Maximum *a Posteriori* Estimation for Information Source Detection

Biao Chang, Enhong Chen, *Senior Member, IEEE*, Feida Zhu, Qi Liu, *Member, IEEE*, Tong Xu, and Zhefeng Wang

***Abstract*—Information source detection is to identify nodes initiating the diffusion process in a network, which has a wide range of applications including epidemic outbreak prevention, Internet virus source identification, and rumor source tracing in social networks. Although it has attracted ever-increasing attention from research community in recent years, existing solutions still suffer from high time complexity and inadequate effectiveness, due to high dynamics of information diffusion and observing just a snapshot of the whole process. To this end, we present a comprehensive study for *single information source detection in weighted graphs*. Specifically, we first propose a maximum *a posteriori* (MAP) estimator to detect the information source with other methods as the prior, which ensures our method can be integrated with others naturally. Different from many related works, we exploit both infected nodes and their uninfected neighbors to calculate the *effective propagation probability*, and then derive the exact formation of likelihood for general weighted graphs. To further improve the efficiency, we design two approximate MAP estimators, namely brute force search approximation (BFSA) and greedy search bound approximation (GSBA), from the perspective of likelihood approximation. BFSA tries to traverse the permitted permutations to directly compute the likelihood, but GSBA exploits a strategy of greedy search to find a surrogate upper bound of the likelihood, and thus avoids the enumeration of permitted permutations. Therefore, detecting with partial nodes and likelihood approximation reduces the computational complexity drastically for large graphs. Extensive experiments on several data sets also clearly demonstrate the effectiveness of our methods on detecting the single information source with different settings in weighted graphs.**

***Index Terms*—Greedy search, information source detection, likelihood approximation, maximum *a posteriori* (MAP).**

Manuscript received October 30, 2017; revised December 19, 2017; accepted February 14, 2018. Date of publication May 2, 2018; date of current version May 15, 2020. This work was supported in part by the National Natural Science Foundation of China under Grant U1605251, Grant 61672483, and Grant 61703386, in part by the National Science Foundation for Distinguished Young Scholars of China under Grant 61325010, and in part by the Anhui Provincial Natural Science Foundation under Grant 1708085QF140. The work of E. Chen was supported by the U.S. National Science Foundation for Distinguished Young Scholars of China. This paper was recommended by Associate Editor M. Celenk. *(Corresponding author: Enhong Chen.)*

B. Chang, E. Chen, Q. Liu, T. Xu, and Z. Wang are with the Anhui Province Key Laboratory of Big Data Analysis and Application, School of Computer Science and Technology, University of Science and Technology of China, Hefei 230026, China (e-mail: chbiao@mail.ustc.edu.cn; cheneh@ustc.edu.cn; qiliuql@ustc.edu.cn; tongxu@ustc.edu.cn; zhefwang@mail.ustc.edu.cn).

F. Zhu is with the School of Information Systems, Singapore Management University, 178902 Singapore (e-mail: fdzhu@smu.edu.sg).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TSMC.2018.2811410

I. Introduction

THE BOOM of research on social network analysis [1]–[5] has brought ever-increasing attention to the topics of information source detection [6]–[8], influence maximization [9]–[12], and so on [13]–[16] in recent years. *Information source detection* aims to identify the nodes initiating the diffusion process based on a single snapshot of the infected network (e.g., diffusion of opinion, rumor, and epidemic). Its wide range of applications include epidemic outbreak prevention, Internet virus source identification, and rumor source tracing in social networks [7], [17]–[20].

The research challenges of this problem come from a number of aspects. First, information diffusion is characteristic of high dynamics and displays a great variety of patterns when initiating from different sources [21]. For example, in social networks, a photograph will be shared more times if it is posted by a celebrity. Second, the actual information diffusion laws are unknown. Although many models have been proposed such as the susceptible-infected-recovered (SIR) model [22] and independent cascade (IC) model [23], they cannot describe information diffusion comprehensively. Third, we only observe a snapshot of the infected network, which is just a part of the whole diffusion process. Nevertheless, various methods have been introduced along the years to overcome these challenges and detect the source of a diffusion for different situations, including methods based on centrality [8], [24], spectral [17], [19], belief propagation (BP) [25]–[27], and so on. However, existing methods are still deemed inadequate due to their high computational complexity and yet-to-be-improved effectiveness. For example, rumor center (RC) [8] only considers the utility of infected nodes to detect the source in homogeneous graphs, and Jordan center (JC) [6] just aims to minimize the maximum distance from the source to others and neglects the dynamicity of information diffusion. Dynamic message passing (DMP) [27], one of the state-of-the-arts, exploits all nodes in the network to estimate the marginal probability of a given node to be in a given state and how long the information has already propagated, which is too time-consuming.

Therefore, in this paper we extend our preliminary work [28] which focused on unweighted graphs, and present a comprehensive study for *single information source detection in weighted graphs*. The intuition behind our method is that uninfected (or susceptible) nodes provide important negative signals for detecting the source. We illustrate this point with

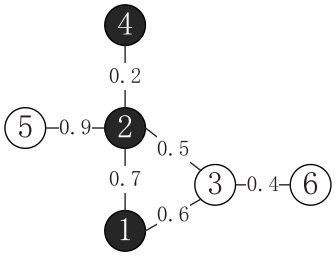


Fig. 1. Snapshot of the information diffusion under the SI model on a toy graph, where black nodes are infected and others are susceptible. Numbers on edges are the information propagation probability between two adjacent nodes.

Fig. 1 which shows a snapshot of the diffusion example on an undirected graph. The susceptible-infected (SI) model [8], [19] which is a variant of SIR, is used to describe the process of information diffusion. It assumes that every node has two potential states, namely susceptible and infected. If node 2 is indeed the source, its two uninfected neighbors nodes 3 and 5 would be more likely to be infected. That means the presence of these two uninfected neighbors reduces the probability of node 2 being the source. Additionally, nodes 3 and 5 should have different importance for node 2, due to their different propagation probabilities between node 2. Although this has been noticed by some work [19], [27], we exploit this intuition along the following different direction.

Specifically, we first derive a maximum *a posteriori* (MAP) estimator to detect the single information source in undirected and weighted graphs, which selects a node with the maximal posterior probability as the detected source. It applies other simple but effective methods such as *rumor centrality* (RC) [8] and *Jordan centrality* [24] as the prior because Comin and da Fontoura Costa [29] have shown that the source node tends to have higher centrality measurement values. This ensures our method can be integrated with others naturally. Then we infer the exact likelihood formation of the observed infected subgraph, based on the hypothesis that the likelihood equals to the sum of probabilities of all permitted permutations starting with a node. A permitted permutation [8] is corresponding to the node infection sequence which is generated by information diffusion and can span the observed infected subgraph. To compute the probability of a permitted permutation, we exploit both infected and their uninfected neighbors to get an *effective propagation probability* in this paper. It generalizes the probability of propagating information from a set of nodes to a single node, and thus makes it also suitable for weighted graphs with loops. For example, in Fig. 1, $\{1, 2, 4\}$ is a permitted permutation but $\{1, 4, 2\}$ is not if node 1 is the source. When nodes 1 and 2 are infected, the effective propagation probability from nodes 1 and 2 to node 3 is $1 - (1 - 0.6) \cdot (1 - 0.5) = 0.8$. Additionally, unlike DMP [27], our method only exploits a part of nodes to detect, which reduces the computational complexity drastically for large graphs. Combining the above prior and likelihood, we obtain the MAP estimator.

For better efficiency, we design two approximate MAP estimators, namely brute force search approximation (BFSA) and greedy search bound approximation (GSBA), from the perspective of likelihood approximation. Inspired by Shah and Zaman [8], BFSA uses a breadth-first search tree to estimate the latent spanning tree generated by the diffusion, and then enumerates the corresponding permitted permutations to derive the approximate likelihood. However, BFSA is still time-consuming as Shah and Zaman's [8] research results have shown the factorial complexity of the number of permitted permutations for general trees. Therefore, we further propose GSBA which uses an upper bound to approximate the probabilities of permitted permutations starting with a given node. To find this upper bound, GSBA exploits a strategy of greedy search to find a *surrogate* bound, which selects the node maximizing the increment of its likelihood when generating a permitted permutation. GSBA effectively avoids the enumeration of permitted permutations and further reduces the computational complexity. Experimental results on several data sets also clearly validate the effectiveness of our methods on detecting the single information source with different settings in weighted graphs.

To sum up, our contributions are listed as follows.

1) We present a comprehensive study for *single information source detection in weighted graphs* and derive an MAP estimator. It can integrate with other methods naturally and exploit the effective propagation probability to infer the exact formation of likelihood for general graphs.
2) To improve the efficiency, we develop two approximation variants of the MAP estimator, namely BFSA and GSBA, from the perspective of likelihood approximation.
3) We conduct comprehensive experiments on several networks to validate our methods. The experimental results clearly demonstrate the effectiveness of our proposed approaches for single information source detection with different settings in weighted graphs.

Roadmap: The remainder of this paper is organized as follows. Section II provides a brief review of related works. Then we introduce some preliminaries of information source detection in Section III. Sections IV and V give the details of our methods. In Section VI, we report the experimental results. Finally, we conclude this paper and discuss some future works in Section VII.

II. Related Work

In general, research work related to our problem can be discussed by two categories: 1) information diffusion modeling and 2) information source detection.

A. Information Diffusion Modeling

It is a fundamental problem to model information diffusion process, which has attracted research efforts from various communities including epidemiology, ethnography, and sociology [1]. Kermack and McKendrick [22] introduced the SIR model to describe epidemic spreading. The model assumes that every node has three possible states, i.e., susceptible, infected, and recovered. Once a susceptible node is infected, it can further infect its susceptible neighbors, but it may recover

and never get infected again. Note that the SI model used in this paper supposes that infected nodes would never recover, which is a special case of SIR. In social network analysis, IC model [23] and linear threshold model [30] are widely used to describe the information diffusion in social networks. Other models such as SI-susceptible model and diffusion of innovations can be found in [1].

Intuitively, information source detection can be viewed as the reverse process of information diffusion [9]. Lappas *et al.* [31] defined a similar problem, *k*-effectors, which selects a set of *k* active nodes that can best explain the observed activation states in social networks. They proved that the *k*-effectors(0) problem is NP-complete under the IC model, and gave two approximate solutions. Nguyen *et al.* [32] studied the *k*-suspector problem which aims to find the top *k* most suspected sources of misinformation, and claimed NP-hardness of the problem under the IC model. Gundecha *et al.* [18] tried to seek the provenance of information for a few known recipients by recovering the information propagation paths in social media. Feng *et al.* [33] studied the problem of recovering other unknown recipients and seeking the provenance of information based on a few known recipients. They exploited frequent pattern propensity and node centrality measures to find important nodes.

B. Information Source Detection

Various methods (e.g., those based on centrality, spectral, BP, and so on) have been proposed to identify the single diffusion source for different situations. For example, Shah and Zaman [7], [8] are among the first to consider this problem. They proposed RC to implement the maximum likelihood estimation (MLE) for single rumor source detection under the homogeneous SI model. For a node, its RC is the number of infection sequences which can span the observed infected subgraph. We can see RC only considers the utility of infected nodes to detect the source for unweighted graphs. Dong *et al.* [34] explored the MAP estimation with different settings of the prior. For instance, the suspects may be all the infected nodes, or at most *k* infected nodes. However, the likelihood is computed based on RC, which also has the above drawbacks. Besides, Wang *et al.* [35] addressed the problem of rumor source detection with multiple independent observations, under the SI model. For trees, they found that multiple independent observations can dramatically increase the detection probability. Jain *et al.* [36] assumed the infected subgraph is observed at some known time instant, and proposed a random walk-based method to approximate the MLE of the rumor source.

