How to choose friends strategically [⋆]

Lata Narayanan and Kangkang Wu

Department of Computer Science and Software Engineering
Concordia University
Montréal, Canada

Abstract. Alice wants to join a new social network, and influence its members to adopt a new product or idea. Each person v in the network has a certain threshold $t(v)$ for *activation*, i.e adoption of the product or idea. If v has at least $t(v)$ activated neighbors, then v will also become activated. If Alice wants to make k new friends in the network, and thereby activate the most number of people, how should she choose these friends? We study the problem of choosing the k people in the network to befriend, who will in turn activate the maximum number of people. This *Maximum Influence with Links* Problem has applications in viral marketing and the study of epidemics. We show that the solution can be quite different from the related and widely studied influence maximization problem where the objective is to choose a seed or target set with maximum influence. We prove that the Maximum Influence with Links problem is NP-complete even for bipartite graphs in which all nodes have threshold 1 or 2. In contrast, we give polynomial time algorithms that find optimal solutions for the problem for trees, paths, cycles, and cliques.

1 Introduction

The strategy of *viral marketing* for promoting new products or ideas is based on the observation that once a certain fraction of a social network adopts a product, we can expect a cascade of further adoptions [3, 16, 27]. Domingos and Richardson [12, 33] were the first to raise the following important algorithmic problem in the context of social network analysis: If a company can turn a subset of customers in a given network into early adopters, and the goal is to trigger a large cascade of further adoptions, which set of customers should they target?

The social network can be modelled by a node-weighted graph $G = (V, E, t)$ with $V(G)$ representing individuals in the social network, $E(G)$ denoting the social connections, and t an integer-valued *threshold function*. Starting with an initial *seed set or target set*, that is, a subset $S \subseteq V$ of nodes in the graph, that are *activated* by some external incentive, influence propagates deterministically in discrete time steps, and is said to *activate* nodes. For any unactivated node v, if the number of its activated neighbors at time step $i-1$ is at least $t(v)$, then node v will be activated in step i. A node once activated stays activated. Clearly the process terminates after at most $|V|-1$ steps. We call the set of nodes that are activated when the process terminates as the *activated set*. The

[⋆] Research supported by NSERC, Canada

problem proposed by Domingo and Richardson [12, 33] can now be formulated as follows: Given a social network $G = (V, E, t)$, and an integer k, find a subset $S \subseteq V$ of size k so that the resulting activated set is as large as possible. In the context of viral marketing, the parameter k corresponds to the budget, and S is a target set that maximizes the size of the activated set. This influence maximization problem has been widely studied [18, 22, 28, 19, 5, 17, 2, 14].

Recent work [10, 20, 25] points to some flaws in the above model for viral marketing. The authors of [10, 20] observe that a limitation of the model is that there is no possibility of giving *partial* external incentives; indeed the initial seed set is activated *wholly* by external incentives, and the remaining nodes *only* by the internal network effect. The authors of [25] further critique the fact that the nodes in the seed set are assumed to be activated immediately by external incentives, *regardless of their own thresholds of activation*. They point out that this is unrealistic assumption, and suggest that it is likely that highly influential nodes have high thresholds, and cannot be activated by external incentives alone.

In this paper, we provide a new way to model a *viral marketing strategy with a fixed budget*, that addresses the flaws mentioned above. In particular, we model strategies in which each node in the target set is given some partial incentive, eg. a $10 coupon; for some people this may be enough for them to buy the product, for others, it reduces their resistance to buying it. We represent the initiator of the viral marketing strategy by a node *external to the social network*. We can now restate the influence maximization problem as follows: Suppose Alice, the external initiator, wants to join a new social network, and can make k friends, how should she choose these friends if her goal is to influence as many people as possible? In other words, to which k people already in the social network should Alice create links, so that she can activate the maximum number of people? If Alice creates a link to a node v, the threshold of v is only effectively reduced by one, and so v in turn is activated only if its threshold is one. We call our problem the *Maximum Influence with Links* problem (Max-Inf-Links).

Notice that the links added from the external node Alice correspond to the external incentive given to the endpoints of these links. The nodes that are the endpoints of these new links may not be immediately completely activated, but their thresholds are effectively reduced; this corresponds to their receiving partial incentives. Individuals with high thresholds cannot be activated only by external incentives, which better models reality. The Max-Inf-Links problem also has applications in epidemiology or the spread of diseases: if an infected person arrives from *outside* a community, the Max-Inf-Links problem corresponds to identifying the set of k people such that if the infected external person has contact with this set, the highest number of people in the community could potentially be infected. The problem can obviously be generalized to a set of external influencers wishing to infect or gain control of a network.

Observe that the solution to the Max-Inf-Links problem can be quite different from the solution to choosing an optimal seed set for a given network. For example, consider a clique K_n in which node 1 has threshold $n-1$ and the remaining nodes have threshold 1. If the budget k is 1, the optimal target set is clearly the node x, as choosing it activates the entire network. However, if we choose to *link* to the node x, it does not get activated, since its threshold is $n-1$, and therefore no node in the network gets activated. The optimal solution to the Max-Inf-Links problem is in fact to choose any of the other nodes in K_n; this would activate the entire network. This would however be a sub-optimal seed set.

1.1 Our Results

It was shown recently that it is NP-hard to find the minimum number of links required for an external influencer to activate the entire network [25]. Since this is a special case of our Max-Inf-Links problem, it follows that our problem is NP-hard in general. We prove here that the Max-Inf-Links problem is NP-hard, even if all nodes have threshold 1 or 2, even in bipartite graphs, which have many applications in social networks. Note that the NP-hardness proof in [25] does not apply to bipartite graphs. In light of the hardness result, we study the complexity of the problem for social networks that can be represented as trees, cycles, and cliques. We show that optimal solutions can be found for the Max-Inf-Links problem with k links in time $\Theta(kn)$ for paths, $\Theta(kn^2)$ for cycles, $\Theta(k^2n^2)$ for trees, and $\Theta(n)$ for cliques. Note that k is always upper bounded by n. Our algorithms for paths, trees, and cycles rely on non-trivial dynamic programming formulations. Note that the algorithm for trees can be used for paths, but we are able to get a much faster implementation for paths by making careful use of the solution properties.

1.2 Related work

The problem of identifying the most influential nodes in a social network has received a tremendous amount of attention [18, 22, 28, 19, 5, 17, 2, 14]. The algorithmic question of choosing a seed set of size k that activates the most number of nodes in the context of viral marketing was first posed by Domingos and Richardson [12]. Many natural heuristics were proposed for the problem, such as choosing the nodes of highest degree, or those central in terms of distance [23, 24]. Kempe et al [23] started the study of this problem as a discrete optimization problem, and studied it in both the probabilistic independent cascade model and the threshold model of the influence diffusion process. They showed the NP-hardness of the problem in both models, and showed that a natural greedy strategy has a $(1-1/e-\varepsilon)$-approximation guarantee in both models; these results were generalized to a more general cascade model in [24]. Mossel and Roch [30] further generalized the results of [23, 24] by positively resolving their conjecture that whenever the local threshold functions are local and submodular, the resulting influence function is also a submodular function. Borgs *et al* [2] recently obtained a significant improvement by giving an algorithm that obtains an approximation factor of $1-1/e-\varepsilon$ for any $\varepsilon > 0$, in time $O((m+n)k\log(n)/\varepsilon^2)$. A large body of work studies algorithms that have performance guarantees and at the same time scale in practice to real-life and very large-scale social networks [36, 35, 31]. Influence diffusion under time window or deadline constraints has also been studied [15, **?**,29, 26].

In the Target Set Selection problem ([4, 1, 32], the size of the target set or budget is not specified in advance, but the goal is to activate the entire network or a fixed fraction of nodes. Partial incentives were studied in the context of the target set selection problem in [20], [21], [10], [25], and [8]. The closest formulation to our paper is the

Minimum Links problem introduced in [25], where it is required to find the minimum number of links an external influencer needs to form to nodes in the network so that the entire network is eventually activated. It was shown in [25] that the Minimum Links problem is NP-hard, and in fact, it is even hard to approximate with ratio ε for some constant ε. The authors gave linear time greedy algorithms for trees, cycles, and cliques.

