Optimal Auctions with Positive Network Externalities

Nima Haghpanah* Nicole Immorlica* Kamesh Munagala† Vahab Mirrokni‡

Abstract

We consider the problem of designing auctions in social networks for goods that exhibit *single-parameter submodular network externalities* in which a bidder's value for an outcome is a fixed private type times a known submodular function of the allocation of his friends. Externalities pose many issues that are hard to address with traditional techniques; our work shows how to resolve these issues in a specific setting of particular interest. We operate in a Bayesian environment and so assume private values are drawn according to known distributions. We prove that the optimal auction is NP-hard to approximate pointwise, and APX-hard on average. Thus we instead design auctions whose revenue approximates that of the optimal auction. Our main result considers *step-function externalities* in which a bidder's value for an outcome is either zero, or equal to his private type if at least one friend has the good. For these settings, we provide a $\frac{e}{e+1}$-approximation. We also give a 0.25-approximation auction for general single-parameter submodular network externalities, and discuss optimizing over a class of simple pricing strategies.

1 Introduction

Many goods have higher value when used in conjunction with others. A classic example of this phenomenon is the telephone, which clearly has positive value for a consumer only if he or she has people to call. Telephones, and other goods with similar stories, are called *networked goods* and said to exhibit *network effects* or *network externalities*. Modern technology has given birth to a new generation of networked goods. Internet services like email, instant messaging, and online social networks are used primarily to connect with friends and, as such, have strong network externalities. But even more significantly, these services, particularly online social networks, provide platforms upon which developers can generate new applications – applications with very strong networking components. It is now possible to play poker in *Texas HoldEm*, visit cafes in *Cafe World*, and even be a virtual farmer in the immensely popular *FarmVille* with friends in online social networks like

*Northwestern University, nima.haghpanah@gmail.com, nicimm@gmail.com. Supported by a Microsoft New Faculty Fellowship and by NSF via grants CCF-1055020 and CCF-1019169.

†Duke University, kamesh@cs.duke.edu. Supported by an Alfred P. Sloan Research Fellowship, a gift from Cisco, and by NSF via grants CCF-0745761, CCF-1008065, and IIS-0964560.

‡Google, mirrokni@gmail.com.

Facebook. Such applications are more fun when used with friends, and many such applications even explicitly reward players with many friends. The unique feature of such modern networked goods is that the underlying social network is *explicit*. This enables application distributors to use the network structures to market and sell these goods.

In this paper, we leverage explicit network structure to design mechanisms for selling networked goods. We primarily focus on goods, such as applications like *FarmVille* in online social networks, that are available in unlimited supply or, more precisely, can be produced at zero marginal cost. The network externalities of the good are implied by the private valuations of the social network users. In the most general case, users have a private value for each possible allocation of the good to a subset of users. This allows for arbitrary externalities, enabling say John Doe to value the good only if Kim Kardashian owns it despite having no direct relationship to her. While this makes sense for some goods, like fashion, many network goods like telephones or social network applications have value to a user only if users in his or her immediate neighborhood also own the good. The main focus of the paper is on a special case of this sort of direct externality, which we call *step function externalities*: that is, we suppose a user's value for the good is zero unless at least one of his or her neighbors or friends in the social network is also allocated the good.

We study *auction mechanisms*, or mechanisms that solicit bids from agents indicating their private value for various allocations, and then determine an allocation and prices in a way that maximizes expected revenue. As is common in economics, we work in a Bayesian setting where, while the realization of the private value is known only to the agent, it is drawn according to a commonly known distribution. Most literature on mechanism design assumes that agents value allocations solely based on the bundle of goods they receive, i.e., they are indifferent about the allocations of the other players. This is clearly violated in settings with externalities. Unfortunately, externalities significantly complicate mechanism design for the following reasons:

1. The efficient representation of values is no longer a trivial task, since in the most general case each bidder might need to report a value for each subset of allocated bidders.
2. More dimensions make satisfying incentive constraints harder (multi-parameter mechanism design is not well understood).
3. The space of feasible allocations might be more complex, which can make finding the optimal allocation a computationally hard problem.
4. Furthermore, the complexity of the feasible allocation space can easily cause the setting to violate *downward-closure*, i.e., not every subset of a feasible allocation is necessarily feasible. Thus the few known results for multi-parameter mechanism design can not be adopted generically.

We circumvent the first two issues by assuming a special structure on the players' values, namely that valuations satisfy step function externalities as defined above. Thus our problem is a single-parameter one, and so the representation and incentive constraints are straight-forward. Revenue maximization is also well understood for single-parameter settings. The seminal paper by Myerson [22] fully characterizes mechanisms that maximize revenue in expectation over the value distributions. By this characterization, the expected revenue of any mechanism is equal to the expected *virtual value* of the allocated agents, where the virtual value of an agent is a function of the valuation and its distribution and may be negative. In our setting, this characterization converts the optimal allocation problem to a combinatorial optimization problem, which is to maximize the sum of the virtual values over all *feasible* subsets. For step-function externalities, the feasibility constraint requires that all allocated agents have a neighbor who is also allocated. Graph-theoretically, this equates to finding, in a vertex-weighted graph with possibly negative vertex weights, a maximum-weight subset of vertices whose induced subgraph has no singleton components.

Although the optimal mechanism is easy to define, the third and fourth issues of mechanism design with externalities remain in our setting. We observe via reduction to set-buying that approximating the optimization problem within even a linear factor on *every* sampling of the values is NP-hard. On the other hand, we only need to find algorithms that perform well *in expectation* rather than in worst case: the Myerson mechanism we wish to approximate anyway provides an optimal average-case guarantee, and there is no mechanism with high revenue for every instantiation of values. Even on average, we prove that our problem remains APX-hard. However, we are able to design constant approximations for several versions of the problem.

We first note that there's a simple $(1/2)$-approximation for our problem. The algorithm divides the graph into two subsets of vertices, such that each vertex in each set has a neighbor in the other. This can be done, for example, by constructing a spanning tree of the graph and then taking a bipartite partitioning of it. The allocation strategy is to then pick the set with better expected revenue, extract revenue from that set, and allocate to users in the other set in order to maintain feasibility. This very simple algorithm does not use the structure of the social network in any deep way, and is therefore unable to give better approximations in even very simple social networks consisting of a single edge. In order to leverage knowledge of the network structure, we consider a greedy algorithm that iteratively allocates to influential vertices and their neighbors. Our main result shows that this can be used to obtain an $\frac{e}{e+1} \approx 0.73$-approximation to the optimal revenue for any distribution of values.

We additionally formulate our problem as a linear program (LP) whose variables represent the allocation and whose constraints use the network structure to characterize feasibility. We show how to round this LP to give a $\frac{e}{e+1}$-approximation, thereby matching the performance of our main greedy algorithm. The LP has several advantages however. First, it is hypothetically easy to incorporate additional feasibility constraints by simply including additional inequalities in the polytope and so might be of use in specific externality settings. Second, the LP exhibits some interesting mathematical properties. Namely, the gap of this LP is linear in the number of agents for a particular instantiation of values, and nonetheless we manage to prove a constant approximation on average. We do this through a novel average-case analysis of the rounding technique which may be useful in other applications. We also show that the expected integrality gap of our LP is 0.828, and thereby bound the approximation ratio of any LP-based mechanism.

We extend our setting to the more general *single-parameter submodular externalities* in which a bidder's value for an outcome is his private value times a known function of the set of players who receive the good. For such settings we study a class of mechanisms called *influence and exploit* in which some bidders (the *influencers*) are given the good for free and the remainder (the *exploited*) are offered an optimal price conditioned on the set of influencers. We show that the revenue is a submodular function of the set of influencers and hence we can use recent submodular function maximization results [7, 10] to design an influence-and-exploit mechanism whose revenue is within a 0.41-factor of the optimal influence and exploit mechanism. We also show that a randomization over influence and exploit mechanisms gives a 0.25-approximation to the optimal expected revenue of any mechanism by further submodularity arguments.

Related Work. Various settings with positive, negative, or mixed externalities have been studied in economics as well as computer science literature. Rohlfs [23] discusses positive externalities in the telephone industry in which a person's value for a telephone increases as more friends use it. A well-studied scenario with negative externalities is the allocation of ad slots in which a company's valuation for being listed as one of the *sponsored search* results decreases if their competitor is also listed [1, 4, 11, 12, 16, 19]. Finally, the valuation might have mixed externalities, as in the sale of nuclear weapons [15], in which countries prefer their allies rather than their foes to win the auction. Our work can be viewed as another in this line of literature, which addresses the difficulties of externalities in a specific setting of practical import by making application-specific assumptions.

Our work considers *auction mechanisms* with externalities. In contrast, some prior work considers instead the problem of posted price mechanisms [3, 2, 5, 14]. Particularly relevant to our work is that of Hartline, Mirrokni, and Sundararajan [14]. They consider the problem of finding a revenue-maximizing sequence of prices that are offered sequentially to buyers. They observed that simple influence and exploit strategies have revenue within a constant factor of the revenue of any equilibrium of any pricing sequence. They are reminiscent of our auction mechanisms which subsidize certain subsets of agents, and also our influence and exploit mechanisms for general single-parameter submodular externalities. However, unlike Hartline, Mirrokni, and Sundararajan [14], we provide approximation results with regards to the optimal auction revenue, which has a higher value than the optimal pricing strategy.

