PASS Approximation: A Framework for Analyzing and Designing Heuristics

Uriel Feige [*†] Nicole Immorlica [*‡] Vahab S. Mirrokni [*§] Hamid Nazerzadeh [¶]

September 15, 2009

Abstract

We introduce a new framework for designing and analyzing algorithms. Our framework applies best to problems that are inapproximable according to the standard worst-case analysis. We circumvent such negative results by designing guarantees for classes of instances, parameterized according to properties of the optimal solution. Given our parameterized approximation, called *PArametrized by the Signature of the Solution (PASS)* approximation, we design algorithms with optimal approximation ratios for problems with additive and submodular objective functions such as the capacitated maximum facility location problems. We consider two types of algorithms for these problems. For greedy algorithms, our framework provides a justification for preferring a certain natural greedy rule over some alternative greedy rules that have been used in similar contexts. For LP-based algorithms, we show that the natural LP relaxation for these problems is not optimal in our framework. We design a new LP relaxation and show that this LP relaxation coupled with a new randomized rounding technique is optimal in our framework.

In passing, we note that our results strictly improve over previous results of Kleinberg, Papadimitriou, and Raghavan [JACM 2004] concerning the approximation ratio of the greedy algorithm.

[*]Work performed in part at Microsoft Research.

[†]Weizmann Institute, Rehovot, Israel, email: uriel.feige@weizmann.ac.il

[‡]Northwestern University, Chicago, IL, email: nickle@eecs.northwestern.edu

[§]Google Research, New York, NY, email: mirrokni@theory.csail.mit.edu

[¶]Microsoft Research, Cambridge, MA, email: hamidnz@microsoft.com

1 Introduction

Many important optimization problems in practice are inapproximable in theory. Practitioners deal with inapproximability issues by designing heuristics that, while provably bad on some instances, appear to perform well in practice. But for theoreticians, designing a formal framework to help guide algorithmic development for inapproximable problems has proved largely elusive.

In this paper, we present a new framework, called *PArametrized by the Signature of the Solution (PASS) approximations*. Our framework attempts to categorize instances according to how "easy" or "hard" they are, and design guarantees for all instances simultaneously with a single algorithm (the offered guarantee depends on the class of the instance and will degrade to arbitrarily bad factors for inapproximable problems, but in a controlled way). We show how this framework can be applied to a general class of optimization problems, including *capacitated maximum facility location*, that can be described as maximizing a non-decreasing submodular revenue function minus a linear cost function. We then show how the new framework affects the choice of algorithms. Two standard approaches for handling such problems are via greedy and LP-based algorithms. We study a natural greedy algorithm and prove that it is an optimal PASS approximation whereas other greedy algorithms that give optimal worst-case approximations are not. For LP-based algorithms, we show that a natural LP relaxation cannot be used to design an optimal PASS approximation. Instead, we provide a different LP relaxation and an associated rounding technique that is optimal. Our new LP relaxation is unconventional in the sense that instead of providing an upper bound on the optimal solution (this is a maximization problem), it provides a lower bound.

The current paper outlines the theory of PASS approximations. We describe the general technique and how to apply this technique to a wide range of theoretical problems using both greedy and LP-based algorithms. A conference version of this paper appeared in [8]. In a companion paper [7], we apply the notion of approximation developed here to a specific problem (banner advertising) of practical significance. This problem is a special case of the broad class of problems studied in this paper.

The rest of the paper is organized as follows. After defining the problems, in Section 2, we describe the theory of PASS approximation and our results. In Section 3, we compare our results to the previous approaches proposed to cope with the hardness of approximation. The greedy and LP-based algorithms are presented respectively in Sections 5 and 6.

2 The theory of PASS approximation

Let us illustrate our framework with a discussion of the following natural problem of maximum facility location [1, 3].

Problem 2.1. Maximum Facility Location (MFL). *A set $\mathcal{F}$ of m facilities is given. For every facility i, there is an opening cost of c_i. There is also a set $\mathcal{J}$ of n clients. The revenue of connecting client j to facility i is $u_{ij} \geq 0$ (this may be interpreted as a client revenue minus a connection cost). Every client can connect to at most one open facility (or none). The goal in MFL is to open some facilities and connect clients to them so as to maximize the total revenue from the connected clients minus the total cost of the opened facilities.*

Despite its wide practicality, this problem has remained elusive due to its inapproximability. In fact, via a reduction from the Maximum Independent Set (MIS) problem, one can show that MFL is hard to approximate within any nontrivial factor. Given a graph, an independent set is defined as a subset of vertices with no edge among them. MIS is to find an independent set of the maximum cardinality. For any $\epsilon > 0$, it is NP-hard to approximate the MIS problem within a factor of $n^{-1+\epsilon}$ [14, 20] (n is the number of vertices).

Any instance of MIS can be reduced to an instance of MFL using the following reduction. Each vertex of the graph corresponds to a facility and each edge to a client. For each facility, the opening cost is equal to its degree minus 1. For each pair of facility i and client j, the revenue is equal to 1 only if edge j is connected to vertex i, otherwise, the revenue is equal to 0. It is easy to see that any solution of MFL - in which only facilities with positive value are open - corresponds to an independent set, and vice versa. Therefore, MFL is hard to approximate within any non-trivial factor.

Nonetheless, there are large classes of interesting instances in which the approximation ratio can be much better than the worst-case guarantees. For example, if the cost of opening facilities is significantly smaller than the revenue they generate, then intuitively the instance is "easy" because the optimum is large. Our goal is to get a better understanding of the approximation ratio, exposing classes of input instances for which a constant approximation ratio is possible.

We first present two plausible approaches that guide us in our definition of the PASS approximation framework. Based on the intuition of the previous paragraph, one might expect the ratio of facility cost-to-profit to provide a good proxy for approximability. Unfortunately, as discussed below, the inapproximability of the problem carries through to such instances.

