How to Win Friends and Influence People, Truthfully: Influence Maximization Mechanisms for Social Networks

Yaron Singer
Computer Science Division
University of California at Berkeley, CA, 94720 USA
yaron@cs.berkeley.edu

ABSTRACT

Throughout the past decade there has been extensive research on algorithmic and data mining techniques for solving the problem of influence maximization in social networks: if one can incentivize a subset of individuals to become early adopters of a new technology, which subset should be selected so that the word-of-mouth effect in the social network is maximized? Despite the progress in modeling and techniques, the incomplete information aspect of the problem has been largely overlooked. While data can often provide the network structure and influence patterns may be observable, the inherent cost individuals have to become early adopters is difficult to extract.

In this paper we introduce mechanisms that elicit individuals' costs while providing desirable approximation guarantees in some of the most well-studied models of social network influence. We follow the mechanism design framework which advocates for allocation and payment schemes that incentivize individuals to report their true information. We also performed experiments using the Mechanical Turk platform and social network data to provide evidence of the framework's effectiveness in practice.

Categories and Subject Descriptors

H.2.8 [**Database Management**]: Database applications – Data mining

General Terms

Algorithms, Economics, Experimentation, Theory

1. INTRODUCTION

The emergence of online social networks in the past decade has given birth to the study of information propagation in social networks. Motivated by marketing applications, a central problem that received considerable attention in this domain is the *influence maximization problem*, where the goal is to identify a small subset of individuals in a social network that can serve as early adopters of a new technology and trigger a large word-of-mouth cascade in the network.

The main focus in the study of the influence maximization problem has been on descriptive influence models and efficient optimization machinery. Since the problem is known to be NP-hard, there has been substantial focus on characterization of influence models and their approximation guarantees (e.g. [19, 18, 26, 5]). To enable better predictions, a different line of work studies techniques for inferring influence models from observable data and analyzing large scale information cascades (e.g. [8, 30, 11, 23, 14, 13, 25, 4]).

These developments in algorithmic and data mining techniques focus on identifying influencers and predicting their potential influence in the social network. To execute viral marketing campaigns, however, an important question remains: *How do we influence the influencers?*

For concreteness, consider *Ticketbuster*, a (fictitious) online retailer that sells concert tickets and wishes to promote a concert through word-of-mouth in a social network (e.g. Google+). To do this, Ticketbuster is willing to provide discounts to individuals who purchase tickets and are willing to announce their purchase to their friends in the social network. Given that it is limited by the total amount of discounts it can offer and that individuals respond differently to incentives, Ticketbuster needs to decide who should be offered a discount for announcing their purchase to their friends, and importantly, how much they should be offered.

The above example is an illustration of the kind of problems we wish to address in this work: how to design algorithms for viral marketing campaigns that provide appropriate incentives for individuals to become early adopters[1]. The challenge lies in the fact that individuals attribute different costs for becoming early adopters, and our algorithms must somehow be designed to account for this.

The Influence Maximization Model

In the influence maximization model, as formalized by Kempe, Kleinberg and Tardos in [19] we are given a social network graph and an *influence function* that takes a subset of nodes of initial adopters and returns the expected number of nodes influenced in the graph. Given some parameter k the goal is to find k initial adopters that will maximize the expected number of nodes influenced. This elegant model captures the decisions often made in viral marketing campaigns where a set of consumers is chosen to receive incentives for being

---

[1]The term *early adopters* here is used to generally describe the subset of individuals we are seeking to incentivize in order to create the word-of-mouth effect.

early adopters of a technology (e.g. receive a product at a discounted price), so that their experiences would be advertised by word-of-mouth through the social network.

The underlying assumption in the influence maximization model is that each node in the graph has some inherent *cost* for being an initial adopter[2] and that the word-of-mouth process described by the influence function does not depend on additional incentives. In simple cases one may assume that costs are uniform (e.g. the cost for being an initial adopter may simply be the cost of buying the technology at an advertised market price), or that the incentives can be rewarding enough to cover any reasonable cost that may be associated with being an initial adopter. In general however, as in the case of our example, costs may be subjective and must be considered in the input to the problem.

Challenges of Inferring Incentives

Despite the growing availability of personalized data and the emergence of sophisticated machine learning techniques, inferring personal costs is a difficult task: individuals often have subjective costs that heavily depend on various aspects that are difficult to observe like personal history or opinions.

A reasonable approach may be to estimate an optimal reward value and use this estimate as an offer to individuals. The problem however is that this approach leads to poor results both in theory and in practice. In Appendix A we show that even when given access to individuals' costs, no mechanism that offers uniform rewards can approximate the optimal solution within a factor better than $\Omega(\frac{\log n}{\log \log n})$. In Section 4 we show that such mechanisms perform poorly in practice as well. This implies that even under the strong assumption that costs are known, the influence maximization techniques that use the optimal reward perform poorly.

Finally, we can consider algorithms that pay individuals their declared costs. This naïve approach, however, is susceptible to manipulation: individuals may declare higher costs to increase their reward.

A Mechanism Design Approach

To address the above challenges, we introduce machinery from mechanism design theory. The theory is based on the idea of designing mechanisms[3] that are *incentive compatible* (aka *truthful*): mechanisms for which it is provably in each agent's best interest to report her true private information, regardless of the other agents' reports. Thus, by assuming that agents are rational[4], an incentive compatible mechanism extracts the agents' true information and uses it to identify and reward the subset of initial adopters. Our goal is therefore to design *incentive compatible* mechanisms that compare favorably against the optimal influence maximization solution.

1.1 Our Results

In this paper we introduce incentive compatible mechanisms for the influence maximization problem in some of the most well-studied models of influence in social networks.

---

[2]Otherwise no incentives would be necessary to become an initial adopter and there would be no restriction on the number of nodes that can be selected.

[3]A mechanism is simply an algorithm that decides who is allocated and how much they are paid.

[4]In our case, rationality simply means that an agent wishes to maximize her profit.

In particular we show constant-factor approximation mechanisms [5] for the Coverage, Linear Threshold, Independent Cascades, the Voter and general submodular influence models. To evaluate their practical performance, we provide experimental evidence of the mechanisms' effectiveness on real data sets. In our experiments we used network data that has almost a million nodes and over 72 million edges, together with a representative cost distribution that was obtained by running a simulated campaign on Amazon's Mechanical Turk platform [1].

1.2 Related Work

The problem of maximizing influence was first posed by Domingos and Richardson in [8, 30] and has been extensively studied since. In their seminal work [19], Kempe, Kleinberg and Tardos characterize some of the main models of influence in terms of their computational approximability. While they show that the optimization problem in many of the models is NP-hard, their work identifies important classes of influence models in which the objective function is submodular – a condition that allows providing desirable approximation guarantees. The approximation frontiers of the influence problem in more general models have been further explored in [5, 26].

