

# Selecting Nodes and Buying Links to Maximize the Information Diffusion in a Network

Gianlorenzo D'Angelo<sup>1</sup>, Lorenzo Severini<sup>2</sup>, and Yllka Velaj<sup>3</sup>

- 1 Gran Sasso Science Institute (GSSI), L'Aquila, Italy  
gianlorenzo.dangelo@gssi.infn.it
- 2 ISI Foundation, Torino, Italy  
lorenzo.severini@isi.it
- 3 Gran Sasso Science Institute (GSSI), L'Aquila, Italy, and  
Department of Economic Studies, University of Chieti-Pescara, Pescara, Italy  
yllka.velaj@unich.it

---

## Abstract

The Independent Cascade Model (ICM) is a widely studied model that aims to capture the dynamics of the information diffusion in social networks and in general complex networks. In this model, we can distinguish between active nodes which spread the information and inactive ones. The process starts from a set of initially active nodes called seeds. Recursively, currently active nodes can activate their neighbours according to a probability distribution on the set of edges. After a certain number of these recursive cycles, a large number of nodes might become active. The process terminates when no further node gets activated.

Starting from the work of Domingos and Richardson [10, 26], several studies have been conducted with the aim of shaping a given diffusion process so as to maximize the number of activated nodes at the end of the process. One of the most studied problems has been formalized by Kempe et al. and consists in finding a set of initial seeds that maximizes the expected number of active nodes under a budget constraint [14]. In this paper we study a generalization of the problem of Kempe et al. in which we are allowed to spend part of the budget to create new edges incident to the seeds. That is, the budget can be spent to buy seeds or edges according to a cost function. The problem does not admit a PTAS, unless  $P = NP$ . We propose two approximation algorithms: the former one gives an approximation ratio that depends on the edge costs and increases when these costs are high; the latter algorithm gives a constant approximation guarantee which is greater than that of the first algorithm when the edge costs can be small.

**1998 ACM Subject Classification** F.2 Analysis of Algorithms and Problem Complexity

**Keywords and phrases** Approximation algorithms, information diffusion, complex networks, independent cascade model, network augmentation

**Digital Object Identifier** 10.4230/LIPIcs.MFCS.2017.75

## 1 Introduction

When a new idea or innovation arises in a network of individuals, it can either quickly propagate to a large part of the network and be adopted by many individuals or immediately expire. Understanding the dynamics that regulate these behaviours has been one of the main goals in the field of complex network analysis and has been studied under the name of *influence spreading* or *information diffusion* analysis problem [10, 14]. The motivating application span several fields: from marketing with the aim of evaluating the success of a new product or maximizing its adoption [3, 4, 11, 21, 26], to epidemiology in order to limit the diffusion of a virus or disease [22, 23], the study of adoption of innovations [5, 27, 30],



© Gianlorenzo D'Angelo, Lorenzo Severini, and Yllka Velaj;  
licensed under Creative Commons License CC-BY

42nd International Symposium on Mathematical Foundations of Computer Science (MFCS 2017).

Editors: Kim G. Larsen, Hans L. Bodlaender, and Jean-Francois Raskin; Article No. 75; pp. 75:1–75:14

Leibniz International Proceedings in Informatics



LIPICs Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

the analysis of social networks to find influential users and to study how information flows through the network [2], and the analysis of cascading failures in power networks [1].

Different models of information diffusion have been introduced in the literature. Two widely studied models are the *Linear Threshold Model* (LTM) [13, 15, 28] and the *Independent Cascade Model* (ICM) [11, 12, 14, 15]. In both models, we can distinguish between *active*, or *affected*, nodes which spread the information and inactive ones. At the beginning of the process a small percentage of nodes of the graph is set to active in order to let the information diffusion process start. These nodes are called *seeds*. Recursively, currently affected nodes can infect their neighbours with some probability. After a certain number of these cascading cycles, a large number of nodes might become affected in the network. In LTM the idea is that a node becomes active as more of its neighbours become active. Formally, each node  $u$  has a threshold  $t_u$  chosen uniformly at random in the interval  $[0, 1]$ . The threshold represents the weighted fraction of neighbours of  $u$  that must become active in order for  $u$  to become active. During the process, a node  $u$  becomes active if the total weight of its active neighbours is greater than  $t_u$ . In ICM, instead, an active node  $u$  tries to influence one of its inactive neighbours but the success of node  $u$  in activating the node  $v$  only depends on the propagation probability of the edge from  $u$  to  $v$  (each edge has its own value). Regardless of its success, the same node will never get another chance to activate the same inactive neighbour. The process terminates when no further node gets activated.

An interesting question, in the analysis of the information diffusion through a network, is how to shape a given diffusion process so as to maximize or minimize the number of activated nodes at the end of the process by taking intervention actions. Many intervention actions have been studied in the literature, the most important one has been proposed by Domingos and Richardson in the field of viral marketing and asks to find a small set of “influential” seeds in a network in order to activate a large part of the network [10, 26]. The problem has been formalized by Kempe et al. [15] as follows: if we are allowed to choose at most  $k$  seeds, which ones should be selected so as to maximize the number of active nodes resulting from the diffusion process [15]. This problem admits a  $(1 - \frac{1}{e})$ -approximation algorithm and this factor cannot be improved, unless  $P = NP$  [15]. Besides seeds selection, other intervention actions may be used to facilitate the diffusion processes, such as inserting or deleting edges and adding or deleting nodes in the network. Since in social networks and in other complex networks users can add edges incident to themselves, in this paper we consider the possibility to create a limited number of new edges incident to the initial seed nodes. In detail, we study the following generalization of the problem of Kempe et al.: we are given a cost for each possible edge that can be created in a network and a budget  $k$ , and we want to find a set of seed nodes  $A$  and a set of edges  $S$  incident to the nodes in  $A$  such that the expected number of active nodes at the end of the diffusion process is maximized and the overall cost of  $A$  and  $S$  does not exceed  $k$ , assuming that all the seeds have the same cost.

**Related work.** The problem of choosing initially active nodes to maximize the information diffusion in a network has been widely studied, we refer the interested reader to [15] and references therein for more detail, while the budgeted version of the problem was proposed in [24]. In the following we focus on the problem of modifying a graph in order to maximize or minimize the spread of information in a network under LTM and ICM models.

To the best of our knowledge, under LTM, the problems that have been studied are those outlined in what follows. Khalil et al. [16] consider two types of actions, adding edges to or deleting edges from the existing network to minimize the information diffusion and they show that this network structure modification problem has a supermodular objective and therefore

can be solved by algorithms with provable approximation guarantees. Zhang et al. [32] consider arbitrarily specified sets of nodes, and interventions that involve both edge and node removal from the sets. They develop algorithms with rigorous performance guarantees and good empirical performance. Kimura et al. [18] use a greedy approach to delete edges under the LTM but do not provide any rigorous approximation guarantees. Kuhlman et al. [20] propose heuristic algorithms for edge removal under a simpler deterministic variant of LTM which is not only hard, but also has no approximation guarantee. Papagelis [25] and Crescenzi et al. [6] study the problem of augmenting the graph in order to increase the connectivity or the centrality of a node, respectively and experimentally show that this increases the average number of eventual active nodes.

Under ICM, Wu et al. [31] consider intervention actions other than edge addition, edge deletion and seed selection, such as increasing the probability that a node infects its neighbours. They proved that optimizing the selection of these actions with a limited budget is  $NP$ -hard and is neither submodular nor supermodular. Sheldon et al. [29] study the problem of node addition to maximize the spread of information, and provide a counterexample showing that the objective function is not submodular. Kimura et al. [19] propose methods for efficiently finding good approximate solutions on the basis of a greedy strategy for the edge deletion problem under the ICM, but do not provide any approximation guarantees. D'Angelo et al. [7, 8] focus on the case in which the initial set of seeds is given and provide a constant approximation algorithm.