Example 2.2. Relatively small costs. *For $0 < \alpha < 1$, let us call an instance α-bounded if for every facility, the cost of opening the facility is at most α times the revenue one gets by connecting all clients to the facility. Is it the case that when α is sufficiently small there is a constant approximation for MFL for α-bounded instances? The answer is negative. The proof involves starting from a hard to approximate instance of MFL and adding an additional client that provides revenue $\max(c_i/\alpha)$ regardless of which facility services it. This forces the instance to be α-bounded, while increasing the value of an optimal solution by only $\max(c_i/\alpha)$. An appropriate choice of parameters leads to the desired hardness result.*

The problem with the definition in the previous paragraph is that it is insensitive to the *structure of the solution*. Kleinberg, Papadimitriou, and Raghavan [16] introduced a measure to correct for this. Unlike the notion of an α-bounded instance discussed previously, the idea is to use the notion of α-boundedness not with respect to the input instance, but rather with respect to its optimal solution.

Example 2.3. [16] *Call a solution α-bounded if the total cost of opening the facilities in this solution is at most an α-fraction of the total revenue derived from all clients in the solution.* [1] *In [16], it is shown that for a special case of MFL called the catalog segmentation problem (see Appendix A), whenever there exists an α-bounded solution with α away from 1, the approximation ratio of a natural greedy algorithm is a constant (that tends to 1 as α tends to 0).*

In the catalog segmentation problem considered by Kleinberg et al. [16], the facilities all have the same opening cost, and their notion of approximation is tailored to this situation. When opening costs differ, as in MFL, or even when they are the same as in the catalog segmentation problem, we can improve the precision of the approximation notion by introducing a measure that is sensitive to the impact of individual facilities.

2.1 PASS Approximation Framework

In our framework of *PASS approximation*, we express the approximation ratios of algorithms as a function of the *signature* of the optimal solutions. This signature, defined below, can be interpreted as an extension of the Kleinberg et al. [16] framework to our general setting which allows us to distinguish between instances

[1] Technically, in [16] a different parameter μ is considered, which in our terminology is $\mu = \frac{1}{\alpha} - 1$. It is straightforward to translate results expressed in terms of μ to results expressed in terms of α and vice versa.

and hence quantify their approximability. We use the signature to define the *recoverable value* of an instance and design instance-blind algorithms that generate the recoverable value on every instance. Our framework is optimal in that it is NP-hard to generate more than the recoverable value.

2.1.1 Signatures

We first introduce the following notation summarized in Section 2.1.3. Consider an arbitrary MFL instance I and an arbitrary feasible solution S. For each facility i open in S, let c_i be its opening cost, and let $r_i = \sum u_{ij}$ (where the sum is taken over clients j connected to facility i in S) be the total revenue derived from clients connected in S to facility i. [2] Let α_i denote the ratio c_i/r_i. On a global scale, the total revenue of S is $R(S) = \sum_{i \in S} r_i$, the total cost is $C(S) = \sum_{i \in S} c_i$, and the value of solution S is $V(S) = R(S) - C(S)$. Similar to the local values α_i, we shall use α to denote an aggregate value $\alpha = C(S)/R(S)$.

Definition 2.4. *Given an instance I of MFL and a feasible solution S, and using notation as above:*

- *The* expanded signature *of S is the collection $\{(q_i, \alpha_i)\}$, where i ranges over all facilities open in S, $q_i = r_i/R(S)$, and $\alpha_i = c_i/r_i$.*
- *The* signature *$sig(S)$ of S is the collection $\{(q_i, \alpha_i)\}$ obtained from the expanded signature by unifying components that share the same value of α_i. Namely, in the signature i no longer refers to a specific facility, all α_i are distinct, r_i denotes the total revenue that comes from open facilities which share the same α_i value (namely $r_i = \sum_{\text{facilities } i' : \alpha'_i = \alpha_i} r_{i'}$), and q_i denotes the fraction of revenue that comes from open facilities which share the same α_i value (namely, $q_i = r_i/R(S)$).*

For every signature $\sum q_i = 1$. If all open facilities in S have the same value α_i then the signature is $(1, \alpha)$, in which case we abbreviate it to α. Intuitively, a value of α_i close to 0 indicates that opening the facility i was a favorable decision, because the revenue r_i that resulted from this opening came at relatively little cost. A value of α_i close to 1 indicates that the opening of facility i may have been questionable, as most of the revenue r_i is offset by the cost c_i.

We note it is important to distinguish between the *expanded signature* and the *signature* in order to be able to talk about asymptotics in the hardness results as for any fixed expanded signature there are a fixed number of facilities. For the positive results, the notions of expanded signature and signature are interchangeable by changing the index of summation, and the reader may find it easier to interpret the positive results using the expanded signature. In order to compare our results to the prior work of Kleinberg et al. [16], we also introduce the *summary signature* $\alpha = C(S)/R(S)$. This parameter can be interpretted as summarizing to some extent the signature, even though it does not have the same distinguishing power among solutions as the signature does. These definitions are illustrated by example in Figure 1.

2.1.2 Recoverable Value

We express the approximation ratios of algorithms as a function of the signature. Observe that an instance may have multiple different signatures (one for each feasible solution). Our approximation ratios will apply to all of them (and hence to the best of them). Nevertheless, the reader may find it convenient to think of the signature of an instance as that of (one of) its optimal solution(s). Given any feasible solution (e.g., an optimal one) with signature S, for every index i solution S generates a value of $r_i(1-\alpha_i)$ from facilities with α value equal to α_i. Our algorithms may open facilities different than those opened by S, but our accounting

[2] A better notation might be to write $r_i(S)$ instead of r_i, but we use r_i for brevity.

