Minimum-sized Positive Influential Node Set Selection for Social Networks: Considering Both Positive and Negative Influences

Jing (Selena) He†, Shouling Ji‡, Xiaojing Liao‡, Hisham M. Haddad†, Raheem Beyah‡

†*Department of Computer Science, Kennesaw State University, Kennesaw, Georgia 30144, USA*
‡*School of Electrical and Computer Engineering, Georgia Institute of Technology, Atlanta, GA 30308, USA*
Email: jhe4@kennesaw.edu, sji@gatech.edu, xliao@gatech.edu, hhaddad@kennesaw.edu, rbeyah@ece.gatech.edu

***Abstract*—Social networks are important mediums for spreading information, ideas, and influences among individuals. Most of existing research work focus on understanding the characteristics of social networks, investigating spreading information through the "word of mouth" effect of social networks, or exploring social influences among individuals and groups. However, most of existing work ignore negative influences among individuals or groups. Motivated by alleviating social problems, such as drinking, smoking, gambling, and influence spreading problems (e.g., promoting new products), we take both positive and negative influences into consideration and propose a new optimization problem, named the Minimum-sized Positive Influential Node Set (MPINS) selection problem, to identify the minimum set of influential nodes, such that every node in the network can be positively influenced by these selected nodes no less than a threshold θ. Our contributions are threefold. First, we propose a new optimization problem MPINS, which is investigated under the independent cascade model considering both positive and negative influences. Moreover, we claim that MPIMS is NP-hard. Subsequently, we present a greedy approximation algorithm to address the MPINS selection problem. Finally, to validate the proposed greedy algorithm, extensive simulations are conducted on random Graphs representing small and large size networks.**

I. Introduction

A social network (e.g., Facebook, Google+, and MySpace) is composed of a set of nodes (such as individuals or organizations) that share the same interest or purpose. The network provides a powerful medium of communication for sharing, exchanging, and disseminating information, and for spreading influence beyond the traditional social interactions. Ever since social networks came to exist, they significantly enlarge our social circles, and they become a bridge to connect our daily physical life and the virtual web space. With the emergence of social applications (such as Flickr, Wikis, Netflix, and Twitter), there has been tremendous interests in how to effectively utilize social networks to spread ideas or information within a community [1]–[4]. Capturing the dynamics of a social network is a complex problem that requires an approach to analyze the dynamics of positive and negative social influences resulting from individual-to-individual and individual-to-group interactions. In a social network, individuals may have both positive and negative influence on each other. For example, within the context of gambling, a gambling insulator has positive influence on his friends/neighbors. Moreover, if many of an individual's friends are gambling insulators, the aggregated positive influence is exacerbated. However, an individual might turn into a gambler, who brings negative impact on his friends/neighbors. For example, in the social network shown in Fig. 1, the social influences are represented by weights assigned to edges. If Jack and Bob (marked by the person with the red tie) are gambling insulators, then they have positive influence on their neighbors. To be specific, Jack has a positive influence on Chris with probability 60%. Similarly, Mary has a negative influence on Tony with probability 90%, since Mary is a gambler. Moreover, in the community shown in Fig. 1, only Tony has not been influenced by any gambling insulator. Hence, motivated by alleviating social problems, such as drinking, smoking, and gambling, this work aims to find a Minimum-sized Positive Influential Node Set (MPINS), which positively influences every individual in a social network no less than a predefined threshold θ.

One application of MPINS is described as follows. A community wants to implement a smoking intervention program. To be cost effective and get the maximum effect, the community wishes to select a small number of influential individuals in the community to attend a quit-smoking campaign. The goal is that all other individuals in the community will be positively influenced by the selected users. Constructing an MPINS is helpful to alleviate the aforementioned social problem, and it is also helpful to promote new products in the social network. Consider the following scenario as another motivation example. A small company wants to market a new product in a community. To be cost effective and get maximum profit, the company would like to distribute sample products to a small number of initially chosen influential users in the community. The company wishes that these initial users would like the product and positively influence their friends in the community. The goal is to have other users in the community be positively influenced by the selected users no less than θ eventually. To sum up, the specific problem we investigate in this work

978-1-4799-3214-6/13/$31.00 ©2013 IEEE

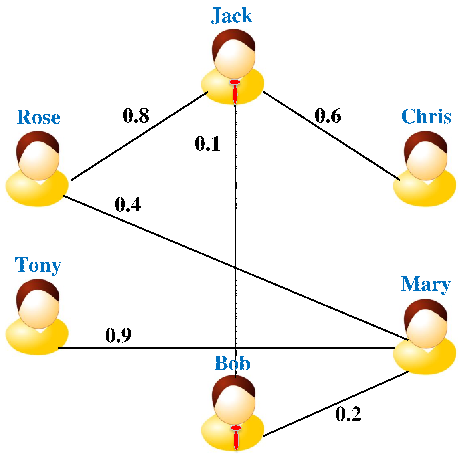


Figure 1: A social network with social influences on edges.

is the following: given a social network and a threshold θ, identify a minimum-sized subset of individuals in the network such that the subset can result in a positive influence on every individual in the network no less than θ.