Zhu and Ying [6] developed a sample-path-based approach to detect the source under the SIR model. The source is supposed to be the root of a sample path which is the node most likely resulting in the infected subgraph. They proved that for a tree graph, the output of their method is a JC [24], which minimizes the maximum distance from a node to others. Recently, Lokhov *et al.* [27] made use of the infected and uninfected nodes to detect the source and introduced a time-consuming yet effective inference algorithm based on DMP equations. It first uses DMP to estimate the marginal probability of a given node to be in a given state, and then exploits a mean-field-type approach to approximate the likelihood. Altarelli *et al*. [26] conducted Bayesian inference for this problem on a factor graph under the SIR model. They derived BP equations for the probability distribution of system states conditioned on some observations, which is more accurate than DMP. They further considered this problem with noisy observations [25].

Sometimes, we can observe the infection time of specific sensor nodes, not only their states. Under a specific continuous-time epidemic process, Pinto *et al.* [37] studied the problem when only a small fraction of nodes, instead of the whole graph, can be observed. After that, Agaskar and Lu [38] described an alternate representation for the SI model, which allows us to estimate the marginal distributions for each observer and compute a pseudo-likelihood function that is maximized to find the source and Shen *et al.* [39] developed a time-reversal backward spreading algorithm to locate the source of a diffusion-like process efficiently, which detects the node with the minimum variance of reversed arrival time from sensors. Kumar *et al.* [40] inferred the source of a rumor on the network with relative information about the infection times of a fraction of node pairs, and proposed Markov chain Monte Carlo-based schemes.

In addition, many researchers focused on detecting multiple information sources. Prakash *et al.* [19] started to explore the detection of multiple information sources under the homogeneous SI model. They applied the minimum description length principle to automatically decide the number of source nodes, and then identified the best source nodes according to *exoneration* of infected nodes with many uninfected neighbors. Subsequently, Fioriti and Chinnici [17] proposed to use the *node dynamical importance* to estimate nodes' age, and designed a spectral technique to predict the sources of an outbreak. Dynamical importance of a node is the reduction of the largest eigenvalue of the adjacent matrix after it is removed from the network [41]. Luo *et al.* [42] extended RC for multiple sources detection, and they also tried to estimate the infection regions (i.e., nodes infected by each source). Nguyen *et al.* [43] proposed an approximation algorithm for multiple infection sources identification with provable guarantees under the homogeneous SI model. It minimizes the symmetric difference between observed infected nodes and the cascade from source nodes, and then identify infection sources without the prior knowledge on the number of source nodes.

In spite of all these existing work, we approach the problem of information source detection under the heterogeneous SI model by maximizing *a posteriori*. Our solution makes full use of both infected nodes and their uninfected neighbors, like [19]. We assume that every infected node could be the source and use the output of other methods as the prior. Then we infer the exact formation of the likelihood for general weighted graphs. Additionally, we also design two approximate MAP estimators from the perspective of likelihood approximation for better efficiency.

TABLE I
TERMS AND NOTATIONS

Notations	Description
$G = G(V, E, A)$	an undirected graph
G_I	an infected subgraph with N nodes of G
v^*	the real information source
$\hat{v}$	the detected source
σ	a permitted permutation
$\sigma(1, \ldots, i)$	the first i nodes of σ
$R(v, G_I)$	rumor centrality of node v
$G_s(U)$	a spanning graph of a node set U
$N(U)$	neighbors of a node set U
$d(U)$	degree of a node set U
$E(v, U)$	a set of bridging edges between v and the node set U
$\Omega(v, G_I)$	a set of permitted permutations starting with v and spanning G_I.

III. PRELIMINARIES

In this section, we will first give the problem definition, revisit RC for explaining some basic concepts, and then introduce the framework of our solution. Important terms and notations are listed in Table I for easy reference.

A. Problem Definition

Let $G(V, E, A)$ denote the undirected and weighted network, where V is the node set and E is the edge set. $A = [a_{ij}]$, and $a_{ij} \in [0, 1]$ is the information propagation probability from nodes i to j. In real-world scenarios, A can be learned from historical interactions by partial credits [44]. The information (such as opinion, rumor, and epidemic) will spread on this network under a contagious information diffusion model. In this paper, we assume the source consists of a single node, and apply the heterogeneous SI model to describe this diffusion process.

Heterogeneous SI is a variant of the popular SIR model [22]. It assumes that every node has two possible states: 1) susceptible and 2) infected. Once a node i is infected (or receives the information), it will remain infected and never recover any more. Meanwhile, node i will spread the information to its susceptible neighboring node j with probability a_{ij} in the next. The diffusions along edges are supposed to be independent [19], [45].

After the information has spread on the network for some time, there are N infected nodes, denoted by V_I, including the source node. These nodes and their interedges E_I can span an *infected subgraph* $G_I(V_I, E_I)$ of $G(V, E)$, which are referred to as G_I and G, respectively. G_I is connected because the information diffusion model is contagious, and thus every susceptible node can only be infected by its neighbors. For example, recall the snapshot of information diffusion on the toy graph in Fig. 1, nodes 1, 2, and 4 are infected, and others are susceptible. Here the observed data only includes the graph structure and a snapshot of the diffusion indicating who are infected, but we do not know when each infection occurs in many scenarios, such as computer virus spread over the Internet and epidemic outbreak among the crowd. How can we find the real source node initiating the diffusion based on this snapshot? Therefore, the problem of *single information source detection* is defined as follows [8].

Problem 1 (Single Information Source Detection): Given an undirected and weighted graph $G(V, E, A)$ and a snapshot of the infected subgraph $G_I(V_I, E_I)$ at some unknown time stamp, the problem of *single information source detection* is to find the source v^* among those infected nodes, which can infect others and span G_I.

B. Rumor Centrality

This problem was first studied by Shah and Zaman [7], [8]. They showed that it is one of the #P-complete problems, and proposed the following RC to detect the source with MLE under the homogeneous SI model which assumes all the propagation probabilities along edges are equal.

After the information diffusion starts from the source, it generates an infection node sequence $\sigma = \{v_1, \ldots, v_N\}$ $(1 \leq i \leq N)$, where $v_i \in V_I$ is the ith infected node (i.e., $\sigma(i) = v_i$). This sequence is sorted in chronological order by when the nodes were infected and corresponds to a permutation of these infected nodes, which is also referred to as *permitted permutation* by Shah and Zaman [8]. Additionally, a permitted permutation corresponds one to one with an infection sequence. This means that a permutation is permitted only if it exactly matches the topological constrain specified by G_I. For example, if node 4 is the information source in Fig. 1, {4,2,1} is a permitted permutation, but {4,1,2} is not because node 2 must be infected before node 1. Note that we also use σ to denote the nodes appearing in the sequence without ambiguity.

If G_I is a general tree, Shah and Zaman [8] have shown that the number of permitted permutations starting with v, namely RC of node v, is defined as

$$R(v, G_I) = \prod_{u \in G_I} \frac{N!}{T_u^v} \tag{1}$$

where u is a node of G_I and T_u^v is the number of nodes in the subtree rooted at u with v as the source. They assumed that each node is equally likely to be the source, and exploited RC to estimate the likelihood probability $P(G_I|v^* = v)$ given that node v is the information source. The detected source is an RC maximizing the RC, i.e., MLE.

Although RC is effective in many cases, it has three limitations. First, it only considers the infected subgraph and neglects other uninfected nodes which are also important for detecting the information source. As mentioned in the introduction, those uninfected neighbors may indicate lower probability to be the source for infected nodes. Second, it assumes that the propagation probabilities along all edges are equal, and detects the source only based on the topological structure, which is obviously found from (1). But this assumption is invalid under many scenarios where the graph is weighted. Third, RC assumes that the probabilities of all permitted permutation are equal for general graphs. It is easy

to show that this assumption is not valid when the degrees of nodes are different, especially for weighted graphs with loops.

C. Framework of Our Solution

To overcome the above limitations and improve the accuracy, we design a solution based on MAP estimation. Let $P(v^* = v)$ denote the prior probability that node v is the source, and $P(G_I|v^* = v)$ denote the likelihood probability that G_I will be observed if the information propagates from v. Based on Bayes theorem, we can derive the posterior probability $P(v^* = v|G_I)$ of node v being the real source given the infected subgraph G_I as follows:

$$\begin{aligned} P\big(v^* = v|G_I\big) &= \frac{P(v^* = v)P(G_I|v^* = v)}{P(G_I)} \\ &= \frac{P(v^* = v)P(G_I|v^* = v)}{\sum_{u\in G_I} P(G_I|v^* = u)} \\ &\propto P(v^* = v)P\big(G_I|v^* = v\big) \end{aligned} \tag{2}$$

because the denominator $P(G_I)$ is the sum of values appearing in the numerator and can be regarded as the normalization constant to be removed [46]. We can see that the posterior is proportional to the product of the prior probability $P(v^* = v)$ and likelihood $P(G_I|v^* = v)$.

Given G_I, we can select the node maximizing the above posterior as the detected source $\hat{v}$. This is the following MAP estimator.

$$\begin{aligned} \hat{v} &= \arg\max_{v\in G_I} P\big(v^* = v|G_I\big) \\ &= \arg\max_{v\in G_I} P\big(G_I|v^* = v\big)P(v^* = v). \end{aligned} \tag{3}$$

We will introduce the above prior probability $P(v^* = v)$ and likelihood $P(G_I|v^* = v)$ in the next section.

IV. Details of Maximum *a Posteriori* Estimation

In this section, we will give the details about how to determine the prior and likelihood for MAP estimation.

A. Choosing the Prior

Although many works assume every node has the same prior probability to be the source [8], [27], [35], Comin and da Fontoura Costa [29] have shown that the source node tends to have higher centrality measurement values. Therefore, we choose some effective centralities as the prior knowledge, which can ensemble other methods with ours naturally. Let us take the RC as an example to show how to achieve this idea

$$P(v^* = v) = \frac{R(v, G_I)}{\sum_{u\in V_I} R(u, G_I)} \propto R(v, G_I). \tag{4}$$

That means the prior $P(v^* = v)$ is proportional to the RC. For general graphs, we apply the following method used by Shah and Zaman [8] to compute $R(v, G_I)$,

$$R(v, G_I) \approx R\big(v, T_{bfs}(v)\big) \tag{5}$$

where $T_{bfs}(v)$ is a breadth-first search spanning tree of G_I starting with v. It uses the RC in $T_{bfs}(v)$ to approximate $R(v, G_I)$, and $R(v, T_{bfs}(v))$ can be computed by (1).

