SIAM J. DISCRETE MATH.
Vol. 23, No. 3, pp. 1400–1415

ON THE APPROXIMABILITY OF INFLUENCE IN SOCIAL NETWORKS*

NING CHEN†

Abstract. In this paper, we study the spread of influence through a social network in a model initiated by Kempe, Kleinberg, and Tardos [*Maximizing the spread of influence through a social network*, in Proceedings of the 9th ACM SIGKIDD International Conference, Washington, D.C., 2003, pp. 137–146], [*Influential nodes in a diffusion model for social networks*, in Proceedings of the 32nd International Colloquium on Automata, Languages, and Programming (ICALP), Lisbon, Portugal, CITI, 2005, pp. 1127–1138]: Given a graph modeling a social network, where each node v has a (fixed) threshold t_v, the node will adopt a new product if t_v of its neighbors adopt it. Our goal is to find a small set S of nodes such that targeting the product to S would lead to adoption of the product by a large number of nodes in the graph. We show strong inapproximability results for several variants of this problem. Our main result says that the problem of minimizing the size of S, while ensuring that targeting S would influence the whole network into adopting the product, is hard to approximate within a polylogarithmic factor. This implies a similar result if only a fixed fraction of the network is ensured to adopt the product. Further, the hardness of approximation result continues to hold when all nodes have majority thresholds or have constant degrees and thresholds two. The latter answers a complexity question proposed in [P. A. Dreyer, *Applications and Variations of Domination in Graphs*, Ph.D. thesis, Rutgers University, Piscataway, NJ, 2000], [F. S. Roberts, *Graph-theoretical problems arising from defending against bioterrorism and controlling the spread of fires*, in Proceedings of DIMACS/DIMATIA/Renyi Combinatorial Challenges Conference, Piscataway, NJ, 2006]. When the underlying graph is a tree, we give a polynomial-time algorithm to find an optimal solution.

Key words. approximability, social networks

AMS subject classifications. 68W01, 91D30

DOI. 10.1137/08073617X

1. Introduction.

It is well-documented that information spreads via social networks. The dynamic processes governing the diffusion of information and "word-of-mouth" effects have been studied in many fields, including epidemiology [9, 26, 31], sociology [27, 28, 35, 24, 7], economics, and computer science [11, 30, 16, 15, 17, 5, 8, 25, 14, 18, 3, 6]. For example, a recently studied problem in the area of viral marketing is the following: Suppose that we would like to market a new product and hope it will be adopted by a large fraction of individuals in the network. Which set of individuals should we "target" (for instance, one form of "targeting" involves offering free samples of the product)? The answer to the question depends crucially on the network structure and the extent to which "word-of-mouth" effects will take hold.

One simple way to model diffusion with discrete dynamics is to assume that each individual in the network has a "threshold": The individual becomes *influenced* (i.e., adopts the new product) if a certain prespecified number of its neighbors have adopted the product. A natural algorithmic problem arises: Given knowledge of these thresholds, which individuals should be targeted so as to create a large wave of adoptions? Domingos and Richardson [11, 30] studied this problem in a probabilistic setting, and heuristic solutions were given. Kempe, Kleinberg, and Tardos [16, 17]

*Received by the editors September 23, 2008; accepted for publication (in revised form) June 17, 2009; published electronically September 23, 2009. A preliminary version of the paper appeared in proceedings of the ACM-SIAM Symposium on Discrete Algorithms (SODA), 1029-1037, 2008.
http://www.siam.org/journals/sidma/23-3/73617.html

†Division of Mathematical Sciences, School of Physical and Mathematical Sciences, Nanyang Technological University, Singapore (ningc@ntu.edu.sg).

modeled the question as an optimization problem, showed that it is NP-hard to compute the optimal subset to target, and developed approximation algorithms in a submodular framework. Other related literature about diffusion with thresholds includes, e.g., [23, 27, 28, 12, 7, 25].

In many studies, e.g., [23, 27, 12, 9, 31], researchers are interested in the long-term effects of diffusion and whether some consensus can be reached. For example, in a virus propagation network, which nodes should be immunized so that the whole network is protected? Related work can be found in, e.g., [9, 26, 13]. In applications like this, a key requirement is that all (or a large fraction of) individuals in the network are influenced. In the current paper, we focus on this problem in the threshold model. In particular, we address the question as an optimization problem: Find a small set S of individuals such that targeting S would lead to influencing a large fraction of individuals in the network.

1.1. The model. We now define the problem formally. Given a connected undirected graph $G = (V, E)$, let $d(v)$ be the degree of $v \in V$. For each $v \in V$, there is a *threshold* value $t(v) \in \mathbb{N}$, where $1 \leq t(v) \leq d(v)$. Initially, the states of all vertices are *inactive*. We pick a subset of vertices, the *target set*, and set their state to be *active*. After that, in each discrete time step, the states of vertices are updated according to the following rule: An inactive vertex v *becomes* active if at least $t(v)$ of its neighbors are active. The process runs until either all vertices are active or no additional vertices can update states from inactive to active (it is easy to verify that the process runs at most $n - 1$ rounds, where $n = |V|$ is the number of vertices in the graph). The process we consider is *progressive*; i.e., a vertex can only become active from inactive but not vice versa.

We are interested in the following optimization problem, called Target Set Selection: Which subset of vertices should be targeted at the beginning such that all (or a fixed fraction of) vertices in the graph are active at the end? Observe that a trivial solution is to target all vertices in the graph. The goal we consider in this paper is to minimize the size of the target set.

Our model is different from Kempe, Kleinberg, and Tardos [16, 17] in the following two respects: First, Kempe, Kleinberg, and Tardos [16, 17] focused on the maximization problem—for any given k, find a target set of size k to maximize the (expected) number of active vertices at the end of the process. In our paper, however, we ask for a target set of minimum size that guarantees that all (or a fixed fraction of) vertices are eventually active. Second, we consider deterministic, explicitly given, thresholds, whereas the main focus of [16, 17] was on probabilistic thresholds where all thresholds are drawn randomly from a given distribution. (For deterministic thresholds, Kempe, Kleinberg, and Tardos [16] showed strong hardness of approximation results for the maximization problem.)

1.2. Our results. For the general Target Set Selection problem, we show a polylogarithmic lower bound on the approximation ratio. Specifically, the Target Set Selection problem cannot be approximated within a ratio of $O(2^{\log^{1-\epsilon} n})$ for any fixed constant $\epsilon > 0$, unless $NP \subseteq DTIME(n^{polylog(n)})$. Our proof is based on a reduction from the minimum representative problem [21, 22].

Our result gives further evidence that, without additional assumptions such as the probabilistic thresholds in [16, 17], the problem is completely intractable (even in constant degree graphs with thresholds of at most two). Indeed, in the maximization problem studied in [16, 17], for deterministic thresholds, the problem is NP-hard to approximate within a ratio of $n^{1-\epsilon}$ [16]. In related work, Aazami and Stilp [1] studied

a nonthreshold propagation process called power dominating set and showed a similar hardness of approximation result.