Incentives have been considered in mechanisms for finding information within a social network in [21] using payments at Nash-equilibrium. In [17] a game theoretic influence model is suggested, where incentives for adoption in the influence process are considered. Network externalities were considered for pricing mechanisms in [16] and revenue maximizing auctions in [15]. There has been recent work on incentives in social network motivated by marketing applications [3, 9], though not in the influence maximization framework, nor are the mechanisms incentive compatible.

The theoretical work in this paper follows recent work in [31], which establishes a theory for designing incentive compatible mechanisms under a budget constraint on payments. The main result shows that for any submodular optimization function, there exists a randomized incentive compatible mechanism with a constant factor approximation guarantee. In this paper we rely on the characterization of payments from this line of work to develop improved mechanisms for the standard models in social networks.

2. THE MODEL

In our model there is a social network that is represented by a graph $G = (\mathcal{N}, \mathcal{E})$ and an influence function $f : 2^{\mathcal{N}} \to \mathcal{R}_{\geq 0}$. Given a subset $S \subseteq \mathcal{N}$ the value $f(S)$ represents the expected number of nodes influenced in the social graph. Each node in the graph represents an agent a_i that has a *private* cost $c_i \in \mathcal{R}_{\geq 0}$ of being an initial adopter, and this cost is not known to the mechanism designer. We assume that agents are rational and strategic in a game theoretic sense. That is, given a choice of payments an agent will choose to maximize her profit[6] and may misreport her cost to do so. In particular, an agent can become an initial adopter only if the payment it receives is greater or equal to her cost.

---

[5]Our comparisons follow the strictest benchmarks one can consider: We compare our mechanisms against the computationally unbounded optimal solution that has full information about the agents' true costs.

[6]The profit of an agent with cost c that receives payment p is simply $p - c$.

Given a budget $B \geq 0$ our goal is to maximize the influence by paying a subset of agents to be the initial adopters of the technology. A solution to the problem is a subset of agents and a payment vector that describes the payments to each agent in the subset. A mechanism is simply a pair $M = (A, p)$ where A is an algorithm and p is a payment rule. We sometimes refer to the algorithm of the mechanism as the *allocation rule*. The mechanism takes the social network, the influence function, the budget and the agents' reported costs as input [7].

Approximation

Since the problem is known to be NP-hard, as in the case of standard algorithms, we restrict our mechanisms to run in polynomial time, and seek solutions that are provably close as possible to optimal. For $\alpha \geq 1$ we say that a mechanism $M = (A, p)$ is α-approximate if A is an α approximation algorithm: A returns a solution S such that $f(S^*) \leq \alpha f(S)$, where S^* is the optimal solution in the full-information case (i.e. the same benchmark that is used for an algorithm when the costs are known and the payment equals the costs).

Incentive Compatible Mechanisms

An incentive compatible mechanism is a mechanism in which the algorithm and payment rule are designed in such a way so that each agent maximizes her profit by revealing her true cost. Rather than proving that a mechanism is incentive compatible directly we will rely on a well known characterization from the literature that will reduce incentive compatibility to a simple condition on the algorithm.

In our setting, where each agent's private information is her cost we can use a characterization for incentive compatible mechanisms by Myerson [28]. According to this characterization, a mechanism is incentive compatible in our setting if and only if it is *monotone* and uses threshold payments. Informally, this means that as long as other agents do not change their bids, an agent that is selected should remain selected when declaring a lower cost and that her payment is the maximal cost she can declare and remain selected. Formally, given an algorithm A and an agent a_i let $A(c_i, c_{-i})$ be the subset selected by the algorithm when a_i declares cost c_i and the rest of the agents' costs are denoted by the vector c_{-i}. The monotonicity rule requires that if $a_i \in A(c_i, c_{-i})$ then $a_i \in A(c'_i, c_{-i})$, for any $c'_i < c_i$. The threshold payment that ensures incentive compatibility is $p_i = \inf \{b_i : a_i \notin A(b_i, c_{-i})\}$.

To prove that a mechanism is incentive compatible we will therefore show that the algorithm is monotone. An important requirement is that the mechanism's payments do not exceed the budget. Fortunately, the main component of our mechanisms will be the proportional share allocation rule, for which a result in [31] gives a characterization of its incentive compatible payments that can be immediately translated to a condition on the algorithm. We will therefore use the characterization as a black box and discuss the restrictions the payments impose in the appropriate sections.

---

[7] This is known as a *direct revelation mechanism:* agents report their cost to the mechanisms. On Ticketbuster, for example, once completing her purchase, an individual would report how much she would like to be rewarded for sharing her purchase on Google+. Note that we assume that the social network and budget are known to the mechanism.

3. MECHANISMS FOR MAX INFLUENCE

In this section we present several mechanisms for some of the well-studied models of influence in social networks. To simplify the exposition, we begin with the Coverage influence model and later show how the mechanism can be adapted to accommodate other influence models.

The Coverage Model

In the Coverage model, each agent a_i in the network is associated with a subset of nodes R_i and the influence in this model is $f(S) = |\cup_{i \in S} R_i|$. The optimization problem is the well known NP-hard Max-k-Cover problem, and it can be easily verified that this influence model is submodular. It is a fundamental model that serves as a basis for many of the well-studied submodular influence models.

Intuitively, a good mechanism will select agents that yield high coverage while the incentive compatible payments will not exceed the given budget. The main component of our mechanism achieves this by greedily selecting agents based on their marginal contributions and enforcing an appropriate stopping condition. The greedy selection uses the weighted marginal contribution sorting: an ordering in which at each stage the agent selected is the one that has the maximal weighted contribution given the previously selected agents. That is, under this ordering, the jth agent is:

$$a \in \mathrm{argmax}_{a_i \in \mathcal{N}} \frac{f(S_{j-1} \cup \{a_i\}) - f(S_{j-1})}{c_i} \qquad (1)$$

where S_{j-1} denotes the set of $j - 1$ agents that have been previously selected by this rule. This is the generalized version of the sorting rule used in the celebrated greedy algorithm for submodular function maximization [29], and it has beed used to obtain constant factor approximation guarantees in [20, 22, 24].

To enforce a stopping condition with desirable payment guarantees, after sorting the agents we will include them into the solution as long as the ratio between their cost and the budget is smaller than half of the ratio between their marginal contribution and the value of the subset already selected. That is, we consider agent a_i that meets the condition:

$$c_i \leq \frac{B}{2} \cdot \left(\frac{f(S_i) - f(S_{i-1})}{f(S_i)} \right). \qquad (2)$$

This allocation rule is a variation of *proportional share* rule which is the basis for mechanisms under a budget [31, 6, 7, 32]. Observe that if k agents are selected according to this rule and each selected agent is paid according to their proportional contribution, the total payments do not exceed the budget:

$$\sum_{i=1}^{k} B \cdot \left(\frac{f(S_i) - f(S_{i-1})}{f(S_i)} \right) \qquad (3)$$

$$\leq \frac{B}{f(S_k)} \cdot \sum_{i=1}^{k} \left(f(S_i) - f(S_{i-1}) \right) \qquad (4)$$

$$= \frac{B}{f(S_k)} \cdot f(S_k) = B. \qquad (5)$$

The desirable property of this allocation rule is that for

any submodular function, it can be proved that the incentive compatible payments do not exceed the agents' proportional contributions. We will further discuss payments after describing the entire mechanism below. Importantly, this allocation rule provides desirable guarantees in terms of approximation as well. We use OPT to denote the optimal solution and denote the agent with maximal value by a^*. The main idea behind the proof is similar to that in [31] and compares the proportional allocation rule against the standard greedy algorithm which is known to have a good approximation guarantee.