**Our results.** In this paper, we focus on the independent cascade model and investigate the problem of selecting a set of initial seeds and adding a small number of edges incident to the seeds, without exceeding a given budget  $k$ , in order to maximize the spread of information in terms of the expected number of nodes that eventually become active. The problem we analyse differs from above mentioned ones since we make the reasonable restriction that the edges to be added can only be incident to the seed nodes. To our knowledge, similar problems have never been studied for the independent cascade model. We refer to this problem as the *Budgeted Influence Maximization with Augmentation problem* (BIMA).

We observe that the BIMA problem is a generalization of the problem in [15] and therefore cannot be approximated within a factor greater than  $1 - \frac{1}{e}$ , unless  $P = NP$ .

We then focus on approximation algorithms. We first assume that the edge costs are all greater than a given constant  $c_{\min}$  and, in Section 3, we propose two approximation algorithms that guarantee approximation factors of  $1 - \frac{1}{e^{1+c_{\min}}}$  and  $(1 - \frac{1}{e})c_{\min}$ , respectively. Note that these factors increase with  $c_{\min}$  and the second algorithm achieves the optimal approximation of  $1 - \frac{1}{e}$  when all the edge costs are equal to 1. It is somewhat expected that when the all costs are high the approximation factor gets closer to the lower bound of  $1 - \frac{1}{e}$ . Indeed, when the cost of buying an edge approaches the cost of buying the node at the tail of this edge, we can buy the node instead of the edge with small increase in the cost. This allows us to exploit the  $1 - \frac{1}{e}$  approximation algorithm for the seed selection problem proposed in [15]. However, the challenge of our problem consists in finding a good approximation even when the cost function includes small values. In Section 4, we focus on the general case and propose an algorithm that guarantees a constant approximation ratio of about 0.0878. This algorithm outperforms the other two algorithms when the cost is allowed to be small. See Figure 1 for a summary of the approximation ratio of our algorithms.

Some of the proofs are omitted due to space constraints and can be found in the full version of the paper [9].

## 2 Preliminaries

A social network is represented by a weighted directed graph  $G = (V, E, p, c)$ , where  $V$  represents the set of nodes,  $E$  represents the set of relationships,  $p : V \times V \rightarrow [0, 1]$  is the propagation probability of an edge, that is the probability that the information is propagated from  $u$  to  $v$  if  $(u, v) \in E$ , and  $c : V \times V \rightarrow [0, 1]$  is the cost of adding an edge to  $E$ . In ICM, each node can be either *active* or *inactive*. If a node is active (or adopter of the innovation), then it is already influenced by the information under diffusion. If a node is inactive, then it is unaware of the information or not influenced. The process runs in discrete steps. At the beginning of the ICM process, few nodes are given the information, they are known as *seed nodes*. Upon receiving the information these nodes become active. In each discrete step, an active node tries to influence its inactive neighbours. The success of node  $u$  in activating the node  $v$  depends on the propagation probability of the edge  $(u, v)$ , independently of the history so far. Regardless of its success, the same node will never get another chance to activate the same inactive neighbour. The process terminates when no further nodes become activated from inactive state.

We define the influence of a set  $A \subseteq V$  in the graph  $G$ , denoted  $\sigma(A)$ , to be the expected number of active nodes in  $G$  at the end of the process, given that  $A$  is the set of seeds. Given a set  $S$  of edges not in  $E$ , we denote by  $G(S)$  the graph augmented by adding the edges in  $S$  to  $G$ , i.e.  $G(S) = (V, E \cup S)$ . We denote by  $\sigma(A, S)$  the influence of  $A$  in  $G(S)$ .

In this paper we look for a set of seeds  $A$  and a set of edges  $S$ , to be added to  $G$ , incident to these seeds that maximize  $\sigma(A, S)$ . W.l.o.g. we assume that each seed node can be selected with cost 1, while each edge  $e \in (V \times V) \setminus E$  can be selected with cost  $c_e \in [0, 1]$ . In detail, the BIMA problem is defined as follows: given a graph  $G = (V, E)$  and a budget  $k$ , find a set  $A$  of seeds and a set  $S$  of edges such that  $S \subseteq (V \times V) \setminus E$ ,  $c(A, S) \leq k$ , and  $\sigma(A, S)$  is maximum, where  $c(A, S) = |A| + \sum_{e \in S} c_e$ .

A *live-edge graph*  $X = (V, E_X)$  of  $G$  is a directed graph where the set of nodes is the same set  $V$  and the set of edges  $E_X$  is a subset of  $E$  given by an edge selection process in which each edge in  $E$  belongs to  $E_X$  or not according to its propagation probability. In detail, we can assume that for each edge  $e = (u, v)$  in the graph, we flip a coin of bias  $p_e$  and only the edges for which the coin indicated an activation belong to  $E_X$ . It is easy to show that the information diffusion process is equivalent to a reachability problem in live-edge graphs: given any seed set  $A$ , the distribution of active node sets after the diffusion process ends is the same as the distribution of node sets reachable from  $A$  in live-edge graphs. We denote by  $\chi$  the probability space in which each sample point specifies one possible set of outcomes for all the coin flips on the edges, that is the set of all possible live-edge graphs of  $G$ . For a set of edges  $S \subseteq (V \times V) \setminus E$ , the set of all possible live-edge graphs of  $G(S)$  is denoted by  $\chi(S)$ . Given two set of edges  $S, T$ , such that  $S \subseteq T$ , for each live-edge graph  $X$  in  $\chi(S)$  we denote by  $\chi(T, X)$  the set of live-edge graphs in  $\chi(T)$  that have  $X$  as a subgraphs and possibly contain other edges in  $T \setminus S$ . In other words, a live-edge graphs in  $\chi(T, X)$  has been generated with the same outcomes as  $X$  on the coin flips in the edges of  $E \cup S$  and it has other outcomes for edges in  $T \setminus S$ . The following holds:  $|\chi(T, X)| = 2^{|T \setminus S|}$ , for each  $Y \in \chi(T, X)$   $\mathbb{P}[Y] = \mathbb{P}[Y|X]\mathbb{P}[X]$ ,  $\mathbb{P}[X] = \sum_{Y \in \chi(T, X)} \mathbb{P}[Y]$ , and  $\sum_{Y \in \chi(T, X)} \mathbb{P}[Y|X] = 1$ . For a node  $a \in V$  and a live-edge graph  $X$  in  $\chi(S)$ , let  $R(a, X)$  be the set of all nodes that can be reached from  $a$  in graph  $X$ , that is for each node  $u \in R(a, X)$ , there exists a path from  $a$  to  $u$  consisting entirely of live edges with respect to the outcome of the coin flips that generates  $X$ . Let  $R(A, X) = \bigcup_{a \in A} R(a, X)$ , then  $\sigma(A, S)$  can be computed as  $\sigma(A, S) = \sum_{X \in \chi(S)} \mathbb{P}[X] \cdot |R(A, X)|$ .

Computing  $\sigma(A)$  is  $\#P$ -complete [4], however it has been proven by using the Chernoff bound that it can be approximated within an arbitrarily good factor by simulating the random process a polynomial number of times [15]. Therefore, in the rest of the paper we can assume that we can compute  $\sigma(A)$  (and  $\sigma(A, S)$ ) within an arbitrary bound. This reflects to an additional factor  $1 + \epsilon$ , for any  $\epsilon > 0$ , to all algorithms presented in this paper. For the sake of clarity, we omit this factor from the approximation factor of our algorithms.

Given a set of edges  $S$ , for each graph  $X \in \chi(S)$  and subset of edges  $T \subseteq S$ , we denote by  $X^T$  the graph obtained by removing edges in  $T$  from  $X$ . Given two feasible solutions  $(A_1, S_1)$  and  $(A_2, S_2)$ , such that  $A_2 \subseteq A_1$  and  $S_2 \subseteq S_1$ , we denote with  $\delta(A_1, S_1, A_2, S_2)$  the expected number of nodes affected by  $(A_1, S_1)$  and not affected by  $(A_2, S_2)$ , formally:  $\delta(A_1, S_1, A_2, S_2) = \sum_{X \in \chi(S_1)} \mathbb{P}[X] \cdot (|R(A_1, X)| - |R(A_2, X^T)|)$ , where  $T = S_1 \setminus S_2$ .

