

# Catch'Em All: Locating Multiple Diffusion Sources in Networks with Partial Observations

**Kai Zhu**  
 Google Inc.  
 zhukai93@gmail.com

**Zhen Chen and Lei Ying**  
 School of of Electrical, Computer and Energy Engineering  
 Arizona State University  
 {Zhen.Chen.1, Lei.Ying.2}@asu.edu

## Abstract

This paper studies the problem of locating multiple diffusion sources in networks with partial observations. We propose a new source localization algorithm, named Optimal-Jordan-Cover (OJC). The algorithm first extracts a subgraph using a candidate selection algorithm that selects source candidates based on the number of observed infected nodes in their neighborhoods. Then, in the extracted subgraph, OJC finds a set of nodes that “cover” all observed infected nodes with the minimum radius. The set of nodes is called the Jordan cover, and is regarded as the set of diffusion sources. Considering the heterogeneous susceptible-infected-recovered (SIR) diffusion in the Erdős-Rényi (ER) random graph, we prove that OJC can locate all sources with probability one asymptotically with partial observations. OJC is a polynomial-time algorithm in terms of network size. However, the computational complexity increases exponentially in  $m$ , the number of sources. We further propose a low-complexity heuristic based on the K-Means for approximating the Jordan cover, named Approximate-Jordan-Cover (AJC). Simulations on random graphs and real networks demonstrate that both AJC and OJC significantly outperform other heuristic algorithms.

## 1 Introduction

Diffusion source localization (or called the information source detection) is to infer the source(s) of an epidemic diffusion process in a network based on some observations of the diffusion. Possible observed information includes node states (e.g., infected or susceptible) and the timestamps at which nodes changed their states. The solution to this problem has a wide range of applications. In epidemiology, identifying patient zero helps diagnose the cause and the origin of the disease. For cybersecurity, tracing the source of malware is an important step in the investigation of a cyber attack. On on-line social networks, the trustworthiness of news/information heavily depends on its source.

Since the seminal work of Shah and Zaman (Shah and Zaman 2010), the problem has received a lot of attention from different research communities, including machine learning, signal processing and information theory (a detailed discussion can be found in the related work). However, most existing results assume that the diffusion is from single source

and the observed information is one or multiple complete network snapshot(s). Furthermore, provably guarantees on the detection rate have only been established for tree networks except a recent paper by Zhu and Ying (Zhu and Ying 2016), where they proposed a Short-Fat Tree (SFT) algorithm and proved that under the single-source, homogeneous Independent-Cascade (IC) model, SFT locates the actual source in the Erdős-Rényi random graph (Erdos and Renyi 1959) with probability one asymptotically (as the size of the network increases) when a complete snapshot of the network is given.

In this paper, we consider the diffusion source localization problem in a setting that generalizes the existing ones at several important directions.

- *Multiple sources versus single source:* In this paper, the diffusion can be originated from multiple nodes simultaneously, instead of from a single source. When the infection duration is sufficiently short, the infected subnetworks from different sources are disconnected components. In such cases, the single-source localization algorithms can be applied to each of the infected subnetwork. We, however, do not make such an assumption, and consider the scenario where the infected subnetworks may overlap with each other, so the single-source localization algorithms cannot be directly applied.
- *A partial snapshot versus a complete snapshot:* In this paper, we assume a partial snapshot in which each node reports its state with some probability, which is in contrast to a complete snapshot assumed in the literature where all nodes' states are observed. Because of a partial snapshot, the sources may not report their states and be observed as infected nodes; and the observed infected nodes may not form a connected component. Both increase the uncertainty and complexity of the problem. In fact, it turns out to be critical to have a candidate selection algorithm to select source candidates from unobserved nodes but only use observed infected nodes in computing the infection eccentricity. The selection step yields  $27\times$  reduction on the computing time in our simulations while guaranteeing the same the detection rate, and yields  $600\times$  reduction on the computing time with a slight reduction of the detection rate.
- *Heterogeneous diffusion versus homogeneous diffusion:*

Our algorithm applies to the heterogeneous SIR diffusion model where links have different infection probabilities and nodes have different recovery probabilities. The asymptotic guarantees on the detection rate hold for the heterogeneous SIR model.

While some of these extensions have been investigated in the literature individually, our model includes all three extensions. We propose a novel algorithm for locating multiple sources for such a general model and prove theoretical guarantees on the detection rate for non-tree networks. The main results of the paper are summarized below.