A related work [5] to our research finds a minimum-sized *Positive Influence Dominating Set* (PIDS) D, so that every other node has at least half of its neighbors in D. In this work, only positive influence from neighbors is considered while the negative influence is totally ignored. Moreover, the authors in [5] studied the PIDS selection problem under the *deterministic linear threshold model*, in which the influence from a pair of nodes is represented by a weight and an individual can be positively influenced when the sum of the weights exceeds a pre-determined threshold. To be specific, the authors in [5] assume that the influence of a pair of nodes is always 1, and an individual can be positively influenced when at least half of its neighboring nodes are in D. Nevertheless, the deterministic linear threshold model cannot fully characterize the social influence between each pair of nodes in a real social network. This is because, in the physical world, the strength of the social influence between different pairs of nodes may be different and is actually a probabilistic value [6]–[9]. Hence, we explore the MPINS selection problem under the *independent cascade model* considering both positive and negative influences, where individuals can positively or negatively influence their neighbors with certain probabilities.

In this paper, first we formally define the MPINS problem and then propose a greedy approximation algorithm to solve it. Particularly, the main contributions of this work are summarized as follows:

1) Taking into consideration both positive and negative influences, we introduce a new optimization problem, named the Minimum-sized Positive Influential Node Set (MPINS) selection problem, for social networks, to identify the minimum-sized set of influential nodes, that could positively influence every node in the network no less than a pre-defined threshold θ. We claim that it is a NP-hard problem under the independent cascade model.
2) We define a contribution function using a greedy approximation algorithm, called MPINS-GREEDY, to address the MPINS selection problem. The correctness of the proposed algorithm is also analyzed in the paper.
3) We conduct extensive simulations to validate the proposed algorithm. The simulation results show that the proposed greedy algorithm works well to solve the MPINS selection problem. More importantly, the solutions obtained by the greedy algorithm are very close to the optimal solution of MPINS in small scale networks.

The rest of the paper is organized as follows: Section II introduces the network model and then formally defines the MPINS selection problem. The greedy algorithm and theoretical analysis of its correctness are presented in Section III. The simulation results are presented in Section IV to validate the proposed algorithm. Finally, the paper is concluded in Section V.

II. Problem Definition and Hardness Analysis

In this section, we first introduce the network model. Subsequently, we formally define the MPINS selection problem. Finally we make some remarks on the proposed problem.

A. Network Model

We model a social network by an undirected graph $\mathcal{G}(\mathcal{V}, \mathcal{E}, \mathcal{P}(\mathcal{E}))$, where $\mathcal{V}$ is the set of n nodes, denoted by u_i, and $0 \leq i < n$. i is called the node ID of u_i. An undirected edge $(u_i, u_j) \in \mathcal{E}$ represents a social tie between the pair of nodes. $\mathcal{P}(\mathcal{E}) = \{p_{ij} \mid \text{if } (u_i, u_j) \in \mathcal{E}, 0 < p_{ij} \leq 1, \text{ else } p_{ij} = 0\}$, where p_{ij} indicates the social influence between nodes u_i and u_j[1]. It is worth mentioning that the social influence can be categorized into two groups: positive influence and negative influence. For example, in the smoking intervention program, an individual initially chosen to attend the quit-smoking campaign has positive influence on all neighbors; while the smokers definitely have negative influences on their neighbors. The formal definition of positive influence and negative influence will be given in Definition II.5 and Definition II.6. For simplicity, we assume the links are undirected (bidirectional), which means two linked nodes have the same social influence (*i.e.*, p_{ij} value) on each other.

B. Problem Definition

The objective of the MPINS selection problem is to identify a subset of influential nodes as the initial nodes. Such that, all the other nodes in a social network can

[1]This model is reasonable since many empirical studies have analyzed the social influence probabilities between nodes [6]–[9].

be positively influenced by these nodes no less than a threshold θ. For convenient, we call the initial nodes as *active nodes*, otherwise, *inactive nodes*. Therefore, how to define *positive influence* is critical to solving the MPINS selection problem. In the following, we first formally define some terminologies, and then give the definition of the MPINS selection problem.

Definition II.1. *Positive Influential Node Set ($\mathcal{I}$).* For social network $\mathcal{G}(\mathcal{V}, \mathcal{E}, \mathcal{P}(\mathcal{E}))$, the positive influential node set is a subset $\mathcal{I} \subseteq \mathcal{V}$, such that all the nodes in $\mathcal{I}$ are initially selected to be the active nodes.

Definition II.2. *Neighboring Set ($\mathcal{B}(u_i)$).* For social network $\mathcal{G}(\mathcal{V}, \mathcal{E}, \mathcal{P}(\mathcal{E}))$, $\forall u_i \in \mathcal{V}$, the neighboring set of u_i is defined as:

$$\mathcal{B}(u_i) = \{u_j \mid (u_i, u_j) \in \mathcal{E}, p_{ij} > 0\}.$$

Definition II.3. *Active Neighboring Set ($\mathcal{A}^{\mathcal{I}}(u_i)$).* For social network $\mathcal{G}(\mathcal{V}, \mathcal{E}, \mathcal{P}(\mathcal{E}))$, $\forall u_i \in \mathcal{V}$, the active neighboring set of u_i is defined as:

$$\mathcal{A}^{\mathcal{I}}(u_i) = \{u_j \mid u_j \in \mathcal{B}(u_i), u_j \in \mathcal{I}\}.$$

