gamma) - 1) \leq \exp\left(-\frac{1}{4}t(\gamma + \frac{1}{t})^2\right), \\
&\mathbb{P}(\Pi(t) \geq t(1+\gamma)) \leq \mathbb{P}(\Pi_1(t) \geq t(1+\gamma)) \leq \exp\left(-\frac{1}{4}t\gamma^2\right).
\end{aligned}$$

The rest of the proof is the same as that of Theorem 5 of [23], and the proof is complete.

Proof of Theorem 3.2

We first show that under (3.18), the interval (3.19) is non-empty. The condition (3.18) implies that

$$d_{\min} > \frac{3}{2} + \sqrt{2d_{\max}\frac{c^2}{b^2} - \frac{1}{4}},$$

which after some algebraic manipulations yields

$$\begin{aligned}
&b^2(d_{\min} - 1)(d_{\min} - 2) > 2c^2 d_{\max}, \\
&1 \leq \frac{cd_{\max}}{b(d_{\min} - 1)} < \frac{b(d_{\min} - 2)}{2c}.
\end{aligned}$$

Therefore (3.19) is a non-empty interval. Fix a δ in the interval. Then for all $\epsilon > 0$ sufficiently small, we have

$$\begin{aligned} \frac{b(d_{\min}-1)(1+\delta)}{cd_{\max}} &> \frac{1}{1-\epsilon}, \\ \frac{b(d_{\min}-2)}{2(1+\delta)c} &> \frac{1}{1-\epsilon}. \end{aligned}$$

Recall that t is the time from the start of the infection spreading to our observation of $G_{\mathbf{i}}$. From Lemma 3.3, for each ϵ, there exists t_0 such that if $t \geq t_0$, we have

$$\frac{(d_{\min}-1)(1+\delta)N_{\min}(t)}{d_{\max}N_{\max}(t)} > 1, \tag{3.24}$$

$$\frac{(d_{\min}-2)N_{\min}(t)}{2(1+\delta)N_{\max}(t)} > 1. \tag{3.25}$$

We will make use of the two inequalities (3.24) and (3.25) extensively in the following proof steps. Let $\mathcal{E}_t$ be the event defined in Lemma 3.3. Then from Lemma 3.3, we have for $t \geq t_0$,

$$\mathbb{P}(\tilde{S} = S^* \mid S^*) \geq \mathbb{P}(\tilde{S} = S^* \mid S^*, \mathcal{E}_t)\mathbb{P}(\mathcal{E}_t \mid S^*) \geq (1-\epsilon)\mathbb{P}(\tilde{S} = S^* \mid S^*, \mathcal{E}_t). \tag{3.26}$$

In the following, we show that $\mathbb{P}(\tilde{S} = S \mid S, \mathcal{E}_t) = 1$ for $t \geq t_0$. The proof then follows from (3.26) as ϵ can be chosen arbitrarily small.

To show that $\mathbb{P}(\tilde{S} = S \mid S, \mathcal{E}_t) = 1$ is equivalent to showing that with probability one, $\tilde{C}(S \mid G_{\mathbf{i}}) > \tilde{C}(u_m, v_l \mid G_{\mathbf{i}})$, for all node pairs $u_m, v_l \in G_{\mathbf{i}}$ such that at least one of them is not in S. Let u_0 and v_0 be the first nodes in $\rho(s_1, s_2)$ that are connected to u_m and v_l respectively. We divide the proof into two cases, cases, depending on whether u_0 and v_0 are distinct or not, as shown in Figures 3-10 and 3-11.

Suppose that $u_0 \neq v_0$. A typical network for this case is shown in Figure 3-10, where m, l, n, p, and k are non-negative integers, and at least one of u_m and v_l is not in S, i.e., either $m + l > 0$ or $n + p > 0$. We let $u_0 = s_1$ if $n = 0$, and $v_0 = s_2$ if $p = 0$.

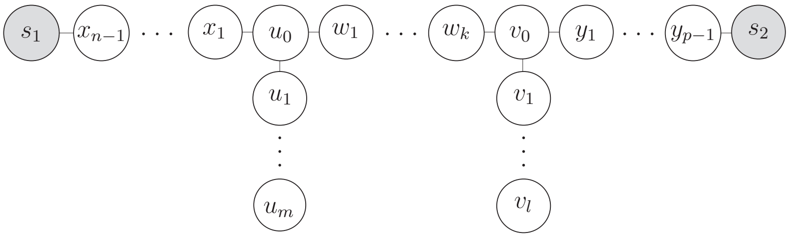


Figure 3-10: Illustration of the network structure when $u_0 \neq v_0$. Not all nodes are shown.

We will show that if either $m + l > 0$ or $n + p > 0$, we have for $t \geq t_0$,

$$\frac{\tilde{C}(s_1, s_2 \mid G_{\mathbf{i}})}{\tilde{C}(u_m, v_l \mid G_{\mathbf{i}})} = \frac{\tilde{C}(s_1, s_2 \mid G_{\mathbf{i}})}{\tilde{C}(u_0, v_0 \mid G_{\mathbf{i}})} \cdot \frac{\tilde{C}(u_0, v_0 \mid G_{\mathbf{i}})}{\tilde{C}(u_m, v_l \mid G_{\mathbf{i}})} > 1. \tag{3.27}$$

The proof follows by showing that $\tilde{C}(u_0, v_0 \mid G_{\mathbf{i}}) \geq \tilde{C}(u_m, v_l \mid G_{\mathbf{i}})$, where strict inequality holds if $m + l > 0$, and $\tilde{C}(s_1, s_2 \mid G_{\mathbf{i}}) \geq \tilde{C}(u_0, v_0 \mid G_{\mathbf{i}})$ with strict inequality holding if $n + p > 0$. From (3.17), we have [2]

$$\begin{aligned}
\frac{\tilde{C}(u_0, v_0 \mid G_{\mathbf{i}})}{\tilde{C}(u_m, v_l \mid G_{\mathbf{i}})} &= \frac{Q(u_0, v_0)}{Q(u_m, v_l)} \cdot \prod_{w \in \rho(u_m, u_1) \cup \rho(v_l, v_1)} |T_w(u_0, v_0)|^{-1} \\
&= [2(1+\delta)]^{-(m+l)} \cdot \frac{\prod_{i=1}^{m+l+k+2} I_i^*(u_m, v_l)}{\prod_{i=1}^{k+2} I_i^*(u_0, v_0)} \cdot \prod_{w \in \rho(u_m, u_1) \cup \rho(v_l, v_1)} |T_w(u_0, v_0)|^{-1} \\
&\geq [2(1+\delta)]^{-(m+l)} \cdot \prod_{i=1}^{m+l} I_i^*(u_m, v_l) \cdot \prod_{w \in \rho(u_m, u_1) \cup \rho(v_l, v_1)} |T_w(u_0, v_0)|^{-1} \\
&\geq \left[\frac{\max\{|T_{u_0}(u_m, v_l)|, |T_{v_0}(u_m, v_l)|\}}{2(1+\delta) \cdot \max\{|T_{u_1}(u_0, v_0)|, |T_{v_1}(u_0, v_0)|\}} \right]^{m+l} \\
&\geq \left[\frac{(d_{\max} - 2) N_{\min}(t) + 1}{2(1+\delta) \cdot N_{\max}(t)} \right]^{m+l} \\
&> 1,
\end{aligned}$$

if $m + l > 0$. The first inequality follows because $I_{m+l+i}^*(u_m, v_l) \geq I_i^*(u_0, v_0)$ for $i = 1, \ldots, k + 2$, and the last inequality follows from (3.25) when $t \geq t_0$.

[2] We define products over empty sets to be 1.

Let $\psi = \deg_G(s_1) + \deg_G(s_1)$. We have for $t \geq t_0$,

$$\begin{aligned}
\frac{\tilde{C}(s_1, s_2 \mid G_{\mathbf{i}})}{\tilde{C}(u_0, v_0 \mid G_{\mathbf{i}})} &= \frac{Q(s_1, s_2)}{Q(u_0, v_0)} \cdot \prod_{w \in \rho(s_1, x_1) \cup \rho(y_1, s_2)} |T_w(u_0, v_0)| \\
&= [2(1+\delta)]^{n+p} \cdot \frac{\prod_{i=1}^{k+2} I_i^*(u_0, v_0)}{\prod_{i=1}^{n+p+k+2} I_i^*(s_1, s_2)} \cdot \prod_{w \in \rho(s_1, x_1) \cup \rho(y_1, s_2)} |T_w(u_0, v_0)| \\
&\geq [2(1+\delta)]^{n+p} \cdot \prod_{i=k+3}^{n+p+k+2} I_i^*(s_1, s_2)^{-1} \cdot \prod_{w \in \rho(s_1, x_1) \cup \rho(y_1, s_2)} |T_w(u_0, v_0)| \\
&\geq \left[\frac{2(1+\delta) \cdot \min\left\{|T_{s_1}(u_0, v_0)|, |T_{s_2}(u_0, v_0)|\right\}}{\psi N_{\max}(t) + 2} \right]^{n+p} \\
&\geq \left[\frac{(1+\delta)(d_{\min} - 1) \cdot N_{\min}(t) + 1 + \delta}{d_{\max} N_{\max}(t) + 1} \right]^{n+p} \\
&> 1,
\end{aligned}$$

where the first inequality follows because $I_i^*(u_0, v_0) \geq I_i^*(s_1, s_2)$ for $i = 1, \ldots, k+2$, and the last inequality follows from (3.24) if $n + p > 0$. The bound (3.27) is now proved.

We next consider the case where $u_0 = v_0 = w_0$ in Figure 3-11, where k, m and l are non-negative integers. When $t \geq t_0$, we have the following bounds, which are straight forward to verify and whose proofs are omitted here.

(i) $I_i^*(u_m, v_l) \geq (\psi - 2)N_{\min}(t) + 2 \geq (d_{\min} - 2)N_{\min}(t)$ for $i = 1, \ldots, d(u_m, v_l) + 1$,

(ii) $I_i^*(s_1, s_2) \leq \psi N_{\max}(t) + 2 \leq 2d_{\max}N_{\max}(t) + 2$ for all $i = 1, \ldots, d(s_1, s_2) + 1$,

(iii) $|T_{w_i}(u_m, v_l)| \geq (\psi - 2)N_{\min}(t) + 2 \geq (d_{\min} - 2)N_{\min}(t)$ for all $i = 1, \ldots, k-1$,

(iv) $|T_w(u_m, v_l)| \geq (d_{\min} - 1)N_{\min}(t) + 1$ for all $w \in \rho(s_1, s_2)$,

(v) $|T_{w_i}(s_1, s_2)| \leq N_{\max}(t)$ for all $i = 1, \ldots, k-1$, and

(vi) $|T_w(s_1, s_2)| \leq N_{\max}(t)$ for all $w \in \rho(u_m, v_l)$.

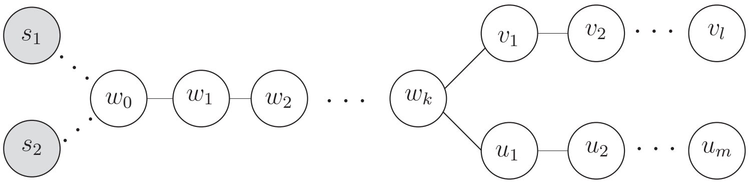


Figure 3-11: Illustration of the case where $u_0 = v_0 = w_0$.

The above bounds yield

$$
\begin{aligned}
&\frac{\tilde{C}(s_1, s_2 \mid G_{\mathbf{i}})}{\tilde{C}(u_m, v_l \mid G_{\mathbf{i}})}\\
=&\frac{Q(s_1, s_2)}{Q(u_m, v_l)} \frac{\prod_{w\in G_{\mathbf{i}}\backslash\rho(u_m,v_l)}|T_w(u_m, v_l)|}{\prod_{w\in G_{\mathbf{i}}\backslash\rho(s_1,s_2)}|T_w(s_1, s_2)|}\\
=&(2(1+\delta))^{d(s_1,s_2)-d(u_m,v_l)} \frac{\prod_{i=1}^{d(u_m,v_l)+1} I_i^*(u_m, v_l)}{\prod_{i=1}^{d(s_1,s_2)+1} I_i^*(s_1, s_2)} \frac{\prod_{i=1}^{k-1}|T_{w_i}(u_m, v_l)|\prod_{w\in\rho(s_1,s_2)}|T_w(u_m, v_l)|}{\prod_{i=1}^{k-1}|T_{w_i}(s_1, s_2)|\prod_{w\in\rho(u_m,v_l)}|T_w(s_1, s_2)|}\\
=&\prod_{i=1}^{k-1} \frac{|T_{w_i}(u_m, v_l)|}{|T_{w_i}(s_1, s_2)|} \cdot (2(1+\delta))^{-d(u_m,v_l)-1} \frac{\prod_{i=1}^{d(u_m,v_l)+1} I_i^*(u_m, v_l)}{\prod_{w\in\rho(u_m,v_l)}|T_w(s_1, s_2)|}\\
&\cdot (2(1+\delta))^{d(s_1,s_2)+1} \frac{\prod_{w\in\rho(s_1,s_2)}|T_w(u_m, v_l)|}{\prod_{i=1}^{d(s_1,s_2)+1} I_i^*(s_1, s_2)}\\
\geq&\left[\frac{(d_{\min}-2)N_{\min}(t)}{N_{\max}(t)}\right]^{k-1} \left[\frac{(d_{\min}-2)N_{\min}(t)}{2(1+\delta)N_{\max}(t)}\right]^{d(u_m,v_l)+1}\\
&\cdot \left[\frac{(1+\delta)((d_{\min}-1)N_{\min}(t)+1)}{d_{\max}N_{\max}(t)+1}\right]^{d(s_1,s_2)+1}\\
>&1,
\end{aligned}
$$

where the last inequality follows from (3.24) and (3.25). The theorem is now proved.

Proof of Theorem 3.3

Let t be the elapsed time from the start of an infection spreading from a single s to the time we observe $G_{\mathbf{i}}$. We wish to show that Algorithm TSE estimates as sources s and one of its neighbors with probability (conditioned on s being the infection source) converging to 1 as $t \to \infty$. This is equivalent to showing that for t sufficiently large, and for each pair of nodes $u_m, v_l \in G_{\mathbf{i}}$ where either $d(u_m, s) > 1$ or $d(v_l, s) > 1$, there exists a neighbor r of s such that $\tilde{C}(s, r \mid G_{\mathbf{i}}) > \tilde{C}(u_m, v_l \mid G_{\mathbf{i}})$.

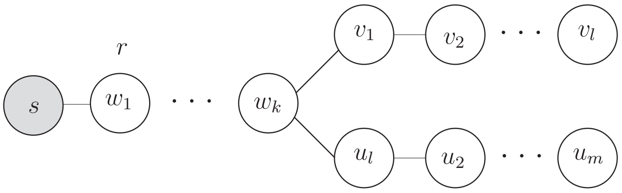


Figure 3-12: A typical network for a single source tree.

A typical network is shown in Figure 3-12, where k, m and l are non-negative integers. If m, l and k are positive, we let r be the neighbor of s that lies on the path connecting s to u_m (i.e., the node w_1 in Figure 3-12). If m and l are positive and $k = 0$, then r is chosen to be either u_1 or v_1. If $m = 0$, we must have $k > 0$ so that $w_k = u_m$ and $r = w_1$. A similar remark applies for the case $l = 0$. Note that $m + l > 0$. For t sufficiently large, we have

$$\begin{aligned}
\frac{\tilde{C}(s, r \mid G_{\mathbf{i}})}{\tilde{C}(u_m, v_l \mid G_{\mathbf{i}})} &= \frac{Q(s,r)}{Q(u_m,v_l)} \cdot \frac{\prod\limits_{w \in G_{\mathbf{i}} \setminus \rho(u_m,v_l)} |T_w(u_m,v_l)|}{\prod\limits_{w \in G_{\mathbf{i}} \setminus \{s,r\}} |T_w(s,r)|} \\
&= [2(1+\delta)]^{1-(m+l)} \cdot \frac{\prod_{i=1}^{m+l+1} I_i^*(u_m,v_l)}{\prod_{i=1}^{2} I_i^*(s,r)} \cdot \frac{\prod_{w \in \rho(s,w_{k-1})} |T_w(u_m,v_l)|}{\prod_{i=2}^{k-1} |T_{w_i}(s,r)| \cdot \prod\limits_{w \in \rho(u_m,v_l)} |T_w(s,r)|} \\
&= [2(1+\delta)]^{1-(m+l)} \cdot \prod_{i=1}^{m+l} I_i^*(u_m,v_l) \cdot \frac{\prod_{i=1}^{k-1} |T_{w_i}(u_m,v_l)|}{\prod_{i=2}^{k-1} |T_{w_i}(s,r)| \cdot \prod\limits_{w \in \rho(u_m,v_l)} |T_w(s,r)|} \\
&\geq [2(1+\delta)]^{1-(m+l)} \cdot |T_{w_k}(u_m,v_l)|^{m+l} \cdot \frac{|T_s(u_m,v_l)|^{k-1}}{N_{\max}(t)^{k-2} \cdot N_{\max}(t)^{m+l+1}} \\
&\geq [2(1+\delta)]^{k} \cdot \left[\frac{(d_{\min}-1)N_{\min}(t)}{2(1+\delta) \cdot N_{\max}(t)} \right]^{m+l+k-1} \\
&> 1,
\end{aligned}$$

where the last inequality follows from (3.25) and Lemma 3.3. The proof of the theorem is now complete.

Chapter 4

Single Infection Source Identification in the SI Model with Limited Observations under the MLIP Criterion

In the previous chapter, we assumed that we can observe the set of all infected nodes. However, as alluded to previously in Section 1.3.2, in some applications, it is impractical to obtain a full observation of the states of all nodes in a network. In this chapter, we consider the problem of identifying a single infection source in the SI model (cf. Section 2.1) under the MLIP criterion (cf. Section 2.3) based on the observations of a limited set of nodes in the network.

4.1 Problem Formulation

In this section, we describe our system model and assumptions, and then we summarize various notations and definitions that we use in the rest of this dissertation.

We assume the infection spreads according to the discrete time SI model as discussed in Section 2.1. We assume that a susceptible node becomes infected with probability $p_{\mathbf{s}}$ at the beginning of the next time slot, where $p_{\mathbf{s}} \in [0, 1]$. We assume that we know the complete network topology of the underlying graph G, however, we may not be able to observe some infected nodes. The Google+ example described in Section 1.3.2 is one example. Similarly, individuals who are carriers of a disease may appear to be asymptomatic. An infected node that exhibits its infected status is said to be *explicit*, and we let $\mathbf{X}(u, t) = \mathbf{e}$ when u becomes infected. We let $\mathbf{X}(u, t) = \mathbf{i}$ if u is infected but is non-explicit. A node that is observed to be uninfected can then either be actually uninfected or non-explicit. We call these nodes *non-observable.*

We let q_u be the probability that the node u is explicit, and assume that nodes are explicit or not independently of each other. If $q_u = 1$ for all nodes $u \in V$, our model reduces essentially to that in [23], whereas if q_u is close to zero for all nodes $u \in V$, the problem becomes intractable as most nodes appear to be uninfected, making the performance of any estimator of the source node poor in practice. Intuitively, if $q_u \geq p_{\mathbf{s}}$ for all nodes u, then with high probability, we will be able to observe enough number of infected nodes in order to estimate the infection source with reasonably good accuracy relative to that for the SI model where all infected nodes are explicit. However, we only require a weaker assumption in this thesis. Throughout this chapter, we assume that for all $u \in V$, we have

$$\max\left(0, 2 - \frac{1}{p_{\mathbf{s}}}\right) \leq q_u \leq 1. \tag{4.1}$$

(Note that $2 - 1/p_{\mathbf{s}} < p_{\mathbf{s}}$.) In the case where $p_{\mathbf{s}} > 1/2$, our assumption requires that q_u is sufficiently large for every node u. This is because given a sufficiently long amount of time, a large number of nodes will become infected, and if most of these nodes are non-explicit, then it becomes very difficult to find a good estimator for the source. On the other hand, if $p_{\mathbf{s}} \leq 1/2$, our assumption is trivial and holds automatically. In this case, we note that our model allows us to set $q_u = 0$ or 1 for each node u, where those nodes with $q_u = 1$ are the ones we actively monitor for the infection. This is similar to [36], except that we do not make the additional assumptions that we know the time an explicit node gets infected, and the neighbor it gets the infection from. The transition probability of a susceptible node is summarized in Fig. 4-1.

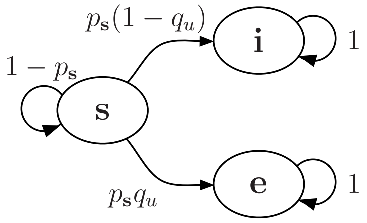


Figure 4-1: Transition probability of a susceptible node u.

Consider the extreme case where the source node has only one neighbor. In this case, the infection can spread in only one direction and any centrality based estimator is expected to perform badly. To avoid this kind of boundary effect, we assume that every node has degree at least two.

We adopt the MLIP criterion as discussed in Section 2.3 and want to find the solution of (2.1), i.e., the source node associated with a *most likely infection path* out of all possible infection paths that are consistent with $V_{\mathbf{i}}$. See Fig. 4-2 for an example of a most likely infection path. Note that there may be many possible most likely infection paths, and our aim is to find the source node of any one of them.

In Section 4.2, we first find a characterization for a most likely infection path when the underlying graph G is a tree, and then derive a source estimator based on that. For general networks G, finding a most likely infection path is difficult. Therefore, in Section 4.3, we propose approximate estimators to identify the source. Extensive simulation results are provided in Section 4.4 to verify the performance of our proposed estimators.

4.1.1 Some Notations and Definitions

In this subsection, we list some notations and definitions that we use in the rest of this dissertation. We refer the reader to the summary of basic notations given in Table 4.1.

For a given tree network A with v being the root, we assign directions to each edge of A so that all edges point towards v. For any $u \in A$, let $\mathrm{pa}(u)$ be the parent node of node u (i.e., the node with an incoming edge from u), and $\mathrm{ch}(u)$ be the set of child nodes of u in A (i.e., the set of nodes with outgoing edges to u).

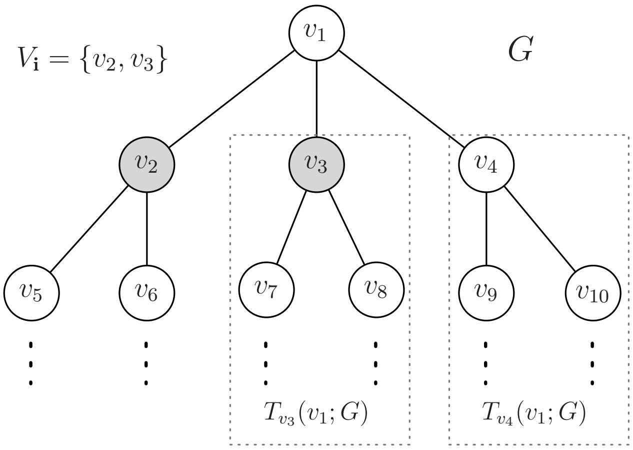


Figure 4-2: An example network G, where shaded nodes are the explicit nodes. Assume that $p_{\mathbf{s}} = 1/2$ and $q_u = 1/2$, $\forall u \in V$. If the elapsed time t is 1, only v_1 can be the source, and the most likely infection path is $X^1(\{v_1, v_2, v_3, v_4\}, 1) = \{\mathbf{i}, \mathbf{e}, \mathbf{e}, \mathbf{s}\}$. The conditional probability of X^1 is $(1 - q_{v_1})(p_{\mathbf{s}} q_{v_2})(p_{\mathbf{s}} q_{v_3})(1 - p_{\mathbf{s}}) = (1/2)^6$. If $t = 2$, the possible source nodes are v_1, v_2, v_3 and v_4. The probability of the most likely infection path for each of these nodes are $(1/2)^9$, $(1/2)^{10}$, $(1/2)^{10}$ and $(1/2)^{11}$, respectively. It can be shown that if $t > 2$, the infection path probabilities become smaller. Therefore, we see that the most likely elapsed time is $t = 1$, the most likely infection path is X^1, and v_1 is our estimated source.

Taking v_1 as the root, we have $\mathrm{pa}(v_2) = v_1$, and $\mathrm{ch}(v_2) = \{v_5, v_6\}$. Since v_7 and v_8 are one hop away from v_3, we have $N_1(v_3; T_{v_3}(v_1; G)) = 2$. The largest distance between v_1 and any explicit node is 1, so $\bar{d}(v_1, V_{\mathbf{i}}) = 1$. We also have $T_{v_4}(v_1; G)$ as an example of a non-observable subtree since all nodes in this subtree do not belong to $V_{\mathbf{i}}$.

