Information Gathering with Peers: Submodular Optimization with Peer-Prediction Constraints

Goran Radanovic
Harvard University
Cambridge, USA
gradanovic@g.harvard.edu

Adish Singla
MPI-SWS
Saarbrücken, Germany
adishs@mpi-sws.org

Andreas Krause
ETH Zurich
Zurich, Switzerland
krausea@ethz.ch

Boi Faltings
EPFL
Lausanne, Switzerland
boi.faltings@epfl.ch

Abstract

We study a problem of optimal information gathering from multiple data providers that need to be incentivized to provide accurate information. This problem arises in many real world applications that rely on crowdsourced data sets, but where the process of obtaining data is costly. A notable example of such a scenario is crowd sensing. To this end, we formulate the problem of optimal information gathering as maximization of a submodular function under a budget constraint, where the budget represents the total expected payment to data providers. Contrary to the existing approaches, we base our payments on incentives for accuracy and truthfulness, in particular, *peer-prediction* methods that score each of the selected data providers against its best peer, while ensuring that the minimum expected payment is above a given threshold. We first show that the problem at hand is hard to approximate within a constant factor that is not dependent on the properties of the payment function. However, for given topological and analytical properties of the instance, we construct two greedy algorithms, respectively called PPCGreedy and PPCGreedyIter, and establish theoretical bounds on their performance w.r.t. the optimal solution. Finally, we evaluate our methods using a realistic crowd sensing testbed.

Introduction

The recent success of various machine learning techniques can partly be attributed to the existence of large sets of labeled data that can readily be used for training purposes. In the past decade, the predominant form of obtaining useful data is through crowdsourcing approaches, where human subjects either directly label data or have private devices that provide measurements about spatially distributed phenomena.

One of the most important aspects of data is its accuracy, which can only be established if data providers (e.g. crowd-participants) report accurate information. To incentivize accurate reporting, a data collector can provide incentives that compensate data providers for their effort. In its simplest form, this type of data elicitation process can be modeled as a three step protocol:

- Data providers acquire accurate data experiencing a cost of effort;

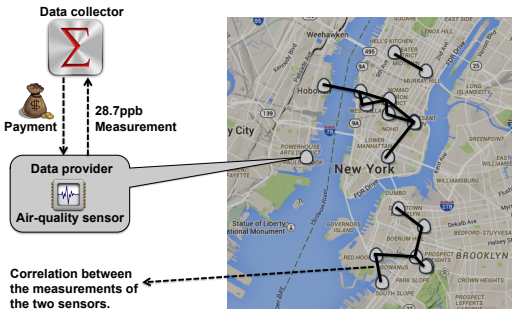


Figure 1: An example of crowdsourcing with incentives: Crowd-sensors (*air-quality eggs*[1]) report air-quality measurements and a data collector rewards them with monetary payments. An edge (black line) indicates that there is a sufficient correlation between the measurements of two sensors to verify accuracy.

- Data providers report the acquired data to a data collector;
- The data collector pays to the data providers a value that compensates for the cost of effort.

An example scenario is shown in Figure 1.

The problem, however, arises if the data providers are susceptible to moral hazard, that is, deviation to reporting *heuristically* without obtaining the data in the first place. In fact, such a behavior is expected for a rational participant who aims to maximize their utility, since heuristic reporting typically carries no cost of effort. To avoid this problem, the data collector can design payment functions τ that are dependent on the accuracy of the reported information, for example, by installing random spot-checks that validate some of the reports (Gao, Wright, and Leyton-Brown 2016). While this approach has often been used in standard micro-task crowdsourcing, it is often too costly to apply it in a more complex elicitation setting. Consider, for example, a crowd-sensing scenario shown in Figure 1, where sensors measure spatially distributed phenomenon that is highly localized. To apply a spot-checking procedure, the data collector would need to have mobile sensors that would change their locations at each time-step. Furthermore, due to the localized nature of the measured phenomenon, the density of the spot-check sensor network has to be relatively large.

[1] https://airqualityegg.wickeddevice.com

Instead of evaluating data providers against trusted reports, Dasgupta and Ghosh (2013), Jurca and Faltings (2011), Radanovic, Faltings, and Jurca (2016), Shnayder et al. (2016), Witkowski et al. (2017), and Baillon (2017), propose *peer-prediction* mechanisms for incentivizing distributed information sources. Peer-prediction mechanisms reward data providers by measuring consistency among their reports—thus, if a data provider believes that others are honest, she is also incentivized to report truthfully.[1] The most important condition to hold when applying a peer-prediction mechanism is that a data provider and her peer have correlated private information. Furthermore, this correlation, when expressed through expected payments, should be greater than the cost of effort. The latter property can always be achieved by scaling, provided that a considered peer-prediction method provides strict incentives for truthfulness. The scaling approach, however, neglects potential budget concerns that are important when collecting large data sets.

Overview of Our Approach

We, therefore, focus on the limited budget concern in a distributed data collection process that uses peer-prediction incentives. There are two important aspects to this problem:

- which data providers to select given that we only have a limited budget to spend on incentives—thus, only those data providers who received incentives can be considered to be reliable;
- how to ensure that all of the selected data providers have a proper peer—this constrains the selection problem to always include a proper peer of each data provider that is to be selected.

To quantify the usefulness of each data provider, we adopt a *submodular* utility function, which can, for example, measure the information gain of the data collector for obtaining the reports of the selected data-providers. We will insist that each data provider can be scored against a peer report with resulting expected payment being greater than a given threshold. Furthermore, the total expected payment should be bounded by a budget, while a data provider should be always scored against the best peer among the selected data providers. Our main contributions are:

- A formal model of information gathering with budget and peer-prediction constraints that is based on submodular maximization.
- Showing that the studied optimization problem is hard to approximate within a constant factor independent of the properties of the applied payment function.
- Novel algorithms for maximizing submodular functions with peer-prediction constraints that have provable guarantees for given topological and analytical properties of payments.
- Experimental evaluation of the proposed algorithms on a crowd-sensing test-bed.

Notice that we do not focus on a particular peer-prediction mechanism, but rather we allow a wide range of possible mechanisms (those that are robust in terms of the number of peers and produce bounded expected payments); thus, we complement the prior work on peer-predictions by examining orthogonal aspects of elicitation without direct verification. We provide the proofs to our formal claims in the extended version of the paper (Radanovic et al. 2017).

Problem Statement

We now formalize the problem addressed in this paper. We model data providers as nodes in a graph, whereas the underlying peer-prediction dependencies are modeled via edges whose weights are defined by the expected payments. The overall goal is to select a set of nodes that maximize a submodular utility function, while satisfying the constraint that the cost of the data collector (i.e., the total expected payment to nodes) is within a predefined budget. The following subsections provide more precise modeling details.