Definition II.4. *Non-active Neighboring Set ($\mathcal{N}^{\mathcal{I}}(u_i)$).* For social network $\mathcal{G}(\mathcal{V}, \mathcal{E}, \mathcal{P}(\mathcal{E}))$, $\forall u_i \in \mathcal{V}$, the non-active neighboring set of u_i is defined as:

$$\mathcal{N}^{\mathcal{I}}(u_i) = \{u_j \mid u_j \in \mathcal{B}(u_i), u_j \notin \mathcal{I}\}.$$

Following Definition II.3 and Definiton II.4, we know that the set $\mathcal{A}^{\mathcal{I}}(u_i)$ includes all the active neighboring nodes of u_i and the set $\mathcal{N}^{\mathcal{I}}(u_i)$ includes all the non-active neighboring nodes. How those neighboring nodes collaboratively influence each individual is critical to solving the MPINS selection problem. Next, we define some other terminologies as follows:

Definition II.5. *Positive Influence ($p_{u_i}(\mathcal{A}^{\mathcal{I}}(u_i))$).* For social network $\mathcal{G}(\mathcal{V}, \mathcal{E}, \mathcal{P}(\mathcal{E}))$, a node $u_i \in \mathcal{V}$, and a positive influential node set $\mathcal{I}$, we define a joint influence probability of $\mathcal{A}^{\mathcal{I}}(u_i)$ on u_i, denoted by $p_{u_i}(\mathcal{A}^{\mathcal{I}}(u_i))$ as

$$p_{u_i}(\mathcal{A}^{\mathcal{I}}(u_i)) = 1 - \prod_{u_j \in \mathcal{A}^{\mathcal{I}}(u_i)} (1 - p_{ij}).$$

Definition II.6. *Negative Influence ($p_{u_i}(\mathcal{N}^{\mathcal{I}}(u_i))$).* For social network $\mathcal{G}(\mathcal{V}, \mathcal{E}, \mathcal{P}(\mathcal{E}))$, a node $u_i \in \mathcal{V}$, and a positive influential node set $\mathcal{I}$, we define a joint influence probability of $\mathcal{N}^{\mathcal{I}}(u_i)$ on u_i, denoted by $p_{u_i}(\mathcal{N}^{\mathcal{I}}(u_i))$ as

$$p_{u_i}(\mathcal{N}^{\mathcal{I}}(u_i)) = 1 - \prod_{u_j \in \mathcal{N}^{\mathcal{I}}(u_i)} (1 - p_{ij}).$$

Definition II.7. *Ultimate Influence ($\varrho^{\mathcal{I}}(u_i)$).* For social network $\mathcal{G}(\mathcal{V}, \mathcal{E}, \mathcal{P}(\mathcal{E}))$, a node $u_i \in \mathcal{V}$, and a positive influential node set $\mathcal{I}$, we define an ultimate influence of $\mathcal{B}(u_i)$ on u_i, denoted by $\varrho^{\mathcal{I}}(u_i)$ as

$$\varrho^{\mathcal{I}}(u_i) = p_{u_i}(\mathcal{A}^{\mathcal{I}}(u_i)) - p_{u_i}(\mathcal{N}^{\mathcal{I}}(u_i)).$$

Moreover, if $\varrho^{\mathcal{I}}(u_i) < 0$, we set $\varrho^{\mathcal{I}}(u_i) = 0$. If $\varrho^{\mathcal{I}}(u_i) \geq \theta$, where $0 < \theta < 1$ is a pre-defined threshold, then u_i is said to be *positively influenced*. Otherwise, u_i is not positively influenced.

Here, we assume that the *ultimate influence* of any active nodes is greater than or equal to θ, *i.e.*, $\forall u_i \in \mathcal{I}, \varrho^{\mathcal{I}}(u_i) \geq \theta$. Moreover, if $\mathcal{I} = \emptyset$, then $\forall u_i \in \mathcal{V}, \varrho^{\mathcal{I}}(u_i) = 0$. Finally, we are ready to give the formal definition of the MPINS selection problem.

Definition II.8. *Minimum-sized Positive Influential Node Set (MPINS).* For social network $\mathcal{G}(\mathcal{V}, \mathcal{E}, \mathcal{P}(\mathcal{E}))$, the MPINS selection problem is to find a minimum-sized positive influential node set $\mathcal{I} \subseteq \mathcal{V}$, such that $\forall u_i \in \mathcal{V} \setminus \mathcal{I}$, u_i is positively influenced, *i.e.*,

$$\varrho^{\mathcal{I}}(u_i) = p_{u_i}(\mathcal{A}^{\mathcal{I}}(u_i)) - p_{u_i}(\mathcal{N}^{\mathcal{I}}(u_i)) \geq \theta,$$

where $0 < \theta < 1$.

In this paper, we study the MPINS selection problem under independent cascade model. First, we analyze the complexity of the problem, which is NP-hard. The authors in [5] prove that the minimum-sized Positive Influence Dominating Set (PIDS) selection problem, which to guarantee that every other node has at least half of its neighbors in PIDS, is NP-hard. As we mentioned in Section I, PIDS is investigated under deterministic linear threshold model taking only positive influences into consideration. Hence, we claim that MPINS is NP-hard, since it is studied under a more general and practical scenario, *i.e.*, under independent cascade mode considering both positive and negative influences. Subsequently, we propose a greedy algorithm, called MPINS-GREEDY, to solve the problem.