- (1) We introduce the concept of Jordan cover, which is an extension of Jordan center. Loosely speaking, a Jordan cover with size  $m$  is a set of  $m$  nodes that can reach all *observed* infected nodes with the minimum hop-distance. We propose Optimal-Jordan-Cover (OJC), which consists of two steps: OJC first selects a subset of nodes as the set of the candidates of the diffusion sources; and then it finds a Jordan cover in the subgraph induced by the candidate nodes and the observed infected nodes. We emphasize that only the hop-distance to the observed infected nodes is considered in computing a Jordan cover.
- (2) We analyze the performance of OJC on the ER random graph, and establish the following performance guarantees.
  - (i) When the infection duration is shorter than  $\frac{2}{3} \frac{\log n}{\mu}$ , where  $\mu$  is the average node degree and  $n$  is the number of nodes in the network, OJC identifies the sources with probability one asymptotically as  $n$  increases.
  - (ii) When the infection duration is at least  $\left\lceil \frac{\log n}{\log \mu + \log q} \right\rceil + 2$  where  $q$  is the minimum infection probability, *under any source location algorithm*, the detection rate diminishes to zero as  $n$  increases under the Susceptible-Infected (SI) and Independent-Cascade (IC) models, which are special cases of the SIR model.
- (3) The computational complexity of OJC is polynomial in  $n$ , but exponential in  $m$ . We further propose a heuristic based on the K-Means for approximating the Jordan cover, named Approximate-Jordan-Cover (AJC). Assuming a constant number of iterations when using the K-Means, the computational complexity of AJC is  $O(nE)$ , where  $E$  is the number of edges. Our simulations on random graphs and real networks demonstrate that both AJC and OJC significantly outperform other heuristic algorithms.

## 1.1 Related Work

Shah and Zaman (Shah and Zaman 2010; 2011) studied the rumor source detection problem, and proposed a new graph centrality called *rumor centrality*. They proved that the node with the maximum rumor centrality is the maximum likelihood estimator (MLE) of the diffusion source on regular trees under the continuous-time SI model. In addition, it has been proved that the detection probability is greater than zero on regular trees and approaches one for geometric trees. (Shah and Zaman 2012) analyzed the detection probability of rumor centrality for general random trees. Later, the performance

of rumor centrality has been studied under different models, including multiple sources (Luo, Tay, and Leng 2013), incomplete observations (Karamchandani and Franceschetti 2013), multiple independent diffusion processes from the same source (Wang et al. 2014).

Kai and Ying (Zhu and Ying 2013) proposed a sample path based approach for single source detection under the SIR model and introduced the concept of *Jordan infection center*. Assuming the homogeneous SIR model, a complete network snapshot and regular tree networks, they (Zhu and Ying 2013) proved that the Jordan infection center is the root of the most likely diffusion sample path, and it is within a constant hop-distance from the actual source with a high probability. *Assuming tree networks*, the performance of the Jordan infection center has been further studied, including partial observations under the heterogeneous SIR model (Zhu and Ying 2014), multiple sources (Chen, Zhu, and Ying 2014), the SI model and SIS model (Luo, Tay, and Leng 2014).

Besides the rumor centrality and sample path based approach, diffusion source localization has also been investigated from other perspectives: 1) (Lappas et al. 2010) proposed a dynamical programming algorithm for the IC model; 2) A variation of eigen centrality was proposed in (Prakash, Vreeken, and Faloutsos 2012) to detect multiple sources under the SI model; 3) (Lokhov et al. 2014) derived the mean field approximation of the MLE of the actual source and proposed a dynamic message passing algorithm based on that. Furthermore, (Pinto, Thiran, and Vetterli 2012; Zhu, Chen, and Ying 2015; Agaskar and Lu 2013; Zejnolovic, Gomes, and Sinopoli 2013; Farajtabar et al. 2015) have proposed algorithms to identify the source with partial timestamps.

In this paper, we propose two new source localization algorithms, OJC and AJC, based on the Jordan cover. OJC guarantees probability one detection asymptotically on the ER random graph under the multi-source, heterogeneous SIR model, and with an incomplete snapshot, which is the first multi-source localization algorithm with provable guarantees for non-tree networks. AJC is a low-complexity approximation of OJC.

## 2 Problem Formulation

We assume the network is represented by an undirected graph  $g$ . Denote by  $\mathcal{E}(g)$  the set of edges and  $\mathcal{V}(g)$  the set of nodes in graph  $g$ . Let  $n$  denote the number of nodes and  $E$  denote the number of edges. We further assume a heterogeneous SIR model for diffusion. In this model, each node has three possible states: susceptible (S), infected (I) and recovered (R). Time is slotted. At the beginning of each time slot, each infected node (say node  $u$ ) attempts to infect its neighbor (say node  $v$ ) with probability  $q_{uv}$ , independently across edges. We call  $q_{uv}$  the *infection probability* of edge  $(u, v)$ . At the end of each time slot, each infected node (say node  $u$ ) recovers with probability  $r_u$ , independent of other infected nodes. We call  $r_u$  the *recovery probability*. We further assume  $q_{uv} \in (0, 1]$  for all edges  $(u, v) \in \mathcal{E}(g)$  and  $r_v \in [0, 1]$  for all nodes  $v \in \mathcal{V}(g)$ .