LEMMA 3.1. *Let S_k be the set that is selected by the algorithm that sorts agents according to their weighted marginal contribution as in (1) and selects the maximal number of agents that respect the proportional allocation rule as in (2). Then, $OPT \leq \left(\frac{5e}{e-1}\right)\max\{f(S_k), f(a^*)\}$.*

PROOF. Let the agents be sorted as describes above, and let ℓ be the maximal index s.t. $\sum_{i=1}^{\ell} c_i \leq B$. For sake of the analysis, consider adding a new agent, a' which does not share any friends with the agents in $\mathcal{N}$, declares cost $B - \sum_{i=1}^{\ell} c_i$ and $f(a') = \left(\frac{B - \sum_{i=1}^{\ell} c_i}{c_{\ell+1}}\right) \cdot \left(f(S_{\ell+1}) - f(S_\ell)\right)$. Observe that the weighted marginal contribution of this agent is identical to that of agent $a_{\ell+1}$ and that the solution $f(S_\ell \cup \{a'\})$ is feasible. Obviously, the optimal solution over all agents in $\mathcal{N}' = \mathcal{N} \cup \{a\}$ is an upper bound on the optimal solution over all agents in $\mathcal{N}$. Due to the decreasing marginal utilities property, we are guaranteed that the first ℓ agents selected in both $\mathcal{N}$ and $\mathcal{N}'$ are identical. We can therefore analyze our solution over the set of agents $\mathcal{N}'$. For notational convenience in the following analysis we will denote agent a' as $a_{\ell+1}$.

In [20] and its generalization [22], it is shown that for any submodular function we have that $f(S_{\ell+1}) \geq (1 - \frac{1}{e})OPT$. We will use $S_{\ell+1}$ as a benchmark. Since $S_{\ell+1}$ is feasible:

$$\sum_{i=k+1}^{\ell+1} \left(\frac{c_i}{f(S_i) - f(S_{i-1})}\right)\left(f(S_i) - f(S_{i-1})\right)$$

$$\leq \sum_{i=1}^{\ell+1} \left(\frac{c_i}{f(S_i) - f(S_{i-1})}\right)\left(f(S_i) - f(S_{i-1})\right)$$

$$= \sum_{i=1}^{\ell+1} c_i = B.$$

Since agents are sorted according to their weighted marginal contributions, for every $i \in \{k+1, \ldots, \ell+1\}$ we have that:

$$\frac{f(S_i) - f(S_{i-1})}{c_i} \leq \frac{f(S_{k+1}) - f(S_k)}{c_{k+1}}. \quad (6)$$

Putting the above inequalities together we get:

$$\frac{c_{k+1}}{f(S_{k+1}) - f(S_k)}\left(\sum_{i=k+1}^{\ell+1} f(S_i) - f(S_{i-1})\right)$$
$$= \left(\frac{c_{k+1}}{f(S_{k+1}) - f(S_k)}\right) \cdot \left(f(S_{\ell+1}) - f(S_{k+1})\right) \leq B.$$

Note that this implies that $2f(S_{k+1}) > f(S_{\ell+1}) - f(S_k)$ as otherwise $c_{k+1} \leq \frac{B}{2}\left(\frac{\cdot(f(S_{k+1}) - f(S_k))}{f(S_{k+1})}\right)$ which contradicts the maximality of k. Thus, together with submodularity this implies that:

$$f(S_{\ell+1}) = f(S_{\ell+1}) - f(S_k) + f(S_k)$$
$$< 2f(S_{k+1}) + f(S_k)$$
$$\leq 2f(a_{k+1}) + 3f(S_k).$$

In conclusion, we get:

$$OPT \leq \left(\frac{e}{e-1}\right)f(S_{\ell+1}) \leq \left(\frac{5e}{e-1}\right)\max\{f(S_k), f(a^*)\}. \quad \square$$

The above lemma suggests that the approximation guarantee of the proportional share allocation rule depends on the selected subset S_k being larger than a^*. If it were not for incentive constraints, one could simply compare $f(S_k)$ and $f(a^*)$ and by selecting the solution that yields the higher value, guarantee a constant-factor approximation. In our case however, one can show that this process would violate incentive compatibility. This happens since there are cases where by declaring a lower cost, an agent changes the allocation of the mechanism which may reduce the value of the proportional share solution. That is, letting A denote the proportional share allocation, there may be cases where $f(A(c_i, c_{i-1})) > f(a^*)$ but $f(A(c'_i, c_{i-1})) < f(a^*)$ when $c'_i < c_i$. To address this, we develop the following methodology. We will compare between a^* and a solution to a nonlinear relaxation program that doesn't suffer from this property while the value of its solution is provably "close" to that of the proportional share mechanism. Observe that the Coverage optimization problem can be reformulated as the following integer program:

$$\max \quad \sum_{j=1}^{m} z_j \qquad j \in U \qquad (7)$$
$$\sum_{i \in C_j} x_i \geq z_j, \quad j \in U, \qquad (8)$$
$$\sum_{i=1}^{n} c_i x_i \leq B, \qquad (9)$$
$$0 \leq x_i, z_j \leq 1 \quad i \in \mathcal{N}, j \in \mathcal{U}, \qquad (10)$$
$$x_i \in \{0, 1\} \quad i \in \mathcal{N} \qquad (11)$$

where z_j are variables representing the agents we wish to cover, x_i are the variables representing the agents we can select, and C_j denotes the subset of agents that cover an agent $a_j \in \mathcal{N}$. In our case $z_i = x_i$ and C_j is the set of friends of agent a_j. For a relaxation we will use:

$$\max \quad \sum_{j=1}^{m} \min\left\{1, \sum_{i \in C_j} x_i\right\} \quad j \in U \qquad (12)$$
$$\sum_{i \in C_j} x_i \geq z_j, \qquad j \in U, \qquad (13)$$
$$\sum_{i=1}^{n} c_i x_i \leq \frac{B}{2}, \qquad (14)$$
$$0 \leq x_i, z_j \leq 1 \qquad i \in \mathcal{N}, j \in \mathcal{U}, \qquad (15)$$
$$x_i \in [0, 1] \qquad i \in \mathcal{N} \qquad (16)$$

The optimal solution to this relaxation, denoted x^* is computable in polynomial time and is a relaxation of the integer program in the sense that for all $x \in \{0, 1\}$ it is equivalent to the integer program. In our mechanism we will compare a^* and this relaxation to circumvent the problem with incentive compatibility. We denote this relaxation as L, and use $L(x)$ to denote a solution that respects the constraints above where $x \in [0, 1]^n$. We will show that the solution of $L(x)$ is "close" to $f(S_k)$, which guarantees that comparing a^* to $L(x)$ rather than $f(S_k)$ does not hurt the solution by more than a constant factor. We will use $L(x^*)$ to denote

the value of the optimal solution to the relaxation over the subset $\mathcal{N}_- = \mathcal{N} \setminus \{a^*\}$. After describing the mechanism we show how this is used to maintain incentive compatibility and the approximation guarantee.