Set of Nodes and the Utility Function

We consider a set of nodes (e.g., a population of people or sensors deployed in a city) denoted by set $V = \{v_1, v_2, \ldots, v_{|V|}\}$, of size $|V|$. Hereafter, we denote a generic node by v. We associate a function over the set of nodes $f : 2^V \rightarrow \mathbb{R}_{\geq 0}$ that quantifies their utility (e.g., informativeness). That is, given a set of selected nodes S, the utility achieved from this set is equal to $f(S)$. Furthermore, for given set S, and a node $v \in V \setminus S$, we define the marginal utility of adding v to S as follows:

$$f(v|S) = f(S \cup \{v\}) - f(S). \tag{1}$$

Here, the function f is assumed to be *submodular* and *monotone*. Submodularity is an intuitive notion of diminishing returns, stating that, for any sets $S \subseteq S' \subseteq V$, and any given node $v \notin S'$, it holds that $f(v|S) \geq f(v|S')$. Monotonicity requires that the function f increases as we add more elements to set S. That is, for any sets $S \subseteq S' \subseteq V$, it holds that $f(S) \leq f(S')$. These conditions are relatively general, and are satisfied by many realistic, as well as complex utility functions for information gathering (Krause and Guestrin 2011; Krause and Golovin 2012; Singla and Krause 2013; Singla et al. 2014; Tschiatschek, Singla, and Krause 2017). W.l.o.g., we assume that function f is normalized, i.e., $f(\emptyset) = 0$.

Peer-prediction Constraints (PPC)

The nodes in general exhibit dependencies with other nodes. We consider a particular form of constraints that is associated with information elicitation via *peer-prediction mechanisms* (Miller, Resnick, and Zeckhauser 2005). A canonical peer-prediction $\tau : V \times V \rightarrow \mathbb{R}$ scores the information reported by node v using the information of a peer node v_p. Mechanism $\tau(v, v_p)$ is said to be *proper* if node v's best response to accurate reporting of node v_p is to report accurately, where the quality of a response is measured in terms

[1] Peer-predictions are in general susceptible to collusion, but in many cases one can establish relatively strong incentive properties for a wide variety of reporting strategies (Kong and Schoenebeck 2016). We do not focus on collusion resistance, so we use standard peer-predictions in our setting.

of node v's expected payoff over possible (accurate) reports of node v_p.[2] We denote node v's expected payoff for accurate reporting by $\mathbb{E}(\tau(v, v_p))$. To establish the properness of τ, one needs to ensure that peer v_p provides statistically correlated information to that of node v, so that the expected payoff $\mathbb{E}(\tau(v, v_p))$ is strictly greater than the cost of accurate reporting (which models, for example, participants' effort exertion). Therefore, node v has only a limited number of peers defined as nodes that lead to the expected payoff $\mathbb{E}(\tau(v, v_p)) \geq \tau_{min}$, where $\tau_{min} > 0$ is a problem specific threshold dependent on the cost of accurate reporting. We will further require that the same holds for node v's peers, i.e., $\mathbb{E}(\tau(v_p, v)) \geq \tau_{min}$, and assume that mechanism τ provides bounded payments, so that $\mathbb{E}(\tau(v, v_p)) \leq \tau_{max}$. Notice that as τ_{min} increases, a node is expected to have a smaller number of peers, which makes the problem of selecting an optimal set of nodes more constrained. In Section 'Experimental Evaluation', we confirm this observation by showing the drop in the obtained utility.

Example: Output Agreement (OA). Arguably the simplest peer prediction method is the output agreement of von Ahn and Dabbish (2004), which gives a strictly positive payment only for matching reports. In our experiments, reported information can in general take real values. In that case, as explained by Waggoner and Chen (2014), the OA mechanism can be defined as:

$$\tau(v, v_p) = 1 - d(v, v_p)^2, \tag{2}$$

where d is Euclidian distance between reported values of v and v_p. Note that more complex designs are also allowed by our framework, such as the one proposed in (Faltings, Li, and Jurca 2014). For more information on the properties of different minimal peer-predictions and their relationships, we refer the reader to (Frongillo and Witkowski 2016).

With this in mind, we can model the dependencies among nodes using an undirected graph $G = (V, E)$, where edges are defined as $E = \{\{v, u\} : v, u \in V, v \neq u, \mathbb{E}(\tau(v, u)) \geq \tau_{min}, \mathbb{E}(\tau(u, v)) \geq \tau_{min}\}$. We require that each node in a selected set S has a neighboring node G that is also in S, which implies that we can properly evaluate the reported information of each selected node. We denote the set of neighboring nodes to node v as $\mathcal{N}_v$, i.e., $\mathcal{N}_v = \{u \in V : \{v, u\} \in E\}$. W.l.o.g. we can assume that every node in G has at least one peer node v_p, i.e., $|\mathcal{N}_v| \geq 1$. Namely, nodes that do not have a peer cannot be incentivized to report accurately in our setting, so they bring 0 utility in terms of f. Finally, let us denote by ω the maximum number of peers that a node in graph G has, i.e., $\omega(G) = \max_{v \in V} |\mathcal{N}_v|$.

Cost of Incentivizing Accuracy

Given a selected set of nodes S, an information elicitation procedure needs to spend a certain amount of budget, hereafter denoted by B, to incentivizing accurate reporting. To quantify the cost of accurate elicitation, one needs to specify a peer selection procedure when a node v has multiple peers. We take the approach of selecting the best peer, that is the peer that has the most correlated information to that of the considered node according to the expected payoffs—this leads to the strongest incentives in terms of the separation between the expected payoffs for accurate and inaccurate reporting. With this choice of peer selection procedure, we can define the cost of selecting nodes S as function $c : 2^V_\phi \to \mathbb{R}$:

$$c(S) = \sum_{v \in S} \max_{v_p \in S \cap \mathcal{N}_v} \mathbb{E}(\tau(v, v_p)). \tag{3}$$

Here, 2^V_ϕ contains only sets S such that each node $v \in S$ has a peer node $v_p \in S$, which makes c well defined.

Optimization Problem

Our goal is to select a *feasible* set $S \in 2^V_\phi$ that maximizes the utility $f(S)$ given budget B, i.e., $c(S) \leq B$. More precisely, the budget denotes the total expected payment that a data collector is willing to provide for incentivizing accurate reporting. We therefore pose the following optimization problem:

$$S^* = \operatorname*{arg\,max}_{S \in 2^V_\phi \text{ s.t. } c(S) \leq B} f(S). \tag{4}$$

Ignoring the computational constraints, we denote the optimal solution to this problem as OPT.

Methodology

Instead of operating directly on optimization problem (4), we reformulate it so that the budget constraint is expressed through a cost function defined over all subsets of nodes 2^V, not just the feasible set 2^V_ϕ. We first show that the new optimization problem is equivalent to (4). Unfortunately, it is hard to approximate without any dependency on the structure of the cost function. We then relax it to an optimization problem that uses a modular approximation to the cost function, but operates with reduced budget to satisfy the original budget constraint. This relaxation is the basis for our algorithms developed in the next section, and is sound if the cost function of the original problem has certain topological and analytical constraints. The following subsections explain our methodology in more details.

Expansion of the Cost Function

We start by expanding the domain of cost function c to power-set 2^V, which will provide us with better insights on the computational complexity of the original problem. In particular, we consider the following expansion $c_e : 2^V \to \mathbb{R}$:

$$c_e(S) = c(S_p) + \sum_{v \in S \setminus S_p} \min_{v_p \in V \cap \mathcal{N}_v} \mathbb{E}(\tau(v, v_p)), \tag{5}$$

where S_p is a set of all nodes in S who also have a peer in S, i.e., $S_p = \{v \in S : \exists v_p \in S \text{ s.t. } v_p \in \mathcal{N}_v\}$. In other words, cost function c_e acts as if all the nodes in S who have a peer in S are rewarded as usual, while those that do not have a peer in S are rewarded with the expected payoff they obtain when scored against the worst peer. Notice that $c_e(S) = c(S)$ for all $S \in 2^V_\phi$, which makes the expansion sound. We

[2] Therefore, *properness* is here defined as *Bayes-Nash incentive compatibility* in game-theoretic sense.

denote by $c_e(v|S)$ the marginal increase of cost c_e for adding an element v to S, i.e., $c_e(v|S) = c_e(S \cup \{v\}) - c_e(S)$. We establish the monotonicity of cost function c_e with the following lemma; notice, however, that the cost function is not necessarily sub/super-modular.