Note that the SIR model includes two important special cases. When the recovery probability is zero, the SIR model

becomes the Susceptible-Infected (SI) model (Bailey 1975), where infected nodes cannot recover. When the recovery probability is one, the SIR model becomes the Independent Cascade (IC) model (Goldenberg, Libai, and Muller 2001) by regarding both infected nodes and recovered nodes as active nodes and regarding the susceptible nodes as inactive nodes.

We assume the epidemic diffusion starts from  $m$  sources in the network. In other words, at time slot 0,  $m$  nodes (sources) are in the infected state and all other nodes are in the susceptible state. Denote by  $s_1, s_2, \dots, s_m$  the sources and  $\mathcal{S}$  the set of sources, i.e.,  $\mathcal{S} = \{s_1, s_2, \dots, s_m\}$ . We assume  $m$  is a constant independent of  $n$ .

Finally, we assume that a *partial* snapshot of the network state at time slot  $t$  is given, with an unknown observation time  $t$ . In the snapshot, each infected or recovered node reports its state with probability  $\theta_v \in (0, 1)$ , independent of other nodes. If a node reports its state, we call it an observed node. Denote by  $\mathcal{I}'$  the set of observed infected and recovered nodes. In this paper, we call infected nodes and recovered nodes as “infected nodes” unless explicitly clarified.

Based on  $\mathcal{I}'$ , the source localization problem is to find  $\mathcal{S}$  that solves the following maximum likelihood (ML) problem

$$\mathcal{W}^* = \operatorname{argmax}_{\mathcal{W} \subset \mathcal{V}(g)} \Pr(\mathcal{S} = \mathcal{W} | \mathcal{I}').$$

Even with a single diffusion source, this problem is known to be a difficult problem (Shah and Zaman 2010; Zhu and Ying 2013) on non-tree networks. Therefore, instead of solving the ML problem above, we are interested in algorithms with *asymptotic perfect detection*, i.e., finding all sources with probability one as the network size increases. We believe this alternative metric is reasonable because we often need to solve the problem for large-size networks such as online social networks, and an algorithm with asymptotic perfect detection can detect sources with a high probability when the network size is large.

### 3 Algorithms

In this section, we present OJC and AJC based on the concept of Jordan cover. Define the hop-distance between a node  $v$  and a *node set*  $\mathcal{W}$  to be the minimum hop-distance between node  $v$  and any node in  $\mathcal{W}$ , i.e.,

$$d(v, \mathcal{W}) \triangleq \min_{u \in \mathcal{W}} d(v, u).$$

We then define the *infection eccentricity* of node set  $\mathcal{W}$  to be the maximum hop-distance from an infected node in  $\mathcal{I}'$  to set  $\mathcal{W}$ , i.e.,

$$e(\mathcal{W}, \mathcal{I}') = \max_{v \in \mathcal{I}'} d(v, \mathcal{W}). \quad (1)$$

We further define  $m$ -Jordan-cover ( $m$ -JC) to be the set  $\mathcal{W}^*(\mathcal{K}, \mathcal{I}', m)$  such that

$$\mathcal{W}^*(\mathcal{K}, \mathcal{I}', m) = \operatorname{argmin}_{\mathcal{W} \in \{\mathcal{W} | |\mathcal{W}| = m, \mathcal{W} \subset \mathcal{K}\}} e(\mathcal{W}, \mathcal{I}'). \quad (2)$$

where  $\mathcal{I}'$  is the set of observed infected nodes that  $m$ -JC needs to cover and  $\mathcal{K}$  is the *candidate set* for the sources. Therefore,  $m$ -JC is the set of  $m$  nodes in  $\mathcal{K}$  with the minimum infection eccentricity.

We now introduce the *optimal Jordan cover* (OJC) algorithm whose asymptotic detection rate will be analyzed in Section 4.1.

#### The Optimal Jordan Cover (OJC) Algorithm

- **Step 1: Candidate Selection:** Let  $Y$  be a positive integer. The candidate set  $\mathcal{K}$  is the set of nodes with more than  $Y$  observed infected neighbors. In addition, define  $\mathcal{K}^+ \triangleq \mathcal{K} \cup \mathcal{I}'$ . Denote by  $g^-$  a connected subgraph of  $g$  induced by node set  $\mathcal{K}^+$ . An induced graph is a subset of nodes of a graph with all edges whose endpoints are both in the node subset. If the induced graph is not connected, we select a random node in each component, randomly pick one selected node and add the shortest paths from this node to all other selected nodes to form a connected  $g^-$ . We call  $Y$  the *selection threshold*. Please check the full version (Zhu, Chen, and Ying 2016) for the pseudo code of the candidate selection algorithm for selecting  $\mathcal{K}$  and  $g^-$ .
- **Step 2: Jordan Cover:** For any  $m$  combination of nodes in  $\mathcal{K}$  in Step 1, we compute the infection eccentricity of the node set as defined in (1) on subgraph  $g^-$ , and select the combination with the minimum infection eccentricity as the set of sources. Ties are broken by the total distance from the observed infected to the node set, i.e.,  $\sum_{v \in \mathcal{I}'} d(v, \mathcal{W})$ .  $\square$

