NETWORK OBSERVABILITY AND LOCALIZATION OF SOURCES OF DIFFUSION IN TREE NETWORKS WITH MISSING EDGES

Sabina Zejnilović,[1,2] *João Gomes*[2] *and Bruno Sinopoli*[1]

[1] Carnegie Mellon University, Department of Electrical and Computer Engineering, Pittsburgh, PA

sabinaz@cmu.edu, brunos@ece.cmu.edu

[2] Institute for Systems and Robotics – Instituto Superior Técnico (ISR/IST), University of Lisbon, Portugal

jpg@isr.ist.utl.pt

ABSTRACT

In order to quickly curb infections or prevent spreading of rumors, first the source of diffusion needs to be localized. We analyze the problem of source localization, based on infection times of a subset of nodes in incompletely observed tree networks, under a simple propagation model. Our scenario reflects the assumption that having access to all the nodes and full network topology is often not feasible. We evaluate the number of possible topologies that are consistent with the observed incomplete tree. We provide a sufficient condition for the selection of observed nodes, such that correct localization is possible, i.e. the network is observable. Finally, we formulate the source localization problem under these assumptions as a binary linear integer program. We then provide a small simulation example to illustrate the effect of the number of observed nodes on the problem complexity and on the number of possible solutions for the source.

Index Terms— Network observability, source localization, tree graphs, missing edges

1. INTRODUCTION

Many different phenomena, such as the dissemination of information in social networks and spreading of infectious diseases in networks are modeled as diffusion over a network of nodes. Localizing the source of diffusion becomes crucial for damage control and prevention of further infection, or for identifying the perpetrators who spread rumors and even for identifying influential individuals who start certain trends.

The issue of the origin of the diffusion was first addressed in [1], where based on a metric called rumor centrality, the rumor source is estimated by observing which nodes got infected. In [2], assuming that not only the state of the nodes is known, but also infection times, an optimal estimator for trees, and suboptimal for general networks, is proposed. Since in most real world networks it is unfeasible to monitor all nodes, estimation is based only on a subset of nodes, called observer nodes and several strategies for the selection of these nodes are experimentally evaluated. A sufficient condition for source localization based on the choice of observers, under the assumption of deterministic propagation is provided in [3]. The work of [1] is extended in [4], from the case of single source to the case of unknown number of sources.

As the network data is often incompletely known due to the fact that individuals are not always willing to share all the information about their social connections or information is exchanged through other means than observed social network sites, another active area of research in network diffusion deals with inference of the underlying network over which the propagation takes place. Given the times when nodes became infected, [5] proposes a near-optimal network. Assuming a part of the network topology is known, inference of the missing edges and nodes is tackled in [6].

We address the issues of localizing the source of diffusion and network observability, assuming not only that the infection times of all the nodes are not available, but also that the network topology is not fully known. However, estimating the source of diffusion or achieving network observability in general networks, even when the topology is fully known, is very challenging [1–3]. Therefore, here we assume a simplified deterministic noiseless propagation scenario and analyze tree topologies, in order to theoretically model certain aspects of a diffusion process in networks and gain crucial insight for tackling real-world scenarios. Even though tree networks are used to model hierarchical structures, for example an organizational chart of a company or phylogenetic information, most real-world networks have a non-tree like structure. Still, the diffusion in a general network occurs along the spanning tree of the network, which corresponds to the first time each node became informed or infected. Consequently, source localization problem in general networks is often reduced to source localization in trees [1, 2].

Even in a simple topology like a tree network, inability to observe all the edges leads to a dramatic increase in the

This research was supported by Fundação para a Ciência e a Tecnologia (project PEst-OE/EEI/LA0009/2013 and a PhD grant from the Carnegie Mellon-Portugal program) and EU FP7 project MORPH (grant agreement no. 288704.)

complexity of the source localization problem. Depending on the number of missing edges and the number of nodes, it is possible to construct many different trees that are consistent with the observed topology. We provide an expression for the exact number of trees in order to assess the complexity of the problem. Since the choice of observed nodes highly affects the localization [2,3], we present a sufficient condition for observer selection in incomplete trees that leads to the network being observable, i.e. unambiguous source identification. Finally, we formulate source localization problem as a binary integer program. For each node in a network, the feasibility of the proposed program is verified in order to identify possible sources. We provide a small example to illustrate the effect of the number of observers on the number of possible solutions and on the time needed to obtain them.