B. Deriving the Likelihood

Formally, let $G_s(U)$, $N(U)$, and $d(U)$ be the spanning graph, neighbors, and degree of a node set U, respectively. A spanning graph of a node set consists of these nodes and interedges among them. Note that $G_s(\sigma)$ is the spanning graph of nodes appearing in the permitted permutation σ. Let

$$\Omega(v, G_I) = \{\sigma|\sigma(1) = v, G_s(\sigma) = G_I\} \tag{6}$$

denote the set of permitted permutations each of which starts with v and could span the observed infected subgraph G_I. Let

$$E(v, U) = \{(v, u)|(v, u) \in E, u \in U\} \tag{7}$$

be the set of bridging edges between v and the node set U in G. For example, in Fig. 1, $d(\{1, 2\}) = 4$ because there are four edges linked to nodes 1 and 2, $N(\{1, 2\}) = \{3, 4, 5\}$, $G_s(\{1, 2, 4\}) = G_I$, $\Omega(2, G_I) = \{\{2, 1, 4\}, \{2, 4, 1\}\}$, and $E(3, \{1, 2\}) = \{(3, 1), (3, 2)\}$.

Recall that if node v is selected to be the source v^*, the likelihood $P(G_I|v^* = v)$ is the probability to get the observed subgraph graph G_I. On the other hand, every permitted permutation can span G_I according to its definition. That means the likelihood $P(G_I|v^* = v)$ is the sum of probabilities of all permitted permutations which begin with v [8]. Therefore, $P(G_I|v^* = v)$ can be decomposed as follows,

$$P\big(G_I|v^* = v\big) = \sum_{\sigma\in\Omega(v,G_I)} P\big(\sigma|v^* = v\big) \tag{8}$$

where $P(\sigma|v^* = v)$ is the probability to observe a permitted permutation σ give $v^* = v$. In the following, we will show how to derive $P(\sigma|v^* = v)$ in detail.

According to the chain rule in probability theory, $P(\sigma|v^* = v)$ can also be decomposed into the product of many conditional probabilities as follows:

$$\begin{aligned} P\big(\sigma|v^* = v\big) = P(\sigma(2)|\sigma(1) = v)P(\sigma(3)|\sigma(1, 2)) \cdots \\ P(\sigma(N)|\sigma(1, \ldots, N-1)) \end{aligned} \tag{9}$$

where $P(\sigma(i)|\sigma(1, \ldots, i-1))$ $(2 \le i \le N)$ is the probability that $\sigma(i)$ is the ith node to be infected after $\sigma(1, \ldots, i-1)$ are infected.

Recall that the information diffusion along edges are independent under SI. Thus, when all nodes in $\sigma(1, \ldots, i-1)$ are infected, the probability $w_{u|\sigma(1,\ldots,i-1)}$ that node $u \in N(\sigma(1, \ldots, i-1))$ can receive the information in the next round, is determined by

$$w_{u|\sigma(1,\ldots,i-1)} = 1 - \prod_{e\in E(\sigma(i),\sigma(1,\ldots,i-1))} (1 - a_e) \tag{10}$$

where a_e is the information propagation probability along the corresponding edge e. The second term in the right side corresponds to the probability that node u is not infected by all of its active in-neighbors during the next round. We call $w_{u|\sigma(1,\ldots,i-1)}$ as the *effective propagation probability* from $\sigma(1, \ldots, i-1)$ to u, which generalizes the propagation probability from a set of nodes to a single node, and thus makes our method also suitable for weighted graphs with loops. When there is only one edge between $\sigma(i)$ and

$\sigma(1,\ldots,i-1)$, i.e., $E(\sigma(i),\sigma(1,\ldots,i-1)) = \{e\}$, (10) changes into $w_{u|\sigma(1,\ldots,i-1)} = a_e$.

Intuitively, the more probable a susceptible node receives information from $\sigma(1,\ldots,i-1)$, the more likely it will be the next to be infected. Therefore, for an infection sequence or permitted permutation $\sigma \in \Omega(v, G_I)$, the probability $P(\sigma(i) = u|\sigma(1,\ldots,i-1))$ $(2 \le i \le N)$ that node $u \in N(\sigma(1,\ldots,i-1))$ is selected to be the ith infected node, can be defined by

$$P(\sigma(i) = u|\sigma(1,\ldots,i-1)) = \frac{w_{u|\sigma(1,\ldots,i-1)}}{\sum_{v\in N(\sigma(1,\ldots,i-1))} w_{v|\sigma(1,\ldots,i-1)}}. \quad (11)$$

The denominator is a normalization constant, which ensures that the conditional probability on the left-hand side is valid and adds up to one over all values of u. We can see that $P(\sigma(i) = u|\sigma(1,\ldots,i-1))$ is proportional to the effective propagation probability $w_{u|\sigma(1,\ldots,i-1)}$. It means that all the adjacent nodes of $\sigma(1,\ldots,i-1)$ (not only including the infected) should be processed by (10) to get the effective propagation probability, due to the existence of circles in general graphs.

Therefore, if substituting (11) into (9), we have the following probability for any permitted permutation σ:

$$P(\sigma|v^* = v) = P(\sigma(2)|\sigma(1))\cdots P(\sigma(N)|\sigma(1,\ldots,N-1)) = \prod_{i=2}^{N} \frac{w_{\sigma(i)|\sigma(1,\ldots,i-1)}}{\sum_{v\in N(\sigma(1,\ldots,i-1))} w_{v|\sigma(1,\ldots,i-1)}}. \quad (12)$$

Accordingly, we can expand $P(G_I|v^* = v)$ with the aforementioned results as follows:

$$P(G_I|v^* = v) = \sum_{\sigma\in\Omega(v,G_I)} P(\sigma|v^* = v) = \sum_{\sigma\in\Omega(v,G_I)} \prod_{i=2}^{N} \frac{w_{\sigma(i)|\sigma(1,\ldots,i-1)}}{\sum_{v\in N(\sigma(1,\ldots,i-1))} w_{v|\sigma(1,\ldots,i-1)}}. \quad (13)$$

After substituting (4) and (13) into (2), we obtain the following formation of the posterior probability:

$$P(v^* = v|G_I) \propto R(v, G_I) \times \sum_{\sigma\in\Omega(v,G_I)} \prod_{i=2}^{N} \frac{w_{\sigma(i)|\sigma(1,\ldots,i-1)}}{\sum_{v\in N(\sigma(1,\ldots,i-1))} w_{v|\sigma(1,\ldots,i-1)}}. \quad (14)$$

It is clear that this method indeed considers the states of both infected and susceptible nodes, and computes the probability for every permitted permutation from the global perspective. Note that, if choosing other priors, we should revise the above equation correspondingly.

In fact, when all nodes are infected (i.e., $G_I = G$), the MAP estimator defined by the above equation degenerates into the MLE in [8]. Because every infected node can try to infect its neighbors in each time interval until successful under the SI model. Thus if $G_I = G$ and v is the source, the information must follow one permitted permutation in $\Omega(v, G_I)$ to spread such that the sum, $P(G_I|v^* = v)$, in (8) equals to 1. In other words, when $G_I = G$, every node could be the source and infect all the others as long as the information spreads for a sufficiently long period of time. At this moment, we can only use the prior knowledge depicted by other methods in Section IV-A to distinguish these nodes.

C. Two Special Cases of Our Method

In the following part, we will describe two special cases of the above method to show its relationship with our preliminary work [28] and RC [7], [8].

1) Special Case 1: Let $\mathbf{E}^i_\sigma = E(\sigma(i),\ \sigma(1,\ldots,i-1))$ for the convenience of derivation. After expanding the second term in the bottom line of (10), the effective propagation probability $w_{u|\sigma(1,\ldots,i-1)}$ changes into the following:

$$\begin{aligned} w_{u|\sigma(1,\ldots,i-1)} &= 1 - \prod_{e\in\mathbf{E}^i_\sigma}(1-a_e) \\ &= 1 - \left[1 - \sum_{e\in\mathbf{E}^i_\sigma} a_e + \sum_{e_j,e_k\in\mathbf{E}^i_\sigma,\ j<k} \sum a_{e_j}\cdot a_{e_k} + \cdots + (-1)^{|\mathbf{E}^i_\sigma|} \prod_{e\in\mathbf{E}^i_\sigma} a_e\right] \\ &= \sum_{e\in\mathbf{E}^i_\sigma} a_e - \sum_{e_j,\ e_k\in\mathbf{E}^i_\sigma,\ j<k} \sum a_{e_j}\cdot a_{e_k} + \cdots + (-1)^{|\mathbf{E}^i_\sigma|} \prod_{e\in\mathbf{E}^i_\sigma} a_e. \end{aligned} \quad (15)$$

When

$$\sum_{e\in\mathbf{E}^i_\sigma} a_e \gg \sum_{e_j,\ e_k\in\mathbf{E}^i_\sigma,\ j<k} \sum a_{e_j}\cdot a_{e_k} + \cdots + (-1)^{|\mathbf{E}^i_\sigma|} \prod_{e\in\mathbf{E}^i_\sigma} a_e \quad (16)$$

we can drop out the second term and get an approximation for $w_{u|\sigma(1,\ldots,i-1)}$

$$w_{u|\sigma(1,\ldots,i-1)} \approx \sum_{e\in E(\sigma(i),\ \sigma(1,\ldots,i-1))} a_e. \quad (17)$$

This approximate result has been used in our preliminary work [28].

Additionally, according to [47], the condition in (16) equals to that

$$a_e \ll 1\ \forall e \in E(\sigma(i),\ \sigma(1,\ldots,i-1)) \quad (18)$$

and the size of $E(\sigma(i),\ \sigma(1,\ldots,i-1)$ is small. On the other hand, in real-world social networks, the propagation probabilities among users are very small. For example, according to [2], the probability that a Facebook user will share a URL is less than 0.02 even when there are five of his friends who have shared it before. Thus the approximation in (17) indeed makes sense for real-world social networks.

This special case shows that our preliminary work is generalized by this paper. Their relationship can be established if the condition in (16) or (18) is valid.

Algorithm 1: BFSA

input : G - the undirected graph
G_I - the infected subgrpah
output: $\hat{v}$ - the detected source
1 **for** $v \in V_I$ **do**
2 $\quad$ P$(G_I|v^* = v) = 0$;
3 $\quad$ span the breadth first search spanning tree $T_{bfs}(v)$;
4 $\quad$ calculate the prior P$(v^* = v)$ by Eq. (5);
5 σ = an array of V_I;
6 getLikelihoodByBFSA(σ,1,N);
7 select $\hat{v}$ by Eq. (3);
8 return $\hat{v}$;

Algorithm 2: getLikelihoodByBFSA(σ, p, q)

input : σ - the infected node array
p - the starting index
q - the end index
1 **if** $p == q$ **then**
2 $\quad$ get P$(\sigma|v^* = v)$ by Eq. (12);
3 $\quad$ P$(G_I|v^* = v)$ += P$(\sigma|v^* = v)$;
4 **else**
5 $\quad$ **for** $i = p; i \le q; ++i$ **do**
6 $\quad\quad$ **if** $\sigma[p]$ *is a node of the subtree rooted at* $\sigma[i]$ *of* $T_{bfs}(\sigma[1])$ **then**
7 $\quad\quad\quad$ swap($\sigma[p], \sigma[i]$);
8 $\quad\quad\quad$ getLikelihoodByBFSA($\sigma, p+1, q$);
9 $\quad\quad\quad$ swap($\sigma[p], \sigma[i]$);

2) Special Case 2: When $a_{ij} \equiv \lambda$ and the graph is a tree, every node has only one path to connect with others. This means $|E(\sigma(j), \sigma(1, \ldots, j-1))| = 1$. Thus (12) becomes the following succinct form:

$$P(\sigma|v^* = v) = \prod_{i=2}^{N} \frac{\lambda}{\lambda \cdot d(\sigma(1, \ldots, i-1))} = \prod_{i=2}^{N} \frac{1}{d(\sigma(1, \ldots, i-1))} \tag{19}$$

where $d(\sigma(1, \ldots, i-1))$ is the degree of $\sigma(1, \ldots, i-1)$, and can be determined by

$$(\sigma(1, \ldots, i-1)) = \sum_{j=1}^{j=i-1} (d(\sigma(j)) - 2)) \tag{20}$$

because adding each node $\sigma(j)$ will contribute $d(\sigma(j)) - 2$ new edges. Equation (19) has been used for deriving the RC by Shah and Zaman [7], [8]. They noted for regular trees where every node has the same degree, (19) is identical for each permitted permutation. Thus, we can figure out the result in Section III-B.