Lemma 1. *Cost function c_e defined by* (5) *is monotone. Furthermore: $c_e(\{v\}) \leq c_e(v|S)$, for all $S \in 2^V \backslash \{v\}$.*

Hardness Result

To prove the complexity of our initial problem, we adapt optimization problem (4) to use the extended cost function c_e. In particular, we consider the optimization problem defined as:

$$S^* = \underset{S \in 2^V_\phi \text{ s.t. } c_e(S) \leq B}{\arg\max} f(S) \tag{6}$$

Clearly, any feasible solution to the problem (6) is also a feasible solution to the original problem due to the constraint $S \in 2^V_\phi$, while the optimality alignment is ensured by having the same objective value. Now, to show the hardness result for approximating OPT, we reduce the maximum clique problem to optimization problem (6) in a computationally efficient way, thus, obtaining:

Theorem 1. *For any $\epsilon > 0$, it is NP-hard to find a solution S to optimization problem* (6) *(and thus* (4)*) such that $\frac{f(S)}{f(\text{OPT})} \geq \frac{1}{|V|^{1-\epsilon}}$.*

Proof. Consider an arbitrary undirected unweighted graph $G' = (V, E')$ for which we wish to compute the maximum clique. To reduce the maximum clique problem to (6): 1) define function f as $f(S) = |S|$, which is clearly monotone and submodular; 2) define payment function as: $\tau(v, v_p) = \tau(v_p, v) = \tau_{max}$ if $(v, v_p) \notin E'$, and $\tau(v, v_p) = \tau(v_p, v) = \tau_{min}$ otherwise; 3) set budget B to $B = |V| \cdot \tau_{min}$; 4) and set $\tau_{max} > B$. Notice that such an arrangement induces a fully connected graph G. Furthermore, we defined deterministic payment functions $\tau(v, v_p)$ and $\tau(v_p, v)$, but one can use $\mathbb{E}(\tau(v, v_p))$ and $\mathbb{E}(\tau(v_p, v))$ instead. Points 2 and 4 ensure that any solution to optimization problem (6) is a clique in graph G'; otherwise, the budget constraints would be violated in solving (6). Likewise, points 2 and 3 ensure that any clique is permitted as a potential solution w.r.t. the budget constraint. Finally, point 1 ensures that we search for a clique with the maximum number of vertices. Since the reduction is computationally efficient (polynomial in the graph size, i.e. $|V|$ and E), optimization (4) is at least as hard as the maximum clique problem. Using the fact that the maximum clique problem is hard to approximate within factor $\frac{1}{|V|^{1-\epsilon}}$ (Hastad 1999), we obtain the claim. □

Structural Properties of the Cost Function

To cope with the computational hardness of the problem at hand, we identify two structural properties of cost function c_e (or equivalently, the structural properties of payment function τ). The first one is related to *topological* properties of graph G, and can be quantified with the maximum number of peers that a node in the graph can have. As explained earlier in this section, we denote this number by ω.

The second property is similar to the notion of *curvature* of a submodular function (Iyer, Jegelka, and Bilmes 2013), but now defined over cost function c_e that is not necessarily sub/super-modular. In particular, we define the *slope* α of cost function c_e as:

$$\alpha = 1 - \min_{v \in V, S \in 2^V \backslash \{v\}} \frac{c_e(\{v\})}{c_e(v|S)}.$$

The slope of cost function c_e, as defined above, measures how much marginal gains of c_e change as we add more to initially empty set of selected nodes.[3] Intuitively, it measures the deviation of c_e from modularity. A specific case of our interest is when $\alpha = 0$, which indicates that c_m is modular and, thus, can be decomposed into a sum of costs c_m dependent only on one vertex, i.e., $\sum_{v \in S} c_m(v)$. In the next subsection, we discuss how to utilize modular approximations of c_e when c_e itself is not modular. First, let us upper bound the slope α using the fact that payments are bounded.

Lemma 2. *The slope of cost function c_e is upper-bounded by $\alpha \leq 1 - \frac{\tau_{min}}{\omega \cdot \tau_{max}}$.*

Relaxed Optimization Problem

To make use of the structural constraints of cost function c_e, let us consider a relaxed version of optimization problem (6) with budget constraints defined via a *modular lower bound* to cost function c_e, denoted by c_M. As we show in the next section, for such a relaxation, one can develop a greedy approach that has provable approximation guarantees on the quality of the obtained solution relative to OPT. More precisely, consider the modular function $c_M : 2^V \to \mathbb{R}$ defined via a cost function $c_m : V \to \mathbb{R}$:

$$c_M(S) = \sum_{v \in S} c_m(v) \tag{7}$$
$$\text{where } c_m(v) = \min_{v_p \in \mathcal{N}_v} \mathbb{E}(\tau(v, v_p)).$$

Clearly, $c_M(S)$ lower bounds $c_e(S)$ as it calculates the expected payoffs of nodes in S when they are scored against their worst peers (not necessarily in S). Now, we relax optimization problem (6) to:

$$S^* = \underset{S \in 2^V_\phi \text{ s.t. } c_M(S) \leq B'}{\arg\max} f(S). \tag{8}$$

In order to make the relaxation sound, any selected set S^* in problem (8) should also be feasible in problem (6) (and thus (4)). We can ensure this by reducing the available budget, i.e., by making B' appropriately smaller than B. Using the slope of cost c_e, we can obtain that the following budget reduction satisfies our requirement.

Lemma 3. *Any feasible solution S to optimization problem* (8) *is also a feasible solution to optimization problem* (6) *(and thus* (4)*) for $B' \leq (1 - \alpha) \cdot B$, where α is the slope of cost function c_e.*

[3] That is, α quantifies the maximum increase in c_e for adding a node v (see Lemma 1).

Algorithm 1: Algorithm PPCGREEDY

Input:
- PPC graph: $G(V, E)$;
- Utility function : f; budget B;
- Cost function : c, slope α, modular approx. c_M;

Output: selected set $\widetilde{S^*}$;
Initialize:
- $t = 0$; $\widetilde{S^*} = \emptyset$; budget $B^t = (1-\alpha) \cdot B$;
- COUPLESUPERSET $\mathcal{Z} = \emptyset$;

//Create COUPLESUPERSET $\mathcal{Z}$
foreach $v \in V$ **do**
  $\mathcal{N}_v \leftarrow \{u : \{v, u\} \in E\}$;
  foreach $u \in \mathcal{N}_v$ **do**
    $z = \{v, u\}$; $\mathcal{Z} = \mathcal{Z} \cup \{z\}$;
  end
end
//Compute $\widetilde{S^*}$
while $B^t > 0$ **do**
  $z_t^* = \arg\max_{z \in \mathcal{Z}, z \setminus \widetilde{S^*} \neq \emptyset, c_M(z \setminus \widetilde{S^*}) \leq B^t} \frac{f(z|\widetilde{S^*})}{c_M(z \setminus \widetilde{S^*})}$;
  if $z_t^* = NULL$ **then**
    break;
  end
  $B^{t+1} = B^t - c_M(z_t^* \setminus \widetilde{S^*})$;
  $\widetilde{S^*} = \widetilde{S^*} \cup z_t^*$;
  $t = t + 1$;
end
Output: $\widetilde{S^*}$

Algorithm

We now present a new greedy algorithm for solving the optimization problem with peer-prediction constraints (PPC), called PPCGreedy (Algorithm 1). It is similar to standard greedy approaches for submodular maximization with budget constraints (e.g., Nushi et al. (2015)), but it additionally ensures that a tentative output $\widetilde{S^*}$ at a certain iteration is an element of 2^V_ϕ. To do so, it initially constructs a set of couples $\mathcal{Z}$ that contains all the peer pairs and selects at each iteration t either a node that already has a peer in the selected set or a pair of nodes that are peers. The selection procedure makes a choice z^* that maximizes the ratio between the utility gain and the cost increase, while not exceeding a given budget $B' = (1-\alpha) \cdot B$. If there are multiple choices that maximize this ratio, the selection procedure selects one of them, whereas if there is no choice that fits the budgets constraints, z^* is set to $NULL$, which ends the search and outputs the current solution $\widetilde{S^*}$.