Table 4.1: Summary of some notations used in Chapters 4-6

$V_{\mathbf{i}}$	the set of observed infected nodes
H_v	the minimum connected subgraph of G that contains $V_{\mathbf{i}}$ and the node v
$\mathcal{X}_S$	the set of all possible infection paths consistent with $V_{\mathbf{i}}$ conditioned on S being the source set
$\mathcal{T}_S$	the set of the feasible elapsed time corresponding to $\mathcal{X}_S$
$\|A\|$	the number of elements in A if A is a set, or the number of nodes in A if A is a graph
$V_u(i)$	the set of nodes i hops away from node u
$T_u(v; A)$	the subtree rooted at node u of the tree A, with the first link of the path from u to v in A removed
$d(s, u)$	the length of the shortest path between s and u in the graph G (i.e., the distance between them)
t^S	a most likely elapsed time conditioned on S being the infection source set

For any infection path X^t, a subset $J \subset V$, and $0 \leq i \leq j \leq t$, let $X^t(J, [i, j])$ be the states of nodes in J from time slots i to j in the infection path X^t. To avoid cluttered expressions, we abuse notations and let

$$P_S\left(X^t(J, [i, j])\right) \triangleq \mathbb{P}(\mathbf{X}^t(J, [i, j]) = X^t(J, [i, j]) \mid S^* = S).$$

Therefore, $P_S(X^t)$ represents the probability of X^t conditioned on S being the sources and t being the elapsed time. Moreover, when we want to remind the reader of the state of a node u at a specific time in the conditional probability $P_S(X^t)$, we use the notation $P_S(X(u, i) = a)$, where a is the state of u at time i.

Definition 4.1 (Most likely infection paths)**.** For any $S \subset V$ and any feasible elapsed time $t \in \mathcal{T}_S$, we say that

- an infection path X^t is most likely for (S, t) if $X^t \in \arg\max_{\tilde{X}^t \in \mathcal{X}_S} P_S(\tilde{X}^t)$;
- an infection path X^t is most likely for S if $(X^t, t) \in \arg\max_{t' \in \mathcal{T}_S, \tilde{X}^{t'} \in \mathcal{X}_S} P_S(\tilde{X}^{t'})$.

Moreover, for a given $k > 0$ number of infection sources, an infection path X^t is called a most likely infection path if there exists some $S \subset V$, $|S|= k$ and $t \in \mathcal{T}_S$ such that

$$P_S(X^t) = \max_{A \subset V, |A|=k, t' \in \mathcal{T}_A, \tilde{X}^{t'} \in \mathcal{X}_A} P_A(\tilde{X}^{t'}).$$

In the following, we introduce a generalization of the Jordan center of a graph. If we set $k = 1$ in Definition 4.2, we obtain the original Jordan center definition [64].

Definition 4.2 (k-Jordan center set)**.** For a given set of nodes $S = \{s_1, s_2, \cdots, s_k\}$ and a node u, the distance between S and u is the smallest distance between any node $s_i \in S$ and u in the graph G, i.e.,

$$d(S, u) \triangleq \min_{s_i \in S} d(s_i, u).$$

For any set of nodes A in the graph G, we let the eccentricity $\bar{d}(S, A)$ of S with respect to (w.r.t.) A be the largest distance between S and any node in A, given by

$$\bar{d}(S, A) \triangleq \max_{u \in A} d(S, u).$$

The set of k nodes in G with minimum eccentricity w.r.t. A is called the k-Jordan center set of A. For simplicity, we say that $\bar{d}(S, V_{\mathbf{i}})$ is the infection range of the set S.

Finally, in several of our proofs, we need to differentiate between subtrees that have observed infected nodes or not.

Definition 4.3 (Non-observable subtree and observable subtree)**.** Suppose that v is the infection source. For any node u, we say that $T_u(v; G)$ is an *non-observable* if [1]

$$T_u(v; G) \bigcap V_{\mathbf{i}} = \emptyset;$$

and we say that $T_u(v; G)$ is an *observable subtree* if

$$T_u(v; G) \bigcap V_{\mathbf{i}} \neq \emptyset.$$

4.2 Source Estimation for Trees

In this section, we consider the case where the underlying network G is a tree. We first derive some properties of a most likely infection path, and then show that a Jordan center of $V_{\mathbf{i}}$ is an optimal source estimator under the MLIP criterion.

4.2.1 A Most Likely Infection Path

In this subsection, we show that although we have assumed that G is an infinite tree, we can restrict our search for a most likely infection path for v to the subgraph of G with nodes within the infection range $\bar{d}(v, V_{\mathbf{i}})$ of v. In the following lemma, we first show that for any source node $v \in V$, a most likely infection path for v with a finite number of infected nodes can be found. Its proof is provided in Section 4.5.

[1] See Table 4.1 for the definition of $T_u(v; G)$.

Lemma 4.1. Suppose that G is a tree. Then, for any node $v \in V$, and feasible elapsed time $t \in \mathcal{T}_v$, there exists a most likely infection path X^t for (v, t) such that for any $u \in V$ with non-observable $T_u(v; G)$, we have $X(u, \tau) \neq \mathbf{i}$ for all $\tau \leq t$.

The following lemma, whose proof is given in Section 4.5, shows that given the elapsed time t, a most likely infection path for a node v is given by a path whose nodes "resist" the infection, and each node becomes infected only at the latest possible time.

Lemma 4.2. Suppose that G is a tree, and the infection source is $v \in V$. Suppose that the elapsed time is $t \in \mathcal{T}_v$. Then, for any $u \in H_v \backslash \{v\}$, the first infection time t_u for u in any infection path is bounded by

$$d(v, u) \leq t_u \leq t - \bar{d}(u, T_u(v; H_v)). \tag{4.2}$$

Furthermore, there exists a most likely infection path X^t for (v, t) such that the first infection time for $u \in H_v \backslash \{v\}$ is given by

$$t_u = t - \bar{d}(u, T_u(v; H_v)). \tag{4.3}$$

Lemma 4.1 and Lemma 4.2 characterize a most likely infection path consistent with $V_{\mathbf{i}}$, with the property that a minimum number of nodes are infected, and each infected node becomes infected at the latest possible time. We call this *the latest infection path.*

Definition 4.4. Suppose that G is a tree. For any $v \in V$, and any feasible elapsed time $t \in \mathcal{T}_v$, the latest infection path X^t for (v, t) is the infection path that satisfies the following properties:

(i) $X^t(u, \tau) \in \{\mathbf{s}, \mathbf{n}\}$ for all $u \notin H_v$ and for all $\tau \leq t$.

(ii) For each $u \in H_v \backslash \{v\}$, the first infection time of u is $t_u = t - \bar{d}(u, T_u(v; H_v))$.

The following proposition then follows from Lemma 4.1 and Lemma 4.2.

Proposition 4.1. Suppose that G is a tree. Then for any $v \in V$, and any feasible elapsed time $t \in \mathcal{T}_v$, the latest infection path for (v, t) is a most likely infection path for (v, t).

We now show that a most likely infection path for any $v \in V$ is the latest infection path for (v, t), where t is chosen to be as small as possible.

Proposition 4.2. Suppose that G is a tree. For any $v \in V$, we have the following.

(a) The set of all feasible elapsed times is $\mathcal{T}_v = [\bar{d}(v, V_{\mathbf{i}}), \infty)$.

(b) For the sequence of latest infection paths $\{X^t\}_{t \in \mathcal{T}_v}$, we have $P_v(X^t)$ is non-increasing in $t \in \mathcal{T}_v$.

(c) A most likely elapsed time is given by $\bar{d}(v, V_{\mathbf{i}})$, and a most likely infection path for v is the latest infection path for $(v, \bar{d}(v, V_{\mathbf{i}}))$.

Proof. Claim (a) follows because the infection can propagate at most one hop further from the source node v in one time slot, therefore if $t < \bar{d}(v, V_{\mathbf{i}})$, the infection can not reach the explicit nodes $\bar{d}(v, V_{\mathbf{i}})$ hops away from v.

Next, we show claim (b). Fix a $t \in \mathcal{T}_v$ and consider the latest infection paths X^t and X^{t+1}. Suppose that $H_v \neq \{v\}$, then from Definition 4.4, we have $X^{t+1}(V, [2, t+1]) = X^t(V, [1, t])$ and $X^{t+1}(w, 1) = \mathbf{s}$ for all $w \in N_1(v; H_v)$, yielding

$$\begin{aligned}\frac{P_v(X^{t+1})}{P_v(X^t)} \leq & \prod_{w \in N_1(v;H)} P_v(X^{t+1}(w, 1)) \\ =& (1 - p_{\mathbf{s}})^{|N_1(v;H_v)|} \leq 1,\end{aligned}$$

where the last inequality follows because $H_v \neq \{v\}$ and $|N_1(v; H_v)| > 1$. On the other hand if $H_v = \{v\}$, the infection does not spread from v in both latest paths X^t and X^{t+1}, and we have

$$\frac{P_v(X^{t+1})}{P_v(X^t)} = (1 - p_{\mathbf{s}})^{|N_1(v;G)|} \leq 1,$$

since all neighbors of v remain susceptible throughout the elapsed time, and we have assumed that the degree of v is at least two. This proves claim (b). Claim (c) now follows from claim (a) and claim (b), and the proof of the lemma is complete. □

4.2.2 Source Associated with a Most Likely Infection Path

In this subsection, we derive the source estimator associated with a most likely infection path. We first show that we can find an infection path for a node with a smaller infection range that is not less likely than any most likely infection path of another node with a larger infection range. This in turn implies that a Jordan center of $V_{\mathbf{i}}$ is the source estimator we are looking for. We start with two lemmas that show the relationship between the latest infection paths of two neighboring nodes.

Lemma 4.3. Suppose that G is a tree. Then, for any pair of neighboring nodes u and v in $H_v \bigcup H_u$ with $\bar{d}(v, V_{\mathbf{i}}) < \bar{d}(u, V_{\mathbf{i}})$, we have

(i) $l \in T_v(u; H_v \bigcup H_u)$, for all $l \in \arg\max_{x \in V_{\mathbf{i}}} d(u, x)$; and

(ii) $\bar{d}(v, V_{\mathbf{i}}) = \bar{d}(u, V_{\mathbf{i}}) - 1$, and there exists $l \in T_v(u; H_v \bigcup H_u)$ such that $d(v, l) = \bar{d}(v, V_{\mathbf{i}})$.

Proof. To prove (i), we note that if $l \notin T_v(u; H_v \bigcup H_u)$, we have $\bar{d}(v, V_{\mathbf{i}}) \geq d(v, l) = d(u, l) + 1 = \bar{d}(u, V_{\mathbf{i}}) + 1$, a contradiction. Therefore, (i) holds. Then for l such that $d(u, l) = \bar{d}(u, V_{\mathbf{i}})$, we have $\bar{d}(v, V_{\mathbf{i}}) \geq d(v, l) = d(u, l) - 1 = \bar{d}(u, V_{\mathbf{i}}) - 1$ since $l \in T_v(u; H_v \bigcup H_u)$. This implies (ii), and the lemma is proved. □

Lemma 4.4. Suppose that G is a tree, and let H be the minimum connected subtree of G spanning $V_{\mathbf{i}}$. Then, for any pair of neighboring nodes u and v in H with $d_v = \bar{d}(v, V_{\mathbf{i}}) < d_u = \bar{d}(u, V_{\mathbf{i}})$, we have

$$P_v(X^{d_v}) \geq P_u(X^{d_u}),$$

where X^{d_v} and X^{d_u} are the latest infection paths for (v, d_v) and (u, d_u) respectively.

Proof. To prove the lemma, it suffices to construct an infection path $\tilde{X}^{d_v}$ with source node v, and show that it has at least the same conditional probability as that for X^{d_u}. Let t_v be the first infection time of node v in the infection path X^{d_u}. We first show that $t_v = 1$. Since u is the infection source, the infection can propagate at most $d_u - t_v$ hops away from node v within the subtree $T_v(u; H)$. From Lemma 4.3(ii), if $t_v > 1$, we have $d_v = d_u - 1 > d_u - t_v$, a contradiction. Therefore, we must have $t_v = 1$ in the infection path X^{d_u}. Let $\tilde{X}^{d_v}(T_v(u; H), [1, d_v]) = X^{d_u}(T_v(u; H), [2, d_u])$, and we have

$$\frac{P_u(X^{d_u}(T_v(u; H), [1, d_u]))}{P_v(\tilde{X}^{d_v}(T_v(u; H), [1, d_v]))} = p_{\mathrm{s}}, \tag{4.4}$$

where the equality holds because the probability that v is explicit or not appears in both the numerator and denominator.

Consider any node $w \in N_1(u, T_u(v; H))$. Since $d(v, w) = 2$, it takes at least two time slots for an infection starting at v to reach w. Moreover, since $d_u = d_v + 1$, by Lemma 4.2 and Lemma 4.3(i), the first infection time of w in the path X^{d_u} is at least 3. Therefore, we can set $\tilde{X}^{d_v}(T_u(v; H), [1, d_v]) = X^{d_u}(T_u(v; H), [2, d_u])$, and we obtain

$$\begin{aligned}\frac{P_u(X^{d_u}(T_u(v; H), [1, d_u]))}{P_v(\tilde{X}^{d_v}(T_u(v; H), [1, d_v]))} &= \frac{1}{p_{\mathrm{s}}}(1 - p_{\mathrm{s}})^{2|N_1(u, T_u(v; H))|} \\ &\le \frac{1}{p_{\mathrm{s}}}, \end{aligned} \tag{4.5}$$

where the inequality follows by the assumption that every node has degree at least two. Multiplying (4.4) by (4.5), we obtain

$$\frac{P_u(X^{d_u}(H, [1, d_u]))}{P_v(\tilde{X}^{d_v}(H, [1, d_v]))} \le p_{\mathrm{s}} \cdot \frac{1}{p_{\mathrm{s}}} = 1. \tag{4.6}$$

Finally, we consider the nodes in $G \backslash H$. Let $T_z(v; G)$ be any non-observable subtree such that $z \in G \backslash H$ has a neighboring node $w \in H$. By Lemma 4.1, z remains uninfected in X^{d_u}. Let z stay uninfected in $\tilde{X}^{d_v}$ as well. Let the node w first become infected at time $t_w(u)$ in X^{d_u} and at time $t_w(v)$ in $\tilde{X}^{d_v}$. Then, z stays uninfected in X^{d_u} and $\tilde{X}^{d_v}$ with probabilities $(1-p_{\mathbf{s}})^{d_u - t_w(u)}$ and $(1-p_{\mathbf{s}})^{d_v - t_w(v)}$, respectively. Since $d_v = d_u - 1$ and $t_w(v) \geq t_w(u) - 1$, we have $d_u - t_w(u) \geq d_v - t_w(v)$, and $P_u(X^{d_u}(G\backslash H, [1, d_u])) \leq P_v(\tilde{X}^{d_v}(G\backslash H, [1, d_v]))$. Combining this with (4.6), we conclude that $P_u(X^{d_u}) \leq P_v(\tilde{X}^{d_v})$, and the proof is complete. □

We are finally ready to show that the a Jordan center of $V_{\mathbf{i}}$ is an optimal single source estimator for (2.1).

Theorem 4.1. Suppose that G is a tree and the set of observed infected nodes $V_{\mathbf{i}}$ is non-empty. For an infection under the SI model with a single infection source, a Jordan center of $V_{\mathbf{i}}$ is an optimal source estimator for (2.1).

Proof. It can be shown that if G is a tree, then there are at most two Jordan centers for $V_{\mathbf{i}}$, and if there are indeed two Jordan infection centers, they are neighboring nodes [37]. If there are two neighboring Jordan centers, we can treat them as a single virtual node, therefore without loss of generality, we assume that there is only one Jordan center $\hat{s}$. Let H be the minimum connected subtree of G spanning $V_{\mathbf{i}}$. Then $\hat{s} \in H$. Consider any path $(\hat{s}, v_1, v_2, \cdots, v_m)$ in H, where $m \geq 1$. We show that

$$\bar{d}(\hat{s}, V_{\mathbf{i}}) < \bar{d}(v_1, V_{\mathbf{i}}) < \ldots < \bar{d}(v_m, V_{\mathbf{i}}). \tag{4.7}$$

The first inequality in (4.7) holds by assumption. We now show that the rest of the inequalities in (4.7) also hold. Choose a $l \in V_{\mathbf{i}}$ such that $d(\hat{s}, l) = \bar{d}(\hat{s}, V_{\mathbf{i}})$. If $l \notin T_{v_1}(\hat{s}; H)$, we have $\bar{d}(v_i, V_{\mathbf{i}}) = d(\hat{s}, l) + i$ for $1 \leq i \leq m$, so (4.7) holds. Suppose now that $l \in T_{v_1}(\hat{s}; H)$. Note that the set $A = V_{\mathbf{i}} \backslash T_{v_1}(\hat{s}; H)$ is non-empty, otherwise $\bar{d}(v_1, V_{\mathbf{i}}) < \bar{d}(\hat{s}, V_{\mathbf{i}})$ and $\hat{s}$ cannot be a Jordan center. Consider a node l' such that $l' = \arg\max_{v \in A} d(\hat{s}, v)$. If $d(\hat{s}, l') \leq d(\hat{s}, l) - 2$, we have

$$\begin{aligned}\bar{d}(v_1, V_{\mathbf{i}}) &= \max\left(d(v_1, l'), d(v_1, l)\right)\\ &= \max\left(d(\hat{s}, l') + 1, d(\hat{s}, l) - 1\right)\\ &= d(\hat{s}, l) - 1,\end{aligned}$$

and the infection range of v_1 is less than that of $\hat{s}$, a contradiction. Therefore, we have $d(\hat{s}, l') \geq d(\hat{s}, l) - 1$. Suppose that $\bar{d}(v_{i+1}, V_{\mathbf{i}}) \leq \bar{d}(v_i, V_{\mathbf{i}})$ for some $i \in [1, m-1]$. Let $\tilde{l}$ be a node such that $d(v_i, \tilde{l}) = \bar{d}(v_i, V_{\mathbf{i}})$. Then, we must have $\tilde{l} \in T_{v_{i+1}}(v_i; H)$, otherwise we have a contradiction. We then have

$$\begin{aligned}\bar{d}(v_{i+1}, V_{\mathbf{i}}) &\geq d(v_{i+1}, l') \\ &= d(\hat{s}, l') + i + 1 \\ &\geq i + d(\hat{s}, l) \\ &\geq 2i + d(v_i, \tilde{l}) \\ &> d(v_i, \tilde{l}),\end{aligned}$$

a contradiction. Therefore (4.7) holds. By repeatedly applying Lemma 4.4, Proposition 4.1 and Proposition 4.2(c), we have that any most likely infection path for $\hat{s}$ has at least the same probability as that for v_m, and the theorem is proved. □

4.2.3 Finding a Jordan Center

A centralized linear time complexity algorithm has been proposed in [65] to find the Jordan center in a tree. It first computes the diameter of the tree and then returns a midpoint of any longest path in the tree as the Jordan center. In this subsection, we present a message passing algorithm, somewhat similar to that of [65] in the quantities being computed at each node, but which can be implemented in a distributed fashion. Our algorithm also has linear time complexity.

Let H be the minimum connected subtree of G spanning $V_{\mathbf{i}}$. We assume that $|H| > 2$ since otherwise finding the Jordan center is trivial. Our proposed Jordan Center Estimation (JCE) algorithm is formally presented in Algorithm 4.1. The main idea behind the algorithm is that for $|H| > 2$, a Jordan center v must satisfy the following necessary and sufficient conditions: (i) it has degree at least 2 in H, and (ii) if $M = \{\rho_u : u \in N_1(v; H),\ \rho_u$ is a path with the maximum length among all paths with first edge being $(v, u)\}$, then the difference in lengths of the longest and second longest paths in M is at most 1. This can be shown using the same arguments as that in the proof of Theorem 4.1.

JCE first randomly chooses a non-leaf node $r \in H$ as the root node. It then performs an Upward Message-passing procedure, starting from the leaf nodes up to the root r, where the message passed from a node v to its parent node $\text{pa}(v)$ consists of its own identity and the length of the longest path in $T_v(r; H)$.

This upward message-passing procedure terminates when the root receives all messages from its child nodes. The details of the upward message-passing procedure are shown in lines 4 to 13 in Algorithm 4.1. Since each node only passes one message to its parent, the overall complexity of the upward message-passing procedure is $O(|H|)$.

In the Downward Message-passing procedure, the root node r first identifies the two paths in M with the longest lengths $\ell_1(r)$ and $\ell_2(r)$. If $\ell_1(v) - \ell_2(v) \leq 1$, JCE returns r as the Jordan center. Otherwise, it computes a message $g_r(r^{(1)}) = \ell_2(r) + 1$ and sends to $r^{(1)}$, the child node with the longest path in the Upward Message-passing procedure. The same process is repeated until a leaf node is reached. The details are presented in lines 14 to 24 in Algorithm 4.1. The complexity of the downward message-passing process is bounded by the diameter of H. As a result, the overall complexity of JCE is $O(|H|)$.

4.3 Source Estimation for General Networks

In this section, we derive an approximate source estimator for the case where the underlying network G is a general network. We also suggest heuristic algorithms to find our proposed source estimator.

Suppose that the neighbor from which a susceptible node obtains its infection is randomly chosen from one of its infected neighbors. Then, the path traced out by an infection spreading in G is a tree. For any infection path X^t, let $T(X^t)$ be the subtree of G traced out by X^t. Any tree T_v with root v, for which there exists an infection path consistent with V_i is said to be an infection tree consistent with V_i. Let $\mathbb{T}_v$ be

Algorithm 4.1 Jordan Center Estimation (JCE) Algorithm

```
**Input**: $H$ is the minimum connected subtree of $G$ spanning $V_{\mathbf{i}}$, with $|H|> 2$.
**Output**: $\hat{s}$, the Jordan center for $V_{\mathbf{i}}$.
**Initialization**: randomly select a non-leaf node $r \in H$ as the root node
**Upward Message-passing:**
for each $v \in H$ do
    if $v$ is a leaf then
        $f_v(\mathrm{pa}(v)) = 1$
    else
        Store $v^{(1)} = \arg\max_{u\in \mathrm{ch}(v)} f_u(v)$, $\ell_1(v) = f_{v^{(1)}}(v)$, and $\ell_2(v) = \max_{u\in C} f_u(v)$,
        where $C = \mathrm{ch}(v)\backslash\{v^{(1)}\}$, with $\ell_2(v) = 0$ if $C = \emptyset$. Ties are broken randomly.

        $f_v(\mathrm{pa}(v)) = \ell_1(v) + 1$
    end if
    Pass $f_v(\mathrm{pa}(v))$ and its identity to $\mathrm{pa}(v)$
end for
**Downward Message-passing:**
for each $v \in H$ starting from root $r$ do
    if $v$ is not the root $r$ then
        $\ell_2(v) = \max(\ell_2(v), g_{\mathrm{pa}(v)}(v))$
    end if
    if $\ell_1(v) - \ell_2(v) \leq 1$ then
        $\hat{s} = v$
    else
        Pass $g_v(v^{(1)}) = \ell_2(v) + 1$ to $v^{(1)}$
    end if
end for
return $\hat{s}$
```

the set of infection trees consistent with $V_{\mathbf{i}}$, and have source node v. Then, we have

$$\max_{v\in V} \max_{\substack{t\in\mathcal{T}_v \\ X^t\in\mathcal{X}_v}} P_v(X^t) = \max_{v\in V} \max_{T\in\mathbb{T}_v} \max_{\substack{t\in\mathcal{T}_v \\ X^t:T(X^t)=T}} P_v(X^t), \tag{4.8}$$

which implies that to find a source node associated with the most likely infection path in (2.1), we first find the most likely infection tree consistent with $V_{\mathbf{i}}$, and then find a most likely infection path that traces out this infection tree using the results in Section 4.2.2. However, finding the set of infection trees consistent with $V_{\mathbf{i}}$ is difficult. We derive a simple property that $\mathbb{T}_v$ must satisfy for each v, suggest an approximation for it, and provide two heuristic methods to find the approximate most likely infection tree.

Lemma 4.5. Suppose that G is a general network, and $v \in V$ is the infection source. Then, there is no loss in optimality in (4.8) if we restrict $\mathbb{T}_v$ to be the set of all infection trees consistent with $V_\mathbf{i}$ that have all non-source leaf nodes in $V_\mathbf{i}$.