III. Greedy algorithm and Performance Analysis

Before introducing MPINS-GREEDY, we first define a useful contribution function as follows:

Definition III.1. *Contribution Function ($f(\mathcal{I})$).* For social network $\mathcal{G}(\mathcal{V}, \mathcal{E}, \mathcal{P}(\mathcal{E}))$, and a positive influential node set $\mathcal{I}$, the contribution function of $\mathcal{I}$ to $\mathcal{G}$ is defined as:

$$f(\mathcal{I}) = \sum_{i=1}^{|\mathcal{V}|} \max[\min(\varrho^{\mathcal{I}}(v_i), \theta), 0].$$

Based on the defined contribution function, we propose a heuristic algorithm, which has two phases.

1) We find the node u_i with the maximum $f(\mathcal{I})$, where $\mathcal{I} = \{u_i\}$. After that, we select a Maximal Independent Set (MIS)[2] induced by a Breadth-First-Search

[2]MIS can be defined formally as follows: given a graph $G = (V, E)$, an Independent Set (IS) is a subset $I \subset V$ such that for any two vertex $v_1, v_2 \in I$, they are not adjacent, *i.e.*, $(v_1, v_2) \notin E$. An IS is called an MIS if we add one more arbitrary node to this subset, the new subset will not be an IS any more.

(BFS) ordering with respect to u_i as the root node [11]–[13].

2) We employ the pre-selected MIS, denoted by $\mathcal{M}$, as the initial active node set to perform the greedy algorithm MPINS-GREEDY as shown in Algorithm 1. MPINS-GREEDY starts from $\mathcal{I} = \mathcal{M}$. Each time, it adds the node having the maximum $f(\cdot)$ value into $\mathcal{I}$. The algorithm terminates when $f(\mathcal{I}) = |\mathcal{V}|\theta$.

Algorithm 1 MPINS-GREEDY Algorithm

Require: Social network $\mathcal{G}(\mathcal{V}, \mathcal{E}, \mathcal{P}(\mathcal{E}))$; a pre-defined threshold θ.

Initialize $\mathcal{I} = \mathcal{M}$
while $f(\mathcal{I}) < |\mathcal{V}|\theta$ **do**
  choose $u \in \mathcal{V} \setminus \mathcal{I}$ to maximize $f(\mathcal{I} \bigcup \{u\})$
  $\mathcal{I} = \mathcal{I} \bigcup \{u\}$
end while
return $\mathcal{I}$

To better understand the proposed heuristic algorithm, we use the social network represented by the graph shown in Fig. 2(a) to illustrate the selection procedure as follows. In this example, $\theta = 0.8$. Since u_1 has the maximum $f(\{u_i\})$ value, we construct a BFS tree rooted at u_1, as shown in Fig. 2(b). With the help of BFS ordering, we find the MIS set $\mathcal{M} = \{u_1, u_6\}$. Next, we go to the second phase to perform Algorithm 1.

1) First round: $\mathcal{I} = \mathcal{M} = \{u_1, u_6\}$.
2) Second round: we first compute $f(\mathcal{I} = \{u_1, u_2, u_6\}) = 4.45$, $f(\mathcal{I} = \{u_1, u_3, u_6\}) = 3.018$, $f(\mathcal{I} = \{u_1, u_4, u_6\}) = 3.65$, $f(\mathcal{I} = \{u_1, u_5, u_6\}) = 3.65$, and $f(\mathcal{I} = \{u_1, u_6, u_7\}) = 3.778$. Therefore, we have $\mathcal{I} = \{u_1, u_2, u_6\}$, which has the maximum $f(\mathcal{I})$ value. However, $f(\mathcal{I} = \{u_1, u_2, u_6\}) = 4.45 < 7 * 0.8 = 5.6$. Consequently, the selection procedure continues.
3) Third round: we first computer $f(\mathcal{I} = \{u_1, u_2, u_3, u_6\}) = 4.45$, $f(\mathcal{I} = \{u_1, u_2, u_4, u_6\}) = 5.6$, $f(\mathcal{I} = \{u_1, u_2, u_5, u_6\}) = 5.6$, and $f(\mathcal{I} = \{u_1, u_2, u_6, u_7\}) = 4.45$. Therefore, we have $\mathcal{I} = \{u_1, u_2, u_4, u_6\}$[3]. Since $f(\mathcal{I} = \{u_1, u_2, u_4, u_6\}) = 7*0.8 = 5.6$, the algorithm terminates and outputs set $\mathcal{I} = \{u_1, u_2, u_4, u_6\}$ as shown in Fig. 2(c), where black nodes represent the selected influential nodes.

It is easy to check that u_3, u_5 and u_7 are all positively influenced. Hence, the constructed $\mathcal{I}$ is a feasible solution for the MPINS selection problem.

The proposed algorithm starts searching from an MIS set ($\mathcal{M}$) instead of an empty set, so that the algorithm convergent time should be shorten. Next, we theoretically show the correctness of Algorithm 1 in the following theorem.

[3]If there is a tie on the $f(\mathcal{I})$ value, we use the node ID to break the tie.

Theorem 1. Algorithm 1 produces a feasible solution for the MPINS selection problem. To be specific,

1) Algorithm 1 terminates for sure.
2) $f(\mathcal{I}) = |\mathcal{V}|\theta$ if and only if $\mathcal{I}$ is a positive influential node set, such that every node (*i.e.*, $\forall u_i \in \mathcal{V}$) is positively influenced by nodes in $\mathcal{I}$ no less than θ.