There has recently been a growing attention to the average case modeling of the optimization problems as opposed to the classical worst-case/adversarial agenda. It has been shown in different settings that such stochastic analyses help us achieve stronger guarantees than the worst-case analysis. An example is the online bipartite matching problem. In the adversarial setting, the celebrated result due to Karp, Vazirani and Vazirani [18] proves the tight approximation guarantee of $1 - 1/e$ for this problem. On the other hand, a sequence of papers initiated by the work of Feldman et al. [8] show improved guarantees for the stochastic version of the problem in which either the values are drawn from a known distribution or the sequence of arrivals is a random permutation [17, 20, 21]. Other papers study stochastic optimization problems in other settings such as Steiner tree and set cover [9, 13].

2 Preliminaries

We consider a society of n bidders located on the vertices in a social network $G(V, E)$, where the undirected edges model *friendship*. We assume for ease of exposition that the social network is connected. There is a supply k of a homogeneous good. Unless otherwise specified, we assume $k \geq n$ so that the supply is essentially unlimited (equivalently, the good can be reproduced at zero marginal cost).

An outcome $o \in \Omega = \{0, 1\}^n$ is a distribution of goods among bidders, where $o_i = 1$ if bidder i receives a copy of the good and 0 otherwise. Bidder i's *type* $v_i : \Omega \to \mathbb{R}^+ \cup \{0\}$ maps outcomes to non-negative real numbers, where $v_i(o)$ represents his value for outcome o and is positive only if he receives a copy of the good (i.e., $o_i = 1$). We study Bayesian mechanism design, in which one assumes that each type $v_i(\cdot)$ is drawn independently from a commonly-known distribution F_i. Let $F = F_1 \times \ldots \times F_n$ be the product distribution of F_i for all i; v be the vector of types, called the *type profile*; v_{-i} be the vector of types of agents other than i, and F_{-i} the distribution of v_{-i}. Throughout the paper, our algorithms assume access to expectations defined with respect to the distribution F. We assume these can be computed to within sufficient accuracy via sampling.

A (direct) mechanism is specified by two functions $\chi : \mathbb{R}^{n2^n} \to \Omega$ and $\rho : \mathbb{R}^{n2^n} \to \mathbb{R}^n$ in which $\chi(v)$ is the outcome given the reported type profile v, and $\rho_i(v)$ is the *payment* of agent i given the reported type profile v.[1] The utility of an agent for outcome o and price p is his value for the outcome minus the price he pays, $v_i(o) - p$. We say that a mechanism (χ, ρ) is Bayesian incentive compatible (BIC) if reporting the true type maximizes any player i's expected utility assuming that other players also report their true types, that is for every agent i and types v_i and v_i',

$$\begin{aligned}&\mathbf{E}_{v_{-i}\sim F_{-i}}[v_i(\chi(v)) - \rho_i(\chi(v))]\\ &\qquad \geq \mathbf{E}_{v_{-i}\sim F_{-i}}[v_i(\chi(v_i', v_{-i})) - \rho_i(\chi(v_i', v_{-i}))].\end{aligned}$$

Note that this is an interim notion, i.e., the agents choose the strategy that gives them the highest expected utility after observing their own private value. Similarly, we assume an interim notion of *individual rationality*, i.e., each agent's expected utility conditioned on their private value should be non-negative.

We consider *single-parameter* settings. In these settings, agents' values are a function of just one private parameter, called their *type*. As types are represented by a single parameter, v_i, the Bayesian assumption reduces to assuming that v_i is drawn independently from a distribution F_i over the non-negative reals, henceforth referred to as the type distribution of bidder i. We assume type distributions are *regular* and hence the corresponding *virtual values* are non-decreasing (see Subsection 2.1 for definitions).[2]

In the following subsection, we discuss optimal auction design for single-parameter settings. We encourage the reader familiar with these subjects to skip to Subsection 2.2 where we define the problem studied in this paper.

2.1 Optimal Auction Characterization

In his seminal paper, Myerson characterized the revenue of the optimal (i.e., revenue-maximizing) auction in terms of the *virtual values* of the agents [22]. We first define virtual values and then discuss the characterization result.

[1] Note the domain is exponential in general as types may assign different values to each of the 2^n possible outcomes.

[2] If the distributions are not regular, we can still apply our techniques using standard ironing arguments of Myerson [22].

Definition 1. *Suppose type v is drawn independently from a continuous distribution and let $F(v) = \Pr_z[z \leq v]$ be the cumulative function and $f(v) = F'(v)$ be the density function of the distribution. Then the virtual value function $\phi(v)$ is $v - \frac{1-F(v)}{f(v)}$.*

Virtual values may also be defined for discrete distributions.

Definition 2. *Suppose type v is drawn independently from a discrete distribution with support $\{v^1, \ldots, v^k\}$. Let $F(v^j) = \Pr[v \leq v^j]$ and $f(v^j) = \Pr[v = v^j]$. Then the virtual value function $\phi(v^j)$ is $v^j - \frac{1-F(v^j)}{f(v^j)}(v^{j+1} - v^j)$ for $j < k$ and $\phi(v^k) = v^k$.*

Note that virtual values may be negative. However, they are non-negative in expectation, a fact which enables many of our results.

Fact 1. *For any distribution F and value v, the virtual value $\phi(v)$ is non-negative.*

We will further assume that the distributions we study are *regular*, meaning that the corresponding virtual value function is non-decreasing in the support of F.

For a mechanism (χ, ρ) in a single-parameter setting, let $x_i(v) = v_i(\chi(v))/v_i$ if $v_i > 0$, and zero otherwise. In Myerson's characterization, it is the function x that is relevant for determining the revenue of the mechanism, and hence in a slight abuse of terminology we will refer to x as the *allocation function* even though there may be bidders i with $x_i(v) = 0$ that receive copies of the good (however they do not value the copy of the good because of the externalities). Accordingly define $x_i(v_i) = \mathbf{E}_{v_{-i} \sim F_{-i}}[x_i(v_i, v_{-i})]$ to be agent i's expected allocation for type v_i, where the expectation is over the types of other players.

In the single-parameter setting with regular distributions, Myerson showed that for any monotone increasing rule x, there is a unique corresponding payment rule ρ such that the resulting mechanism (χ, ρ) is BIC (where χ is any function that induces allocation function x and is not necessarily unique). The expected revenue of the mechanism is equal to its expected virtual value, $\sum \mathbf{E}_{v_i \sim F_i}[x_i(v_i)\phi_i(v_i)]$. Furthermore, if x is not monotone increasing, then there is no payment rule that makes the corresponding mechanism BIC. Restricting attention to BIC mechanisms is without loss of generality due to the revelation principle, and so to maximize revenue, one simply needs to find a rule χ satisfying all exogenous constraints (e.g., limited supply) whose corresponding feasible allocation function x is monotone and maximizes expected virtual value. We can therefore analyze the revenue of any monotone mechanism without explicitly defining the prices.

2.2 Externality Model

In this setting, we assume that each player i is assigned *local influence function* $g_i : 2^V \to \Re$ which is common knowledge. Following the previous literature on network influence, we assume that this local influence function is submodular[3] for each player, i.e., $g_i(S \cup \{j\}) - g_i(S) \geq g_i(S' \cup \{j\}) - g_i(S)$, for all $S \supseteq S'$, and $j \notin S$. Without loss of generality assume g_i is normalized such that $g_i(V) = 1$. Given this function and i's type v_i, define $S(o) = \{j : o_j = 1\}$ to be the

[3]Submodularity is used to model settings in which influence exerts diminishing returns.

set of players that are given the good in an outcome o. Then the value of i for o is defined to be $v_i(o) = v_i \cdot g_i(S(o))$.

For a mechanism (χ, ρ), the allocation function $x_i(v)$ is, by definition, $x_i(v) = v_i(\chi(v))/v_i = g_i(S(\chi(v)))$ and, invoking Myerson's characterization, we can write the expected revenue of the mechanism as $\sum_i E[x_i(v)\phi_i(v_i)]$.

We consider two special cases of submodular externalities: *concave externalities* and *step-function externalities*.

2.2.1 Concave Externalities

Let $N(i)$ be the neighborhood of i in G, i.e., $N(i) = \{j : (i,j) \in E\}$. In concave externalities, for each player i and subset S, $g_i(S) = \mathcal{G}(|S \cap N(i)|)$ for some concave function $\mathcal{G}(.)$ if $i \in S$. That is, the valuation of each bidder i depends on *the number* of his neighbors who have the good but not their identity, and also the local influence function is the same among all players.

2.2.2 Step-Function Externalities

Step-function externalities are a special case of submodular externalities in which the value of the influence function is 0 if the set of neighbors who receive the good is empty, and 1 otherwise. Let $N(i)$ be the neighborhood of i in G, i.e., $N(i) = \{j : (i,j) \in E\}$. Formally, bidder i's local influence function for an outcome o in which players $S(o)$ receive the good is:

$$g_i(S(o)) = \begin{cases} 1 & : i \in S(o), |S(o) \cap N(i)| \geq 1 \\ 0 & : \text{otherwise} \end{cases}$$

We say that a bidder i is *satisfied* by an allocation if $g_i(S(o)) = 1$, in which case $v_i(o) = v_i$. Otherwise we have $g_i(S(o)) = v_i(o) = 0$, and we say i is not satisfied by o. This models applications, like bridge tournaments, that require just one friend to be of value.