Proof. The proof follows from Lemma 4.1 because any non-observable leaf node is the root of a non-observable subtree. □

From Lemma 4.5 and the fact that the source estimator for a tree is given by a Jordan center, we construct a subgraph G_v of G for each $v \in V$, by first finding a shortest path tree from v to each node in $V_\mathbf{i}$, and then adding all edges in G incident to the nodes of the shortest path tree. We approximate $\mathbb{T}_v$ by the set of spanning trees of G_v, denoted as $\hat{\mathbb{T}}_v$, and adopt the approximate source estimator given by

$$\tilde{s} = \max_{v \in V} \max_{T \in \hat{\mathbb{T}}_v} \max_{\substack{t \in \mathcal{T}_v \\ X^t : T(X^t) = T}} P_v(X^t). \tag{4.9}$$

We now turn to finding the most likely infection tree in the set $\hat{\mathbb{T}}_v$. We start with a characterization of its probability in the following result. For any source node v, and any infection tree $T \in \hat{\mathbb{T}}_v$, let $D_T(u) = \bar{d}(u, T_u(v; T))$ be the height of the subtree of T rooted at u.

Lemma 4.6. Suppose that G is a general network, and $v \in V$ is the infection source. Then, for any infection tree T with root v and consistent with $V_\mathbf{i}$, we have

$$\begin{aligned} &\max_{\substack{t \in \mathcal{T}_v \\ X^t : T(X^t) = T}} P_v(X^t) \\ &= p_\mathbf{s}^{|T|-1}(1-p_\mathbf{s})^{\sum_{u \in T \backslash \{v\}}(D_T(\mathrm{pa}(u)) - D_T(u)) - |T| + 1} \prod_{u \in V_\mathbf{i}} q_u \prod_{u \in T \backslash V_\mathbf{i}} (1 - q_u). \end{aligned} \tag{4.10}$$

Proof. Every non-source node $u \in T \backslash \{v\}$ gets infected during the infection spreading process, and this occurs with probability $p_\mathbf{s}^{|T|-1}$. From Lemma 4.2 and Proposition 4.2, in a most likely infection path for v, the first infection time for any node $u \in T$ is $\tilde{t}_u = d - D_T(u)$, where $d = \bar{d}(v, V_\mathbf{i})$. The parent node $\mathrm{pa}(u)$ gets infected at time

$\tilde{t}_{\mathrm{pa}(u)} = d - D_{\mathrm{pa}(u)}$, therefore there are $\tilde{t}_u - \tilde{t}_{\mathrm{pa}(u)} - 1 = D_T(\mathrm{pa}(u)) - D_T(u) - 1$ time slots in which that u remains susceptible, and this occurs with probability $(1 - p_{\mathbf{s}})^{D_T(\mathrm{pa}(u)) - D_T(u) - 1}$. Taking the product over all non-source nodes in $\mathbb{T}_v$, we obtain the lemma. □

From Lemma 4.6, we see that our proposed estimator in (4.9) can be found by finding the node v that maximizes

$$|T_v^*| \log p_{\mathbf{s}} + (F_v^* - |T_v^*|) \log(1 - p_{\mathbf{s}}) + \sum_{u \in T_v^* \setminus V_{\mathbf{i}}} \log(1 - q_u),$$

where

$$F_v^* = \min_{T \in \hat{\mathbb{T}}_v} \sum_{u \in T \setminus \{v\}} \left(D_T(\mathrm{pa}(u)) - D_T(u) \right), \tag{4.11}$$

and T_v^* is the minimizer in the right hand side of (4.11).

In the following, we propose two heuristic methods to find T_v^* and F_v^* for any $v \in V$. Although we have assumed that G is infinite in Section 4.1 to simplify our theoretical analysis, in practice, we have access to only a finite graph G. We then iterate either of our proposed methods over all nodes in G.

4.3.1 MIQCQP

For any $v \in V$, let G_v be the subgraph of G used in defining $\hat{\mathbb{T}}_v$ (cf. discussion preceding (4.9)). By expanding the telescoping sum in (4.11), it can be shown that the objective function in (4.11) is equivalent to the objective function of the following

MIQCQP,

$$\min \sum_{i,j \in G_v} E_{ij} D_i - \sum_{i \in G_v \setminus \{v\}} D_i \tag{4.12}$$

$$\text{subject to } \sum_{j \in G_v} E_{ji} = \begin{cases} 0, & \text{if } i = v \\ 1, & \text{if } i \neq v \end{cases} \tag{4.13}$$

$$\sum_{i,j \in G_v} E_{ij} = |G_v| - 1 \tag{4.14}$$

$$E_{ij} \in \{0,1\}, \quad \forall i,j \in G_v \tag{4.15}$$

$$D_i \geq (D_j + 1) E_{ij}, \quad \forall i \in G_v, \forall j \in N_1(i; G_v) \tag{4.16}$$

$$D_i \leq \sum_{j \in N_1(i;G_v)} (D_j + 1) E_{ij}, \quad \forall i \in G_v \tag{4.17}$$

where $E_{ij} = 1$ if and only if the edge (i,j) is chosen as part of T_v^*, and D_i is a variable corresponding to $D_{T_v^*}(i)$.

The above MIQCQP does not find the exact optimal T_v^* in (4.11) because of several approximations that we have made. The constraints (4.13)-(4.15) restrict the feasible solution space to the spanning trees of G_v. However, it is not possible represent the quantities $D_T(u)$ in (4.11) exactly as quadratic constraints. We use the quadratic constraints (4.16)-(4.17) as approximations. The constraint (4.16) derives from the fact that for all nodes i in a tree T spanning G_v, we have $D_T(i) \geq D_T(j) + 1$ for all j that are child nodes of i in T. In addition, for a non-source node i that has only one child node j, the value D_i is canceled out in (4.12). In order to improve the convergence speed, we introduce the quadratic constraint (4.17) as an upper bound for D_i.

The MIQCQP in (4.12) can be solved using the OPTI Toolbox [66], which utilizes Solving Constraint Integer Programs (SCIP) [67] as the solvers. However, these solvers have high complexity, and are not suitable for large networks. Therefore, we propose an alternative, low-complexity heuristic algorithm in the following and compare their performances in Section 4.4.

4.3.2 Reverse Greedy

It can be shown that

$$F_v^* = \min_{T \in \mathbb{T}_v} \sum_{u \in T} (\mathrm{Deg}_T(u) - 2) D_T(u) + 2D_T(v), \tag{4.18}$$

where $\mathrm{Deg}_T(u)$ is the number of neighboring nodes of u in T. We see from (4.18) that to obtain the minimizer, the degree of u should be chosen to be small if $D_T(u)$ is large. Based on this intuition, we propose a heuristic algorithm in Algorithm 4.2, which we call the Reverse Greedy (RG) algorithm. The algorithm attempts to adjust the infection tree found so that those nodes close to the source node has lower degrees, while those further away have correspondingly higher degrees.

Given a node $v \in V$, RG first constructs a shortest path tree T rooted at v and spanning G_v using a breadth first search algorithm [68], incurring a time complexity of $O(|G_v|)$. Starting with a leaf node x that is furthest away from v, RG adjusts the tree T by choosing a neighboring node y of x with the largest $d(v, y)$ value (with ties broken by another criterion given in lines 6 to 17 in Algorithm 4.2), and attaching x to y. This procedure is repeated up the tree T until the node v is reached. At each node x, the number of neighbors is $O(|G_v|)$ and lines 19 to 28 has a time complexity of $O(|G_v|)$. Therefore, the complexity of RG is $O(|G_v|^2)$. Since we need to iterate over all nodes in order to find the source estimator, the overall complexity is $O(|V|^3)$.

4.4 Simulation Results

In this section, we present simulation results using both synthetic and real world networks to evaluate the performance of the proposed estimators. We use the following three common centrality measures as benchmarks to compare with our estimators.

(i) The distance centrality of $v \in G$ is defined as

$$C_D(v) = \sum_{i \in V_{\mathbf{I}}} d(v, i),$$

Algorithm 4.2 Reverse Greedy (RG) Algorithm

```
**Inputs**: $v \in V$, and $G_v$ (cf. discussion preceding (4.9))
**Outputs**: $F_v^*$ in (4.18) and its minimizer $T_v^*$.
Construct a shortest path tree $T$ rooted at $v$ and that spans $G_v$ using breadth first search [68]. Compute $D_T(u)$ and $U_T(u) = d(v, u)$ for all $u \in T$.
for $d = D_T(v)$ DownTo 1 do
    for each node $x \in \{u : U_T(u) = d\}$, in order of increasing $D_T(x)$ do
        Set $Y = \emptyset$
        for each neighboring node $y$ of $x$ in $G_v$ do
            if $U_T(y) \leq D_T(v) - D_T(x) - 1$ then
                Add $y$ into set $Y$
                if $y$ is the parent of $x$ in $T$ then
                    Set $D'_T(y)$ to be $\bar{d}(y, T_y(v;T) \backslash T_x(v;T))$
                else
                    Set $D'_T(y) = D_T(y)$
                end if
            end if
        end for
        Choose $y \in Y$ with the largest $U_T(y)$, with ties broken by choosing the $y$ with the largest $D'_T(y)$.
        Modify $T$ by removing the edge between $x$ and its parent and connecting $x$ to $y$.
        Set $\Delta_x = U_T(y) + 1 - U_T(x)$
        if $\Delta_x \neq 0$ then
            Set $U_T(z) = U_T(z) + \Delta_x, \ \forall z \in T_x(v;T)$
        end if
        Set $\Delta_T(y) = \max(D_T(y), D_T(x) + 1) - D_T(y)$
        while $\Delta_y \neq 0$ do
            Set $D_T(y) = D_T(y) + \Delta_y$
            Set $\Delta_y = \max(D_T(\mathrm{pa}(y)), D_T(y) + 1) - D_T(\mathrm{pa}(y))$
            Set $y = \mathrm{pa}(y)$
        end while
    end for
end for
return $T_v^* = T$ and $F_v^*$ computed using (4.18).
```

and the node with minimum distance centrality is called the distance center (DC). It is shown in [23] that the DC is the ML estimator for regular trees under a SI model where all nodes are explicit.

(ii) The closeness centrality of $v \in G$ is defined as

$$C_C(v) = \sum_{i \in V_{\mathbf{i}}, i \neq v} \frac{1}{d(v,i)},$$

and the node with maximum closeness centrality is called the closeness center (CC).

(iii) The betweenness centrality of $v \in G$ is defined as

$$C_B(v) = \sum_{i,j \in V_{\mathbf{i}}, i \neq j \neq v} \frac{\sigma_{ij}(v)}{\sigma_{ij}},$$

where σ_{ij} is the number of shortest paths between node i and node j, and $\sigma_{ij}(v)$ is the number of those shortest paths that contain v. We call the node with maximum betweenness centrality the betweenness center (BC).

4.4.1 Tree Networks

We evaluate the performance of the JCE algorithm on three kinds of synthetic tree networks: regular tree networks where the node degree is randomly chosen from $[3, 6]$, and two types of random trees, denoted as random-1 and random-2, where the degree of every node is randomly chosen from $[3, 6]$ and $\{3, 6\}$ respectively. Note that random-1 trees are less symmetric than regular trees, and random-2 trees are even less symmetric. For each kind of synthetic tree network, we perform 1000 simulation runs. In each simulation run, we randomly generate a tree, and choose a node to be the infection source. Then we simulate the infection using the SI model, where $p_{\mathbf{s}}$ is chosen uniformly from $(0, 1)$ and q_v is chosen uniformly from $[\max(0, 2 - 1/p_{\mathbf{s}}), 1]$ for each node v. The spreading terminates when the number of infected nodes is greater than 200. We then run JCE on the explicit nodes to estimate the infection source and compare the results with the benchmarks.

The error distance is the number of hops between the estimated and the actual infection source, and is shown in Fig. 4-3. We see that as the underlying network

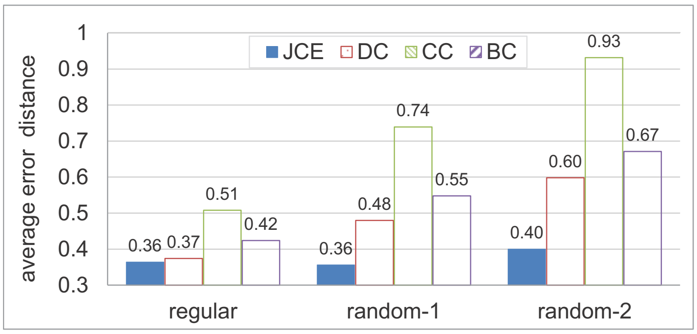


Figure 4-3: Average error distances for different tree networks. The average diameters of the regular, random-1, and random-2 trees are 14, 13 and 13 hops, respectively.

becomes less symmetric, the error distances of all benchmarks increase, while that of JCE remains relatively stable. In practice, the structures of the underlying tree networks are usually far from symmetric.

4.4.2 General Networks

We evaluate the performance of the MIQCQP and RG algorithms on three kinds of general networks: synthetic small-world networks [63], the western states power grid network of the United States [63] and a small part of the Facebook network with 4039 nodes [69], which is shown in [70] to be a scale-free network.

Since the MIQCQP approach has very high complexity with an average running time of 25 minutes for the small-world networks (the average running time of RG is 0.8 seconds), we restrict the number of infected nodes to be 50 in the first comparison. In each simulation run, the infection probability $p_{\mathbf{s}}$ is chosen uniformly from $(0, 1)$ and q_v is chosen uniformly from $[\max(0, 2 - 1/p_{\mathbf{s}}), 1]$ for each node v. Fig. 4-4 shows that both MIQCQP and RG have smaller average error distances than all benchmarks, with RG only slightly worse than MIQCQP. Therefore, in the rest of the simulations, we will only compare RG with the benchmarks.

In the next comparison, we consider infection sizes of more than 200 nodes. In each simulation run, we randomly choose a fraction of the infection nodes to be explicit. We call this fraction the explicit ratio. We perform simulations with the explicit ratio ranging from 10% to 100%. Fig. 4-5 shows the average error distances of RG and

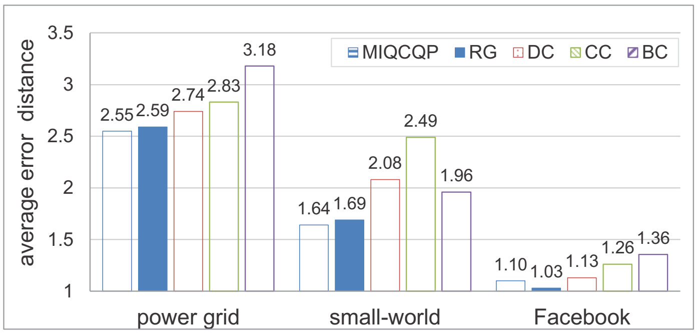


Figure 4-4: Average error distances for different general networks in the first comparison. The average diameters of the power grid network, small-world network, and Facebook network are 46, 20, and 18 hops, respectively.

all benchmarks for different explicit ratios on all three kinds of general networks. We see that RG has smaller average error distances in almost all cases.

4.5 Proofs

In this section, we provides proofs of some of the results in this chapter.

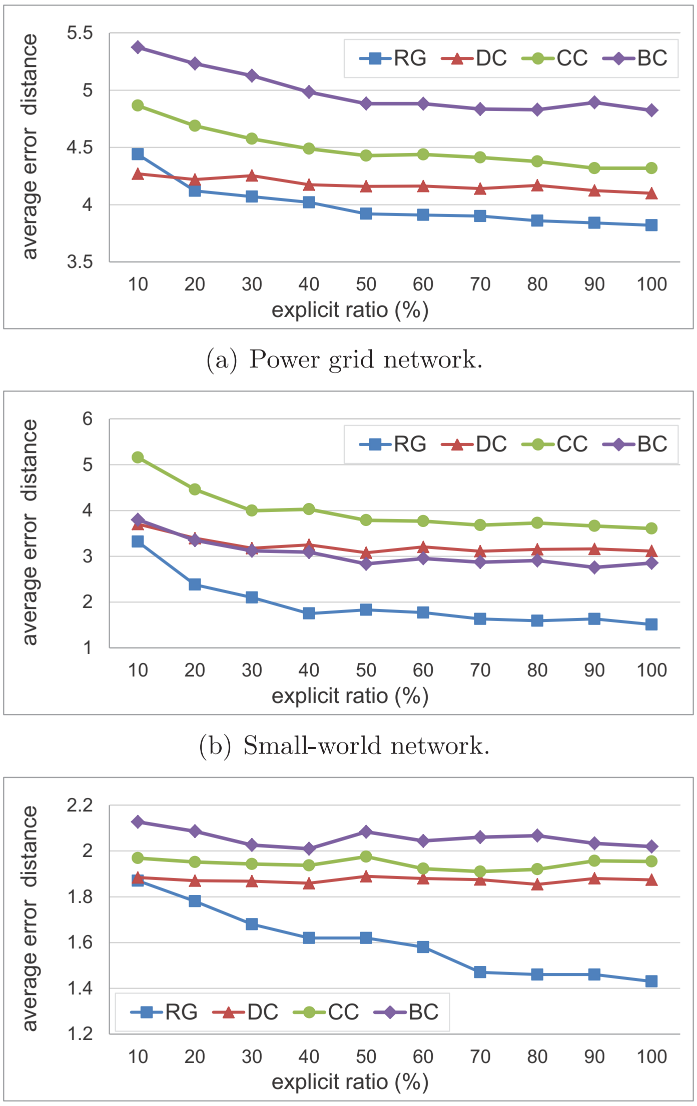
Proof of Lemma 4.1

The proof proceeds by mathematical induction on the elapsed time t.

Basis step: Suppose that $t = 1$, and consider any non-observable subtree $T_u(v; G)$. Suppose that $d(v, u) > 1$. Since the infection can spread at most one hop away from v in each time slot, the node u must remain uninfected up to time $t = 1$ and the claim holds trivially. We now consider only the neighbors of v. Let u be a neighbor of v that is the root of a non-observable subtree, and suppose that every most likely infection path $\tilde{X}^1$ for (v, t) has $\tilde{X}(u, 1) = \mathbf{i}$. Choose one such most likely infection path $\tilde{X}^1$, and let X^1 be another infection path with the same states as $\tilde{X}^1$, except that $X(u, 1) = \mathbf{s}$. Then, we have

$$\frac{P_v(X^1)}{P_v(\tilde{X}^1)} = \frac{1 - p_{\mathbf{s}}}{p_{\mathbf{s}}(1 - q_u)} \geq 1,$$

where the last inequality follows from the assumption (4.1). This is a contradiction,



(a) Power grid network.

(b) Small-world network.

(c) Facebook network.

Figure 4-5: Average error distances for different explicit ratios in different general networks. The average diameters of the power grid network, small-world network, and Facebook network are 46, 20, and 18 hops, respectively.

and therefore the claim holds if $t = 1$.

Inductive step: Suppose that the claim holds for all elapsed times $t \leq n$. Let $t = n+1$, and consider any non-observable subtree $T_u(v; G)$ and a most likely infection path $\tilde{X}^t$ for (v, t) with $\tilde{X}(u, t) = \mathbf{i}$.

If $d(v, u) > 1$, let $\tilde{t}$ denote the first infection time of the parent node $r = \text{pa}(u)$ in $\tilde{X}^t$, where $\tilde{t} \geq 1$. Since the infection process follows the SI model, we can treat r as the infection source of $T_r(v; G)$, and the remaining elapsed time of the infection process is $t' = t - \tilde{t} \leq n$. From the induction hypothesis, we can construct a most likely infection path $X^{t'}(T_r(v;G),[1,t'])$ such that u remains uninfected up to time t. The new infection path constructed from X^t by setting $\tilde{X}^t(T_r(v;G),[\tilde{t}+1,t]) = X^{t'}(T_r(v;G),[1,t'])$ then has probability at least that of $\tilde{X}^t$, thus proving our claim.

Now suppose that $d(v,u) = 1$ and the first infection time of node u in the path $\tilde{X}^t$ is $t_u \in [1,t]$. The previous argument shows that it is possible to choose $\tilde{X}^t$ so that w remains uninfected up to time t for all $w \in T_u(v;G) \backslash \{u\}$ since $d(v,w) > 1$. We now suppose $\tilde{X}^t$ is chosen as such. Let X^t be an infection path that is the same as the most likely infection path $\tilde{X}^t$ but with $X^t(u,\tau) = \mathbf{s}$ for all $\tau \leq t$. Then, we have

$$\begin{aligned} P_v(X^t) &= a \prod_{\tau=1}^{t} P_v(X^t(u,\tau)) \\ &= a(1-p_{\mathbf{s}})^t, \end{aligned} \tag{4.19}$$

where $a = P_v(X^t(V \backslash T_u(v;G),[1,t]))$. Similarly, we have

$$\begin{aligned} P_v(\tilde{X}^t) =& a \prod_{\tau=1}^{t_u-1} (P_v(\tilde{X}^t(u,\tau)) \cdot P_v(\tilde{X}^t(u,\tau))) \\ &\cdot \prod_{w\in \mathrm{ch}(u)} \prod_{\tau=t_u+1}^{t} P_v(\tilde{X}^t(w,\tau)) \\ =& a(1-p_{\mathbf{s}})^{t_u-1} p_{\mathbf{s}}(1-q_u)(1-p_{\mathbf{s}})^{(t-t_u)|\mathrm{ch}(u)|} \\ \leq& a(1-p_{\mathbf{s}})^{t-1} p_{\mathbf{s}}(1-q_u), \end{aligned} \tag{4.20}$$

where the last inequality follows because we have assumed that $|\mathrm{ch}(u)| \geq 1$. Comparing (4.19) and (4.20), and using (4.1), we obtain $P_v(X^t) \geq P_v(\tilde{X}^t)$, which implies that X^t is also a most likely infection path. Repeating the same argument for all non-observable subtrees with roots that are neighbors of v, we obtain the claim, and the lemma is proved.

Proof of Lemma 4.2

Firstly, it is easy to see that every node in H_v is infected since otherwise, the infection can not reach the leaf nodes of H_v, some of which belong to the set $V_{\mathbf{i}}$. The lower bound in (4.2) follows because the infection can spread at most one hop away from v in each time slot, and the earliest time for u to be infected is $d(v, u)$. After node u gets infected at time t_u, the infection can spread at most $t - t_u$ hops away from u. Consider a node u_l such that $d(u, u_l) = \bar{d}(u, T_u(v; H_v))$. By definition, $u_l \in V_{\mathbf{i}}$. In order for the infection to reach node u_l, we require $t - t_u \geq d(u, u_l)$, and (4.2) is shown.

Next, we show (4.3) using mathematical induction on $\bar{d}(v, V_{\mathbf{i}})$. Note that we do not need to consider the case where $\bar{d}(v, V_{\mathbf{i}}) = 0$, i.e., no non-source nodes exist in H_v.

Basis step: Suppose that $\bar{d}(v, V_{\mathbf{i}}) = 1$. Then we have $d(v, u) = 1$ for every non-source node u in H_v. We want to show that there exists a most likely infection path so that the first infection time for u is $t_u = t$. If $t = 1$, we have $t_u = 1$ because both the lower and upper bounds in (4.2) are equal to 1. We now suppose that $t > 1$, and we have $t_u \in [1, t]$ by (4.2). Let $\tilde{X}^t$ denote a most likely infection path with $t_u = i$, where $1 \leq i \leq t - 1$. We construct another infection path X^t from $\tilde{X}^t$ so that $X^t(V\backslash\{u\}, [1, t]) = \tilde{X}^t(V\backslash\{u\}, [1, t])$ and node u becomes infected only at time t. Let $a = P_v(X^t(V\backslash\{u\}, [1, t]))$. We then have

$$\begin{aligned} P_v(X^t) &= aP_v(X^t(T_u(v; G), [1, t])) \\ &= (1 - p_{\mathbf{s}})^{t-1} p_{\mathbf{s}} q_u. \end{aligned} \tag{4.21}$$

The same derivation as in (4.20), except that here $\tilde{X}^t(u, t)$ is explicit instead of non-observable, gives

$$\begin{aligned} P_v(\tilde{X}^t) =& a(1 - p_{\mathbf{s}})^{i-1} p_{\mathbf{s}} q_u (1 - p_{\mathbf{s}})^{(t-i)|\text{ch}(u)|} \\ \leq& a(1 - p_{\mathbf{s}})^{t-1} p_{\mathbf{s}} q_u, \end{aligned} \tag{4.22}$$

where we use the assumption that $|\text{ch}(u)| \geq 1$ in the last inequality. By comparing (4.21) and (4.22), we obtain $P_v(X^t) \geq P_v(\tilde{X}^t)$, and by repeating the same argument for all neighbors of v in $V_{\mathbf{i}}$, we obtain the claim for the basis step.