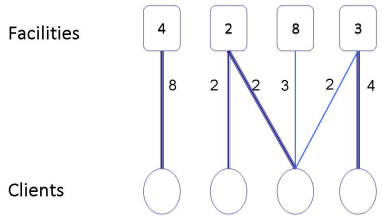

Figure 1: An instance of the facility location problem: the costs of the facilities and the revenue from the clients (on the edges). An optimal solution is depicted by solid lines. The expanded signature is $\{(\frac{1}{2},\frac{1}{2}),(\frac{1}{4},\frac{1}{2}),(\frac{1}{4},\frac{3}{4})\}$, the signature is $\{(\frac{3}{4}\ ,\frac{1}{2}),(\frac{1}{4},\frac{3}{4})\}$, and the summary signature is $\frac{9}{16}$.

method will show that our algorithms recover value at least $\hat{v}_i = r_i(1-\alpha_i-\alpha_i \ln \frac{1}{\alpha_i})$ in exchange to the value generated by S from index i. This parameter $\hat{v}_i$ is therefore called the *recoverable value*, and, as we will prove, it is the optimal recoverable value (i.e., it is NP-hard to recover more). We note the precise mathematical notion of recoverable value may differ from problem to problem; in general it says that there is a certain value that we can recover in exchange to the value obtained by the optimal solution. Below we state this definition as it applies to our flagship problem of MFL.

Definition 2.5. *Given an instance I of MFL and a feasible solution S, and using notation as above, the* recoverable value *of the instance is $\hat{v}_i = r_i(1-\alpha_i-\alpha_i \ln \frac{1}{\alpha_i})$.*

Note that $0 \leq \hat{v}_i \leq r_i$, with $\hat{v}_i = 0$ when $\alpha_i = 1$ (i.e., we cannot recover any value from facilities whose cost equals their revenue) and $\hat{v}_i = r_i$ when $\alpha_i = 0$ (i.e., we can recover all the revenue from facilities with zero cost). To simplify the presentation in this paper, and with no significant effect on the results, we pretend that quantities such as $\ln x$ can be computed exactly in polynomial time for every x.

2.1.3 A summary of notations

For a solution S and open facility i we have:

c_i**:** the opening cost of facility i.

r_i**:** the revenue obtained from clients connected to i.

$R(S), C(S), V(S)$**:** total revenue, cost, and value (= revenue minus cost) of S.

$\alpha_i \doteq c_i/r_i$

$\alpha \doteq C(S)/R(S)$

$q_i \doteq r_i/R(S)$, relative revenue of i.

$\hat{v}_i \doteq r_i(1-\alpha_i-\alpha_i \ln \frac{1}{\alpha_i})$, the recoverable value of i.

2.2 Our Results

Our main contribution is the introduction of the PASS approximation framework. We demonstrate the power of this framework by providing a full analysis of a general class of problems, called *submodular maximum facility location*, that can be described as maximizing a non-decreasing submodular revenue function minus a linear cost function.

Problem 2.6. Submodular Maximum Facility Location (SMFL). *Consider a set N of n facilities and a set function $f : 2^N \to R^+$. For any subset $S \subset N$, $f(S) = R(S) - c(S)$, where R is a non-negative non-decreasing submodular set function corresponding to the revenue, and $c(S) = \sum_{i \in S} c_i$ is a linear cost function. As a result, set function f is a non-monotone submodular function and the goal is to find a subset S that maximizes $f(S)$.* [3] *We assume a value oracle for the revenue function R and a description for the cost c (this is of polynomial size) are given.*

The maximum facility location problem discussed above is a special case of SMFL, as is the catalog segmentation problem studied by Kleinberg et al. [16]. Other examples include a variety of optimization problems such as capacitated maximum facility location, set buying, banner ad allocation problem with guaranteed delivery [7], viral marketing in social networks [15, 18], and optimal sensor installation for outbreak detection [17]. See Appendix A for the complete definitions.

For this class of problems, we first present a hardness result and then give tight greedy and LP-based algorithms.

Theorem 2.7. *Let sig $= \{(q_i, \alpha_i)\}$ be an arbitrary signature, and consider the class of MFL instances that have an optimal solution with signature sig. For simplicity of notation, for each such instance, normalize the costs and revenues such that the revenue of the optimal solution having signature sig is 1, and hence its value is $1 - \sum q_i \alpha_i$. Then on this class of instances, for every $\epsilon > 0$, it is NP-hard to find a solution of value $\sum \hat{v}_i + \epsilon$ where $\hat{v}_i = q_i(1 - \alpha_i - \alpha_i \ln \frac{1}{\alpha_i})$.*

Corollary 2.8. *For any $\epsilon > 0$ and $\alpha = C(S)/R(S)$, for any optimal solution S, it is NP-hard to approximate MFL within a ratio better than $\frac{1-\alpha-\alpha \ln \frac{1}{\alpha}}{1-\alpha} + \epsilon$.*

As MFL is a special case of SMFL, the above applies to SMFL as well.

We next show that there are algorithms with approximation ratios that match the hardness results. The first class of algorithms that we consider is that of greedy algorithms. We shall distinguish between two types of greedy algorithms depending on whether it is greedy with respect to margin or to rate. Only one of these versions is optimal in our framework.

Theorem 2.9. *Let I be an arbitrary instance of MFL, let S be an arbitrary feasible solution and let $\{(q_i, \alpha_i)\}$ be the signature of S. For simplicity of notation, normalize the costs and revenues in I such that the revenue of S is 1, and hence its value is $1 - \sum q_i \alpha_i$. Then the greedy-rate algorithm produces a solution of value at least $\sum \hat{v}_i$ where $\hat{v}_i = q_i(1 - \alpha_i - \alpha_i \ln \frac{1}{\alpha_i})$.*

Corollary 2.10. *The greedy-rate algorithm approximates MFL within a ratio of at least $\frac{1-\alpha-\alpha \ln \frac{1}{\alpha}}{1-\alpha}$, where $\alpha = C(S)/R(S)$ for any optimal solution S.*

As shown in Section 5.2, the result above holds for the SMFL problem as well, under the appropriate definition of signature. We also remark that Corollaries 2.8 and 2.10 are each stronger than previous results proved in [16]. These issues will be discussed in Section 5.

[3] Note that function f can be possibly negative and therefore the result of Feige et al. [11] does not apply.

The next class of algorithms that we consider is based on linear programming. We also show that the natural linear programming relaxation does not result in approximation ratios that match the hardness results of Theorem 2.7. Hence, we introduce a new linear program, called the *recoverable value LP*, whose objective is to maximize the (fractional) recoverable value rather than the (fractional) true value.[4] We then show that the LP can be rounded to give a feasible solution of value not lower than the recoverable value of the LP.