Our result implies the same hardness of approximation ratio if, instead of ensuring all vertices in the network are active, we need only to activate a fixed fraction of vertices. Our hardness result gives a negative answer to the problem proposed by Roberts [31]—what vertices need to be "vaccinated" to make sure a virus does not spread to a fixed fraction of the whole network?

By considering different types of thresholds and network structures, we show the following additional results.

Majority thresholds. One important and well-studied threshold is majority, where a vertex becomes active if at least half of its neighbors are active. It has many applications in distributed computing, voting systems, etc. [27]. For example, Peleg [28] proposed the use of a majority update rule for maintaining data consistency in a distributed system. Peleg [27] proved that it is NP-hard to compute the optimal target set for majority thresholds. For different variants of majorities and progressive processes, different lower bounds on the size of the target set were obtained [23, 28, 7]. For further information about majority thresholds, see [27].

For the majority thresholds setting, we show that the problem shares the same hardness of approximation ratio as the general setting. In particular, this implies that the majority thresholds setting does not admit any approximation algorithm of a ratio better than $O(2^{\log^{1-\epsilon} n})$ for any fixed constant $\epsilon > 0$. To the best of our knowledge, this is the first inapproximability result for majority thresholds.

Small thresholds. Another interesting special case is when all thresholds are small, say, constant [32]. Dreyer [12] showed that if the threshold of every vertex is k for any $k \geq 3$, the Target Set Selection problem is NP-hard. However, it leaves as an open problem [32] for the case of $k = 2$. Note that, the problem can be solved trivially for the case of $k = 1$: Target an arbitrary vertex in each connected component.

In this paper, we solve the problem by proving it is NP-hard as well when $k = 2$. Indeed, we show a much stronger and surprising result: Approximating the Target Set Selection problem in the threshold 2 setting is as hard as approximating the problem in the general setting, even for constant degree graphs. Our result implies that, to study upper or lower bounds on the approximation ratio of the Target Set Selection problem, it suffices to consider the threshold 2 setting.

Our proof is based on our hardness result for majority thresholds and the simulation of monotone boolean circuits. Specifically, observe that the state of each vertex can be viewed as a boolean variable and written as a majority boolean function of the states of its neighbors. By the results built on sorting networks [20], e.g., the seminal work by Ajtai, Komlós and Szemerédi [2], a majority boolean function can be simulated by a polynomial size monotone circuit. Thus, the influence propagation in a social network can be viewed as running a polynomial size monotone circuit on each vertex locally. Given this idea, we construct gadgets composed of vertices with thresholds of at most 2 to simulate each AND and OR gate in the circuit.

Unanimous thresholds. The most influence-resistant setting are unanimous thresholds; i.e., the threshold of each node is equal to its degree. For example, in an ideal virus-resistant network, a vertex is infected only if all of its neighbors are infected. Understanding this particular case can help us to construct robust virus-resistant network structures. We show that the problem with unanimous thresholds is equivalent to vertex cover, which implies that it admits a 2-approximation algorithm.

Tree structure. One simple, but important, class of social networks are trees. For example, in query incentive networks [19, 4], the graph is modeled by a tree structure. When the underlying social network is a tree, Dreyer [12] gave a polynomial-time algorithm to compute the optimal target set if all thresholds are the same. We generalize the result to arbitrary thresholds, using dynamic programming. In a recent paper, Ben-Zwi et al. [6] generalize our result to show a polynomial-time algorithm when the graph has constant treewidth.

2. A polylogarithmic hardness result. In general, the TARGET SET SELECTION problem is hard to approximate within a near polynomial ratio.

THEOREM 2.1. *Unless* $NP \subseteq DTIME(n^{polylog(n)})$, *the* TARGET SET SELECTION *problem cannot be approximated within the ratio of* $O(2^{\log^{1-\epsilon} n})$ *for any fixed constant* $\epsilon > 0$.

We will prove the theorem by a reduction from the minimum representative (MINREP) problem [21, 22]. We begin by describing the MINREP problem and then show the reduction.

2.1. The MINREP problem. Given a bipartite graph $G = (A, B; E)$, where A and B are disjoint sets of vertices, there are explicit partitions of A and B into equal-sized subsets. That is, $A = \bigcup_{i=1}^{\alpha} A_i$ and $B = \bigcup_{j=1}^{\beta} B_j$, where all sets A_i have the same size $|A|/\alpha$ and all sets B_j have the same size $|B|/\beta$. The partition of G induces a supergraph H as follows: There are $\alpha + \beta$ supervertices, corresponding to each A_i and B_j, respectively, and there is a superedge between A_i and B_j if there exist some $a \in A_i$ and $b \in B_j$ that are adjacent in G. Figure 1 gives an example of the MINREP problem, where each set A_i has three vertices and B_j has four vertices.

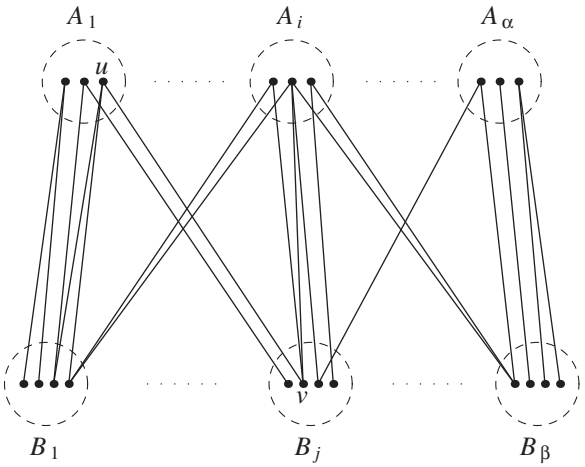

FIG. 1. *An instance of the* MINREP *problem.*

We say a pair (a, b) *covers* a superedge (A_i, B_j) if $a \in A_i$ and $b \in B_j$ are adjacent in G. For example, in Figure 1, (u, v) covers superedge (A_1, B_j). We say $S \subseteq A_i \cup B_j$ *covers* a superedge (A_i, B_j) if there exist $a, b \in S$ such that (a, b) covers (A_i, B_j).

The goal of the MINREP problem is to select the minimum number of representatives from $A \cup B$ such that all superedges are covered. That is, we wish to find subsets $A' \subseteq A$ and $B' \subseteq B$ with the minimum total size $|A'| + |B'|$ such that, for every superedge (A_i, B_j), there exist representatives $a \in A' \cap A_i$ and $b \in B' \cap B_j$ that are adjacent in G.

The MinRep problem is closely related to the label cover problem that models two-prover one-round proof systems, and the following result follows directly from the parallel repetition theorem [29].

Theorem 2.2. *For any fixed $\epsilon > 0$, the* MinRep *problem cannot be approximated within the ratio of $O(2^{\log^{1-\epsilon} n})$ unless $NP \subseteq DTIME(n^{polylog(n)})$.*

2.2. Proof of Theorem 2.1. For any given MinRep instance $G = (A, B; E)$, let M be the number of superedges and N be the total input size. In the reduction, we will use a number of the following basic gadgets Γ_ℓ (see Figure 2), where $t(v_i) = 1$ for $i = 1, \ldots, \ell$.