Inductive step: We assume that (4.3) holds when $\bar{d}(v, V_{\mathbf{i}}) \leq n$. Fix any $u \in \mathrm{ch}(v)$ and let $m = \bar{d}(u, T_u(v; H_v))$. Treat node u as the infection source of the subtree $T_u(v; H_v)$, then by the induction hypothesis, we obtain a most likely infection path $\tilde{X}^t$ with (4.3) holding for any node $w \in T_u(v; H_v) \backslash \{u\}$ since $m \leq \bar{d}(v, V_{\mathbf{i}}) - 1 = n$.

For any node $w \in N_1(u; T_u(v; H_v))$, we have the first infection time of w is $t - (m-1)$ (note that $T_u(v; H_v)$ is not a non-observable subtree), which implies that $t_u \leq t - m$. We want to show that $\hat{t}_u = t - m$. Suppose that we have $t_u = i$, for some $1 \leq i \leq t - m - 1$ in $\tilde{X}^t$. Then, using the same arguments as in the basis step, we can construct another infection path with probability at least that of $\tilde{X}^t$ but with $t_u = t - m$. This implies our claim, and the lemma is proved.

Chapter 5

Single Infection Source Identification in the SI, SIR, SIRI and SIS Models under the MLIP Criterion

In Chapter 3 and Chapter 4, we focused on the SI model. However, the SIRI and SIS infection spreading models have many applications in practice (cf. Section 1.3.3) and, to the best of our knowledge, identifying infection sources under these models have not been investigated. Moreover, the knowledge of the underlying infection spreading model is assumed to be known in all the existing works, which may be difficult to obtain in practice (cf. Section 1.3.3). In this chapter, we consider the problem of identifying a single infection source in the SI, SIR, SIRI and SIS models (cf. Section 2.1) under the MLIP criterion (cf. Section 2.3).

5.1 Problem Formulation

In this chapter, we consider the discrete time SI, SIR, SIRI and SIS models discussed in Section 2.1. We assume that there is a single infection source. We use the notations and definitions introduced in Section 4.1.1. We consider the MLIP criterion as discussed in Section 2.3 and want to find the solution of (2.1).

For any node $v \in V$, we let $p_{\mathbf{s}}(v)$, $p_{\mathbf{i}}(v)$ and $p_{\mathbf{r}}(v)$ be the probability for v to be in state $\mathbf{i}$ in the next time slot conditioned on v being susceptible, infected, or recovered in the current time slot, respectively. The value of these probabilities characterize different infection spreading models, and their value ranges are summarized in Assumption 5.1-5.4. Let $\alpha = \min_{u\in V} p_{\mathbf{s}}(u)$ and $\beta = \max_{u\in V} p_{\mathbf{s}}(u)$.

Assumption 5.1. Under the SI model, for every $v \in V$, we have

$$\beta \leq \frac{\alpha}{(1-\alpha)^4}. \tag{5.1}$$

In the inequality (5.1), we assume that the infection probabilities at each node in the network does not differ drastically for the SI model. This is required because in this work, we do not assume knowledge of the exact infection rates at each node. Therefore, if part of the network has nodes that are much easier to infect than other nodes, then any estimator with no knowledge of the infection rates will result in a highly biased result, which may not do better on average than making random choices for the infection sources.

Assumption 5.2. Under the SIR model, for every $v \in V$, we have

$$0 \leq p_{\mathbf{i}}(v) \leq \sqrt{\frac{\alpha}{\beta}}. \tag{5.2}$$

Assumption 5.3. Under the SIRI model, for every $v \in V$, we have

$$\frac{\beta-\alpha}{1-\alpha} \leq p_{\mathbf{i}}(v) \leq \sqrt{\frac{\alpha}{\beta}}, \tag{5.3}$$

$$1-\sqrt{\frac{\alpha}{\beta}} \leq p_{\mathbf{r}}(v) \leq \min\left\{1, \sqrt{\frac{\alpha}{\beta}}\frac{p_{\mathbf{i}}(v)}{1-p_{\mathbf{i}}(v)}\right\}. \tag{5.4}$$

Note that if $\alpha = \beta$, (5.2) and (5.3) both reduce to $0 \leq p_{\mathbf{i}}(v) \leq 1$, and (5.4) reduces to $0 \leq p_{\mathbf{r}}(v) \leq \min\left\{1, \frac{p_{\mathbf{i}}(v)}{1-p_{\mathbf{i}}(v)}\right\}$. Inequality (5.4) implies that a node does not easily relapse into an infected state (i.e., small $p_{\mathbf{r}}$) if it recovers quickly (i.e., small $p_{\mathbf{i}}$). This is intuitively appealing as it corresponds to the case where if an infected node has a low probability of staying infected in the next time slot, then it is unlikely for the node to relapse into the infection once it has recovered. A practical example is it is hard to re-convince someone to believe a rumor if he already has a reason to reject the rumor.

Assumption 5.4. Under the SIS model, for every $v \in V$, we have

$$\begin{aligned} p_{\mathbf{s}}(v) &= p_{\mathbf{s}}, \\ p_{\mathbf{i}}(v) &= p_{\mathbf{i}}, \\ 0 \leq p_{\mathbf{s}} &\leq p_{\mathbf{i}} \leq 1. \end{aligned} \tag{5.5}$$

Inequality (5.5) helps us to avoid the case where an infection spreads very fast (i.e., large $p_{\mathbf{s}}$) and infected nodes also recover relatively quickly (i.e., $p_{\mathbf{i}} < p_{\mathbf{s}}$) from happening. In such cases, infected nodes close to sources are likely to have recovered by the time we observe the state of the network, while there may be a significant set of infected nodes at a distance away from the source. Therefore, trying to estimate the source nodes will result in a large bias.

In Assumptions 5.1-5.3 for the SI, SIR and SIRI models, the infection, recovery, and relapse probabilities can vary between different nodes. We call such networks *heterogeneous*. On the other hand, in Assumption 5.4, the infection and recovery probabilities are the same for all nodes in the network. We call such networks *homogeneous*.

As noted in Section 1.3.3, proving optimality results for infection source estimators is in general challenging. In this chapter, we restrict ourselves to the following specific graph networks depending on the infection spreading model. We say that a tree is an *infinite tree* if every node in it has degree at least two.

Assumption 5.5. For an infection spreading according to the SI, SIR or SIRI models, the underlying graph G is an infinite tree. For an infection spreading according to the SIS model, the underlying graph G is a regular infinite tree, i.e., every node has the same degree.

For the SI, SIR, and SIRI models, Assumption 5.5 is adopted to avoid boundary effects. Consider the extreme case where a source node has only one neighbor. Then, the infection can spread away from the source in only one direction. In this case, any estimator based only on the graph topology is expected to perform badly. In the SIS model, a recovered node is the same as a susceptible node, which leads to more complex evolution of the node states in the network as compared to the SIRI model in which the state evolution of a recovered node becomes independent from the rest of the network. To simplify the problem, we restrict to regular trees for the SIS model in Assumption 5.5. The problem of finding optimal source estimators for the SIS model in more general network topologies remains open.

5.2 Source Estimation for Trees

In this section, we study the source estimation problem when the underlying graph G is a tree network. We show that a Jordan center of the infected node set $V_{\mathtt{i}}$ is an optimal infection source estimator universally applicable for infection spreading under the SI, SIR, and SIRI models for trees, and the SIS model for regular trees. The Jordan center has previously been shown to be optimal estimators for the SIR model [37].

5.2.1 Most Likely Elapsed Time

We assume no knowledge of the elapsed time when the set of nodes $V_{\mathtt{i}}$ is observed. Suppose that $v \in V$ is the source, then the feasible set of all elapsed times is given by $\mathcal{T}_v = [\bar{d}(v, V_{\mathtt{i}}), +\infty)$, where the lower bound is the minimum amount of time required for the infection to spread from v to all the nodes in $V_{\mathtt{i}}$. It is obviously computationally inefficient to search over all elapsed times. In Proposition 5.1, we show how to find a *most likely elapsed time* t_v that maximizes the probability of observing $V_{\mathtt{i}}$.

Proposition 5.1. Suppose that Assumptions 5.1-5.4 hold, $v \in V$ is the infection source, and a non-empty set of infected nodes $V_{\mathtt{i}}$ is observed. For an infection under the SI, SIR, SIRI or SIS model in a network satisfying Assumption 5.5, we have for any $t \in \mathcal{T}_v$, and any two most likely infection paths X^t for (v, t) and Y^{t+1} for $(v, t+1)$,

(a) $P_v(Y^{t+1}) \leq \delta P_v(X^t)$, where $\delta = (1-\alpha)^2, \sqrt{\frac{\alpha}{\beta}}, \sqrt{\frac{\alpha}{\beta}}$ and 1 for the SI, SIR, SIRI and SIS model, respectively; and

(b) conditioned on v being the infection source, a most likely elapsed time is given by

$$t_v = \bar{d}(v, V_{\mathbf{i}}).$$

The proof of Proposition 5.1 is provided in Section 5.5. Proposition 5.1(b) shows a universal property that is robust to the underlying infection spreading models: a most likely elapsed time t_v is the infection range of v (cf. Definition 4.2). Moreover, Proposition 5.1(a) shows that a most likely elapsed time should be as small as possible. This result is intuitive. Consider the conditional probability

$$P_v(X^t) = \prod_{u \in V, \tau \in [1,t]} P_v(u, \tau),$$

where the value of each term in the product on the right hand side is at most 1. When t decreases, there are less terms in the product, which in turn increases the value of $P_v(X^t)$.

Following Proposition 5.1, the problem in (2.1) is now reduced to

$$\hat{s} \in \arg \max_{\substack{v \in V, t_v = \bar{d}(v, V_{\mathbf{i}}) \\ X^{t_v} \in \mathcal{X}_v}} P_v(X^{t_v}).$$

After the most likely elapsed time has been identified, we can now proceed to find the source node associated with the most likely infection path.

5.2.2 Source Associated with the Most Likely Infection Path

In this subsection, we derive the source estimator associated with a most likely infection path for all four considered infection spreading models, under specific graph networks. Although Proposition 5.1 gives a most likely elapsed time t_v conditioned on a node $v \in V$ being the infection source, it is still difficult to count the number of infection paths that are consistent with V_{i}, not to mention finding the most likely infection path for (v, t_v). Therefore, instead of directly looking for the most likely infection path, we first consider the conditional probabilities $P_v(X^{t_v})$ and $P_u(Y^{t_u})$ of two infection paths, where v and u are a pair of neighboring nodes, X^{t_v} is a most likely infection path for (v, t_v), and Y^{t_u} is a most likely infection path for (u, t_u). We then show that if v has a smaller infection range, $P_v(X^{t_v})$ is not less than $P_u(Y^{t_u})$. Upon establishing this neighboring node relationship, we can find a path on which the infection range of each node is decreasing, and the conditional probability of the most likely infection path is non-decreasing. This in turn implies that the Jordan center of V_{i} is the source estimator we are looking for. The neighboring node relationship is summarized in Proposition 5.2, the proof of which is provided in Section 5.5.

Proposition 5.2. Suppose that V_{i} is non-empty. For an infection process under the SI, SIR, SIRI or SIS model satisfying Assumptions 5.1-5.5, and for any pair of neighboring nodes u and v, we have

$$P_v(X^{t_v}) \geq P_u(Y^{t_u}), \text{ if } t_v < t_u,$$

where X^{t_v} and Y^{t_u} are most likely infection paths for (v, t_v) and (u, t_u) respectively.

We note that Proposition 5.1 and Proposition 5.2 match Proposition 4.2 and Lemma 4.4, respectively. Then following similar proof of Theorem 4.1, we have the following result.

Theorem 5.1. Suppose that V_{i} is non-empty. For an infection process under the SI, SIR, SIRI or SIS model satisfying Assumptions 5.1-5.5, a Jordan center of V_{i} is an optimal source estimator for (2.1).

Theorem 5.1 shows that for regular infinite trees, a Jordan center is an optimal source estimator, regardless of which of the four considered infection spreading model the infection is following. This is a somewhat surprising result since the four infection spreading models are fundamentally different. The "universality" of the Jordan center makes it highly desirable in practice, where the underlying infection spreading model is usually unknown a priori. We propose a distributed linear time complexity algorithm (cf. Algorithm 4.1) to find the Jordan center in a tree, which makes timely estimation of the infection source possible.

5.3 Source Estimation for General Graphs

In this section, we consider the case where the underlying network G is a general graph. Inspired by the robustness of the Jordan center estimator in tree networks, we heuristically extend it to general graphs.

A simple algorithm was proposed in [37] to find the Jordan center of $V_{\mathbf{i}}$ when there is a single source and the underlying network is a general graph. We briefly review this algorithm in the following. Let any node in $V_{\mathbf{i}}$ broadcast a message containing its own identity. The first node that receives messages from every node in $V_{\mathbf{i}}$ declare itself as a Jordan center and the algorithm terminates. We call this algorithm the Single Jordan Center estimation (SJC) algorithm, with a time complexity of $O(|V||E|)$.

In Section 5.4, we perform extensive simulations on both synthetic and real world networks to verify the performance of the proposed estimators.

5.4 Simulation Results

In this section, we present simulation results using both synthetic and real world networks to evaluate the performance of the proposed estimators.

We use the same centrality measures used in Section 4.4 as benchmarks to compare with our estimator. We evaluate the performance of the proposed estimator on three kinds of networks: random tree networks where the degree of every node is randomly chosen from $[3, 5]$, a small part of the Facebook network with 4039 nodes [69], which is shown in [70] to be a scale-free network, and the western states power grid network of the United States [63]. We consider both homogeneous and heterogeneous networks. In the homogeneous networks, we vary the recovery and relapse probabilities to demonstrate the impact of these spreading parameters on the performance of the proposed estimator. In the heterogeneous networks, we evaluate the robustness of the proposed estimator on a wide range of randomly generated spreading parameters. In the following, we describe the four different simulation experiments.

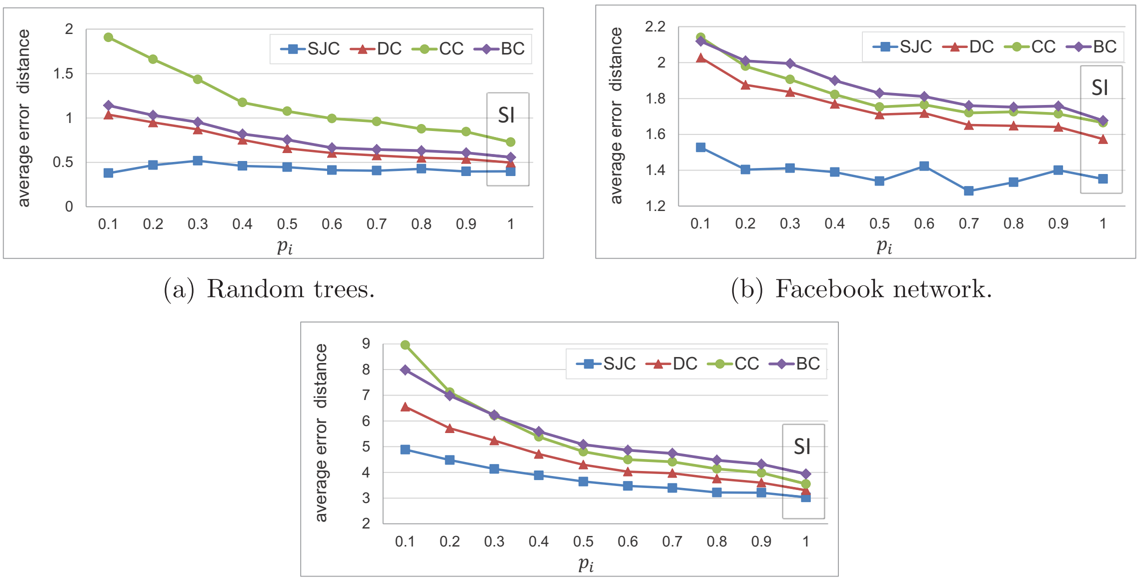
SI and SIRI models in homogeneous networks

For every $v \in V$, we let $p_{\mathbf{s}}(v) = p_{\mathbf{s}}$, $p_{\mathbf{i}}(v) = p_{\mathbf{i}}$ and $p_{\mathbf{r}}(v) = p_{\mathbf{r}}$, where the infection probabilities are set as follows: $p_{\mathbf{s}}$ is randomly chosen from $[0, 1]$, $p_{\mathbf{i}}$ is set to be $0.1, 0.2, \cdots, 1$, respectively, and $p_{\mathbf{r}}$ is randomly chosen from $[0, \min\{1, \frac{p_{\mathbf{i}}}{1-p_{\mathbf{i}}}\}]$. For each kind of network and each value of $p_{\mathbf{i}}$, we perform 1000 simulation runs. In each simulation run, we randomly pick a node as the infection source and simulate the infection using the above parameters. The spreading terminates when the number of infected nodes is greater than 100. We then run SJC on the observed infected nodes to estimate the infection source and compare the result with the benchmarks.

The error distance is the number of hops between the estimated and the actual infection source, and is shown in Fig. 5-1. We see that the proposed estimator performs consistently better than the benchmarks for all considered networks.

SIR and SIRI models in homogeneous networks

The infection probabilities are set as follows: $p_{\mathbf{s}}$ is randomly chosen from $[0, 1]$, $p_{\mathbf{i}}$ is randomly chosen from $[0.5, 1]$, and $p_{\mathbf{r}}$ is set to be $0, 0.1, \cdots, 1$. We compare the performances in Fig. 5-2. We see that our proposed estimator again performs consistently better than the benchmarks.

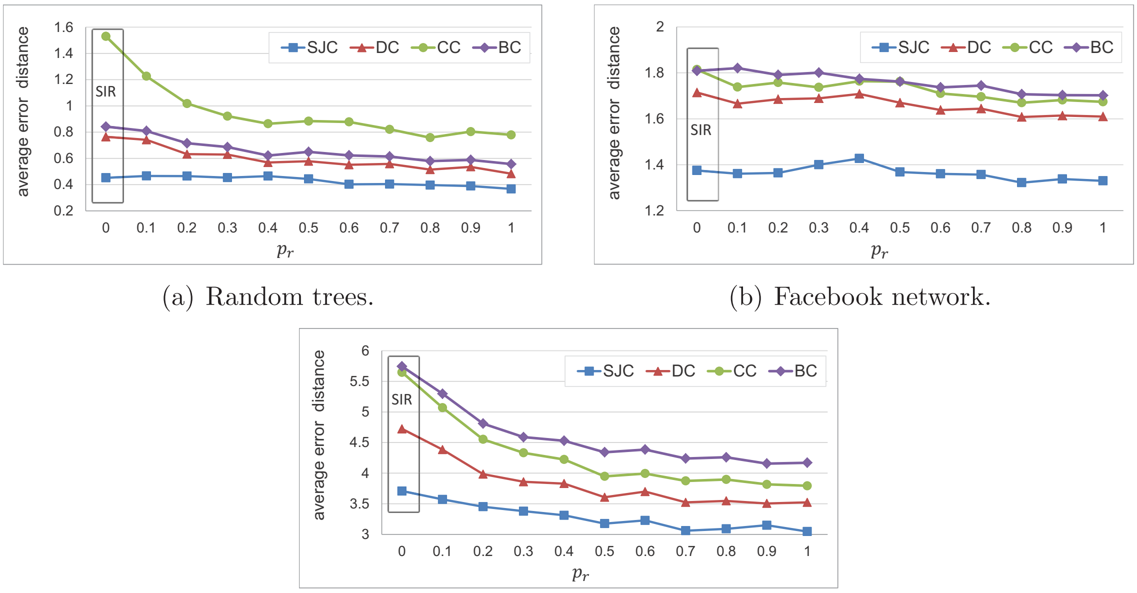


(a) Random trees.

(b) Facebook network.

(c) Power grid network.

Figure 5-1: Average error distances for various networks and different values of $p_\mathbf{i}$ in homogeneous networks. The underlying infection follows the SIRI model for all values of $p_\mathbf{i}$ and the infection follows the SI model when $p_\mathbf{i} = 1$.



(a) Random trees.

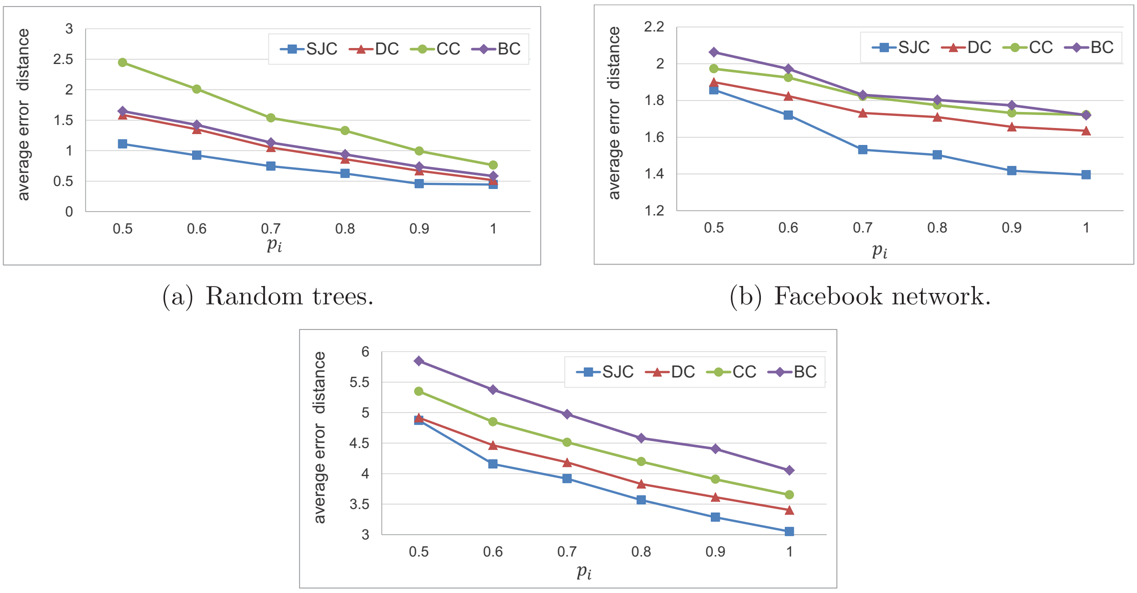
(b) Facebook network.

(c) Power grid network.

Figure 5-2: Average error distances for various networks and different values of $p_\mathbf{r}$ in homogeneous networks. The underlying infection follows the SIRI model for all values of $p_\mathbf{r}$ and the infection follows the SIR model when $p_\mathbf{r} = 0$.

SIS model in homogeneous networks

We consider the SIS model where $p_\mathbf{i}$ is set to be $0.5, 0.6, \cdots, 1$, respectively, and $p_\mathbf{s}$ is randomly chosen from $[0, p_\mathbf{i}]$. In Fig. 5-3, we observe that our proposed estimator always results in smaller average error distances than the benchmarks for all considered networks.



(a) Random trees.

(b) Facebook network.

(c) Power grid network.

Figure 5-3: Average error distances for various networks and different values of $p_{\mathbf{i}}$ in homogeneous networks under the SIS model.

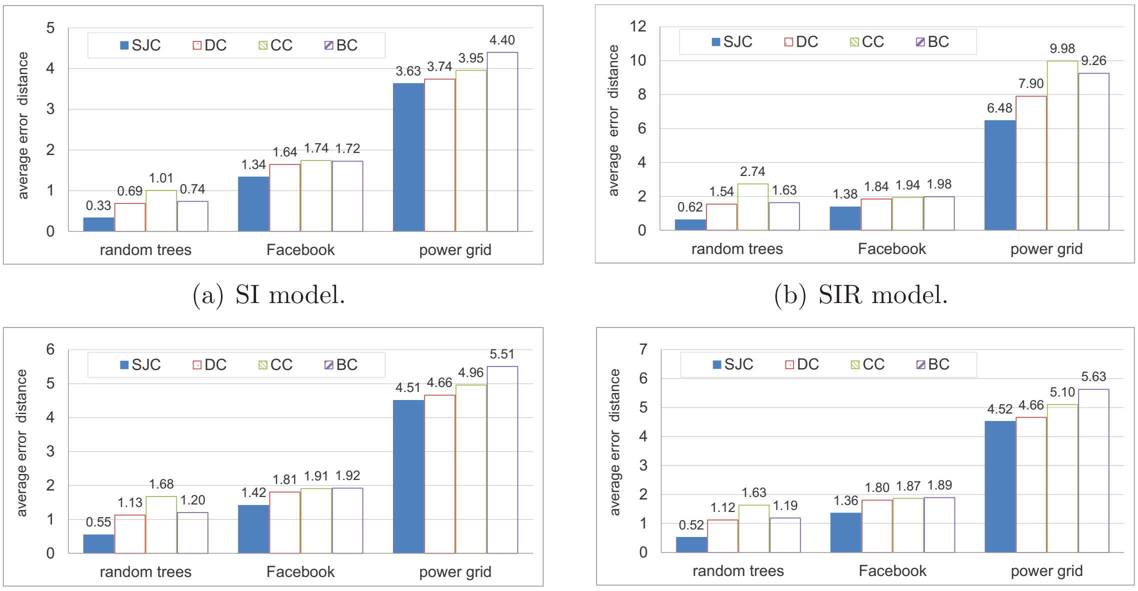
SI, SIR, SIRI, and SIS models in heterogeneous networks

In this experiment, we drop the Assumptions 5.1-5.4 and randomly choose the infection probabilities $p_{\mathbf{s}}(v)$, $p_{\mathbf{i}}(v)$, $p_{\mathbf{r}}(v)$ from $[0, 1]$ for any node v. We then run simulations under the SI, SIR, SIRI, and SIS models, and compare the performances in Fig. 5-4. We see that SJC outperforms all the benchmarks.