In this setting, for any mechanism (χ, ρ), we have $x_i(v) = g_i(S(\chi(v))) = 1$ if outcome $\chi(v)$ satisfies bidder i and zero otherwise. As a result, allocation functions x must satisfy the condition that $x_i(v) = 1$ only if for at least one (more generally, s) neighbor $j \in N(i)$ of i, we also have $x_j(v) = 1$. This means that in the subgraph induced by the allocated agents, every vertex must have degree at least 1 (more generally, s). Call such a subset of agents *feasible*. By the Myerson characterization discussed above, the optimal auction is thus specified by an allocation function that, given a type profile, allocates to a feasible subset of agents with maximum sum of virtual values (note this rule is necessarily monotone).

Graph-theoretically, the problem of finding an optimal allocation function equates to finding a subset of vertices of maximum weight whose induced subgraph has no isolated vertices. Unfortunately, we show in Section 3 that this problem is more general than the *set buying* problem, and therefore hard to approximate within a linear factor *on every* sampling of values. We also show that the problem of maximizing the *expected* revenue (over randomness of values), is APX-hard.

As the problem is NP-hard to solve optimally, we instead design a polynomial-time monotone allocation function whose expected revenue (as defined by the sum of virtual values) is close to the optimal expected revenue OPT, where the expectations are over the type distributions. We say an auction is an *α-approximation* if its expected revenue is at least $\alpha \times OPT$.

3 Hardness

By Myerson's characterization of optimal allocations, the problem of finding an optimal allocation function equates to finding a subset of vertices of maximum weight whose induced subgraph has no isolated vertices. Unfortunately, since virtual values and hence vertex weights might be negative, this problem is more general than the set buying problem (see, e.g., Feige et. al. [6]). We prove this formally in Lemma 1. We next show that the problem of maximizing the expected revenue, over the randomness of values, is APX-hard. Therefore our problem does not admit a PTAS unless P=NP, which justifies the search for constant factor approximations to the problem in later sections. The reduction is from a special case of set buying, which we call the *prize collecting set cover problem* (PCSCP).

Definition 3. *A* set-buying instance *is specified by a set of elements U and a collection $\mathcal{F}$ of subsets of U. There is a non-negative cost $c(S)$ associated with each set $S \in \mathcal{F}$, and a non-negative value $v(u)$ associated with each element $u \in U$. The* set-buying problem *is to pick some subsets $\mathcal{S} \subseteq \mathcal{F}$ to maximize the value of the elements covered by those sets minus the total cost of those sets, that is $\sum_{u \in Span(\mathcal{S})} v(u) - \sum_{S \in \mathcal{S}} c(S)$, where $Span(\mathcal{S}) = \cup_{S \in \mathcal{S}} S$.*

Theorem 2 (Feige et al [6])**.** *It is NP-hard to approximate the set-buying problem to within a linear factor.*

Lemma 1. *The optimal auction with step-function externalities is NP-hard to approximate to within a linear factor on every instantiation of values.*

Proof. For any instance $\mathcal{I} = (U, \mathcal{S})$ of the set-buying problem we construct a bipartite graph $G_{\mathcal{I}} = ((L, R), E)$ with a vertex $l_u \in L$ for each $u \in U$ and a vertex $r_S \in R$ for each $S \in \mathcal{F}$. We introduce an edge $(l_u, r_S) \in E$ for any element u and set S such that $u \in S$.

Consider an instance $\mathcal{I}$ and social network defined by the corresponding bipartite graph $G_{\mathcal{I}}$. Let the type distribution of bidder l_u be $v(u)$ with probability 1; and let the type distribution[4] of bidder r_S be 0 with probability $1/2$ and $c(S)$ with probaiblity $1/2$. Consider an instantiation of types in which each bidder l_u has type $v(u)$ and each bidder r_S has type 0. The induced virtual values are $v(u)$ for each bidder l_u and $-c(S)$ for each bidder r_S. For any feasible subset of bidders, include, without loss of generality, all bidders $l_u \in L$ with an allocated neighbor $r_S \in R$. Note that any feasible solution thus corresponds to a solution of the set-buying instance with the same value. The lemma then follows from the inapproximability of set buying. ☐

[4]Note this type distribution is not regular (and indeed our positive results hold for arbitrary distributions). For a reduction using regular distributions, consider drawing types uniform $[0, c(S)]$ and then consider the same instantion of values as before. The dissatisfying aspect of this proof, and the reason we do not include it, is that the required instantiation of types is a zero probability event.

The prize-collecting set cover problem (PCSCP) is a type of set cover problem in which all sets and all elements have equal costs and values, respectively. The problem seeks to maximize the value of covered elements plus the cost of unused sets.

Definition 4. *In the* prize collecting set cover problem *(PCSCP), we are given a collection of* n *sets* $\{S_1, S_2, \dots, S_n\}$ *over a universe* U*. For a collection* C *of sets, let* $Q_C = \cup_{i \in C} S_i$*. The goal is to find a collection* C^* *that maximizes* $\alpha|Q_{C^*}| + n - |C^*|$ *for some* $\alpha > 0$*.*

While this is equivalent, in optimality, to the set-buying problem of maximizing the value of covered elements *minus* the cost of the used sets, the two problems differ in approximability. The PCSCP is easier to approximate: although, as we show, it is APX-hard, there is a $e/(e+1)$-approximation for it. On the other hand, set-buying is not approximable to within a linear factor. We will show an approximation-preserving reduction from PCSCP to our problem, implying APX-hardness of our problem. We will then give an $e/(e+1)$-approximation for our problem.

Lemma 2. *There is an approximation preserving reduction from the prize collecting set cover problem to our problem.*

Proof. Given an instance of the prize collecting set cover problem, where the sets are denoted $\{S_1, S_2, \dots, S_n\}$ and the elements are denoted $e_1, e_2, \dots, e_m$, we construct a graph where there is a vertex for each set and each element, and an edge between S_i and e_j if $e_j \in S_i$. For each element e_j, the value is α with probability 1. Let $L \gg mn\alpha$. For each set S_i, the valuation follows distribution Bernoulli$(L-1, 1/L)$, so that the virtual valuation is -1 w.p. $1 - 1/L$ and $(L-1)$ w.p. $1/L$. To compute the revenue, we let $L \to \infty$. There are two events:

1. If at least one set has positive virtual valuation, the solution chooses all such sets and the corresponding covered elements. The revenue from each set is $(L-1)$ with probability $1/L$ for a total contribution to the expected revenue approaching n as $L \to \infty$. To compute the revenue from the elements, note that there is a set with positive virtual value with probability n/L, in which case the revenue of the elements is at most αm. Therefore, the contribution to the expected revenue from the elements is $\alpha mn/L \to 0$ as $L \to \infty$. Therefore, the optimal solution has contribution n from this event as $L \to \infty$, and this solution is trivial to compute.

2. If no set has positive virtual valuation (which happens w.p. $1 - n/L \to 1$), the solution chooses the sets (and the elements they cover) of the optimal PCSCP solution to get the value precisely $\alpha|Q_{C^*}| - |C^*|$, and this is the contribution from this event.

Therefore, the value of the optimal revenue solution is $\alpha|Q_{C^*}| + n - |C^*|$ as $L \to \infty$, and this completes the reduction. □

Theorem 3. *The prize collecting set cover problem (PCSCP) is* APX*-complete.*

Proof. We start with a 4-regular graph. On such a graph with $n = 152k$ nodes, for any $\epsilon > 0$, it is NP-HARD to decide if there is an independent set of size at least $(74 - \epsilon)k$ or at most $(73 + \epsilon)k$. Given such a graph G, construct the following prize collecting set cover instance: there is a set S_v

for every vertex v, and an element u_e for every edge e. Each set S_v contains the four elements u_e such that vertex v is adjacent to edge e in G. We further set $\alpha = 1/3$.

We first note that we can assume, without loss of generality, that any optimal solution C^* to the induced PCSCP instance uses only disjoint sets: i.e., $\forall i, j \in C^*, S_i \cap S_j = \emptyset$. Assume not and let $i, j \in C^*$ be two sets such that $S_i \cap S_j \neq \emptyset$. Consider the alternative solution $C = C^* \setminus \{j\}$. Since each set contains exactly four elements, Q_C contains at least $|Q_{C^*}| - 3$ elements, and so the value of C is $(1/3)|Q_C| + n - |C| \geq (1/3)(|Q_{C^*}| - 3) + n - (|C^*| - 1) = (1/3)|Q_{C^*}| + n - |C^*|$. Therefore, C is optimal as well.

Now consider a solution in which the chosen sets C are disjoint. Any such solution covers $4|C|$ vertices and so has value $n + (1/3)|C|$, and it corresponds to an independent set of vertices in G of size $|C|$. Thus it is NP-hard to distinguish between instances with an optimal solution of value at least $152k + (1/3)(74 - \epsilon)k$ or at most $152k + (1/3)(73 + \epsilon)k$, so it is NP-hard to approximation PCSCP to within a factor of $\frac{152+(1/3)(74)}{152+(1/3)73} \approx 1.002$. □

Corollary 4. *The problem of maximizing the expected revenue is APX-hard.*

4 Step-Function Externalities

Although the optimal auction is NP-hard to compute and NP-hard to approximate on *every* instantiation of values, it is in fact easy to approximate on average. The following very simple allocation function has expected revenue within a factor $1/2$ of the optimal expected revenue. In Appendix B, we show that this can be generalized to a $(1/4)$-approximation for the limited-supply setting.