Theorem 2.11. *The recoverable value LP for MFL can be solved in polynomial time. For every input instance and feasible solution S with signature $\{(q_i, \alpha_i)\}$, the LP has a solution of value at least as high as $\sum \hat{v}_i$, where $\hat{v}_i = r_i(1 - \alpha_i - \alpha_i \ln \frac{1}{\alpha_i})$. Any solution of the LP can be rounded in random polynomial time to give a feasible solution of expected value at least as high as the value of the objective function in the LP solution.*

Unfortunately, we are unable to apply our LP-based approach to the general class of SMFL as we do not know how to solve the LP efficiently or round it in those instances. Furthermore, in the case of MFL, the performance guarantees that we prove for the greedy algorithm and the LP-based algorithm are the same. We argue that, nonetheless, the LP-based approach is a significant contribution of this work. For one, it diversifies our algorithmic toolbox, and showing how to use linear programming relaxations in the context of PASS approximations (which turns out to be different than the way linear programming relaxations are typically used in "classical" approximations) is anticipated to lead to rewards in future work. But more importantly, there is a conceptual difference between the use of PASS approximation framework for our greedy versus LP-based algorithms. For our greedy algorithm, the theory of structural approximation is *descriptive*. It describes the approximation ratios of existing algorithms, and may guide us in the choice of the greedy rule to use. For the LP-based approach, however, the theory of PASS approximation is not only descriptive, but also *prescriptive*. It guides us in the design of new algorithms. The definitions of the signature and recoverable value define for us the linear program and the rounding technique. While in our examples, the greedy algorithms happens to be tight, the LP-based approach may still be of value for other problems precisely because of its prescriptive nature – *it is designed to produce tight algorithms*.

3 Motivating PASS approximations

In this section we present arguments in favor of our notion of PASS approximation. The point that we will try to make is that performance measures guide the design of algorithms, and our performance measure appears to us to be a very good guide. We assume in the discussion below that the true goal is to maximize revenue minus cost, and compare various approaches that can be used in order to circumvent the inapproximability results for this measure. Recall that our approach of PASS approximation is to express the approximation ratio not as a function of the size of the input instance, but as of its signature.

3.1 Prior attempts

As mentioned, MFL and SMFL are NP-hard to approximate. Therefore, researchers have attempted to present other types of performance guarantees. We analyze a few such approaches below. One approach is to change the objective function in a way that preserves the spirit of the original problem. For example, one might consider the complement of the objective (e.g., vertex cover as opposed to independent set), or

[4] We note this relaxation is not a relaxation in the usual sense, because the value of the objective function of the LP is a lower bound on the value of an optimal solution, rather than an upper bound.

bicriteria approximations (e.g., bisection in graphs). Another approach is to impose additional constraints that make the problem approximable. Yet another approach is to perturb the instance, provide performance guarantees, and thereby argue that the problem is "typically" solvable.

3.1.1 Shifted scale

In Cornuejols et al. [3] and Ageev and Sviridenko [1], the approach taken was to measure the quality of a solution on a shifted scale which is always nonnegative. This is equivalent to changing the objective function by adding to it a sufficiently large constant that ensures that all solutions have nonnegative value. As an example, consider algorithms for MFL based on linear programming. In Ageev and Sviridenko [1] a combination of a linear program and rounding technique is designed. They show that the approximation ratio of $2(\sqrt{2}-1)$ that they obtain is best possible (matches the integrality gap), but with respect to a shifted scale of the objective function. As we do not claim the same about our linear programming approach, then clearly there are instances in which their algorithm is better than ours. Likewise, there are instances on which our LP plus rounding gives better results (because we are optimal with respect to the PASS approximation measure, whereas Ageev and Sviridenko [1] are not). Hence it appears as if the results are incomparable. Nevertheless, we would like to convince the reader that even though the result of Ageev and Sviridenko [1] is interesting mathematically, it does not really provide the kind of algorithmic insights that are relevant to the original problem. To obtain an approximation ratio of $2(\sqrt{2}-1)$ with respect to the shifted scale, it is safe to open every facility with probability at least $1-2(\sqrt{2}-1)$ (and at most 1), regardless of the cost of the facility, and regardless of whether any client wants to connect to the facility. This is a simple (and obviously counterproductive) rule of thumb that comes out of the shifted scale performance measure, and in fact the algorithm of Ageev and Sviridenko [1] follows it. We view this as evidence that the shifted scale performance measure is not a good guide in the design of algorithms (with respect to the original objective function).

3.1.2 Budget constraints

Another approach taken by theoreticians is to introduce some budget limit B, and then seek to maximize revenue subject to keeping the cost below the budget. (Of course, in some cases there really is a budget limit and then this may be the appropriate measure to consider. However, in the discussion here we assume that this change in the objective function is done so as to circumvent the inapproximability results.) The kind of problems discussed in this paper typically have an approximation ratio of $1-1/e$ with respect to this measure. Thereafter, to get good results with respect to the original measure one may try all possible values of B (lets assume for simplicity here that there are only polynomially many such values), and take the best solution found. This approach need not give any performance guarantee with respect to the original objective function. If one uses an LP approach to approximate the budgeted problem then the LP and its rounding will be different than the LP and rounding developed in the current paper, and will not provide optimal results for the PASS approximation framework (not even if B is set to be the exact cost of the solution returned by our LP approach). However, if one uses a greedy approach to approximate the budgeted problem, it will use the greedy-rate rule, and hence this will lead to essentially the same greedy algorithm that is used in this paper. In this case, our contribution is *descriptive*, allowing one to provide an expression for the approximation ratio achieved.