With a properly chosen threshold  $Y$ , the candidate selection step includes all sources in  $\mathcal{K}$  with a high probability and excludes nodes that are more than  $t + 1$  hops away from all sources. By limiting the computation on the induced subgraph  $g^-$ , the computational complexity is reduced significantly. From simulations, we will see that it results in  $27\times$  reduction of the running time without affecting the detection rate. The asymptotic detection rate of OJC will be studied in Theorem 2. Under some conditions, OJC identifies all sources with probability one asymptotically.

OJC is a polynomial-time algorithm for given  $m$ , but the complexity increases exponentially in  $m$ . To further reduce the complexity, we propose Approximate Jordan Cover (AJC), which replaces Step 2 of OJC with the K-Means algorithm (Hartigan and Wong 1979). As shown in the simulations, the performance of the AJC algorithm, in terms of both detection rate and the error distance, is close to OJC with much shorter running time. The computational complexity of both algorithms are summarized in the following theorem. The analysis can be found in the full version (Zhu, Chen, and Ying 2016).

**Theorem 1.** *The computational complexity of OJC is*

$$O\left(|\mathcal{I}'| \left( |\mathcal{E}(g)| + m \binom{|\mathcal{V}(g^-)|}{m} \right)\right),$$

*and the computational complexity of AJC is*

$$O\left(|\mathcal{I}'| (|\mathcal{E}(g)| + H|\mathcal{V}(g^-)|)\right),$$

*where  $H$  is the number of iterations used in the K-Means algorithm in AJC.*  $\square$

## 4 Asymptotic Analysis of OJC

In this section, we present the asymptotic analysis of the detection rate of OJC on the ER random graph. The results include the conditions that guarantee probability one detection and the conditions under which it is impossible to detect the source set with nonzero probability under any source localization algorithm.

### 4.1 Asymptotic perfect detection on the ER random graph

We first present the positive result that shows that on the ER random graph, OJC identifies the  $m$  sources with probability one asymptotically under some conditions. Recall that  $n$  is the number of nodes in the graph, and  $m$  is the number of sources, which is a constant independent of  $n$ . Denote by  $p$  the wiring probability of the ER random graph, which is the probability that there exists a link between two nodes. Let  $\mu = np$ , which is the average node degree. Define  $q \triangleq \min_{u,v \in \mathcal{V}(g)} q_{u,v}$ , i.e., the minimum infection probability over all edges and  $\theta \triangleq \min_{v \in \mathcal{V}(g)} \theta_v$  i.e., the minimum report probability over all nodes.

**Theorem 2.** *OJC identifies all  $m$  sources with probability one as  $n \rightarrow \infty$  when the following conditions hold:*

- (c1):  $\mu q \theta = \Omega(\log n)$ ,<sup>1</sup>
- (c2):  $\limsup_{n \rightarrow \infty} \frac{Y}{\mu q \theta} < 1$  and  $\liminf_{n \rightarrow \infty} \frac{Y}{\mu q \theta} > 0$ , and
- (c3):  $t = \omega(D)$  and  $\limsup_{n \rightarrow \infty} \frac{t}{\frac{\log n}{\log \mu}} < \frac{2}{3}$ .  $\square$

We now briefly explain the conditions. Recall that  $\mu$  is the average node degree,  $q$  is the lower bound on the infection probability and  $\theta$  is the lower bound on the reporting probability, so  $\mu q \theta$  is a lower bound on the average number of observed infected neighbors of a node that was infected before time slot  $t$ . Therefore, condition (c1) requires that this lower bound is  $\Omega(\log n)$ , and condition (c2) requires that the threshold used in the candidate selection algorithm is a constant fraction of the average number of observed infected neighbors. Applying the Chernoff bound, conditions (c1) and (c2) together yield the following conclusions:

- (i) any node who was infected before or at time slot  $t - 1$  (hence, including the sources) will be selected into the candidate set with a high probability,
- (ii) any node that is  $t + D + 1$  hops away from the set of sources will not have  $Y$  or more observed infected neighbors with a high probability, and
- (iii) any node that is more than  $t + D + 1$  hops away from the set of sources will not have any observed infected neighbors.

Based the above facts, with a high probability, the candidate set includes all nodes who were infected at  $t - 1$  or earlier, and any node in  $g^-$  is at most  $t + D$  hops away from all sources.