2. PROBLEM FORMULATION

We assume the widely used Susceptible-Infected propagation model, where once a node is infected or informed, it remains in that state [1, 2, 4]. The diffusion starts at time 0 from a single infected node in a network, denoted as the source node. With simplified dynamics of deterministic diffusion, once a node is infected at $t-1$, in the following time instant t, where t is a discrete time index, it will infect all of its neighbors, with probability 1. The time it takes for a certain node to become infected is equal to its distance from the source node. The nodes whose state can be observed, and whose infection times are known, are denoted as observer nodes. The source localization problem then consists of identifying the first infected node based on the infection times of observer nodes.

A network of n nodes is represented using a graph $G = \{V, E\}$, where $V = \{1, \ldots, n\}$ is the set of vertices representing the nodes, and $E \subseteq \binom{V}{2}$ is the set of edges. There is an edge between nodes i and j if nodes i and j can communicate directly. We assume the underlying network is an undirected tree, i.e., a connected graph without any cycles. If *k-1* edges are not observed in a tree, the observed topology becomes a forest, a disjoint union of k smaller trees that we will refer to as the components or subtrees. In order to tackle source localization in a partially observed tree, we first evaluate the cardinality of the set $\mathcal{T}$, which represents the set of all possible trees that are consistent with the observed forest. From each tree belonging to $\mathcal{T}$, an observed k-component forest can be obtained if *k-1* edges are removed.

Lemma 1. *From the forest of k components, each comprising c_i nodes, $i = 1 \ldots k$, the total number of trees that could be constructed by adding* k-1 *edges equals*

$$|\mathcal{T}| = \sum_{p=1}^{k^{k-2}} \prod_{(i,j)\in T_p} c_i c_j, \tag{1}$$

where T_p denotes the tree belonging to a class of trees that can be formed on a set of k nodes.

Proof. Given k nodes, Cayley's well known formula states that there are k^{k-2} spanning trees that can be constructed. Considering components as isolated supernodes, we denote with T_p, for $p = 1, \ldots, k^{k-2}$, the possible trees that can be formed by joining supernodes in a tree, by adding *k-1* edges between them. Since an edge between two components can connect any two nodes on each component, there are a total of $c_i c_j$ ways in which components i and j can be linked. Therefore, as each edge between components i and j in a tree T_p can be realized in $c_i c_j$ ways, each tree of supernodes, T_p, can be realized as $\prod_{(i,j)\in T_p} c_i c_j$ different trees on the set of nodes, from which (1) follows. □

Arranging the component sizes in ascending order, $c_1 \leq \ldots \leq c_k$, the number of trees, $|\mathcal{T}|$, can be bounded as

$$k^{k-2}(c_1)^{k-1} \prod_{j=2}^{k} c_j \leq |\mathcal{T}| \leq k^{k-2}(c_k)^{k-1} \prod_{j=1}^{k-1} c_j. \tag{2}$$

From Lemma 1 and (2), it follows that the number of possible trees scales exponentially with the number of observed components, and also grows with the components' sizes. Hence, solving the source localization problem by constructing each possible topology and then searching for the source node within each tree based on observed infection times is computationally very challenging even for small networks with only a few missing edges. In the next section we discuss an observer selection strategy that guarantees that a single node can be identified as the source despite the exponential number of trees that are consistent with the observed topology.

3. OBSERVABILITY OF A TREE NETWORK WITH MISSING EDGES

The infection times of the observers correspond to their distance to the source. Since the node distances are different for different trees that belong to the set $\mathcal{T}$, not all topologies are consistent with the observed infection times. The number of nodes and topologies that match the observed times and network structure depend on the number of missing edges, the placement of observers and the source. When the choice of observers is such that unambiguous source identification is guaranteed, for each possible source, we denote this forest as observable [3]. Now we review the necessary concepts.