3.1 Influence Maximization in Coverage

We can now formally describe the mechanism:

An Approximation Mechanism for Coverage

Require: $< G = (\mathcal{N}, \mathcal{E}), c, B >$

    $a^* \longleftarrow \operatorname{argmax}_{j \in [n]} f(a_j)$;

    $a_i \longleftarrow \operatorname{argmax}_{j \in [n]} \frac{f(a_j)}{c_j}$;

    while $c_i \leq \frac{B}{2} \cdot \left(\frac{f(S_i) - f(S_{i-1})}{f(S_i)} \right)$ **do:**

        $a_i \longleftarrow \operatorname{argmax}_{j \in [n]} \frac{f(S_i \cup \{a_j\}) - f(S_i)}{c_j}$;

        $S_i \longleftarrow S_i \cup \{a_i\}$;

    if $L(x^*) > \left(\frac{6e^2}{(e-1)^2} \right) f(a^*)$

        $S \longleftarrow S_i$

    else do: $S \longleftarrow \{a^*\}$

Output: S

The mechanism above uses the proportional share algorithm with budget $\frac{B}{2}$, and returns the set selected by this algorithm if the value of the relaxation is greater than a factor of $f(a^*)$. The payment characterization in [31] shows that the incentive compatible payments are bounded by a constant factor of the proportional contributions, which implies that when using this constant fraction of the budget in the stopping condition the payments do not violate the budget constraint. The subsequent improved analysis in [6] shows that this constant factor is no more than 2, and thus by using $\frac{B}{2}$ in the stopping condition we are guaranteed that the payments do not exceed the budget. We defer the proof of the to the full version of the paper.

LEMMA 3.2. *The mechanism is monotone.*

PROOF. To see that the mechanism is indeed monotone assume for purpose of contradiction that for a given cost profile c there is an agent $a_i \in S$ with declared cost c_i, that is not allocated when declaring $c_i' \leq c_i$ while the rest of the agents declare the same cost as in c. Let $\alpha = \frac{6e^2}{(e-1)^2}$. If $a_i \neq a^*$ then it has been selected by the proportional share allocation rule and $L(x^*) > \alpha f(a^*)$. By decreasing her cost the agent will move ahead in the sorting to place $j \leq i$, and due to the decreasing marginal contribution property her marginal contribution will only increase. Thus, $f(S_{j-1} \cup \{a_i\}) - f(S_{j-1}) \geq f(S_{i-1} \cup \{a_i\}) - f(S_{i-1})$ and since f is an increasing function $f(S_{j-1} \cup \{a_i\}) \leq f(S_i)$. This implies:

$$c_i' < c_i \leq \frac{B}{2} \left(\frac{f(S_i) - f(S_{i-1})}{f(S_i)} \right)$$
$$\leq \frac{B}{2} \left(\frac{f(S_{j-1} \cup \{a_i\}) - f(S_{j-1})}{f(S_{j-1}) \cup \{a_i\}} \right)$$

and thus a_i will meet the proportional share allocation rule. Since $L(x^*)$ is the value of the *optimal* fractional solution, it can only increase when the costs are reduced, and therefore its value will be larger than that of a^* in this case as well. In case $a_i = a^*$, it is allocated only if $L(x^*) < \alpha f(a^*)$. Note that her cost does not change the value of $f(a^*)$, nor

that of $L(x^*)$ which computes its solution over $\mathcal{N} \setminus \{a^*\}$. Thus, $\alpha f(a^*)$ remains larger and a^* is allocated. In summary, in all cases we get a contradiction. □

The argument for showing that the mechanism has a constant factor guarantee depends on showing that $L(x^*)$ and the value of the proportional share solution $f(S_k)$ are close. Consider the nonlinear program relaxation:

$$\max \quad \textstyle\sum_{j=1}^m (1 - \Pi_{i \in C_j}(1 - x_i)) \quad j \in U \tag{17}$$
$$\textstyle\sum_{i \in C_j} x_i \geq z_j, \qquad j \in U, \tag{18}$$
$$\textstyle\sum_{i=1}^n c_i x_i \leq \frac{B}{2}, \tag{19}$$
$$0 \leq x_i, z_j \leq 1 \qquad i \in \mathcal{N}, j \in \mathcal{U}, \tag{20}$$
$$x_i \in [0, 1] \qquad i \in \mathcal{N} \tag{21}$$

Let F denote the above relaxation. Note that both $F(x)$ and $L(x)$ identify with our integer program over integral solutions. The following lemma can be shown directly using the pipage rounding technique in [2], and we reserve the proof to the full version of the paper.

LEMMA 3.3. *Let x^* be the optimal solution for L. Then, there exists an integral solution $\bar{x}$ s.t. $L(x^*) \leq (\frac{2e}{e-1})F(\bar{x})$.*

Using the above lemmas, we can now conclude with our main theorem.

THEOREM 3.4. *The mechanism is incentive compatible and guarantees an approximation ratio of $\frac{12e^2}{(e-1)^2} + 1$.*

PROOF. Incentive compatibility is a direct result of the monotonicity and the threshold payments discussed. Recall that the optimal *fractional* solution $L(x^*)$ is computed over $\mathcal{N}_- = \mathcal{N} \setminus \{a^*\}$ with budget $\frac{B}{2}$. Let OPT' be the optimal solution over the set of agents $\mathcal{N}_-$ with budget $\frac{B}{2}$.

First, in a similar way to the proof of Lemma 3.1 we can prove that $OPT' < \frac{3e}{e-1} \max\{f(S_k), f(a^*)\}$. Since F is equivalent to our integer program over integral solutions, then OPT' is an upper bound on any of its integral solutions. Together with Lemma 3.3 these two bounds imply:

$$L(x^*) \leq \left(\frac{2e}{e-1} \right) OPT' < \left(\frac{6e^2}{(e-1)^2} \right) \max\{f(S_k), f(a^*)\}.$$

If $L(x^*) \geq \left(\frac{6e^2}{(e-1)^2} \right) f(a^*)$, the above inequality implies that $f(a^*) < f(S_k)$, and by Lemma 3.1 we have that $OPT \leq (\frac{5e}{e-1})f(S_k)$, and we're done.