Condition (c3) is on the infection duration. We first restrict  $t = \omega(D)$  so that the infection subgraphs starting from different sources are likely to overlap and form a connected

<sup>1</sup>Throughout this paper, the asymptotic order notation is defined for  $n \rightarrow \infty$ .

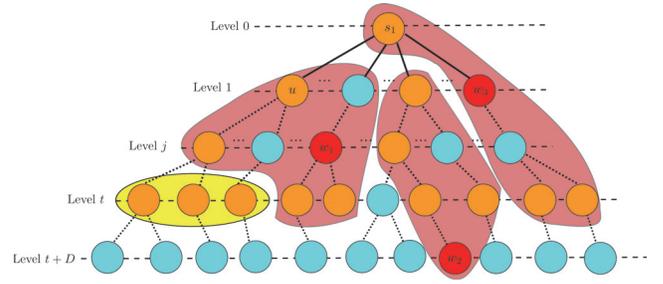


Figure 1: A pictorial example for Theorem 2.

component. This is a more interesting regime than the one in which infection subgraphs are disconnected from each other.  $\limsup_{n \rightarrow \infty} \frac{t}{\frac{\log n}{\log \mu}} < \frac{2}{3}$  is a critical condition. The intuition why it is required is explained below. Figure 1 provides a pictorial explanation of the proof. The picture illustrates the breadth-first-search (BFS) tree  $T^\dagger$  rooted at source  $s_1$  with height  $t + D$ , where  $s_1$  is one of the  $m$  sources. The nodes in orange are the observed infected nodes whose infection was originated from  $s_1$ . The blue nodes are unobserved nodes. A node is said to be on level  $i$  of the BFS if its hop-distance to  $s_1$  is  $i$ . Assume  $m = 3$  and consider a set of three nodes who are within  $t + D$  hops from  $s_1$  but not includes  $s_1$  (e.g.,  $\mathcal{W} = \{w_1, w_2, w_3\}$  in Figure 1). Suppose the infection eccentricity of  $\mathcal{W}$  is  $\leq t$ . Since  $s_1$  has a sufficient number of neighbors according to the definition of  $\mu$ , with a high probability, there exists a subtree of  $T^\dagger$  starting from an offspring of  $s_1$ , which does not include any node in  $\mathcal{W}$ . Assume  $u$  is the root of such a subtree in Figure 1. The yellow area in Figure 1 includes the level- $t$  observed infected nodes on subtree  $T_u^{-s_1}$ . Any path from  $w_1, w_2$  or  $w_3$  to the yellow area, formed by edges on  $T^\dagger$ , must have hop-distance larger than  $t$ . Therefore, if the infection eccentricity of  $\mathcal{W}$  is at least  $t$ , there must exist a path from  $\mathcal{W}$  to each of the nodes in the yellow area with hop-distance  $\leq t$ , and such a path must include at least one edge which is not in  $T^\dagger$  (we call these edges *collision edges*). In the detailed analysis, we will prove that with a high probability, the number of nodes within  $t$  hops from  $\mathcal{W}$  via the collision edges is order-wise smaller than the number of nodes in the yellow area when (c3) holds. Therefore, the Jordan cover has to include  $s_1$ . The same argument applies to other sources. The detailed proof, which formalizes the argument above, can be found in the full version (Zhu, Chen, and Ying 2016).

### 4.2 Impossibility results

Theorem 5 in (Zhu and Ying 2016) presents the conditions under which it is impossible to identify the single source under the IC diffusion on the ER random graph, which is a special case of the model in this paper. Assuming SI or IC model, based on Lemma 1 in (Zhu and Ying 2016), we have that with a high probability, all nodes of the ER graph become infected when the infection duration is at least

$$t_u \triangleq \left\lceil \frac{\log n}{\log \mu + \log q} \right\rceil + 2.$$

When this occurs, it is impossible to detect the sources since the nodes are indifferentiable.

**Theorem 3.** *Assume the multi-source diffusion follows the IC or SI model. If  $24 \log n < q\mu \ll \sqrt{n}$  and  $q$  is a constant independent of  $n$ , then*

$$\lim_{n \rightarrow \infty} \Pr(\mathcal{I} = \mathcal{V}(g)) = 1$$

when the observation time  $t \geq t_u$ . In other words, the entire network is infected after  $t_u$  with a high probability. In such a case, the probability of finding the sources diminishes to zero as  $n \rightarrow \infty$ .  $\square$

## 5 Simulations

In this section, we evaluated the performance of our algorithms via simulations. The performance metrics used in this paper include:

- Error distance: The error distance is defined to be

$$\min_{\mathcal{P} \in \text{permutation}(\mathcal{S}')} \sum_{i=1}^m \frac{d(s_i, p_i)}{m},$$

where  $s_1, s_2, \dots, s_m$  are the real sources,  $\mathcal{S}'$  is the set of detected sources and  $\mathcal{P} = (p_1, p_2, \dots, p_m)$  is a permutation of  $\mathcal{S}'$ .