A leaf in a forest or a tree is a vertex of degree 1. A path $i - j$ is a sequence of all different nodes starting from i and ending with j. The distance between two nodes i and j in a tree T is denoted as $d_T(i, j)$ and it represents the number of edges in the shortest path between them. Let L_T be the set of all leaves of a tree T and $L_{\mathcal{F}}$ the set of all leaves in a forest $\mathcal{F}$. If O denotes the set of observers $\{o_1, \ldots, o_r\}$, then $\boldsymbol{d}_T(i, O)$ is the r-vector of distances from node i to the set of observers $[d_T(i, o_1), \ldots, d_T(i, o_r)]$. In [3], it was shown that for a given graph G, unambiguous source localization

is possible if the set of observers forms a resolving set, i.e., when $\boldsymbol{d}_G(i,O) \neq \boldsymbol{d}_G(j,O)$ for all i,j pairs of nodes [7]. This concept can be extended to the case when some of the edges are missing. Correct source localization is possible if all nodes have different distance to the set of observers across all possible trees that correspond to the observed forest. We will show in Theorem 1 that if all the leaves of the forest are observed, the source can be correctly identified, but first we need the following lemmas.

Lemma 2. *All nodes in a tree T have unique distance to the set of leaves, i.e., $\boldsymbol{d}_T(i,L_T) \neq \boldsymbol{d}_T(j,L_T)$, $\forall i,j \in V$, $i \neq j$.*

Proof. The minimum resolving set of a tree T equals $|L_T| - |K_T|$, where K_T is a set of nodes of degree 3 or more that are connected by paths to one or more leaves [7]. Here we prove a weaker claim, needed for the following steps. The proof is by contradiction. Let us assume there exist two nodes i and j such that $\boldsymbol{d}_T(i,\ell) = \boldsymbol{d}_T(j,\ell)$, for all $\ell \in L_T$. Nodes $i,j \notin L_T$, as we would have $d_T(i,i) = 0 \neq d_T(i,j)$. Let us root the tree on an arbitrary leaf and let w be the lowest node in T that has both i and j as descendants. Since nodes i and j are at the same distance to the root, they are also at the same distance from w, i.e. $d_T(i,w) = d_T(j,w)$. As i is not a leaf, it has a degree of at least two, and hence it has a descendant leaf l on the path that does not include w. As $l \in L_T$, by assumptions $d_T(i,l) = d_T(j,l)$ holds. Then we have $d_T(l,w) = d_T(l,i) + d_T(i,w) = d_T(l,j) + d_T(j,w)$. Now we have that between nodes l and w there exist two equal length paths, one passing through i and the other through j. As there is only one path in a tree between any pair of nodes, we arrive at a contradiction and this proves the claim. □

Lemma 3. *All nodes in a forest $\mathcal{F}$ have unique distance to the set of forest leaves within each tree that could be constructed from the forest, i.e. $\boldsymbol{d}_T(i,L_\mathcal{F}) \neq \boldsymbol{d}_T(j,L_\mathcal{F})$ for all i,j pairs of nodes, $i \neq j$, and for all $T \in \mathcal{T}$.*

Proof. Connecting all the subtrees in a forest into a single tree by adding new edges does not create new leaves. Hence, $L_T \subseteq L_\mathcal{F}$, for any $T \in \mathcal{T}$. From Lemma 2, we have $\boldsymbol{d}_T(i,L_T) \neq \boldsymbol{d}_T(j,L_T)$, from which $\boldsymbol{d}_T(i,L_\mathcal{F}) \neq \boldsymbol{d}_T(j,L_\mathcal{F})$ follows. □

Theorem 1. *All nodes in a forest $\mathcal{F}$ have unique distance to the set of forest leaves across all trees that could be constructed from the forest, i.e. $\boldsymbol{d}_{T_1}(i,L_\mathcal{F}) \neq \boldsymbol{d}_{T_2}(j,L_\mathcal{F})$ for all i,j pairs of nodes, $i \neq j$, and for all $T_1,T_2 \in \mathcal{T}$.*

Proof. Since Lemma 3 states that no two nodes can have the same distance to the forest leaves within the same fixed tree, now we only need to prove that the same holds for any two different trees from the class of $\mathcal{T}$. The proof is by contradiction. Let us assume there exist two different nodes i and j, and two different trees $T_1,T_2 \in \mathcal{T}$, such that $\boldsymbol{d}_{T_1}(i,L_\mathcal{F}) = \boldsymbol{d}_{T_2}(j,L_\mathcal{F})$. Let V_c denote the set of nodes from component subtree c, for $c = 1,\dots k$. With the same argument as in Lemma 2, nodes $i,j \notin L_\mathcal{F}$. Since the leaves of each component are included in $L_\mathcal{F}$, Lemma 2 applies, and all the nodes on the same component have different distances to the set $L_\mathcal{F}$. Therefore nodes i and j can only belong to two different components. Let us assume $i \in V_a$ and $j \in V_b$. Let p and q be two nodes such that the components a and b in tree T_2 are connected by path $p-q$, with $p \in V_b$ and $q \in V_a$. Node p can be node j and q can be node i, or any other node, and between nodes p and q there may be just an edge or a path through some other subtree. If q is a leaf, let $l = q$, otherwise let l be an arbitrary leaf on the subtree of i. Since $l \in L_\mathcal{F}$, from our assumptions we have