V. Approximate MAP Estimators

If enumerating all the permitted permutations, (14) tells that we can have the theoretically optimal MAP estimator. However, (1) has shown the factorial complexity of the number of permitted permutations even for general trees, not to mention for general graphs. We will show how to get the approximate estimators to speed up the detection from the perspective of likelihood approximation.

A. Brute Force Search Approximation

BFSA tries to enumerate all the permitted permutations to derive the MAP estimator. Algorithm 1 shows the pseudo codes. Specifically, it first initializes the likelihood, gets the breadth-first search spanning tree and the prior $R(v, G_I)$ for every infected node, from lines 1 to 4. Then it calls Algorithm 2 to generate permitted permutations and obtains the likelihood. Finally, it selects the detected source according to (3).

Algorithm 2 extends heap's permutation generating algorithm [48]. During the generating process, it prunes the searching branches that do not follow the topological constrain by line 6. That means if $\sigma[p]$ is a descendant of $\sigma[i]$ in $T_{bfs}(\sigma[1])$, we can swap them to get a new permitted permutation.

BFSA can output the optimal MAP estimator for general trees, but may miss some permitted permutations for graphs with loops. Nevertheless, we will show the effectiveness of BFSA for source detection in the experiment part. However, its time complexity is exorbitantly high. To further improve the efficiency, we propose the following approximate estimator.

B. Greedy Search Bound Approximation

The basic idea of GSBA is to find the upper bound of the probability $P(\sigma|v^* = v)$ of permitted permutations starting with the same node, and then to reduce the computational complexity of computing the likelihood $P(G_I|v^* = v)$.

Recall that $\Omega(v, G_I)$ is the set of permitted permutations beginning with v, and $P(G_I|v^* = v)$ can be decomposed as the sum of probabilities of all permitted permutations in $\Omega(v, G_I)$. Among $\Omega(v, G_I)$, there must be a permutation σ such that its probability $P(\sigma|v^* = v)$ in (8) is maximal, which is denoted as $\sigma^v_{\max}$. Therefore, for $\sigma \in \Omega(v, G_I)$, we have

$$P(\sigma|v^* = v) \le P\left(\sigma^v_{\max}|v^* = v\right) = \prod_{i=2}^{N} \frac{w_{\sigma^v_{\max}(i)|\sigma^v_{\max}(1,\ldots,i-1)}}{\sum_{v \in N\left(\sigma^v_{\max}(1,\ldots,i-1)\right)} w_{v|\sigma^v_{\max}(1,\ldots,i-1)}}. \tag{21}$$

More importantly, if we adopt the permitted permutation generation method in Algorithm 2, there exists the following approximation:

$$|\Omega(v, G_I)| = R(v, G_I) \tag{22}$$

where $R(v, G_I)$ is the RC, computed by (5). Note that the above is exact when the graph is a tree. Combining (21) and (22)

with (14), we derive an upper bound of the posterior $P(v^* = v|G_I)$ like

$$P\big(v^* = v|G_I\big) \leq R^2(v, G_I)P\big(\sigma^v_{\max}|v^* = v\big). \tag{23}$$

Accordingly, if we exploit this upper bound to approximate $P(v^* = v|G_I)$, the MAP estimator of (3) changes into

$$\hat{v} = \arg\max_{v\in G_I} R^2(v, G_I)P\big(\sigma^v_{\max}|v^* = v\big). \tag{24}$$

We denote it as MAP upper-bound (MAP-ub).

Now the only issue left is how to find $\sigma^v_{\max}$ and determine the upper bound. When $\sigma(1, \ldots, i-1)$ is given, if exploiting the greedy search strategy to select h from the neighbors of $\sigma(1, \ldots, i-1)$ to be the ith infected node such that $([w_{h|\sigma(1,\ldots,i-1)}]/[\sum_{v\in N(\sigma(1,\ldots,i-1))} w_{v|\sigma(1,\ldots,i-1)}])$ is maximal, we can get a permitted permutation σ^v_{gs}. If we set σ^v_{gs} as a *surrogate* of $\sigma^v_{\max}$, and the estimator of (24) becomes

$$\hat{v} = \arg\max_{v\in G_I} R^2(v, G_I)P\Big(\sigma^v_{gs}|v^* = v\Big). \tag{25}$$

Note that when choosing other priors, (23)–(25) should be modified accordingly.

So far, we have derived the final GSBA. The pseudo codes of GSBA is mostly like BFSA's in Algorithm 1, except for lines 6 and 7. GSBA will approximate the likelihood by Algorithm 3 and detect the source according to the estimator in (25). Specifically, lines 2–7 in Algorithm 3 initialize the program to find σ^v_{gs}. Line 9 selects the first node u of Q with the maximal effective propagation probability $w[u]$ to be the next infected node. If $w[h]$ has been computed, line 16 exploits the old value to update $w[h]$ according to (10), which can avoid traversing the adjacent nodes of h when updating $w[h]$. Otherwise, h has not been visited and there must be only one edge e_{uh} between h and the already selected infected-nodes, thus line 18 uses the propagation probability a_{uh} to update $w[h]$. When inserting a new node into the queue Q in line 20, we use the concept of insertion sort to ensure that the first node of Q has the maximal probability to be the next infected node. The computational complexity of Algorithm 3 is $O(c \cdot N^2)$, where c is the average node degree, and we will show its effectiveness in the experiment. GSBA is a tradeoff between effectiveness and efficiency to approximate the likelihood. We leave it as future works to explore other algorithms to find $\sigma^v_{\max}$ more accurate such as dynamic programming.

Algorithm 3: getLikelihoodByGSBA(G_I)

input : $G(V, E, A)$ - the whole graph with $A = [a_{ij}]$
$G_I(V_I, E_I)$ - the infected subgraph

1 **for** $v \in V_I$ **do**
2 $\quad \mathrm{P}(\sigma^v_{gs}|v^* = v) = 1$;
3 $\quad Q =$ an empty queue storing nodes to be selected into σ^v_{gs};
4 $\quad S =$ an empty set storing nodes selected already;
5 $\quad w =$ an empty hash-table storing the effective propagation probabilities in Eq. (10);
6 $\quad$ add v into Q;
7 $\quad w[v] = 1$;
8 $\quad$ **while** $S \neq V_I$ **do**
9 $\quad\quad u =$ the first node of Q;
10 $\quad\quad \mathrm{P}(\sigma^v_{gs}|v^* = v) = \mathrm{P}(\sigma^v_{gs}|v^* = v) \cdot \frac{w[u]}{\sum_{j\in Q} w[j]}$;
11 $\quad\quad$ add u into S; # i.e., select u into σ^v_{gs}
12 $\quad\quad$ remove u from Q;
13 $\quad\quad$ **for** $h \in$ *neighbors of* u *in* G **do**
14 $\quad\quad\quad$ **if** h *is not in* S **then**
15 $\quad\quad\quad\quad$ **if** $w[h]$ *has been computed* **then**
16 $\quad\quad\quad\quad\quad w[h] = 1 - (1 - w[h]) * (1 - a_{uh})$;
17 $\quad\quad\quad\quad$ **else**
18 $\quad\quad\quad\quad\quad w[h] = a_{uh}$;
19 $\quad\quad\quad\quad$ **if** $h \in V_I$ **then**
20 $\quad\quad\quad\quad\quad$ insert h into Q according to the descending order of $w[h]$;
21 $\quad\quad\quad$
22 $\quad\quad$

VI. Experiment

In this section, we first present some experimental settings, including datasets, baselines and evaluation measures. Then we explore how different kinds of *a priori* affect our method, and compare our methods with baselines for single information source detection on different networks.

A. Datasets

Our datasets are simulations about information diffusion on four networks, namely SCALE-FREE, POWER-GRID, WIKI-VOTE, and CA-ASTROPH. This kind of datasets are widely used in [8] and [27]. Specifically, a scale-free network is a connected graph, whose degree distribution nearly follows a power law. We generate it with the Barabási–Albert [49] model. POWER-GRID [50] is an undirected network containing information about the power grid of Western States of the USA.[1] WIKI-VOTE [51] is a who-voted-who graph on Wikipedia,[2] and CA-ASTROPH [52] is a collaboration network of Astro Physics category on arXiv.[3] WIKI-VOTE and CA-ASTROPH are downloaded from *Stanford Network Analysis Project* [53]. We assume that node u and v have an undirected edge if there is an edge between them in the original network, and the information propagation probability along this edge is randomly sampled from the uniform distribution [0, 1]. As we said in Section III-A, an infected subgraph should be connected. So we remove the disconnected nodes and keep the maximal connected component. After this filtering, their statistical information is listed in Table II. We can see the edges of POWER-GRID are extremely sparse.

To simulate the information diffusion on a graph G, we adopt the following two strategies to select the source node.

1) *Random Test:* We randomly select a node from G as the infection source.

[1] http://konect.uni-koblenz.de/networks/opsahl-powergrid
[2] http://www.wikipedia.org/
[3] https://arxiv.org/

TABLE II
STATISTICS OF OUR DATASETS

Network	# of Nodes	# of Edges	Edge Density
SCALE-FREE	500	996	0.798%
POWER-GRID	4,941	6,594	0.054%
WIKI-VOTE	7,066	100,736	0.404%
CA-ASTROPH	17,903	197,031	0.123%

2) *Full Test:* We select the source node by turns such that each node has a chance to be the source, which is more suitable for real scenarios.

After selecting a source node, we start to run the SI model until the number of infected nodes equals to a given value N. Repeating this process, and finally we have M infected subgraphs with a given size for each network. Specifically, for random test, let $M = 100$, and for full test, $M = |V|$, where $|V|$ is the node number of G.

B. Baselines and Evaluation Measures

To validate our methods, namely BFSA and GSBA, we compare them with the following methods which are suitable for the SI model and adopt the same settings with ours.