Demaine *et al* [10] were the first to introduce the study of the maximization of influence with partial incentives and a fixed budget [10]. However, they consider thresholds chosen uniformly at random, while we study arbitrary thresholds. Additionally, they allow arbitrary fractional influence to be applied externally on any node, while in our model, every node that receives a link has its threshold reduced by the same amount. Eftekhar et al. studied a model where nodes become seed nodes with a fixed probability [13]. Recently, this idea was extended by considering the idea of offering discounts to nodes, which would cause some nodes to be be activated with a probability proportional to the amount of the discount [34, 37].

2 Notation and preliminaries.

Given a social network represented by an undirected graph $G = (V, E, t)$, we introduce a set of external nodes U that are assumed to be already activated. We assume that all edges have unit weight; this is generally called the *uniform weight assumption*, and has previously been considered in many papers [4, 6, 7, 15, 25]. A *link set* for (G, U) is a set S of links between nodes in U and nodes in V, i.e $S \subseteq \{(u, v) \mid u \in U; v \in V\}$. For a link set S, we define $E(S) = \{v \in V \mid \exists (u, v) \in S\}$, that is, $E(S)$ is the set of V-endpoints of links in S. For a node $v \in V(G)$, define $r(v)$ to be the number of links in S for which v is an endpoint. Since the set of external nodes U is already activated, observe that adding the link set S to G is equivalent to reducing the threshold of the node v by $r(v)$. In the viral marketing scenario, the link set S represents giving v a partial incentive of $r(v)$.

Given a link set S for a graph G, we define $I(G, S)$ to be the set of nodes in G that are eventually activated as a result of adding the link set S, that is, by reducing the threshold of each node $v \in E(S)$ by $min\{r(v), t(v)\}$, and then running the influence diffusion process. See Figure 1 for an illustration. Observe that this is the same as the set of nodes activated by using U as the target set in the graph G', the graph obtained from G by adding the set U to the vertex set and the set S to the set of edges.

Definition 1. Maximum Influence with Links problem
(Max-Inf-Links): *Given a social network $G = (V, E, t)$, and a set of external nodes U, and an integer k, find a link set S of size k that maximizes $I(G, S)$ among all k-sized link sets. We denote by $MI(G, k)$ the maximum number of people that can be influenced in G by an optimal link set of size k, that is, if S is an optimal solution to the* Max-Inf-Links*(G, k) problem, then $MI(G, k) = |I(G, S)|$.*

In our algorithms, we consider the case of a *single influencer*, that is, $U = \{\mu\}$. In this case, a link given to a vertex v reduces its threshold by 1. Since μ must be an endpoint of each edge in the link set S, each such edge can be uniquely specified by a vertex in V. We therefore generally omit mention of μ in the rest of the paper, and the link set S is referred to by its V-endpoints. For each such node $v \in E(S)$, we say we *give* v a link, or that v *receives* a link.

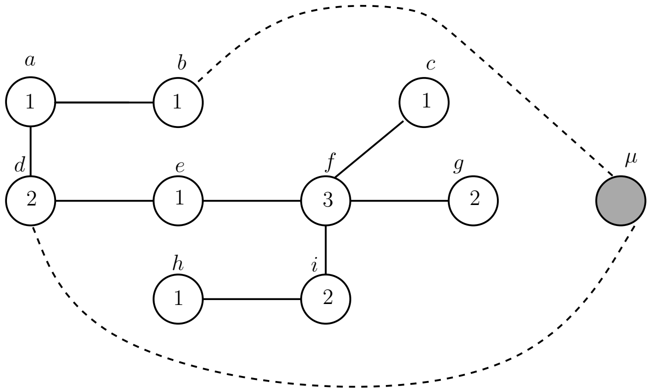

Fig. 1. Node μ is the external influencer and is assumed to be activated. Links in the link set are shown with dashed edges. The given link set is an optimal link set of size 2 that activates 4 nodes: a,b,d,e.

3 NP-hardness of Max-Inf-Links

In this section, we will prove that the decision version of the Max-Inf-Links problem is NP-hard even for bipartite graphs in which thresholds of all nodes are either one or two. It can be posed as follows: Given a social network $G = (V,E,t)$, a set of external influencers A, and integers k and p, is there a link set S of size k such that $|I(G,S)| = p$? Note that in [25], the Min-Links problem (finding the smallest set of links which can activate the entire network) was shown to be NP-hard, even for graphs of degree 3 and threshold at most 2. This immediately implies the NP-hardness of the Max-Inf-Links problem as well. However, the reduction in [25] yielded a graph that was not bipartite. Here, we extend the result for bipartite graphs.

Theorem 1. *The decision version of the* Max-Inf-Links *problem is NP-hard, even for a single external influencer, and bipartite graphs with all nodes having threshold 1 or 2.*

Proof. We give a reduction from the *Max-Clique problem*: Given a graph $G = (V,E)$ and an integer k, does G contain a clique of size at least k?

Given an instance of the Max-Clique problem (G,k), we construct a bipartite graph G' = ($V_1 \cup V_2$, E', t) as follows. For every node $v \in V$, we create a corresponding node v of threshold 1 in V_1. For every edge $\{u,v\} \in E$, we create a corresponding node (uv) of threshold 2 in V_2. Next, for every edge $\{u,v\} \in E$, we create the edges $(u,(uv))$ and $(v,(uv))$. Clearly, the transformation can be done in $O(V+E)$ time. We show that G has a clique of size k if and only if G' has a link set of size k that can activate at least $C_k^2 + k$ nodes. The full proof is in the appendix. □

4 Optimal Algorithm for Trees

In contrast to the result of the previous section, in this section, we show that the Max-Inf-Links problem can be solved in polynomial time in trees. Let $T = (V,E,t)$ be a tree with n nodes, $V = \{1,2,...,n\}$ and $t : t(v) \rightarrow \mathcal{Z}^+$. Fix an arbitrary root of the tree and order the children of every node in an arbitrary fashion. We define T_v and d_v to be the sub-tree rooted at node v, and the number of children of node v respectively. We define T_v^{-1} to be the same sub-tree as T_v except that the threshold of the root v is reduced by 1. Note that while in the input, all thresholds are ≥ 1, in a tree T_v^{-1}, the threshold of v may be reduced to 0. We also define v_i to be i^{th} child of node v and T_{v_i} to be the sub-tree rooted at v_i.

Given a tree T_v, the optimal solution to the Max-Inf-Links problem for T_v may or may not activate the root. For example, in Figure 2, for the tree on the left, it can be verified that every optimal link set of size 1 activates the root (one such link set gives a link to the root node a and activates a and d), but for the tree on the right, there is no link set of size 1 that activates the node a (an optimal solution gives a link to node b, and activates both b and c) . Let $MIA(T_v,k)$ be the problem of finding an link set of at most k links to nodes in the sub-tree T_v that maximizes the number of influenced nodes *while ensuring that root v is activated*. Similarly, let $MIB(T_v,k)$ be the problem of finding an link set of at most k links to nodes in the sub-tree T_v that maximizes the number of influenced nodes *while ensuring that root v is not activated*. Let $A(T_v,k)$ and $B(T_v,k)$ be the maximum number of nodes that can be influenced by an optimal solution to $MIA(T_v,k)$ and $MIB(T_v,k)$ respectively. Clearly, given a tree $T = (V,E,t)$ rooted at node r, an optimal link set S for problem Max-Inf-Links(T_n,k) either activates or does not activate the root r. Therefore:

$$MI(T_n,k) = max\{A(T_r,k),B(T_r,k)\}$$

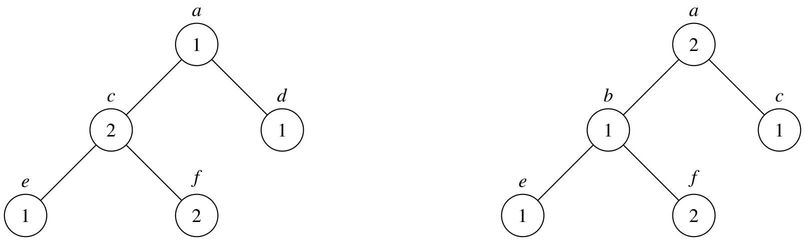

Fig. 2. For the tree on the left, $A(T,1) = 2$, and $B(T,1) = 1$ while for the tree on the right, $A(T,1) = -\infty$, and $B(T,1) = 1$

We start with link sets that do activate the root. We first prove a critical lemma that shows that any link set S that activates the root can be converted to an equivalent link set that gives a link to the root.