5.5 Proofs

Proof of Proposition 5.1

For any $t \in \mathcal{T}_v$, consider any most likely infection path Y^{t+1} for $(v, t+1)$. To show



(a) SI model.

(b) SIR model.

(c) SIRI model.

(d) SIS model.

Figure 5-4: Average error distances for various networks under the SI, SIR, SIRI and SIS models in heterogeneous networks.

claim (a), it suffices to construct an infection path $\tilde{X}^t$ for (v, t) such that

$$P_v(Y^{t+1}) \leq \delta P_v(\tilde{X}^t), \tag{5.6}$$

since $P_v(\tilde{X}^t) \leq P_v(X^t)$.

- SI model

We first focus on any neighboring node u of v and consider $T_u(v; G)$. We claim that there exists an infection path $\tilde{X}^t$ such that

$$P_v(Y^{t+1}(T_u(v; G), [1, t+1])) \leq (1 - \alpha) P_v(\tilde{X}^t(T_u(v; G), [1, t])). \tag{5.7}$$

We can see that $T_u(v; G)$ is either an uninfected subtree or infected subtree. In the following, we consider these two cases in order.

Suppose that $T_u(v;G)$ is an uninfected subtree. We have for any $\tilde{X}^t$

$$\begin{aligned}\frac{P_v(Y^{t+1}(T_u(v;G),[1,t+1]))}{P_v(\tilde{X}^t(T_u(v;G),[1,t]))} &= \frac{(1-p_{\mathbf{s}}(u))^{t+1}}{(1-p_{\mathbf{s}}(u))^t} \\ &\leq 1-\alpha. \end{aligned} \tag{5.8}$$

Suppose that $T_u(v;G)$ is an infected subtree.

If $Y^{t+1}(u,1)=\mathbf{s}$, we let $\tilde{X}^t(T_u(v;G),[1,t]) = Y^{t+1}(T_u(v;G),[2,t+1])$, yielding

$$\begin{aligned}\frac{P_v(Y^{t+1}(T_u(v;G),[1,t+1]))}{P_v(\tilde{X}^t(T_u(v;G),[1,t]))} &= \frac{P_v(Y^{t+1}(u,1)=\mathbf{s})P_v(Y^{t+1}(T_u(v;G),[2,t+1]))}{P_v(\tilde{X}^t(T_u(v;G),[1,t]))} \\ &= 1-p_{\mathbf{s}}(u) \\ &\leq 1-\alpha. \end{aligned}$$

If $Y^{t+1}(u,1)=\mathbf{i}$, we show (5.7) by mathematical induction on $\bar{d}(v,V_{\mathbf{i}})$.

Basis step: Suppose that $\bar{d}(v,V_{\mathbf{i}})=1$.

We let $\tilde{X}^t(u,1)=\mathbf{i}$. After it gets infected at time slot 1, node u serves as the infection source of the subtree $T_u(v;G)$ with the infection starting at time 1. From the assumption $\bar{d}(v,V_{\mathbf{i}})=1$, it follows that $T_w(u;G)$ is an uninfected subtree for any $w\in V(u,1)\bigcap T_u(v;G)$. Then following (5.8), we have

$$\begin{aligned}\frac{P_v(Y^{t+1}(T_u(v;G),[1,t+1]))}{P_v(\tilde{X}^t(T_u(v;G),[1,t]))} &= \frac{P_v(Y^{t+1}(u,1)=\mathbf{i})P_v(Y^{t+1}(T_u(v;G),[2,t+1]))}{P_v(\tilde{X}^{t+1}(u,1)=\mathbf{i})P_v(\tilde{X}^t(T_u(v;G),[2,t]))} \\ &= \prod_{w\in V(u,1)\bigcap T_u(u;G)} \frac{P_v(Y^{t+1}(T_w(u;G),[2,t+1]))}{P_v(\tilde{X}^t(T_w(v;G),[2,t]))} \\ &\leq (1-\alpha)^{|V(u,1)\bigcap T_u(v;G)|} \\ &\leq 1-\alpha, \end{aligned}$$

where the last inequality follows from Assumption 5.5. This completes the proof for the basis step.

Inductive step: Assume (5.7) holds for $\bar{d}(v,V_{\mathbf{i}})\leq n-1$, where $n\geq 2$. We want to show that (5.7) also holds for $\bar{d}(v,V_{\mathbf{i}})=n$.

Assume $\bar{d}(v, V_{\mathbf{i}}) = n$ and let $\tilde{X}^t(u, 1) = \mathbf{i}$. After it becomes infected at time slot 1, node u serves as the infection source of the subtree $T_u(v; G)$ with the infection starting at time 1. Since $\bar{d}(u, V_{\mathbf{i}} \bigcap T_u(v; G)) \leq n - 1$, from the induction assumption and for any $w \in V(u, 1) \bigcap T_u(v; G)$, we can find a $\tilde{X}^t$ such that

$$P_v(Y^{t+1}(T_w(u; G), [2, t+1])) \leq (1 - \alpha) P_v(\tilde{X}^t(T_w(u; G), [2, t])).$$

We then have,

$$\begin{aligned}
\frac{P_v(Y^{t+1}(T_u(v; G), [1, t+1]))}{P_v(\tilde{X}^t(T_u(v; G), [1, t]))} &= \frac{P_v(Y^{t+1}(u, 1) = \mathbf{i}) P_v(Y^{t+1}(T_u(v; G), [2, t+1]))}{P_v(\tilde{X}^{t+1}(u, 1) = \mathbf{i}) P_v(\tilde{X}^t(T_u(v; G), [2, t]))} \\
&= \prod_{w \in V(u,1) \bigcap T_u(u;G)} \frac{P_v(Y^{t+1}(T_w(u; G), [2, t+1]))}{P_v(\tilde{X}^t(T_w(v; G), [2, t]))} \\
&\leq (1 - \alpha)^{|V(u,1) \bigcap T_u(v;G)|} \\
&\leq 1 - \alpha,
\end{aligned}$$

where the last inequality follows from Assumption 5.5. This completes the proof for the inductive step, and the claim is now proved.

By constructing $\tilde{X}^t$ to satisfy (5.7) for all $u \in V(v, 1)$, we have

$$\begin{aligned}
\frac{P_v(Y^{t+1})}{P_v(\tilde{X}^t)} &= \prod_{u \in V(v,1)} \frac{P_v(Y^{t+1}(T_u(v; G), [1, t+1]))}{P_v(\tilde{X}^t(T_u(v; G), [1, t]))} \\
&\leq (1 - \alpha)^{|V(v,1)|} \\
&\leq (1 - \alpha)^2,
\end{aligned}$$

where the last inequality follows from Assumption 5.5. This completes the proof of claim(a) for the SI model.

- SIR and SIRI models

We first present a property of the SIRI model in Lemma 5.1.

Lemma 5.1. Suppose that $v \in V$ is the infection source and v has only one neighboring node u. Suppose that the set of observed infected nodes $V_{\mathbf{i}}$ is non-empty.

Consider an infection under the SIRI model and suppose Assumptions 5.3 and 5.5 hold. For any $t \in \mathcal{T}_v$ and any most likely infection path Y^{t+1} for $(v, t+1)$, there exists an infection path $\tilde{X}^t$, such that

(a) $P_v(Y^{t+1}(v, [1, t+1])) \leq \sqrt{\frac{\alpha}{\beta}} P_v(\tilde{X}^t(v, [1, t]))$;

(b) $P_v(Y^{t+1}(T_u(v; G), [1, t+1])) \leq P_v(\tilde{X}(T_u(v; G), [1, t]))$; and

(c) $P_v(Y^{t+1}) \leq \sqrt{\frac{\alpha}{\beta}} P_v(\tilde{X}^t)$.

The proof of Lemma 5.1 is provided in Section 5.5. Lemma 5.1 shows that, in the SIRI model, a most likely elapsed time t_v should be as small as possible when the source has only one neighboring node. We now extend this result to prove Proposition 5.1(a) for the SIRI model where v has more than one neighboring node.

In the SIRI model, since v is the source node, $\tilde{X}^t(v, [1, t])$ is independent of the states of other nodes. Furthermore, for any pair of neighboring nodes u and u' of v, the states of $T_u(v; G)$ and $T_{u'}(v; G)$ are independent conditioned on the states of node v. Therefore, by applying Lemma 5.1 to v and each of its neighboring nodes, we have an infection path $\tilde{X}^t$ such that

$$\begin{aligned}\frac{P_v(Y^{t+1})}{P_v(\tilde{X}^t)} =& \frac{P_v(Y^{t+1}(v, [1, t+1])) \prod_{u \in N_v(1)} P_v(Y^{t+1}(T_u(v; G), [1, t+1]))}{P_v(\tilde{X}^t(v, [1, t])) \prod_{u \in N_v(1)} P_v(\tilde{X}^t(T_u(v; G), [1, t]))} \\ \leq& \sqrt{\frac{\alpha}{\beta}}.\end{aligned}$$

This completes the proof of claim (a) for the SIRI model. The proof of claim (a) for the SIR model is similar to that of the SIRI model, and we omit it here to avoid repetition.

- SIS model

For the SIS model, a node can become infected, recover, and then be reinfected again for multiple times by the observation time. We characterize the time when a node is first infected (first infection time) in the following lemma, whose proof is provided in Section 5.5. Recall that H_v is the minimum connected subgraph of G that contains V_i and v.

Lemma 5.2. Suppose that $v \in V$ is the infection source and a non-empty set of infected nodes $V_{\mathbf{i}}$ is observed. Suppose the infection follows the SIS model and Assumption 5.4 and 5.5 hold. Then, for any $t \in \mathcal{T}_v$, there exists a most likely infection path X^t for (v, t), such that, for any $u \in H_v \backslash \{v\}$, the first infection time $t_{int}(u)$ of u in X^t is given by

$$t_{int}(u) = t - \bar{d}(u, T_u(v; H_v)). \tag{5.9}$$

Lemma 5.2 enables us to calculate the first infection time of each node in H_v in a most likely infection path under the SIS model. Moreover, it shows that given the elapsed time, a most likely infection path for a node v is given by a path whose nodes "resist" the infection, and each node becomes infected only at the latest possible time. Therefore, intuitively the most likely elapsed time t_v should be as small as possible to minimize the time that nodes "resist" the infection spreading, so as to maximize the probability of the infection path.

Since $t \in \mathcal{T}_v$, we have $t \geq \bar{d}(v, V_{\mathbf{i}})$. For any $u \in V(v, 1)$, from Lemma 5.2, we have that the first infection time $t_{int}(u)$ of u in Y^{t+1} is given by

$$\begin{aligned} t_{int}(u) &= t + 1 - \bar{d}(u, T_u(v; H_v)) \\ &\geq \bar{d}(v, V_{\mathbf{i}}) + 1 - \bar{d}(u, T_u(v; H_v)) \\ &\geq 2. \end{aligned} \tag{5.10}$$

We claim that $Y^{t+1}(v, 1) = \mathbf{i}$. Otherwise, v and all its neighboring nodes are not infected at time 1 because of (5.10). Because the infection can propagate at most 1 hop away from v at time 1, all nodes are uninfected at time 1, and the infection propagation process stops. This contradicts the assumption that the set of observed infected nodes $V_{\mathbf{i}}$ is non-empty. Then, following Lemma 5.2, we can let

$\tilde{X}^t(V,[1,t]) = Y^{t+1}(V,[2,t+1])$, yielding

$$\begin{aligned}\frac{P_v(Y^{t+1})}{P_v(\tilde{X}^t)} =& \frac{P_v(Y^{t+1}(v,1))P_v(Y^{t+1}(V(v,1),1))P_v(Y^{t+1}(V,[2,t+1]))}{P_v(\tilde{X}^t(V,[1,t]))}\\ =& p_{\mathbf{i}}(1-p_{\mathbf{s}})^{|V(v,1)|}\\ \leq& 1.\end{aligned}$$

This completes the proof of claim (a) for the SIS model.

It is easy to see that $\delta \leq 1$ for all considered infection spreading models and claim (b) now follows from claim (a), and the proof of Proposition 5.1 is complete.

Proof of Proposition 5.2

To prove Proposition 5.2, it suffices to construct an infection path $\tilde{X}^{t_v}$ with source node v, and show that $P_v(\tilde{X}^{t_v}) \geq P_u(Y^{t_u})$. Let $t_{int}(v)$ be the first infection time of node v in the infection path Y^{t_u} with source node u. We first show that $t_{int}(v) = 1$. Since u is the infection source, the infection can propagate at most $t_u - t_{int}(v)$ hops away from node v within the subtree $T_v(u; H_v \bigcup H_u)$. From Lemma 4.3(ii), if $t_{int}(v) > 1$, we have $t_v = t_u - 1 > t_u - t_{int}(v)$, a contradiction. Therefore, we must have $t_{int}(v) = 1$ in the infection path Y^{t_u}.

For the SI, SIR and SIRI models, we let $\tilde{X}^{t_v}(T_v(u;G),[1,t_v]) = Y^{t_u}(T_v(u;G),[2,t_u])$, yielding

$$\begin{aligned}\frac{P_u(Y^{t_u}(T_v(u;G),[1,t_u]))}{P_v(\tilde{X}^{t_v}(T_v(u;G),[1,t_v]))} &= \frac{P_u(Y^{t_u}(v,1)=\mathbf{i})P_u(Y^{t_u}(T_v(u;G),[2,t_u]))}{P_v(\tilde{X}^{t_v}(T_v(u;G),[1,t_v]))}\\ &= p_{\mathbf{s}}(v). \end{aligned} \tag{5.11}$$

Let $\tilde{X}^{t_v}(u,1) = \mathbf{i}$ and u can be seen as the infection source of the subtree $T_u(v;G)$ with the infection starting at time 1. Applying Proposition 5.1(a) twice, we have

$$\begin{aligned}\frac{P_u(Y^{t_u}(T_u(v;G),[1,t_u]))}{P_v(\tilde{X}^{t_v}(T_u(v;G),[1,t_v]))} &= \frac{P_u(Y^{t_u}(T_u(v;G),[1,t_u]))}{P_v(\tilde{X}^{t_v}(u,1)=\mathbf{i})P_v(\tilde{X}^{t_v}(T_u(v;G),[2,t_v]))}\\ &\leq \frac{\delta^2}{p_{\mathbf{s}}(u)}, \end{aligned} \tag{5.12}$$

where $\delta = (1-\alpha)^2$, $\sqrt{\frac{\alpha}{\beta}}$ and $\sqrt{\frac{\alpha}{\beta}}$ for the SI, SIR and SIRI model, respectively.

Multiplying (5.11) by (5.12), for the SI model, we obtain

$$\begin{aligned}\frac{P_u(Y^{t_u})}{P_v(\tilde{X}^{t_v})} &\le \frac{p_{\mathbf{s}}(v)\cdot(1-\alpha)^4}{p_{\mathbf{s}}(u)} \\ &\le \frac{\beta(1-\alpha)^4}{\alpha} \\ &\le 1,\end{aligned}$$

where the last inequality follows from (5.1). For the SIR and SIRI models, we have

$$\frac{P_u(Y^{t_u})}{P_v(\tilde{X}^{t_v})} \le \frac{p_{\mathbf{s}}(v)}{\beta}\cdot\frac{\alpha}{p_{\mathbf{s}}(u)} \le 1.$$

This completes the proof of Proposition 5.2 in the SI, SIR and SIRI models.

We next consider the SIS model. Following Lemma 5.2, we have $Y^{t_u}(V(u,1)\backslash\{v\},1) = \mathbf{s}$ and we can let $\tilde{X}^{t_v}(V,[1,t_v]) = Y^{t_u}(V,[2,t_u])$. Moreover, following similar arguments as the worst case in (5.20), we have that $Y^{t_u}(u,1) \neq \mathbf{i}$, yielding

$$\begin{aligned}\frac{P_u(Y^{t_u})}{P_v(\tilde{X}^{t_v})} &= \frac{P_u(Y^{t_u}(v,1))P_u(Y^{t_u}(u,1))P_u(Y^{t_u}(V(u,1)\backslash\{v\},1))P_u(Y^{t_u}(V,[2,t_u]))}{P_v(\tilde{X}^{t_v}(V,[1,t_v]))} \\ &= p_{\mathbf{s}}(1-p_{\mathbf{i}})(1-p_{\mathbf{s}})^{|V(u,1)\backslash\{v\}|} \\ &\le 1.\end{aligned}$$

This completes the proof of Proposition 5.2 in the SIS model. The proof of Proposition 5.2 is now complete.

Proof of Lemma 5.1

We first show the following property of the SIRI model in a network with only one node.

Lemma 5.3. Suppose that G has only one node v. For any $t \in [1, +\infty)$, consider any two most likely infection paths X^t for (v,t) and Y^{t+1} for $(v,t+1)$ under the SIRI model. Assume Assumption 5.3 holds. We have

$$\frac{P_v(Y^{t+1})}{P_v(X^t)} \leq \sqrt{\frac{\alpha}{\beta}}. \tag{5.13}$$

Proof. Given any most likely infection path Y^{t+1} for $(v, t+1)$ with $t \in [1, +\infty)$, it suffices to construct another infection path $\tilde{X}^t$ for (v, t) such that

$$\frac{P_v(Y^{t+1})}{P_v(\tilde{X}^t)} \leq \sqrt{\frac{\alpha}{\beta}}. \tag{5.14}$$

Let $\tilde{X}^t(v, [1, t]) = Y^{t+1}(v, [2, t+1])$, we have three cases for Y^{t+1} which are discussed in the following.

Case 1: If $Y^{t+1}(v, 1) = \mathbf{i}$, we have

$$\begin{aligned}\frac{P_v(Y^{t+1})}{P_v(\tilde{X}^t)} &= \frac{P_v(Y^{t+1}(v,1))P_v(Y^{t+1}(v,[2,t+1]))}{P_v(\tilde{X}^t(v,[1,t]))}\\ &= p_{\mathbf{i}}(v)\\ &\leq \sqrt{\frac{\alpha}{\beta}},\end{aligned}$$

where the last inequality holds from (5.3).

Case 2: If $Y^{t+1}(v, 1) = \mathbf{r}$ and $Y^{t+1}(v, 2) = \mathbf{i}$, we have

$$\begin{aligned}\frac{P_v(Y^{t+1})}{P_v(\tilde{X}^t)} &= \frac{P_v(Y^{t+1}(v,[1,2]))P_v(Y^{t+1}(v,[3,t+1]))}{P_v(\tilde{X}^t(v,1))P_v(\tilde{X}^t(v,[2,t]))}\\ &= \frac{(1-p_{\mathbf{i}}(v))p_{\mathbf{r}}(v)}{p_{\mathbf{i}}(v)}\\ &\leq \sqrt{\frac{\alpha}{\beta}},\end{aligned}$$

where the last inequality holds from (5.4).

Case 3: If $Y^{t+1}(v,1) = \mathbf{r}$ and $Y^{t+1}(v,2) = \mathbf{r}$, we have

$$\begin{aligned}\frac{P_v(Y^{t+1})}{P_v(\tilde{X}^t)} &= \frac{P_v(Y^{t+1}(v,[1,2]))P_v(Y^{t+1}(v,[3,t+1]))}{P_v(\tilde{X}^t(v,1))P_v(\tilde{X}^t(v,[2,t]))}\\ &= \frac{(1-p_{\mathbf{i}}(v))(1-p_{\mathbf{r}}(v))}{1-p_{\mathbf{i}}(v)}\\ &\le \sqrt{\frac{\alpha}{\beta}},\end{aligned}$$

where the last inequality holds from (5.4).

We see that (5.14) holds for all three possible cases. The proof for Lemma 5.3 is now complete. □

We note that $T_u(v;G)$ is either an uninfected subtree or infected subtree. In the following, we prove these two cases separately.

- Proof of Lemma 5.1 for Uninfected Subtree

If $T_u(v;G)$ is an uninfected subtree, we can easily see that $\mathcal{T}_v = [1,+\infty)$. It is clear that claim (c) follows from claim (a) and (b). In the following, we prove claim (a) and (b) by mathematical induction on the elapsed time t.

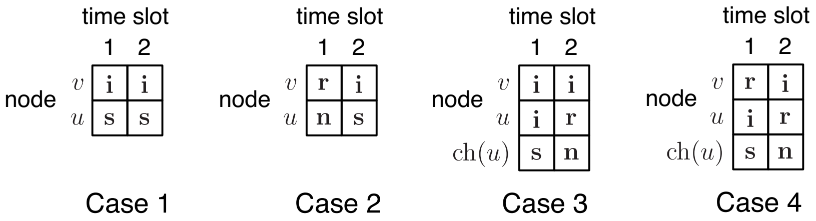


Figure 5-5: Illustration of four possible cases for Y^2, where we omit the states for any node that only have non-susceptible state. We have $Y^2(v,[1,2]) = p_{\mathbf{i}}(v)^2$, $(1-p_{\mathbf{i}}(v))p_{\mathbf{r}}(v)$, $p_{\mathbf{i}}(v)^2$, or $(1-p_{\mathbf{i}}(v))p_{\mathbf{r}}(v)$ for four cases respectively. Moreover, we have $X^2(T_u(v;G),[1,2]) = (1-p_{\mathbf{s}}(u))^2$, $1-p_{\mathbf{s}}(u)$, $p_{\mathbf{s}}(u)(1-p_{\mathbf{i}}(u))\prod_{w\in\mathrm{ch}(u)}(1-p_{\mathbf{s}}(w))$, or $p_{\mathbf{s}}(u)(1-p_{\mathbf{i}}(u))\prod_{w\in\mathrm{ch}(u)}(1-p_{\mathbf{s}}(w))$ for four cases respectively.

Basis step: $t = 1$.

If $v \in V_{\mathbf{i}}$, we let $\tilde{X}^1(v,1) = \mathbf{i}$ and $\tilde{X}^1(u,1) = \mathbf{s}$, then $P_v(\tilde{X}^1(v,1)) = p_{\mathbf{i}}(v)$ and $P_v(\tilde{X}^1(T_u(v;G),1)) = P_v(\tilde{X}^1(u,1)) = 1 - p_{\mathbf{s}}(u)$. As shown in Figure 5-5, there are four possible cases for Y^2. Following Assumption 5.5, we have

$$\begin{aligned}\frac{p_{\mathbf{s}}(u)(1-p_{\mathbf{i}}(u))\prod_{w\in \mathrm{ch}(u)}(1-p_{\mathbf{s}}(w))}{1-p_{\mathbf{s}}(u)} &\leq \frac{(1-p_{\mathbf{i}}(u))(1-\alpha)}{1-\beta}\\ &\leq 1, \qquad (5.15)\end{aligned}$$

where the last inequity holds from (5.3). Then following (5.3), (5.4) and (5.15), we have

$$\begin{aligned}&\frac{P_v(Y^2(v,[1,2]))}{P_v(\tilde{X}^1(v,1))}\\ &= \frac{\max\{p_{\mathbf{i}}(v)^2, (1-p_{\mathbf{i}}(v))p_{\mathbf{r}}(v)\}}{p_{\mathbf{i}}(v)}\\ &\leq \sqrt{\frac{\alpha}{\beta}},\\ &\frac{P_v(Y^2(T_u(v;G),[1,2]))}{P_v(\tilde{X}^1(T_u(v;G),1))}\\ &= \frac{\max\left\{(1-p_{\mathbf{s}}(u))^2, 1-p_{\mathbf{s}}(u), p_{\mathbf{s}}(u)(1-p_{\mathbf{i}}(u))\prod_{w\in \mathrm{ch}(u)}(1-p_{\mathbf{s}}(w))\right\}}{1-p_{\mathbf{s}}(u)}\\ &= 1.\end{aligned}$$

If $v \notin V_{\mathbf{i}}$, we have $\tilde{X}^1(v,1) = \mathbf{r}$ and $\tilde{X}^1(u,1) = \mathbf{n}$, then $P_v(\tilde{X}^1(v,1)) = 1 - p_{\mathbf{i}}(v)$ and $P_v(\tilde{X}^1(T_u(v;G),1)) = P_v(\tilde{X}^1(u,1)) = 1 - p_{\mathbf{s}}(u)$. Change the states of node v at time slot 2 for all four cases in Figure 5-5 from infected to recovered. Then following (5.3), (5.4) and (5.15), we have

$$\begin{aligned}\frac{P_v(Y^2(v,[1,2]))}{P_v(\tilde{X}^1(v,1))} &= \frac{\max\{p_{\mathbf{i}}(v)(1-p_{\mathbf{i}}(v)), (1-p_{\mathbf{i}}(v))(1-p_{\mathbf{r}}(v))\}}{1-p_{\mathbf{i}}(v)},\\ &\leq \sqrt{\frac{\alpha}{\beta}},\\ \frac{P_v(Y^2(T_u(v;G),[1,2]))}{P_v(\tilde{X}^1(T_u(v;G),1))} &= \frac{\max\left\{1-p_{\mathbf{s}}(u), p_{\mathbf{s}}(u)(1-p_{\mathbf{i}}(u))\prod_{w\in \mathrm{ch}(u)}(1-p_{\mathbf{s}}(w))\right\}}{1-p_{\mathbf{s}}(u)}\\ &= 1.\end{aligned}$$

This completes the proof for the basis step.