- Detection rate: The detection rate is defined as

$$\frac{|\mathcal{S} \cap \mathcal{S}'|}{m}.$$

We compared our algorithms with two heuristic algorithms (DC and CC) based on K-Means, which have been used for comparison in (Luo, Tay, and Leng 2014). The algorithms proposed in (Luo, Tay, and Leng 2014) and (Chen, Zhu, and Ying 2014) are the same as AJC without candidate selection. In both DC and CC, the initial centroids are randomly chosen. During the clustering step of each iteration in K-Means, we selected distance centroid of each cluster in DC and selected closeness centroid in CC, where distance centroid is defined as  $\arg \min_{v \in \mathcal{C}} \sum_{u \in \mathcal{C} \cap \mathcal{I}'} d(v, u)$ , where  $\mathcal{C}$  is the set of nodes in the cluster. Closeness centroid is defined as  $\arg \max_{v \in \mathcal{C}} \sum_{u \in \mathcal{C} \cap \mathcal{I}', u \neq v} \frac{1}{d(u, v)}$ . The following experiments were conducted on a server with 8 Intel Xeon X3450 CPUs and 16G RAM with Linux 64 bit system. All algorithms were implemented with Python 2.7.

### 5.1 OJC with different thresholds

In Figure 2, we evaluated OJC on the ER random graph. In the experiments, we generated an ER random graph with 5,000 nodes and wiring probability 0.002. We used the homogeneous SI model for diffusion with infection probability 0.8. In this experiment, we limited the infection network size to be  $100 \sim 300$  and the number of sources to be 2 due to the computational complexity of the OJC algorithm. Figure 2 shows the performance of OJC with different thresholds. From the results, the detection rate is close to one and the error distance is close to zero under OJC with threshold 0 or 1. However, the running time is 1,817 seconds versus 68 seconds. So the candidate selection algorithm with threshold

one results in  $27 \times$  reduction of the running time. When the threshold increases 2, the running time reduces to 3 seconds, which is a  $600 \times$  reduction of the running time. Both the detection rate and the error distance became slightly worse in this case. The detection rate in this case is 0.961 and the error distance is 0.056.

### 5.2 OJC, AJC and other heuristics

We further evaluated the performance of OJC and AJC on both the power grid network (Watts and Strogatz 1998) and ER random graph (size: 5000, wiring probability: 0.002) and compared them with DC and CC heuristics. We used the homogeneous SI model with infection probability 0.8 to generate the diffusion sequences. For AJC/CC/DC, for each diffusion sequence, we repeated the algorithm 100 times from different initial conditions and chose the source set with the smallest the smallest-infection-eccentricity/largest-closeness-centrality/smallest-distance-centrality. In Figure 3 and 4, the  $x$ -axis represents the combinations of sample rate and threshold. On the ER random graph, we increased the threshold as the sample rate increased to control the running time. For the power-grid network, since the average node degree is only 2, we set threshold equal to 2 for experiments for all sample rates. As we can see from the figures that when fixing the threshold, the performance of all algorithms (in terms of both error distance and detection rate) improves as the sample rate increases because we had more information about the diffusion. From Figure 3 and 4, we can also see that AJC outperforms DC and CC, and has similar performance with OJC. Note that with four sources, OJC became very slow on both the ER random graph and the power grid network because its complexity increases exponentially in the number of sources. So for the cases with four sources, we only simulated AJC.

## 6 Conclusions

In this paper, we studied the problem of detecting multiple diffusion sources under the heterogeneous SIR model with incomplete observations. We defined a concept called Jordan cover and developed the OJC algorithm based on that. Our theoretical analysis showed that OJC finds the set of sources in the ER random graph with probability one asymptotically under mild conditions. To the best of our knowledge, this is the first theoretic performance guarantee for multiple information sources detection in non-tree networks. Since the computational complexity of OJC is polynomial in  $n$  but exponential in  $m$ , we proposed a heuristic algorithm — the AJC algorithm. Our simulation results showed that OJC and AJC algorithms have similar performance and both significantly outperform existing algorithms.

## 7 Acknowledgments

This work was supported in part by the U.S. Army Research Laboratory's Army Research Office (ARO Grant No. W911NF1310279) and by the Defense Threat Reduction Agency (DTRA Grant HDTRA1-16-0017).