3.1.3 Average-case and smoothed analysis

Another approach theoreticians take uses average-case performance measures to analyze heuristics for the original problem. For this, one needs to define a probability distribution over inputs, and show that with high probability over the choice of an input instance from this distribution, the algorithm performs well. The difficulty in this approach is to justify the choice of a particular probability distribution. Semi-random inputs [2, 9, 19] try to resolve this issue by involving both worst case and random ingredients. In smoothed analysis [19], for example, one applies a small random perturbation to the input and hopes that this suffices to move from a worst case instance to an easy instance. This framework has proved successful in explaining why some important heuristics perform well in practice, most notably in showing that (a version of) the simplex algorithm for linear programming has smoothed polynomial running-time [19]. While typically applied to analyzing running times of worst-case exponential-time algorithms, smoothed analysis can also be used to analyze approximation ratios. However, for the problems studied in this paper the approximation ratio for smoothed instances (under reasonable models for the smoothing operation) remains poor. As an example, for the MFL problem, one can apply smoothed analysis by choosing for every revenue u_{ij} a random *noise* o_{ij} (say, from a distribution with mean 0 and absolute value bounded by some small ϵ) and change the revenue to $(1 + o_{ij})u_{ij}$. One might hope that this would make the input instance easier to approximate, for example, admit constant-factor approximation algorithms where the constant depends on ϵ. Unfortunately, this is not the case: for any instance if one duplicates clients, the sum of revenues of duplicated clients becomes highly concentrated around the expectation. A similar attempt to add independent random noise to every cost value c_i also fails, this time by duplicating facilities, and considering concentration of the minimum of $(1 + o_i)$ over multiple random choices of o_i.

3.2 PASS approximations

In our work, we do not attempt to circumvent the inapproximability result by changing the objective (Section 3.1.1), the constraints (Section 3.1.2), or by perturbing the input (Section 3.1.3). Rather we work directly on the original problem and provide performance guarantees that degrade with the approximability of the instance. In this vein, our work is quite similar to that of Kleinberg et al. [16]. In their work, however, the approximation ratio is expressed as a function of one parameter that we refer to as the summary signature α. What is the advantage of presenting the more complicated signature $\{(q_i, \alpha_i)\}$?

We see two advantages (beyond the obvious advantage of always providing a performance guarantee that is at least as good as that provided by the summary signature). One is *prescriptive*: the design of our LP is a natural consequence of our signature, valuing each star according to its own recoverable value. It would have been very difficult to design and analyze it without having at least implicitly a notion similar to the expanded signature.

The other advantage is *conceptual*: our signature enjoys closure properties that the summary signature does not have. Given two disjoint instances of SMFL, the expanded signature becomes simply the union of the original expanded signatures, and the output guarantee (approximation ratio times value of optimal solution) is simply the sum of output guarantees of the two instances. For the summary signature, this is not true.

A notion of approximation related to that of PASS has been used in the context of *max-cut*. The approximation ratio for max-cut is sometimes expressed as a function of a property of the optimal solution, namely, as a function of the fraction of edges that are cut in the optimal solution (see [21] for example). This parametrization is quite informative, and alerts one to the fact that the approximation ratios tend to 1 when the value of this parameter (which is always between 1/2 and 1) approaches 1, and when it approaches 1/2.

One may think of this parameter as a summary signature of the solution, and it would be interesting to see if the introduction of a notion of a expanded signature for max-cut can lead to further improvements in the approximation ratios.

4 Hardness of approximation

In this section, we prove Theorem 2.7 for a simple yet important special case of MFL problem called set buying.

Set Buying (SB). *The input to SB consists of a set $\mathcal{J}$ of n items, where each item $j \in \mathcal{J}$ has* revenue $u_j \geq 0$*, and a family $\mathcal{F}$ of m sets of items, where each set $S_i \in \mathcal{F}$ has* cost $c_i \geq 0$*. The value of a subfamily $\mathcal{W} \subset \mathcal{F}$ of sets is $V(\mathcal{W}) = \sum_{j \in \cup_{S_i \in \mathcal{W}}} u_j - \sum_{S_i \in \mathcal{W}} c_i$. The goal is to find a collection $\mathcal{W}$ with the maximum value $V(\mathcal{W})$.*

For intuition, consider an instance of SB in which every item has revenue 1, all sets are of exactly the same size k (where k is large, so as to be able to later use a continuous approximation) and the same cost (αk, for $\alpha < 1$), and that the number of items n is divisible by k. We define two types of instances: *yes* and *no*. Assume that on *yes* instances, there exists a disjoint cover. Hence the value is $\frac{n}{k}(1-\alpha)k = (1-\alpha)n$. On *no* instances, the sets are random, which essentially implies (up to low order terms) that for every β, once a β fraction of the items are covered, in every remaining set, a β fraction of its items are already covered. Hence on *no* instances, one should keep on picking sets until a $(1-\alpha)$ fraction of the items are covered, as afterward every additional set costs more than its marginal revenue. Covering a $(1-\alpha)$ fraction of the items takes s random sets, where $(1-\frac{k}{n})^s = \alpha$. This gives $s = \frac{n}{k}\ln\frac{1}{\alpha}$, and the value is then $(1-\alpha)n - s\alpha k = (1-\alpha-\alpha\ln\frac{1}{\alpha})n$. It is known that it is hard to distinguish *yes* instances from *no* instances [10], and we can use this to prove hardness of approximation ratio for our problem. We formalize these ideas in the following theorem (which directly implies Corollary 2.8).

Theorem 4.1. *Consider instances of the set buying problem in which each item has revenue 1 and each set S has cost $\alpha|S|$, $0 \leq \alpha \leq 1$. For any $\varepsilon > 0$, it is NP-hard to approximate set buying problem over these instances within a ratio of*

$$\frac{1-\alpha-\alpha\ln\frac{1}{\alpha}+\varepsilon}{1-\alpha}$$

To prove the theorem above, we give a reduction from the *k-uniform set cover* problem defined as follows. Given a family of subsets of a ground set, each of size k, find the minimum number of subsets which cover all the elements. Following the intuition given in the beginning of the section, we shall use the following proposition.