Inductive step: assume claim (a) and (b) hold for $t = \tau - 1$, where $\tau \geq 2$. We want to show that claim (a) and (b) also hold for $t = \tau$.

Assume $t = \tau$ and consider the following six possible cases for Y^{t+1}.

Case 1: $Y^{t+1}(v, 1) = \mathbf{i}$ and $Y^{t+1}(u, 1) = \mathbf{s}$.

Let $\tilde{X}^t(V, [1, t]) = Y^{t+1}(V, [2, t+1])$. Then following (5.3), we have

$$\begin{aligned}
\frac{P_v(Y^{t+1}(v, [1, t+1]))}{P_v(\tilde{X}^t(v, [1, t]))} &= \frac{P_v(Y^{t+1}(v, 1) = \mathbf{i}) P_v(Y^{t+1}(v, [2, t+1]))}{P_v(\tilde{X}^t(v, [1, t]))} \\
&= p_{\mathbf{i}}(v) \\
&\leq \sqrt{\frac{\alpha}{\beta}}, \\
\frac{P_v(Y^{t+1}(T_u(v; G), [1, t+1]))}{P_v(\tilde{X}^t(T_u(v; G), [1, t]))} &= \frac{P_v(Y^{t+1}(u, 1) = \mathbf{s}) P_v(Y^{t+1}(T_u(v; G), [2, t+1]))}{P_v(\tilde{X}^t(T_u(v; G), [1, t]))} \\
&= 1 - p_{\mathbf{s}}(u) \\
&\leq 1.
\end{aligned}$$

Case 2: $Y^{t+1}(v, 1) = \mathbf{i}$ and $Y^{t+1}(u, 1) = \mathbf{i}$.

Let $\tilde{X}^t(v, [1, t]) = Y^{t+1}(v, [2, t+1])$ and $\tilde{X}^t(u, 1) = \mathbf{i}$. In this case, the states of v do not depend on the states of any other nodes, therefore, it can be seen as the infection source of a graph containing only itself with the infection starting at time 1. Then by Lemma 5.3, we can find a $\tilde{X}^t$ such that

$$\frac{P_v(Y^{t+1}(v, [2, t+1]))}{P_v(\tilde{X}^t(v, [2, t]))} \leq \sqrt{\frac{\alpha}{\beta}}.$$

We then have

$$\begin{aligned}
\frac{P_v(Y^{t+1}(v, [1, t+1]))}{P_v(\tilde{X}^t(v, [1, t]))} &= \frac{P_v(Y^{t+1}(v, 1) = \mathbf{i}) P_v(Y^{t+1}(v, [2, t+1]))}{P_v(\tilde{X}^t(v, 1) = \mathbf{i}) P_v(\tilde{X}^t(v, [2, t]))} \\
&\leq \sqrt{\frac{\alpha}{\beta}}.
\end{aligned}$$

After it gets infected at time 1, node u serves as the infection source of $T_u(v; G)$ with the infection starting at time 1. By the induction assumption, we can find a $\tilde{X}^t$

such that

$$\frac{P_v(Y^{t+1}(T_u(v;G),[2,t+1]))}{P_v(\tilde{X}^t(T_u(v;G),[2,t]))} \leq \sqrt{\frac{\alpha}{\beta}} \leq 1.$$

The following inequality then holds,

$$\begin{aligned}\frac{P_v(Y^{t+1}(T_u(v;G),[1,t+1]))}{P_v(\tilde{X}^t(T_u(v;G),[1,t]))} &= \frac{P_v(Y^{t+1}(u,1)=\mathbf{i})P_v(Y^{t+1}(T_u(v;G),[2,t+1]))}{P_v(\tilde{X}^t(u,1)=\mathbf{i})P_v(\tilde{X}^t(T_u(v;G),[2,t])} \\ &\leq 1.\end{aligned}$$

Case 3: $Y^{t+1}(v,1)=\mathbf{r}$, $Y^{t+1}(v,2)=\mathbf{i}$ and $Y^{t+1}(u,1)=\mathbf{n}$.

Let $\tilde{X}^t(V,[1,t]) = Y^{t+1}(V,[2,t+1])$. Following (5.4), we have

$$\begin{aligned}&\frac{P_v(Y^{t+1}(v,[1,t+1]))}{P_v(\tilde{X}^t(v,[1,t]))} \\ &= \frac{P_v(Y^{t+1}(v,1)=\mathbf{r})P_v(Y^{t+1}(v,2)=\mathbf{i})P_v(Y^{t+1}(v,[3,t+1]))}{P_v(\tilde{X}^t(v,1)=\mathbf{i})P_v(\tilde{X}^t(v,[2,t])} \\ &= \frac{(1-p_{\mathbf{i}}(v))p_{\mathbf{r}}(v)}{p_{\mathbf{i}}(v)} \\ &\leq \sqrt{\frac{\alpha}{\beta}}, \\ &\frac{P_v(Y^{t+1}(T_u(v;G),[1,t+1]))}{P_v(\tilde{X}^t(T_u(v;G),[1,t]))} \\ &= \frac{P_v(Y^{t+1}(u,1)=\mathbf{n})P_v(Y^{t+1}(u,2)=\mathbf{s})P_v(Y^{t+1}(T_u(v;G),[3,t+1]))}{P_v(\tilde{X}^t(u,1)=\mathbf{s})P_v(\tilde{X}^t(T_u(v;G),[2,t])} \\ &= 1.\end{aligned}$$

Case 4: $Y^{t+1}(v,1)=\mathbf{r}$, $Y^{t+1}(v,2)=\mathbf{i}$ and $Y^{t+1}(u,1)=\mathbf{i}$.

Let $\tilde{X}^t(v,[1,t]) = Y^{t+1}(v,[2,t+1])$ and $\tilde{X}^t(u,1) = \mathbf{i}$. Following (5.4), we have

$$\begin{aligned}\frac{P_v(Y^{t+1}(v,[1,t+1]))}{P_v(\tilde{X}^t(v,[1,t]))} &= \frac{P_v(Y^{t+1}(v,1)=\mathbf{r})P_v(Y^{t+1}(v,2)=\mathbf{i})P_v(Y^{t+1}(v,[3,t+1]))}{P_v(\tilde{X}^t(v,1)=\mathbf{i})P_v(\tilde{X}^t(v,[2,t]))}\\ &= \frac{(1-p_{\mathbf{i}}(v))p_{\mathbf{r}}(v)}{p_{\mathbf{i}}(v)}\\ &\le \sqrt{\frac{\alpha}{\beta}}.\end{aligned}$$

Following the same arguments as that in case 2, we can find a $\tilde{X}^t$ such that

$$\begin{aligned}\frac{P_v(Y^{t+1}(T_u(v;G),[1,t+1]))}{P_v(\tilde{X}^t(T_u(v;G),[1,t]))} &= \frac{P_v(Y^{t+1}(u,1)=\mathbf{i})P_v(Y^{t+1}(T_u(v;G),[2,t+1]))}{P_v(\tilde{X}^t(u,1)=\mathbf{i})P_v(\tilde{X}^t(T_u(v;G),[2,t]))}\\ &\le 1.\end{aligned}$$

Case 5: $Y^{t+1}(v,1) = \mathbf{r}$, $Y^{t+1}(v,2) = \mathbf{r}$ and $Y^{t+1}(u,1) = \mathbf{n}$.

Let $\tilde{X}^t(V,[1,t]) = Y^{t+1}(V,[2,t+1])$. Following (5.4), we have

$$\begin{aligned}&\frac{P_v(Y^{t+1}(v,[1,t+1]))}{P_v(\tilde{X}^t(v,[1,t]))}\\ &= \frac{P_v(Y^{t+1}(v,1)=\mathbf{r})P_v(Y^{t+1}(v,2)=\mathbf{r})P_v(Y^{t+1}(v,[3,t+1]))}{P_v(\tilde{X}^t(v,1)=\mathbf{r})P_v(\tilde{X}^t(v,[2,t]))}\\ &= \frac{(1-p_{\mathbf{i}}(v))(1-p_{\mathbf{r}}(v))}{1-p_{\mathbf{i}}(v)}\\ &\le \sqrt{\frac{\alpha}{\beta}},\\ &\frac{P_v(Y^{t+1}(T_u(v;G),[1,t+1]))}{P_v(\tilde{X}^t(T_u(v;G),[1,t]))}\\ &= \frac{P_v(Y^{t+1}(u,1)=\mathbf{n})P_v(Y^{t+1}(u,2)=\mathbf{n})P_v(Y^{t+1}(T_u(v;G),[3,t+1]))}{P_v(\tilde{X}^t(u,1)=\mathbf{n})P_v(\tilde{X}^t(T_u(v;G),[2,t]))}\\ &= 1.\end{aligned}$$

Case 6: $Y^{t+1}(v,1) = \mathbf{r}$, $Y^{t+1}(v,2) = \mathbf{r}$ and $Y^{t+1}(u,1) = \mathbf{i}$.

Let $\tilde{X}^t(v,[1,t]) = Y^{t+1}(v,[2,t+1])$ and $\tilde{X}^t(u,1) = \mathbf{i}$. Following the same arguments as that in case 2, we can find a $\tilde{X}^t$ such that

$$\frac{P_v(Y^{t+1}(T_u(v;G),[2,t+1]))}{P_v(\tilde{X}^t(T_u(v;G),[2,t]))} \le 1.$$

Then following (5.4), we have

$$\begin{aligned}
&\frac{P_v(Y^{t+1}(v,[1,t+1]))}{P_v(\tilde{X}^t(v,[1,t]))}\\
&= \frac{P_v(Y^{t+1}(v,1)=\mathbf{r})P_v(Y^{t+1}(v,2)=\mathbf{r})P_v(Y^{t+1}(v,[3,t+1]))}{P_v(\tilde{X}^t(v,1)=\mathbf{r})P_v(\tilde{X}^t(v,[2,t]))}\\
&= \frac{(1-p_{\mathbf{i}}(v))(1-p_{\mathbf{r}}(v))}{1-p_{\mathbf{i}}(v)}\\
&\le \sqrt{\frac{\alpha}{\beta}},\\
&\frac{P_v(Y^{t+1}(T_u(v;G),[1,t+1]))}{P_v(\tilde{X}^t(T_u(v;G),[1,t]))}\\
&= \frac{P_v(Y^{t+1}(u,1)=\mathbf{i})P_v(Y^{t+1}(T_u(v;G),[2,t+1]))}{P_v(\tilde{X}^t(u,1)=\mathbf{i})P_v(\tilde{X}^t(T_u(v;G),[2,t]))}\\
&\le 1.
\end{aligned}$$

Therefore, we have shown that claim (a) and (b) hold for all six possible cases. This completes the proof for the inductive step. The proof of Lemma 5.1 for uninfected subtree is now complete.

- Proof of Lemma 5.1 for Infected Subtree

If $T_u(v;G)$ is an infected subtree, we can see that $\mathcal{T}_v = [\bar{d}(v,V_{\mathbf{i}}),+\infty)$. We prove claim (a) and (b) for infected subtree by mathematical induction on $\bar{d}(v,V_{\mathbf{i}})$.

Basis step: $\bar{d}(v,V_{\mathbf{i}})=1$**.**

For any $t \ge 1$, we consider any most likely infection path Y^{t+1} for $(v,t+1)$. In the following, six possible cases for Y^{t+1} are discussed in order.

Case 1: $Y^{t+1}(v,1)=\mathbf{i}$ and $Y^{t+1}(u,1)=\mathbf{s}$.

Let $\tilde{X}^t(V,[1,t]) = Y^{t+1}(V,[2,t+1])$. Then following (5.3), we have

$$\begin{aligned}
\frac{P_v(Y^{t+1}(v,[1,t+1]))}{P_v(\tilde{X}^t(v,[1,t]))} &= \frac{P_v(Y^{t+1}(v,1)=\mathbf{i})P_v(Y^{t+1}(v,[2,t+1]))}{P_v(\tilde{X}^t(v,[1,t]))} \\
&= p_{\mathbf{i}}(v) \\
&\le \sqrt{\frac{\alpha}{\beta}}, \\
\frac{P_v(Y^{t+1}(T_u(v;G),[1,t+1]))}{P_v(\tilde{X}^t(T_u(v;G),[1,t]))} &= \frac{P_v(Y^{t+1}(u,1)=\mathbf{s})P_v(Y^{t+1}(T_u(v;G),[2,t+1]))}{P_v(\tilde{X}^t(T_u(v;G),[1,t]))} \\
&= 1-p_{\mathbf{s}}(u) \\
&\le 1.
\end{aligned}$$

Case 2: $Y^{t+1}(v,1)=\mathbf{i}$ and $Y^{t+1}(u,1)=\mathbf{i}$.

Let $\tilde{X}^t(v,[1,t]) = Y^{t+1}(v,[2,t+1])$ and $\tilde{X}^t(u,1)=\mathbf{i}$, following (5.3), we have

$$\begin{aligned}
\frac{P_v(Y^{t+1}(v,[1,t+1]))}{P_v(\tilde{X}^t(v,[1,t]))} &= \frac{P_v(Y^{t+1}(v,1)=\mathbf{i})P_v(Y^{t+1}(v,[2,t+1]))}{P_v(\tilde{X}^t(v,[1,t]))} \\
&= p_{\mathbf{i}}(v) \\
&\le \sqrt{\frac{\alpha}{\beta}}.
\end{aligned}$$

After it gets infected at time slot 1, node u serves as the infection source of the subtree $T_u(v;G)$ with the infection starting at time 1. From the assumption $\bar{d}(v,V_{\mathbf{i}}) = 1$, it follows that $T_w(u;G)$ is an uninfected subtree for any $w \in V(u,1)\bigcap T_u(v;G)$. Then by Lemma 5.1(c) for uninfected subtree, we can find a $\tilde{X}^t$ such that

$$\begin{aligned}
\frac{P_v(Y^{t+1}(T_u(v;G),[2,t+1]))}{P_v(\tilde{X}^t(T_u(v;G),[2,t]))} &\le \sqrt{\frac{\alpha}{\beta}} \\
&\le 1. \qquad (5.16)
\end{aligned}$$

We then have,

$$\frac{P_v(Y^{t+1}(T_u(v;G),[1,t+1]))}{P_v(\tilde{X}^t(T_u(v;G),[1,t]))} = \frac{P_v(Y^{t+1}(u,1)=\mathbf{i})P_v(Y^{t+1}(T_u(v;G),[2,t+1]))}{P_v(\tilde{X}^t(u,1)=\mathbf{i})P_v(\tilde{X}^t(T_u(v;G),[2,t]))}$$
$$\leq 1.$$

Case 3: $Y^{t+1}(v,1)=\mathbf{r}$, $Y^{t+1}(v,2)=\mathbf{i}$ and $Y^{t+1}(u,1)=\mathbf{n}$.

Let $\tilde{X}^t(V,[1,t]) = Y^{t+1}(V,[2,t+1])$. Following (5.4), we have

$$\begin{aligned}
&\frac{P_v(Y^{t+1}(v,[1,t+1]))}{P_v(\tilde{X}^t(v,[1,t]))}\\
&= \frac{P_v(Y^{t+1}(v,1)=\mathbf{r})P_v(Y^{t+1}(v,2)=\mathbf{i})P_v(Y^{t+1}(v,[3,t+1]))}{P_v(\tilde{X}^t(v,1)=\mathbf{i})P_v(\tilde{X}^t(v,[2,t])}\\
&= \frac{(1-p_{\mathbf{i}}(v))p_{\mathbf{r}}(v)}{p_{\mathbf{i}}(v)}\\
&\leq \sqrt{\frac{\alpha}{\beta}},\\
&\frac{P_v(Y^{t+1}(T_u(v;G),[1,t+1]))}{P_v(\tilde{X}^t(T_u(v;G),[1,t]))}\\
&= \frac{P_v(Y^{t+1}(u,1)=\mathbf{n})P_v(Y^{t+1}(u,2)=\mathbf{s})P_v(Y^{t+1}(T_u(v;G),[3,t+1]))}{P_v(\tilde{X}^t(u,1)=\mathbf{s})P_v(\tilde{X}^t(T_u(v;G),[2,t])}\\
&= 1.
\end{aligned}$$

Case 4: $Y^{t+1}(v,1)=\mathbf{r}$, $Y^{t+1}(v,2)=\mathbf{i}$ and $Y^{t+1}(u,1)=\mathbf{i}$.

Let $\tilde{X}^t(v,[1,t]) = Y^{t+1}(v,[2,t+1])$ and $\tilde{X}^t(u,1)=\mathbf{i}$. Following the same arguments as that in case 2, we can find a $\tilde{X}^t$ such that (5.16) holds. Then following (5.4), we have

$$\begin{aligned}\frac{P_v(Y^{t+1}(v,[1,t+1]))}{P_v(\tilde{X}^t(v,[1,t]))} &= \frac{P_v(Y^{t+1}(v,1)=\mathbf{r})P_v(Y^{t+1}(v,2)=\mathbf{i})P_v(Y^{t+1}(v,[3,t+1]))}{P_v(\tilde{X}^t(v,1)=\mathbf{i})P_v(\tilde{X}^t(v,[2,t]))} \\ &= \frac{(1-p_{\mathbf{i}}(v))p_{\mathbf{r}}(v)}{p_{\mathbf{i}}(v)} \\ &\leq \sqrt{\frac{\alpha}{\beta}}, \\ \frac{P_v(Y^{t+1}(T_u(v;G),[1,t+1]))}{P_v(\tilde{X}^t(T_u(v;G),[1,t]))} &= \frac{P_v(Y^{t+1}(u,1)=\mathbf{i})P_v(Y^{t+1}(T_u(v;G),[2,t+1]))}{P_v(\tilde{X}^t(u,1)=\mathbf{i})P_v(\tilde{X}^t(T_u(v;G),[2,t]))} \\ &\leq 1.\end{aligned}$$

Case 5: $Y^{t+1}(v,1)=\mathbf{r}$, $Y^{t+1}(v,2)=\mathbf{r}$ and $Y^{t+1}(u,1)=\mathbf{n}$.

Let $\tilde{X}^t(V,[1,t])=Y^{t+1}(V,[2,t+1])$. Then following (5.4), we have

$$\begin{aligned}&\frac{P_v(Y^{t+1}(v,[1,t+1]))}{P_v(\tilde{X}^t(v,[1,t]))} \\ &= \frac{P_v(Y^{t+1}(v,1)=\mathbf{r})P_v(Y^{t+1}(v,2)=\mathbf{r})P_v(Y^{t+1}(v,[3,t+1]))}{P_v(\tilde{X}^t(v,1)=\mathbf{r})P_v(\tilde{X}^t(v,[2,t]))} \\ &= \frac{(1-p_{\mathbf{i}}(v))(1-p_{\mathbf{r}}(v))}{1-p_{\mathbf{i}}(v)} \\ &\leq \sqrt{\frac{\alpha}{\beta}}, \\ &\frac{P_v(Y^{t+1}(T_u(v;G),[1,t+1]))}{P_v(\tilde{X}^t(T_u(v;G),[1,t]))} \\ &= \frac{P_v(Y^{t+1}(u,1)=\mathbf{n})P_v(Y^{t+1}(u,2)=\mathbf{n})P_v(Y^{t+1}(T_u(v;G),[3,t+1]))}{P_v(\tilde{X}^t(u,1)=\mathbf{n})P_v(\tilde{X}^t(T_u(v;G),[2,t]))} \\ &= 1.\end{aligned}$$

Case 6: $Y^{t+1}(v,1)=\mathbf{r}$, $Y^{t+1}(v,2)=\mathbf{r}$ and $Y^{t+1}(u,1)=\mathbf{i}$.

Let $\tilde{X}^t(v,[1,t])=Y^{t+1}(v,[2,t+1])$ and $\tilde{X}^t(u,1)=\mathbf{i}$. Following the same arguments as that in case 2, we can find a $\tilde{X}^t$ such that (5.16) holds. Then following (5.4), we have

$$
\begin{aligned}
&\frac{P_v(Y^{t+1}(v,[1,t+1]))}{P_v(\tilde{X}^t(v,[1,t]))}\\
&=\frac{P_v(Y^{t+1}(v,1)=\mathbf{r})P_v(Y^{t+1}(v,2)=\mathbf{r})P_v(Y^{t+1}(v,[3,t+1]))}{P_v(\tilde{X}^t(v,1)=\mathbf{r})P_v(\tilde{X}^t(v,[2,t]))}\\
&=\frac{(1-p_{\mathbf{i}}(v))(1-p_{\mathbf{r}}(v))}{1-p_{\mathbf{i}}(v)}\\
&\leq\sqrt{\frac{\alpha}{\beta}},\\
&\frac{P_v(Y^{t+1}(T_u(v;G),[1,t+1]))}{P_v(\tilde{X}^t(T_u(v;G),[1,t]))}\\
&=\frac{P_v(Y^{t+1}(u,1)=\mathbf{i})P_v(Y^{t+1}(T_u(v;G),[2,t+1]))}{P_v(\tilde{X}^t(u,1)=\mathbf{i})P_v(\tilde{X}^t(T_u(v;G),[2,t]))}\\
&\leq 1.
\end{aligned}
$$

We have shown that claim (a) and (b) hold for all six possible cases. This completes the proof for the basis step.

Inductive step: assume claim (a) and (b) hold for $\bar{d}(v,V_{\mathbf{i}})\leq n-1$, where $n\geq 2$. We want to show that claim (a) and (b) also hold for $\bar{d}(v,V_{\mathbf{i}})=n$.

Assume $\bar{d}(v,V_{\mathbf{i}})=n$ and consider any most likely infection path Y^{t+1} for $(v,t+1)$, where $t\geq n$. We first show that (5.16) in case 2 also holds in the inductive step. For case 2, we have $Y^{t+1}(v,1)=\mathbf{i}$ and $Y^{t+1}(u,1)=\mathbf{i}$. Let $\tilde{X}^t(v,[1,t])=Y^{t+1}(v,[2,t+1])$ and $\tilde{X}^t(u,1)=\mathbf{i}$, after it gets infected at time slot 1, node u will serve as the infection source of the subtree $T_u(v;G)$ with the infection starting at time 1. Since $\bar{d}(u,V_{\mathbf{i}}\bigcap T_u(v;G))\leq n-1$, from the induction assumption, we can find a $\tilde{X}^t$ such that (5.16) holds. From the same arguments as that in the basis step, it follows that claim (a) and (b) hold for all six possible cases. This completes the proof for the inductive step. By the spirit of mathematical induction, the proof of Lemma 5.1 for infected subtree is now complete. This completes the proof of Lemma 5.1.