Analysis

We will now show the main property of our algorithm: its near optimality when cost function c_e has a low slope α, i.e., when the difference between τ_{max} and τ_{min} is small. Notice that parameters τ_{max} and τ_{min} are controllable through our design of a peer-prediction method τ and the requirements on minimal expected payments, which implies that α can be tuned. For all practical reasons, it is also reasonable to assume that $\frac{(1-\alpha)\cdot B}{\tau_{max}} > 2$, which simply states that our algorithm is always able to initially select any pair of nodes.

Theorem 2. *Let the maximal relative difference between modular costs of two peer nodes be bounded by r, i.e., $r \geq \max_{v \in V, v_p \in \mathcal{N}_v} \frac{c_m(v)}{c_m(v_p)}$, and let $\gamma = \max_{v \in V} \frac{c_m(v)}{B'} \in (0, \frac{1}{2})$. Then, the output $\widetilde{S^*}$ of Algorithm 1 has the following guarantees on the utility:*

$$f(\widetilde{S^*}) \geq \left(1 - e^{-\frac{(1-\alpha)\cdot(1-2\cdot\gamma)}{1+r}}\right) \cdot f(\text{OPT}). \quad (9)$$

Proof (Sketch). The proof of the theorem is non-trivial, so we outline only its basic steps (see (Radanovic et al. 2017) for more details). Using the fact that f is submodular, while Algorithm 1 is greedy in terms of f/c_M ratio, we show that:

$$f(z_t^*|S_t) \geq \frac{c_M(z_t^* \setminus S_t)}{(1+r) \cdot B} \cdot [f(\bar{\text{OPT}}) - f(S_t)],$$

where S_t is equal to $\widetilde{S^*}$ at time-step t, while $\bar{\text{OPT}}$ is the optimum solution to optimization problem (8) when budget $B' = B$. Now, following the the proofs of related results for submodular maximization under budget constraints (e.g., Sviridenko (2004), Nushi et al. (2015)), and adapting them to our setting, we obtain that:

$$f(\widetilde{S^*}) \geq \left(1 - e^{-\frac{(1-\alpha)\cdot(1-2\cdot\gamma)}{1+r}}\right) f(\bar{\text{OPT}}).$$

As we argue in the full proof, $f(\bar{\text{OPT}}) \geq f(\text{OPT})$ because $\bar{\text{OPT}}$ is obtained for the same budget as OPT, but the cost c_M that lower bounds c. Together with the above inequality, this implies the statement of the theorem. □

We see that the quality of the approximation ratio depends on the structural properties of the cost function, including slope α, the maximum cost discrepancy between two nodes measured by r, and the maximum fraction of the budget assigned to a node, measured by γ. As α approaches its maximum value, i.e., $\alpha \to 1$, the approximation ration goes to 0. This is consistent with the hardness result presented in Section 'Methodology', which shows the necessity of imposing structural constraints. One can reach a similar conclusion by analyzing r as it goes to its maximal value, i.e., $r \to \infty$.

To see this more clearly, we can express the results of the theorem in terms of the original optimization problem and the structural properties of payment function τ. Using the bound on slope α (Lemma 2), the boundedness of payments, which imply $r \leq \frac{\tau_{max}}{\tau_{min}}$, we obtain:

Corollary 1. *Assuming $B > 2 \cdot \frac{\omega \cdot \tau_{max}^2}{\tau_{min}}$, the output $\widetilde{S^*}$ of Algorithm 1 has the following guarantees on the utility:*

$$f(\widetilde{S^*}) \geq \left(1 - e^{-\frac{\tau_{min}^2}{\omega \tau_{max}^2} \cdot \left(\frac{1}{2} - \frac{\omega \cdot \tau_{max}^2}{B \cdot \tau_{min}}\right)}\right) \cdot f(\text{OPT}). \quad (10)$$

Therefore, whenever the maximum payment τ_{max} or the number of possible peers ω go to large values, the approximation factor becomes negligible. Notice that the number of

Algorithm 2: Algorithm PPCGREEDYITER

Output: selected set $\widetilde{S^*}$;
Initialize:
□ $t = 1$; $\widetilde{S^*} = \emptyset$; budget $B^0 = B + \epsilon$; $B^1 = B$;
//Compute $\widetilde{S^*}$
while $B^t < B^{t-1}$ **do**
  $\widetilde{S^*}_t = PPCGreedy(f(\cdot \cup \widetilde{S^*}), c(\cdot \cup \widetilde{S^*}), B^t, \widetilde{S^*})$;
  $B^{t+1} = B - c(\widetilde{S^*}_t \cup \widetilde{S^*})$;
  $\widetilde{S^*} = \widetilde{S^*} \cup \widetilde{S^*}_t$;
  $t = t + 1$;
end
Output: $\widetilde{S^*}$

possible peers ω is dependent on τ_{min}, so we can alternatively say that for small values of τ_{min}, i.e., $\tau_{min} \approx 0$, the quality of the obtained greedy solution is relatively low. In practice, however, we can often avoid these corner cases by adjusting the payment function, and thus τ_{max} and τ_{min}.

More Efficient Budget Expenditure

The PPCGreedy algorithm, as described by Algorithm 1 does not necessarily spend the full budget on incentivizing nodes. This is because we use a reduced budget B' when running the main steps of the algorithm. One can achieve a better budget efficiency by iteratively calling PPCGreedy method, as shown in Algorithm 2, that we refer to as PPCGreedyIter. It is important to note that in the sub-procedure $PPCGreedy$ we take into account the current set of selected nodes $\widetilde{S^*}$ when examining the feasibility of a solution and evaluating the utility and cost functions. The budget reduction in the $PPCGreedy$ subroutine can, on the other hand, be done with the same (initial) α. The procedure terminates when no new node is added, which is equivalent to the budget not changing between two consecutive iterations.

The utility function f is always evaluated with the selected set of nodes $\widetilde{S^*}$ from previous iterations, in the algorithm denoted by $f(\cdot \cup \widetilde{S^*})$. The same is true for cost function c, denoted by $c(\cdot \cup \widetilde{S^*})$, and its modular approximation c_M. Due to monotonicity of f, this means that the reached solution is always as good as the one obtained by PPCGreedy. Furthermore, the cost of the solution is within the budget constraints: this is because $c_e(\{v\}) \leq c_e(v|S)$ (Lemma 1), so the slope α defined on $c_e(\cdot)$ upper bounds the one defined on $c(\cdot \cup \widetilde{S^*})$, which implies that the subroutine $PPCGreedy$ makes a proper budget reduction. Therefore, the results of Theorem 2 and Corollary 1 are preserved.