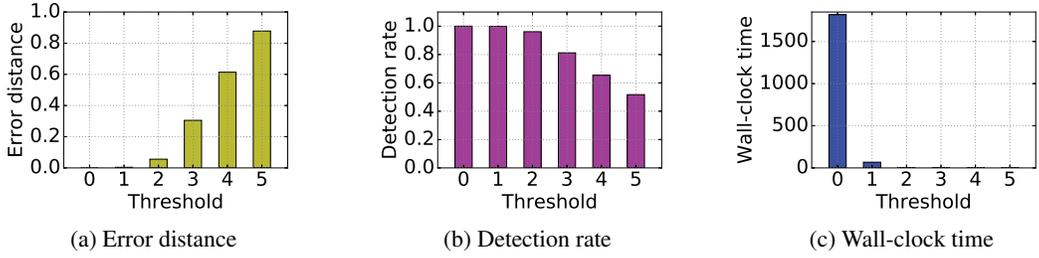


Figure 2: Performance of OJC with different threshold values on the ER random graph

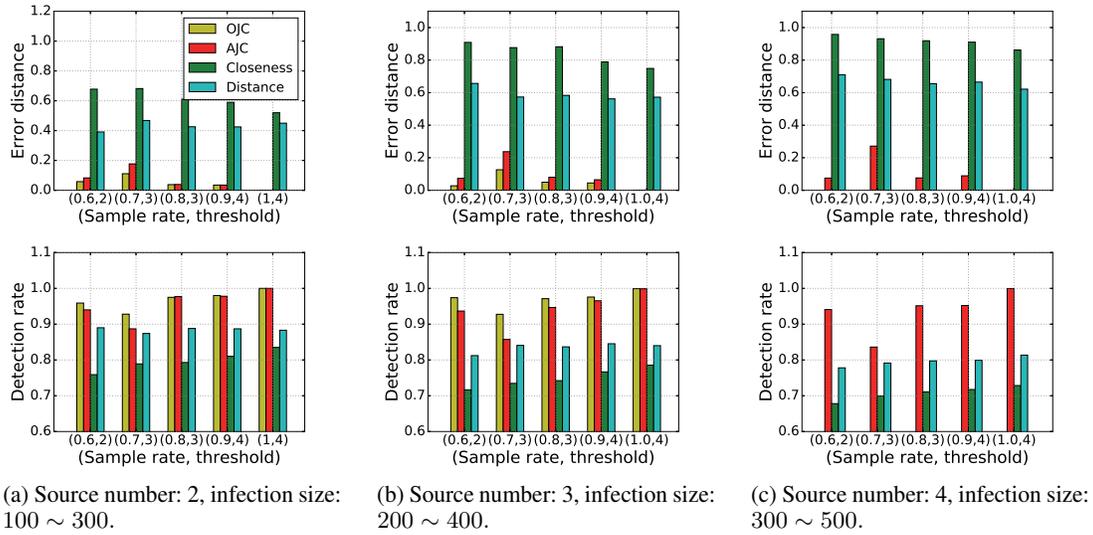


Figure 3: The Performance of OJC, AJC, CC and DC on the ER random graph with different sample rates and threshold values

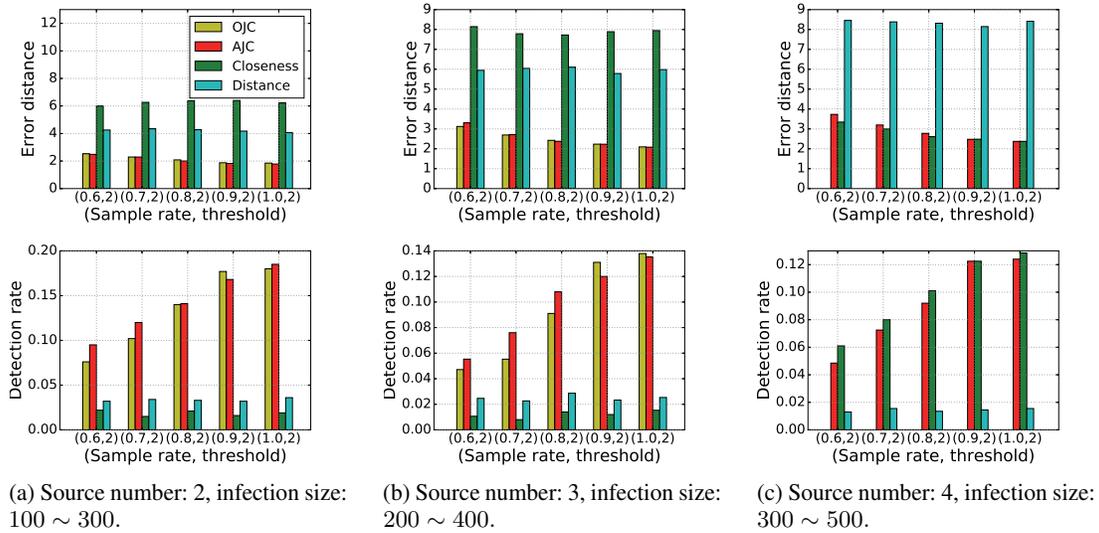


Figure 4: The Performance of OJC, AJC, CC and DC on the power grid network with different sample rates and threshold values