1) *Distance Center (DC):* It selects an infected node which has the minimal distance centrality as the source. Distance centrality is a sum of the shortest distance from a node to any others [8].
2) *JC:* It selects an infected node which minimizes the maximum distance to others as the source [24].
3) *RC:* It selects an infected node which has the maximal RC as the source. RC is defined as (1) [8].
4) *Reverse Infection (RI):* The algorithm lets every infected node broadcast its identity to the neighbors. Once a node receives a new identity, it will record the arriving time, and then broadcasts the identity to its neighbors. At last, the node which has received all the identities and the sum of their arriving times is minimal, is selected as the source [6].
5) *DMP:* The algorithm uses DMP equations to estimate the marginal probability of a given node to be in a given state, and then exploits a mean-field-type approach to approximate the likelihood. It selects an infected node corresponding to the maximal likelihood as the source [27].
6) *Dynamic Importance (DI):* This is a method of the spectral family. It selects an infected node which has the maximal reduction of the largest eigenvalue of the adjacent matrix after it is removed from the network, as the source [17].
7) *GSBA$^-$:* It is from our preliminary work [28], which is designed for unweighted graphs and approximates the likelihood of a permitted permutation like (25).

Note that Lokhov *et al.* [27] have shown DMP is the state of the art among these methods. We implement all the methods based on *NetworkX*,[4] which is a python package for manipulations on graphs. Their codes can be found from our Github.[5] Indeed, full test can validate the stability of a method more accurately. But we only use full test to compare DC, JC, RC, RI, and DI with GSBA in this paper, because others are too time-consuming. Their time complexities are listed in Table III. Our experiments are conducted on a personal computer with a 3.1-GHz i5-3450 CPU and 8-GB RAM.

TABLE III
TIME COMPLEXITIES OF DIFFERENT METHODS, WHERE c IS THE AVERAGE NODE DEGREE, AND d_I IS THE DIAMETER OF THE INFECTED SUBGRAPH G_I

Methods	Time Complexities
DC	$O(\|E_I\| \cdot \|V_I\|)$
JC	$O(\|E_I\| \cdot \|V_I\|)$
RC	$O(\|V_I\|^2)$
RI	$O(c \cdot d_I \cdot \|V_I\|)$
DMP	$O(c \cdot d_I \cdot \|V_I\| \cdot \|V\|)$
DI	$O(\|V_I\|^3)$
BFSA	$O(c \cdot \|V_I\| \cdot \|V_I\|!)$
GSBA	$O(c \cdot \|V_I\|^2)$

We apply the following three widely used measures to evaluate the performances of different methods [8], [27], [32]. Let v^* be the real source and $\hat{v}$ be the detected source.

1) *Detection Rate:* It is defined as

$$\text{Detection Rate} = \frac{M_T}{M} \tag{26}$$

where M is the running number of tests and M_T is the number of tests which detect the source correctly.
2) *Detection Error:* It is the average shortest topological distance between v^* and $\hat{v}$.
3) *Normalized Ranking:* We first rank the infected nodes in descending order by the probability to be the source computed by a method. Normalized ranking is defined as

$$\text{Normalized Ranking} = \frac{\text{Ranking}(\text{v}^*) - 1}{N} \tag{27}$$

where N is the number of infected nodes and *Ranking*(v^*) is the ranking of v^* in the sorted list.

To some degree, *detection rate* can reflect the detection accuracy of a method, and *detection error* shows how far the detected source is away from the real source on the network, while *normalized ranking* can validate the precision that a method sorts the real source. Note that the larger *detection rate* is, the better performance the corresponding method achieves, but *detection error* and *normalized ranking* are opposite. In the following, we will show the performances of these methods under different settings.

C. Effect of Different Priors

As said in Section III-C, based on MAP estimation, our method can be easily integrated with other methods by involving them as *a priori*. Therefore, this section is to explore the effect of different kinds of *a priori* on our method and select the appropriate one for GSBA. Specifically, we first select three simple but effective baselines (i.e., RC, DC, and JC) as

[4]https://networkx.github.io/

[5]https://github.com/biaochangb/research/tree/master/SourceDetection

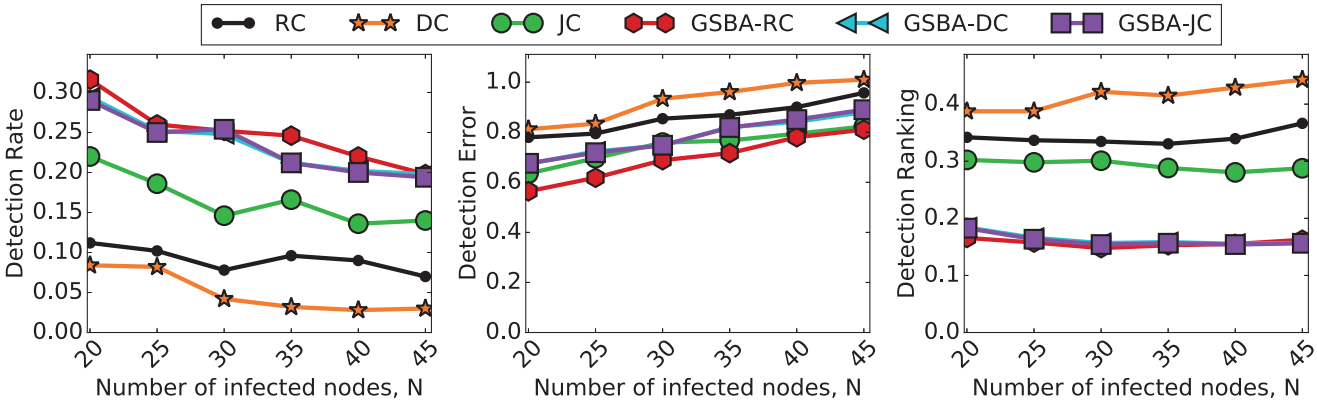


Fig. 2. GSBA with different kinds of *a priori* on the SCALE-FREE network.

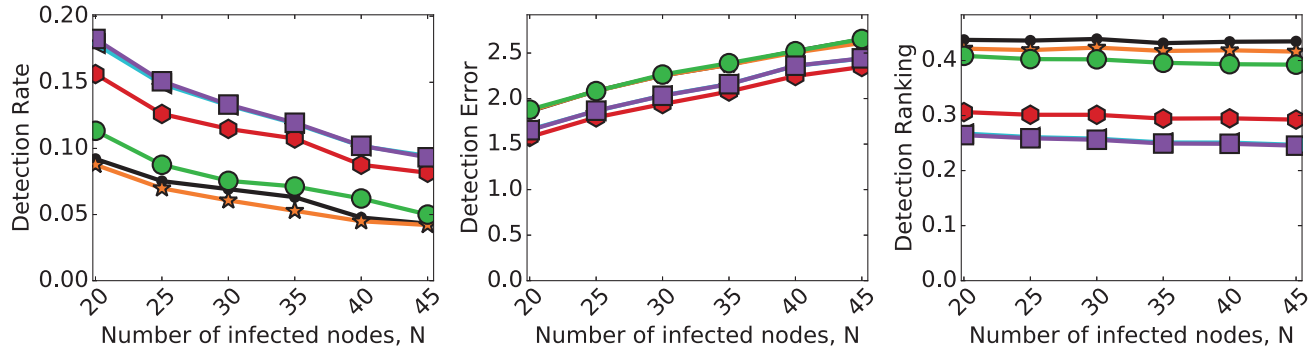


Fig. 3. GSBA with different kinds of *a priori* on the POWER-GRID network.

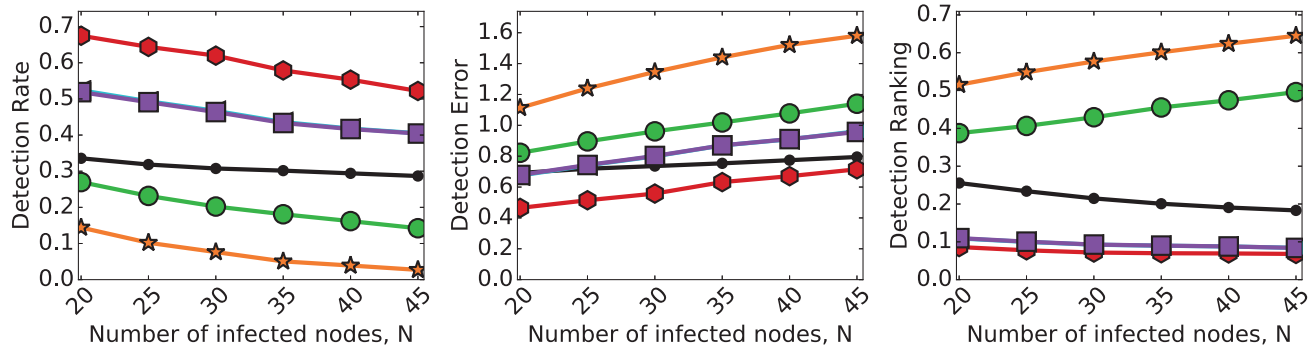


Fig. 4. GSBA with different kinds of *a priori* on the WIKI-VOTE network.

the prior for GSBA, respectively, and have three corresponding instances GSBA-RC, GSBA-DC, and GSBA-JC. Then we run the full test on SCALE-FREE, POWER-GRID, and WIKI-VOTE networks. The number of infected nodes ranges from 20 to 45, which can help us quickly find the appropriate priori for following comparisons. Figs. 2–4 show the evaluation results of source detection, and we can have several obvious conclusions.

First, involving other methods into our model as *a priori* can significantly improve their performances with respect to all of the three measures: 1) *detection rate*; 2) *detection error*; and 3) *normalized ranking*, especially on WIKI-VOTE which has a larger graph size. This proves the reasonableness of our solution based on MAP in Section III-C.

Second, different kinds of *a priori* indeed make GSBA have varying detection performances. Note that GSBA-DC and GSBA-JC have similar performances on these three networks with respect to all of three measures. From Fig. 4, we can find that GSBA-RC performs worse than GSBA-DC and GSBA-JC. The main reason we think is that RC only counts the number of permitted permutations but does not consider the heterogeneity of edge weights when detecting. This also shows the necessity of extending our previous work [28] for weighted graphs. However, GSBA-RC performs better in Figs. 2 and 4 on denser graphs, SCALE-FREE and WIKI-VOTE.

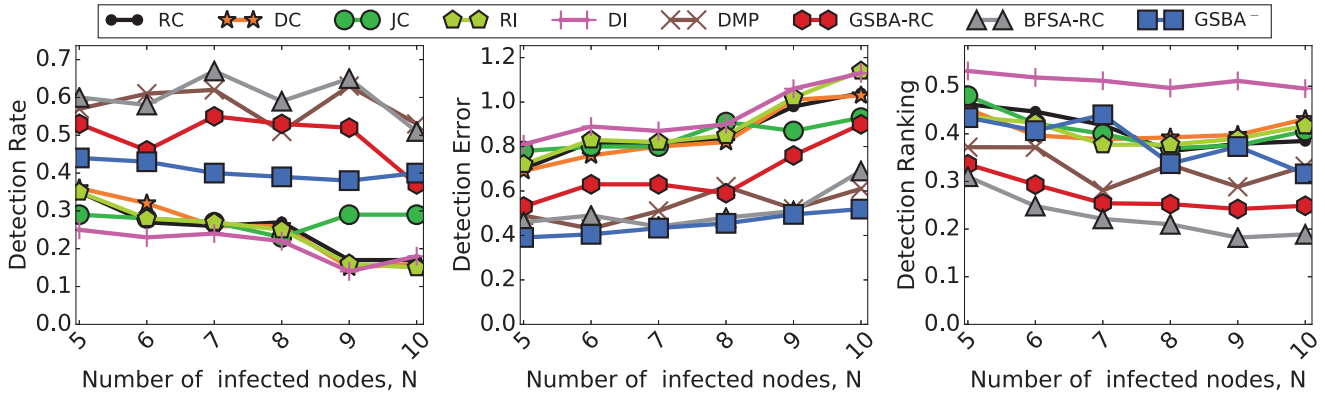


Fig. 5. Random test performances of different methods on the SCALE-FREE network.