Experimental Evaluation

To evaluate our approach, we use a crowd sensing test-bed of Singla (2017), constructed from real measurements of CO_2 and user locations across an urban area. The concentrations of CO_2 in the city of Zurich were acquired with a *NODE+*[4] sensor. These measurements were used to fit a Gaussian variogram whose parameters indicate that the relevant correlation range between two measurement locations is about $R = 236$ meters. We use this distance to define a *disk* coverage function—for a set of points of interest, we count how many of these are within R meters away from the set of selected points. More formally, given a set of points S that represent the location of the selected sensors and set of points S_{poi} that represent locations for which we would like to obtain CO_2 measurements, the objective function f is defined as: $f(S) = \sum_{s \in S_{poi}} \mathbb{1}_{min_{s' \in S} d(s,s') \leq R}$. Here, $\mathbb{1}_{cond}$ is an indicator variable, evaluating to 1 when $cond$ is satisfied, and is 0 otherwise, while $d(s, s')$ measures the distance in meters between locations s and s'. The function f is a coverage function, which is monotone and submodular (Krause and Golovin 2012).

Points of interest S_{poi} are predefined, and in total, there are 300 of them. These were obtained using a publicly available data (*OpenStreetMap*[5]), from which we randomly selected 300 locations from an area in the center of New York City. To identify the locations of available crowd-sensors, i.e., the ground set V, we use the population statistics of the test-bed, which give us the likelihood of a user appearing in one of the 300 points. This statistics is inferred from a publicly accessible dataset (*Strava*[6]) that contains the mobility patterns of cyclists for a period of 6 days. We sample from the likelihood 1000 points to obtain sensing locations and then we perturb them by 50 meters.

As a peer prediction scoring rule, we use the output agreement mechanism as described in Section *Problem Statement*. The expected score of this mechanism for two points s and s' in truthful reporting regime is equal to $\mathbb{E}(d(s, s')^2)$. We approximate the expected value of OA for two sensors s and s' by using the variogram of the test-bed. More precisely, given the range parameter $R = 236$, we estimate the expected payoff between two sensors as: $\mathbb{E}(\tau(s_1, s_2)) = 1 - e^{-\frac{d(s_1,s_2)^2}{a \cdot R^2}}$, where a is set to $\frac{1}{3}$.

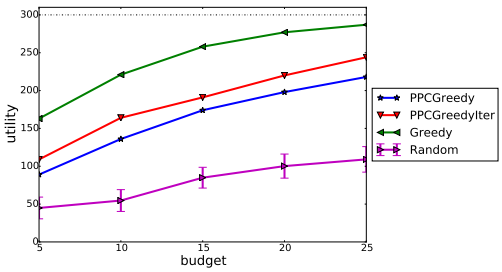
Results. We test our approaches, PPCGreedy and PPCGreedyIter, against two other baselines: (a) a random selection (denoted by *Random*) that satisfies peer-prediction constraints, (b) a greedy approach (denoted by *Greedy*) that assumes it suffices to reward each sensor with τ_{min}, without providing incentives for accurate reporting. Clearly, the latter baseline represents an optimistic approach whose performance upper bounds that of the proposed algorithms, while the former one is likely to lower bound their performance. In all the cases, the expected budget is at most B.

We perform two different tests. In the first test, we vary the total available budget B from 5 to 25 at steps of size 5. At the same time, we keep the minimal expected payment to $\tau_{min} = 0.5$. As we can see from Figure 2a, as the total budget increases, all the methods perform better. However, the increase is more notable for non-random algorithms. The performance of PPCGreedyIter is generally better than the one of PPCGreedy, and this is due to the spent budget —

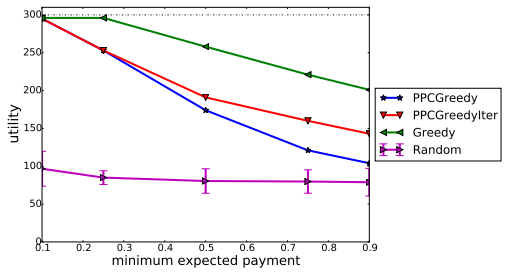
[4] http://www.variableinc.com/node1

[5] http://wiki.openstreetmap.org/wiki/Node

[6] http://metro.strava.com/



(a) Performance as we vary the available budget.



(b) Performance as we vary the minimum payments.

Figure 2: Experimental results show how the utility $f(\widetilde{S^*})$ changes as we increase budget B or the minimum expected payments τ_{min}. $\widetilde{S^*}$ is the output of different methods. Increasing B is beneficial as it allows more sensors to be selected. On the other hand, as we increase τ_{min}, the sensors have less peers and they need to be payed more. *Random* is run 10 times and we show the means and standard deviations.

as explained in the previous section, PPCGreedy is only the first step of PPCGreedyIter, which further iteratively runs PPCGreedy on the remaining budget.

In the second test, we vary the minimal expected payment τ_{min}, which now takes values in $\{0.1, 0.25, 0.5, 0.75, 0.9\}$. Budget B is set to 15. The results are given in Figure 2b. Except for Random, a general trend is that increasing the minimal payments leads to lower performance, which is not surprising given that the number of peers and the budget per sensor decrease in that case. Random in general performs much worse than other techniques for all values of τ_{min}. Moreover, notice that the discrepancy in performance between PPCGreedy, PPCGreedyIter, and Greedy, increases with τ_{min}. Initially, all the non-random algorithms find an optimum of function f, achieving the utility equal to 300.

Related Work

Information elicitation. A standard incentives for quality are typically categorized into gold standard techniques, such as proper scoring rules (Gneiting and Raftery 2007) or prediction markets (Chen and Pennock 2007), or peer-prediction techniques, such as the classical peer-prediction (Miller, Resnick, and Zeckhauser 2005) or Bayesian truth serums (Prelec 2004; Witkowski and Parkes 2012; Radanovic and Faltings 2013). We focus in this paper on peer-prediction techniques due to the scalability of elicitation without verification for acquiring large amounts of highly distributed information. Recently, several peer-predictions were proposed for various crowdsourcing scenarios. These included micro-task crowdsourcing (Dasgupta and Ghosh 2013), opinion polling (Jurca and Faltings 2011), information markets (Baillon 2017), peer grading (Shnayder et al. 2016), and most importantly for this work, crowdsensing (Radanovic, Faltings, and Jurca 2016). The proposed mechanisms for these domains follow the standard principles of the classical peer-prediction, e.g., incentivizing participants by comparing their reports and placing higher scores for a priori less likely matches. However, they also often extend the design of the original methods by making them more robust in terms of the required number of participants and the knowledge about them, the heterogeneity of users and tasks, or susceptibility to collusive behaviors (Faltings and Radanovic 2017). We analyze orthogonal characteristics important for deploying such mechanisms in practice, i.e., budget and cost acquisition constraints. Although the prior work (e.g., Liu and Chen (2016)) does study meta-mechanisms that make peer-predictions proper in terms of effort exertion, it is often based on scaling techniques, which either ignore budget limitations or the cost of effort.