## References

- Agaskar, A., and Lu, Y. M. 2013. A fast monte carlo algorithm for source localization on graphs. In *SPIE Optical Engineering and Applications*.
- Bailey, N. T. J. 1975. *The mathematical theory of infectious diseases and its applications*. Hafner Press.
- Chen, Z.; Zhu, K.; and Ying, L. 2014. Detecting multiple information sources in networks under the SIR model. In *Proc. IEEE Conf. Information Sciences and Systems (CISS)*.
- Erdos, P., and Renyi, A. 1959. On random graphs I. *Publ. Math. Debrecen* 6:290–297.
- Farajtabar, M.; Gomez-Rodriguez, M.; Zamani, M.; Du, N.; Zha, H.; and Song, L. 2015. Back to the past: Source identification in diffusion networks from partially observed cascades. In *Int. Conf. Artificial Intelligence and Statistics*.
- Goldenberg, J.; Libai, B.; and Muller, E. 2001. Talk of the network: A complex systems look at the underlying process of word-of-mouth. *Marketing Letters* 12(3):211–223.
- Hartigan, J. A., and Wong, M. A. 1979. Algorithm as 136: A k-means clustering algorithm. *Journal of the Royal Statistical Society. Series C (Applied Statistics)* 28(1):100–108.
- Karamchandani, N., and Franceschetti, M. 2013. Rumor source detection under probabilistic sampling. In *Proc. IEEE Int. Symp. Information Theory (ISIT)*.
- Lappas, T.; Terzi, E.; Gunopulos, D.; and Mannila, H. 2010. Finding effectors in social networks. In *Proc. Ann. ACM SIGKDD Conf. Knowledge Discovery and Data Mining (KDD)*, 1059–1068.
- Lokhov, A. Y.; Mézard, M.; Ohta, H.; and Zdeborová, L. 2014. Inferring the origin of an epidemic with a dynamic message-passing algorithm. *Phys. Rev. E* 90.
- Luo, W.; Tay, W. P.; and Leng, M. 2013. Identifying infection sources and regions in large networks. *IEEE Trans. Signal Process.* 61:2850–2865.
- Luo, W.; Tay, W. P.; and Leng, M. 2014. On the universality of jordan centers for estimating infection sources in tree networks. *arXiv preprint arXiv:1411.2370*.
- Pinto, P. C.; Thiran, P.; and Vetterli, M. 2012. Locating the source of diffusion in large-scale networks. *Phys. Rev. Lett.* 109(6):068702.
- Prakash, B. A.; Vreeken, J.; and Faloutsos, C. 2012. Spotting culprits in epidemics: How many and which ones? In *IEEE Int. Conf. Data Mining (ICDM)*, 11–20.
- Shah, D., and Zaman, T. 2010. Detecting sources of computer viruses in networks: Theory and experiment. In *Proc. Ann. ACM SIGMETRICS Conf.*, 203–214.
- Shah, D., and Zaman, T. 2011. Rumors in a network: Who's the culprit? *IEEE Trans. Inf. Theory* 57:5163–5181.
- Shah, D., and Zaman, T. 2012. Rumor centrality: a universal source detector. In *Proc. Ann. ACM SIGMETRICS Conf.*, 199–210.
- Wang, Z.; Dong, W.; Zhang, W.; and Tan, C. W. 2014. Rumor source detection with multiple observations: fundamental limits and algorithms. In *Proc. Ann. ACM SIGMETRICS Conf.*
- Watts, D. J., and Strogatz, S. H. 1998. Collective dynamics of 'small-world' networks. *nature* 393(6684):440–442.
- Zejnivic, S.; Gomes, J.; and Sinopoli, B. 2013. Network observability and localization of the source of diffusion based on a subset of nodes. In *Proc. Annu. Allerton Conf. Communication, Control and Computing*.
- Zhu, K., and Ying, L. 2013. Information source detection in the SIR model: A sample path based approach. In *Proc. Information Theory and Applications Workshop (ITA)*.
- Zhu, K., and Ying, L. 2014. A robust information source estimator with sparse observations. In *Proc. IEEE Int. Conf. Computer Communications (INFOCOM)*.
- Zhu, K., and Ying, L. 2016. Information source detection in networks: Possibility and impossibility results. In *Proc. IEEE Int. Conf. Computer Communications (INFOCOM)*.
- Zhu, K.; Chen, Z.; and Ying, L. 2015. Locating the contagion source in networks with partial timestamps. *Data Mining and Knowledge Discovery*.
- Zhu, K.; Chen, Z.; and Ying, L. 2016. Catch'em all: Locating multiple diffusion sources in networks with partial observations. *arXiv:1611.06963*.