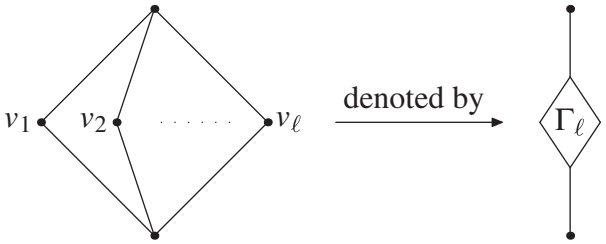

Fig. 2. *The basic gadget Γ_ℓ.*

We next describe the construction of graph G' for the Target Set Selection problem. Basically, G' consists of four different groups of vertices V_1, V_2, V_3, and V_4, where the vertices between different groups are connected by the basic gadgets described above.

- $V_1 = \{a \mid a \in A\} \cup \{b \mid b \in B\}$ and each vertex has threshold N^2.
- $V_2 = \{u_{a,b} \mid (a, b) \in E\}$ and each vertex has threshold $2N^5$. Vertex $u_{a,b} \in V_2$ is connected to each of $a, b \in V_1$ by a basic gadget Γ_{N^5}.
- $V_3 = \{v_{i,j} \mid A_i, B_j$ are connected by a superedge$\}$ and each vertex has threshold N^4. Vertex $u_{a,b} \in V_2$ is connected to $v_{i,j} \in V_3$ by a basic gadget Γ_{N^4} if $a \in A_i$ and $b \in B_j$.
- $V_4 = \{w_1, \ldots, w_N\}$ and each vertex has threshold $M \cdot N^2$. Each vertex $v_{i,j} \in V_3$ is connected to each $w_k \in V_4$ by a basic gadget Γ_{N^2}, and each vertex $a, b \in V_1$ is connected to each $w_k \in V_4$ by a basic gadget Γ_N.

Figure 3 displays the structure of the construction.

We claim that the size of the optimal MinRep solution of G is within a factor of two of the size of the optimal Target Set Selection solution of G'. Thus, any approximation algorithm for Target Set Selection essentially gives the same approximation ratio (up to at most a constant factor) for MinRep.

Assume that $A' \subseteq A$ and $B' \subseteq B$ are an optimal MinRep solution of G. We claim that $A' \cup B' \subseteq V_1$ is a Target Set Selection solution of G'. Since $A' \cup B'$ is a MinRep solution, for any superedge (A_i, B_j), there exist $a \in A' \cap A_i$ and $b \in B' \cap B_j$ such that $(a, b) \in E$. Thus, vertex $u_{a,b} \in V_2$ can be activated, which implies that $v_{i,j} \in V_3$ can be activated as well. This is true for all superedges, and thus all vertices in V_3 are active, which implies that all vertices in V_4 are active. Therefore, all vertices in V_1 can be activated, which induces all vertices in G' to be active at the end.

On the other hand, let S be an optimal Target Set Selection solution of G'. First of all, it is safe to assume that no middle vertices $v_1, \ldots, v_\ell$ from any basic gadget Γ_ℓ are in S. Second, we can assume without loss of generality that no vertices in V_3 are in S. This is because if there is a vertex $v_{i,j} \in S \cap V_3$, then we can remove $v_{i,j}$

←This indicates that V_3 is replaceable.

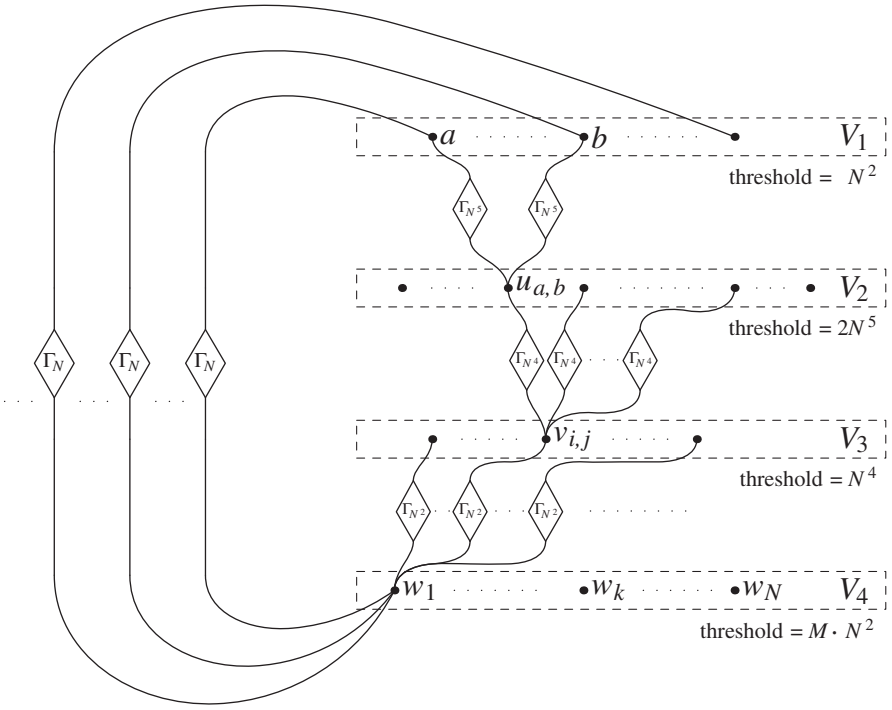

FIG. 3. *The structure of graph G'.*

from S and include $u_{a,b} \in V_2$ to S, where $a \in A_i$ and $b \in B_j$, which gives a solution of the same size. Finally, if there is a vertex $u_{a,b} \in S \cap V_2$, we can remove $u_{a,b}$ from S and include $a, b \in V_1$ to S. By doing this, the size of S is increased by at most a factor of two. Now S contains only vertices from V_1 and V_4, i.e., $S \subseteq V_1 \cup V_4$. According to our construction, those vertices in $S \cap V_4$ cannot affect any other vertices until all vertices in V_4 are active. Therefore, the only direction for influence to flow in G' is through the channel $V_1 \to V_2 \to V_3 \to V_4$. However, to activate any vertex $w \in V_4 \setminus S$, all vertices in V_3 have to be activated. This implies that $S \cap V_1$ is a MINREP solution of G.

By Theorem 2.2, we have the same hardness of approximation result for the TARGET SET SELECTION problem, which completes the proof of Theorem 2.1. □

2.3. Extensions. We observe that Theorem 2.1 continues to hold for a few extensions:

- The optimal solution influences each vertex in a constant number of rounds. This follows directly from the above construction, i.e., Figure 3.
- Instead of ensuring that all vertices in the network are active, only a fixed fraction of vertices need to be activated. This can be done by the following simple reduction: For the given graph $G = (V, E)$, let $n = |V|$. We construct a new graph G' as follows: Replace each edge in E by a basic gadget Γ_n, and define the new threshold of each $v \in V$ in G' to be $t'(v) = n \cdot t(v)$. It is easy to see that G and G' have the same optimal solution. In graph G', by adding many dummy vertices (with thresholds being equal to their degrees) and connecting to all original vertices in V, it can be seen that to activate a fixed fraction of vertices, all vertices in V have to be activated.