► **Proposition 1.** *For each  $A_2 \subseteq A_1 \subseteq V$  and  $S_2 \subseteq S_1 \subseteq V \times V$ , such that the edges in  $S_1$  and  $S_2$  are outgoing  $A_1$  and  $A_2$ , respectively, then  $\delta(A_1, S_1, A_2, S_2) = \sigma(A_1, S_1) - \sigma(A_2, S_2)$ .*

We observe that our problem is a generalization of the influence maximization problem in [15], indeed it is enough to set  $c_e = 1$  for each  $e \in (V \times V) \setminus E$ . It follows that BIMA cannot be approximated within a factor greater than  $1 - \frac{1}{e}$ , unless  $P = NP$ .

### 3 Lower-bounded edge costs

In this section we consider the case in which the edge costs are at least a given value  $c_{\min}$ , that is for each  $e \in V \times V, c_e \geq c_{\min}$ . It can be easily shown that in this case selecting a set  $A$  of  $k$  seed nodes that maximizes  $\sigma(A, \emptyset)$  guarantees an approximation factor of  $c_{\min}$ . Since this problem can be optimally approximated within  $1 - \frac{1}{e}$  [15], we can obtain an overall  $(1 - \frac{1}{e}) c_{\min}$  approximation. In what follows we give an approximation algorithm that improves over this bound for small values of  $c_{\min}$ . The following analysis serves also as a warm-up for the analysis of the algorithm proposed in the next section.

Our algorithm, whose pseudocode is reported in Algorithm 1, finds two candidate solutions: the first solution is obtained by a greedy algorithm at lines 2–25, the second solution is found at line 26 and is made of a single node  $a_M$  and a single edge  $(a_M, v_M)$  for which  $\sigma(\{a_M\}, \{(a_M, v_M)\})$  is maximized. Then, the algorithm outputs one of the candidate solutions that maximizes the expected number of affected nodes (line 27).

The greedy phase, at each iteration, selects a solution  $(A, S)$  that adds at most one node and one edge to the current solution  $(A', S')$  and that maximizes the ratio between  $\delta(A, S, A', S')$  and the marginal cost of  $(A, S)$ , that is the cost of the added node or edge. In particular, it considers three possible ways of obtaining  $(A, S)$  from  $(A', S')$ :

**line 3:** select a seed node  $a$  that maximizes  $r_1 = \delta(A' \cup \{a\}, S', A', S')$ ,  $(A, S) = (A' \cup \{a\}, S')$ ;

**line 4:** select an edge  $(a, v)$  incident to a seed  $a$  in  $A'$  that maximizes  $r_2 = \frac{\delta(A', S' \cup \{(a, v)\}, A', S')}{c_{(a, v)}}$ ,  $(A, S) = (A', S' \cup \{(a, v)\})$ ;

**line 5:** select a seed node  $a$  not in  $A'$  and edge  $(a, v)$  incident to  $a$  that maximize  $r_3 = \frac{\delta(A' \cup \{a\}, S' \cup \{(a, v)\}, A', S')}{1 + c_{(a, v)}}$ ,  $(A, S) = (A' \cup \{a\}, S' \cup \{(a, v)\})$ .

The greedy phase of the algorithm selects a solution  $(A, S)$  that maximizes the three above ratios. If  $(A, S)$  does not violate the budget, i.e. cost  $c(A, S)$  is at most  $k$ , then it is chosen as new solution.

We denote by  $(A^*, S^*)$  an optimal solution to the BIMA problem. Let us consider the iterations executed by the greedy algorithm in which an element is added to  $(A, S)$ . For  $i \geq 1$ , let us denote by  $j_i$  the index of these iterations,  $j_i < j_{i+1}$ , and let  $j_{l+1}$  be the index of the first iteration in which an element in  $(A^*, S^*)$  is considered (i.e. it maximizes the

---

**Algorithm 1:**

**Input** : A directed graph  $G = (V, E)$  and an integer  $k \in \mathbb{N}$   
**Output** : A set of nodes  $A$  and a of edges  $S \subseteq (A \times V) \setminus E$  such that  $c(A, S) \leq k$

```

1  $A := \emptyset; S := \emptyset; U := V; T := (V \times V) \setminus E;$ 
2 while  $T \neq \emptyset$  or  $U \neq \emptyset$  do
3    $r_1 = \max_{a \in U} \{\delta(A \cup \{a\}, S, A, S)\};$ 
4    $r_2 = \max_{(a,v) \in (A \times V) \cap T} \{\delta(A, S \cup \{(a,v)\}, A, S) / c_{(a,v)}\};$ 
5    $r_3 = \max_{a \in U, (a,v) \in (\{a\} \times V) \cap T} \{\delta(A \cup \{a\}, S \cup \{(a,v)\}, A, S) / (1 + c_{(a,v)})\};$ 
6   if  $\max\{r_1, r_2, r_3\} = r_1$  then
7      $\hat{a} = \arg \max_{a \in U} \{\delta(A \cup \{a\}, S, A, S)\};$ 
8     if  $k - 1 \geq 0$  then
9        $A := A \cup \{\hat{a}\};$ 
10       $k := k - 1;$ 
11       $U := U \setminus \{\hat{a}\};$ 
12   else
13     if  $\max\{r_1, r_2, r_3\} = r_2$  then
14        $(\hat{a}, \hat{v}) := \arg \max_{(a,v) \in A \times V \cap T} \{\delta(A, S \cup \{(a,v)\}, A, S) / c_{(a,v)}\};$ 
15       if  $k - c_{(\hat{a}, \hat{v})} \geq 0$  then
16          $S := S \cup \{(\hat{a}, \hat{v})\};$ 
17          $k := k - c_{(\hat{a}, \hat{v})};$ 
18          $T := T \setminus \{(\hat{a}, \hat{v})\};$ 
19       else
20          $(\hat{a}, (\hat{a}, \hat{v})) := \arg \max_{a \in U, (a,v) \in (\{a\} \times V) \cap T} \{\delta(A \cup \{a\}, S \cup \{(a,v)\}, A, S) / (1 + c_{(a,v)})\};$ 
21         if  $k - c_{(\hat{a}, \hat{v})} - 1 \geq 0$  then
22            $(A, S) := (A \cup \{\hat{a}\}, S \cup \{(\hat{a}, \hat{v})\});$ 
23            $U := U \setminus \{\hat{a}\};$ 
24            $k := k - 1 - c_{(\hat{a}, \hat{v})};$ 
25            $T := T \setminus \{(\hat{a}, \hat{v})\};$ 
26  $(a_M, (a_M, v_M)) := \arg \max_{a \in V, (a,v) \in (\{a\} \times V) \setminus E} \{\sigma(A \cup \{a\}, S \cup \{(a,v)\})\};$ 
27 return  $\arg \max\{\sigma(A, S), \sigma(\{a_M\}, \{(a_M, v_M)\})\};$ 

```

---

above ratios) but not added to  $(A, S)$  because it violates the budget constraint. We denote by  $(A_i, S_i)$  the solution at the end of iteration  $j_i$  and by  $\bar{c}_i$  the marginal cost of  $(A_i, S_i)$  as computed in the above three ratios,

$$\bar{c}_i = \begin{cases} 1 & \text{if } A_i \setminus A_{i-1} = \{a\} \text{ and } S_i = S_{i-1} & (\text{i.e. } \max\{r_1, r_2, r_3\} = r_1) \\ c_e & \text{if } A_i = A_{i-1} \text{ and } S_i \setminus S_{i-1} = \{(a,v)\} & (\text{i.e. } \max\{r_1, r_2, r_3\} = r_2) \\ 1 + c_e & \text{if } A \setminus A_{i-1} = \{a\} \text{ and } S_i \setminus S_{i-1} = \{(a,v)\} & (\text{i.e. } \max\{r_1, r_2, r_3\} = r_3). \end{cases}$$

The next two lemmas are the core of our analysis [17].