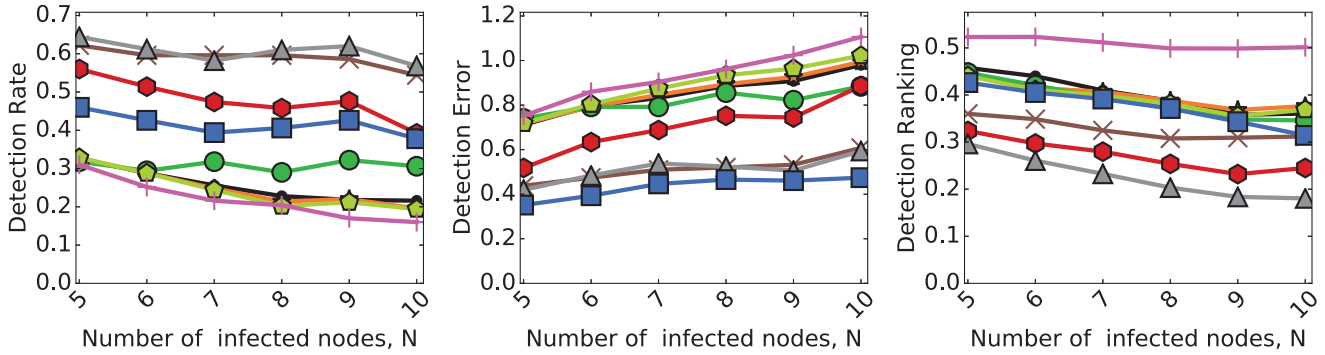


Fig. 6. Full test performances of different methods on the SCALE-FREE network.

Therefore, in the following parts, we will choose RC as the prior of GSBA except for specific explanations, because GSBA-RC is more effective than others in most cases over these three types of networks.

D. Results on the Scale-Free Network

In this section, we compare our methods, namely GSBA-RC and BFSA-RC, with other baselines including RC, DC, JC, RI, DI, DMP, and GSBA$^-$ on the SCALE-FREE network. BFSA-RC is the method described in Algorithm 1 with RC as the prior. GSBA$^-$ is our preliminary work [28] with RC as the prior. However, Table III has shown the high time complexities of DMP and BFSA-JC. Besides, when evaluating the source detection performance, it needs to repeat M times for each value of N to avoid stochastic errors, which corresponds to our random test (i.e., $M = 100$) and full test (i.e., $M = |V|$) strategies. This further restricts the application of DMP and BFSA. Therefore, the number of observed infected nodes, N, ranges from 5 to 10 for comparison. Results of random and full tests are shown in Figs. 5 and 6, respectively. We can have the following observations.

First, for both of random test and full test, our methods (GSBA-RC and BFSA-RC) achieve better performance than RC under all the three measures. The reason is that our methods consider uninfected nodes and the heterogeneity of edge weights when inferring the probabilities of permitted permutations for general graphs. This proves once more that uninfected nodes are also helpful for detecting the information source.

Second, Figs. 5 and 6 clearly show GSBA-RC, BFSA-RC, and DMP outperform other methods no matter under random or full test. Additionally, BFSA-RC performs nearly the same as DMP with respect to *detection rate* and *detection error*, even better in some cases such as $N = 9$. But GSBA-RC and BFSA-RC achieve much smaller normalized ranking than DMP. This means our methods can sort the real source more accurately. Our preliminary work, GSBA$^-$, achieves nearly the best *detection error*, but its performances about *detection rate* and normalized ranking are unsatisfactory. The above indicates the feasibility of MAP-ub and the greedy search strategy in Section V-B. In other words, after selecting the prior, determining the likelihood like Section IV can improve the detection performance drastically.

Finally, we display the average running times of different methods on the SCALE-FREE network in Fig. 7(a). We can see their trends are similar with the increasing number of infected nodes, but the running time of DMP is larger than GSBA-RC's by more than an order of magnitude. The reason is that DMP has to iteratively compute the DMP equations for each node in the whole network including all uninfected nodes, but GSBA-RC only needs infected nodes and their neighbors. Therefore, DMP is not applicable for other datasets with larger scales. Other centralities such as RC, DC, and JC are faster, but their detection performances are unsatisfactory as said before.

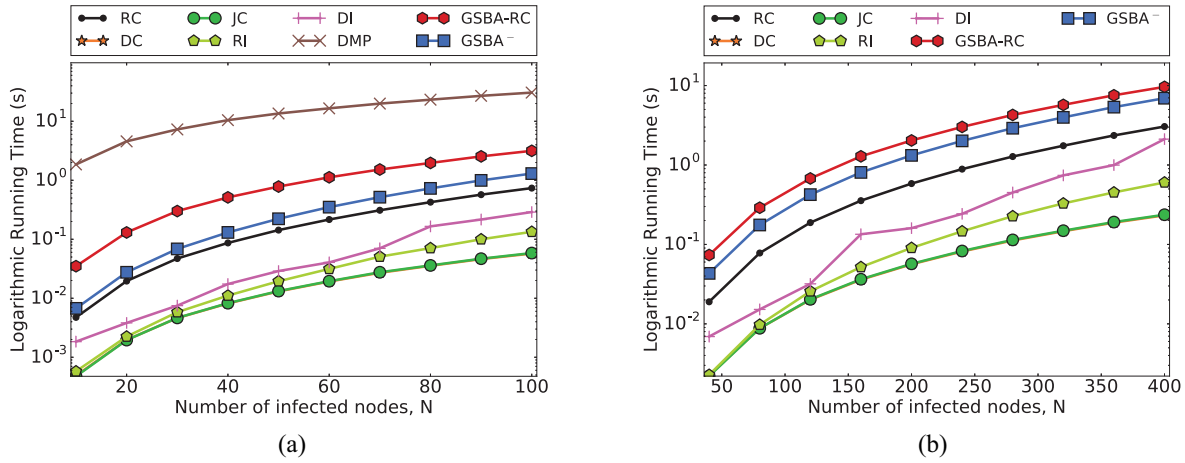


Fig. 7. Average running times of different methods on (a) SCALE-FREE and (b) POWER-GRID.

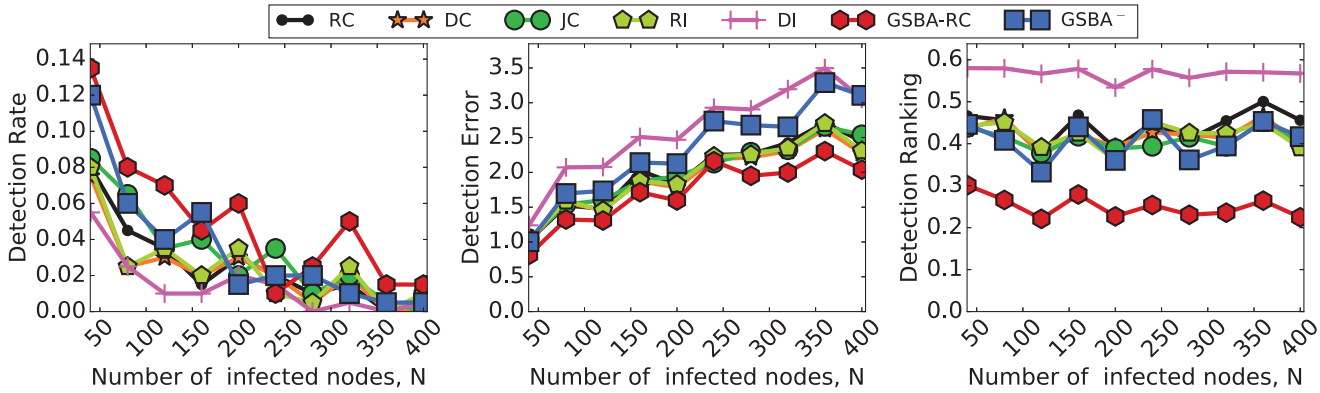


Fig. 8. Random test performances of different methods on the POWER-GRID network.

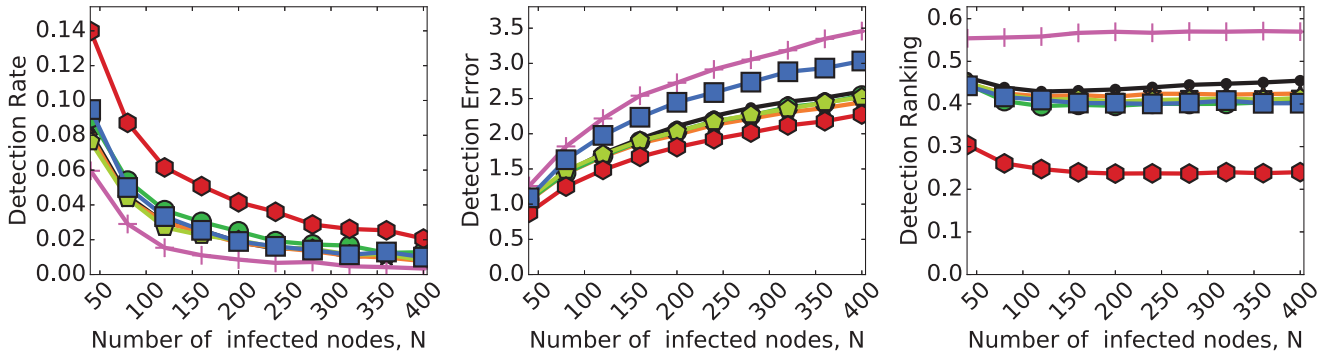


Fig. 9. Full test performances of different methods on the POWER-GRID network.

To sum up, if combined with an appropriate priori, BFSA can achieve better performance than DMP with respect to Normalized Ranking on the SCALE-FREE network, and GSBA is a tradeoff between effectiveness and efficiency. These results prove the feasibility of our estimation approximations in Section V.

E. Results on Other Networks

In the following, we will first display the experimental results for larger sizes N of infected subgraphs on the POWER-GRID Network. But as mentioned before, DMP and BFSA are time-consuming and not applicable for large-scale networks, and BFSA behaves similarly to DMP on scale-free networks. Therefore, we only compare RC, DC, JC, RI, DI, and GSBA$^-$ with GSBA-RC, and the number of observed infected nodes N ranges from 50 to 400 due to the time-consuming repeating detection evaluations for each value of N to avoid stochastic errors. Results of random and full tests are shown in Figs. 8 and 9. We can have some interesting findings.