> Divide vertices into two sets S_0 and S_1 such that each vertex $i \in S_0$ (respectively S_1) has a neighbor in the opposing set S_1 (respectively S_0). Note that this can be done efficiently, e.g. by computing a spanning tree of G and considering an arbitrary 2-coloring of it. Suppose S_0 has higher expected positive virtual value, i.e., $\sum_{i \in S_0} E[\max(\phi_i(v_i), 0)] \geq \sum_{i \in S_1} E[\max(\phi_i(v_i), 0)]$. For each vertex $i \in S_0$, choose an arbitrary neighbor $j_i \in S_1$. These vertices will be used to make our desired allocation feasible. Let $S_0^+ = \{i \in S : \phi(v_i) \geq 0\}$ be the bidders with positive virtual value in set S_0 for a particular instantiation of values, and $S_1' = \{j : j = j_i, i \in S_0^+\}$ be their designated neighbors. Then allocate to every bidder in $S_0^+ \cup S_1'$.

To see that this is a $(1/2)$-approximation, note that the expected optimum revenue is at most $\sum_i E[\max(\phi_i(v_i), 0)]$ since at best a mechanism can extract $\phi_i(v_i)$ from all bidders i with positive virtual value. The above mechanism gets expected revenue $\sum_{i \in S_0^+} E[\max(\phi_i(v_i), 0)]$ from the bidders in S_0^+, which is at least half the optimum expected revenue by linearity of expectation and our choice of S_0^+. For bidders $j \in S_1'$, note that j's expected allocation is independent of its value, i.e., we have $x_i(v_i) = x_i$ for some constant x_i. As a result, the revenue from i is $E[x_i \phi_i(v_i)] = x_i E[\phi_i(v_i)]$. Thus, since the expected virtual value of any bidder is non-negative (see Fact 1), the expected revenue of bidders in S_1' is non-negative.

Further note that this analysis is tight, as shown by the simple example of a single edge whose endpoints have value 1 with probability p and 0 with probability $1 - p$ for some $0 < p < 1$. Then the virtual value is 1 with probability p and $\frac{-p}{1-p}$ with probability $1-p$. Consider the mechanism which allocates to both nodes when at least one of them has positive value. The expected revenue of this mechanism is $2p^2 + 2p(1-p)(1 + \frac{-p}{1-p}) = 2p - 2p^2$ whereas the $(1/2)$-approximation described above has expected revenue p. The ratio of the two approaches $1/2$ as $p \to 0$.

The main reason why our analysis can not guarantee better than a 0.5-approximation is that the upper bound is quite loose. In fact, we show in Example A.1 in Appendix A that there exists a gap of 0.75 between the value of the upper bound and the optimum solution. Furthermore, our mechanism is "close to" a *threshold strategy* in which each player receives the good whenever his value surpasses a pre-defined threshold.[5] Using thresholds of 0 for players in S_0 and $\phi^{-1}(0)$ for players in S_1 yields a mechanism with the same revenue as that outlined above. We show in Appendix C that no threshold strategy can have better than 0.5 approximation.

In order to improve this approximation ratio, we need to leverage our detailed knowledge of the graph structure. In the remainder of this section, we present both a greedy and a linear-programming-based approach that get a 0.73-approximation for general distributions. Both approaches follow the same general auction scheme.

4.1 General Auction Scheme

The key observation is that any auction gets positive contributions from two types of nodes: those with positive virtual value who also have a neighbor with positive virtual value, and those with positive virtual value whose neighbors all have negative virtual value. Our general auction scheme first estimates the relative contributions of these two types and then tailors its strategy accordingly. In the extremes, where one type contributes most of the revenue, a simple deterministic scheme has a good approximation. When the contributions are more-or-less equal, we use either a greedy or LP-based algorithm to get a constant approximation.

To define the auction, we first introduce some notation to capture the contribution from the types discussed above. For an instantiation of values v, let $x_i^*(v)$ be the optimum allocation to agent i. Then optimal expected revenue is $E_v[\sum_i x_i^*(v)\phi_i(v_i)]$. Fix a player i and define the following events:

- P_i^+ is the event that $\phi_i(v_i) \geq 0$ and there exists $j \in N(i)$ such that $\phi_j(v_j) \geq 0$.
- P_i^- is the event that $\phi_i(v_i) \geq 0$ and all neighbors of i have negative virtual value.
- N_i is the event that $\phi_i(v_i) < 0$.

Observe that the expected revenue of the optimum allocation from agent i can be written as

$$\begin{aligned} E_v[x_i(v)\phi_i(v_i)] &= E_v[x_i^*(v)\phi_i(v_i)|P_i^+]Pr(P_i^+) \\ &\quad +E_v[x_i^*(v)\phi_i(v_i)|N_i]Pr(N_i) \\ &\quad +E_v[x_i^*(v)\phi_i(v_i)|P_i^-]Pr(P_i^-) \end{aligned}$$

Define

$$A_i^* = E_v[x_i^*(v)\phi_i(v_i)|P_i^-]Pr(P_i^-),$$

[5] Whether he is then *allocated* depends on whether any of his friends also pass their thresholds and receive the good.

$$B_i^* = E_v[x_i^*(v)\phi_i(v_i)|P_i^+]Pr(P_i^+),$$

and

$$C_i^* = E_v[x_i^*(v)\phi_i(v_i)|N_i]Pr(N_i)$$

(note C_i^* is negative). Let $A^* = \sum_i A_i^*$, $B^* = \sum_i B_i^*$, and $C^* = \sum_i C_i^*$ (note we do not need to compute these values in our auction scheme). The auction scheme runs three algorithms and then takes the best solution, breaking ties randomly. The first algorithm tries to extract a revenue of A^*; the second aims for a revenue of B^*; the third aims for a revenue of $(1-1/e)A^* + B^* + C^*$.

> **General Auction Scheme.** Run the following three algorithms and output the one with highest virtual value. In case of a tie, break the tie randomly.
>
> 1. Allocate to all nodes i for which $\phi_i(v_i) \geq 0$ as well as all nodes i for which N_i happens *and* for some neighbor j of i, $\phi_j(v_j) \geq 0$.
> 2. Allocate to all nodes for which P_i^+ happens.
> 3. Use one of the below subroutines.

The subroutines are discussed in the following sections. The combinatorial subroutine is greedy and uses intuition from the greedy algorithm for set cover. The LP-based subroutine uses a dependent randomized rounding scheme. The key property of each subroutine, proved in lemmas in the corresponding sections, is that each generates revenue $R = (1-1/e)A^* + B^* + C^*$. We show that this implies an $e/(e+1)$-approximation for our general auction scheme.

Theorem 5. *For any subroutine with expected revenue at least equal to $R = (1-1/e)A^* + B^* + C^*$, the approximation guarantee of the general auction scheme is $e/(e+1) \approx 0.73$.*

Proof. For an instantiation of values v, let $x_i(v)$ be the expected allocation of i (over the randomization in the auction scheme). Correspondingly, define A, B, C for the auction's allocation function, and note that the auction scheme's expected revenue is $A + B + C$. There are three cases depending on which algorithm the auction scheme selects. If the auction scheme selects the first algorithm, then $x_i(v) = 1$ for all i for which $\phi_i(v_i) \geq 0$ so $A \geq A^*$ (conditioned on the selection of the first algorithm). Lemma 3 further shows that $B + C \geq 0$ and so the total revenue of the auction in this case is at least $A \geq A^*$. If the auction scheme selects the second algorithm, then $x_i(v) = 1$ for all i such that P_i^+ happens, and so the revenue of the scheme is at least B. In the optimal allocation, $x_i^*(v)$ also equals 1 for all such i and hence the revenue of the auction is at least $B = B^*$. Finally, if the subroutine is invoked, by assumption it guarantees a revenue of $R = (1-1/e)A^* + B^* + C^*$.

The optimal expected revenue is at most $A^* + B^* + C^*$, and so the approximation ratio of the auction is at least

$$\min \frac{\max(A^*, B^*, (1-1/e)A^* + B^* + C^*)}{A^* + B^* + C^*}.$$

For computing the above minimum, normalize $A^* = 1$ and suppose $B^* = x$ and $B^* + C^* = rx$ for $0 \leq r \leq 1$ (such r exists since $B^* + C^* \geq 0$ by Lemma 3 and $C^* \leq 0$). Thus we want to compute the minimum of $\max(1, x, 1 - 1/e + xr)/(1 + xr)$ where $0 \leq r \leq 1$. We can do a case analysis on the maximum:

1. $xr \leq 1/e$. Then, we are minimizing $\max(1, x)/(1 + xr)$. We can set $xr = 1/e$, so that the lowest possible value is $e/(e + 1)$.
2. $xr \geq 1/e$ and $x(1-r) \leq 1-1/e$. Then we have $(1-1/e+xr)/(1+xr)$. Setting $xr = 1/e$ implies $e/(e + 1)$.
3. $x \geq 1$ and $x(1 - r) \geq 1 - 1/e$. Then we have $x/(1 + xr)$. But $xr \leq x + 1/e - 1$, so that we are minimizing $x/(x + 1/e)$ for $x \geq 1$, so that we again have $e/(e + 1)$.

Thus the approximation ratio is $e/(e + 1) \approx 0.73$. □

The proof of the approximation guarantee requires the following technical lemma which shows that the contribution of a node i when P_i^+ happens outweighs his contribution when N_i happens (for reasonable allocation rules).