► **Lemma 2.** *After each iteration  $j_i$ ,  $i = 1, 2, \dots, l + 1$ ,  $\sigma(A_i, S_i) - \sigma(A_{i-1}, S_{i-1}) \geq \frac{\bar{c}_i - c_{\min}}{k - 1 + c_{\min}} (\sigma(A^*, S^*) - \sigma(A_{i-1}, S_{i-1}))$ .*

**Proof.** We denote by  $\delta_i$  the expected number of nodes affected by solution  $(A_i, S_i)$  and not affected by solution  $(A_{i-1}, S_{i-1})$ ,  $\delta_i = \delta(A_i, S_i, A_{i-1}, S_{i-1})$ . We first show that the value  $\sigma(A^*, S^*) - \sigma(A_{i-1}, S_{i-1})$  is at most the sum, for each element in  $(A^*, S^*)$  and not in  $(A_{i-1}, S_{i-1})$ , of the expected number of nodes affected by this element and not affected by solution  $(A_{i-1}, S_{i-1})$ , that is the following inequality holds:

$$\begin{aligned} \sigma(A^*, S^*) - \sigma(A_{i-1}, S_{i-1}) &\leq \sum_{a \in A_1^*} \delta(A_{i-1} \cup \{a\}, S_{i-1}, A_{i-1}, S_{i-1}) + & (1) \\ &\sum_{\substack{e=(a,v) \in S^* \setminus S_{i-1} \\ \text{s.t. } a \in A_{i-1}}} \delta(A_{i-1}, S_{i-1} \cup \{e\}, A_{i-1}, S_{i-1}) + \sum_{\substack{a \in A_2^* \\ e=(a,v) \in S^* \setminus S_{i-1}}} \delta(A_{i-1} \cup \{a\}, S_{i-1} \cup \{e\}, A_{i-1}, S_{i-1}), \end{aligned}$$

where we divided the set  $A^* \setminus A_{i-1}$  into two subsets:  $A_1^*$  is the subset of  $A^* \setminus A_{i-1}$  that contains the seeds  $a$  with no incident edges in  $S^*$  (i.e.  $\nexists(a, v) \in S^*$ ), and  $A_2^* = A^* \setminus (A_{i-1} \cup A_1^*)$ .

The difference  $\sigma(A^*, S^*) - \sigma(A_{i-1}, S_{i-1})$  is at most  $\sigma(A^* \cup A_{i-1}, S^* \cup S_{i-1}) - \sigma(A_{i-1}, S_{i-1})$  and therefore upper-bounded by:

$$\begin{aligned}
& \sum_{X \in \chi(S^* \cup S_{i-1})} \mathbb{P}[X] |R(A^* \cup A_{i-1}, X)| - \sum_{X \in \chi(S_{i-1})} \mathbb{P}[X] |R(A_{i-1}, X)| \\
&= \sum_{X \in \chi(S_{i-1})} \sum_{Y \in \chi(S^* \cup S_{i-1}, X)} \mathbb{P}[Y] |R(A^* \cup A_{i-1}, Y)| - \sum_{X \in \chi(S_{i-1})} \mathbb{P}[X] |R(A_{i-1}, X)| \\
&= \sum_{X \in \chi(S_{i-1})} \mathbb{P}[X] \sum_{Y \in \chi(S^* \cup S_{i-1}, X)} \mathbb{P}[Y|X] |R(A^* \cup A_{i-1}, Y)| - \sum_{X \in \chi(S_{i-1})} \mathbb{P}[X] |R(A_{i-1}, X)| \\
&= \sum_{X \in \chi(S_{i-1})} \mathbb{P}[X] \left[ \sum_{Y \in \chi(S^* \cup S_{i-1}, X)} \mathbb{P}[Y|X] |R(A^* \cup A_{i-1}, Y)| - |R(A_{i-1}, X)| \right] \\
&= \sum_{X \in \chi(S_{i-1})} \mathbb{P}[X] \left[ \sum_{Y \in \chi(S^* \cup S_{i-1}, X)} \mathbb{P}[Y|X] |R(A^* \cup A_{i-1}, Y)| - \sum_{Y \in \chi(S^* \cup S_{i-1}, X)} \mathbb{P}[Y|X] |R(A_{i-1}, X)| \right] \\
&= \sum_{X \in \chi(S_{i-1})} \mathbb{P}[X] \sum_{Y \in \chi(S^* \cup S_{i-1}, X)} \mathbb{P}[Y|X] (|R(A^* \cup A_{i-1}, Y)| - |R(A_{i-1}, X)|) \tag{2}
\end{aligned}$$

For each  $X \in \chi(S_{i-1})$  and  $Y \in \chi(S^* \cup S_{i-1}, X)$ , the difference  $|R(A^* \cup A_{i-1}, Y)| - |R(A_{i-1}, X)|$  between the nodes reachable from  $A^* \cup A_{i-1}$  in  $Y$  and those reachable from  $A_{i-1}$  in  $X$  can be bounded as follows:

$$\begin{aligned}
|R(A^* \cup A_{i-1}, Y)| - |R(A_{i-1}, X)| &\leq \sum_{a \in A_1^*} (|R(A_{i-1} \cup \{a\}, X)| - |R(A_{i-1}, X)|) + \tag{3} \\
&\sum_{\substack{e=(a,v) \in Y \setminus X \\ \text{s.t. } a \in A_{i-1}}} (|R(A_{i-1}, X \cup \{e\})| - |R(A_{i-1}, X)|) + \sum_{\substack{a \in A_2^* \\ e=(a,v) \in Y \setminus X}} (|R(A_{i-1} \cup \{a\}, X \cup \{e\})| - |R(A_{i-1}, X)|).
\end{aligned}$$

Combining (2) and the first term of (3), we obtain the first term of (1). To show the second and third term of (1), observe that for a function  $f : V \times V \rightarrow \mathbb{N}$  and for each  $X \in \chi(S_{i-1})$ ,

$$\sum_{Y \in \chi(S^* \cup S_{i-1}, X)} \mathbb{P}[Y|X] \sum_{e \in Y \setminus X} f(e) \leq \sum_{e \in S^* \setminus S_{i-1}} p_e \sum_{Y \in \chi(S^* \cup S_{i-1} \setminus \{e\}, X)} \mathbb{P}[Y|X \cup \{e\}] f(e) = \sum_{e \in S^* \setminus S_{i-1}} p_e f(e).$$

This shows inequality (1).

Since the greedy phase of the algorithm selects a solution that that maximizes the ratio between the marginal increment in the objective function and the cost, the following holds:

- For each  $a \in A_1^*$ ,  $\delta(A_{i-1} \cup \{a\}, S_{i-1}, A_{i-1}, S_{i-1}) \leq \frac{\delta_i}{\bar{c}_i}$ ;
- For each  $e = (a, v) \in S^* \setminus S_{i-1}$  such that  $a \in A_{i-1}$ ,  $\frac{\delta(A_{i-1}, S_{i-1} \cup \{e\}, A_{i-1}, S_{i-1})}{c_e} \leq \frac{\delta_i}{\bar{c}_i}$ ;
- For each  $a \in A_2^*$  and  $e = (a, v) \in S^* \setminus S_{i-1}$ ,  $\frac{\delta(A_{i-1} \cup \{a\}, S_{i-1} \cup \{e\}, A_{i-1}, S_{i-1})}{1+c_e} \leq \frac{\delta_i}{\bar{c}_i}$ .