Submodular function maximization. From the technical side, the most important aspect relates to submodular function maximization. While there is a sizeable literature on this topic (e.g., Krause and Guestrin (2011), Krause and Golovin (2012)), we mostly focus on the prior work that is closely related to the techniques used in this paper. Our basic objective is a subset selection under budget (knapsack) constraints (e.g., Sviridenko (2004)), and we base our algorithmic techniques on a simple greedy approach (Nushi et al. 2015). Notice that we additionally have a graph based constraint, which is in spirit similar to Singla et al. (2015), although we are solving a different optimization problem. Arguably, this paper is most related to submodular maximization with submodular budget constraints (Iyer and Bilmes 2013); contrary to this work, our budget constraints are not necessarily sub/super-modular. It is also worth mentioning the hardness results that relate to the ones obtained in this paper, such as the inapproximability of the maximum of a submodular non-monotone, possibly negative, profit function (Feige et al. 2008).

Conclusion

In this paper, we have introduced an information elicitation model for data collection from distributed sources when the incentive mechanism is based on peer-predictions. We have shown that optimal information gathering is computationally infeasible in that even approximating the optimal solution is NP-hard. However, given structural constraints on peer-prediction incentives, we have proposed two greedy methods that achieve good performance relative to the optimum, and have tested their performance empirically on a realistic crowd-sensing test-best.

Acknowledgments This work was supported in part by the Swiss National Science Foundation, Nano-Tera.ch program as part of the Opensense II project, ERC StG 307036, a SNSF Early Postdoc Mobility fellowship, and a Facebook Graduate fellowship.

References

Baillon, A. 2017. Bayesian markets to elicit private information. *Proceedings of the National Academy of Sciences* 114(30):7958–7962.

Chen, Y., and Pennock, D. M. 2007. A utility framework for bounded-loss market makers. In *Proceedings of the Twenty-Third Conference on Uncertainty in Artificial Intelligence*.

Dasgupta, A., and Ghosh, A. 2013. Crowdsourced judgement elicitation with endogenous proficiency. In *Proceedings of the 22nd ACM International World Wide Web Conference*.

Faltings, B., and Radanovic, G. 2017. *Game Theory for Data Science: Eliciting Truthful Information*. Morgan & Claypool Publishers.

Faltings, B.; Li, J. J.; and Jurca, R. 2014. Incentive Mechanisms for Community Sensing. *IEEE Transaction on Computers* 63(1):115–128.

Feige, U.; Immorlica, N.; Mirrokni, V.; and Nazerzadeh, H. 2008. A combinatorial allocation mechanism with penalties for banner advertising. In *Proceedings of the 17th International Conference on World Wide Web*.

Frongillo, R., and Witkowski, J. 2016. A geometric method to construct minimal peer prediction mechanisms. In *Proceedings of the 30th AAAI Conference on AI*.

Gao, A.; Wright, J. R.; and Leyton-Brown, K. 2016. Incentivizing evaluation via limited access to ground truth: Peer-prediction makes things worse. *CoRR* abs/1606.07042.

Gneiting, T., and Raftery, A. E. 2007. Strictly proper scoring rules, prediction, and estimation. *Journal of the American Statistical Association* 102:359–378.

Hastad, J. 1999. Clique is hard to approximate within $n^{1-\epsilon}$. *Acta Math.* 182(1):105–142.

Iyer, R., and Bilmes, J. 2013. Submodular optimization with submodular cover and submodular knapsack constraints. In *Proceedings of the 26th International Conference on Neural Information Processing Systems*.

Iyer, R.; Jegelka, S.; and Bilmes, J. 2013. Curvature and optimal algorithms for learning and minimizing submodular functions. In *Proceedings of the 26th International Conference on Neural Information Processing Systems*.

Jurca, R., and Faltings, B. 2011. Incentives for answering hypothetical questions. In *Workshop on Social Computing and User Generated Content*.

Kong, Y., and Schoenebeck, G. 2016. A framework for designing information elicitation mechanisms that reward truth-telling. *CoRR* abs/1603.07751.

Krause, A., and Golovin, D. 2012. Submodular function maximization. *Tractability: Practical Approaches to Hard Problems* 3:19.

Krause, A., and Guestrin, C. 2011. Submodularity and its applications in optimized information gathering. *ACM Transactions on Intelligent Systems and Technology* 2(4):32.

Liu, Y., and Chen, Y. 2016. Learning to incentivize: Eliciting effort via output agreement. In *Proceedings of the 25th International Joint Conference on Artificial Intelligence*.

Miller, N.; Resnick, P.; and Zeckhauser, R. 2005. Eliciting informative feedback: The peer-prediction method. *Management Science* 51:1359–1373.

Nushi, B.; Singla, A.; Gruenheid, A.; Zamanian, E.; Krause, A.; and Kossmann, D. 2015. Crowd access path optimization: Diversity matters. In *AAAI Conference on Human Computation and Crowdsourcing*.

Prelec, D. 2004. A bayesian truth serum for subjective data. *Science* 34(5695):462–466.

Radanovic, G., and Faltings, B. 2013. A robust bayesian truth serum for non-binary signals. In *Proceedings of the 27th AAAI Conference on Artificial Intelligence*.

Radanovic, G.; Singla, A.; Krause, A.; and Faltings, B. 2017. Information gathering with peers: Submodular optimization with peer-prediction constraints (extended version). *CoRR* abs/1711.06740.

Radanovic, G.; Faltings, B.; and Jurca, R. 2016. Incentives for effort in crowdsourcing using the peer truth serum. *ACM Transactions on Intelligent Systems and Technology* 7:48:1–48:28.

Shnayder, V.; Agarwal, A.; Frongillo, R.; and Parkes, D. C. 2016. Informed truthfulness in multi-task peer prediction. In *Proceedings of the 2016 ACM Conference on Economics and Computation*.

Singla, A., and Krause, A. 2013. Incentives for privacy tradeoff in community sensing. In *AAAI Conference on Human Computation and Crowdsourcing (HCOMP)*.

Singla, A.; Horvitz, E.; Kamar, E.; and White, R. W. 2014. Stochastic privacy. In *Proc. Conference on Artificial Intelligence (AAAI)*.

Singla, A.; Horvitz, E.; Kohli, P.; White, R.; and Krause, A. 2015. Information gathering in networks via active exploration. In *IJCAI*.

Singla, A. 2017. *Learning and Incentives in Crowd-Powered Systems*. Ph.D. Dissertation, ETH.

Sviridenko, M. 2004. A note on maximizing a submodular set function subject to a knapsack constraint. *Oper. Res. Lett.* 32(1):41–43.

Tschiatschek, S.; Singla, A.; and Krause, A. 2017. Selecting sequences of items via submodular maximization. In *Proceedings of the 31th AAAI Conference on Artificial Intelligence (AAAI'17)*.

von Ahn, L., and Dabbish, L. 2004. Labeling images with a computer game. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*.

Waggoner, B., and Chen, Y. 2014. Output agreement mechanisms and common knowledge. In *Proceedings of the Second AAAI Conference on Human Computation and Crowdsourcing*.

Witkowski, J., and Parkes, D. C. 2012. A robust bayesian truth serum for small populations. In *Proceedings of the 26th AAAI Conference on Artificial Intelligence*.

Witkowski, J.; Atanasov, P.; Ungar, L. H.; and Krause, A. 2017. Proper proxy scoring rules. In *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence*.