Proposition 4.2. *[5, 10] For every choice of constants $s_0 > 0$ and $\varepsilon > 0$ and for sufficiently large k, it is NP-hard to distinguish between k-uniform instances of set cover with n elements for which all elements can be covered by $t = \frac{n}{k}$ disjoint sets (*yes *instances), and instances in which every $s \leq s_0 t$ sets cover at most a fraction of $1-(1-\frac{1}{t})^s+\varepsilon$ of the elements (*$\mathrm{no}(s_0,\varepsilon)$ *instances).*

Given an instance of $k-$uniform set cover, assume the revenue of each item is 1 and the cost of each set is αk. On *yes* instances, there is a disjoint cover. Hence the value is $\frac{n}{k}(1-\alpha)k = (1-\alpha)n$. On the other hand:

Lemma 4.3. *Consider α, $0 < \alpha < 1$, and let $s_0 = \frac{1}{\alpha}$. For any collection of s sets, the maximum value could be obtained from a* $\mathrm{no}(s_0,\varepsilon)$ *instance is at most* $(1-\alpha-\alpha\ln\frac{1}{\alpha}+\varepsilon)n$.

Proof. Choosing any collection of $s \geq s_0 t$ sets incurs a cost of at least $s\alpha k \geq s_0 tk\alpha \geq \alpha s_0 n \geq n$. Therefore, the value would be non-positive. For a *no*(s_0, ε) instance, any collection of s sets, $0 \leq s \leq s_0 t$, covers at most a fraction $1-(1-\frac{1}{t})^s+\varepsilon$ of all elements which obtains a value of at most $(1-(1-\frac{1}{t})^s+\varepsilon)n - sk\alpha$. Since t can be made arbitrary large, this can be approximated, with arbitrary high accuracy, by $(1-e^{\frac{-s}{t}}+\varepsilon)n - sk\alpha$. Using the first order conditions, we get $s = t\ln\frac{1}{\alpha}$ as the best value for s.This gives an upper bound $(1-\alpha-\alpha\ln\frac{1}{\alpha})n$ on the value. □

By the lemma above, for $0 \leq \alpha \leq 1$, the ratio between the values of *yes* instances and *no*$(\frac{1}{\alpha}-1, \varepsilon)$ instances is at least: $(1-\alpha-\alpha\ln\frac{1}{\alpha}+\varepsilon)/(1-\alpha)$. The claim follows by Proposition 4.2.

The proof of Theorem 2.7 (which refers to arbitrary signatures $\{q_i, \alpha_i\}$) can now be deduced from the closure under union property of signatures. Take a union of disjoint instances as referred to in Theorem 2.7, each with the appropriate α_i, and with relative weights as dictated by the respective q_i.

5 A greedy approach

One standard approach for the MFL problems are greedy algorithms. In this section, we describe two plausible greedy algorithms for SMFL and prove that only one of them is optimal with regards to the PASS approximation. Given a set S of facilities, let $C(S)$ denote the total cost of facilities in S, let $R(S)$ denote the revenue of the optimum assignment given that the open facilities are those in S, and let $V(S) = R(S) - C(S)$ denote the total value of S. Given a facility i, let $M(i|S)$ denote the marginal revenue of i with respect to S. Namely, $M(i|S) = R(S \cup \{i\}) - R(S)$. If $i \in S$ then $M(i|S) = 0$.

The greedy algorithms construct a solution iteratively by selecting facilities that maximize some function of the marginal revenue $M(i|S)$. Given a partial solution (set of open facilities) S, the *greedy-rate* algorithm opens the facility i which maximizes the *rate* of increase in value, i.e., $\frac{M(i|S)-c_i}{M(i|S)}$, provided that this rate is positive. The *greedy-margin* algorithm simply opens the facility with the largest marginal value, i.e., $M(i|S) - c_i$, provided that this value is positive.

The greedy step can be implemented in polynomial time for the special cases of SMFL mentioned in Section 2.2. (For example, for CMFL, implementing the greedy step involves computing the optimal assignment of clients to the open facilities subject to the capacity constraints. This can be solved in polynomial time via an algorithm for the so called B-matching problem in bipartite graphs.) However, in general, the greedy step for SMFL might be NP-hard.

5.1 Comparison to KPR

The greedy-margin algorithm was studied by Kleinberg, Papadimitriou, and Raghavan [16] for the catalog segmentation problem (and generalizations which maintain the property of uniform-cost facilities). [5] They proved the following theorem (Theorems 2.3 and 2.4 in [16]):

Theorem 5.1. [16] *For the catalog segmentation problem, the greedy-margin algorithm achieves an approximation ratio of at least* $1+\alpha-2\sqrt{\alpha}$*, where* $\alpha = C(S)/R(S)$ *for any optimal solution* S*. There are instances on which the approximation ratio of the algorithm is no better than* $1-\alpha$*.*

[5] The greedy algorithm specified prior to Theorem 2.3 in [16] does not specify a rule of which facility to open next, as long as its marginal revenue is larger than its cost. However, the proof of Theorem 2.4 in [16] is based on the use of a greedy-margin rule, without stating this explicitly.

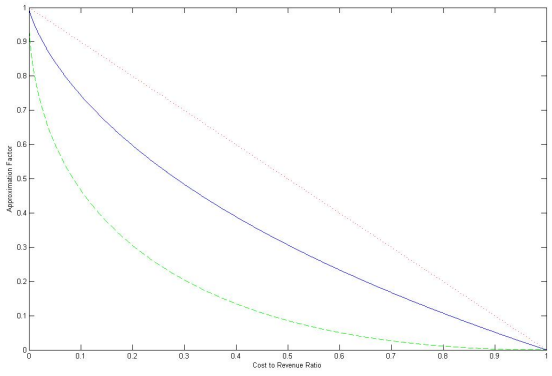

Figure 2: The solid line is the approximation ratio of the greedy-rate algorithm plotted for $\alpha \in [0, 1]$, as proved in Corollary 2.10. For every value of α improving over this approximation ratio is NP-hard, as proved in Theorem 2.7. The dashed and dotted lines depict the lower and upper bounds proved in [16].

In this section, we will improve upon this result by generalizing the analysis to accommodate non-uniform facility costs and providing improved approximation guarantees (Theorem 2.9) together with a matching NP-hardness result (Theorem 2.7). In doing so, we must be careful with our choice of greedy algorithm. We have defined two natural greedy algorithms – the greedy-rate and the greedy-margin algorithm – and in fact in uniform-cost settings such as that of [16] these two algorithms coincide as the rate of a facility is monotone in its marginal revenue. But for non-uniform facility costs, as the following simple example illustrates, the greedy-margin algorithm gives very poor results.