Since the edge costs are at least  $c_{\min}$ , then the number of edges in  $S^* \setminus S_{i-1}$  incident to each  $a \in A_2^*$  is at most  $\lfloor \frac{k}{c_{\min}} \rfloor \leq \frac{k}{c_{\min}}$ . Therefore, the right hand side of (1) is at most:

$$\begin{aligned}
& \sum_{a \in A_1^*} \frac{\delta_i}{\bar{c}_i} + \sum_{\substack{e=(a,v) \in S^* \setminus S_{i-1} \\ \text{s.t. } a \in A_{i-1}}} \frac{\delta_i}{\bar{c}_i} c_e + \sum_{\substack{a \in A_2^* \\ e=(a,v) \in S^* \setminus S_{i-1}}} \frac{\delta_i}{\bar{c}_i} (1 + c_e) \\
&\leq \frac{\delta_i}{\bar{c}_i} \left( |A_1^*| + \sum_{\substack{e=(a,v) \in S^* \setminus S_{i-1} \\ \text{s.t. } a \in A_{i-1}}} c_e + \frac{k}{c_{\min}} + \sum_{\substack{a \in A_2^* \\ e=(a,v) \in S^* \setminus S_{i-1}}} c_e \right) \leq \left( 1 + \frac{1}{c_{\min}} \right) k \frac{\delta_i}{\bar{c}_i}.
\end{aligned}$$

Where the last inequality is due to  $|A_1^*| + \sum_{\substack{e=(a,v) \in S^* \setminus S_{i-1} \\ \text{s.t. } a \in A_{i-1}}} c_e + \sum_{\substack{a \in A_2^* \\ e=(a,v) \in S^* \setminus S_{i-1}}} c_e \leq k$ .

To conclude the proof,  $\delta_i = \sigma(A_i, S_i) - \sigma(A_{i-1}, S_{i-1})$  follows from Proposition 1. ◀

The next lemma can be proven by induction on iterations  $j_i$  and by using Lemma 2.

► **Lemma 3.** *After each iteration  $j_i$ ,  $i = 1, 2, \dots, l+1$ ,*

$$\sigma(A_i, S_i) \geq \left[ 1 - \prod_{\ell=1}^i \left( 1 - \frac{\bar{c}_\ell}{k} \frac{c_{\min}}{1 + c_{\min}} \right) \right] \sigma(A^*, S^*).$$

► **Theorem 4.** *Algorithm 1 achieves an approximation factor of  $\frac{1}{2} \left( 1 - \frac{1}{e^{1+c_{\min}}} \right) \sigma(A^*, S^*)$ .*

**Proof.** We observe that since  $(A_{l+1}, S_{l+1})$  violates the budget, then  $c(A_{l+1}, S_{l+1}) > k$ . Moreover, for a sequence of numbers  $a_1, a_2, \dots, a_n$  such that  $\sum_{\ell=1}^n a_\ell = B$ , the function  $\left[ 1 - \prod_{i=1}^n \left( 1 - \frac{a_i}{B \cdot \beta} \right) \right]$  achieves its minimum when  $a_i = \frac{B}{n}$  and that  $\left[ 1 - \prod_{i=1}^n \left( 1 - \frac{a_i}{B \cdot \beta} \right) \right] \geq 1 - \left( 1 - \frac{1}{n \cdot \beta} \right)^n \geq 1 - e^{-\frac{1}{\beta}}$ . Therefore, by applying Lemma 3 for  $i = l+1$  and observing that  $\sum_{\ell=1}^{l+1} \bar{c}_\ell = c(A_{l+1}, S_{l+1})$ , we obtain:

$$\begin{aligned} \sigma(A_{l+1}, S_{l+1}) &\geq \left[ 1 - \prod_{\ell=1}^{l+1} \left( 1 - \frac{\bar{c}_\ell}{k \left( \frac{1+c_{\min}}{c_{\min}} \right)} \right) \right] \sigma(A^*, S^*) \\ &\geq \left[ 1 - \prod_{\ell=1}^{l+1} \left( 1 - \frac{\bar{c}_\ell}{c(A_{l+1}, S_{l+1}) \left( \frac{1+c_{\min}}{c_{\min}} \right)} \right) \right] \sigma(A^*, S^*) \\ &\geq \left[ 1 - \left( 1 - \frac{1}{(l+1) \frac{1+c_{\min}}{c_{\min}}} \right)^{l+1} \right] \sigma(A^*, S^*) \geq \left( 1 - \frac{1}{e^{1+c_{\min}}} \right) \sigma(A^*, S^*). \end{aligned}$$

By Proposition 1, follows that:

$$\sigma(A_{l+1}, S_{l+1}) = \sigma(A_l, S_l) + \delta_{l+1} \geq \left( 1 - \frac{1}{e^{1+c_{\min}}} \right) \sigma(A^*, S^*). \quad (4)$$

Since  $\delta_{l+1} \leq \sigma(\{a_M\}, \{(a_M, v_M)\})$ , we get

$$\sigma(A_l, S_l) + \sigma(\{a_M\}, \{(a_M, v_M)\}) \geq \left( 1 - \frac{1}{e^{1+c_{\min}}} \right) \sigma(A^*, S^*).$$

Hence,  $\max\{\sigma(A_l, S_l), \sigma(\{a_M\}, \{(a_M, v_M)\})\} \geq \frac{1}{2} \left( 1 - \frac{1}{e^{1+c_{\min}}} \right) \sigma(A^*, S^*)$ . ◀

We now propose an algorithm which improves the performance guarantee of Algorithm 1. Let  $M$  a fixed integer, we consider all the solutions  $(A, S)$  with cardinality  $M$  (i.e.  $|A| + |S| = M$ ) and cost at most  $k$  ( $c(A, S) \leq k$ ), and we complete all these solutions by using the greedy algorithm. The pseudocode is reported in Algorithm 2.

► **Theorem 5.** *For  $M \geq 4$  Algorithm 2 achieves an approximation factor of  $1 - \frac{1}{e^{1+c_{\min}}}$ .*

**Algorithm 2:**


---

**Input** : A directed graph  $G = (V, E)$ , integer  $M \in \mathbb{N}$  and an integer  $k \in \mathbb{N}$   
**Output** : A set of nodes  $A$  and a set of edges  $S \subseteq (A \times V) \setminus E$  such that  $c(A, S) \leq k$

- 1  $(A_1, S_1) := \arg \max\{\sigma(A, S) : |A| + |S| < M, c(A, S) \leq k\}$ ;
- 2  $A_2 := \emptyset; S_2 := \emptyset; U := V; T := (V \times V) \setminus E$ ;
- 3 **foreach**  $A \subseteq U, S \subseteq (A \times V) \setminus E$  such that  $|A| + |S| = M$  and  $c(A, S) \leq k$  **do**
- 4      $U := U \setminus A; T := T \setminus S$ ;
- 5     Complete  $(A, S)$  by using Algorithm 1 with  $U$  and  $T$  as possible nodes and edges;
- 6     **if**  $\sigma(A, S) > \sigma(A_2, S_2)$  **then**
- 7          $A_2 := A$ ;
- 8          $S_2 := S$ ;
- 9 **return**  $\arg \max\{\sigma(A_1, S_1), \sigma(A_2, S_2)\}$ ;

---

**Proof.** We assume that  $|A^*| + |S^*| > M$  since otherwise Algorithm 2 finds an optimal solution. We sort the elements in  $(A^*, S^*)$  by selecting at each step the element, which can be either a seed or an edge, that maximizes the marginal increment in the number of influenced nodes. Let  $Z = (A_Z, S_Z)$  be the first  $M$  elements in this order. We now consider the iteration of Algorithm 2 in which element  $Z$  is considered. We define  $(A_{Z'}, S_{Z'})$  as the elements added by the algorithm to  $(A_Z, S_Z)$  and  $(A, S) = (A_Z \cup A_{Z'}, S_Z \cup S_{Z'})$ . By Proposition 1 follows that  $\sigma(A, S) = \sigma(A_Z, S_Z) + \delta(A_Z \cup A_{Z'}, S_Z \cup S_{Z'}, A_Z, S_Z)$ .