Proof: For 1), Based on Algorithm 1, in each iteration, only one node is selected to be added into the output set $\mathcal{I}$. In the worst case, all nodes are added into $\mathcal{I}$ in the $|\mathcal{V}|$-th iteration. Then, $f(\mathcal{I}) = f(\mathcal{V}) = |\mathcal{V}|\theta$ and Algorithm 1 terminates and outputs $\mathcal{I} = \mathcal{V}$. Therefore, Algorithm 1 terminates for sure.

For 2), $\Rightarrow$: if $f(\mathcal{I}) = |\mathcal{V}|\theta$, then $\forall u_i \in \mathcal{V}$, $\varrho^{\mathcal{I}}(v_i) \geq \theta$ followed by Definition III.1. Therefore, all nodes in the network are positively influenced.

$\Leftarrow$: if $\forall u_i \in \mathcal{V}, \varrho^{\mathcal{I}}(v_i) \geq \theta$, then we obtain $\forall u_i \in \mathcal{V}, \min(\varrho^{\mathcal{I}}(v_i), \theta) = \theta$. According to the definition of the contribution function III.1, $f(\mathcal{I}) = \sum_{i=1}^{|\mathcal{V}|} \max[\min(\varrho^{\mathcal{I}}(v_i), \theta), 0] = |\mathcal{V}|\theta$.

Based on the above two aspects, Algorithm 1 must produce a feasible solution for the MPINS selection problem. ■

IV. Performance Evaluation

Currently, there is no existing work studying the MPINS selection problem under the independent cascade model. The simultion results of MPINS-GREEDY (denoted by MPINS) are compared to the related work [5] (denoted by PIDS), and the optimal solution of MPINS (obtained by exhaustive search, denoted by OPTIMAL). To ensure the fairness of comparison, the termination condition of the algorithm proposed in [5] is changed to find a PIDS, such that every node in the network is positively influenced no less than the same threshold θ in MPINS.

A. Simulation Setting

We build our own simulator to generate random graphs based on the random graph model $G(n, p) = \{G \mid G \text{ has } n \text{ nodes, and an edge between any pair of nodes is generated with probability } p\}$. For $G = (V, E) \in G(n, p)$, $u_i, u_j \in V$, and $(u_i, u_j) \in E$, the associated social influence $0 < p_{ij} \leq 1$ is randomly generated. For each specific setting, 100 instances are generated. The results are the average values of these 100 instances. Below, we show the simulation results under different scenarios.

B. Simulation Results

The objectives of MPINS and PIDS are both to minimize the size of the constructed subsets. In this subsection, we check the size of the solutions for MPINS, PIDS and OPTIMAL under different scenarios in random graphs. In this simulation, we consider the following tunable parameters: the network size n, the possibility to create an edge p in

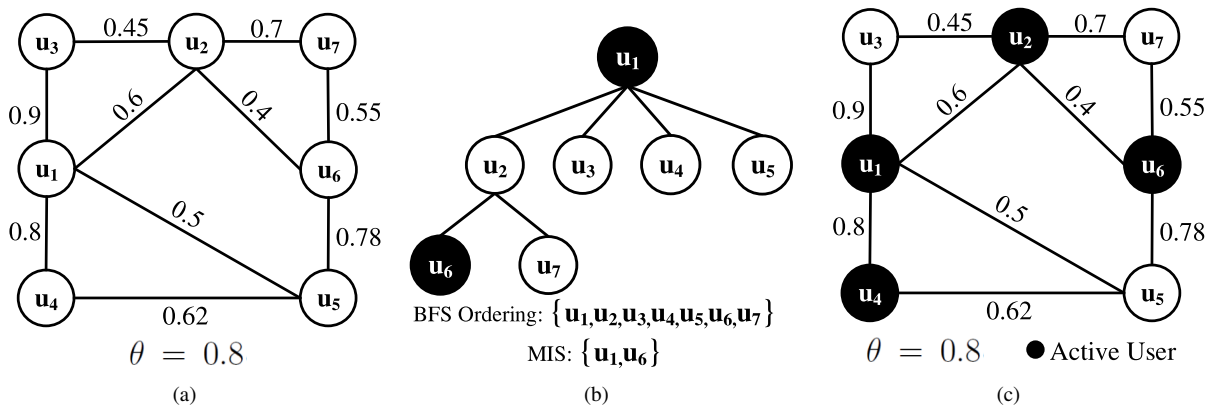


Figure 2: Illustration of MPINS-Greedy algorithm.

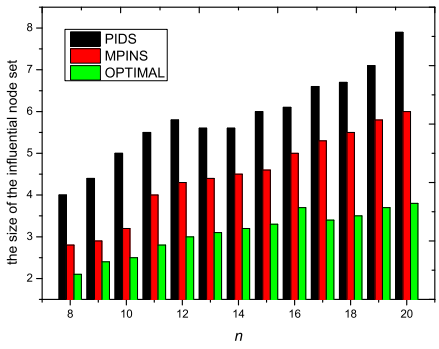
the random graph model $G(n, p)$, and the user pre-defined influence threshold θ. Since we adopt exhaustive search to find the optimal solution for MPINS, it is impractical to test on large scale networks. Hence, we first run a set of simulations on small scale networks of network size changing from 10 to 20 nodes, and the results are shown in Fig. 3.