$$\begin{aligned} d_{T_2}(j,l) &= d_{T_2}(j,p) + d_{T_2}(p,q) + d_{T_2}(q,l) \\ &= d_{T_1}(i,l) = d_{T_2}(i,l). \end{aligned} \tag{3}$$

The last equality comes from the fact that $l,i \in V_a$ and the distances within the subtree are the same, regardless of which edges are added between the subtrees. Since $d_{T_2}(p,q) \geq 1$, from (3) we have that

$$d_{T_2}(j,l) > d_{T_2}(q,l). \tag{4}$$

If node i was on the path between q and l, for the case of $q \neq l$, we would have $d_{T_2}(q,l) = d_{T_2}(q,i) + d_{T_2}(i,l) = d_{T_2}(q,i) + d_{T_2}(j,l)$, which contradicts (4). Therefore, we assume i is not on the path $q-l$, for the case $q \neq l$. As i has a degree greater than 1, it has a descendant leaf, node f, that is not on the path $i-q$. Now, the distance between two leaves f and l equals

$$\begin{aligned} d_{T_2}(f,l) &= d_{T_2}(f,i) + d_{T_2}(i,q) + d_{T_2}(q,l) \\ &= d_{T_2}(f,i) + d_{T_2}(i,l) = d_{T_1}(f,i) + d_{T_1}(i,l) \\ &= d_{T_2}(f,j) + d_{T_2}(j,l). \end{aligned}$$

The last equality comes from the assumption that node j has the same distance to all the leaves in T_2, as node i in tree T_1. Finally, from the above we have that between two nodes f and l, there exist two equal length paths, one passing through node i, and the other through node j in a tree T_2. Since there is only one path between any pair of nodes in a given tree, we again arrive at contradiction and this completes the proof. □

4. SOURCE LOCALIZATION

Localization of a source in a tree network, when some edges are unknown, under the assumption of deterministic diffusion, can be formulated as a binary integer linear program. For each node s in a network, a multicommodity flow problem with side constraints [8] is formulated, assuming node s is the source, and its feasibility is checked. If the problem is feasible, node s is a viable source suspect. A feasible solution implies that there exists a tree consistent with the observed forest, such that the distances from the assumed source to the

observers are equal to the observed infection times. Assuming the optimization problem is feasible for a certain node, by solving it, the tree topology is recovered. There might exist more than one tree with the required distances for each source suspect. Solving the optimization problem gives us the structure of at most one tree, as the main goal is to verify the possibility of a node being the source, and for this, the existence of a single tree is sufficient.

The incidence matrix B of a graph with m edges and n nodes is an $n \times m$ matrix, with entry b_{ij} equal to 1, if node i is the head of the edge j, equal to -1 if node i is the tail of the edge j and zero otherwise. Since the graphs we are interested in are undirected, the direction of the edges can be set arbitrarily, as later we will consider each edge twice, once for each direction, as it is typically done for similar network optimization problems [8]. Let us denote with E_p a set of all possible missing edges comprising only edges between the nodes of different components. With B_{ext} we denote an extended incidence matrix, where the edges are from the set of known edges, as well as from the set of possible missing edges E_p. With $\tilde{B} = [B_{ext}, -B_{ext}]$ we denote an incidence matrix, where we consider all the edges of B_{ext} in both directions. The set E_j is a set of edges that are incident with the observer j, and similarly, the set E_s is a set of edges that are incident with the assumed source s. The set of edges that connect components a and b is denoted as $E_{a,b}$, for $a, b = 1, \ldots, k$ and $a \neq b$.