3. Majority thresholds. In this section, we consider *majority thresholds*; i.e., a vertex becomes active if at least half of its neighbors are active. Formally, for each $v \in V$, $t(v) = \lceil \frac{d(v)}{2} \rceil$.[1]

THEOREM 3.1. *Assume that the* TARGET SET SELECTION *problem with arbitrary thresholds cannot be approximated within the ratio of $f(n)$ for some polynomial-time computable function $f(n)$. Then the problem with majority thresholds cannot be approximated within the ratio of $O(f(n))$.*

Proof. For any graph $G = (V, E)$ with arbitrary thresholds, we will construct another graph G' with majority thresholds such that the size of the optimal TARGET SET SELECTION solution of G' and G differs by at most 1. The basic idea is, for each $v \in V$ with $t(v) \neq \lceil \frac{d(v)}{2} \rceil$, to add some dummy vertices incident to v (and change the threshold of v, if necessary) such that the threshold of v in the new setting is majority.

To be specific, for any $v \in V$ with threshold $t(v)$ and degree $d(v)$, there are the following two cases:

Case 1 $\left(t(v) > \lceil \frac{d(v)}{2} \rceil\right)$. For this case, we add $2t(v) - d(v)$ isolated dummy vertices incident to v and with threshold 1 each.

Case 2 $\left(t(v) < \lceil \frac{d(v)}{2} \rceil\right)$. For this case, we add $d(v) - 2t(v)$ isolated dummy vertices incident to v and with threshold 1 each. Furthermore, let the new threshold of v be $d(v) - t(v)$.

In addition, we add a "super" vertex u and connect u to all dummy vertices added in the above Case 2. Let the threshold of u be its majority. Denote the resulting graph by G'. Note that the thresholds of all vertices in G' are majority.

We claim that the size difference between the optimal TARGET SET SELECTION solution of G' and G is at most 1. For any given solution S of G, it can be seen that $S \cup \{u\}$ is a TARGET SET SELECTION solution of G'. On the other hand, consider any optimal solution S' of G'. Assume without loss of generality that no dummy vertices added in Cases 1 and 2 are in S'. If $u \in S'$, then $S' \setminus \{u\}$ is a solution of G. Otherwise, S' itself is a solution of G.

Therefore, the size of the optimal solution of G' and G differs by at most 1. Thus, essentially they share the same hardness of approximation ratio. ∎

Given the hardness result of Theorem 2.1 and the above result, the following conclusion follows immediately.

COROLLARY 3.1. *The* TARGET SET SELECTION *problem with majority thresholds cannot be approximated within the ratio of $O(2^{\log^{1-\epsilon} n})$ for any fixed constant $\epsilon > 0$, unless $NP \subseteq DTIME(n^{polylog(n)})$.*

4. Small thresholds. When the thresholds are small, say, all equal to one (i.e., $t(v) = 1$ for any $v \in V$), the problem can be solved trivially: For each connected component of the graph, we target a vertex arbitrarily. Surprisingly, the problem becomes even hard to approximate when we extend the thresholds to be at most 2. We will first show a simple NP-hardness proof and then prove the hardness of approximation result.

4.1. NP-hardness.

THEOREM 4.1. *The* TARGET SET SELECTION *problem is NP-hard when the thresholds are at most* 2*, even for bounded bipartite graphs.*

Proof. We reduce from the following restricted version of 3SAT [33]: Given a formula $\phi(x_1, \ldots, x_n) = C_1 \wedge \cdots \wedge C_m$, where x_i and $\bar{x}_i$ appear at most three times and

←It means x_i and its complement together appear at most three times.

[1] In our discussions, we assume that ties are broken in favor of "prefer-to-change." All our results continue to hold for other tie-breaking rules.

each clause C_j contains at most three literals, we are asked if there is an assignment that satisfies ϕ. Note that we can assume without loss of generality that both x_i and $\bar{x}_i$ appear at least once in ϕ, because otherwise we can simply decide the assignment of x_i and remove those already satisfied clauses.

We construct a ~~bipartite~~ graph $G = (V, E)$ as follows. For each $i = 1, \ldots, n$, given the occurrences of x_i and $\bar{x}_i$, we construct the following gadget in Figure 4 (use the left one if x_i occurs twice and $\bar{x}_i$ occurs once, the middle one if x_i occurs once and $\bar{x}_i$ occurs twice, and the right one if both x_i and $\bar{x}_i$ occur once), where the numbers on the vertices are their thresholds.

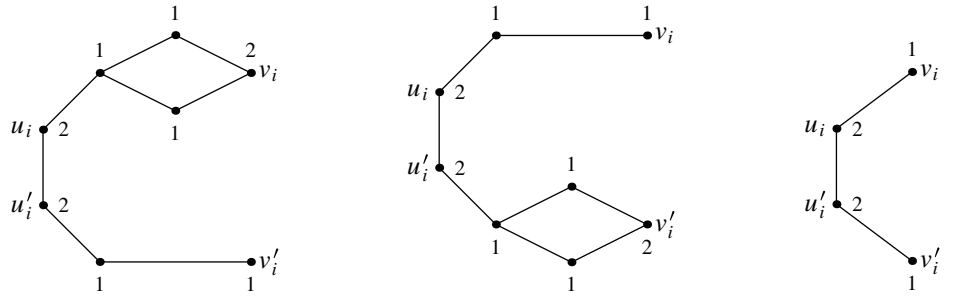

FIG. 4. *Gadget for each variable x_i.*

For each clause C_j, we add a vertex w_j with threshold 1. Let $W = \{w_1, \ldots, w_m\}$. If $x_i \in C_j$, connect v_i and w_j. If $\bar{x}_i \in C_j$, connect v'_i and w_j. Observe that, in our construction, the threshold of each v_i (resp., v'_i) is equal to the number of edges between v_i (resp., v'_i) and W. We claim that ϕ is satisfiable if and only if the optimal target set of G has size n.

Assume that ϕ is satisfiable. Define the target set

$$S = \{u_i \mid 1 \le i \le n : x_i = true\} \cup \{u'_i \mid 1 \le i \le n : x_i = false\}. \tag{1}$$

Note that $|S| = n$. If $u_i \in S$, it is easy to see that v_i (and all other vertices between u_i and v_i on the gadget of x_i) becomes active. Similarly, if $u'_i \in S$, v'_i (and all other vertices between u'_i and v'_i on the gadget of x_i) becomes active. Since ϕ is satisfiable, by the construction of G, all vertices in W become active, which in turn implies that all v_i and v'_i become active (if they were inactive). Hence, all vertices in G become active finally.

On the other hand, let T be the set of targeted vertices. Suppose that $|T| \le n$. Observe that, for any $i = 1, \ldots, n$, the threshold of u_i and u'_i is 2, which implies that at least one of them is in T. Hence, $|T| = n$ and T does not contain any vertex other than u_i, u'_i. Consider any vertex w_j. To make w_j active, at least one of its neighbors should be active before w_j. Assume without loss of generality that $(v_i, w_j) \in E$ and v_i is active before w_j. To make v_i active, given the threshold and our construction, u_i must be activated before v_i, which could happen only if $u_i \in T$. In other words, if all vertices in W are active by targeting T, the assignment corresponding to T (i.e., $x_i = true$ if $u_i \in T$, and $x_i = false$ if $u'_i \in T$) satisfies ϕ.