Otherwise, if $L(x^*) \leq \left(\frac{6e^2}{(e-1)^2} \right) f(a^*)$, let $L(x^*, \mathcal{N}', B')$ denote the optimal solution of the relaxation L over the set of agents $\mathcal{N}'$ and budget B'. In this case we have:

$$OPT \leq L(x^*, \mathcal{N}, B) \leq L(x^*, \mathcal{N}_-, B) + f(a^*)$$
$$= 2L(x^*, \mathcal{N}_-, \frac{B}{2}) + f(a^*)$$
$$\leq \left(\frac{12e^2}{(e-1)^2} + 1 \right) f(a^*)$$

and we obtain the desired bound. □

3.2 Submodular Influence Models

In the main models studied in the past, influence is described as a step-by-step procedure that is governed by some probabilistic interaction rule. The seminal work of [19] shows that many of the probabilistic influence models considered in the literature are characterized by their submodularity. The two basic diffusion models studied there are the Linear Threshold and Independent Cascades models.

The Linear Threshold Model

In this model, for each neighbor w a node v associates a weight $b_{v,w} \geq 0$ where $\sum_{w \in N(v)} b_{v,w} \leq 1$, and chooses some threshold $\theta_v \in [0,1]$ uniformly at random. The node v is activated at time step t if $\sum_{w \in N_t(v)} b_{v,w} \geq \theta_v$, where $N_t(v)$ denotes the neighbors of v that are active at time step t.

The Independent Cascades Model

Here we assume that each node v is independently influenced by each neighbor w with some probability $p_{v,w}$. When a node w is activated at some time step t it has a single chance to activate each neighbor v with probability $p_{v,w}$.

In each realization of the above models, the influence maximization problem becomes equivalent to the problem in the Coverage model. In the Independent Cascades Model, a realization of the probabilities of each interaction creates a graph that deterministically describes the spread of influence. We can therefore construct a reachability graph that describes for each node v the subset of nodes it influences, and the nodes influenced by a subset S is its coverage in the reachability graph. In the Linear Threshold case [19] show that the model is equivalent to a process in which each node v picks one incoming edge with probability $b_{v,w}$ and does not pick any edges with probability $1 - \sum_w b_{v,w}$ which also reduces to the Coverage model. Therefore, for both models, in any realization we can apply the mechanism described for the Coverage model.

While the Linear Threshold and the Independent Cascades models can be used to describe many influence functions considered in the literature, there are more general submodular influence functions that are not captured by these models. For such models we can obtain a constant factor approximation mechanism by allowing for randomization:[8] Since Lemma 3.1 applies to any submodular function, randomly choosing between the most influential agent and the proportional share allocation, guarantees, in expectation, a constant factor approximation mechanism.

COROLLARY 3.5. *For any submodular influence function there is a randomized incentive compatible constant-factor approximation mechanism. Furthermore, for any realization of the Linear Threshold and Independent Cascades models there is a deterministic constant factor approximation mechanisms that is incentive compatible.*

The Voter Model

In the Voter model, at each time step, each agent is influenced with probability that is proportional to the number of

---

[8]In general submodular functions may require a description that is exponential in the size of the problem. In such cases we can apply our mechanisms efficiently in the *value query* model where it is assumed that any subset can be evaluated by an oracle in polynomial time.

---

friends that were influenced in the previous time step. This influence model has been extensively covered in the past and recently also in the computer science community [27, 10]. More formally, assuming a set of nodes has been activated at step $t = 0$, the probability of v to be influenced at time step $t > 1$ is $p_{v,t} = \frac{|\{u:u \in N_{t-1}(v)\}|}{|N(v)|}$, where $N_{t-1}(v)$ denotes the subset of neighbors of v that are influenced at time step $t - 1$ and $N(v)$ denotes the set of neighbors of v.

An equivalent formulation of the Voter model can be stated in terms of a random walk on the graph that represents the social network: if u is activated at time step 0, the probability of some node v in the graph to be activated by u after t time steps is equivalent to the probability of a random walk on the graph that starts at v and terminates at u after t steps. To see this, observe that if w is activated at step 0 for $t = 1$ the probability of v to be influenced by its neighbor w is $\frac{1}{|N(v)|}$, and that the probability of v to be influenced at a general time step t can be recursively computed.

The random walk process can be computed by associating a transition matrix M_G with the graph. In this matrix $M_G(u,v) = \frac{1}{|N(u)|}$ if v is a neighbor of u and 0 otherwise, and the probability of a random walk that starts at node v and terminates at node u after t steps is $M_G^t(v,u)$. Using this representation we can easily compute the expected number of nodes that terminate at u after t time steps, or equivalently the number of nodes influenced by u: $f_t(u) = \sum_v M_G^t(v,u)$. Importantly, the matrix formulation implies that we can compute the expected influence of a given subset S additively:

$$f_t(S) = \sum_{u \in S} \sum_v M^t(v,u).$$

The linearity allows using variation of the mechanism from Section 3 with a better approximation guarantee.

THEOREM 3.6. *For the Voter Model there is an incentive compatible 4-approximation mechanism.*

PROOF SKETCH. The mechanism is similar to the mechanism for Coverage, except that it uses the entire budget in the stopping condition of the proportional allocation rule, and an integer programming relaxation suitable for this linear problem rather than the relaxation used in the previous mechanism. The proportional contribution of an agent a_i in this case is:

$$\frac{B \cdot f_t(a_i)}{\sum_{j \leq i} f_t(a_j)} \tag{22}$$

and can be efficiently computed by $\sum_v M_G^t(v,u)$, and the total marginal contribution of a subset S is $\sum_{u \in S} \sum_v M^t(v,u)$. Due to linearity, by case analysis we can prove that an agent that declares a cost that is larger than her proportional contribution will not be allocated. This implies that the incentive compatible payments can be bounded by agents' proportional contributions, thus allowing us to use the proportional share allocation rule with the entire budget. The approximation guarantee then follows with an analysis similar to the one in Lemma 3.1 that can be used to show that the optimal solution is bounded by $2 \sum_{i=1}^k f(a_i) + f(a_{k+1}) + f(a^*)$ when k agents selected and a^* is the agent with maximal influence. Similar arguments to those used above can be used to show the mechanism is monotone and incentive compatible. $\square$

4. EXPERIMENTAL EVALUATION

Given the theoretical guarantees discussed above, we would like to show that our mechanisms also work well in practice. While it may be the case that our mechanisms provide theoretical guarantees in the worst case, it is not obvious a priori that they outperform simpler heuristics on representative instances. Furthermore, as we discussed above, one may argue that mechanisms with differential prices like the ones suggested here, are too complex or alternatively that eliciting costs from individuals is an unnecessary procedure.

To examine these issues we performed an experimental evaluation of our mechanisms in different influence models over a large sample of a social network, and compared their performance against other mechanisms. While network data that is representative of real social networks is relatively accessible, the greatest crux in performing this experiment was the lack of knowledge about individuals' costs for making a recommendation in their social network. To address this, we performed an experiment on Amazon's Mechanical Turk platform to obtain a descriptive cost distribution.