Lemma 1. *Suppose S is a link set of size k for a tree T_v in which $v = root(T)$ is activated, and $(\mu,v) \notin S$. Then there exists a link set S' with $(\mu,v) \in S'$ such that $I(T,S') = I(T,S)$ and $|S'| = k$.*

Proof. We prove the lemma by induction on the height of the tree T. Clearly, the lemma is true for trees of height 0; the only way to activate the root in such a tree is to give a link to the root v.

Now consider a tree T of height h and a link set S for T that activates v without giving v a link. Clearly, there must exist a child c of v which is activated before v, and contributes to the activation of v. By the inductive hypothesis, we can assume that c is given a link in S. Consider $S' = S - \{(\mu, c)\} \cup \{(\mu, v)\}$. Any node in $T_v - T_c$ that was activated by S before v, is also activated in S' before v. Therefore v will be activated by S', and will subsequently activate c, and any nodes in T_c that were activated by c. All other children of v that were activated after v in S will also be activated after v by S'. Therefore, $I(T,S) = I(T,S')$. □

Lemma 1 tells us that an optimal link set S of size k that activates the root v of a tree T_v can be assumed to give a link to the root v. Then the remaining $k-1$ links must be given to other nodes in T_v, but since v is activated by S, it must be that at least $t(v)-1$ children of v are activated *before* v and contribute to the activation of v. Observe that once v is activated, the thresholds of its remaining children are effectively reduced by 1. This, together with the links given by the link set S to nodes in the subtrees of the remaining children, may activate some of these children and subsequently other nodes in their subtrees. Let $F_{v,d}$ be the forest of subtrees rooted at the first d children of v. We see that there exists some partition (C,D) of the roots of trees in F_{v,d_v} such that $|C| \geq t(v)-1$ and S activates the nodes in C before v, and subsequently v gets activated, which in turn reduces the thresholds of the nodes in D by one, perhaps contributing to their activation.

We formalize and generalize the above to find appropriate link sets for the forest F_{v,d_v}. Let $F_{v,d}$ be the forest consisting of the subtrees rooted at the first d children of v and let (C,D) be a partition of the roots of the d trees. Let $F(C,D)$ be a forest derived from $F_{v,d}$ by reducing the threshold of all nodes in D. We call a link set for $F(C,D)$ a C-activating link set if it activates all nodes in C, and maximizes the total number of activated nodes in $F(C,D)$. Given a forest F, and an integer i, we call a link set for F an i-first link set if it is a C-activating link set for some partition (C,D) of the roots of trees in F with $|C| \geq i$. We now define $MIA(F_{v,d}, i, k)$ to be the problem of finding an i-first link set of size k that maximizes the total number of activated nodes in F.

Next consider a link set that does not activate v. In this case, at most $t(v)-1$ of v's children can be activated, as otherwise v would also be activated. To find such a link set, we define $MIB(F_{v,d}, i, k)$ to be the problem of finding a link set of size at most k that activates the most nodes in $F_{v,d}$ while activating exactly i roots of trees in $F_{v,d}$. Let $A(F_{v,d}, i, k)$ and $B(F_{v,d}, i, k)$ be the number of influenced nodes by an optimal solution to $MIA(F_{v,d}, i, k)$. and $MIB(F_{v,d}, i, k)$ respectively.

We now give recursive formulations for $A(T_v, k)$, $B(T_v, k)$, $A(F_{v,d}, i, k)$, and $B(F_{v,d}, i, k)$; as we have already observed, they are inter-dependent. We start with a recursive formulation for $A(v,k)$.

Lemma 2.

$$A(T_v, k) = \begin{cases} 1 + A(F_{v,d_v}, 0, k) & \textit{if}\, t(v) = 0 \\ 1 + A(F_{v,d_v}, t(v)-1, k-1) & \textit{if}\, 1 \leq t(v) \leq k \\ -\infty & \textit{if}\, k < t(v) \end{cases}$$

Proof. First suppose $k \geq t(v) = 0$. In this case, there is no need to give v a link, and none of v's children need to be activated before it. Therefore, $A(T_v, k) = 1 + A(F_{v,d_v}, 0, k)$ as claimed.

Next suppose $k \geq t(v) = 1$. Then by Lemma 1, there exists an optimal link set S in which root v receives a link, thereby activating v. It is then straightforward to see that $S - \{v\}$ is also an optimal solution to problem $MIA(F_{v,d_v}, 0, k-1)$. It follows that $A(T_v, k) = 1 + A(F_{v,d_v}, 0, k-1)$ in this case.

Next, suppose $1 < t(v) \leq k$. Then giving root v a link does not suffice to activate v; in fact $t(v) - 1$ of v's children have to be activated before v, otherwise v cannot be activated. By Lemma 1, however, there exists an optimal link set S which gives root v a link. We claim that $S - \{v\}$ is an optimal solution to problem $MIA(F_{v,d_v}, t(v) - 1, k-1)$. Suppose not, let $|S| = k$ and let S' be an optimal solution to $MIA(F_{v,d_v}, t(v) - 1, k-1)$ which can activate more nodes than $S - \{v\}$. With S', $t(v) - 1$ children of v have been activated. These activations, together with the link to v, are enough to activate node v, and would reduce the threshold of the remaining children of v by 1. Therefore $S' \cup \{v\}$ activates v as well as all nodes activated by S' in $F_{v,d(v)}$. Then $S' \cup \{v\}$ will be a link set of k links which can activate more nodes in T_v than S, contradicting the optimality of S. Therefore: $A(T_v, k) = 1 + A(F_{v,d_v}, t(v) - 1, k-1)$.

Finally suppose $t(v) > k \geq 0$. Then we claim it is impossible to activate the root. If it were possible to activate the root using k links, then by Lemma 1, there is an optimal link set that activates the root and gives a link to v. Then the remaining $k-1$ links must be given to nodes in F_{v,d_v}. However using $k - 1 < t(v) - 1$ links, we can activate at most $k-1$ children of v, and together with the link given to v, the reduction in threshold of v is $< t(v)$, a contradiction. Thus in this case $A(T_v, i) = -\infty$ as claimed. This completes the proof of the lemma. □

Lemma 3. $A(F_{v,d}, i, k)$

$$= \begin{cases} -\infty & \textit{if } i > d \\ 0 & \textit{if } i = d = 0 \\ \max\limits_{0 \leq p \leq k} \{A(F_{v,d-1}, i-1, p) + A(T_{v_d}, k-p)\} & \textit{if } i = d > 0 \\ \max \begin{cases} \max\limits_{0 \leq p \leq k} A(F_{v,d-1}, i, p) + max \begin{cases} A(T_{v_d}^{-1}, k-p) \\ B(T_{v_d}^{-1}, k-p) \end{cases} \\ \max\limits_{0 \leq p \leq k} \{A(F_{v,d-1}, i-1, p) + A(T_{v_d}, k-p)\} \end{cases} & \textit{if } 0 < i < d \\ \max\limits_{0 \leq p \leq k} A(F_{v,d-1}, i, p) + max \begin{cases} A(T_{v_d}^{-1}, k-p) \\ B(T_{v_d}^{-1}, k-p) \end{cases} & \textit{if } 0 = i < d \end{cases}$$

Proof. If $i > d$, clearly it is impossible to activate at least i of the first d children of v, therefore $A(F_{v,d}, i, k) = -\infty$. If $i = d = 0$, the forest $F_{v,d}$ is empty, and therefore, no nodes can be influenced by any link set of any size.