An optimization variable in our formulation is a binary vector, $\boldsymbol{x} \in \mathbb{R}^m$, whose i-th entry is set to 1 if edge i, from matrix B_{ext}, is selected to be a part of the final constructed tree. For each observer node j, an auxiliary binary optimization variable $\boldsymbol{x}_j \in \mathbb{R}^{2m}$ is used to construct a path between the observer j and the assumed source s. If its i-th entry equals 1, then this implies that edge i from matrix $\tilde{B}$ is a part of the constructed path. The infection time of the observer node j is denoted as t_j. With $\mathbf{1}$ we denote a vector where all entries are equal to 1, while $\boldsymbol{e}_i$ denotes a column vector where all entries are equal to 0 except for the i-th entry, which equals 1. Matrix A has the structure $[I_m, I_m]$, where I_m is an identity matrix of size m. Finally, we state a binary integer program that verifies whether a node s could be the source of diffusion.

$$\min \ \mathbf{1}^T \boldsymbol{x} \tag{5}$$

subject to

$$\tilde{B}\boldsymbol{x}_j = \boldsymbol{e}_j - \boldsymbol{e}_s \tag{6}$$

$$\mathbf{1}^T \boldsymbol{x}_j = t_j \tag{7}$$

$$A\boldsymbol{x}_j \leq \mathbf{1} \tag{8}$$

$$\sum_{p \in E_j} \boldsymbol{e}_p^T A \boldsymbol{x}_j \leq 1 \tag{9}$$

$$\sum_{p \in E_s} \boldsymbol{e}_p^T A \boldsymbol{x}_j \leq 1 \tag{10}$$

$$A\boldsymbol{x}_j \leq \boldsymbol{x} \tag{11}$$

$$\sum_{o \in O} A\boldsymbol{x}_o \geq \boldsymbol{x} \tag{12}$$

$$\boldsymbol{x} \leq \mathbf{1} \tag{13}$$

$$\sum_{p \in E_{a,b}} \boldsymbol{x}(p) \leq 1 \tag{14}$$

$$\sum_{p \in E_p} \boldsymbol{x}(p) \leq k - 1 \tag{15}$$

$$\boldsymbol{x}_j \in \{0,1\}^{2m} \tag{16}$$

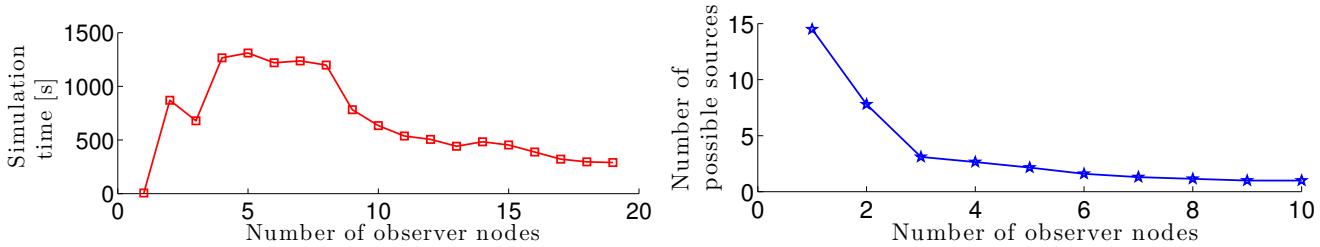
$$\forall j = 1, \ldots, r$$

$$\forall a, b = 1, \ldots, k, \ a \neq b$$

Constraint (6) sets the path between each observer node j and the assumed source s [8], while constraint (7) makes the length of that path equal to the infection time. Each observer - source path should not include the same edge in two different directions, and this is set by (8). Additionally, observer j and the source s should only be the terminal points of the path, which is constrained by (9) and (10), respectively. The paths are directed from the observers to the source, and therefore constructed from the edges of matrix $\tilde{B}$, which includes each edge in both directions. However, the goal is to construct an undirected tree from the union of these paths. By (11)–(13), an edge is used to construct a tree if it is in at least one observer-source path, regardless of the direction in which it was used. Hence the dimension of $\boldsymbol{x}$, a variable that represents edges in the final tree is only half of the dimension of the variables $\boldsymbol{x}_j$ that represent the paths. Constraint (14) ensures that the constructed union of paths contains at most one edge that connects two components, in order to avoid inter-component loops. Since there are k components, at most *k-1* edges can be added from the set of all the possible edges, and this is the role of the constraint (15). Finally, (16) constrains the variables $\boldsymbol{x}_j$ to be binary. Explicit enforcement for the binary structure of $\boldsymbol{x}$ is not necessary, as it is formed through constraints (11)–(13) on binary variables $\boldsymbol{x}_j$.