Hence, ϕ is satisfiable if and only if the optimal target set of G has size n, which implies that the TARGET SET SELECTION problem is NP-hard. ∎

4.2. Hardness of approximation. Beyond NP-hardness, the problem is even hard to approximate within a ratio of $O(2^{\log^{1-\epsilon} n})$ when $t(v) = 2$ (or $t(v) \leq 2$). In particular, we will show the following result.

THEOREM 4.2. *Assume that the* TARGET SET SELECTION *problem with arbitrary thresholds cannot be approximated within the ratio of $f(n)$ for some polynomial-time computable function $f(n)$. Then the problem cannot be approximated within the ratio of $O(f(n))$ when all thresholds are at most 2.*

We have the following corollary, which answers a complexity question proposed in [12, 32].

COROLLARY 4.1. *Given any graph where $t(v) = 2$ (or $t(v) \leq 2$) for any vertex v, the* TARGET SET SELECTION *problem cannot be approximated within the ratio of $O(2^{\log^{1-\epsilon} n})$ for any fixed constant $\epsilon > 0$, unless $NP \subseteq DTIME(n^{polylog(n)})$.*

Proof. The case where $t(v) \leq 2$ follows directed from Theorems 2.1 and 4.2. It remains to consider the case where $t(v) = 2$ for any vertex v.

We will prove the $t(v) = 2$ case by a reduction from the $t(v) \leq 2$ case. Given a graph $G = (V, E)$ where $t(v) \leq 2$ for any $v \in V$, we add a "super" vertex u and connect u to each $v \in V$ with $t(v) = 1$. Let the resulting graph be G' and all thresholds in G' be 2. We claim that the size difference between the optimal solution of G' and G is at most 1. Then the claim follows from the hardness of the $t(v) \leq 2$ case.

For any TARGET SET SELECTION solution S of G, it is easy to see that $S \cup \{u\}$ is a solution of G'. On the other hand, assume that S' is an optimal solution of G'. If $u \in S'$, then $S' \setminus \{u\}$ is a solution of G. If $u \notin S'$, then S' itself is a solution of G. This completes the proof. ∎

Next we will prove Theorem 4.2. Our reduction is built on (1) the hardness result of majority thresholds given by Theorem 3.1, (2) the monotone boolean circuits of computing majority functions, and (3) the gadgets of simulating majority boolean circuit. We begin by describing how to do the simulation and then show the reduction.

4.2.1. Simulating majority boolean circuit. A boolean function $f : \{0,1\}^n \to \{0,1\}$ is called a *majority* function if

$$f(x_1, \ldots, x_n) = \begin{cases} 1 & \text{if } x_1 + \cdots + x_n \geq \lceil \frac{n}{2} \rceil, \\ 0 & \text{otherwise.} \end{cases}$$

We will use the following result by Ajtai, Komlós, and Szemerédi [2].

THEOREM 4.3 (see [2]). *There exist polynomial-size monotone circuits to compute majority boolean functions, where monotone means only AND and OR gates are in the circuit.*

The basic idea is to construct small gadgets composed of vertices of thresholds of at most 2 to simulate AND and OR gates in the circuit. For a circuit that computes a majority function $f(x_1, \ldots, x_n)$, let us denote the gates in the circuit by u_i. Denote the final output gate by u_0 and input gates by $u_1, \ldots, u_n$ (corresponding to $x_1, \ldots, x_n$, respectively). Thus, each gate u_i, $i > n$, is the output of an AND or an OR gate with other u_j's as inputs. The graph we construct has a vertex w_i with threshold 2 for each u_i and a gadget for each AND and OR gate in the circuit. We consider AND and OR gates, respectively, as follows.

For any AND gate, we construct the following gadget (see Figure 5), where the value on each vertex is its threshold.

It can be seen that for the "bottom-to-top" channel (i.e., $w_j, w_k \to w_i$), w_i is active (corresponding to the output u_i being 1) only if both w_j and w_k are active (corresponding to the inputs u_j and u_k being 1). In addition, if only one of w_j and

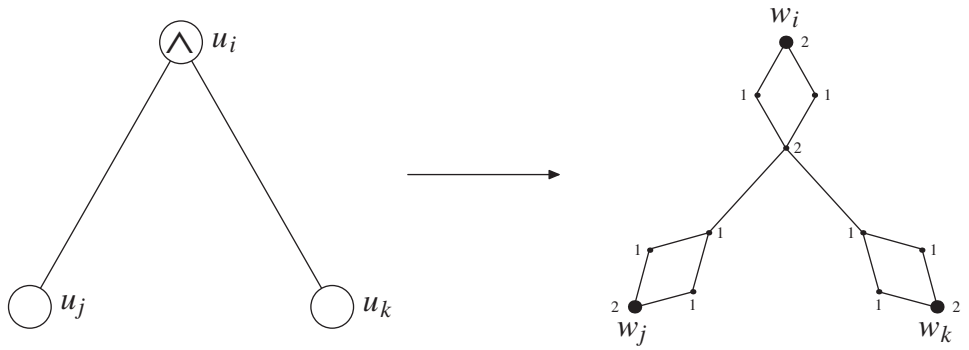

FIG. 5. *Gadget for AND gate.*

w_k is active (say, w_j), the center vertex of threshold 2 ensures that neither w_i nor w_k can get activated due to the influence from w_j. On the other hand, considering the channel from "top-to-bottom," once w_i is active, both w_j and w_k become active as well.

For any OR gate with output u_i, we construct the following gadget (see Figure 6), where the value on each vertex is its threshold. Recall that w_0 is the vertex corresponding to the final output u_0 of the circuit.

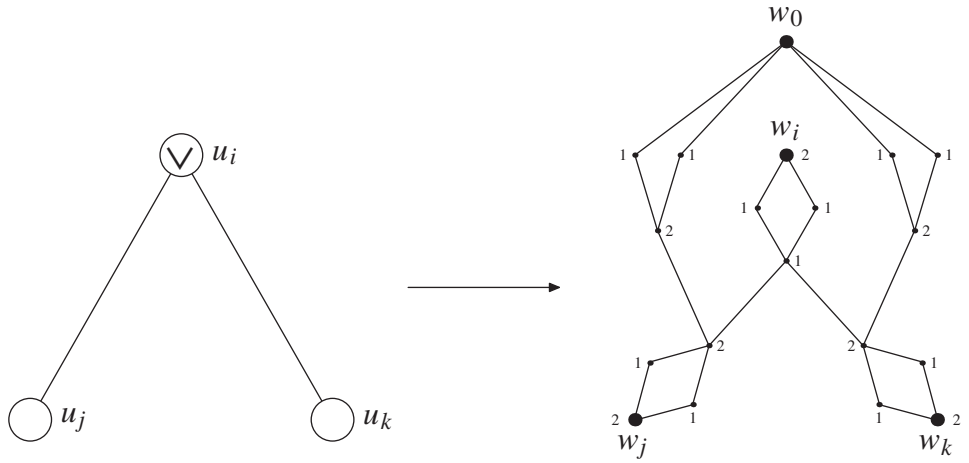

FIG. 6. *Gadget for OR gate.*

As in AND case, for the "bottom-to-top" channel (i.e., $w_j, w_k \to w_i$), w_i is active (corresponding to the output u_i being 1) if at least one of w_j and w_k is active (corresponding to at least one of the inputs u_j and u_k being 1). In addition, if only one of w_j and w_k is active (say, w_j), even though w_i can be activated, neither w_0 nor w_k can get activated due to the influence from w_i, w_j. On the other hand, for the channel from "top-to-bottom," when w_i is active, w_j and w_k can be activated once w_0 is active as well.