If $i = d$, this means that all children v_j with $1 \leq j \leq d$ have to be activated before v. The optimal solution S will assign p links to $F_{v,d-1}$ and $k - p$ links to T_{v_d}, for some p. It follows that $A(F_{v,d}, i, k) = \max\limits_{0 \leq p \leq k} \{A(F_{v,d-1}, i-1, p) + A(T_{v_d}, k-p)\}$ If $i < d$, and v_d is not activated before v by S, observe that v_d may or may not be activated after v.

Besides, due to the fact that its parent has already been activated, the threshold of v_d is effectively reduced by one. An optimal link set S will assign p links to $F_{v,d-1}$ and $k-p$ links to T_{v_d}, therefore we try all possibilities of p to find the best distribution. Therefore:

$$A(F_{v,d},i,k) = \max_{0\le p\le k}\{A(F_{v,d-1},i,p) + max\left\{\begin{array}{l} A(T^{-1}_{v_d},k-p) \\ B(T^{-1}_{v_d},k-p) \end{array}\right.\}$$

Finally, if $i<d$ and v_d is activated before v by S, at least $i-1$ children of v in $F_{v,d-1}$ are required to be activated before v. Since v_d contributes to the activation of v, the threshold of v_d remains unchanged.

$$A(F_{v,d},i,k) = \max_{0\le p\le k}\{A(F_{v,d-1},i-1,p) + A(T_{v_d},k-p)\}$$

When $i=0$, we only need to consider the first of the two situations above. □

Next we consider link sets that do not activate the vertex v (the proof of Lemma 4 is in the appendix).

Lemma 4.

$$B(T_v,k) = \left\{\begin{array}{ll} \max\limits_{0\le i<min(t(v),d_v+1)} B(F_{v,d_v},i,k) & if\, t(v)>0 \\ -\infty & if\, t(v)=0 \end{array}\right.$$

Lemma 5. $B(F_{v,d},i,k)$

$$= \left\{\begin{array}{ll} \max\left\{\begin{array}{l} \max\limits_{0\le p\le k}\{B(F_{v,d-1},i,p)+B(T_{v_d},k-p)\} \\ \max\limits_{0\le p\le k}\{B(F_{v,d-1},i-1,p)+A(T_{v_d},k-p)\} \end{array}\right. & if\, 0<i,d \\ \max\limits_{0\le p\le k}\{B(F_{v,d-1},i,p)+B(T_{v_d},k-p)\} & if\, 0=i<d \\ 0 & if\, d=0 \end{array}\right.$$

Proof. Suppose $i,d>0$. Then in an optimal solution S to $MIB(F_{v,d},i,k)$, either v_d is activated or it is not. If v_d is activated, then some p links are given to nodes in $F_{v,d-1}$ and exactly $i-1$ nodes are activated by S in $F_{v,d-1}$, while the remaining $k-p$ links must constitute an optimal solution to $MIA(T_{v_d},k-p)$. That is, $B(F_{v,d},i,k) = B(F_{v,d-1},i-1,p)+A(T_{v_d},k-p)$. Alternatively, if v_d is not activated by S, then some p links are given to nodes in $F_{v,d-1}$ and exactly i nodes are activated by S in $F_{v,d-1}$, while the remaining $k-p$ links must constitute an optimal solution to $MIB(T_{v_d},k-p)$. That is, $B(F_{v,d},i,k) = B(F_{v,d-1},i,p)+B(T_{v_d},k-p)$. If $i=0$, this means no roots of trees in $F_{v,d}$ are to be activated, which means neither v_d is activated, nor any roots in $F_{v,d-1}$. Finally, if the forest is empty ($d=0$), then clearly the maximum number of nodes that can be influenced is 0, regardless of i and k. □

Theorem 2. *The Maximum Influence problem for a tree $T_n=(V,E,t)$ using k links, can be solved in time $O(n^2k^2)$.*

Proof. The recursive definitions given in Lemmas 2 to 5 can be computed using dynamic programming to obtain an optimal solution to the Max-Inf-Links problem. In the worst case, for any node w with d children, we need to compute $A(F_{v,p},i,j)$ and $B(F_{v,p},i,j)$ for $1\le p\le d$, $0\le i\le d$, and $0\le j\le k$. These are d^2k values, each of which can take $\Theta(k)$ time to compute. The values $A(T_v,k)$ and $B(T_v,k)$ can be computed in constant time, and there are k such values. Using the result of [9], the sum of squares of degrees of a node in a graph is upper bounded by $e(2e/(n-1)+n-2)$, which for a tree is $\Theta(n^2)$. Summing up the time over all vertices in the graph, we obtain a total time of $O(n^2k^2)$. □

5 Faster Algorithm for Paths

The algorithm for trees in the previous section obviously also applies to paths. In this section, we give a $\Theta(kn)$ algorithm for the Max-Inf-Links problem in a path, by exploiting its simpler structure. Let $P_n = (V,E,t)$ be a path with n nodes, $V = \{1,2,...,n\}$, $E = \{(i,(i+1)) \mid 1 \leq i \leq n-1\}$, and $t : t(v) \rightarrow \mathcal{Z}^+$ (the threshold function). For $1 \leq i \leq j \leq n$, we define $P_{i,j}$ to be the sub-path of P_n consisting of all nodes in $\{i,\ldots,j\}$.

An optimal solution to Max-Inf-Links$(P_{i,j},k)$ may or may not activate the node i. Accordingly, let $MIA(P_{i,j},k)$ be the problem of finding a link set of size at most k for the sub-path $P_{i,j}$ that maximizes the number of influenced nodes *while ensuring that node i is activated*. Similarly, let $MIB(P_{i,j},k)$ be the problem of finding a link set of size at most k for the sub-path $P_{i,j}$ that maximizes the number of influenced nodes *while ensuring that node i is not activated*. Let $A(i,j,k)$ and $B(i,j,k)$ be the number of nodes that can be influenced by optimal solutions to $MIA(P_{i,j},k)$ and $MIB(P_{i,j},k)$ respectively.

The key idea of our algorithm is to break the path into subpaths containing only nodes of threshold 1 and 2, separated by nodes of threshold 3 or greater. We give recursive definitions for $A(i,j,k)$ and $B(i,j,k)$ when $P_{i,j}$ has only nodes of threshold 1 or 2. As we will see, these definitions are inter-dependent. We need the following definition.

Definition 2. *Given a path P_n, fix i such that $1 \leq i \leq n$. We define $next(i) = min\{j \mid i < j \leq n+1$ and either $t(j) = 1$ or $j = n+1\}$.*

We see that $next(i)$ is the first node after i to have threshold 1, unless i is the rightmost node in the path with threshold 1, in which case $next(i) = n+1$. Clearly $A(n+1,n,k) = B(n+1,n,k) = 0$ for all k. We now consider the case when $1 \leq i \leq j \leq n$. We start with the case when $t(i) = 2$.