The impacts of n, p, and θ on the size of the solutions for MPINS, PIDS, and OPTIMAL are shown in Fig. 3(a), (b), and (c), respectively. From Fig. 3(a), we can see that the sizes of the solutions for all three algorithms increase when n increases. This is because more nodes need to be influenced when the network size increases. Additionally, for a specific network size, PIDS produces larger sized solution than MPINS. This is because MPINS tries to find the most influential MIS of the network first, and then add the node which has the largest $f(\mathcal{I})$ value in each iteration, while PIDS gives the node with the largest degree the highest priority instead. However, a large degree does not necessarily imply high *ultimate influence* on the individuals in the network, since some neighbors may have high negative influences on the individuals. Moreover, MPINS selects an MIS first, which avoids the nodes selection bias to some specific region, so that more nodes need to be added into the subset to influence all the nodes in the network. Furthermore, we can see that the size of MPINS solution is very close to the optimal result. To be specific, on average, MPINS produces 1.07 more nodes than the optimal solution, while PIDS produces 3.75 more nodes than the optimal solution. The results imply that our proposed greedy algorithm MPINS-GREEDY can produce a very close approximation solution to the optimal solution in small scale networks.

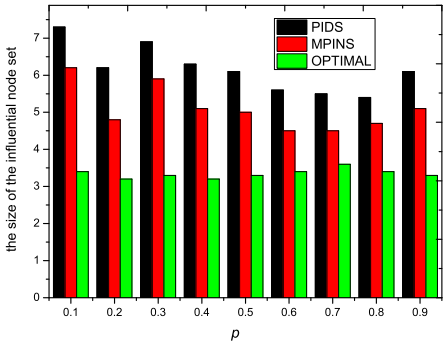
From Fig. 3(b), we can see that there is no obvious trend on the solution sizes for all three algorithms when p increases. This is because when p increases means more edges in the network, so that one specific node may have more negative or positive neighbors. In a very crowded network, it is hard to tell the pattern of the sizes of selected influential node sets. On the other hand, for a specific p, PIDS produces larger sized solutions than MPINS. This is because the objective of PIDS is not aimed to obtain the most influential and no-regional-biased nodes in the network. MPINS again can construct the solution with similar size of the optimal solution. On average, MPINS only produces 1.6 more nodes than the optimal solution, while PIDS produces 3.16 more nodes than the optimal solution.

From Fig. 3(c), we can see that the solutions sizes for all solutions increase when θ increases, since large θ value means that more nodes need to be put into the initial active node set to influence all other nodes. Furthermore, MPINS has similar performance with optimal, and has a better performance than PIDS since the greedy criterion of PIDS is node with highest degree first. On average, MPINS produces 1.3 more nodes than the optimal solutions, while the sizes of PIDS solutions are a little away from the optimal results. On average, PIDS produces 3.7 more nodes than the optimal solution. The reason is similar as we mentioned above.

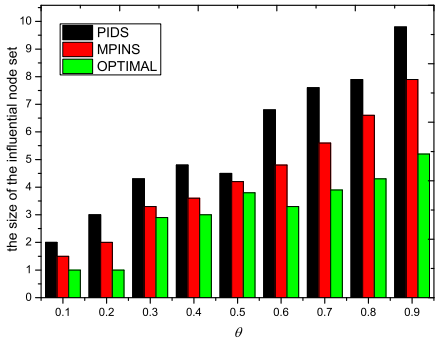
Additionally, we ran a set of simulations on large scale networks of network size changing from 100 to 1000 nodes. The impacts of n, p, and θ on MPINS and PIDS are shown in Fig. 4. From Fig. 4(a), we can see that the solution sizes of MPINS and PIDS are both increase when n increases. This is because more active influential nodes are needed for larger social networks. Moreover, with n increases, the difference between the sizes of MPINS and PIDS increases also. At a specific n, MPINS can find a positive influential node set that is smaller than that of PIDS. This is because in small scale network (*i.e.*, $n < 500$), the initial active node set size is small (no more than 30 from Fig. 4). Hence, the differences between the two methods are not very obvious. However, in large scale network, say $n = 1000$, our proposed MPINS has a significant improvement on the size of the initial active node set compared to PIDS. The reason is similar as we mentioned earlier. On average, MPINS produces a positive



(a)



(b)



(c)

Figure 3: The size of solutions on small scale networks. The default setting are $n = 15$, $p = 0.5$, and $\theta = 0.5$.

influential node set of size 22.5 less than PIDS.

From Fig. 4(b), we can see that the solution sizes of PIDS and MPINS decrease when p increases. p increases means that the number of edges in the network increases, which further implies that the average number of neighbors of each node increases. Hence, one selected active node may influence more nodes when p increases. For a specific p, PIDS again produces larger sized solution than MPINS. When the solution size is small, it is hard to tell which method outperform the other. However, MPINS clearly outperforms PIDS on the sparse network, such as $p = 0.1$. It is worth to mention that the decreasing trend of PIDS is very fast when p increases. This is because when p is small, the degrees of all nodes are small. Hence, PIDS may find a solution through many iterations till it find a solution satisfying that every node in the network is positively influenced by the solution no less than θ. When p is large, larger degree nodes could be added into the solution first, so that PIDS might terminate quicker, resulting in a smaller sized positive influential node set. On average, PIDS produces 31.52 more nodes than MPINS.