Lemma 3. *For any monotone non-decreasing allocation function x that allocates to nodes i with $\phi_i(v_i) < 0$ only if there is a neighbor j with $\phi_j(v_j) \geq 0$, and corresponding B, C, we have $B + C \geq 0$.*

Proof. We prove the inequality for each node i separately. Let $N(i)$ be the neighborhood of i and note that:

$$\begin{aligned}
B_i + C_i &= E_{v_i}[x_i(v_i)\phi_i(v_i)|P^+]Pr(P^+) \\
&\quad +E_{v_i}[x_i(v_i)\phi_i(v_i)|N]Pr(N) \\
&= E_{v_i}[x_i(v_i)\phi_i(v_i)|P^+]Pr(P^+) \\
&\quad +(E_{v_i}[x_i(v_i)\phi_i(v_i)|N, \exists j \in N(i), \phi_j(v_j) \geq 0] \\
&\quad \cdot Pr(\exists j \in N(i), \phi_j(v_j) \geq 0)Pr(N) \\
&\quad +E_{v_i}[x_i(v_i)\phi_i(v_i)|N, \forall j \in N(i), \phi_j(v_j) < 0] \\
&\quad \cdot Pr(\forall j \in N(i), \phi_j(v_j) < 0))Pr(N).
\end{aligned}$$

But by assumption conditioned on N and the event $[\forall j \in N(i), \phi_j(v_j) < 0]$, $x_i(v_i) = 0$, and therefore, letting E be the event $[\exists j \in N(i), \phi_j(v_j) \geq 0]$, we have

$$\begin{aligned}
B + C &= E_{v_i}[x_i(v_i)\phi_i(v_i)|P^+]Pr(\phi_i(v_i) \geq 0)\Pr(E) \\
&\quad +E_{v_i}[x_i(v_i)\phi_i(v_i)|N, E]\Pr(N)\Pr(E) \\
&= (E_{v_i}[x_i(v_i)\phi_i(v_i)|P^+, E]Pr(\phi_i(v_i) \geq 0) \\
&\quad +E_{v_i}[x_i(v_i)\phi_i(v_i)|N, E]Pr(N))\Pr(E) \\
&= E_{v_i}[x_i(v_i)\phi_i(v_i)|E]\Pr(E) \\
&\geq 0,
\end{aligned}$$

where the second equality follows because the event P_i^+ implies event E and the last inequality follows because $x(v_i)$ is a monotone non-decreasing function of v_i as $\phi(\cdot)$ is regular and also that $E_{v_i}[\phi_i(v_i)] = 0$ (see Fact 1). □

The last step is to show that our auction scheme is BIC by proving that it is monotone. It is easy to check the monotonicity of the first two algorithms, and also both subroutines used as the third algorithm. Some attention has to be payed to the cases in which we switch between algorithms when an agent changes his value. One can check that as a player increases his value, if the value of any of the algorithms increase, that player has to be allocated in the new solution. Thus, when we consider the set of algorithms that produce the maximum value, the algorithms that are added to the set of maximizers (if any) allocate that player (possibly some algorithms are dropped out of the set of maximizers). By our random tie-breaking among algorithms, this does not decrease the probability of allocation.

4.2 Greedy Subroutine

The greedy subroutine follows intuition from the greedy algorithm for set cover. Let P be set of agents i with non-negative virtual value $\phi_i(v_i) \geq 0$ who have neighbors with non-negative virtual value, i.e., $\{i : \phi_i(v_i) \geq 0 \text{ and } \exists i' \in N(i), \phi_{i'}(v_{i'}) \geq 0\}$. For each node j with negative virtual value $\phi_j(v_j) < 0$, associate a set $Q_j = \{i : i \in N(j), i \notin P, \phi_i(v_i) \geq 0\}$, i.e., Q_j is the set of neighbors of j with non-negative virtual value who are not in P. If we select j (which comes at a cost of $\phi_j(v_j)$), then we cover Q_j (gaining revenue equal to the sum of virtual values of agents $i \in Q_j$). The greedy subroutine initially selects P and then iteratively select sets Q_j whose marginal "bang-per-buck" is maximized.

Greedy Subroutine.

1. Initialize the set of allocated nodes $S \leftarrow P$.
2. Initialize the bang-per-buck of each Q_j to $b_j = -\sum_{i \in Q_j} \phi_i(v_i)/\phi_j(v_j)$.
3. Repeat until for all Q_j, $b_j < 1$:
   (a) Let j^* be the node with $b_{j^*} = \max_j b_j$.
   (b) Set $S \leftarrow S \cup \{j^*\} \cup Q_{j^*}$.
   (c) For all Q_j, update $b_j = -\sum_{i \in Q_j \cap (V-S)} \phi_i(v_i)/\phi_j(v_j)$.

Lemma 4. *The expected value of greedy is at least* $(1 - 1/e)A^* + B^* + C^*$.

Proof. Note that both OPT and the greedy algorithm select all the vertices in P, and therefore get revenue of B^* from them.

Without loss of generality, assume that the rest of the positive elements all have unit value by replicating them. Let n_j be the *number* of elements in Q_j and let Q^* be the set of nodes with negative virtual value that OPT picks. Therefore,

$$OPT = \sum_{j \in Q^*} n_j - \phi_j(v_j)$$

For each Q_j, sort the elements by the decreasing order of the time greedy covers them. Let the time stamp be some very small value for any element not covered. Notice that we sort the elements of each set independently, and therefore an element which is in multiple sets is going to have a possibly different index in each of them. So when greedy covers the i'th element of a set Q_j, all the elements $1, \ldots, i-1$ of that set are uncovered. Note that if $\phi_j(v_j) \leq i$ then i is covered by greedy since otherwise Q_j has positive value. At the time i is covered by greedy, the option of picking set Q_j gives the per-element reward of $1 - \phi_j(v_j)/i$. So we can write the following lower bound for the value that greedy gets:

$$
\begin{aligned}
\sum_{j \in Q^*} \sum_{c_j \leq i}^{n_j} 1 - \phi_j(v_j)/i &= \sum_{j \in Q^*} n_j - \phi_j(v_j) - \phi_j(v_j) \ln(\phi_j(v_j)/n_j) \\
&= \sum_{j \in Q^*} \phi_j(v_j)\left(\frac{n_j}{\phi_j(v_j)} - \ln\left(\frac{n_j}{\phi_j(v_j)}\right)\right) - \phi_j(v_j) \\
&\geq \sum_{j \in Q^*} \phi_j(v_j)\left(\frac{n_j}{\phi_j(v_j)}(1 - 1/e)\right) - \phi_j(v_j) \\
&= \sum_{j \in Q^*} n_j(1 - 1/e) - \phi_j(v_j),
\end{aligned}
$$

where the inequality followed because for any $a \geq 1$, $a - \ln(a) \geq a(1 - 1/e)$. □

4.3 LP-Based Subroutine

As discussed above, the main hurdle in the analysis of the simple auction schemes was the loose upper bound on the optimal expected revenue. In this section, we use a linear program whose constraints characterize the feasible allocation rules as an upper bound. We then use this LP to bound the expected revenue of an LP-based subroutine for the auction scheme.

Recall that for each profile of types v with virtual valuation functions $\{\phi_i(\cdot)\}$, the optimal revenue is equal to the maximum sum of virtual values among feasible allocations. In step-function externalities, an allocation $x(\cdot)$ is feasible if each vertex i with $x_i(v) = 1$ had a neighbor j with $x_j(v) = 1$. Hence we can write the following LP relaxation of the optimum revenue:

$$
\begin{array}{lll}
\max\limits_{x} & \sum_i x_i(v)\phi_i(v_i) & \\
s.t. & x_i(v) \leq \sum_{j \in N(i)} x_j(v) & \forall i \\
 & 0 \leq x_i(v) \leq 1 & \forall i.
\end{array}
\tag{1}
$$

Each instantiation of types induces one such LP. As discussed in Section 3, given the instantiation of types, our problem is more general than the set-buying problem studied in Feige et. al. [6] whose LP-relaxation is shown to have linear gap. Hence the LP value might seem like a very loose upper bound. However, recall that we only require our auction to have close-to-optimal revenue *on average*. In other words, we need a rounding scheme whose expected value, over the distribution of LPs induced by the type distributions, is close to the expected value of the LPs. Thus we can perform poorly on hard instances so long as we do well on average, and so LPs with linear worst-case integrality gaps might still be useful in designing an LP-based subroutine with good approximation ratios.

LP-Based Subroutine. Solve LP 1 for the instantiation of types v and let $x_i^*(v)$ be an optimal solution.

1. For each i with $\phi_i(v_i) < 0$, give i a copy of the good with probability $x_i^*(v)$.
2. For each i with $\phi_i(v_i) \geq 0$, give i a copy of the good if it has a neighbor j that either
   (a) has non-negative virtual value $\phi_j(j) \geq 0$, or
   (b) has negative virtual value $\phi_j(j) < 0$ and received the good in step 1.