Lemma 6. *Given a sub-path $P_{i,j}$ in which all nodes have threshold 1 or 2, and $t(i) = 2$:*

$$A(i,j,k) = \begin{cases} 0 & \textit{if } next(i) > j \\ 0 & \textit{if } A(i+1,j,k-1) = 0 \\ 1 + A(i+1,j,k-1) & \textit{if } A(i+1,j,k-1) > 0 \end{cases}$$

$$B(i,j,k) = \begin{cases} 0 & \textit{if } next(i) > j \\ max \begin{cases} A(i+1,j,k) \\ B(i+1,j,k) \end{cases} & \textit{if } next(i) \leq j \end{cases}$$

Proof. If $next(i) > j$ then there is no way to activate any node in $P_{i,j}$, therefore $A(i,j,k) = B(i,j,k) = 0$ for every k. If instead $next(i) \leq j$, then it is possible to activate at least one node in $P_{i,j}$. Observe that in any feasible solution for $MIA(P_{i,j},k)$, not only does node i need to receive a link, but its neighbor node $i+1$ needs to be activated as well, therefore $A(i,j,k) = 0$ if $A(i+1,j,k-1) = 0$; otherwise $A(i,j,k) = 1+A(i+1,j,k-1)$. Finally, note that any feasible solution for $MIB(P_{i,j},k)$ is a solution in which i does not receive a link, and the next node may or may not be activated, or it does receive a link, and the next node is not activated. That is $B(i,j,k) = max\{A(i+1,j,k),B(i+1,j,k),B(i+1,j,k-1)\} = max\{A(i+1,j,k),B(i+1,j,k)\}$. □

Next we consider the case when node i has threshold 1. In this case, the optimal substructure of the problem is not so straightforward to prove. The difficulty arises because an optimal solution S to $MIA(P_{i,j},k)$ may or may not activate node $i+1$. In the case when it does not activate node $i+1$, we would like to claim that S consists of a link to i and and an optimal solution to $MIB(P_{i+1,j},k-1)$. However, we do not know whether or not $i+2$ was activated in a solution to $MIB(P_{i,j},k)$. So when we combine such a solution with a link to node i, we may or may not activate node $i+1$.The following technical lemma uncovers the structure of optimal solutions (the proof is in the appendix).

Lemma 7. *Let $P_{i,j}$ be a path with $t(i) = 1$ and $next(i) \leq j$. If in every optimal solution for $MIA(P_{i,j},k)$, node i receives a link, then there exists an optimal solution S for $MIA(P_{i,j},k)$ in which neither $i+1$ nor $i+2$ receives a link.*

The following lemma summarizes the optimal substructure of the problem when $t(i) = 1$.

Lemma 8. *Given a sub-path $P_{i,j}$ in which all nodes have threshold 1 or 2, and $t(i) = 1$:*

$$A(i,j,k) = \begin{cases} min\{k, j-i+1\} & \textit{if } next(i) > j \\ max\begin{cases} 1+A(i+1,j,k) \\ 1+B(i+1,j,k-1) \end{cases} & \textit{if } next(i) \leq j \end{cases}$$

$$B(i,j,k) = \begin{cases} 0 & \textit{if } next(i) > j \\ B(i+1,j,k) & \textit{if } next(i) \leq j \end{cases}$$

Proof. First we prove the correctness of the recursive definitions for $A(i,j,k)$. If $next(i) > j$, then node i must receive a link, as it is not possible otherwise to activate any node in $P_{i+1,j}$. Thus node i receives a link and is activated first. Inductively, we can show that node $i+1$ is the second node which receives a link and gets activated (see Figure 3). Therefore, all nodes in $\{i,\ldots,i+min\{k,j-i+1\}-1\}$ must receive a link. Any node that does not receive a link cannot get activated, the maximum number of activated nodes is $min\{k,n-i+1\}$. This shows that $A(i,j,k) = min\{k,n-i+1\}$ in this case.

Otherwise, $next(i) \leq j$, that is, node i is not the rightmost node with threshold 1. First observe that either there exists an optimal solution for $MIA(P_{i,j},k)$ in which node i does not receive a link, or in every optimal solution for $MIA(P_{i,j},k)$, node i receives a link. In the first case, let S be an optimal solution for $MIA(P_{i,j},k)$ in which node i does not receive a link. It follows that its neighbor node $i+1$ was also activated.

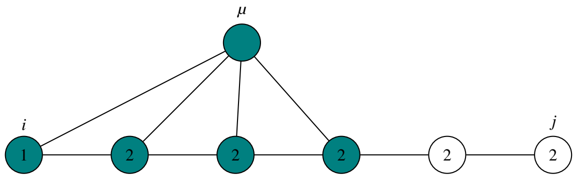

Fig. 3. An optimal link set S for $MIA(P_{i,j},4)$: the case when $t(i)=1$ and $next(i)>j$

Clearly S must be an optimal solution for $MIA(P_{i+1,j},k)$ (if not, and if S' is a solution for $MIA(P_{i+1,j},k)$ that activates more nodes than S, then S' is also a better solution for $MIA(P_{i,j},k)$, contradicting the optimality of S. Therefore, $A(i,k)=1+A(i+1,k)$.

In the second case, by Lemma 7, we have an optimal solution S' in which nodes $i+1$ and $i+2$ do not receive links and are therefore not activated. Furthermore, using a cut-and-paste argument, it is straightforward to see that $S''=S'-\{i\}$ is an optimal solution for $MIB(P_{i+1,j},k-1)$. It follows that $A(i,k)=1+B(i+1,k-1)$.

Finally, any solution in which node i is not activated, we can be sure that neither node i gets a link, nor does its neighbor, node $i+1$ get activated. Therefore $B(i,k)=B(i+1,k)$. This completes the proof. □

We now use the definitions of $A(i,j,k)$ and $B(i,j,k)$ to prove the main result of this section:

Theorem 3. Max-Inf-Links *(P_n,k) can be solved in time $\Theta(kn)$.*

Proof. If P_n has no nodes of threshold >2, then clearly $MI(P_n)=max\{A(1,n,k),B(1,n,k)\}$. If not, let q be the largest index of a node in P_n such that $t(q)\geq 3$. If $t(q)>3$, then q cannot be activated, therefore, there is no need to give a link to q. That is, the optimal solution to the Max-Inf-Links problem with k links does not give node q a link, and instead uses ℓ links to solve the Max-Influence problem on the path $P_{1,q-1}$ and $k-\ell$ links to solve the Max-Influence problem on the path $P_{q+1,n}$ for some value of ℓ such that $0\leq\ell\leq k$. Thus the optimal solution can be found by checking for all possible values of ℓ between 0 and k, yielding:

$$MI(P_n,k)=max_{0\leq\ell\leq k}\{MI(P_{1,q-1},\ell)+MI(P_{q+1,n},k-\ell)\}$$

Finally if $t(q)=3$, then either there exists an optimal solution in which q is not activated, in which case $MI(P_n,k)$ is defined identically to the case when $t(q)>3$, or in every optimal solution q is activated. In this case, let S be an optimal solution to $MI(P_n,k)$. By assumption, q is activated. Since $t(q)=3$, it must be that q gets a link, and also that $q-1$ and $q+1$ are activated. Let $S=S_1\cup\{q\}\cup S_2$ where $S_1=\{1,\ldots,q-1\}\cap S$ and $S_2=S\cap\{q+1,n\}$, and let $|S_1|=\ell$ and $|S_2|=k-1-\ell$ We now claim that S_2 is an optimal solution to $MIA(P_{q+1,n},k-1-\ell)$. If there was a better solution to $MIA(P_{q+1,n},k-1-\ell)$ than S_2, we can replace S_2 by the claimed better solution in S to get a better solution to Max-Inf-Links(P_n,k), a contradiction to the optimality of S.