The completion of  $(A_Z, S_Z)$  to  $(A, S)$  is an application of the greedy algorithm and therefore, we can use the result from the previous theorems. Let us consider the iterations executed by the greedy algorithm during the completion of  $(A_Z, S_Z)$  to  $(A, S)$ . For  $i \geq 1$ , let us denote by  $j_i$  the index of these iterations,  $j_i < j_{i+1}$ , and let  $j_{l+1}$  be the index of the first iteration in which an element in  $(A^* \setminus A_Z, S^* \setminus S_Z)$  is considered but not added to  $(A_{Z'}, S_{Z'})$  because it violates the budget constraint. Applying inequality (4) to the instance of the problem obtained removing the nodes covered by solution  $Z$  we get:

$$\delta(A_Z \cup A_{Z'}, S_Z \cup S_{Z'}, A_Z, S_Z) + \delta_{l+1} \geq \left(1 - \frac{1}{e^{1+c_{\min}}}\right) \sigma(A^* \setminus A_Z, S^* \setminus S_Z).$$

Moreover, since we ordered the elements in  $(A^*, S^*)$  and in iteration  $j_{l+1}$  and most 2 elements are selected, then  $\delta_{l+1} \leq \frac{2\sigma(A_Z, S_Z)}{M}$  and

$$\begin{aligned} \sigma(A, S) &= \sigma(A_Z, S_Z) + \delta(A_Z \cup A_{Z'}, S_Z \cup S_{Z'}, A_Z, S_Z) \\ &\geq \sigma(A_Z, S_Z) + \left(1 - \frac{1}{e^{1+c_{\min}}}\right) \sigma(A^* \setminus A_Z, S^* \setminus S_Z) - \delta_{l+1} \\ &\geq \sigma(A_Z, S_Z) + \left(1 - \frac{1}{e^{1+c_{\min}}}\right) \sigma(A^* \setminus A_Z, S^* \setminus S_Z) - \frac{2\sigma(A_Z, S_Z)}{M} \\ &\geq \left(1 - \frac{2}{M}\right) \sigma(A_Z, S_Z) + \left(1 - \frac{1}{e^{1+c_{\min}}}\right) \sigma(A^* \setminus A_Z, S^* \setminus S_Z) \end{aligned}$$

But,  $\sigma(A_Z, S_Z) + \sigma(A^* \setminus A_Z, S^* \setminus S_Z) \geq \sigma(A^*, S^*)$ , and we get:

$$\sigma(A, S) \geq \left(1 - \frac{1}{e^{1+c_{\min}}}\right) \sigma(A^*, S^*) + \left(\frac{1}{e^{1+c_{\min}}} - \frac{2}{M}\right) \sigma(A_Z, S_Z) \geq \left(1 - \frac{1}{e^{1+c_{\min}}}\right) \sigma(A^*, S^*),$$

for  $M \geq 2e^{\frac{c_{\min}}{1+c_{\min}}}$ . Since  $2e^{\frac{c_{\min}}{1+c_{\min}}} < 4$  for  $c_{\min} \in [0, 1]$ , the theorem follows.  $\blacktriangleleft$

#### 4 General case

In this section we introduce an approximation algorithm for the BIMA problem that guarantees a constant approximation ratio, i.e. it does not depend on the edge costs.

The algorithm is similar to Algorithm 1, that is it chooses the best between two candidate solutions: one solution is made of a single node  $a_M$  and a single edge  $(a_M, v_M)$  that maximizes  $\sigma(\{a_M\}, \{(a_M, v_M)\})$ , the other solution is found by means of a greedy algorithm that at each step maximizes the ratio between the marginal increment in the objective function of a candidate solution and its marginal cost. The main difference consists in the way in which the greedy algorithm computes a candidate solution. In fact, the algorithm presented here might select more than one edge at one time. In particular, at each iteration the greedy algorithm computes four candidate solutions starting from the current solution  $(A', S')$ . The first two solutions correspond to cases  $r_1$  and  $r_2$  of Algorithm 1, and differ from  $(A', S')$  by a single seed node or a single edge incident to a seed in  $A'$ , respectively. To compute the two further candidate solutions, the algorithm divides the edges into two sets: those whose cost is at least  $b$  and those whose cost is smaller than  $b$ , for some given constant  $b$ , and proceeds as follows.

- Select a seed node  $a$  not in  $A'$  and an edge  $(a, v)$ , which cost is  $c_{(a,v)} \geq b$ , incident to  $a$  that maximize  $\frac{\delta(A' \cup \{a\}, S' \cup \{(a,v)\}, A', S')}{1 + c_{(a,v)}}$ ;
- Select a seed node  $a$  not in  $A'$  and a set  $S$  of edges incident to  $a$ , whose overall cost is smaller than  $b$  (i.e.  $\sum_{(a,v) \in S} c_{(a,v)} < b$ ), that maximize  $\frac{\delta(A' \cup \{a\}, S' \cup S, A', S')}{1 + \sum_{(a,v) \in S} c_{(a,v)}}$ .

The greedy phase of the algorithm selects the candidate solution that gives the maximum ratio among the above four possibilities.

To compute the fourth candidate solution we need to solve an optimization problem, which we call RBIMA (where R stands for ratio). In the following, we assume that we can exploit an  $\alpha$ -approximation algorithm for this sub-problem. At the end of this section we will introduce a suitable algorithm for RBIMA.

The analysis of the algorithm is similar to that in the previous section. In particular, the next lemma is the core of the analysis and corresponds to Lemma 2. We denote by  $j_i$  an iteration of the greedy algorithm in which an element is added to the solution, by  $j_{i+1}$  the first iteration in which an element of the optimum is not added to the solution and by  $\bar{c}_i$  the marginal cost of the solution  $(A_i, S_i)$  computed at the iteration  $j_i$ .

► **Lemma 6.** *After each iteration  $j_i$ ,  $i = 1, 2, \dots, l + 1$ ,*

$$\sigma(A_i, S_i) - \sigma(A_{i-1}, S_{i-1}) \geq \frac{\bar{c}_i}{k} \frac{b\alpha}{b\alpha + b + \alpha + 2} (\sigma(A^*, S^*) - \sigma(A_{i-1}, S_{i-1})).$$

**Proof.** We denote by  $\delta_i$  the expected number of nodes affected by  $(A_i, S_i)$  and not affected by  $(A_{i-1}, S_{i-1})$ ,  $\delta_i = \delta(A_i, S_i, A_{i-1}, S_{i-1})$ . We divide the set  $A^* \setminus A_{i-1}$  into two subsets:  $A_1^*$  is the subset that contains the seeds  $a$  with no incident edges in  $S^*$ , and  $A_2^* = A^* \setminus (A_{i-1} \cup A_1^*)$ . For each  $a \in A_2^*$  let us consider the subset of  $S^*$  containing edges incident to  $a$  such that  $c_e < b$ . We partition this set into sets of edges whose overall cost is smaller than  $b$  in such a way that the number of sets in the partition is minimized. We denote this a partition by  $\mathcal{S}_a$  and observe that the overall number of sets in  $\mathcal{S}_a$ , for all  $a \in A_2^*$ , is at most  $2 \frac{k}{b}$  as it is equivalent to the minimum number of bins of size  $b$  needed to pack a set of items of overall size  $k$ . By using similar arguments as in Lemma 2, we can show that

$$\begin{aligned}
\sigma(A^*, S^*) - \sigma(A_{i-1}, S_{i-1}) &\leq \sum_{a \in A_1^*} \delta(A_{i-1} \cup \{a\}, S_{i-1}, A_{i-1}, S_{i-1}) \\
&+ \sum_{\substack{e=(a,v) \in S^* \setminus S_{i-1} \\ \text{s.t. } a \in A_{i-1}}} \delta(A_{i-1}, S_{i-1} \cup \{e\}, A_{i-1}, S_{i-1}) + \\
&\sum_{\substack{a \in A_2^* \\ e=(a,v) \in S^* \setminus S_{i-1} \\ c_e \geq b}} \delta(A_{i-1} \cup \{a\}, S_{i-1} \cup \{e\}, A_{i-1}, S_{i-1}) + \sum_{\substack{a \in A_2^* \\ S \in \mathcal{S}_a}} \delta(A_{i-1} \cup \{a\}, S_{i-1} \cup S, A_{i-1}, S_{i-1}).
\end{aligned} \tag{5}$$