To use this subroutine in our auction scheme, we must argue its expected revenue is at least $R = (1 - 1/e)A^* + B^* + C^*$. The analysis of the randomized rounding requires a key lemma: the LP constraints corresponding to an agent i with positive virtual value must be tight in an optimal solution $x^*(v)$. Namely, $x_i^*(v) = \min(1, \sum_{j \in N(i)} x_j^*(v))$. Hence to round and get constant contribution from these agents, we can round the nodes with negative virtual value with probability equal to their LP values and then round nodes with positive virtual value to one if some neighbor was rounded to one. To bound the expected allocation of such an agent i in the rounding, we note that in the worst-case all neighbors of i have negative virtual value. However, even in this case, i is allocated so long as at least one $j \in N(i)$ receives the good. This happens with probability x_j^* for neighbor j and so the allocation probability of i from the rounding scheme is at least $1 - \prod_{j \in N(i)}(1 - x_j^*)$. This is within a $(1 - 1/e)$ fraction of x_i^*.

Let $x_i(v)$ be the expected allocation of i in the subroutine, and define A, B, C as the expected revenue contributions from nodes of each type accordingly.

Lemma 5. $A \geq (1 - 1/e)A^*$.

Proof. First note that conditioned on event P^-, we have $x_i^*(v) = \min(1, \sum_{j \in N(i)} x_j^*(v))$, and $x_i(v) = 1 - \prod_{j \in N(i)}(1 - x_j^*(v))$. Let $y = \sum_{j \in N(i)} x_j^*(v)$ and $d = |N(i)|$. Fixing the value of $\sum_{j \in N(i)} x_j^*(v)$, the minimum of $1 - \prod_{j \in N(i)}(1 - x_j^*(v))$ happens when all the variables are equal, in which case we have $x_i(v) = 1 - (1 - \frac{y}{d})^d \geq 1 - \frac{1}{e^y}$. Thus when $y \leq 1$, we have $x_i^*(v) = y$ and so $\frac{x_i(v)}{x_i^*(v)}$ is at least $\frac{1-e^{-y}}{y}$, whose minimum value is equal to $1 - \frac{1}{e}$. When $y \geq 1$, we have $x_i^*(v) = 1$ and so $\frac{x_i(v)}{x_i^*(v)}$ is at least $1 - e^{-y}$, whose minimum value is again $1 - \frac{1}{e}$. Therefore we have

$$\begin{aligned} A_i &= E_{v_i}[x_i(v_i)\phi_i(v_i)|P^-]Pr(P^-) \\ &\geq (1 - \frac{1}{e})E_{v_i}[x_i^*(v_i)\phi_i(v_i)|P^-]Pr(P^-) \\ &= (1 - \frac{1}{e})A_i^* \end{aligned}$$

Summing over i yields the result. □

Theorem 6. *The expected revenue of the LP-based subroutine is $R = (1 - 1/e)A^* + B^* + C^*$.*

Proof. Lemma 5 shows $A \geq (1 - 1/e)A^*$. Furthermore, from the construction of x we see that conditioned on P^+ and N, x and x^* are equal so $B + C = B^* + C^*$. Therefore the total revenue of the subroutine is at least $(1 - 1/e)A^* + B^* + C^*$. □

We now prove that the above LP has integrality gap at most 0.828. This means that we can not use the LP solutions as an upper bound in order to get approximation guarantees better than 0.828. We show the gap by proving the gap on the analogous LP for the PCSCP, which using the reduction in Lemma 1 implies the gap on the original LP.

Theorem 7. *The above LP has integrality gap at most* 0.828*.*

Proof. We construct an LP gap instance for the prize collecting set cover problem. In our instance, the input is a graph; the sets are vertices and the elements are edges, so that each edge is present in the sets corresponding to its incident vertices. For an n-vertex graph, the goal is to choose a subset X of vertices to maximize $\alpha|E(X)| + n - |X|$, where $E(X)$ is the subset of edges incident to some vertex in X.

The LP has a variable x_e for each edge, which is 1 if the edge is selected in the event that all vertices in the graph have negative virtual valuation. Similarly, y_v is the variable denoting whether vertex v is selected in the same event. The LP can be reformulated as:

$$\text{Maximize} \quad n - \sum_v y_v + \alpha \sum_e x_e$$

$$\begin{array}{rcll} x_e & \leq & y_u + y_v & \forall e = (u,v) \in E \\ x_e, y_u & \in & [0,1] & \forall e \in E, u \in V \end{array}$$

Consider a complete graph on n vertices, for large n. By appropriate scaling, let us rewrite the objective as $|E(X)| + \alpha n(n - |X|)$. Set all $y_v = 1$ and $x_e = 1/2$. For this fractional solution, the objective is approximately $n^2(1+\alpha)/2$. Suppose the optimal integer solution chooses k vertices and all incident edges. Its value is approximately $nk - k^2/2 + \alpha n(n-k)$. Optimizing over k, we obtain $k = n(1-\alpha)$, so that the optimal value is $n^2(1+\alpha^2)/2$. The ratio is therefore $(1+\alpha^2)/(1+\alpha)$, so that $\alpha = \sqrt{2} - 1$. This yields a ratio of $2(\sqrt{2}-1) = 0.828$. □

5 Submodular Externalities

In order to design an approximately optimal mechanism for the more general problem with submodular externalities, we identify a set of mechanisms, called *influence-and-exploit* mechanisms. In the following, we first show that a simple random-sampling mechanism which belongs to this category of mechanisms achieves a 0.25-approximate mechanism for this problem. Then, we focus on optimizing over these mechanisms and design improved approximation algorithms for this problem. We start by defining influence and exploit-mechanisms:

Definition 5. *For a fixed price p and any set of players S, define the Influence-and-Exploit Mechanism IE(S) as follows. Give the good to any $i \in V \backslash S$ regardless of its value and to any $i \in S$ if his value is more than the threshold p.*

5.1 Constant Approximation

First, we observe that a simple IE mechanism gives a 0.25-approximation to the optimal revenue for the setting of single-parameter submodular externalities. Consider the following algorithm:

- Let S be a random subset of bidders where each $i \in S$ is chosen independently with probability $\frac{1}{2}$.
- *Influence:* Give the good to all $i \in V \setminus S$ regardless of the value.
- *Exploit:* Give the good to a bidder $j \in S$ if $v_j \geq p_j(S)$, where $p_j(S) = \phi_{j,S}^{-1}(0)$ is the inverse virtual value of zero for the distribution $F_{j,S}$.

In order to prove the approximation guarantee, we make use of the following lemma.

Lemma 6 ([7]). *For a ground set V, Let $f : 2^V \to \Re$ be a monotone submodular set function. Form set S by picking elements $i \in V$ independently at random with some fixed probability p. Then*

$$\mathbf{E}[f(S)] \geq p\mathbf{E}[f(V)]$$

Define the *revenue function* $R_i(S) = \max_p p(1 - F_{i,S}(p))$, where $F_{i,S}(p) = Pr_{v_i \in F_i}(v_i g_i(S) \leq p)$. We first prove that $\sum_i R_i(V)$ is an upper bound on the revenue of any mechanism.

Lemma 7. *The expected revenue of any Bayesian incentive compatible mechanism is at most $\sum_i R_i(V)$.*

Proof. Recall that we normalized $g_i(V) = 1$. As a result, $F_{i,V}(p) = F_i(p)$. So by definition $R_i(V) = \max_p p(1 - F(p))$. Consider any mechanism with allocation function $x_i(v_i) \leq 1$. By Myerson's characterization, the expected revenue of the mechanism is $\sum_i E_{v_i}[x_i(v_i)\phi_i(v_i)] \leq \sum_i E[\max(0, \phi_i(v_i))] = \int_{\phi^{-1}(0)}^{\infty} \phi_i(x) f(x) dx = \int_{\phi^{-1}(0)}^{\infty} (xf(x) - (1 - F(x)))dx = -x(1 - F(x))|_{\phi^{-1}(0)}^{\infty} = p_i(S)(1 - F_{i,S}(p_i(S))) = R_i(V)$. □

Lemma 8. *If the revenue function is submodular for all agents, then the above mechanism is a 4-approximation of the optimal mechanism.*

Proof. Consider any agent i. With probability 1/2, it chosen to be in S. Fixing the set S, the expected revenue we get from i is $R_i(S) = p_i(S)(1 - F_{i,S}(p_i(S)))$. Now note that each agent is independently sampled, so over the random choices of the mechanism, and by submodularity of $R_i(S)$, the expected revenue from i (conditioned on being in S) is at least $R_i(V)/2$. Since we get this revenue with probability 1/2, the expected revenue from i is at least $R_i(V)/4$. This gives a 4-approximation. □

Similar to [14], we may simply assume that the revenue function R_i is monotone and submodular for each bidder, and indeed our result holds for any settings that induce monotone submodular revenue functions. Interestingly, for the single-parameter submodular setting, the submodularity of the revenue function follows from the submodularity of the local influence function.