Next we claim that S_1 is an optimal solution to Max-Inf-Links$(P_{1,q-1},\ell)$. Suppose instead that S_1 activates α nodes in $P_{1,q-1}$ and there is another set of links S' of size

ℓ that activates $\beta > \alpha$ nodes in $P_{1,q-1}$. If S' activates $q-1$, then clearly $S' \cup \{q\} \cup S_2$ activates more nodes than S in P_n, a contradiction to the optimality of S. If S' does not activate q, then $S' \cup S_2$ does not activate node q but activates at least the same number of nodes in P_n as S does, a contradiction to the assertion that every optimal solution activates q. Therefore, we conclude that

$$MI(P_n,k) = max_{0 \leq \ell \leq k}\{1 + MI(P_{1,q-1},\ell) + A(P_{q+1,n},k-1-\ell)\}$$

Finally, we prove that the above formulation can be computed using dynamic programming in time $\Theta(kn)$. For any sub-path$P_{i,j}$ containing only nodes of threshold 1 and 2, the values of $A(i,j,r)$, $B(i,j,r)$ and $MI(i,j,r)$ can be found using the definitions in Lemmas 8 and 6 for all $0 \leq r \leq k$ in time $\Theta(k(j-i+1)$. Since the total lengths of all sub-paths is at most n, the total time spent is $\Theta(nk)$. Since there are $O(n)$ such sub-paths, the recursive formulation for $MI(P_n,k)$ above for a paths with no threshold limit takes another $O(kn)$ time to compute. □

5.1 Cycles

By taking out a single node from the cycle, and considering the resulting path, we can obtain an algorithm for the Max-Inf-Linksproblem for cycles, as proved in the appendix:

Theorem 4. *The* Max-Inf-Links *problem for a cycle C_n using k links, can be solved in time* $\theta(kn^2)$.

6 $\Theta(n)$ Algorithm for Cliques

In this section, we give a linear time greedy algorithm for the Max-Inf-Links problem in cliques (the proof is in the appendix).

Theorem 5. Max-Inf-Links*(K_n,k) can be solved in time* $\Theta(n)$.

7 Discussion

In this paper, we introduced and studied the Max-Influence-with-Links problem: given a social network G where every node v has a threshold $t(v)$ to be activated, and an integer k, which k nodes in G should an already activated external influencer μ befriend, so as to influence the maximum possible number of nodes in the network? We showed that the problem is NP-complete, even for a single influencer, and bipartite graphs with maximum threshold 2. In contrast, for the case of a single external influencer, we showed a linear time algorithm for cliques, $\Theta(kn)$ algorithm for paths, $\Theta(kn^2)$ algorithm for cycles, and a $\Theta(k^2n^2)$ algorithm for trees. It seems straightforward to generalize our algorithms for any number k of external influencers. It would be interesting to study the complexity of the problem for graphs of bounded tree width, and the approximability of the problem for constant k. We are also interested in studying the case with non-uniform weights on the edges. Clearly, the problem remains NP-complete in general, but the complexity for special classes of graphs remains open.

References

1. O. Ben-Zwi, D. Hermelin, D. Lokshtanov, and I. Newman. Treewidth governs the complexity of target set selection. *Discrete Optimization*, 8:702–715, 2011.
2. C. Borgs, M. Brautbar, J. Chayes, and B. Lucier. Maximizing social influence in nearly optimal time. In *Proceedings of the ACM-SIAM Symposium on Discrete Algorithms*, SODA '14, pages 946–957, 2014.
3. J. J. Brown and P. H. Reingen. Social ties and word-of-mouth referral behavior. *Journal of Consumer research*, pages 350–362, 1987.
4. N. Chen. On the approximability of influence in social networks. In *Proceedings of the Symposium on Discrete Algorithms*, SODA '08, pages 1029–1037, 2008.
5. W. Chen, Y. Wang, and S. Yang. Efficient influence maximization in social networks. In *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '09, pages 199–208, 2009.
6. F. Cicalese, G. Cordasco, L. Gargano, M. Milanic, J. G.Peters, and U. Vaccaro. How to go viral: Cheaply and quickly. In *Fun with Algorithms*, volume 8496 of *Lecture Notes in Computer Science*, pages 100–112. 2014.
7. F. Cicalese, G. Cordasco, L. Gargano, M. Milanic, and U. Vaccaro. Latency-bounded target set selection in social networks. In *The Nature of Computation. Logic, Algorithms, Applications*, volume 7921 of *Lecture Notes in Computer Science*, pages 65–77. 2013.
8. G. Cordasco, L. Gargano, A. A. Rescigno, and U. Vaccaro. Optimizing Spread of Influence in Social Networks via Partial Incentives. In *Proceedings of SIROCCO*, pages 119–134. 2015.
9. D. de Caen. An upper bound on the sum of squares of degrees in a graph. *Discrete Mathematics*, 185:245–248, 1998.
10. E. D. Demaine, M. T. Hajiaghayi, H. Mahini, D. L. Malec, S. Raghavan, A. Sawant, and M. Zadimoghadam. How to influence people with partial incentives. In *Proceedings of the International Conference on World Wide Web*, WWW '14, pages 937–948, 2014.
11. T.N. Dinh, H. Zhang, D.T. Nguyen, and M.T. Thai. Cost-effective viral marketing for time-critical campaigns in large-scale social networks. *Networking, IEEE/ACM Transactions on*, PP(99):1–1, 2014.
12. P. Domingos and M. Richardson. Mining the network value of customers. In *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '01, pages 57–66, 2001.
13. M. Eftekhar, Y. Ganjali, and N. Koudas. Information cascade at group scale. In *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '13, pages 401–409, 2013.
14. M. A. Fazli, M. Ghodsi, J. Habibi, P. J. Khalilabadi, V. Mirrokni, and S. S. Sadeghabad. On the non-progressive spread of influence through social networks. In *LATIN 2012: Theoretical Informatics*, volume 7256 of *Lecture Notes in Computer Science*, pages 315–326. 2012.
15. L. Gargano, P. Hell, J. Peters, and U. Vaccaro. Influence diffusion in social networks under time window constraints. In *Proceedings of SIROCCO*, volume 8179 of *Lecture Notes in Computer Science*, pages 141–152. 2013.
16. J. Goldenberg, B. Libai, and E. Muller. Talk of the network: A complex systems look at the underlying process of word-of-mouth. *Marketing Letters*, 12:211–223, 2001.
17. A. Goyal, F. Bonchi, and L. V.S. Lakshmanan. A data-based approach to social influence maximization. *Proceedings of the VLDB Endowment*, 5:73–84, 2011.
18. A. Goyal, F. Bonchi, L. V.S. Lakshmanan, and S. Venkatasubramanian. On minimizing budget and time in influence propagation over social networks. *Social Network Analysis and Mining*, 3:179–192, 2013.