Proof of Lemma 5.2

Let d be the degree of any node in G. Fix the elapsed time to be t and consider any most likely infection path X^t for (v, t). Given any $u \in H_v \backslash \{v\}$, we first show that

$$t_{int}(u) \in [d(v, u), t - \bar{d}(u, T_u(v; H_v))]. \tag{5.17}$$

Firstly, it is easy to see that any node in H_v has been infected at least once due to the assumption that the underlying network G is a tree, otherwise, the infection can not reach the leaf nodes of H_v. We now consider the lower bound of $t_{int}(u)$ in (5.17). Since the infection can spread at most one hop away from v in each time slot, the earliest time for u to get the infection is $d(v, u)$. Then we consider the upper bound of $t_{int}(u)$ in (5.17). After node u gets infected for the first time at $t_{int}(u)$, the infection can spread at most $t - t_{int}(u)$ hops away from u. Consider a node w such that $d(u, w) = \bar{d}(u, T_u(v; H_v))$. In order for the infection to reach node w, $t - t_{int}(u) \geq d(u, w)$. So $t_{int}(u) \leq t - d(u, w) = t - \bar{d}(u, T_u(v; H_v))$. The proof for (5.17) is now complete.

Suppose that there exists a node $u \in H_v \backslash \{v\}$ such that the first infection time $t_{int}(u)$ of u in X^t is less than $t - \bar{d}(u, T_u(v; H_v))$. To prove Lemma 5.2, following (5.17), it suffices to show that we can construct another infection path $\tilde{X}^t$ for (v, t) that occurs with at least the same probability as X^t, where the first infection time of u in $\tilde{X}^t$ is $\tilde{t}(u) = t - \bar{d}(u, T_u(v; H_v))$.

We let the states of $G \backslash (T_u(v; G) \bigcup \{\text{pa}(u)\})$ in $\tilde{X}^t$ to be the same as those in X^t, i.e.

$$\tilde{X}^t(G \backslash (T_u(v; G) \bigcup \{\text{pa}(u)\}), [1, t]) = X^t(G \backslash (T_u(v; G) \bigcup \{\text{pa}(u)\}), [1, t]).$$

We let

$$\tilde{X}^t(\text{pa}(u), [1, t_{int}(u) - 1]) = X^t(\text{pa}(u), [1, t_{int}(u) - 1]),$$
$$A = T_u(v; G) \bigcup \{\text{pa}(u)\} \bigcup V(\text{pa}(u), 1).$$

It suffices to show that

$$P_v(\tilde{X}^t(A,[t_{int}(u),t])) \geq P_v(X^t(A,[t_{int}(u),t])). \tag{5.18}$$

We show (5.18) by mathematical induction on $\bar{d}(u,T_u(v;H_v))$.

Basis step: show (5.18) holds for $\bar{d}(u,T_u(v;H_v))=0$.

Let B denote the set of nodes $V(\text{pa}(u),1)\backslash\{u\}$. Consider a time slot $\tau < \tilde{t}(u)$ where $\tilde{X}^t(\text{pa}(u),\tau)=\mathbf{i}$. We show the worst case for $\tilde{X}^t$ at time $\tau+1$.

If $X^t(\text{pa}(u),\tau)=\mathbf{i}$, we have

$$P_v(\tilde{X}^t(B,\tau+1)) = P_v(X^t(B,\tau+1). \tag{5.19}$$

If $X^t(\text{pa}(u),\tau)\neq\mathbf{i}$, we have

$$\frac{P_v(\tilde{X}^t(B,\tau+1))}{P_v(X^t(B,\tau+1)} \geq (1-p_{\mathbf{s}})^{d-1}, \tag{5.20}$$

where the equality holds when every node in B is susceptible in $\tilde{X}^t$ and non-susceptible in X^t at time τ. By (5.19) and (5.20), we can see that the worst case for $\tilde{X}^t$ at time $\tau+1$ is that $X^t(\text{pa}(u),\tau)\neq\mathbf{i}$ and $X^t(B,\tau)=\mathbf{n}$.

We divide the time interval $[t_{int}(u),t]$ into three parts: $t_{int}(u)$, $[t_{int}(u)+1,\tilde{t}(u)-1]$ and $\tilde{t}(u)$, where $\tilde{t}(u)=t-\bar{d}(u,T_u(v;H_v))=t$.

Part 1: time $\tau=t_{int}(u)$.

Since node u is infected for the first time at time slot $t_{int}(u)$ in X^t, we know that node $\text{pa}(u)$ must be infected at time $t_{int}(u)-1$, which in turn suggests that $\tilde{X}^t(\text{pa}(u),\tau-1)=X^t(\text{pa}(u),\tau-1)=\mathbf{i}$, yielding

$$P_v(\tilde{X}^t(B,\tau)) = P_v(X^t(B,\tau)). \tag{5.21}$$

We let $\tilde{X}^t(\text{pa}(u),\tau)=\mathbf{i}$ and consider the worst case in (5.20). Following (5.5) and

(5.21), we have

$$\begin{aligned}\frac{P_v(\tilde{X}^t(A, t_{int}(u)))}{P_v(X^t(A, t_{int}(u)))} &= \frac{P_v(\tilde{X}^t(\mathrm{pa}(u), t_{int}(u)))P_v(\tilde{X}^t(u, t_{int}(u)))P_v(\tilde{X}^t(B, t_{int}(u)))}{P_v(X^t(\mathrm{pa}(u), t_{int}(u)))P_v(X^t(u, t_{int}(u)))P_v(X^t(B, t_{int}(u)))} \\ &\geq \frac{p_{\mathbf{i}}(1-p_{\mathbf{s}})}{(1-p_{\mathbf{i}})p_{\mathbf{s}}} \qquad (5.22) \\ &\geq 1. \qquad (5.23)\end{aligned}$$

Part 2: time $\tau \in [t_{int}(u)+1, t-1]$.

We first consider the case that at least one node in $T_u(v;G)$ is infected at time τ. We let $\tilde{X}^t(\mathrm{pa}(u), \tau) = \mathbf{i}$ and consider the worst case, i.e., $X^t(\mathrm{pa}(u), \tau-1) \neq \mathbf{i}$ and $X^t(B, \tau-1) = \mathbf{n}$. We then have

$$P_v(\tilde{X}^t(A, \tau)) \geq p_{\mathbf{i}}(1-p_{\mathbf{s}})^d.$$

Since $X^t(\mathrm{pa}(u), \tau-1) \neq \mathbf{i}$ and at least one node in $T_u(v;G)$ is infected at time τ, there must exist a node $w \in T_u(v;G)$, s.t., $X^t(w, \tau-1) = \mathbf{i}$. Consider any neighboring node z of w. If $X^t(z, \tau-1) = \mathbf{i}$, due to the fact that $X^t(\mathrm{pa}(u), \tau-1) \neq \mathbf{i}$ and the assumption that G is an infinite tree, we can always find a node $y \in T_z(w;G) \bigcap (T_u(v;G) \bigcup \{\mathrm{pa}(u)\})$, s.t., $X^t(y, \tau-1) = \mathbf{s}$. If $X^t(z, \tau-1) = \mathbf{s}$, following similar arguments as the worst case in (5.20), we can see that $X^t(z, \tau) \neq \mathbf{i}$. We then have

$$P_v(X^t(T_z(w;G) \bigcap (T_u(v;G) \bigcup \{\mathrm{pa}(u)\}), \tau)) \leq 1 - p_{\mathbf{s}},$$

for any neighboring node z of w. Moreover, we have at least one node in $T_u(v;G)$ being infected at time τ, yielding

$$\begin{aligned}P_v(X^t(A, \tau)) &\leq \max\{p_{\mathbf{i}}, p_{\mathbf{s}}\}(1-p_{\mathbf{s}})^d \\ &= p_{\mathbf{i}}(1-p_{\mathbf{s}})^d \\ &\leq P_v(\tilde{X}^t(A, \tau)). \qquad (5.24)\end{aligned}$$

We then consider the case that no node in $T_u(v;G)$ is infected at time τ. Without loss of generality, we assume τ is the earliest time after $t_{int}(u)$ that no node in

$T_u(v;G)$ is infected. We let $\tilde{X}(\text{pa}(u),\tau) = X^t(\text{pa}(u),\tau)$ and consider the worst case for $\tilde{X}^t(\text{pa}(u),\tau)$. If $X^t(\text{pa}(u),\tau) = \mathbf{i}$, we have

$$\begin{aligned}\frac{P_v(\tilde{X}^t(A,\tau))}{P_v(X^t(A,\tau))} &\geq \frac{p_{\mathbf{i}}(1-p_{\mathbf{s}})^d}{p_{\mathbf{s}}(1-p_{\mathbf{i}})(1-p_{\mathbf{s}})^{d-1}} \\ &\geq 1. \end{aligned} \tag{5.25}$$

If $X^t(\text{pa}(u),\tau) \neq \mathbf{i}$, we have

$$\begin{aligned}\frac{P_v(\tilde{X}^t(A,\tau))}{P_v(X^t(A,\tau))} &\geq \frac{(1-p_{\mathbf{i}})(1-p_{\mathbf{s}})^d}{(1-p_{\mathbf{s}})(1-p_{\mathbf{i}})(1-p_{\mathbf{s}})^{d-1}} \\ &= 1. \end{aligned} \tag{5.26}$$

From (5.24)-(5.26), we have

$$P_v(\tilde{X}^t(A,[t_{int}(u)+1,\tau])) \geq P_v(X^t(A,[t_{int}(u)+1,\tau])). \tag{5.27}$$

We have now $\tilde{X}^t(V,\tau) = X^t(V,\tau)$. If there are other time slots after τ that no node in $T_u(v;G)$ is infected, we can apply the same arguments again. Then by (5.24) and (5.27), we have

$$P_v(\tilde{X}^t(A,[t_{int}(u)+1,t-1])) \geq P_v(X^t(A,[t_{int}(u)+1,t-1])). \tag{5.28}$$

Part 3: time $\tau = t$.

If $\text{pa}(u) \in V_{\mathbf{i}}$, we have

$$\begin{aligned}\frac{P_v(\tilde{X}^t(A,t))}{P_v(X^t(A,t))} &\geq \frac{p_{\mathbf{i}}p_{\mathbf{s}}(1-p_{\mathbf{s}})^{d-1}}{p_{\mathbf{s}}p_{\mathbf{i}}(1-p_{\mathbf{s}})^{d-1}} \\ &= 1. \end{aligned} \tag{5.29}$$

Then (5.18) holds from (5.23), (5.28) and (5.29).

If $\mathrm{pa}(u) \notin V_{\mathbf{i}}$, we have

$$\begin{aligned}\frac{P_v(\tilde{X}^t(A,t))}{P_v(X^t(A,t))} &\geq \frac{(1-p_{\mathbf{i}})p_{\mathbf{s}}(1-p_{\mathbf{s}})^{d-1}}{(1-p_{\mathbf{s}})p_{\mathbf{i}}(1-p_{\mathbf{s}})^{d-1}} \\ &= \frac{(1-p_{\mathbf{i}})p_{\mathbf{s}}}{p_{\mathbf{i}}(1-p_{\mathbf{s}})}. \end{aligned} \tag{5.30}$$

Then (5.18) holds from (5.22), (5.28) and (5.30). This competes the proof of the basis step.

Inductive step: assume (5.18) holds for $\bar{d}(u, T_u(v; H_v)) \leq n$, where $0 \leq n \leq \bar{d}(v, H_v) - 1$. Show (5.18) also holds for $\bar{d}(u, T_u(v; H_v)) = n + 1$.

We divide the time interval $[t_{int}(u), t]$ into four parts: $t_{int}(u)$, $[t_{int}(u)+1, \tilde{t}(u)-1]$, $\tilde{t}(u)$ and $[\tilde{t}(u) + 1, t]$, where $\tilde{t}(u) = t - \bar{d}(u, T_u(v; H_v))$. For any node $w \in \mathrm{ch}(u)$, we have $\bar{d}(w, T_w(v; H_v)) \leq \bar{d}(u, T_u(v; H_v)) - 1 = n$. By induction assumption, we have that node w get infected for the first time at $t(w) = t - \bar{d}(w, T_w(v; H_v))$ in X^t, which in turn suggests that $X^t(u, \tilde{t}(u)) = \mathbf{i}$. For the time range $[\tilde{t}(u) + 1, t]$, we let $\tilde{X}^t(A, [\tilde{t}(u) + 1, t]) = X^t(A, [\tilde{t}(u) + 1, t])$, yielding

$$P_v(\tilde{X}^t(A, [\tilde{t}(u) + 1, t])) = P_v(X^t(A, [\tilde{t}(u) + 1, t])).$$

For the first three parts, following similar arguments as in the basis step, we have

$$P_v(\tilde{X}^t(A, [t_{int}(u) + 1, \tilde{t}(u)])) \geq P_v(X^t(A, [t_{int}(u) + 1, \tilde{t}(u)])).$$

We can now conclude that (5.18) holds for the inductive step. By the spirit of mathematical induction, (5.18) holds and the proof of Lemma 5.2 is now complete.

THIS PAGE INTENTIONALLY LEFT BLANK

Chapter 6

Multiple Infection Sources Identification in the SI Model under the MLIP Criterion

As discussed in Section 1.3.1, it is important to study the multiple infection sources estimation problem. In Chapter 3, we proposed a heuristic procedure to find the multiple infection sources in the SI model based on the single source ML estimator for regular trees, but we have not shown it to be optimal. In this chapter, by adopting the most likely infection path statistical criterion (cf. Section 2.3), we are able to propose an optimal estimator for the set of infection sources in the SI model.

6.1 Problem Formulation

In this chapter, we consider the discrete time SI model discussed in Section 2.1. We assume that the infection probability follows Assumption 5.1, and topology of the underlying graph G follows Assumption 5.5. We consider the MLIP criterion discussed in Section 2.3 and use the notations and definitions introduced in Section 4.1.1. We assume there are $k > 1$ infection sources and want to find the solution of (2.1).

6.2 Sources Estimation for Trees

In this section, we show that k-Jordan infection center set (cf. Definition 4.2) is an optimal source set estimator when the underlying graph is a tree network. Without loss of generality, we assume that the minimum subgraph B of G that contains $V_{\mathbf{i}}$ is connected, otherwise the same estimation procedure can be applied to each component of B. We first show a similar result as that in Proposition 5.1. The proof of Proposition 6.1 is provided in Section 6.5.

Proposition 6.1. Suppose that the underlying network G is an infinite tree, the infection sources are $S = \{s_1, s_2, \cdots, s_k\}$, and the set of observed infected nodes $V_{\mathbf{i}}$ is non-empty. For an infection spreading under the SI model, any most likely infection path X^t for (S, t) has the following properties:

(a) $P_S(X^t)$ is non-increasing in $t \in \mathcal{T}_S$; and

(b) conditioned on S being the infection sources, a most likely elapsed time for X^t is given by

$$t_S = \bar{d}(S, V_{\mathbf{i}}).$$

In the following, we show how to transform the k sources estimation problem to an equivalent single source estimation problem, then we can use Theorem 5.1 to find the optimal multiple sources estimator. We first introduce the definition of *super node graph*.

Definition 6.1 (Super node graph)**.** Suppose that G is an infinite tree, and the set $S = \{s_1, s_2, \cdots, s_k\}$ are the infection sources, where $S \subset V$ and $k > 1$. Given any infection path X^t conditioned on S being the infection sources, the super node graph $\tilde{G}(S, X^t)$ is constructed using the following procedure for each $\tau = 0, 1, \ldots, t$:

- Starting at $\tau = 0$, we initialize $A_i = \{s_i\}$ for each $i = 1, \ldots, k$.
- For each $\tau = 1, \ldots, t$, consider every node $v \in V_{\mathbf{i}}$ that becomes susceptible at time τ in X^t for the first time. Let N_v be the set of neighboring nodes of v that

is infected at time $\tau - 1$. We choose a node $u \in N_v$ uniformly at random, and include v and the edge (u, v) in the component A_i that u belongs to.

- Based on the resulting graph $\mathcal{A} = \bigcup_{i=1}^{k} A_i$, the *super node graph* $\tilde{G}(S, X^t)$ is constructed by considering all infection sources as a single virtual node, which we call a super node and denote as Supernode(S).

Given any infection path X^t following the SI model, it can be shown that (with probability one) the conditional probability $P_S(X^t)$ is the same for G and any corresponding $\tilde{G}(S, X^t)$ as defined in Definition 6.1. Consider any node v with $|N_v|> 1$ and assume v becomes susceptible at time slot t_1 and becomes infected at time slot t_2. Then $P_S(X^t(v, [1, t])) = (1 - p_{\mathbf{s}})^{t_2 - t_1 - 1} p_{\mathbf{s}}$, regardless of the number of infected neighbors v has as long as there is at least one infected neighbor.[1] We formally present this result in the following lemma.

Lemma 6.1. Suppose that the set $S = \{s_1, s_2, \cdots, s_k\}$ are the infection sources, where $S \subset V$ and $k > 1$. Given any infection path X^t conditioned on S being the infection sources, $P_S(X^t)$ is the same for both G and any corresponding $\tilde{G}(S, X^t)$ as defined in Definition 6.1.

Following Lemma 6.1, instead of searching for a most likely infection path for S in G, we can now search for a most likely infection path for Supernode(S) in a corresponding super node graph $\tilde{G}(S, X^t)$. In this way, we transform the k sources estimation problem to an equivalent single source estimation problem. As discussed in Chapter 5, Theorem 5.1 shows that a Jordan center of the infected node set is an optimal single source estimator. Therefore, our objective is to find a set of k nodes S, where Supernode(S) is a Jordan center of the infected node set in $\tilde{G}(S, X^t)$. We show in the following lemma that k-Jordan center set is the solution.

Lemma 6.2. Suppose that G is an infinite tree and the set of infected nodes $V_{\mathbf{i}}$ is non-empty. Given any infection path X^t consistent with $V_{\mathbf{i}}$ under the SI model, if

[1] This property does not hold for an infection following the SIR, SIRI or SIS model, where some infected neighbors of v may recover after t_1 and $P_S(X^t(v, [1, t]))$ may change if we remove some edges connecting v.

$S = \{s_1, s_2, \cdots, s_k\}$ is the k-Jordan center set of $V_{\mathbf{i}}$ in G, then Supernode(S) is a Jordan center of $V_{\mathbf{i}}$ in any corresponding super node graph $\tilde{G}(S, X^t)$.

The proof of Lemma 6.2 is provided in Section 6.5. The following theorem follows immediately from Lemma 6.2 and Theorem 5.1.

Theorem 6.1. Suppose that G is an infinite tree and there are $k > 1$ infection sources. For an infection in the SI model, a k-Jordan center set of $V_{\mathbf{i}}$ is an optimal source set estimator for (2.1).

Theorem 6.1 is consistent with Theorem 5.1 for an infection in the SI model. Due to the difficulty described in footnote 1, the multiple-sources estimation problem remains an open problem for more complicated infection spreading models including SIR, SIRI and SIS models. To verify the robustness of the proposed estimators, we conduct extensive simulations on both trees and general networks for SI, SIR, SIRI and SIS models in Section 6.4.

6.3 Sources Estimation for General Graphs

In this section, we consider the case where the underlying network G is a general graph. For $k > 1$, we heuristically extend the k-Jordan center set estimator to general graphs and propose a heuristic algorithm to find the k-Jordan center set.

When k is greater than 1, it is usually impractical to use exhaustive search methods to find the k-Jordan center set as the number of possible k-Jordan center sets is $\binom{|V|}{k}$. Therefore, we propose a heuristic algorithm to find an approximate k-Jordan center set when there are $k > 1$ sources and the underlying network is a general graph, which we call the Multiple Jordan Center set estimation algorithm (MJC). MJC starts with randomly selecting a set of k nodes $\hat{S}^0 = \{s_i^0\}_{i=1}^k$ as the initial guess, and then utilizes an iterative two-step optimization approach. Specifically, in iteration l, let $\hat{S}^l = \{s_i^l\}_{i=1}^k$ be the k-Jordan center set estimate. We perform the following two steps at each iteration l:

- **Partition step**. In this step, MJC partitions $V_{\mathbf{i}}$ into k sets $M_1, M_2, \cdots, M_k$ such that for all $v \in M_i$, $d(s_i^{l-1}, v) \leq d(s_j^{l-1}, v)$ if $i \neq j$. We call M_i the *Voronoi set* corresponding to s_i^{l-1}. To do this, let each s_i^{l-1} broadcast a message. The broadcasting process terminates when each node $v \in V_{\mathbf{i}}$ receives at least one message from a node in $\hat{S}^{l-1}$. In the broadcasting process, each node $v \in V_{\mathbf{i}}$ learns the distance between itself and the nearest nodes in $\hat{S}^{l-1}$. We choose a nearest node in $\hat{S}^{l-1}$ at random, and add v to the Voronoi set corresponding to this node.

- **Re-optimization step**. In this step, MJC updates each estimate s_i^{l-1} in the Voronoi sets M_i. For each Voronoi set M_i, run SJC (cf. Section 5.3) to find the Jordan center of M_i and set it as the new estimate s_i^l.

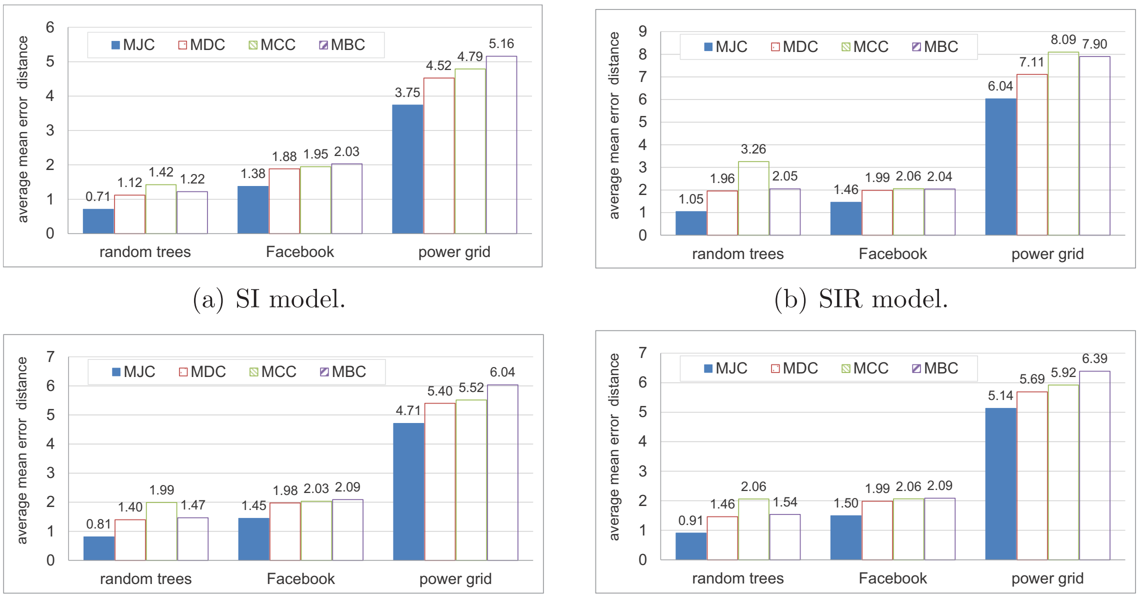
MJC terminates when $\max_{1 \leq i \leq k} d(s_i^{l-1}, s_i^l) \leq \eta$ for some predetermined small positive value η or when the number of iterations reach a predetermined positive number MaxIter. For the partition step in each iteration, the computation complexity is dominated by the broadcasting process, with a computational complexity of $O(k|E|)$. For the re-optimization step in each iteration, the computational complexity is $O(|V||E|)$ due to SJC. Therefore, the overall computational complexity for MJC is $O(\text{MaxIter} \cdot |V||E|)$.

6.4 Simulation Results

In this section, we evaluate the performance of the proposed estimator on the same three kinds of networks as used in Section 5.4: random tree networks where the degree of every node is randomly chosen from $[3, 5]$, a small part of the Facebook network with 4039 nodes [69], which is shown in [70] to be a scale-free network, and the western states power grid network of the United States [63]. We consider the case where two infection sources exist ($k = 2$). By finding distance center, closeness center or betweenness center (cf. Section 4.4) of each Voronoi set in the re-optimization step of MJC, MJC can be adapted to heuristically find multiple distance center set (MDC), multiple closeness center set (MCC) or multiple betweenness center set (MBC), respectively. We use the resulted heuristic algorithms as benchmarks.

For the SI, SIR, SIRI and SIS models, we randomly choose the corresponding infection probabilities $p_{\mathbf{s}}(v)$, $p_{\mathbf{i}}(v)$, $p_{\mathbf{r}}(v)$ from $[0, 1]$ for any node v. For each kind of network and each infection spreading model, we perform 1000 simulation runs. In each simulation run, we randomly pick two nodes as the infection sources and simulate the infection using the above mentioned spreading model. The spreading terminates when the number of infected nodes is greater than 100. We then run MJC on the observed infected nodes to estimate the infection sources and compare the result with the benchmarks.