Example. There are n clients and $n + 1$ facilities. Facility i, $1 \leq i \leq n$, has cost 1, revenue 2 for client i, and 0 revenue for all other clients; Facility $n + 1$ has cost $n - 2$ and revenue 1 per client. The optimal solution will open the first n facilities for a value of n, whereas the greedy-margin rule will open only facility $n + 1$ for a value of 2.

By contrast, for the greedy-rate algorithm, the approximation ratio that we prove is strictly better than that proved in [16]. See Figure 2 for a detailed comparison of our bounds with those of [16].

5.2 Approximation as a function of α

In this section, we prove Corollary 2.10 which gives an approximation factor for the greedy rate algorithm as a function of the summary signature α. The following simple observation (appearing in [16]) is of key importance, and hence we state it as a lemma.

Lemma 5.2. *Let i be a facility and let S and T be sets of facilities. Then the marginal revenue of facility i with respect to S is at least as large as the loss in marginal revenue of T when facility i is added to S.*

$$M(i|S) \geq M(T|S) - M(T|S \cup \{i\})$$

Proof. By the fact that the revenue function is nondecreasing, we have $R(S \cup \{i\} \cup T) \geq R(S \cup T)$. Breaking each revenue to a sum of marginal revenues we have $R(S) + M(i|S) + M(T|S \cup \{i\}) \geq R(S) + M(T|S)$. Canceling the $R(S)$ and subtracting $M(T|S \cup \{i\})$ from both sides, the lemma is proved. □

We now proceed to prove our improved bounds. As stated, our analysis applies to any problem with a nondecreasing submodular revenue function and linear cost function.

Lemma 5.3. *The value of the greedy-rate algorithm is at least $R(O)(1-\alpha-\alpha\ln\frac{1}{\alpha})$ where O is an optimal solution and $\alpha = C(O)/R(O)$.*

Proof. We analyze the value of greedy-rate up to the first point in time in which its total revenue meets or exceeds $R(O) - C(O)$. Let $\mu(x)$ denote the rate at which the value obtained by the greedy-rate algorithm increases when it has already made a revenue of x. Observe that at a point when greedy has already made a revenue of $x < R(O) - C(O)$, the marginal revenue of O is at least $R(O) - x$ (by Lemma 5.2). By submodularity of the revenue function, at this point there must be at least one facility of O with rate $\frac{R(O)-x-C(O)}{R(O)-x}$. The rate at which the value increases at each point in time is at least as high as the rate one would get by choosing the highest rate among the facilities of O at the same time. Therefore, $\mu(x) \geq \frac{R(O)-x-C(O)}{R(O)-x}$. As the total value of greedy-rate, $V(G)$, is the integral of the rate of increase of the value, we have:

$$\begin{aligned}
V(G) = \int \mu(x)dx &\geq \int_0^{R(O)-C(O)} \frac{R(O)-x-C(O)}{R(O)-x}dx \\
&= R(O)\int_0^{1-\alpha} \frac{1-x-\alpha}{1-x}dx \\
&= R(O)\int_0^{1-\alpha} (1-\frac{\alpha}{1-x})dx \\
&= R(O)(1-\alpha-\alpha(\ln\frac{1}{1-(1-\alpha)} - \ln 1)) \\
&= R(O)(1-\alpha-\alpha\ln\frac{1}{\alpha})
\end{aligned}$$

□

The approximation ratio of Corollary 2.10 follows from Lemma 5.3 together with the fact that $V(O) = R(O)(1-\alpha)$. The NP-hardness results appear in Theorem 2.7, and they naturally provide examples where the approximation ratio of greedy-rate is no better than claimed even in the special case of linear revenue functions and uniform costs.

5.3 Approximation as a function of the signature

The approximation ratio in Corollary 2.10 is expressed as a function of the summary signature α, whereas stronger performance guarantees can be given by expressing the approximation ratio as a function of the signature $\{q_i, \alpha_i\}$, as stated in Theorem 2.9.

In fact, we prove Theorem 2.9 for the general submodular facility location problems. To do so, we should extend the notion of a signature to a solution for submodular maximum facility location. The difficulty is that even though the cost of every open facility is well defined, its revenue is not. Hence we refine the notion of a solution to be represented not as a set of open facilities, but as an ordered set (a tuple). Namely, the open facilities are given (after renaming) in some order $1, 2, \ldots$ (even though this order is irrelevant to the actual value of the solution). Thereafter, a refined parameter α'_i in the expanded signature is defined relative to the marginal revenue of facility i with respect to this order. That is, $\alpha'_i = c_i/M_i$, where here M_i is shorthand notation for $M(i|\{1,\ldots,i-1\})$. Likewise, we define $q'_i = M_i/\sum M_j$. Using this notation, we can now strengthen Lemma 5.3.

Lemma 5.4. *Let S be an arbitrary (ordered) solution for submodular maximum facility location with expanded signature $\{(q'_i, \alpha'_i)\}$ and total revenue normalized to 1. Then the value of the greedy-rate algorithm is at least $\sum_{i\in S} q'_i(1-\alpha'_i-\alpha'_i\ln\frac{1}{\alpha'_i})$.*

Proof. Let $G_t = \{g_1, g_2, \cdots, g_t\}$ be the set of facilities opened by the greedy-rate algorithm up to step t, $G_0 = \emptyset$. Also, let μ denote the rate of the increase in the value obtained by the greedy-rate algorithm.