19. A. Goyal, W. Lu, and L. V.S. Lakshmanan. Celf++: Optimizing the greedy algorithm for influence maximization in social networks. In *Proceedings of the International Conference Companion on World Wide Web*, WWW '11, pages 47–48, 2011.
20. D. Gunnec and S. Raghavan. Integrating social network effects in the share-of-choice problem. Technical report, University of Maryland, College Park, 2012.
21. D. Gunnec, S. Raghavan, and R. Zhang. The least cost influence problem. Technical report, University of Maryland, College Park, 2013.
22. J. He, S. Ji, R. Beyah, and Z. Cai. Minimum-sized influential node set selection for social networks under the independent cascade model. In *Proceedings of the ACM International Symposium on Mobile Ad Hoc Networking and Computing*, MobiHoc '14, pages 93–102, 2014.
23. D. Kempe, J. Kleinberg, and É. Tardos. Maximizing the spread of influence through a social network. In *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '03, pages 137–146, 2003.
24. D. Kempe, J. Kleinberg, and É. Tardos. Influential nodes in a diffusion model for social networks. In *Proceedings of the International Conference on Automata, Languages and Programming*, ICALP'05, pages 1127–1138, 2005.
25. M. Lafond, L. Narayanan, and K. Wu. Whom to befriend to influence people. In *Proceedings of SIROCCO*, 2016.
26. H. Lamba and J. Pfeffer. Maximizing the spread of positive influence by deadline. In *Proceedings of the International Conference Companion on World Wide Web*, WWW '16, pages 67–68, 2016.
27. J. Leskovec, L. A. Adamic, and B. A. Huberman. The dynamics of viral marketing. In *Proceedings of the ACM Conference on Electronic Commerce*, pages 228–237, 2006.
28. W. Lu, F. Bonchi, A. Goyal, and L. V.S. Lakshmanan. The bang for the buck: Fair competitive viral marketing from the host perspective. In *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '13, pages 928–936, 2013.
29. S. Lv and L. Pan. Influence maximization in independent cascade model with limited propagation distance. In *Web Technologies and Applications*, volume 8710 of *Lecture Notes in Computer Science*, pages 23–34. 2014.
30. E. Mossel and S. Roch. Submodularity of influence in social networks: From local to global. *SIAM Journal on Computing*, 39:2176–2188, 2010.
31. H. T. Nguyen, M. T. Thai, and T. N. Dinh. Stop-and-stare: Optimal sampling algorithms for viral marketing in billion-scale networks. In *Proceedings of the International Conference on Management of Data*, SIGMOD '16, pages 695–710, 2016.
32. A. Nichterlein, R. Niedermeier, J. Uhlmann, and M. Weller. On tractable cases of target set selection. *Social Network Analysis and Mining*, 3(2):233–256, 2012.
33. M. Richardson and P. Domingos. Mining knowledge-sharing sites for viral marketing. In *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '02, pages 61–70, 2002.
34. S. Tang and J. Yuan. Going viral: Optimizing discount allocation in social networks for influence maximization. *CoRR*, abs/1606.07916, 2016.
35. Y. Tang, Y. Shi, and X. Xiao. Influence maximization in near-linear time: A martingale approach. In *Proceedings of the ACM SIGMOD International Conference on Management of Data*, SIGMOD '15, pages 1539–1554, 2015.
36. Y. Tang, X. Xiao, and Y. Shi. Influence maximization: Near-optimal time complexity meets practical efficiency. In *Proceedings of the ACM SIGMOD International Conference on Management of Data*, SIGMOD '14, pages 75–86, 2014.
37. Y. Yang, X. Mao, J. Pei, and X. He. Continuous influence maximization: What discounts should we offer to social network users? In *Proceedings of the International Conference on Management of Data*, SIGMOD '16, pages 727–741, 2016.

A Appendix

Proof of Theorem 1. We show that G has a clique of size k if and only if G' has a link set of size k that can activate at least C_k^2+k nodes. Figure 4 illustrates the reduction.

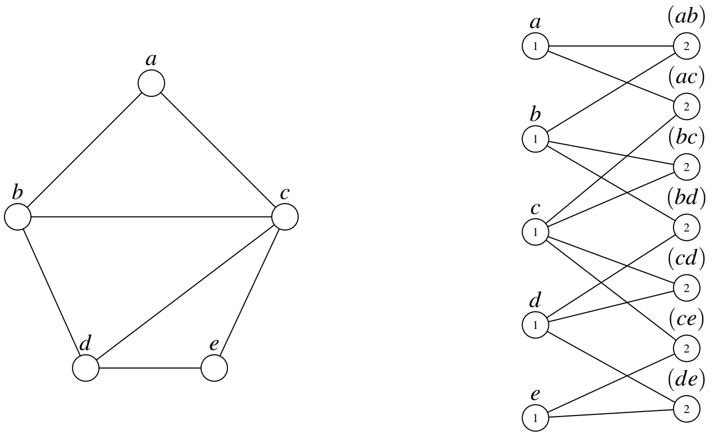

Fig. 4. Reduction from G (top) to G' (bottom); thresholds in G' are indicated inside the circles representing the nodes.

To do this, we first show that it suffices to consider link sets that contain only nodes in V_1.

Claim. For any link set $T \subseteq V_1 \cup V_2$, there exists a set $S \subseteq V_1$ such that $|S| \leq |T|$ and $|I(G',S)| \geq |I(G',T)|$.

Proof. Consider a node $v \in V_2$ that receives a link in T, and is connected to v_1 and v_2 $\in V_1$. We argue that we can either remove the link assigned to v, or assign that link to some other node in V_1 while not decreasing the size of the activated set. The following four cases about the time of activation of v are exhaustive.

Case 1: v is not activated by T: We can simply remove the link assigned to v.

Case 2: v is activated before v_1 and v_2: Since v has threshold 2 while v_1 and v_2 are both of threshold 1, this is impossible with a single influencer, as the link v receives can only reduce its threshold by 1.

Case 3: v is activated after v_1 (or v_2) is activated and before v_2 (rest. v_1) is activated: We can move the link assigned for v to v_2 (resp. v_1); the same set of nodes will be activated eventually.

Case 4: v is activated after v_1 and v_2 are activated: The link given to v is unnecessary and can be removed.

Claim. G' has a link set $S \subseteq V_1$ with $|S| \leq k$ and $|I(G',S)| \geq C_k^2 + k$ if and only if G has a clique of size k.

Proof. Suppose first that G has a clique V' of size k. We claim that $V' \subseteq V_1$ is a link set such that $|I(G',V')| = C_k^2 + k$. Clearly, all nodes in V' are activated at round 1. Then since V' is a clique in G, it is easy to see that all nodes in V_2 corresponding to the C_k^2 edges between nodes in V' will be activated in round 2. This proves that $|I(G',V')| \geq C_k^2 + k$.

Suppose next that there is a link set $S \subseteq V_1$ such that $|I(G',V')| = C_k^2 + k$, then we claim that the corresponding set for S in G forms a clique of size k.

We claim that given $S \subseteq V_1$ as a link set, it is impossible to activate any new node $\in V_1$ except nodes in S and for any node $(uv) \in V_2$ that gets activated, it must be that $u \in S$ and $v \in S$.

Suppose there exists a node $d \in V_1 - S$ such that d is eventually activated. Since d has threshold 1 and d doesn't receive a link, in order for d to be activated, one of d's neighbors must be activated first. Let us say d is connected to (dx) and (dx) is activated before d. Node (dx) is of threshold 2, thus in order for (dx) to be activated, both d and x must be activated before (dx), a contradiction.

For the second part, suppose there exists a node $(uv) \in V_2$ such that $u \notin S$ and (uv) gets activated eventually. But from the previous analysis, we know that u can never be activated. Therefore (uv) cannot be activated either.

We have shown that with a set $S \subseteq V_1$ of size k, only nodes in S can be activated for the nodes in V_1; only those nodes which are connected to two nodes in S can be activated for nodes in V_2. If $|I(G',S)| \geq C_k^2$ + k, then for every pair of nodes in S, they must be connected in G, thus S must be a clique of size k in G.
This completes the proof of the reduction. ■

Proof of Lemma 4. First observe that if $t(v) = 0$, it is impossible to find a link set that does not activate v. Next, assume $t(v) > 0$. Consider a minimum-sized link set S of at most k links to nodes in the tree T_v that maximizes the number of influenced nodes while not activating the root v. Clearly, S does not give a link to v. Furthermore, suppose S activates $i \geq 0$ of the children of v. Then $i < t(v)$, as otherwise S would also activate v. Then S must also be an optimal solution to $.MIB(F_{v,d}, i, k)$. The lemma follows. ■

Proof of Lemma 7. Suppose in every optimal solution for $MIA(P_{i,j}, k)$, node i receives a link. Let S be an optimal solution for $MIA(P_{i,j}, k)$ which uses the fewest links possible.

First we show that $next(i) > i+2$. By assumption, node i receives a link in S. Observe that if $next(i) = i+1$, then $i+1$ cannot have a link since that would contradict the minimality of S. So we can simply move the link from node i to $i+1$ and activate the same set of nodes, but this contradicts the assumption that in every optimal solution, node i must receive a link. Next suppose $next(i) = i+2$. Then it is not possible that both $i+1$ and $i+2$ receive links, as this would contradict the minimality of S. If exactly one of $i+1$ and $i+2$ have a link, we can move the link to i to the node among $i+1$ and $i+2$ that does not have a link, thus creating a solution which activates exactly the same set of nodes, a contradiction. If neither $i+1$ nor $i+2$ has a link, and neither is activated, then the lemma is proved. Finally, if neither has a link, but one of them is activated, it must be that $i+2$ is activated by node $i+3$. In this case, we can move the link from node i to node $i+1$, getting a solution that activates the same set of nodes, a contradiction to the assumption that every optimal solution must give i a link. We conclude that $next(i) > i+2$, that is, there are at least two nodes in between i and $next(i)$.