To quantify the performance of each algorithm, we first match the estimated with the actual sources so that the sum of the error distances between each estimated source and its match is minimized. Then the mean error distance is the average of the error distances for all matched pairs, and is shown in Fig. 6-1. We see that the proposed estimator performs better than the benchmarks for all considered networks under all considered infection spreading models.



(a) SI model.

(b) SIR model.

(c) SIRI model.

(d) SIS model.

Figure 6-1: Average mean error distances for various networks under different infection spreading models when there are two infection sources.

6.5 Proofs

Proof of Proposition 6.1

For any set of nodes $M \subset A$, we let $T_M(S;A) = \bigcup_{v \in M} T_v(S;A)$. For any $t \in \mathcal{T}_S$, consider any most likely infection path Y^{t+1} for $(S, t+1)$. To show claim (a), it suffices to construct an infection path $\tilde{X}^t$ for (S,t) such that

$$P_S(Y^{t+1}) \leq P_S(\tilde{X}^t). \tag{6.1}$$

We start with the case where $k=2$ and show (6.1) by mathematical induction on $d(s_1, s_2)$.

Basis step (i): The inequality (6.1) holds for $d(s_1, s_2) = 1$.

The states of $T_{s_1}(S;G)$ and $T_{s_2}(S;G)$ are independent. We can treat s_1 and s_2 as the infection source of $T_{s_1}(S;G)$ and $T_{s_2}(S;G)$, respectively. Then following Proposition 5.1, we can find a $\tilde{X}^t$ such that

$$\begin{aligned}\frac{P_S(Y^{t+1})}{P_S(\tilde{X}^t)} &= \frac{P_S(Y^{t+1}(T_{s_1}(S;G),[1,t+1]))}{P_S(\tilde{X}^t(T_{s_1}(S;G),[1,t]))}\frac{P_S(Y^{t+1}(T_{s_2}(S;G),[1,t+1]))}{P_S(\tilde{X}^t(T_{s_2}(S;G),[1,t]))}\\ &\leq 1.\end{aligned}$$

Basis step (ii): The inequality (6.1) holds for $d(s_1, s_2) = 2$.

Denote the common neighboring node of s_1 and s_2 to be u. Consider the following two possible cases of Y^{t+1}.

Case 1: $Y^{t+1}(u,1) = \mathbf{i}$.

We let $\tilde{X}^t(u,1) = \mathbf{i}$, conditioning on which the states of $T_{s_1}(S;G)$, $T_{s_2}(S;G)$ and $T_u(S;G)$ are independent. Moreover, u can be seen as the infection source of $T_u(S;G)$ with the infection starting at time 1. Then following Proposition 5.1, we can find a $\tilde{X}^t$ such that

$$\begin{aligned}\frac{P_S(Y^{t+1})}{P_S(\tilde{X}^t)} &= \frac{P_S(Y^{t+1}(T_{s_1}(S;G),[1,t+1]))}{P_S(\tilde{X}^t(T_{s_1}(S;G),[1,t]))}\frac{P_S(Y^{t+1}(T_{s_2}(S;G),[1,t+1]))}{P_S(\tilde{X}^t(T_{s_2}(S;G),[1,t]))}\\ &\quad \frac{P_S(Y^{t+1}(u,1))}{P_S(\tilde{X}^t(u,1))}\frac{P_S(Y^{t+1}(T_u(S;G),[2,t+1]))}{P_S(\tilde{X}^t(T_u(S;G),[2,t]))}\\ &\leq 1.\end{aligned}$$

Case 2: $Y^{t+1}(u,1) = \mathbf{s}$.

We let $\tilde{X}^t(T_u(S;G),[1,t]) = Y^{t+1}(T_u(S;G),[2,t+1])$. Then following Proposition 5.1, we can find a $\tilde{X}^t$ such that

$$\begin{aligned}\frac{P_S(Y^{t+1})}{P_S(\tilde{X}^t)} &= \frac{P_S(Y^{t+1}(T_{s_1}(S;G),[1,t+1]))}{P_S(\tilde{X}^t(T_{s_1}(S;G),[1,t]))}\frac{P_S(Y^{t+1}(T_{s_2}(S;G),[1,t+1]))}{P_S(\tilde{X}^t(T_{s_2}(S;G),[1,t]))}\\ &\frac{P_S(Y^{t+1}(u,1))P_S(Y^{t+1}(T_u(S;G),[2,t+1]))}{P_S(\tilde{X}^t(T_u(S;G),[1,t]))}\\ &\leq 1 - p_{\mathbf{s}}(u)\\ &\leq 1.\end{aligned}$$

Inductive step: If (6.1) holds for $d(s_1,s_2) \leq n$, then (6.1) also holds for $d(s_1,s_2) = n+1$, where $n \geq 2$.

Let $\rho(v,u)$ be the path between two nodes v and u. Denote the neighboring node of s_1 and s_2 in $\rho(s_1,s_2)$ to be u_1 and u_2, respectively. Consider the following four possible cases of Y^{t+1}.

Case 1: $Y^{t+1}(u_1,1) = \mathbf{i}$ and $Y^{t+1}(u_2,1) = \mathbf{i}$.

We let $\tilde{X}^t(u_1,1) = \mathbf{i}$ and $\tilde{X}^t(u_2,1) = \mathbf{i}$. Then u_1 and u_2 can be seen as the pair of infection sources of $T_{\rho(u_1,u_2)}(S;G)$ with the infection starting at time 1. Moreover, we have $d(u_1,u_2) = d(s_1,s_2) - 2 = n-1$. Then by induction assumption, we can find a $\tilde{X}^t$ such that

$$P_S(Y^{t+1}(T_{\rho(u_1,u_2)}(S;G),[2,t+1])) \leq P_S(\tilde{X}^t(T_{\rho(u_1,u_2)}(S;G),[2,t])).$$

Then by Proposition 5.1, we can find a $\tilde{X}^t$ such that (6.1) holds.

Case 2: $Y^{t+1}(u_1,1) = \mathbf{i}$ and $Y^{t+1}(u_2,1) = \mathbf{s}$.

We let $\tilde{X}^t(u_1,1) = \mathbf{i}$ and $\tilde{X}^t(u_2,1) = \mathbf{s}$. Then u_1 and s_2 can be seen as the pair of infection sources of $T_{\rho(u_1,u_2)} \bigcup \{s_2\}$ with the infection starting at time 1. Moreover, we have $d(u_1,u_2) = d(s_1,s_2) - 2 = n-1$. Then by induction assumption, we can find a $\tilde{X}^t$ such that

$$P_S(Y^{t+1}(T_{\rho(u_1,u_2)}(S;G),[2,t+1])) \leq P_S(\tilde{X}^t(T_{\rho(u_1,u_2)}(S;G),[2,t])).$$

Then by Proposition 5.1, we can find a $\tilde{X}^t$ such that (6.1) holds.

Case 3: $Y^{t+1}(u_1,1) = \mathbf{s}$ and $Y^{t+1}(u_2,1) = \mathbf{i}$.

Following similar arguments as that in Case 2, we can find a $\tilde{X}^t$ such that (6.1) holds.

Case 4: $Y^{t+1}(u_1, 1) = \mathbf{s}$ and $Y^{t+1}(u_2, 1) = \mathbf{s}$.

We let $\tilde{X}^t(T_{\rho(u_1,u_2)}(S; G), [1, t]) = Y^{t+1}(T_{\rho(u_1,u_2)}(S; G), [2, t + 1])$. Then following Proposition 5.1, we can find a $\tilde{X}^t$ such that

$$\begin{aligned}\frac{P_S(Y^{t+1})}{P_S(\tilde{X}^t)} &= \frac{P_S(Y^{t+1}(T_{s_1}(S;G),[1,t+1]))}{P_S(\tilde{X}^t(T_{s_1}(S;G),[1,t]))}\frac{P_S(Y^{t+1}(T_{s_2}(S;G),[1,t+1]))}{P_S(\tilde{X}^t(T_{s_2}(S;G),[1,t]))}\\ &\quad \frac{P_S(Y^{t+1}(u_1,1))P_S(Y^{t+1}(u_2,1))P_S(Y^{t+1}(T_{\rho(u_1,u_2)}(S;G),[2,t+1]))}{P_S(\tilde{X}^t(T_{\rho(u_1,u_2)}(S;G),[1,t]))}\\ &\le (1 - p_{\mathbf{s}}(u_1))(1 - p_{\mathbf{s}}(u_2))\\ &\le 1.\end{aligned}$$

This completes the proof for the inductive step. By the spirit of mathematical induction, we have shown that (6.1) holds for $k = 2$. When $k > 2$, similar arguments can be applied to each pair of source nodes, and this completes the proof of claim (a).

We show that $\mathcal{T}_S = [\bar{d}(S, V_{\mathbf{i}}), +\infty)$. Consider any node $l \in V_{\mathbf{i}}$ such that $d(S, l) = \bar{d}(S, V_{\mathbf{i}})$. The infection can propagate at most one hop further from any source node in one time slot. If $t < \bar{d}(S, V_{\mathbf{i}})$, the infection can not reach node l. Claim (b) now follows from claim (a), and the proof of Proposition 6.1 is complete.

Proof of Lemma 6.2

We first show that the value of the minimum infection range in $\tilde{G}(S, X^t)$ can not be less than $\bar{d}(S, V_{\mathbf{i}})$. Assume there is a super node Supernode(S') in $\tilde{G}(S, X^t)$ that is associated with a set of k nodes $S' \subset V$ such that, $\bar{d}(\text{Supernode}(S'), V_{\mathbf{i}}) < \bar{d}(S, V_{\mathbf{i}})$. Then it is implied that $\bar{d}(S', V_{\mathbf{i}}) < \bar{d}(S, V_{\mathbf{i}})$, which contradicts with the assumption that S is a k-Jordan center set.

We then show that Supernode(S) is a Jordan center of $V_{\mathbf{i}}$ in the transformed super node graph $\tilde{G}(S, X^t)$, i.e., Supernode(S) has the minimum infection range in $\tilde{G}(S, X^t)$. In other words, we want to show that $d(\text{Supernode}(S), v) \le \bar{d}(S, V_{\mathbf{i}})$ for any node $v \in V_{\mathbf{i}}$. From Definition 6.1, it suffices to show that $d(s_i, v) \le \bar{d}(S, V_{\mathbf{i}})$ for any node $v \in A_i$, where $i \in \{1, 2, \cdots, k\}$. Suppose that there exists a node $v \in A_i$

such that $d(s_i, v) \geq \bar{d}(S, V_{\mathbf{i}}) + 1$. Then the first infection time $t_{int}(v)$ of v in X^{t_S} is

$$\begin{aligned} t_{int}(v) &\geq d(s_i, v) \\ &\geq \bar{d}(S, V_{\mathbf{i}}) + 1, \end{aligned}$$

because the infection can spread at most one hop further from s_i in one time slot. Following Proposition 6.1(b), we have that $t_S = \bar{d}(S, V_{\mathbf{i}}) < t_{int}(v)$, a contradiction. Therefore we have $d(s_i, v) \leq \bar{d}(S, V_{\mathbf{i}})$ for any $v \in A_i$, where $i \in \{1, 2, \cdots, k\}$. This competes the proof of Lemma 6.2.

Chapter 7

Conclusion and Future Work

7.1 Conclusion

In this thesis, we have investigated the problem of identifying $k \geq 1$ infection source(s) in a tree/general network where the underlying infection spreading follows the SI, SIR, SIRI or SIS model under the ML or MLIP criterion.

In Chapter 3, we considered the multiple infection sources identification problem in the SI model under the ML criterion. We derived estimators for the infection sources and regions when the number of infection sources is bounded but unknown a priori. The estimators are based only on knowledge of the infected nodes and their underlying network connections. We provided an approximation for the infection source estimator for the class of geometric trees, and when there are at most two sources in the network. We showed that this estimator asymptotically correctly identifies the infection sources when the number of infected nodes grows large. We also proposed an algorithm that estimates the number of source nodes, and identify them and their respective infection regions for general infection graphs. Simulation results on geometric trees, regular trees, small-world networks, the US power grid network, and experimental results on the SARS infection network and cascading power outages show that our proposed estimation procedure performs well in general, with an average error distance of less than 4. The estimation accuracy of the number of source nodes is over 65% in all the networks we consider, with the geometric tree networks having an accuracy of over 90%. Furthermore, the minimum infection region covering percentage is more than 59% for all networks.

In Chapter 4, we investigated the single infection source identification problem in the SI model with limited observations under the MLIP criterion. When the network is a tree, we showed that a Jordan center is an optimal source estimator. We proposed an efficient algorithm with complexity $O(n)$ to find the estimator, where n is the size of the network. In the case of general networks, we proposed approximate source estimators based on a MIQCQP formulation, which has high complexity, and a heuristic algorithm with complexity $O(n^3)$.

In Chapter 5, we studied the single infection source identification problem in the SI, SIR, SIRI and SIS models under the MLIP criterion. For the case where the underlying network is a regular tree, we showed that a Jordan center of the infected node set is an universal infection source estimator for the SI, SIR, SIRI or SIS model. This is a somewhat surprising result since the four infection spreading models are fundamentally different.

In Chapter 6, we investigated the multiple infection sources identification problem in the SI model under the MLIP criterion. When the underlying network is a tree, we showed that the k-Jordan center set is an optimal infection source set estimator. Simulations have been conducted on random trees, part of the Facebook network and the western states power grid network of the United States. The results suggest that our estimators perform constantly better than the distance, closeness, and betweenness centrality based heuristics.

7.2 Future Work

In this thesis, we have assumed that we have access to very limited information. A possible future work includes incorporating information like infection times, infection directions and observation time into the estimation procedure.

We have proposed heuristic algorithms for general networks. It would be interesting if theoretical results could be provided for general networks.

Bibliography

[1] B. Viswanath, A. Mislove, M. Cha, and K. P. Gummadi. On the evolution of user interaction in Facebook. In *Proc. 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*, pages 337–357. Springer New York, 2010.

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

[4] H. Kwak, C. Lee, H. Park, and S. Moon. What is Twitter, a social network or a news media? In *Proc. International Conference on World Wide Web*, 2010.

[5] V. Gundotra. Google+: communities and photos. http://googleblog.blogspot.sg/2012/12/google-communities-and-photos.html, December 2012.

[6] R. Alcarria, T. Robles, and G. Camarillo. Towards the convergence between IMS and social networks. In *Proc. International Conference on Wireless and Mobile Communications*, 2010.

[7] G. Kossinets and D. J. Watts. Empirical analysis of an evolving social network. *Science*, 311(5757):88–90, 2006.

[8] J. O. Kephart and S. R. White. Directed-graph epidemiological models of computer viruses. In *Proc. IEEE Computer Society Symp Research in Security and Privacy*, 1991.

[9] K. T. Goh, J. Cutter, B. H. Heng, S. Ma, B. K. W. Koh, C. Kwok, C. M. Toh, and S. K. Chew. Epidemiology and control of SARS in Singapore. *Annals Of The Academy Of Medicine Singapore*, 35(5):301–316, 2006.

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

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

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

[13] S. H. Lim, S. W. Kim, S. Park, and J. H. Lee. Determining content power users in a blog network: an approach and its applications. *IEEE Trans. Syst., Man, Cybern. A*, 41(5):853–862, 2011.

[14] L. Akritidis, D. Katsaros, and P. Bozanis. Identifying the productive and influential bloggers in a community. *IEEE Trans. Syst., Man, Cybern. C*, 41(5):759–764, 2011.

[15] R. D. Smith. Responding to global infectious disease outbreaks: lessons from SARS on the role of risk perception, communication and management. *Social Science & Medicine*, 63(12):3113–3123, 2006.

[16] WHO. Summary of probable SARS cases with onset of illness from 1 November 2002 to 31 July 2003. http://www.who.int/csr/sars/country/table2004˙04˙21/en/, December 2003.

[17] N. Golgowski. '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. http://www.dailymail.co.uk/news/article-2313652/AP-Twitter-hackers-break-news-White-House-explosions-injured-Obama.html, April 2013.

[18] W. Jingqiong and L. Xinzhu. Radiation fears prompt panic buying of salt. http://www.chinadaily.com.cn/cndy/2011-03/18/content˙12189705.htm, March 2011.

[19] United States. Federal Energy Regulatory Commission and North American Electric Reliability Corporation. *Arizona-Southern California Outages on September 8, 2011: Causes and Recommendations*. 2012.

[20] M. Newman and M. Girvan. Finding and evaluating community structure in networks. *Phys. Rev. E*, 69(2), 2004.

[21] C. Scoglio, W. Schumm, P. Schumm, T. Easton, S. Roy Chowdhury, A. Sydney, and M. Youssef. Efficient mitigation strategies for epidemics in rural regions. *PLoS ONE*, 5(7), 2010.

[22] D. Shah and T. Zaman. Rumor in a network: who's the culprit? In *Proc. NIPS Workshop on Analyzing Networks and Learning with Graphs*, 2009.

[23] D. Shah and T. Zaman. Rumors in a network: Who's the culprit? *IEEE Trans. Inf. Theory*, 57(8):5163–5181, 2011.

[24] N. Bailey. *The Mathematical Theory of Infectious Diseases and its Applications*. Griffin, 1975.

[25] W. J. Bai, T. Zhou, and B. H. Wang. Immunization of susceptible-infected model on scale-free networks. *Physica A: Statistical Mechanics and its Applications*, 384(2):656–662, 2007.

[26] A. Wu and Y. Wang. Role of diffusion in an epidemic model of mobile individuals on networks. *The European Physical Journal B*, 85(8):1–6, 2012.

[27] Y. Shang. Mixed SI(R) epidemic dynamics in random graphs with general degree distributions. *Applied Mathematics and Computation*, 219(10):5042–5048, 2013.

[28] Y. F. Chou, H. H. Huang, and R. G. Cheng. Modeling information dissemination in generalized social networks. *IEEE Commun. Lett.*, 17(7):1356–1359, 2013.

[29] L. J. Allen. Some discrete-time SI, SIR, and SIS epidemic models. *Mathematical Biosciences*, 124(1):83–105, 1994.

[30] M. Barthelemy, A. Barrat, R. Pastor-Satorras, and A. Vespignani. Velocity and hierarchical spread of epidemic outbreaks in scale-free networks. *Physical Review Letters*, 92:178701, 2004.

[31] G. Yan, T. Zhou, J. Wang, Z. Q. Fu, and B. H. Wang. Epidemic spread in weighted scale-free networks. *Chinese Physics Letters*, 22(2):510, 2005.

[32] T. Zhou, G. Yan, and B.-H. Wang. Maximal planar networks with large clustering coefficient and power-law degree distribution. *Phys. Rev. E*, 71:046141, 2005.

[33] T. Zhou, J. G. Liu, W. J. Bai, G. Chen, and B.-H. Wang. Behaviors of susceptible-infected epidemics on scale-free networks with identical infectivity. *Phys. Rev. E*, 74:056109, 2006.

[34] S. Tang and W. Li. An epidemic model with adaptive virus spread control for wireless sensor networks. *International Journal of Security and Networks*, 6(4):201–210, 2011.

[35] W. Dong, W. Zhang, and C. W. Tan. Rooting out the rumor culprit from suspects. In *Proc. IEEE International Symposium on Information Theory*, 2013.

[36] P. C. Pinto, P. Thiran, and M. Vetterli. Locating the source of diffusion in large-scale networks. *Phys. Rev. Lett.*, 109:068702, 2012.

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

[38] M. E. J. Newman. Spread of epidemic disease on networks. *Phys. Rev. E*, 66:016128, 2002.

[39] Y. Moreno, R. Pastor Satorras, and A. Vespignani. Epidemic outbreaks in complex heterogeneous networks. *The European Physical Journal B - Condensed Matter and Complex Systems*, 26(4):521–529, 2002.

[40] Y. Moreno, J. B. Gómez, and A. F. Pacheco. Epidemic incidence in correlated complex networks. *Phys. Rev. E*, 68:035103, 2003.

[41] Y. Moreno, M. Nekovee, and A. Vespignani. Efficiency and reliability of epidemic data dissemination in complex networks. *Phys. Rev. E*, 69:055101, 2004.

[42] D. F. Zheng, P. Hui, S. Trimper, and B. Zheng. Epidemics and dimensionality in hierarchical networks. *Physica A: Statistical Mechanics and its Applications*, 352(2-4):659–668, 2005.

[43] R. Yang, B. H. Wang, J. Ren, W. J. Bai, Z. W. Shi, W. X. Wang, and T. Zhou. Epidemic spreading on heterogeneous networks with identical infectivity. *Physics Letters A*, 364(3-4):189–193, 2007.

[44] K. Zhu and L. Ying. A robust information source estimator with sparse observations. In *Proc. IEEE Conference on Computer Communications*, 2014.

[45] Z. Chen, K. Zhu, and L. Ying. Detecting multiple information sources in networks under the SIR model. In *Proc. Annual Conference in Information Sciences and Systems*, 2014.

[46] S. M. Blower, T. C. Porco, and G. Darby. Predicting and preventing the emergence of antiviral drug resistance in HSV-2. *Nature Medicine*, 4:673–678, 1998.

[47] P. V. D. Driessche and X. Zou. Modeling relapse in infectious diseases. *Mathematical Biosciences*, 207(1):89–103, 2007.

[48] V. D. L. Cruz. On the global stability of infectious diseases models with relapse. *Abstraction & Application*, 9:50–61, 2013.

[49] P. Georgescu and H. Zhang. A lyapunov functional for a SIRI model with nonlinear incidence of infection and relapse. *Applied Mathematics and Computation*, 219(16):8496 – 8507, 2013.

[50] H. W. Hethcote and J. A. Yorke. *Gonorrhea Transmission Dynamics and Control*. Lecture Notes in Biomathematics. Springer-Verlag, 1984.

[51] H. Hethcote. Qualitative analyses of communicable disease models. *Mathematical Biosciences*, 28(3-4):335–356, 1976.

[52] M. E. J. Newman. The structure and function of complex networks. *SIAM Review*, 45:167–256, 2003.

[53] W. Luo and W. P. Tay. Identifying infection sources in large tree networks. In *Proc. IEEE International Conference on Sensing, Communication, and Networking*, 2012.

[54] W. Luo and W. P. Tay. Identifying multiple infection sources in a network. In *Proc. Asilomar Conference on Signals, Systems, and Computers*, 2012.

[55] W. Luo, W. P. Tay, and M. Leng. Identifying infection sources and regions in large networks. *IEEE Trans. Signal Process.*, 61(11):2850–2865, 2013.

[56] W. Luo and W. P. Tay. Estimating infection sources in a network with incomplete observations. In *Proc. IEEE Global Conference on Signal and Information Processing*, 2013.

[57] W. Luo, W. P. Tay, and M. Leng. How to identify an infection source with limited observations. *IEEE J. Sel. Top. Sign. Proces.*, 8(4):586–597, 2014.

[58] W. Luo and W. P. Tay. Finding an infection source under the SIS model. In *Proc. IEEE International Conference on Acoustics, Speech, and Signal Processing*, 2013.

[59] W. Luo, W. P. Tay, and M. Leng. On the universality of Jordan centers for estimating infection sources in tree networks. *arXiv:1411.2370*, 2014.

[60] G. Brightwell and P. Winkler. Counting linear extensions is #P-complete. In *Proc. Annual ACM Symposium on Theory of Computing*, 1991.

[61] G. Sabidussi. The centrality index of a graph. *Psychometrika*, 31(4):581–603, 1966.

[62] A. L. Barabasi and R. Albert. Emergence of scaling in random networks. *Science*, 286(5439):509–512, 1999.

[63] D. J. Watts and S. H. Strogatz. Collective dynamics of 'small-world' networks. *Nature*, 393(6684):440–442, 1998.

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

[65] S. M. Hedetniemi, E. J. Cockayne, and S. T. Hedetniemi. Linear algorithms for finding the Jordan center and path center of a tree. *Transportation Science*, 15(2):98–114, 1981.

[66] J. Currie and D. I. Wilson. OPTI: lowering the barrier between open source optimizers and the industrial MATLAB user. In *Proc. Foundations of Computer-Aided Process Operations*, 2012.

[67] T. Achterberg. SCIP: solving constraint integer programs. *Mathematical Programming Computation*, 1(1):1–41, 2009.

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

[69] J. McAuley and J. Leskovec. Learning to discover social circles in ego networks. In *Proc. Neural Information Processing Systems Conference*, 2012.

[70] S. Catanese, P. D. Meo, E. Ferrara, G. Fiumara, and A. Provetti. Extraction and analysis of Facebook friendship relations. *Computational Social Networks: Mining and Visualization*, 2011.