Indeed,  $\sigma(A^*, S^*) - \sigma(A_{i-1}, S_{i-1}) \leq \sum_{X \in \chi(S_{i-1})} \mathbb{P}[X] \sum_{Y \in \chi(S^* \cup S_{i-1}, X)} \mathbb{P}[Y|X] (|R(A^* \cup A_{i-1}, Y)| - |R(A_{i-1}, X)|)$  and for each  $X \in \chi(S_{i-1})$  and  $Y \in \chi(S^* \cup S_{i-1}, X)$ ,

$$\begin{aligned}
|R(A^* \cup A_{i-1}, Y)| - |R(A_{i-1}, X)| &\leq \sum_{a \in A_1^*} (|R(A_{i-1} \cup \{a\}, X)| - |R(A_{i-1}, X)|) \\
&+ \sum_{\substack{e=(a,v) \in Y \setminus X \\ \text{s.t. } a \in A_{i-1}}} (|R(A_{i-1}, X \cup \{e\})| - |R(A_{i-1}, X)|) \\
&+ \sum_{\substack{a \in A_2^* \\ e=(a,v) \in Y \setminus X \\ c_e \geq b}} (|R(A_{i-1} \cup \{a\}, X \cup \{e\})| - |R(A_{i-1}, X)|) \\
&+ \sum_{\substack{a \in A_2^* \\ S \in \mathcal{S}_a}} (|R(A_{i-1} \cup \{a\}, X \cup (S \cap Y))| - |R(A_{i-1}, X)|).
\end{aligned}$$

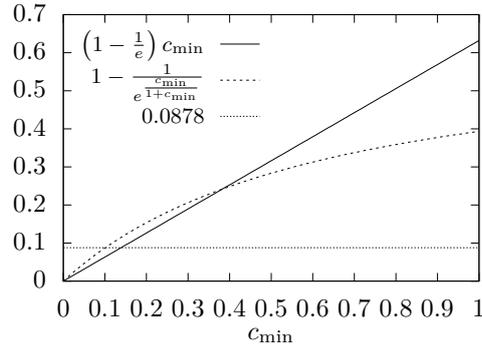
The following holds since the greedy phase of the algorithm selects a solution that maximizes the ratio between the marginal increment in the objective function and the cost:

- For each  $a \in A_1^*$ ,  $\delta(A_{i-1} \cup \{a\}, S_{i-1}, A_{i-1}, S_{i-1}) \leq \frac{\delta_i}{\bar{c}_i}$ ;
- For each  $e = (a, v) \in S^* \setminus S_{i-1}$  such that  $a \in A_{i-1}$ ,  $\frac{\delta(A_{i-1}, S_{i-1} \cup \{e\}, A_{i-1}, S_{i-1})}{c_e} \leq \frac{\delta_i}{\bar{c}_i}$ ;
- For each  $a \in A_2^*$  and  $e = (a, v) \in S^* \setminus S_{i-1}$  s.t.  $c_e \geq b$ ,  $\frac{\delta(A_{i-1} \cup \{a\}, S_{i-1} \cup \{e\}, A_{i-1}, S_{i-1})}{1+c_e} \leq \frac{\delta_i}{\bar{c}_i}$ ;
- For each  $a \in A_2^*$  and  $S \in \mathcal{S}_a$ ,  $\frac{\delta(A_{i-1} \cup \{a\}, S_{i-1} \cup S, A_{i-1}, S_{i-1})}{1+\sum_{e \in S} c_e} \leq \frac{\delta_i}{\bar{c}_i} \cdot \frac{1}{\alpha}$ ,

where the term  $\frac{1}{\alpha}$  in the last inequality is due to the use of an  $\alpha$ -approximation algorithm in the computation of the fourth candidate solution of the greedy phase. The number of edges incident to each  $a \in A_2^*$  with cost at least  $b$  is at most  $\frac{k}{b}$ . Therefore, the right hand side of (5) is at most:

$$\begin{aligned}
&\sum_{a \in A_1^*} \frac{\delta_i}{\bar{c}_i} + \sum_{\substack{e=(a,v) \in S^* \setminus S_{i-1} \\ \text{s.t. } a \in A_{i-1}}} \frac{\delta_i}{\bar{c}_i} c_e + \sum_{\substack{a \in A_2^* \\ e=(a,v) \in S^* \setminus S_{i-1} \\ c_e \geq b}} \frac{\delta_i}{\bar{c}_i} (1+c_e) + \sum_{\substack{a \in A_2^* \\ S \in \mathcal{S}_a}} \frac{1}{\alpha} \frac{\delta_i}{\bar{c}_i} \left(1 + \sum_{e \in S} c_e\right) \\
&= \frac{\delta_i}{\bar{c}_i} \left( |A_1^*| + \sum_{\substack{e=(a,v) \in S^* \setminus S_{i-1} \\ \text{s.t. } a \in A_{i-1}}} c_e + \frac{k}{b} + \sum_{\substack{a \in A_2^* \\ e=(a,v) \in S^* \setminus S_{i-1} \\ c_e \geq b}} c_e + \frac{1}{\alpha} \frac{2k}{b} + \frac{1}{\alpha} \sum_{\substack{a \in A_2^* \\ S \in \mathcal{S}_a}} \sum_{e \in S} c_e \right) \\
&\leq \left(1 + \frac{1}{b} + \frac{2}{b\alpha} + \frac{1}{\alpha}\right) k \frac{\delta_i}{\bar{c}_i} = \left(\frac{b\alpha + \alpha + 2 + b}{b\alpha}\right) k \frac{\delta_i}{\bar{c}_i}.
\end{aligned}$$

To conclude, we observe that by Proposition 1 follows that  $\delta_i = \sigma(A_i, S_i) - \sigma(A_{i-1}, S_{i-1})$ . ◀



■ **Figure 1** Approximation ratio of the three algorithm presented as a function of  $c_{\min}$ . Intersection points are at  $c_{\min} \approx 0.1011$  and  $c_{\min} \approx 0.3821$ .

The statements of Lemmas 2 and 6 differ only for the coefficients  $\frac{c_{\min}}{1+c_{\min}}$  and  $\frac{b\alpha}{b\alpha+b+\alpha+2}$ . Then, equipped with Lemma 6, we prove the next theorem, which corresponds to Theorem 4.

► **Theorem 7.** *The greedy algorithm achieves an approximation factor of  $1 - \frac{1}{e} \frac{b\alpha}{b\alpha+b+\alpha+2}$ .*

It remains to give an  $\alpha$ -approximation algorithm for the RBIMA problem which consists in selecting a seed node  $a$  not in the current solution  $(A', S')$  and a set  $S$  of edges incident to  $a$ , which overall cost is smaller than  $b$  (i.e.  $\sum_{(a,v) \in S} c_{(a,v)} < b$ ), that maximize  $\frac{\delta(A' \cup \{a\}, S' \cup S, A', S')}{1 + \sum_{(a,v) \in S} c_{(a,v)}}$ . To this aim we define the COSTIMA [8] problem as follows. Given a directed graph  $G = (V, E, p, c)$ , an integer  $k \in \mathbb{N}$  and a seed set  $A$ , find a set of edges  $S \subseteq A \times V \setminus E$  such that  $c(S) \leq k$  and  $\sigma(A, S)$  is maximized.

Let us consider the instances of the COSTIMA problem where  $A = \{a\}$  and let  $m^*$  be the maximum, over all  $a \in V \setminus A'$ , among the optima of these instances. It is easy to show that  $m^*$  is at least  $b + 1$  times the optimum of RCOSTIMA. It has been shown that the COSTIMA problem can be approximated within a factor of  $1 - \frac{1}{e}$  by using a greedy algorithm [8]. Therefore we obtain an overall approximation of  $\alpha = (1 - \frac{1}{e}) \frac{1}{b+1}$ . The next corollary follows by applying Theorem 7 with  $\alpha = (1 - \frac{1}{e}) \frac{1}{b+1}$  and optimizing over  $b$ .