4.1 The MTurk Experiment

To obtain a distribution that is reflective of individuals' costs to make recommendations in their social network we ran an experiment for a period of three months on Amazon's Mechanical Turk platform (henceforth MTurk)[1]. The experiment was presented as a competition for receiving bonus money. The workers were explained that a small number of individuals will be selected to advertise a travel agency by posting a message with commercial content in their Facebook page, and that they need to specify how much they want be rewarded and how many friends they have on Facebook. Since a posted message is automatically displayed to an individual's friends in Facebook, the reports on rewards in this experiment represent the reported costs we address in this work. We ran the competition with a small budget to enable cost-effective data collection. Each worker who participated in the competition was paid, and the workers who won the competition received a bonus reward at least as high as their bid. [9]

The experiment ran for a period of three months. The payments for participating in the experiment varied from \$0.05 to \$0.50, and the mean payment was approximately \$0.26. We limited the experiment to workers based in the US with high approval rate (above 97%). The main reason was to increase the likelihood that the worker is an active Facebook user and understands the experiment. Due to individuals' privacy concerns we did not use verification methods provided by Facebook (e.g. Facebook applications) to establish the workers' Facebook identities.

In the Human Intelligence Task (HIT) the competition details were clearly described in two sentences. An image of an example post that advertises a travel agency was provided and workers were asked to give a value of how much they would like to be rewarded for posting the message on Facebook, as well as to provide the number of friends. The example message displayed had clear commercial content, similar to content generated when users share a message of an online retailer on the Facebook platform. We chose ad-

---

[9]Winners were selected by running the mechanism with the influence function $f(S) = \sum_{i \in S} f_i$, where f_i is the reported number of friends of agent a_i. The goal in this experiment was not to test the mechanism, but rather collect bids.
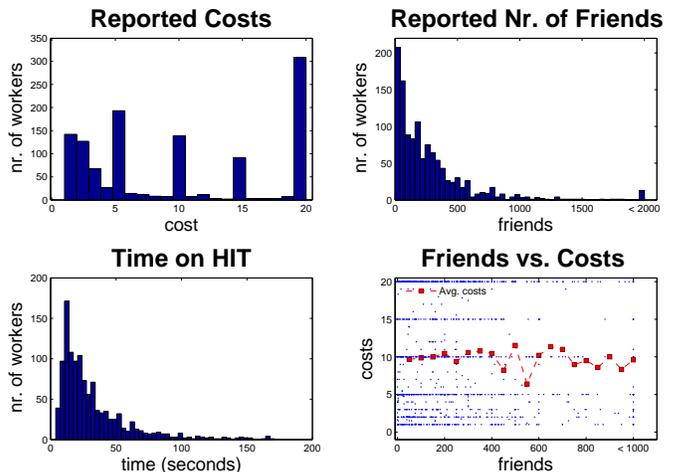

Figure 1: Results from the MTurk experiment.

vertising for a travel agency as it is rather generic, and is not likely to introduce significant gender or age biases. The main reason for requesting the number of friends was to see whether there is a corellation between cost and reported number of friends. We collected over $1,178$ bids from over $1,000$ different MTurk workers. In Figure 1 we plot distributions of reported costs, reported friends, time required to complete the task, as well as a plot of costs vs. reported number of friends.

To avoid biases, we used a single bid from each worker by choosing between bids at random if a user bid more than once. The mean cost that users bid was \$9.77 and the median was \$8. There were 307 workers who bid the maximal allowed bid (\$20), and 364 who bid below \$5. The mean time to complete the HIT was 32.25 seconds.

In our experiment, in order to use the cost distribution generated by MTurk on the sampled Facebook graph, we wanted to verify that there is no correlation between the number of friends and the reported cost. First, we observed that the reported number of friends generated a distribution that is similar to the distribution of friends from crawling Facebook with a BFS search [12]. This leads us to believe that, in general, workers reported an accurate estimate of their number of friends on Facebook. We assume this is due to the fact that workers were told that if they give a false report on the number of friends they will not be eligible for the reward. The mean number of reported friends was 271.3, the median was 185. We then tested to see whether there is any correlation between the number of friends and the reported cost, and found no evidence for such correlation ($corr = 0.04$). We see this as support to the methodology in our data simulation, where we take a cost at random from the data gathered in the MTurk experiment and assign it to a node from the Facebook sampled graph. We plot the costs as a function of friends in Figure 1.

4.2 Computational Experiments

Network Data

To represent the social network we used a graph of $957,459$ nodes and over 72 million edges which was sampled from the

Facebook social graph via a random walk [12]. The average degree of this graph is 95.2 and the median degree is 40.

Influence Models

To compare the mechanisms over different influence models we implemented the Coverage, Independent Cascades, and the Linear Threshold influence models. The Coverage model was implemented by using each agent's friends as its reachable set. In this implementation a node that's selected activates all of its friends, and the influence process does not propagate further. In the Independent Cascades model we ran two separate trials with uniform probabilities of an edge being activated. In one trial the probability of an edge being activated was 1% and in the other trial the probability was 10%. These choices correspond to the distributions used in [19]. In the Linear Threshold model we chose weights based on the degree distribution of nodes in the social graph. For each node v the value d_v was drawn independently at random from an estimated degree distribution of the social graph. For each of its neighbors w we applied the weight $b_{v,w} = d_v^{-1}$. Since we used a subgraph of the larger Facebook social graph, we did not have the degrees of all of the nodes in the graph and therefore used the degree distribution of the subgraph as an estimate.

To have comparable reachability sets, we limited the time steps of the influence process to 5, 10 and 25 steps in the Independent Cascades with probability 10%, 1% and the Linear Threshold model, respectively. Limiting the time steps seems to be reasonable in light of experimental evidence (see e.g. [33], who report that less than 5% of diffusion processes they studied in Twitter are longer than 5 steps).

After applying the influence model on the graph, to simulate the bidding procedure, we randomly selected 5,000 nodes and randomly assigned them costs from the distribution generated using the MTurk data. We ran several iterations with different budgets ranging from $100 to $5000.

Mechanisms

OPS: optimized proportional share mechanism. For the incentive compatible mechanism, we used an optimized version of the mechanisms presented above which allows us to use almost the entire budget in the allocation. Recall that the mechanism presented above uses half the budget, since this way it can ensure the total payments are within budget. Informally, the potential overpayment that imposes the budget reduction is proportional to the effect a selected agent has on the quality of the solution (see the characterization in [31] for more details). Since the network structure, budget and range of bids are known to the mechanism designer, the budget used in the proportional contribution allocation can be optimized accordingly. This optimization step enabled using almost the entire budget in the allocation rule.