Appendix: Information Gathering with Peers

Proof of Lemma 1

Statement: *Cost function c_e defined by* (5) *is monotone. Furthermore: $c_e(\{v\}) \leq c_e(v|S)$, for all $S \in 2^V \setminus \{v\}$.*

Proof. Denote by S_p^v a set of nodes in V who have peers in $S \cup \{v\}$. Notice that $S_p \subseteq S_p^v$. For a given set $S \in 2^V$ and $v \notin S$, we have:

$$
\begin{aligned}
&c_e(S \cup \{v\}) = c(S_p^v) \\
&+ \sum_{v' \in (S \cup \{v\}) \setminus S_p^v} \min_{v_p \in V \cap \mathcal{N}_{v'}} \mathbb{E}(\tau(v', v_p)) \\
&= \sum_{v' \in (S \cup \{v\}) \cap S_p^v} \max_{v_p \in (S \cup \{v\}) \cap \mathcal{N}_{v'}} \mathbb{E}(\tau(v', v_p)) \\
&+ \sum_{v' \in (S \cup \{v\}) \setminus S_p^v} \min_{v_p \in V \cap \mathcal{N}_{v'}} \mathbb{E}(\tau(v', v_p)) \\
&\geq \sum_{v' \in S \cap S_p} \max_{v_p \in S \cap \mathcal{N}_{v'}} \mathbb{E}(\tau(v', v_p)) \\
&+ \sum_{v' \in (S \cap S_p^v) \setminus S_p} \min_{v_p \in V \cap \mathcal{N}_{v'}} \mathbb{E}(\tau(v', v_p)) \\
&+ \min_{v_p \in V \cap \mathcal{N}_v} \mathbb{E}(\tau(v, v_p)) \\
&+ \sum_{v' \in S \setminus S_p^v} \min_{v_p \in V \cap \mathcal{N}_{v'}} \mathbb{E}(\tau(v', v_p)) \\
&= \sum_{v' \in S \cap S_p} \max_{v_p \in S \cap \mathcal{N}_{v'}} \mathbb{E}(\tau(v', v_p)) \\
&+ \sum_{v' \in S \setminus S_p} \min_{v_p \in V \cap \mathcal{N}_{v'}} \mathbb{E}(\tau(v', v_p)) \\
&+ \min_{v_p \in V \cap \mathcal{N}_v} \mathbb{E}(\tau(v, v_p)) \\
&\geq c(S_p) + \sum_{v' \in S \setminus S_p} \min_{v_p \in V \cap \mathcal{N}_{v'}} \mathbb{E}(\tau(v', v_p)) \\
&+ \tau_{min} = c_e(S) + \tau_{min} > c_e(S).
\end{aligned}
$$

Furthermore, from the proof we see that $c_e(v|S) \geq \min_{v_p \in V \cap \mathcal{N}_v} \tau(v, v_p) = c_e(v)$, thus proving the second part of the statement. □

Proof of Theorem 1

Statement: *For any $\epsilon > 0$, it is NP-hard to find a solution S to optimization problem* (6) *(and thus* (4)*) such that $\frac{f(S)}{f(\text{OPT})} \geq \frac{1}{|V|^{1-\epsilon}}$.*

Proof. We prove the statement by reducing the maximum clique problem to optimization (6). Consider an arbitrary undirected unweighted graph $G' = (V, E')$ for which we wish to compute the maximum clique. To reduce the maximum clique problem to (6):

1. define function f as $f(S) = |S|$, which is clearly monotone and submodular;
2. define payment function as: $\tau(v, v_p) = \tau(v_p, v) = \tau_{max}$ if $(v, v_p) \notin E'$, and $\tau(v, v_p) = \tau(v_p, v) = \tau_{min}$ otherwise;
3. set budget B to $B = |V| \cdot \tau_{min}$;
4. and set $\tau_{max} > B$.

Notice that such an arrangement induces a fully connected graph G. Furthermore, we defined deterministic payment functions $\tau(v, v_p)$ and $\tau(v_p, v)$, but one can use $\mathbb{E}(\tau(v, v_p))$ and $\mathbb{E}(\tau(v_p, v))$ instead. Points 2 and 4 ensure that any solution to optimization problem (6) is a clique in graph G'; otherwise, the budget constraints would be violated in solving (6). Likewise, points 2 and 3 ensure that any clique is permitted as a potential solution w.r.t. the budget constraint. Finally, point 1 ensures that we search for a clique with the maximum number of vertices. Since the reduction is computationally efficient (polynomial in the graph size, i.e. $|V|$ and E), optimization (4) is at least as hard as the maximum clique problem. Using the fact that the maximum clique problem is hard to approximate within factor $\frac{1}{|V|^{1-\epsilon}}$ (Hastad 1999), we obtain the claim. □

Proof of Lemma 2

Statement: *The slope of cost function c_e is upper-bounded by:*

$$\alpha \leq 1 - \frac{\tau_{min}}{\omega \cdot \tau_{max}}.$$

Proof. Notice that the slope is maximized when there exist a node v such that:

- the expected payoff of v when it's scored against its worst peer $v_{p,min}$ is τ_{min},
- the expected payoff of v increases to τ_{max} whenever any other peer $v_{p'} \neq v_{p,min}$ is used for scoring,
- any v's peer $v_{p'}$ (including $v_{p,min}$[7]) achieves expected payoff τ_{max} when scored against v and otherwise, when scored against some other peer, they achieve τ_{min}.

This gives us:

$$
\begin{aligned}
\alpha &\leq 1 - \frac{\tau_{min}}{\tau_{max} + (\omega - 1) \cdot \tau_{max} - (\omega - 1) \cdot \tau_{min}} \\
&\leq 1 - \frac{\tau_{min}}{\omega \cdot \tau_{max}}
\end{aligned}
$$

□

Proof of Lemma 3

Statement: Any feasible solution S to optimization problem (8) is also a feasible solution to optimization problem (6) (and thus (4)) for $B' \leq (1 - \alpha) \cdot B$, where α is the slope of cost function c_e.

[7] The analysis slightly changes if v_p is required to have the same payoff as v when they are mutual peers, but the main result stays the same.

Proof. Since both problems require that $S \in 2^V_\phi$, we only need to show that the budget constraint in (6) is not violated when S is selected. Let us enumerate the elements of S, so that each element is assigned an index i. We have:

$$\begin{aligned} c_M(S) &= \sum_{v\in S} c_m(v) = \sum_{v\in S} c_e(\{v\}) \\ &= \sum_{i=1}^{|S|} \frac{c_e(\{v_i\})}{c_e(\{v_i\}|\cup_{j<i}\{v_j\})} \cdot c_e(\{v_i\}|\cup_{j<i}\{v_j\}) \\ &\geq (1-\alpha)\sum_{i=1}^{|S|} c_e(\{v_i\}|\cup_{j<i}\{v_j\}) \\ &= (1-\alpha)\cdot c_e(S). \end{aligned}$$

The (first) inequality is due to the slope of c_e. Now, since $c_M(S) \leq B'$, it follows that $c_e(S) \leq \frac{c_M(S)}{1-\alpha} \leq \frac{B'}{1-\alpha} \leq B$, which proves the statement. □

Proof of Theorem 2

Statement: Let the maximal relative difference between modular costs of two peer nodes be bounded by r, i.e., $r \geq \max_{v\in V, v_p\in\mathcal{N}_v} \frac{c_m(v)}{c_m(v_p)}$, and let $\gamma = \max_{v\in V}\frac{c_m(v)}{B'} \in (0, \frac{1}{2})$. Then, the output $\widetilde{S^*}$ of Algorithm 1 has the following guarantees on the utility:

$$f(\widetilde{S^*}) \geq \left(1 - e^{-\frac{(1-\alpha)\cdot(1-2\cdot\gamma)}{1+r}}\right)\cdot f(\textsc{Opt}). \qquad (11)$$

Proof. Let $\textsc{Opt}'$ denote an optimal solution to optimization problem (8) when budget $B' = (1-\alpha)\cdot B$, $\bar{\textsc{Opt}}$ denote an optimal solution to optimization problem (8) when budget $B' = B$. Cleary $f(\bar{\textsc{Opt}}) \geq f(\textsc{Opt}')$. Furthermore, $f(\bar{\textsc{Opt}}) \geq f(\textsc{Opt})$, where $\textsc{Opt}$ is an optimal solution to optimization problem (6) (and (4)), because cost function $c_e(S)$ is lower bounded by its modular approximation $c_M(S)$. Therefore, it suffices to lower bound $f(\widetilde{S^*})$ with $f(\bar{\textsc{Opt}})$ modulated by the approximation factor in the statement of the theorem.