First, the performances of all methods under the strategy of full test are more regular and distinguishable, but the curves

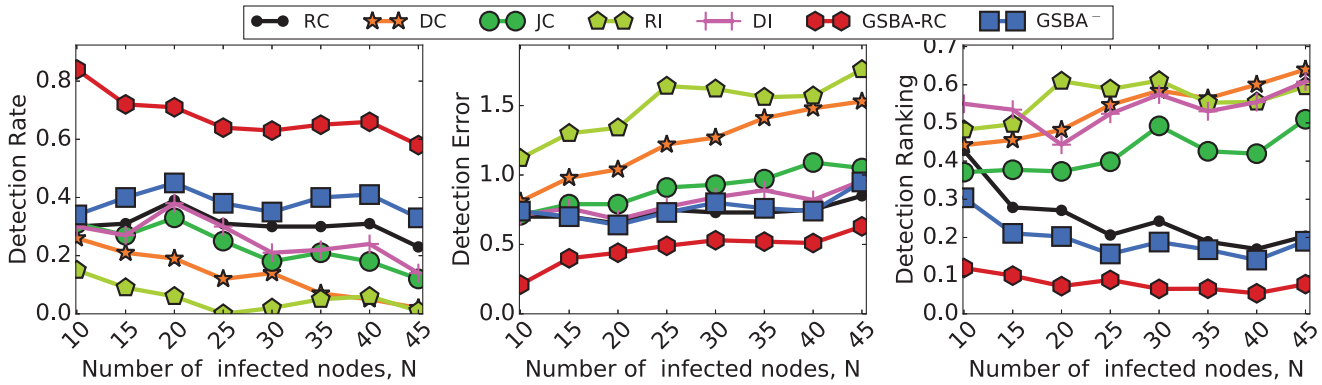


Fig. 10. Random test performances of different methods on the WIKI-VOTE network.

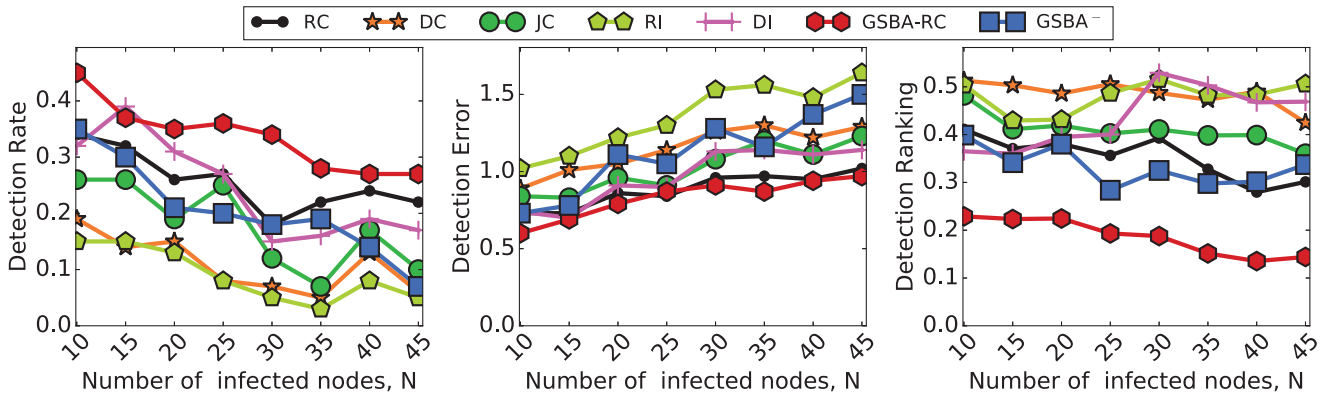


Fig. 11. Random test performances of different methods on the CA-ASTROPH network.

in Fig. 8 shake severely. This shows the necessity of repeating detection for each N to avoid stochastic errors. Second, our GSBA-RC always achieves the best performance for all situations shown in Figs. 8 and 9. RC, DC, JC, RI, and GSBA$^-$ perform similarly in terms of *detection rate* and *normalized ranking* but DI is unsatisfactory. Generally speaking, the results of random test and full test are consistent. Third, *detection rates* of different methods decrease obviously with the increasing number of infected nodes, but our method outperforms others with a big gap in Fig. 9. Forth, *normalized ranking* behaves very steadily and basically keeps invariant as the number of infected nodes are more than 80 in Fig. 9. This is an important property such that we can measure the ability of ranking the real source as higher as possible for a method on small sizes of infected subgraphs. Fifth, the average running times of different methods are shown in Fig. 7(b). GSBA-RC is little slower than our previous method GSBA$^-$, but the improvements of detection performances in Figs. 8 and 9 are significant.

We also repeated the above experiments on WIKI-VOTE and CA-ASTROPH networks. Results of random tests are shown in Figs. 10 and 11. Except for the above similar findings, we have some new ones. First, on the WIKI-VOTE network, RC performs better than DC, JC, RI, and DI for all three measures under random or full test. Let us take $N = 35$ under full test as an example. For POWER-GRID, the average diameter of infected subgraphs G_I is 2.42, and the average ratio of edges to nodes in G_I is 1.34. For WIKI-VOTE, the average diameter and ratio are 2.65 and 3.44, respectively. Indeed, the ratio of a tree is less than 1. Therefore, the infected subgraphs of POWER-GRID are more tree-like. This may explain why RC performs better than JC. Second, on the CA-ASTROPH network, when the number of infected nodes is less than 25, DI performs better than others, except for GSBA-RC. This is quite different from their performances on other three networks. Therefore, the graph structure significantly affects baselines' performances.

To summarize, GSBA-RC always achieves the best performance on these three networks, especially with respect to *detection rate* and *normalized ranking*. The graph topological structures can significantly affect the performances of all methods, but our method can achieve the best if combined with an appropriate priori.

VII. CONCLUSION

In this paper, we revisited the problem of single information source detection in weighted graphs from the perspective of likelihood approximation. After deriving the MAP estimator, we design two approximation approaches to improve the efficiency, namely BFSA and GSBA. Experiments on several networks clearly show the superiority of our methods especially when measured by *normalized ranking*, and the feasibility of likelihood approximation. Additionally,

GSBA is nearly as effective as BFSA, but far more efficient.

Three directions are worth exploring as further study. First, so far we have derived our methods to detect the single information source under the SI model. It is interesting to extend them for multiple information sources detection under other models, such as SIR. The second direction is to explore other more effective approaches to find the upper bound of likelihood, instead of greedy search. We can exploit the Cramer–Rao bound to evaluate their estimation qualities. Third, there may be other methods to approximate the likelihood such as Markov chain Monte Carlo sampling, instead of the upper bound to derive GSBA in Section V-B.

REFERENCES

[1] R. Zafarani, M. A. Abbasi, and H. Liu, *Social Media Mining: An Introduction*. Cambridge, U.K.: Cambridge Univ. Press, 2014.

[2] E. Bakshy, I. Rosenn, C. Marlow, and L. Adamic, "The role of social networks in information diffusion," in *Proc. 21st Int. Conf. World Wide Web (WWW)*, Lyon, France, 2012, pp. 519–528.

[3] Q. Liu *et al.*, "Influential seed items recommendation," in *Proc. 6th ACM Conf. Recommender Syst. (RecSys)*, Dublin, Ireland, 2012, pp. 245–248.

[4] Y. Li, P. Luo, and P. Pin, "Utility-based model for characterizing the evolution of social networks," *IEEE Trans. Syst., Man, Cybern., Syst.*, to be published, doi: 10.1109/TSMC.2017.2690827.

[5] Q. Liu *et al.*, "An influence propagation view of PageRank," *ACM Trans. Knowl. Disc. Data*, vol. 11, no. 3, pp. 1–30, Apr. 2017. [Online]. Available: http://doi.acm.org/10.1145/3046941

[6] K. Zhu and L. Ying, "Information source detection in the sir model: A sample-path-based approach," *IEEE/ACM Trans. Netw.*, vol. 24, no. 1, pp. 408–421, Feb. 2016.

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

[8] D. Shah and T. Zaman, "Detecting sources of computer viruses in networks: Theory and experiment," in *Proc. ACM SIGMETRICS Int. Conf. Meas. Model. Comput. Syst. (SIGMETRICS)*, New York, NY, USA, 2010, pp. 203–214.

[9] D. Kempe, J. Kleinberg, and É. Tardos, "Maximizing the spread of influence through a social network," in *Proc. 9th ACM SIGKDD Int. Conf. Knowl. Disc. Data Min. (KDD)*, Washington, DC, USA, 2003, pp. 137–146.

[10] Q. Liu *et al.*, "Influence maximization over large-scale social networks: A bounded linear approach," in *Proc. 23rd ACM Int. Conf. Inf. Knowl. Manag. (CIKM)*, Shanghai, China, 2014, pp. 171–180.

[11] Z. Wang *et al.*, "Maximizing the coverage of information propagation in social networks," in *Proc. IJCAI*, Buenos Aires, Argentina, 2015, pp. 2104–2110.

[12] K. Kandhway and J. Kuri, "Using node centrality and optimal control to maximize information diffusion in social networks," *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 47, no. 7, pp. 1099–1110, Jul. 2017.

[13] K. Xiao, Q. Liu, C. Liu, and H. Xiong, "Price shock detection with an influence-based model of social attention," *ACM Trans. Manag. Inf. Syst.*, vol. 9, no. 1, pp. 1–21, Feb. 2018.

[14] J. Zhao, Q. Liu, L. Wang, and X. Wang, "Competitiveness maximization on complex networks," *IEEE Trans. Syst., Man, Cybern., Syst.*, to be published, doi: 10.1109/TSMC.2016.2636240.

[15] L. Zhang, K. Xiao, Q. Liu, Y. Tao, and Y. Deng, "Modeling social attention for stock analysis: An influence propagation perspective," in *Proc. IEEE Int. Conf. Data Min. (ICDM)*, Atlantic City, NJ, USA, 2015, pp. 609–618.

[16] T. Xu *et al.*, "Learning to annotate via social interaction analytics," *Knowl. Inf. Syst.*, vol. 41, no. 2, pp. 251–276, Nov. 2014.

[17] V. Fioriti, M. Chinnici, and J. Palomo, "Predicting the sources of an outbreak with a spectral technique," *Appl. Math. Sci.*, vol. 8, no. 135, pp. 6775–6782, 2014, doi: 10.12988/ams.2014.49693.

[18] P. Gundecha, Z. Feng, and H. Liu, "Seeking provenance of information using social media," in *Proc. 22nd ACM Int. Conf. Inf. Knowl. Manag. (CIKM)*, 2013, pp. 1691–1696.

[19] B. A. Prakash, J. Vreeken, and C. Faloutsos, "Spotting culprits in epidemics: How many and which ones?" in *Proc. IEEE 12th Int. Conf. Data Min.*, Brussels, Belgium, Dec. 2012, pp. 11–20.

[20] G. Boracchi, M. Michaelides, and M. Roveri, "A cognitive monitoring system for detecting and isolating contaminants and faults in intelligent buildings," *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 48, no. 3, pp. 433–447, Mar. 2018.

[21] J. L. Iribarren and E. Moro, "Branching dynamics of viral information spreading," *Phys. Rev. E, Stat. Phys. Plasmas Fluids Relat. Interdiscip. Top.*, vol. 84, Oct. 2011, Art. no. 046116.

[22] W. O. Kermack and A. G. McKendrick, "Contributions to the mathematical theory of epidemics. II. The problem of endemicity," *Proc. Roy. Soc. London A Containing Papers Math. Phys. Character*, vol. 138, no. 834, pp. 55–83, 1932.