Benchmarks

Our main goal in this experimental evaluation is to understand the implication of our lack of knowledge of the agents' costs (we compare the mechanism against the greedy algorithm that has full knowledge of the costs) and whether knowing agents' costs is indeed necessary (we compare against a near-optimal uniform price mechanism that is given access to the costs to establish the optimal reward). In both cases we conducted theoretical comparisons (see Corollary 3.5 and Appendix A) and our goal here is to conduct an empirical
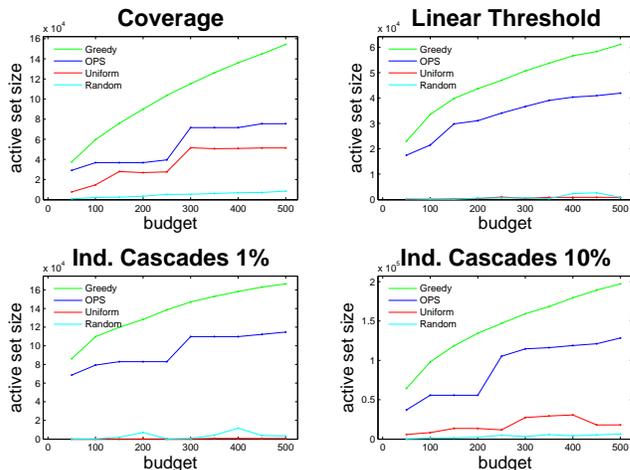

Figure 2: Figures of the different mechanisms with budgets ranging from 50 to 500.

comparison. We further discuss alternative mechanisms in Section 5.

Greedy: algorithmic upper bound. To draw an algorithmic upper bound, we ran the greedy algorithm for maximizing a submodular function under a budget constraint, which is known to work very well both in theory and in practice [20, 24]. Of course, the algorithm is not incentive compatible and assumes full knowledge of the agents' costs.

Uniform-price: cost-oblivious upper-bound. In order to gain insight to whether differential pricing is necessary in practice, we compared against a near-optimal uniform-price mechanism. We describe the mechanism in Appendix A. The mechanism provides an $O(\log n)$ approximation guarantee, which is nearly optimal in the class of uniform price mechanisms (see lower bound in Appendix A). It is important to emphasize that this mechanism represents an upper bound on how well influence maximization algorithms can do assuming uniform rewards are sufficient.

Random: lower bound on algorithmic performance. Finally, we also compared against a random assignment mechanism. The mechanism chooses agents at random and pays them their declared cost. We use this mechanisms to describe how well one can do without any algorithmic machinery. Despite its poor theoretical guarantees, our evaluation shows that in practice this lower bound outperforms the uniform price mechanism in some cases. Note that this mechanism is not incentive compatible since it pays agents their bids. To make the existing mechanism incentive compatible one would need to pay each agent $20, which gives a uniform price mechanism or alternatively offer random rewards. The random assignment mechanism is an upper bound on both of these mechanisms.

4.3 Results

In Figure 2 we plot the results from the experiments with budgets in the range 50–500. In the Coverage model, for a small budget the proportional share mechanism is comparable with the greedy algorithm. As the budget increases one can witness the growing gap. Similar behavior is apparent when running with larger budgets, as further discussed be-
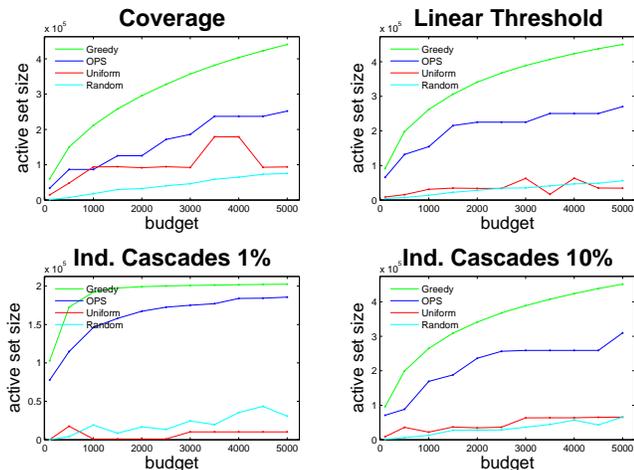
Figure 3: Figures of the different mechanisms with budgets ranging from 100 to 5000.

low. In this model the uniform price mechanism is comparable with the proportional share. Thus, for Coverage it seems that despite its theoretically poor guarantees, the uniform price mechanism seems to be at least a constant fraction from the proportional share and greedy mechanisms. This is mainly due to the fact that the budgets are rather small in comparison to the variance in cost distribution and the marginal value of individuals. Note that in this mechanism in some cases the value in performance decreases while the budget increases. This is due to the fact that at each budget iteration the mechanism uses the given cost distribution to estimate a threshold price, and while the budget may increase, using the value-maximizing bin is not optimal.

In the Independent Cascades model, Figure 2 shows experimental evidence of the poor performance of the uniform price mechanism, in sharp contrast to the Coverage model. This shows that in cases where nodes have larger reachability sets, the uniform price mechanism provides poor guarantees that correspond to its theoretical performance. Furthermore, it seems that this mechanism is not a strict improvement over the random assignment one.

The poor performance of the uniform price mechanism can be further observed in the Independent Cascades models with probabilities set to 10% and the Linear Threshold model in Figure 2 as well as in the experiments with the larger budgets. These results imply that even when one has perfect knowledge about the cost distribution and can extract a near-optimal uniform threshold value, the resulting solution will have poor quality not only in theory but also in practice. We consider this to be a strong argument for the use of differential price mechanisms in these settings.

Observe that the proportional share mechanism exhibits a "step function" behavior: at certain points it levels off, and continues in a steep ascent. This implies that there are budgets which are exhausted well by the mechanism, and exploring these optimal budget points can be crucial when designing viral marketing campaigns.

In Figure 3 we plot the performance of the different mechanisms when the budgets are in the 100–5000 ranges. In the seed graph of 5000 nodes we used, the average degree

was 170.3. Recall that in this model the number of nodes influenced is the number of friends that is covered by the selected subset. In this experiment, for almost all of the different budgets, the performance of the mechanisms had the same ordering: greedy, proportional share, uniform price and the random assignment. As one may expect the random assignment mechanism follows an almost perfect linear increase in performance as the budget increases. While the proportional share mechanism dominates the uniform price method, the gap is less dramatic in this model.

When comparing against the greedy benchmark, it seems that in this model, as the budget increases, the gap between the two mechanisms grows. The analysis in Lemma 3.1 guarantees that if the contribution of the agent with highest influence is not greater than the rest, the gap between the two mechanisms will converge to a factor of two. Interestingly, this gap is almost achieved when the budget is 5000. Since the proportional share solution is a subset of the greedy solution, this effect implies that the "least valuable" half of the greedy solution is as valuable as the "most valuable" half. In the Coverage model on this graph there is little overlap between the nodes' reachability sets, which can explain this phenomenon. This effect implies that the budget is well exhausted. To test this we applied the same experiment in this model with larger budgets and discovered that the ratios begin to shrink at a budget of about 10000. In the other influence models, the gap between the greedy and the proportional share mechanisms is relatively small.

5. DISCUSSION

In this paper we presented mechanisms that incentivize individuals to become early adopters in some of the well-studied models of influence maximization. The negative results presented here for uniform-price mechanisms suggest that differential price mechanisms are necessary.

An alternative to the bidding approach used here would be *posted price* mechanisms, where agents receive take-it-or-leave-it offers, and the mechanism can adjust its offers based on their responses. Such mechanisms are by default incentive compatible. The positive results we presented here can serve as foundations for natural next steps in this direction.