The above formulation does not require that the final constructed topology include all components, but rather only those that contain observers. Of course, by constructing paths some components without any observers might be included, but it is not a requirement. Hence, in the case of observers being chosen from only a small number of components, it is possible that constructed paths might have intra-component cycles resulting in a non-tree final topology. In this case, a node might be falsely assumed to be a suspect source. However, then it is possible to use an algorithm for detecting cycles in the final topology to discard non-tree solutions.

Solving the source localization problem as a binary optimization problem is computationally expensive. However, it does not require explicit construction and storage of an exponential number of trees. In case a sufficient condition given by Theorem 1 is satisfied, as soon as the problem is feasible for one node, the solving process can be terminated, as the solution is unique.



(a) Time required to solve (5)

(b) Number of possible solutions to (5)

Fig. 1. The effect of the number of observers on the optimization problem (5) for a 20 node tree with two unobserved edges.

5. SIMULATION RESULTS

We illustrate the effect of the number of observers on the source localization problem for a tree network of 20 nodes with two unobserved edges. From this partially observed tree, $5,880$ trees can be constructed by adding 2 edges. The total time needed to verify the feasibility of the linear binary program (5) for each node in a network was recorded, as well as the number of nodes for which the problem was feasible. These results were averaged over all possible sources and simulations were repeated for different numbers of observers, selected sequentially from each of the 3 components sized 6, 7 and 7 nodes. During selection, priority was given to 11 leaves. Figure 1 (a) shows that initially the time needed to solve (5) dramatically grows with the number of observers, while Figure 1 (b) shows an opposite trend in the number of possible solutions. Initially, with almost no constraints, a tree consistent with the observations can quickly be constructed for many nodes. Including more observers increases the number of constraints and more time is needed to find such a tree or conclude that the problem is unfeasible. The rate by which the number of suspects decreases with the number of observed leaves dramatically drops after including at least one node from each component. When the number of observers reaches 9, the network becomes observable. Around that point, correspondingly, the solving time starts to decrease with the number of observers. Now, only one node can be the source suspect, and for all the other nodes, the problem can relatively quickly be classified as unfeasible due to an increasing number of constraints.

6. CONCLUSION

We have evaluated the number of topologies to which a partially observed tree might correspond, demonstrating that the number of possible trees scales exponentially with the number of missing edges. We showed that in order for the source to be localized correctly, regardless of its position in the network, all nodes should have unique distance vectors to the set of the observer nodes, in all the corresponding tree networks. We proved that observing the set of leaf nodes of the forest is sufficient for the source to be identified in a simple propagation scenario in tree networks. Finally, we formulated the problem of source localization from the infection times of a subset of nodes, with deterministic propagation model and partially known tree topology, as a binary linear integer optimization problem. Although, computationally intensive, the proposed formulation does not require explicitly enumerating an exponential number of trees that might correspond to the incompletely observed network.

REFERENCES

[1] D. Shah and T. Zaman, "Rumors in a network: who's the culprit?," *IEEE Trans. on Information Theory*, 2011.

[2] P. Pinto, P. Thiran, and M. Vetterli, "Locating the source of diffusion in large-scale networks," *Physical Review Letters*, August 2012.

[3] S. Zejnilovic, J. Gomes, and B. Sinopoli, "Network observability and localization of the source of diffusion based on a subset of nodes," *Allerton*, pp. 847– 852, 2013.

[4] W. Luo, W. P. Tay, and M. Leng, "Identifying infection sources and regions in large networks," *IEEE Transactions on Signal Processing*, vol. 61, no. 11, pp. 2850–2864, June 2013.

[5] M. Gomez-Rodriguez, J. Leskovec, and A. Krause, "Inferring networks of diffusion and influence," *ACM Transactions on Knowledge Discovery from Data*, vol. 5, no. 4, pp. 21–37, February 2012.

[6] M. Kim and J. Leskovec, "The network completion problem: Inferring missing nodes and edges in networks," *SDM*, pp. 47–58, 2011.

[7] G. Chartrand, L. Eroh, M. A. Johnson, and O. R. Oellermann, "Resolvability in graphs and the metric dimension of a graph," *Discrete Applied Mathematics*, vol. 105, pp. 99–113, 2000.

[8] Dimitri P. Bertsekas, *Network Optimization: Continuous and Discrete Models*, Athena Scientific, 1998.