► **Corollary 8.** *There exists an algorithm that achieves an approximation factor  $> 0.0878$  for the BIMA problem.*

## 5 Conclusions

In Figure 1 we summarize our approximation ratios as a function of  $c_{\min}$ . As expected, when all the edge costs are high, it is not worth to buy them. Indeed the algorithm that selects only seeds outperforms the other algorithms and it reaches the optimal approximation of  $1 - \frac{1}{e}$  when all the costs are equal to 1. However, the challenge of our problem consists in finding a good approximation even when the cost function includes small values. In the general case and when the minimum edge cost can be very small ( $c_{\min} < 0.1011$ ) the best algorithm is the constant factor algorithm given in Section 4, while when the minimum cost is in  $(0.1011, 0.3821)$  the best algorithm is the one presented in Section 3.

The main open problem is the gap between the lower bound on approximation of  $1 - \frac{1}{e} \approx 0.6321$  and the constant approximation of 0.0878. To close this gap, we aim at improving the algorithm in Section 4 by devising a better approximation algorithm for

RCostIMA problem, which directly implies a better approximation for the BIMA problem. Other research directions that deserve further investigation include the study of the BIMA problem on different information diffusion models such as LTM or the Triggering Model [15]. Moreover, we plan to design efficient heuristics in order to assess the performance of our greedy algorithm from the experimental point of view.

---

## References

- 1 C. Asavathiratham, S. Roy, B. Lesieutre, and G. Verghese. The influence model. *IEEE Control Systems*, 21(6):52–64, 2001.
- 2 Eytan Bakshy, Jake M. Hofman, Winter A. Mason, and Duncan J. Watts. Everyone's an influencer: Quantifying influence on twitter. In *Proc. of the 4th ACM International Conference on Web Search and Data Mining*, WSDM11, pages 65–74. ACM, 2011.
- 3 Frank M Bass. A new product growth for model consumer durables. *Management science*, 15(5):215–227, 1969.
- 4 Wei Chen, Chi Wang, and Yajun Wang. Scalable influence maximization for prevalent viral marketing in large-scale social networks. In *Proc. of the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD10, 2010.
- 5 James S. Coleman, Elihu Katz, and Herbert Menzel. *Medical innovation: A diffusion study*. The Bobbs-Merrill Company, 1966.
- 6 Pierluigi Crescenzi, Gianlorenzo D'Angelo, Lorenzo Severini, and Yllka Velaj. Greedily improving our own closeness centrality in a network. *ACM Trans. Knowl. Discov. Data*, 11(1):9:1–9:32, 2016.
- 7 Gianlorenzo D'Angelo, Lorenzo Severini, and Yllka Velaj. Influence maximization in the independent cascade model. In *Proceedings of the 17th Italian Conference on Theoretical Computer Science, Lecce, Italy, September 7-9, 2016.*, pages 269–274, 2016.
- 8 Gianlorenzo D'Angelo, Lorenzo Severini, and Yllka Velaj. Recommending links through influence maximization. *arXiv preprint*, 2017. URL: <http://arxiv.org/abs/1706.04368>.
- 9 Gianlorenzo D'Angelo, Lorenzo Severini, and Yllka Velaj. Selecting nodes and buying links to maximize the information diffusion in a network. *arXiv preprint*, 2017. URL: <https://arxiv.org/abs/1706.06466>.
- 10 Pedro Domingos and Matt Richardson. Mining the network value of customers. In *Proceedings of the 7th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD01, pages 57–66. ACM, 2001.
- 11 Jacob Goldenberg, Barak Libai, and Eitan Muller. Talk of the network: A complex systems look at the underlying process of word-of-mouth. *Marketing letters*, 12(3):211–223, 2001.
- 12 Jacob Goldenberg, Barak Libai, and Eitan Muller. Using complex systems analysis to advance marketing theory development: Modeling heterogeneity effects on new product growth through stochastic cellular automata. *Academy of Marketing Science Review*, 2001(9):1, 2001.
- 13 Mark Granovetter. Threshold models of collective behavior. *American journal of sociology*, 83(6):1420–1443, 1978.
- 14 David Kempe, Jon Kleinberg, and Éva Tardos. Maximizing the spread of influence through a social network. In *Proceedings of the 9th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD03, pages 137–146. ACM, 2003.
- 15 David Kempe, Jon M. Kleinberg, and Éva Tardos. Maximizing the spread of influence through a social network. *Theory of Computing*, 11:105–147, 2015.
- 16 Elias Boutros Khalil, Bistra Dilkina, and Le Song. Scalable diffusion-aware optimization of network topology. In *Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD14, pages 1226–1235. ACM, 2014.

- 17 Samir Khuller, Anna Moss, and Joseph Naor. The budgeted maximum coverage problem. *Inf. Process. Lett.*, 70(1):39–45, 1999.
- 18 Masahiro Kimura, Kazumi Saito, and Hiroshi Motoda. Solving the contamination minimization problem on networks for the linear threshold model. In *Proc. of the 10th Pacific Rim International Conference on Artificial Intelligence*, PRICA08, pages 977–984. Springer Berlin Heidelberg, 2008.
- 19 Masahiro Kimura, Kazumi Saito, and Hiroshi Motoda. Blocking links to minimize contamination spread in a social network. *ACM Trans. Knowl. Discov. Data*, 3(2):9:1–9:23, 2009.
- 20 Chris J Kuhlman, Gaurav Tuli, Samarth Swarup, Madhav V Marathe, and SS Ravi. Blocking simple and complex contagion by edge removal. In *IEEE International Conference on Data Mining*, ICDM13. IEEE, 2013.
- 21 Vijay Mahajan, Eitan Müller, and Frank M Bass. New product diffusion models in marketing: A review and directions for research. *Journal of Marketing*, 54:1–26, 1990.
- 22 Lauren Meyers. Contact network epidemiology: Bond percolation applied to infectious disease prediction and control. *Bulletin of the American Mathematical Society*, 44(1):63–86, 2007.
- 23 M. E. J. Newman. Spread of epidemic disease on networks. *Phys. Rev. E*, 66:016128, Jul 2002.
- 24 H. Nguyen and R. Zheng. On budgeted influence maximization in social networks. *IEEE Journal on Selected Areas in Communications*, 31(6):1084–1094, June 2013. doi:10.1109/JSAC.2013.130610.
- 25 Manos Papagelis. Refining social graph connectivity via shortcut edge addition. *ACM Trans. Knowl. Discov. Data*, 10(2):12, 2015.
- 26 Matthew Richardson and Pedro Domingos. Mining knowledge-sharing sites for viral marketing. In *Proceedings of the 8th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD02, pages 61–70. ACM, 2002.
- 27 E. Rogers. *Diffusion of innovations*. Free Press, 1995.
- 28 Thomas C Schelling. *Micromotives and macrobehavior*. Norton & Company, 2006.
- 29 Daniel Sheldon, Bistra N. Dilkina, Adam N. Elmachtoub, Ryan Finseth, Ashish Sabharwal, Jon Conrad, Carla P. Gomes, David B. Shmoys, William Allen, Ole Amundsen, and William Vaughan. Maximizing the spread of cascades using network design. *CoRR*, abs/1203.3514, 2012. URL: <http://arxiv.org/abs/1203.3514>.
- 30 T. Valente. *Network models of the diffusion of innovations*. Hampton Press, 1995.
- 31 Xiaojian Wu, Daniel Sheldon, and Shlomo Zilberstein. Efficient algorithms to optimize diffusion processes under the independent cascade model. In *NIPS Workshop on Networks in the Social and Information Sciences*, Montreal, Quebec, Canada, 2015.
- 32 Y. Zhang, A. Adiga, A. Vullikanti, and B. A. Prakash. Controlling propagation at group scale on networks. In *IEEE International Conference on Data Mining*, ICDM15, pages 619–628, 2015.