Lemma 9. *The revenue function is submodular for the single-parameter submodular externality setting, and the concave externality setting.*

Proof. Consider a player i with distribution F_i over v_i. Then

$$\begin{aligned} R_i(S) &= \max_p p(1 - F_{i,S}(p)) \\ &= \max_p p(1 - F_i(p/g(S))) \\ &= g(S) \max_{p'} p'(1 - F(p')) \end{aligned}$$

where $p' = p/g(S)$. Submodularity of $R_i(.)$ then follows directly from submodularity of $g(.)$. $\square$

Applying the above two lemmas, we conclude that the following:

Theorem 8. *There exists a $\frac{1}{4}$-approximate IE mechanism to the optimum revenue in the single-parameter submodular externality model, and thus in the concave externality model.*

5.2 Optimizing over IE Mechanisms

Now that we proved that IE mechanisms achieve a constant-factor approximation to the optimal revenue, it would be interesting to optimize among IE mechanisms. To do so we need to find a set $V \setminus S$ of initial (influential) bidders to get the good regardless of their value, and then exploit the remaining bidders by setting optimal thresholds as above. Let $\chi(v)$ be the outcome of this strategy, that is, $\chi_i(v) = 1$ if the good is given to i for the profile of types v in IE(S). Let $\Phi(S)$ be the expected revenue of IE(S). Our goal is to find a subset S of bidders that maximizes $\Phi(S)$. We do so by arguing that $\Phi(S)$ is a (not necessarily monotone) submodular function and then using submodular function maximization results. We present the results in this section with regard to concave externalities in order to keep notation simple; the results extend easily to the more general submodular externalities. We first characterize the expected revenue of any IE strategy.

Lemma 10. *Let $X_{i,S}(v) = |\{j \in N(i) : \chi_j(v) = 1\}|$ where $N(i)$ is the neighborhood of i in G. Then the expected revenue of any IE strategy, IE(S), for each $i \in S$ is equal to $p(1 - F_i(p))E_v[h(X_{i,S}(v))]$ where $h(\cdot)$ is the concave function defining the externality (i.e., $g_i(o) = o_i \cdot h(|\{j \in N(i) : o_j = 1\}|)$).*

Proof. Consider IE(S) with allocation function x and outcome function χ. By Myerson's characterization, we can write the expected revenue of i in IE(S) as

$$\begin{aligned} E_v[x_i(v)\phi_i(v_i)] &= E_v[g_i(\chi(v))\phi_i(v)] \\ &= E_v[\chi_i(v)h(X_{i,S}(v))\phi_i(v)]. \end{aligned}$$

Note in any IE strategy, $\chi_i(v)$ and $\chi_j(v)$ are independent random variables (when v is drawn from F) for any $i \neq j$. Thus $\chi_i(v)$ is also independent from $X_{i,S}(v)$. So we can write the revenue of i as $E[h(X_{i,S}(v))]E[\chi_i(v)\phi_i(v_i)]$. Since we set $\chi_i(v_i) = 1$ when $\phi(v_i) \geq 0$, $E[\chi_i(v)\phi(v_i)]$ is equal to the optimum revenue from distribution F_i, which is equal to $p(1 - F_i(p))$. $\square$

We next prove the key structural property of the revenue function $\Phi(S)$ for IE mechanisms, namely that it is submodular.

Lemma 11. *The set function Φ is a non-negative submodular function of S.*

Proof. First note that each agent $i \in V \backslash S$ contributes 0 to the revenue, and each $i \in S$ contributes $\Phi_i(S) = p(1 - F_i(p))E_v[h(X_{i,S}(v))]$, where $X_{i,S}(v)$ is a random variable denoting the number of i's neighbors that are given the good to at profile v, that is $X_{i,S}(v) = |\{j \in N(i) : \chi_j(v) = 1\}|$. For all i, S, let $F_{i,S}$ be the discrete distribution (with density function $f_{i,S}$) of $X_{i,S}(v)$ when v is drawn from the joint distribution of types. We show submodularity of $\Phi(.)$ by proving submodularity of all $\Phi_i(.)$ for all i, that is $\Phi_i(S \cup \{j\}) - \Phi_i(S) \leq \Phi_i(S' \cup \{j\}) - \Phi_i(S')$, for all $S' \subseteq S$ and all i and j. Submodularity of $\Phi(.)$ follows from submodularity of $\Phi_i(.)$'s, since $\Phi(S) = \sum_{i \notin S} \Phi_i(S)$. First note that if i is not a neighbor of j, then we have $0 = \Phi_i(S \cup \{j\}) - \Phi_i(S) \leq \Phi_i(S' \cup \{j\}) - \Phi_i(S') = 0$. Now assume that i is a neighbor of j. Define $\Phi = \Phi_i(S \cup \{j\}) - \Phi_i(S)$. Now we have

$$
\begin{aligned}
\Phi &= p(1 - F_i(p))(E_v[h(X_{i,S\cup\{j\}}(v))] - E_v[h(X_{i,S}(v))]) \\
&= p(1 - F_i(p))(E_{k \sim F_{i,S\cup\{j\}}}[h(k)] - E_{k \sim F_{i,S}}[h(k)]) \\
&= p(1 - F_i(p)) \sum_k h(k)(f_{i,S\cup\{j\}}(k) - f_{i,S}(k))
\end{aligned}
$$

First we show that $f_{i,S\cup\{j\}}(k) = F_j(p)f_{i,S}(k+1) + (1 - F_j(p))f_{i,S}(k)$. To compute the probability that i has k neighbors using strategy IE($S \cup \{j\}$), we consider two events. First is the event in which $v_j < p$, which happens with probability $F_j(p)$. In this case, we need $k - |V \backslash (S \cup \{j\})|$ neighbors of i in set S to have value more than p. If this happens when using strategy IE(S), i is going to have $k - |V \backslash (S \cup \{j\})| + |V \backslash (S \cup \{j\})| + 1 = k + 1$ neighbors that are allocated (note that j is in the influence set and therefore allocated). The probability of this event is $f_{i,S}(k+1)$ by definition. The second event is the event in which $v_j \geq p$, which happens with probability $1 - F_j(p)$. In this case, we need $k - |V \backslash (S \cup \{j\})| - 1$ neighbors of i in set S to have value more than p. If this happens when using strategy IE(S), i is going to have $k - |V \backslash (S \cup \{j\})| - 1 + |V \backslash (S \cup \{j\})| + 1 = k$ neighbors that are allocated. The probability of this event is $f_{i,S}(k)$ by definition. Summing up, we conclude our desired equation, $f_{i,S\cup\{j\}}(k) = F_j(p)f_{i,S}(k+1) + (1 - F_j(p))f_{i,S}(k)$.

As a result,

$$
\begin{aligned}
\Phi &= p(1 - F_i(p))F_j(p) \sum_k h(k)(f_{i,S}(k+1) - f_{i,S}(k)) \\
&= p(1 - F_i(p))F_j(p) \sum_k f_{i,S}(k)(h(k-1) - h(k))
\end{aligned}
$$

Now recall that h is a concave function of k. As a result, $H(k) = h(k-1) - h(k)$ is a non-decreasing function of k. Therefore,

$$\begin{aligned}
& \Phi_i(S' \cup \{j\}) - \Phi_i(S') - (\Phi_i(S \cup \{j\}) - \Phi_i(S)) \\
= \;& p(1-F_i(p))F_j(p) \sum_k H(k)(f_{i,S'}(k) - f_{i,S}(k)) \\
= \;& p(1-F_i(p))F_j(p) \sum_k H(k)(F_{i,S'}(k) - \\
& F_{i,S'}(k-1) - (F_{i,S}(k) - F_{i,S}(k-1))) \\
= \;& p(1-F_i(p))F_j(p) \\
& \cdot \sum_k (F_{i,S'}(k) - F_{i,S}(k))(H(k) - H(k+1))
\end{aligned}$$

Note that for $S' \supset S$, $F_{i,S'}(k) \geq F_{i,S}(k)$. This is because any vertex in $S' \backslash S$ is always allocated in IE(S), but only with some probability in IE(S'), and therefore the probability that i has k or less allocated neighbors in IE(S) is only less than in IE(S'). So $F_{i,S'}(k) - F_{i,S}(k) \geq 0$ for all k. Also, since H is a non-decreasing function, $H(k) - H(k+1) \leq 0$.

It only remains to consider the revenue function of j when we add j to sets. For S such that $j \notin S$, we have

$$\Phi_j(S \cup \{j\}) - \Phi_j(S) \;=\; p(1-F_j(p))E_v[h(X_{j,S}(v))]$$

Again, note that for $S' \supset S$, $F_{j,S}(k) \leq F_{j,S'}(k)$, therefore

$$\begin{aligned}
E_{F_{j,S'}}[h(k)] \;-\; & E_{F_{j,S}}[h(k)] \\
= \;& \sum_k h(k)(f_{j,S'}(k) - f_{j,S}(k)) \\
= \;& \sum_k (F_{j,S'}(k) - F_{j,S}(k)) \\
& \cdot (h(k) - h(k+1)) \leq 0
\end{aligned}$$

□

Function $\Phi(.)$ as described above is non-negative and submodular, but not necessarily monotone. In order to obtain a constant-factor approximation for maximizing over IE mechanisms, we can simply apply non-monotone submodular maximization algorithms for this problem [7, 10]. For example, the following simple local search algorithm gives a 0.33-approximation to this problem [7]: (i) Let $S = \{i | i = \arg\max_{i' \in V}(\Phi(\{i'\})\}$, and (ii) at each step either add or remove a bidder i from S if this adding or removing increases the value of $\Phi(S)$ by a $1 + \frac{\epsilon}{n}$ factor, (ii) After reaching a local optimal L, output the better of L and $\bar{L}$. The above simple algorithm acheives 0.33-approximation for the problem of maximizing over IE strategies. One can apply a recently-developed randomized local search algorithm to acheive a 0.41-approximation of Oveis Gharan and Vondrak [10] for this problem. We conclude with the following theorem:

Theorem 9. *The problem of optimizing over IE mechanisms can be approximated within a factor* 0.41 *in polynomial time.*

References

[1] G. Aggarwal, J. Feldman, S. Muthukrishnan, and M. Pál. Sponsored search auctions with markovian users. In *WINE*, pages 621–628, 2008.