Similar to Fig. 3(c), we can see in Fig. 4(c) that the solution sizes for PIDS and MPINS increase when θ increases. Moreover, PIDS outputs more and more nodes than MPINS when θ increases. On average, PIDS produces 23.2 more nodes than that of MPINS.

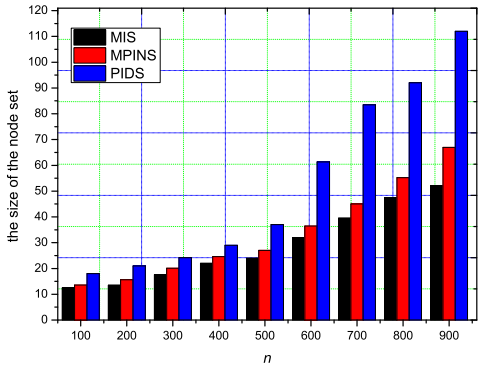


Figure 5: The size of the node set: The default settings are $p = 0.5$, and $\theta = 0.5$.

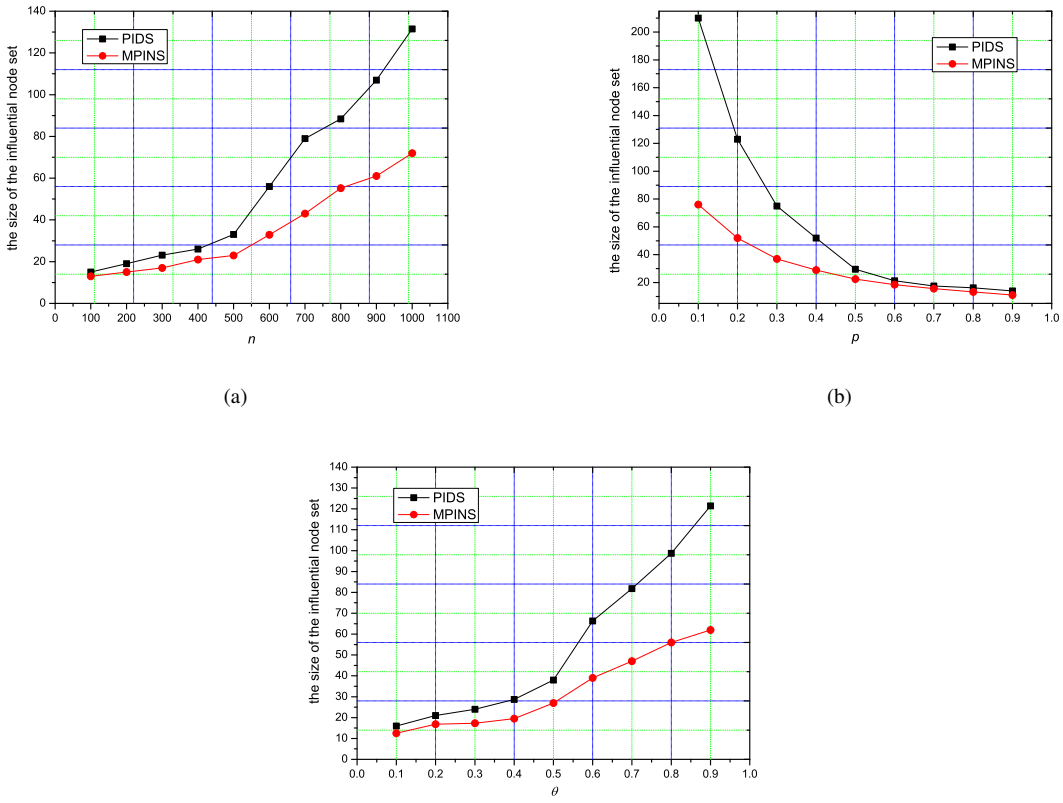


Figure 4: The size of solutions on large scale networks: The default settings are $n = 15$, $p = 0.5$, and $\theta = 0.5$.

One big difference between MPINS and PIDS is that MPINS starts the greedy search on a pre-selected influential MIS set, while PIDS starts searching from an empty set. Moreover, PIDS uses node degree as the greedy search criterion which might lead to finding some regional-biased nodes, so that the final size of the solution may be increased. Our proposed MPINS method selects a Maximal Independent Set (MIS) first which avoids the aforementioned dilemma. Fig. 5, Fig. 6, and Fig. 7 try to compare the size of MIS, MPINS, and PIDS when n, p, and θ changes. All these results indicate that after selecting an influential MIS, only a small number of iterations of MPINS-GREEDY are needed to find a solution for MPINS. However, the number of iterations for the greedy algorithm proposed for solving PIDS is considerable larger compared to the number of iterations of MPINS-GREEDY.

V. Conclusion

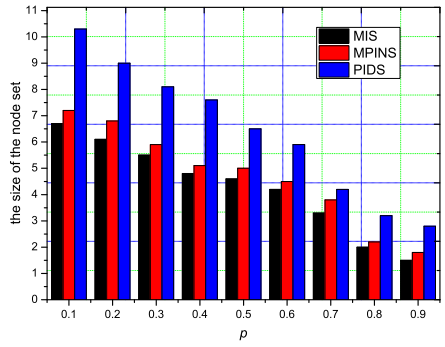
In this paper, we study the Minimum-sized Positive Influential Node Set (MPINS) selection problem which has useful commercial applications in social networks. First, we claim that MPINS is NP-hard under the independent cascade model, and then propose a greedy algorithm, called MPINS-GREEDY, to solve the problem. Subsequently, we validate our proposed algorithm through simulations on random graphs repersenting small size and large size networks. The simulation results indicate that MPINS-GREEDY can construct smaller initial active node sets than the latest related work PIDS [5]. Moreover, for small scale network, MPINS-GREEDY performance is close to the optimal solution of MPINS. Furthermore, MPINS-GREEDY considerably outperforms PIDS in large scale networks, sparse networks, and for high threshold θ.