Let S_t represent the current solution $\widetilde{S^*}$ of the greedy algorithm at time step t and assume w.l.o.g. that z^*_t is not $NULL$.[8] Due to monotonicity and submodularity of function f and the fact that Algorithm 1 is greedy in terms of f/c ratio, we have:

$$\begin{aligned} f(\bar{\textsc{Opt}}) &\leq_{\text{mon.}} f(\bar{\textsc{Opt}}\cup S_t) \leq_{\text{sub.}} f(S_t) + \sum_{v\in\bar{\textsc{Opt}}\setminus S_t} f(v|S_t) \\ &\leq_{\text{mon.:adding a peer}} f(S_t) + \sum_{v\in\bar{\textsc{Opt}}\setminus S_t} f(\{v, v_{v,p}\}|S_t) \\ &= f(S_t) + \sum_{v\in\bar{\textsc{Opt}}\setminus S_t} \frac{c_M(\{v, v_{v,p}\})}{c_M(\{v, v_{v,p}\})}\cdot f(\{v, v_{v,p}\}|S_t), \end{aligned}$$

where $v_{v,p}$ is any peer of v. Using the fact that Algorithm 1 is greedy in terms of f/c ratio, we further obtain:

$$f(\bar{\textsc{Opt}}) \leq f(S_t) + \frac{f(z^*_t|S_t)}{c_M(z^*_t\setminus S_t)}\cdot \sum_{v\in\bar{\textsc{Opt}}\setminus S_t} c_M(\{v, v_{v,p}\}).$$

Now, $r \geq \max_{v, v_p\in V: v_p\in\mathcal{N}_v} \frac{c_m(v)}{c_m(v_p)}$ and $\sum_{v\in\bar{\textsc{Opt}}} c_m(v) \leq B$ give us:

$$\begin{aligned} f(\bar{\textsc{Opt}}) &\leq f(S_t) + \frac{f(z^*_t|S_t)}{c_M(z^*_t\setminus S_t)}\cdot(1+r)\sum_{v\in\bar{\textsc{Opt}}\setminus S_t} c_m(v) \\ &\leq f(S_t) + \frac{f(z^*_t|S_t)}{c_M(z^*_t\setminus S_t)}\cdot(1+r)\cdot B \end{aligned}$$

By rearranging, we get:

$$f(z^*_t|S_t) \geq \frac{c_M(z^*_t\setminus S_t)}{(1+r)\cdot B}\cdot[f(\bar{\textsc{Opt}}) - f(S_t)],$$

and further since $f(S_{t+1}) - f(S_t) = f(z^*_t|S_t)$ and $f(\emptyset) = 0$:

$$\begin{aligned} f(S_{t+1}) - f(\bar{\textsc{Opt}}) &\geq \left(1 - \frac{c_M(z^*_t\setminus S_t)}{(1+r)\cdot B}\right)\cdot[f(S_t) - f(\bar{\textsc{Opt}})] \\ &\geq \ldots \geq -\prod_{i=1}^{t+1}\left(1 - \frac{c_M(z^*_{i-1}\setminus S_{i-1})}{(1+r)\cdot B}\right)\cdot f(\bar{\textsc{Opt}}), \end{aligned}$$

where we used an inductive argument. In other words:

$$f(S_t) \geq \left(1 - \prod_{i=1}^{t}\left(1 - \frac{c_M(z^*_{i-1}\setminus S_{i-1})}{(1+r)\cdot B}\right)\right) f(\bar{\textsc{Opt}}),$$

where we for notational convenience considered S_t instead of S_{t+1}. Now, because:

$$\begin{aligned} &\prod_{i=1}^{t}(1+x_i) = (e^{\frac{1}{t}\sum_{i=1}^{t}\ln(1+x_i)})^t \\ &\leq_{e \text{ is convex}} \left(\frac{1}{t}\sum_{i=1}^{t} e^{\ln(1+x_i)}\right)^t = \left(1 + \sum_{i=1}^{t}\frac{x_i}{t}\right)^t, \end{aligned}$$

for $x_i \in (-1, 1)$, we obtain:

$$\begin{aligned} f(S_t) &\geq \left(1 - \left(1 - \sum_{i=1}^{t}\frac{c_M(z^*_{i-1}\setminus S_{i-1})}{t\cdot(1+r)\cdot B}\right)^t\right) f(\bar{\textsc{Opt}}) \\ &\geq \left(1 - \left(1 - \frac{c_M(S_t)}{t\cdot(1+r)\cdot B}\right)^t\right) f(\bar{\textsc{Opt}}), \end{aligned}$$

where we used the fact that $\sum_{i=1}^{t} c_M(z^*_{i-1}\setminus S_{i-1}) = c_M(S_t)$. Finally, we transform B to $B' = (1-\alpha)\cdot B$ to obtain:

$$f(S_t) \geq \left(1 - \left(1 - \frac{c_M(S_t)\cdot(1-\alpha)}{t\cdot(1+r)\cdot B'}\right)^t\right) f(\bar{\textsc{Opt}}).$$

Now, at a certain time step t^*, Algorithm 1 cannot find any node or a pair of nodes that could be added. Because

[8] This is true for at least $t = 1$ due to the assumption on the boundedness of modular payments, i.e. $\gamma < \frac{1}{2}$.

modular costs are bounded by $c_m(v) \leq \gamma \cdot B'$, we know that at time step t^*, Algorithm 1 spent at least $B' - 2 \cdot \gamma \cdot B'$ budget.[9] Hence, $c_M(S_t) \geq (1 - 2 \cdot \gamma) \cdot B'$, which gives us that:

$$\begin{aligned} f(S_{t^*}) &\geq \left(1 - \left(1 - \frac{(1 - 2 \cdot \gamma) \cdot (1 - \alpha)}{t^* \cdot (1 + r)}\right)^{t^*}\right) f(\bar{\text{OPT}}) \\ &\geq \left(1 - e^{-\frac{(1-\alpha)\cdot(1-2\cdot\gamma)}{1+r}}\right) f(\bar{\text{OPT}}). \end{aligned}$$

Finally, using the fact that $f(\bar{\text{OPT}}) \geq f(\text{OPT})$ and $\widetilde{S^*} = S_{t^*}$ we obtain:

$$f(\widetilde{S^*}) \geq \left(1 - e^{-\frac{(1-\alpha)\cdot(1-2\cdot\gamma)}{1+r}}\right) \cdot f(\text{OPT}).$$

□

[9] Otherwise, there would be a node or a pair of nodes that could be added, or all of the elements are selected, which is the optimal choice and thus trivially proves the statement.