[2] H. Akhlaghpour, M. Ghodsi, N. Haghpanah, V. S. Mirrokni, H. Mahini, and A. Nikzad. Optimal iterative pricing over social networks (extended abstract). In *WINE*, pages 415–423, 2010.

[3] N. Anari, S. Ehsani, M. Ghodsi, N. Haghpanah, N. Immorlica, H. Mahini, and V. S. Mirrokni. Equilibrium pricing with positive externalities (extended abstract). In *WINE*, pages 424–431, 2010.

[4] S. Athey and G. Ellison. Position auctions with consumer search. Forthcoming, Quarterly Journal of Economics, 2010.

[5] O. Candogan, K. Bimpikis, and A. E. Ozdaglar. Optimal pricing in the presence of local network effects. In *WINE*, pages 118–132, 2010.

[6] U. Feige, N. Immorlica, V. S. Mirrokni, and H. Nazerzadeh. Pass approximation. In *APPROX-RANDOM*, pages 111–124, 2009.

[7] U. Feige, V. S. Mirrokni, and J. Vondrak. Maximizing non-monotone submodular functions. In *FOCS '07*, pages 461–471, Washington, DC, USA, 2007. IEEE Computer Society.

[8] J. Feldman, A. Mehta, V. S. Mirrokni, and S. Muthukrishnan. Online stochastic matching: Beating 1-1/e. In *FOCS*, pages 117–126, 2009.

[9] N. Garg, A. Gupta, S. Leonardi, and P. Sankowski. Stochastic analyses for online combinatorial optimization problems. In *SODA*, pages 942–951, 2008.

[10] S. O. Gharan and J. Vondrak. Submodular maximization by simulated annealing. In *SODA*, 2011.

[11] I. Giotis and A. R. Karlin. On the equilibria and efficiency of the gsp mechanism in keyword auctions with externalities. In *WINE*, pages 629–638, 2008.

[12] R. Gomes, N. Immorlica, and V. Markakis. Externalities in keyword auctions: an empirical and theoretical assessment. In *WINE'09*, 2009.

[13] F. Grandoni, A. Gupta, S. Leonardi, P. Miettinen, P. Sankowski, and M. Singh. Set covering with our eyes closed. In *FOCS*, pages 347–356, 2008.

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

[15] P. Jehiel, B. Moldovanu, and E. Stacchetti. How (not) to sell nuclear weapons. *American Economic Review*, 86(4):814–829, 1996.

[16] P. Jeziorski and I. Segal. What makes them click: Empirical analysis of consumer demand for internet search advertising. working paper, 2009.

[17] C. Karande, A. Mehta, and P. Tripathi. Online bipartite matching with unknown distributions. In *STOC*, pages 587–596, 2011.

[18] R. M. Karp, U. V. Vazirani, and V. V. Vazirani. An optimal algorithm for on-line bipartite matching. In *STOC*, pages 352–358, 1990.

[19] D. Kempe and M. Mahdian. A cascade model for externalities in sponsored search. In *WINE*, pages 585–596, 2008.

[20] M. Mahdian and Q. Yan. Online bipartite matching with random arrivals: an approach based on strongly factor-revealing lps. In *STOC*, pages 597–606, 2011.

[21] V. H. Manshadi, S. O. Gharan, and A. Saberi. Online stochastic matching: Online actions based on offline statistics. In *SODA*, pages 1285–1294, 2011.

[22] R. Myerson. Optimal auction design. *Mathematics of Operations Research*, pages 58–73, 1981.

[23] J. Rohlfs. A theory of interdependent demand for a communications service. *The Bell Journal of Economics and Management Science*, 5(1):16–37, 1974.

A Loose upper bound

Consider an upper bound on the optimal expected revenue which is equal to the sum of positive virtual values. The following example shows that the optimal expected revenue can be just a $3/4$-fraction of this upper bound, indicating that better approximations require better upper bounds.

Example A.1. *Consider the following tree: there is a root vertex; the root has n children, called level 1 nodes; and each level 1 node has one child, the level 2 nodes. Assume that the distribution on values is such that the virtual value is 1 with probability $p = 1/2$, and -1 with probability $1/2$. The upper bound's value is $\sum_i E[\max(0, \phi(v_i))] = (2n+1)/2$. Now consider any pair v_1 and v_2. Assuming that the root is always allocated (the best case for the pair), the optimum solution is to allocate both when they are positive, only v_1 when it is the only one with positive value, and neither otherwise. The expected value of this allocation is $3/4$. There are n pairs and at best the optimum can get revenue 1 from the root, and so the optimum revenue is at most $\frac{3}{4}n + 1$. Hence the ratio of the optimum to the upper bound is at most $\frac{\frac{3n}{4}+1}{\frac{2n+1}{2}} \to 3/4$.*

B Limited-Supply Setting

All auctions discussed so far assume the auctioneer has an unlimited supply of the good. When there is a limited supply, we must modify the above techniques to satisfy the supply constraint. The below theorem shows how to extend our simple $(1/2)$-approximation presented at the beginning of Section 4 to get a $(1/4)$-approximation with limited supply. The LP-based auction also has a natural extension to the limited-supply setting. Namely, we can add a constraint to the LP forcing the total number of distributed goods to be at most the supply limit. However, we can not apply our rounding scheme directly to this altered LP: it does not satisfy supply constraints (even in expectation). We leave the problem of rounding this altered LP as an open question.

Theorem 10. *There is a $(1/4)$-approximation auction for the limited-supply setting.*

Proof. Suppose the auctioneer has k copies of the good. Compute a spanning tree of the social network and color the nodes red and blue such that each red node has a blue neighbor in the spanning tree (and vice versa). Pick a color uniformly at random and name the nodes of this color S_1 and nodes of the other color S_2. Allocate to the $k/2$ highest positive virtual values in S_1, and their neighbors in S_2 to ensure feasibility. We now compute the expected virtual value of this allocation for the red nodes. We condition on the event E that the red nodes were chosen to be set S_1.

$$\begin{aligned}
E_v[\sum_{i \text{ red}} x_i(v)\phi_i(v_i)] &= E_v[\sum_{i \text{ red}} x_i(v)\phi_i(v_i)|E]\Pr[E] \\
&\quad + E_v[\sum_{i \text{ red}} x_i(v)\phi_i(v_i)|\overline{E}]\Pr[\overline{E}] \\
&\geq \left(\frac{1}{2}\right) E_v[\sum_{i \text{ red}} x_i(v)\phi_i(v_i)|E] \\
&= \left(\frac{1}{2}\right) E_v[\max_{S\subseteq \text{ red}:|S|\leq k/2} \sum_{i\in S} \phi_i(v_i)]
\end{aligned}$$

where the second step follows since the expected allocation of any red node i is independent of its value conditioned on $\overline{E}$. Therefore by fact 1 each such vertex contributes a non-negative amount to the revenue. The third step follows since conditioned on E we picked the best set of size at most $k/2$ from the red nodes.

Now to prove the approximation guarantee first define $X = E[\max_{S:|S|\leq k} \sum_{i\in S} \phi(v_i)]$, and note this is an upper bound on the optimum revenue, since in the best case we can allocate the highest (positive) k virtual values. But we know that for any sampling of the values,

$$\begin{aligned}
\max_{S:|S|\leq k} \sum_{i\in S} \phi(v_i) &\leq \max_{S\subseteq \text{ red}:|S|\leq k} \sum_{i\in S} \phi(v_i) \\
&\quad + \max_{S\subseteq \text{ blue}:|S|\leq k} \sum_{i\in S} \phi(v_i) \\
&\leq 2\cdot(\max_{S\subseteq \text{ red}:|S|\leq k/2} \sum_{i\in S} \phi(v_i) \\
&\quad + \max_{S\subseteq\text{blue}:|S|\leq k/2} \sum_{i\in S} \phi(v_i)),
\end{aligned}$$

and therefore,

$$\begin{aligned}
X &= E[\max_{S:|S|\leq k} \sum_{i\in S} \phi(v_i)] \\
&\leq 2(E[\max_{S\subseteq S_1:|S|\leq k/2} \sum_{i\in S} \phi(v_i)] \\
&\quad + E[\max_{S\subseteq S_2:|S|\leq k/2} \sum_{i\in S} \phi(v_i)]),
\end{aligned}$$

by linearity of expectation. Recalling that the expected value of our allocation is at least

$$E[\max_{S \subseteq S_1 : |S| \le k/2} (\sum_{i \in S} \phi_i(v_i))],$$

and noting that we picked each of the two sets with probability $1/2$ to be S_1, we conclude that the expected revenue of our allocation (over the randomness of the algorithm and sampling of values), is at least $1/4$ of the upper bound. $\square$

C Threshold Strategies

In this section, we observe that no threshold strategy can have better than 0.5 approximation. Our example compares the value of all possible strategies to the optimum value, and therefore the result holds for any upper bound on optimum. Consider again the previous superstar and assume that p is very small. Consider the strategy in which we allocate both nodes of a pair when at least one of them has positive value. The value that we get is $p^2 + 2p(1 - 2p)$. Now consider any pricing strategy. The value we get when we set one of the thresholds equal to $\frac{-p}{1-p}$ is at most p. If we set both thresholds to p we get p^2. In any case, the ratio of optimum to any pricing strategy goes to 2 as $p \to 0$.