Acknowledgment

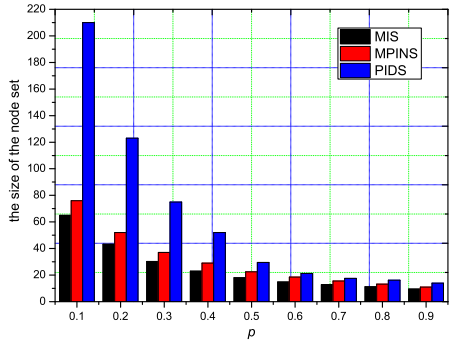
This research is funded in part by the Kennesaw State University College of Science and Mathematics Faculty Summer Research award program and the Interdisciplinary Research Opportunities (IDROP) Program.

References

[1] D. Kempe, J. Kleinberg, and E. Tardos, *Maximizing the Spread of Influence through a Social Network*, KDD'03.

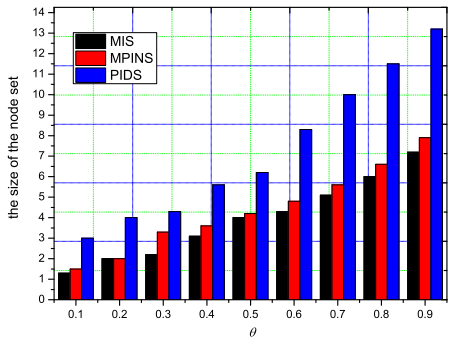


(a)

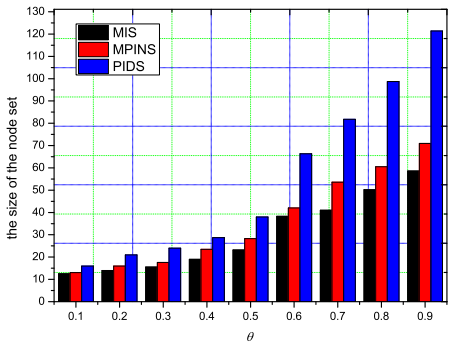


(b)

Figure 6: The size of the node set: (a) $n = 20$, and $\theta = 0.5$; (b) $n = 500$, and $\theta = 0.5$.



(a)



(b)

Figure 7: The size of the node set: (a) $n = 20$, and $p = 0.5$; (b) $n = 500$, and $p = 0.5$.

[2] S. Kazumi, K. Masahiro, M. Hiroshi, *Discoving influential Nodes for SIS Models in Social Networks*, Discov. Sci., 5808:302-316, 2009.

[3] Chi Wang, Wei Chen, Yajun Wang, *Scalable influence maximization for independent cascade model in large-scale social networks*, Data Mining and Knowledge Discovery, 25(3):545-576, 2012.

[4] Y. Li, W. Chen, Y. Wang, and Z. Zhang, *Influence diffusion dynamics and influence maximization in social networks with friend and foe relationships*, WSDM, 2013.

[5] F. Wang, H. Du, E. Camacho, K. Xu, W. Lee, Yan, Shi, and S. Shan, *On Positive Influence Dominating Sets in Social Networks*, TCS, 2011.

[6] K. Saito, R. Nakana, and M. Kimura, *Prediction of Information Diffusion Probabilities for Independent Cadcade Model*, KES'08.

[7] J. Tang, J. Sun, C. Wang, and Z. Yang, *Social Influence Analysis in Large-Scale Networks*, KDD'09.

[8] C. Wang, J. Tang, J. Sun, and J. Han, *Dynamic Social Influence Analysis through Time-Dependent Factor Graphs*, ASONAM'11.

[9] A. Goyal, F. Bonchi, L. Laskhmanan, *Learning Influence Probabilities in Social Networks*, WSDM'10.

[10] Z. Lu, Wei Zhang, W. Wu, B. Fu, and D. Du, *Approxiamtion and Inaaproximation for The Influence Maximization Problem in Social Networks under deterministic Linear Threshold Model*, ICDCSW, pp. 160-165, 2011.

[11] J. He, S. Ji, Y. Pan, and Y. Li, *Constructing Load-Balanced Data Aggregation Trees in Probabilistic Wireless Sensor Networks*, TPDS, 2013.

[12] S. Ji, J. He, A. S. Uluagac, R. Beyah, and Y. Li, *Cell-based Snapshot and Continuous Data Collection in Wireless Sensor Networks*, TOSN, 9(4), 2013.

[13] J. He, S. Ji, Y. pan, and Z. Cai, *Approximation Algorithms for Load-Balanced Virtual Backbone Construction in Wireless Sensor Networks*, Theoretical Computer Science (TCS), 2013.

[14] D.Z. Du, K. Ko, *Theory of Computational Complexity*, John Wiley, 2000.