Denote the resulting graph by G_n. From the argument above, we know that G_n has the following properties:

- If w_0 is active, then all vertices in G_n can become active. This implies that if there is a vertex targeted in G_n, we can assume without loss of generality

that the vertex is w_0. We call w_0 the *output vertex* of G_n and denote it by $r(G_n)$.

- If at least half of the vertices in $\{w_1, \ldots, w_n\}$ are active, then w_0 can be activated. This holds because the circuit correctly computes the majority function and our simulation of each gate. In the following discussions, we denote $w_1, \ldots, w_n$ by the *input vertices* of G_n.
- If a vertex w_i is inactive, then all its neighbors are still inactive. This is important in that the propagation in G_n can only be through the channel of "top-to-bottom" or "bottom-to-top." In particular, this implies that if less than half of the input vertices are active, then the remaining inactive input vertices cannot be activated due to the influence in G_n.

4.2.2. Proof of Theorem 4.2. We are now ready to prove Theorem 4.2. Given a graph $G = (V, E)$, where each $v \in V$ has majority threshold, we will construct a graph $G' = (V', E')$, where $t(v) \leq 2$ for any $v \in V'$, such that the size of the optimal Target Set Selection solution of G is equal to that of G'. The claim then follows from Theorem 3.1.

For each $v \in V$, let $d(v)$ be the degree of v in G. We use a copy G^v of graph $G_{d(v)}$ to replace v and all its incident edges, where $G_{d(v)}$ is the graph constructed above to simulate majority function $f(\cdot)$ with $d(v)$ input variables. Each input vertex in G^v corresponds to an edge incident to v in E. For any edge $(u, v) \in E$, let w_i and w'_j be the two input vertices in G^u and G^v corresponding to (u, v), respectively. We connect w_i and w'_j by a basic gadget Γ_2 (i.e., as Figure 7 shows, we add two vertices a_1 and a_2 with threshold 1 each and connect $(a_1, w_i), (a_1, w'_j), (a_2, w_i)$, and (a_2, w'_j)). Denote the resulting graph by G'.

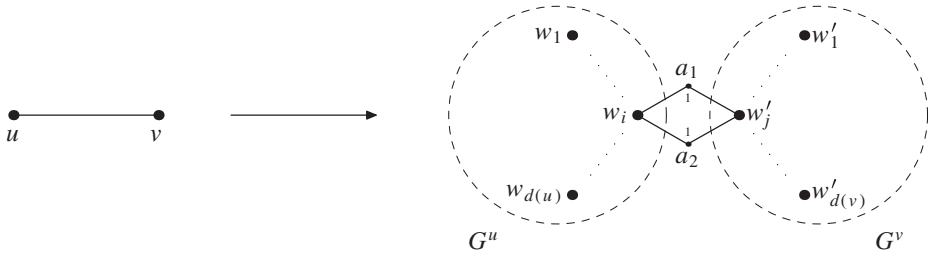

Fig. 7. *Gadget for edge* (u, v).

For any Target Set Selection solution S of G, let $S' = \{r(G^v) \mid v \in S\}$; i.e., S' contains the output vertex of each G^v for $v \in S$. For any $v \in S$, we consider how its neighbor u could be influenced by v. In graph G, we know that u can be influenced from v directly by one unit. In graph G', according to the properties of G^v established in the last subsection, we know that all vertices in G^v are active. Thus, as u and v are connected by an edge, one of the input vertices of G^u becomes active. Since the threshold of u in G is majority, u becomes active when at least half of its neighbors are active, which is equivalent to at least half of the input vertices of G^u being active (and thus, all vertices in G^u are active). Hence, the influence propagation in G' follows exactly the same pattern as that in G, and hence S' is a Target Set Selection solution of G'.

On the other hand, let S' be an optimal Target Set Selection solution of G'. According to the properties of simulation graph discussed above, we can assume without loss of generality that only output vertices are in S'. Define

$S = \{v \in V \mid r(G^v) \in S'\}$. By a similar argument as above, it follows that S is a TARGET SET SELECTION solution of G.

Therefore, the size of the optimal TARGET SET SELECTION solution of G is equal to that of G', which completes the proof of Theorem 4.2. □

4.2.3. Constant degree graphs. In this subsection, we will show that Theorem 4.2, as well as Corollary 4.1, continues to hold for constant degree graphs.

By the construction of the Ajtai, Komlós, and Szemerédi sorting network [2] and the reduction of proving Theorem 4.2, the only vertex that has nonconstant degree in each G^v gadget is its output vertex $r(G^v)$. This is because, for each OR gadget, we add four edges incident to $r(G^v)$ as Figure 6 shows (i.e., w_0). To fix this, we make a few "identical" copies of $r(G^v)$ such that each copy has constant degree. More precisely, we replace $r(G^v)$ and all its incident edges with the gadget shown in Figure 8, where each r_i, $i = 1, \ldots, k$, is an "identical" copy of $r(G^v)$ and $k - 1$ is the number of OR gates in G^v.

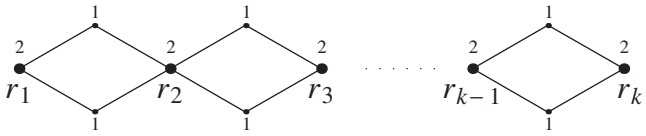

FIG. 8. *Gadget for the output vertex.*

In particular, r_1 corresponds to the original $r(G^v)$ and each r_i, $i = 2, \ldots, k$, corresponds to an OR gate and is used to add the four edges as Figure 6 shows. If one of the r_i's becomes active, all others are active as well, and thus the resulting constant degree graph is equivalent to the original graph.

Note that in the proof of Corollary 4.1 we add a "super" vertex u and connect u to all vertices with threshold one. In general, the degree of u can be arbitrary. Instead of adding one such "super" vertex, we add the following gadget (see Figure 9), where each v_i connects to a vertex in the original graph with threshold one (assume that there are k such vertices in total).