[23] J. Goldenberg, B. Libai, and E. Muller, "Talk of the network: A complex systems look at the underlying process of word-of-mouth," *Marketing Lett.*, vol. 12, no. 3, pp. 211–223, 2001.

[24] C. Jordan, "Sur les assemblages de lignes," *J. für die reine und angewandte Mathematik*, vol. 9186, no. 70, pp. 185–190, 1869, doi: 10.1515/crll.1869.70.185.

[25] F. Altarelli, A. Braunstein, L. Dall'Asta, A. Ingrosso, and R. Zecchina, "The patient-zero problem with noisy observations," *J. Stat. Mech. Theory Exp.*, vol. 2014, no. 10, 2014, Art. no. P10016.

[26] F. Altarelli, A. Braunstein, L. Dall'Asta, A. Lage-Castellanos, and R. Zecchina, "Bayesian inference of epidemics on networks via belief propagation," *Phys. Rev. Lett.*, vol. 112, Mar. 2014, Art. no. 118701.

[27] A. Y. Lokhov, M. Mézard, H. Ohta, and L. Zdeborová, "Inferring the origin of an epidemic with a dynamic message-passing algorithm," *Phys. Rev. E, Stat. Phys. Plasmas Fluids Relat. Interdiscip. Top.*, vol. 90, Jul. 2014, Art. no. 012801.

[28] B. Chang, F. Zhu, E. Chen, and Q. Liu, "Information source detection via maximum a posteriori estimation," in *Proc. 15th IEEE Int. Conf. Data Min. (ICDM)*, Atlantic City, NJ, USA, 2015, pp. 21–30.

[29] C. H. Comin and L. da Fontoura Costa, "Identifying the starting point of a spreading process in complex networks," *Phys. Rev. E, Stat. Phys. Plasmas Fluids Relat. Interdiscip. Top.*, vol. 84, no. 5, Nov. 2011, Art. no. 056105.

[30] M. Granovetter, "Threshold models of collective behavior," *Amer. J. Sociol.*, vol. 83, no. 6, pp. 1420–1443, 1978.

[31] T. Lappas, E. Terzi, D. Gunopulos, and H. Mannila, "Finding effectors in social networks," in *Proc. 16th ACM SIGKDD Int. Conf. Knowl. Disc. Data Min. (KDD)*, Washington, DC, USA, 2010, pp. 1059–1068.

[32] D. T. Nguyen, N. P. Nguyen, and M. T. Thai, "Sources of misinformation in online social networks: Who to suspect?" in *Proc. IEEE Mil. Commun. Conf. (MILCOM)*, Oct./Nov. 2012, pp. 1–6.

[33] Z. Feng, P. Gundecha, and H. Liu, "Recovering information recipients in social media via provenance," in *Proc. IEEE/ACM Int. Conf. Adv. Soc. Netw. Anal. Min. (ASONAM)*, Niagara Falls, ON, Canada, Aug. 2013, pp. 706–711.

[34] W. Dong, W. Zhang, and C. W. Tan, "Rooting out the rumor culprit from suspects," in *Proc. IEEE Int. Symp. Inf. Theory*, Istanbul, Turkey, Jul. 2013, pp. 2671–2675.

[35] Z. Wang, W. Dong, W. Zhang, and C. W. Tan, "Rooting our rumor sources in online social networks: The value of diversity from multiple observations," *IEEE J. Sel. Topics Signal Process.*, vol. 9, no. 4, pp. 663–677, Jun. 2015.

[36] A. Jain, V. Borkar, and D. Garg, "Fast rumor source identification via random walks," *Soc. Netw. Anal. Min.*, vol. 6, no. 1, p. 62, 2016.

[37] P. C. Pinto, P. Thiran, and M. Vetterli, "Locating the source of diffusion in large-scale networks," *Phys. Rev. Lett.*, vol. 109, no. 6, Aug. 2012, Art. no. 068702.

[38] A. Agaskar and Y. M. Lu, "A fast Monte Carlo algorithm for source localization on graphs," in *Proc. SPIE*, San Diego, CA, USA, 2013, p. 8858.

[39] Z. Shen, S. Cao, W.-X. Wang, Z. Di, and H. E. Stanley, "Locating the source of diffusion in complex networks by time-reversal backward spreading," *Phys. Rev. E, Stat. Phys. Plasmas Fluids Relat. Interdiscip. Top.*, vol. 93, no. 3, Mar. 2016, Art. no. 032301.

[40] A. Kumar, V. Borkar, and N. Karamchandani, "Temporally agnostic rumor-source detection," *IEEE Trans. Signal Inf. Process. Over Netw.*, vol. 3, no. 2, pp. 316–329, Jun. 2017.

[41] J. G. Restrepo, E. Ott, and B. R. Hunt, "Characterizing the dynamical importance of network nodes and links," *Phys. Rev. Lett.*, vol. 97, no. 9, Sep. 2006, Art. no. 094102.

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

[43] H. T. Nguyen, P. Ghosh, M. L. Mayo, and T. N. Dinh, "Multiple infection sources identification with provable guarantees," in *Proc. 25th ACM Int. Conf. Inf. Knowl. Manag. (CIKM)*, 2016, pp. 1663–1672.

[44] A. Goyal, F. Bonchi, and L. V. S. Lakshmanan, "Learning influence probabilities in social networks," in *Proc. 3rd ACM Int. Conf. Web Search Data Min. (WSDM)*, 2010, pp. 241–250.
[45] P. Giles, "The mathematical theory of infectious diseases and its applications," *J. Oper. Res. Soc.*, vol. 28, no. 2, pp. 479–480, 1977.
[46] C. M. Bishop *et al.*, *Pattern Recognition and Machine Learning*, vol. 4. New York, NY, USA: Springer, 2006.
[47] Y. Yang *et al.*, *On Approximation of Real-World Influence Spread*. Heidelberg, Germany: Springer, 2012, pp. 548–564.
[48] B. R. Heap, "Permutations by interchanges," *Comput. J.*, vol. 6, no. 3, pp. 293–298, 1963.
[49] A.-L. Barabási and R. Albert, "Emergence of scaling in random networks," vol. 286, no. 5439, pp. 509–512, 1999.
[50] D. J. Watts and S. H. Strogatz, "Collective dynamics of 'small-world' networks," *Nature*, vol. 393, no. 6684, pp. 440–442, Jun. 1998.
[51] J. Leskovec, D. Huttenlocher, and J. Kleinberg, "Predicting positive and negative links in online social networks," in *Proc. 19th Int. Conf. World Wide Web (WWW)*, Raleigh, NC, USA, 2010, pp. 641–650.
[52] J. Leskovec, J. Kleinberg, and C. Faloutsos, "Graph evolution: Densification and shrinking diameters," *ACM Trans. Knowl. Disc. Data*, vol. 1, no. 1, p. 2, Mar. 2007.
[53] J. Leskovec and A. Krevl. (Jun. 2014). *SNAP Datasets: Stanford Large Network Dataset Collection*. [Online]. Available: http://snap.stanford.edu/data

Biao Chang received the B.S. degree in computer science from the University of Science and Technology of China, Hefei, China, in 2012, where he is currently pursuing the Ph.D. degree in computer science with the School of Computer Science and Technology, under the supervision of Prof. E. Chen.

He also visited Singapore Management University, Singapore, as a Research Assistant, under the supervision of Prof. F. Zhu from 2015 to 2016. His research has been published in conference proceedings, including International Joint Conference on Artificial Intelligence, IEEE International Conference on Data Mining, and ACM International Conference on Information and Knowledge Management. His current research interests include social network analysis and recommender systems.

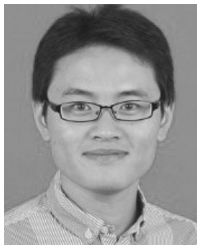
Enhong Chen (SM'07) received the Ph.D. degree in computer science from the University of Science and Technology of China (USTC), Hefei, China.

He is a Professor and the Vice Dean of the School of Computer Science, USTC. He has published over 150 papers in referred conferences and journals, including the IEEE TRANSACTIONS ON KNOWLEDGE AND DATA ENGINEERING, the IEEE TRANSACTIONS ON INDUSTRIAL ELECTRONICS, ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD), IEEE International Conference on Data Mining (ICDM), Neural Information Processing Systems Foundation, and ACM International Conference on Information and Knowledge Management. His current research interests include data mining and machine learning, social network analysis, and recommender systems.

Dr. Chen was a recipient of the Best Application Paper Award at KDD 2008, the Best Research Paper Award at ICDM 2011, and the Best of SIAM International Conference on Data Mining (SDM) 2015 Award. He was on program committees of numerous conferences, including KDD, ICDM, and SDM.

Feida Zhu received the B.S. degree in computer science from Fudan University, Shanghai, China, and the Ph.D. degree in computer science from the University of Illinois at Urbana–Champaign, Champaign, IL, USA.

He is an Assistant Professor with the School of Information Systems, Singapore Management University, Singapore. His current research interests include large-scale graph pattern mining and social network analysis, with applications on Web, management information systems, business intelligence, and bioinformatics.



Qi Liu (M'15) received the Ph.D. degree in computer science from the University of Science and Technology of China (USTC), Hefei, China.

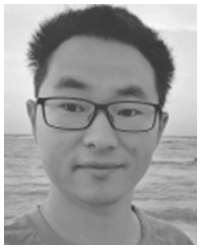
He is an Associate Professor with USTC. His current research interests include data mining and knowledge discovery. He has published prolifically in refereed journals and conference proceedings, such as the IEEE TRANSACTIONS ON KNOWLEDGE AND DATA ENGINEERING, *ACM Transactions on Information Systems*, *ACM Transactions on Knowledge Discovery From Data*, *ACM Transactions on Intelligent Systems and Technology*, ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, International Joint Conference on Artificial Intelligence, AAAI Conference on Artificial Intelligence, IEEE International Conference on Data Mining (ICDM), SIAM International Conference on Data Mining (SDM), and ACM International Conference on Information and Knowledge Management.

Dr. Liu was a recipient of the ICDM 2011 Best Research Paper Award and the Best of SDM 2015 Award. He has served regularly on the program committees of a number of conferences. He is a Reviewer for the leading academic journals in his fields. He is a member of ACM.

Tong Xu received the Ph.D. degree in computer science from the University of Science and Technology of China (USTC), Hefei, China, in 2016.

He is currently a Post-Doctoral Researcher with the Anhui Province Key Laboratory of Big Data Analysis and Application, USTC. He has authored nearly 20 journal and conference papers in the fields of social network and social media analysis, including ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, AAAI Conference on Artificial Intelligence, IEEE International Conference on Data Mining, and SIAM International Conference on Data Mining.

Dr. Xu was a recipient of the ACM (Hefei) Doctoral Dissertation Award in 2016.



Zhefeng Wang received the B.E. degree from the University of Science and Technology of China, Hefei, China, in 2012, where he is currently pursuing the Ph.D. degree in computer science with the School of Computer Science and Technology.

He has published several papers in refereed conference proceedings and journals, such as International ACM SIGIR Conference on Research and Development in Information Retrieval, ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, International Joint Conference on Artificial Intelligence, and the IEEE TRANSACTIONS ON KNOWLEDGE AND DATA ENGINEERING. His current research interests include social network, social media analysis, machine learning, and text mining.