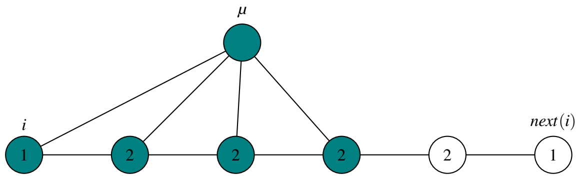

Fig. 5. The case when $next(i)$ is not activated. At least one node in $\{i+1,\ldots,next(i)-1\}$ does not receive a link.

We now show that we can always change S to a solution S' that does not activate $i+1$ or $i+2$ but activates the same number of nodes as S overall. Let $S_1 = S \cap \{i+1,\ldots,next(i)-1\}$ and $S_2 = S \cap \{next(i),\ldots,j\}$. First suppose $next(i)$ is not activated by S. Then $next(i)$ did not get a link, and there must be at least one other node in $\{i+1,\ldots,next(i)-1\}$ that did not receive a link, as if all such nodes received a link, $next(i)$ would be activated (see Figure 5). Therefore $|S_1| \leq next(i)-i-2$. Consider now the link set S' made by *shifting* all the links in S_1 from $i+1$ onwards to a consecutive sequence of nodes ending with $next(i)$, that is, $S' = \{i\} \cup \{next(i)-|S_1|+1,\ldots next(i)\} \cup S_2$. Then S' is the same size as S, it activates exactly the same number of nodes as S, and is therefore also optimal. Observe that $next(i)-|S_1|+1 > i+2$, since $|S_1| \leq next(i)-i-2$. Thus S' is an optimal solution to $MIA(P_{i,j}$ that does not give links to $i+1$ and $i+2$ as needed.

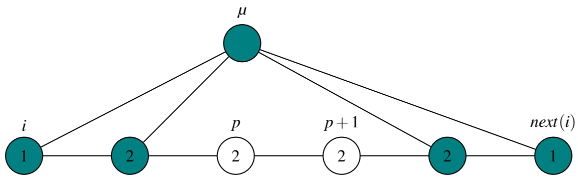

Fig. 6. The case when $next(i)$ is activated either by a link or by $next(i)+1$. At least two nodes in $\{i+1, next(i)-1\}$ do not receive links.

Next suppose $next(i)$ is activated by S. If $next(i)$ was activated by $next(i)-1$, then all nodes in $\{i+1,\ldots,next(i)-1\}$ must have received links, which contradicts either the minimality of S, or the assertion that i gets a link in every optimal solution. So we conclude that either $next(i)$ received a link in S or was activated by $next(i)+1$. Since S is minimal, there must exist at least one node in $\{i+1,\ldots,next(i)-1\}$ that did not receive a link. If there is exactly one such node p, then clearly every node in $\{i,\ldots,next(i)\}$ is activated by S, but then solution S' which moves the link given to i to node p also activates the same nodes, and contradicts the assertion that every optimal solution must give a link to i. Therefore, there must be two nodes in $\{i+1,\ldots,next(i)-1\}$ that do not receive links (see Figure 6). Therefore $|S_1| \leq next(i)-i-3$. Consider now the link set S' made by *shifting* all the links in S_1 from $i+1$ onwards to a consecutive sequence of nodes ending with $next(i)-1$, that is, $S' = \{i\} \cup \{next(i)-|S_1|,\ldots next(i)-1\} \cup S_2$. Then S' is the same size as S, it activates exactly the same number of nodes as S, and is therefore also optimal. Observe that $next(i) - |S_1| > i+2$, since $|S_1| \leq next(i) - i - 3$. Thus S' is an optimal solution to $MIA(P_{i,j}, k)$ that does not give links to $i+1$ and $i+2$ as needed. ■

Proof of Theorem 4. Let $ML(C_n)$ be the minimum number of links needed to activate *all* the nodes in C_n. First, we check using the algorithm in [25] for the Minimum Links problem if $ML(C_n) \leq k$. If so, clearly $MI(C_n, k) = n$; otherwise it is not possible to activate all the nodes in the ring with k links, and there must exist some node p in C_n which is not activated by an optimal link set S for the problem Max-Inf-Links(C_n, k). Since S is optimal, it is clear that S cannot give a link to node p. It is straightforward to see that that S must be an optimal link set for problem Max-Inf-Links$(P_{p+1,p-1}, k)$. Therefore, $MI(C_n, k) = MI(P_{p+1,p-1}, k)$. In order to find such a node p , we can try every possibility of p $(1 \leq p \leq n)$ to find the maximum $MI(P_{p+1,p-1}, k)$, this maximum value is equal to $MI(C_n, k)$.

$$MI(C_n, k) = \begin{cases} n & \text{if } ML(C_n) \leq k \\ \max\limits_{1 \leq i \leq n} \{MI(P_{i+1,i-1}, k)\} & \text{otherwise} \end{cases}$$

Checking if $ML(C_n) \leq k$ can be done in linear time [25]. The time complexity of the Max-Inf-Links algorithm for a path is $\Theta(kn)$ by Theorem 3. Therefore, the Max-Inf-Links problem for a ring C_n can be solved in time $\theta(kn^2)$. ■

Proof of Theorem 5. We give a greedy algorithm for the problem. Since any node of threshold $> n$ cannot possibly be activated, we simply discard such nodes. Next, we sort the nodes in order of threshold. Since all nodes have threshold $\leq n$, we can use counting sort and do the sorting in linear time. It is easy to see that if $t(1) > 1$, there is no way to influence any node, that is $MI(K_n, k) = 0$. Assume now that $t(1) = 1$. We now give a link to node 1, and let $j > 1$ be the minimum value such that $t(j) > 1$. If no such j exists, then 1 link suffices to influence every node in K_n and we are done. Otherwise, all nodes in $\{1, \ldots, j-1\}$ would be successively activated by the link given to node 1; we remove all these nodes and their incident edges, decrement by $j-1$ the thresholds of all nodes $\geq j$ and solve the problem recursively on the resulting clique K' using $k-1$ links. That is, we claim $MI(K_n, k) = j - 1 + MI(K', k-1)$.

We now show that there exists an optimal solution that contains the node 1. Let S be an optimal solution for problem Max-Inf-Links(K_n, k), and let node i be the node with smallest index in S. If $i = 1$, we are done, so assume $i > 1$. It must be that $t(i) = 1$, as otherwise it is impossible to activate any node, therefore, we can create a new solution $S' = S \cup \{1\} - \{i\}$. By the activation of node 1, node i would be also activated automatically in the next round because of internal influence, thus $I(K_n, S) = I(K_n, S')$, and S' is an optimal solution that contains node 1 as needed.

Next we show that if S is an optimal solution for the problem Max-Inf-Links(K_n, k) in which node 1 receives a link, then $S' = S - \{1\}$ is an optimal solution for the problem Max-Inf-Links$(K', k-1)$. Suppose there is a solution S'' to K' that activates more nodes than S'. Consider $S'' \cup \{1\}$ as a solution to K_n. The link to node 1 activates all nodes in $\{1, \ldots, j-1\}$, and the links in S'' now activate all the same nodes they activate in C', since these nodes now have the same effective thresholds they have in C'. Therefore, $S'' \cup \{1\}$ must activate more nodes than S in the clique K_n, contradicting the optimality of S.

The greedy algorithm above can be implemented iteratively as follows: We examine the nodes in order while we still have links to assign. When we process node i, if $t(i) > i$, we stop assigning links and break. If $t(i) < i$, we simply increment i and continue. Finally if $t(i) = i$, we give a link to node i. Clearly the implementation takes $\Theta(n)$ time. ■