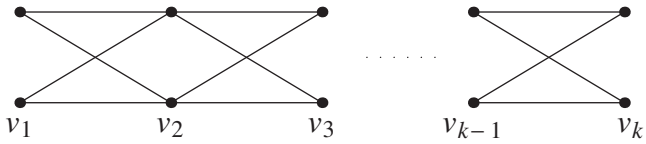

FIG. 9. *Gadget for the "super" vertex.*

The threshold of each vertex in the resulting graph is 2. It can be seen that the size of the optimal solution of the original and resulting graph differs by at most 2. Hence, Corollary 4.1 still holds for constant degree graphs.

5. Unanimous thresholds. The most influence-resistant setting is the *unanimous* thresholds setting. That is, the threshold of each vertex is equal to its degree, i.e., $t(v) = d(v)$ for each $v \in V$. For this case, we have the following hardness result.

THEOREM 5.1. *If all thresholds in a graph are unanimous, the* TARGET SET SELECTION *problem is equivalent to vertex cover.*

Proof. In vertex cover, given a graph $G = (V, E)$, we want to find a subset $V' \subseteq V$ such that for each $(u, v) \in E$, $V' \cap \{u, v\} \neq \emptyset$ and $|V'|$ is as small as possible. We

consider the same graph for the TARGET SET SELECTION problem and claim that G has a vertex cover of size at most k if and only if TARGET SET SELECTION has a solution of size at most k.

For any vertex cover solution V' of G, let the target set of G be V'. Then, for each $v \notin V'$, all edges incident to v are covered by the corresponding vertices in V', which implies that v can be activated. Thus, by targeting V', all vertices are active at the end.

On the other hand, for any TARGET SET SELECTION solution V', we argue that V' is a vertex cover as well. For any edge (u, v), if neither u nor v is in V', both u and v cannot be activated, since their threshold is equal to their degree, which is a contradiction. ∎

As an implication of the above result, the TARGET SET SELECTION problem admits a 2-approximation algorithm [34] and is NP-hard to approximate better than 1.36 [10].

6. Tree structure. When the underlying graph $G = (V, E)$ is a tree, the TARGET SET SELECTION problem can be solved in polynomial-time. The basic observation is that for any leaf $v \in V$, $t(v)$ is equal to 1. Thus, at most one of v and its parent u will be targeted in the optimal solution. Hence, we can assume without loss of generality that v is not targeted; otherwise, we can target u instead of v and get a solution of the same size. The algorithm is as follows.

ALG-TREE

```
1. Let t'(v) = t(v), for v ∈ V
2. Let x(v) = 0, for each leaf v ∈ V
3. While there is x(v) not defined yet
4.  for any vertex u where all x(·)'s of its children have been defined
5.       let w be u's parent
6.       if t'(u) ≥ 2
7.          let x(u) = 1
8.          let t'(w) ← t'(w) − 1
9.       else
10.         let x(u) = 0
11.         if t'(u) < 0
12.            let t'(w) ← t'(w) − 1
13. Output the target set {v ∈ V | x(v) = 1}
```

THEOREM 6.1. ALG-TREE *computes an optimal solution for the* TARGET SET SELECTION *problem when the underlying graph* $G = (V, E)$ *is a tree.*

Proof. Let OPT be an optimal target set solution and S be the set generated by ALG-TREE. For each $u \in V$, let $T(u)$ be the subtree rooted at u. Define

$$S(u) = S \cap T(u)$$

and

$$OPT(u) = OPT \cap T(u).$$

For any vertex u, consider the influence process in $T(u)$. Given a target set (either $S(u)$ or $OPT(u)$), u may or may not become active before its parent. Let $A(u)$ and $B(u)$ be the subset of children of u that become active before u when we target $S(u)$ and $OPT(u)$, respectively.

We inductively prove the following claims:

- $|S(u)| \leq |OPT(u)|$ when $u \in S$ or $u \notin S \cup OPT$.
- $|S(u)| < |OPT(u)|$ when $u \notin S$ and $u \in OPT$.
- If $|S(u)| = |OPT(u)|$, $u \notin S \cup OPT$, and $|B(u)| \geq t(u)$, then $|A(u)| \geq t(u)$.

Note that the first two claims imply that $|S| \leq |OPT|$, and the theorem follows immediately.

The claims trivially hold when u is a leaf of G (note that $u \notin S$ in this case). Consider any internal vertex u in G. There are the following cases to consider.

Case 1 ($u \in S$ and $u \in OPT$). By induction, we know that for each child v of u, $|S(v)| \leq |OPT(v)|$. Thus, $|S(u)| \leq |OPT(u)|$.

Case 2 ($u \notin S$ and $u \in OPT$). Similarly, by induction, we have

$$\begin{aligned} |S(u)| &= \sum_{v \in T(u):\ (u,v) \in E} |S(v)| \\ &\leq \sum_{v \in T(u):\ (u,v) \in E} |OPT(v)| \\ &= |OPT(u)| - 1 \\ &< |OPT(u)|. \end{aligned}$$

Case 3 ($u \notin S$ and $u \notin OPT$). Similarly, by induction, we know that $|S(u)| \leq |OPT(u)|$.

Assume that $|S(u)| = |OPT(u)|$ and $|B(u)| \geq t(u)$. Observe that $|S(v)| \leq |OPT(v)|$ for any child v of u. Thus, essentially $|S(v)| = |OPT(v)|$. By induction, we know that if $v \in OPT$, then $v \in S$. Furthermore, if $B(u)$ contains a vertex $v \notin OPT$, i.e., $v \in B(u) \setminus OPT$, we know that v is active in $T(v)$ given target set $OPT(v)$, i.e., $|B(v)| \geq t(v)$. Hence, by induction, $|A(v)| \geq t(v)$, which implies that v can be activated before u so that $v \in A(u)$. Therefore, $|A(u)| \geq |B(u)| \geq t(u)$.

Case 4 ($u \in S$ and $u \notin OPT$). By induction, it suffices to find a child v of u such that $|S(v)| < |OPT(v)|$. Since $u \in S$, according to step 6 of Alg-Tree, we know that $t(u) \geq 2 + |A(u)|$. Note that u has at most one parent and u is active in OPT; we have $t(u) \leq 1 + |B(u)|$. Therefore, $1 + |A(u)| \leq |B(u)|$. Hence, we know that there is $v \in B(u) \setminus A(u)$. If $v \in OPT$ and $v \notin S$, we are done because of the second inductive claim. Otherwise, $v \notin OPT \cup S$ such that $|B(v)| \geq t(v)$. The third claim says that if $|S(v)|$ is not strictly smaller than $|OPT(v)|$, $|A(v)| \geq t(v)$ and thus v is in $A(u)$. Hence, for both cases, $|S(v)| < |OPT(v)|$.

Therefore, the claim holds, which completes the proof. ∎

Acknowledgments. I would like to thank Paul Beame, Venkatesan Guruswami, Anna Karlin, and David Kempe for helpful discussions. I am especially grateful to Kunal Talwar and Evimaria Terzi; we were working on a different but related model (for which we got a $\log^{2-\epsilon} n$-hardness result), and those ideas inspired me to complete this work. I also thank Anna, Kunal, and Evimaria for their many suggestions on improving the paper.