In our experimental evaluation, to compensate for the lack of data on agents' costs, we simulated a competition on Mechanical Turk. Ideally, we would have used data in which the network also includes agent's costs, though we are not aware of any publicly available data sets. As discussed above however, the lack of correlation between costs and number of friends leads us to believe this approach produces a representative data set, at a relatively low cost. We believe similar methodologies can be used in the future.

Lastly, it seems that beyond the scope of influence maximization, mechanism design can be useful in data mining. The emergence of crowdsourcing and user-generated content on the web present challenges that are similar to the ones addressed in this work, and it seems that incentive-based mechanisms can become an important addition to the arsenal of powerful techniques in the data mining realm.

6. ACKNOWLEDGMENTS

7. REFERENCES

[1] Mechanical turk. In *https://www.mturk.com/*.

[2] A. A. Ageev and M. Sviridenko. Pipage rounding: A new method of constructing algorithms with proven performance guarantee. *J. Comb. Optim.*, 8(3):307–328, 2004.

[3] D. Arthur, R. Motwani, A. Sharma, and Y. X. 0002. Pricing strategies for viral marketing on social networks. In *WINE*, pages 101–112, 2009.

[4] E. Bakshy, J. M. Hofman, W. A. Mason, and D. J. Watts. Everyone's an influencer: quantifying influence on twitter. In *WSDM*, 2011.

[5] N. Chen. On the approximability of influence in social networks. In *SODA*, pages 1029–1037, 2008.

[6] N. Chen, N. Gravin, and P. Lu. On the approximability of budget feasible mechanisms. 2011.

[7] S. Dobzinski, C. Papadimitriou, and Y. Singer. Mechanisms for complement-free procurement. In *EC*, 2011.

[8] P. Domingos and M. Richardson. Mining the network value of customers. In *KDD*, pages 57–66, 2001.

[9] P. Dütting, M. Henzinger, and I. Weber. How much is your personal recommendation worth? In *WWW*, pages 1085–1086, 2010.

[10] E. Even-Dar and A. Shapira. A note on maximizing the spread of influence in social networks. *Inf. Process. Lett.*, 111(4):184–187, 2011.

[11] C. Faloutsos, K. S. McCurley, and A. Tomkins. Fast discovery of connection subgraphs. In *KDD*, pages 118–127, 2004.

[12] M. Gjoka, M. Kurant, C. T. Butts, and A. Markopoulou. Walking in Facebook: A Case Study of Unbiased Sampling of OSNs. In *Proceedings of IEEE INFOCOM '10*, San Diego, CA, March 2010.

[13] M. Gomez-Rodriguez, J. Leskovec, and A. Krause. Inferring networks of diffusion and influence. In *KDD*, pages 1019–1028, 2010.

[14] A. Goyal, F. Bonchi, and L. V. S. Lakshmanan. Learning influence probabilities in social networks. In *WSDM*, pages 241–250, 2010.

[15] N. Haghpanah, N. Immorlica, V. S. Mirrokni, and K. Munagala. Optimal auctions with positive network externalities. In *ACM Conference on Electronic Commerce*, pages 11–20, 2011.

[16] J. D. Hartline, V. S. Mirrokni, and M. Sundararajan. Optimal marketing strategies over social networks. In *WWW*, pages 189–198, 2008.

[17] M. O. Jackson and L. Yariv. Diffusion of behavior and equilibrium properties in network games. *American Economic Review*, 2007.

[18] A. R. Karlin, D. Kempe, and T. Tamir. Beyond VCG: Frugality of truthful mechanisms. In *FOCS*, pages 615–626, 2005.

[19] D. Kempe, J. M. Kleinberg, and É. Tardos. Maximizing the spread of influence through a social network. In *KDD*, pages 137–146, 2003.

[20] S. Khuller, A. Moss, and J. S. Naor. The budgeted maximum coverage problem. *Inf. Process. Lett.*, 70(1):39–45, 1999.

[21] J. Kleinberg and P. Raghavan. Query incentive networks. In *Proc. 46th IEEE Symposium on Foundations of Computer Science*, pages 132–141, 2005.

[22] A. Krause and C. Guestrin. A note on the budgeted maximization of submodular functions. In *Technical Report*, pages CMU– CALD– 0 5 – 1 0 3, 2005.

[23] J. Leskovec, L. A. Adamic, and B. A. Huberman. The dynamics of viral marketing. In *ACM Conference on Electronic Commerce*, pages 228–237, 2006.

[24] J. Leskovec, A. Krause, C. Guestrin, C. Faloutsos, J. M. VanBriesen, and N. S. Glance. Cost-effective outbreak detection in networks. In *KDD*, pages 420–429, 2007.

[25] M. Mathioudakis, F. Bonchi, C. Castillo, A. Gionis, and A. Ukkonen. Sparsification of influence networks. In *KDD*, 2011.

[26] E. Mossel and S. Roch. On the submodularity of influence in social networks. In *STOC*, pages 128–134, 2007.

[27] E. Mossel and G. Schoenebeck. Reaching consensus on social networks. In *ICS*, pages 214–229, 2010.

[28] R. Myerson. Optimal auction design. *Mathematics of Operations Research*, 6(1), 1981.

[29] G. L. Nemhauser, L. A. Wolsey, and M. L. Fisher. An analysis of approximations for maximizing submodular set functions ii. *Math. Programming Study 8*, pages 73–87, 1978.

[30] M. Richardson and P. Domingos. Mining knowledge-sharing sites for viral marketing. In *KDD*, pages 61–70, 2002.

[31] Y. Singer. Budget feasible mechanisms. In *FOCS*, pages 765–774, 2010.

[32] Y. Singer and M. Mittal. Mechanisms for online labor markets. In *Workshop on Human Computation*, 2011.

[33] J. Yang and S. Counts. Predicting the speed, scale, and range of information diffusion in twitter. In *ICWSM*, 2010.

APPENDIX

A. BOUNDS ON UNIFORM PRICES

THEOREM A.1. *Even for $f(S) = \sum_{i \in S} f(a_i)$ no uniform-price mechanism can approximate better than $\Omega\left(\frac{\log n}{\log(\log n)}\right)$.*

PROOF. Consider the instance in which $c_i = f(a_i) = 1/i$ for all $i \in [n]$ and we have a budget of $B = \log n$. For any uniform price mechanism that uses the price p, the solution will include $\lfloor \log n/p \rfloor$ items. The value from this will be:

$$\sum_{i=1/p}^{\log n/p} 1/i = \sum_{i=1}^{\log n/p} 1/i - \sum_{i=1}^{1/p} 1/i = O(\log(\log n))$$

Note that the optimal solution which pays each agent her declared cost has value $O(\log n)$. □

A Near-optimal Uniform Price Mechanism. The algorithm divides the agents into bins based on their cost, where an agent i is binned in bin j if her cost is between $B/2^j$ and $B/2^{j+1}$, and uses the greedy algorithm to find the optimal solution in each bin. The bin that returns the highest value is used as the optimal uniform price. It is easy to show that this algorithm is a $O(\log n)$ approximation.