REFERENCES

[1] A. Aazami and M. D. Stilp, *Approximation algorithms and hardness for domination with propagation*, in Proceedings of the 10th APPROX, Princeton, NJ, Princeton University, 2007, pp. 1–15.

[2] M. Ajtai, J. Komlós, and E. Szemerédi, *An $O(n \log n)$ sorting network (sorting in $c \log n$ parallel steps)*, Combinatorica, 3, (1983), pp. 1–19.

[3] D. Angluin, J. Aspnes, and L. Reyzin, *Optimally learning social networks with activations and suppressions*, in Proceedings of the 19th International Conference on Algorithmic Learning Theory (ALT), Budapest, University of Szeged, 2008, pp. 272–286.
[4] E. Arcaute, A. Kirsch, R. Kumar, D. Liben-Nowell, and S. Vassilvitskii, *On threshold behavior in query incentive networks*, in Proceedings of ACM Electronic Conference, San Diego, 2007, pp. 66–74.
[5] J. Aspnes, K. Chang, and A. Yampolskiy, *Inoculation strategies for victims of viruses and the sum-of-squares partition problem*, in Proceedings of ACM Symposium on Discrete Algorithms 2005, Vancouver, 2005, pp. 43–52.
[6] O. Ben-Zwi, D. Hermelin, D. Lokshtanov, and I. Newman, *An exact almost optimal algorithm for target set selection in social networks*, in Proceedings of ACM Electronic Conference 2009, Stanford, CA, 2009.
[7] E. Berger, *Dynamic monopolies of constant size*, J. Combin. Theory Ser. B, 83 (2001), pp. 191–200.
[8] N. Berger, C. Borgs, J. T. Chayes, and A. Saberi, *On the spread of viruses on the internet*, in Proceedings of ACM Symposium on Discrete Algorithms 2005, Vancouver, 2005, pp. 301–310.
[9] Z. Dezso and A.-L. Barabasi, *Halting viruses in scale-free networks*, Phys. Rev. E (3), 65 (2002), article 055103.
[10] I. Dinur and S. Safra, *The importance of being biased*, in Proceedings fo the 33rd ACM Symposium Theory of computing, Heraklion, Greece, 2002, pp. 33–42.
[11] P. Domingos and M. Richardson, *Mining the network value of customers*, in Proceedings of ACM SIGKDD 2001, San Francisco, 2001, pp. 57–66.
[12] P. A. Dreyer, *Applications and Variations of Domination in Graphs*, Ph.D. thesis, Rutgers University, Piscataway, NJ, 2000.
[13] A. Ganesh, L. Massouli, and D. Towsley, *The effect of network topology on the spread of epidemics*, in Proceedings of INFOCOM 2005, Miami, IEEE, 2005, pp. 1455–1466.
[14] N. Immorlica, J. M. Kleinberg, M. Mahdian, and T. Wexler, *The role of compatibility in the diffusion of technologies through social networks*, in Proceedings of ACM Electronic Conference, San Diego, 2007, pp. 75–83.
[15] M. Kearns, L. Ortiz, *Algorithms for interdependent security games*, in Proceedings of NIPS 2003, Vancouver, Canada, Springer, 2003, pp. 288–297.
[16] D. Kempe, J. Kleinberg, and É. Tardos, *Maximizing the spread of influence through a social network*, in Proceedings of the 9th ACM SIGKDD International Conference, Washington, D.C., 2003, pp. 137–146.
[17] D. Kempe, J. Kleinberg, and É. Tardos, *Influential nodes in a diffusion model for social networks*, in Proceedings of the 32nd International Colloquium on Automata, Languages and Programming (ICALP), Lisbon, Portugal, CITI, 2005, pp. 1127–1138.
[18] D. Kempe and M. Mahdian, *Cascade model for externalities in sponsored search*, in Proceedings of the Workshop on Internet and Network Economics (WINE 2008), Shanghai, Springer, 2008.
[19] J. M. Kleinberg and P. Raghavan, *Query incentive networks*, in Proceedings of IEEE Foundations of Computer Science (FOCS) 2005, Pittsburgh, PA, 2005, pp. 132–141.
[20] D. E. Knuth, *The Art of Computer Programming, Sorting and Searching*, Vol. 3, 3rd ed., Addison–Wesley, Reading, MA, 1997.
[21] G. Kortsarz, *On the hardness of approximating spanners*, Algorithmica, 30 (2001), pp. 432–450.
[22] G. Kortsarz, R. Krauthgamer, and J. R. Lee, *Hardness of approximation of vertex-connectivity network design problems*, SIAM J. Comput., 33 (2004), pp. 704–720.
[23] N. Linial, D. Peleg, Y. Rabinovich, and M. Saks, *Sphere packing and local majorities in graphs*, in Proceedings of the 2nd Israel Symposium on Theory of Computing Systems, Natanya, Israel, IEEE, ISTCS 1993, pp. 141–149.
[24] S. Morris, *Contagion*, Rev. Econom. Stud., 67 (2000), pp. 57–78.
[25] E. Mossel and S. Roch, *On the submodularity of influence in social networks*, in Proceedings of the 39th ACM Symposium on Theory of Computing, San Diego, 2007, pp. 128–134.
[26] R. Pastor-Satorras and A. Vespignani, *Epidemics and immunization in scale-free networks*, in Handbook of Graphs and Networks: From the Genome to the Internet, S. Bornholdt and H. G. Schuster, eds., Wiley-VCH, Weinheim, Germany, 2003, pp. 111–130.
[27] D. Peleg, *Local majority voting, small coalitions and controlling monopolies in graphs: A review*, in Proceedings of the 3rd Colloquium on Structural Information & Communication Complexity, Sienna, Italy, 1996, pp. 170–179.
[28] D. Peleg, *Size bounds for dynamic monopolies*, Discrete Appl. Math., 86 (1998) pp. 263–273.

[29] R. Raz, *A parallel repetition theorem*, SIAM J. Comput., 27 (1998), pp. 763–803.
[30] M. Richardson and P. Domingos, *Mining knowledge-sharing sites for viral marketing*, in Proceedings of the 8th ACM SIGKDD, Edmonton, Canada, 2002, pp. 61–70.
[31] F. S. Roberts, *Challenges for discrete mathematics and theoretical computer science in the defense against bioterrorism*, in Bioterrorism: Mathematical Modeling Applications in Homeland Security, Frontiers Appl. Mathem. 28, H. T. Banks and C. Castillo-Chavez, eds., SIAM, Philadelphia, 2003, pp. 1–34.
[32] F. S. Roberts, *Graph-theoretical problems arising from defending against bioterrorism and controlling the spread of fires*, in Proceedings of the DIMACS/DIMATIA/Renyi Combinatorial Challenges Conference, Piscataway, NJ, 2006.
[33] C. A. Tovey, *A simplified satisfiability problem*, Discrete Appl. Math., 8 (1984), pp. 85–89.
[34] V. V. Vazirani, *Approximation Algorithms*, Springer-Verlag, New York, 2001.
[35] D. J. Watts and S. H. Strogatz, *Collective dynamics of 'small-world' networks*, Nature, 393 (1998), pp. 440–442.