Suppose we add facilities to G_t in the same order as in S. Define x_{ti} to be the marginal revenue obtained from facility i (according to the ordering in S). By Lemma 5.2, we have

$$M(g_t|G_{t-1}) \geq M(S|G_{t-1}) - M(S|G_{t-1} \cup \{g_t\}) = \sum_{i \in S}(x_{t-1,i} - x_{t,i}) \tag{1}$$

In other words, the increase in the revenue of the greedy-rate algorithm is at least equal to the decrease in the total marginal revenue from the facilities is S. The rate at which the value of the greedy-rate algorithm increases at each point in time is at least as high as the rate one would get by choosing the highest rate among the facilities of S. Therefore, when greedy opens facility g_{t+1}, we have $\mu \geq \frac{x_{ti}-c_i}{x_{ti}}$, for all $i \in S$. Also observer that by submodularity of the revenue function, as t increases, x_{ti} decreases. Also, x_{ti} decreases from M_i to c_i for the following reason. If facility i is opened by the greedy-rate algorithm, then $x_{t'i} = 0 \leq c_i$, where t' is the number of facilities opened by the greedy-rate algorithm. Otherwise, if facility i is not opened, then $x_{t'i} \leq c_i$. Therefore, by (1) we have:

$$\begin{aligned}
V(G) &= \int \mu(x)dx \\
&\geq \sum_{i \in S} \int_{c_i}^{M_i} \frac{x - c_i}{x} dx \\
&= \sum_{i \in S} M_i \int_{\alpha'_i}^{1} (1 - \frac{\alpha'_i}{x}) dx \\
&= R(S) \sum_{i \in S} q'_i [x - \alpha'_i \ln(x)]_{\alpha'_i}^{1} \\
&= R(S) \sum_{i \in S} q'_i (1 - \alpha'_i - \alpha'_i \ln \frac{1}{\alpha'_i})
\end{aligned}$$

□

To motivate the following lemma, observe that Lemma 5.4 by itself does not capture the notion of PASS approximation that we have for the special case of MFL. For a given set of open facilities in MFL, the optimal choice of allocation of clients might not correspond to revenues r_i per facility that are equal to M_i for any ordering of facilities. For example, if there are two facilities and two clients, where client i has revenue 2 if connected to facility i and revenue 1 if connected to the other facility, then in the optimal solution $r_1 = r_2 = 2$, whereas for any ordering $M_1 = 3$ and $M_2 = 1$.

Lemma 5.5. *Let S be an arbitrary solution for maximum facility location with expanded signature $\{(q_i, \alpha_i)\}$. Then there is an ordering of the facilities of S giving $\sum_{i \in S} q'_i(1-\alpha'_i-\alpha'_i \ln \frac{1}{\alpha'_i}) \geq \sum_{i \in S} q_i(1-\alpha_i-\alpha_i \ln \frac{1}{\alpha_i})$.*

Proof. Order the facilities of S in order of decreasing α_i and consider the resulting marginal revenues M_i. Any loss in original revenue r_i suffered by a facility can be matched against a marginal revenue gain for a facility i' with $i' < i$ in the given ordering. Such changes only increase the value of $\sum_{i \in S} q_i(1 - \alpha_i - \alpha_i \ln \frac{1}{\alpha_i})$. □

The combination of Lemmas 5.4 and 5.5 imply Theorem 2.9 (and also the generalization of Theorem 2.9 to SMFL).

6 A linear programming approach

In this section, we develop an LP-based approach for MFL. It is based on an interplay between the notions of the true value of a solution and the recoverable value of the solution. Recall that the recoverable value (see definition in Section 2), which in general is lower than the true value, represents our approximation goal in the sense that we wish to find a solution of true value at least equal to that of the recoverable value of the best integral solution. First we show that the natural LP for the problem does not provide the optimal PASS approximation ratio. Then we introduce a new LP relaxation for the general problem called the *recoverable value relaxation*. This LP captures the natural constraints for the MFL problem, but has an objective function describing the recoverable value of the solution rather than the true value. Hence the LP provides a fractional solution that maximizes the recoverable value, and we denote this value by $\hat{V}_f$. We round this fractional solution to an integral one of (expected) true value at least $\hat{V}_f$, thus meeting our approximation goal. Moreover, we can solve the recoverable value LP in polynomial time.

6.1 On rounding the natural LP

Consider the natural linear program for SB.

$$\begin{aligned} \text{maximize} \quad & \sum_j y_j u_j - \sum_i x_i c_i \\ \text{subject to} \quad & y_j \leq \sum_{i:j\in S_i} x_i \qquad j \in \mathcal{J} \\ & x_i \geq 0, y_j \leq 1 \qquad i \in \mathcal{F}, j \in \mathcal{J} \end{aligned} \tag{2}$$

In this section we show that this linear program does not give an optimal PASS approximation. We present two negative examples, each handling a different aspect of this issue. Define

$$\rho(S) = \frac{1 - \sum_i q_i\alpha_i - \sum_i q_i\alpha_i \ln \frac{1}{\alpha_i}}{1 - \sum_i q_i\alpha_i}$$

Rounding techniques. No rounding technique that uses only sets whose associated LP variable has strictly positive value ensures finding a solution of value (or expected value, for randomized rounding) at least $\rho(S)$ times the optimum value. Consider a random n-vertex graph of sufficiently high average degree d. Every edge is an item of revenue 1, every vertex i is a set of cost $d_i - 1$ (where d_i is the degree of vertex i) that covers the edges incident with it. In addition, there is one more set M of cost 0 that covers a perfect matching in the graph (containing $n/2$ edges that touch all vertices). The optimal solution is to take the set M, and its signature-value is 0, for which $\rho(0) = 1$. On the other hand, a possible optimal fractional solution to the LP will pick each vertex to an extent of $\frac{1}{2}$, giving a fractional value of $n/2$. Any rounding technique that uses only sets picked by the fractional solution will be limited to picking an independent set in the graph, and the maximum independent set is of size $O(\frac{n \log d}{d})$. Hence the approximation ratio would be small $O(\frac{\log d}{d})$ even though the signature-value of the instance requires that we find an exact solution.

Integrality gaps. The per instance integrality gap of the LP is worse than $1/\rho(S)$. Consider the graph-based instance of SB as the previous case, but instead of adding to it the set M, add to it $\frac{n}{\sqrt{d}}$ items and a set T of cost 0 which covers these items. Now the optimum solution includes set T (plus a maximum independent set in the graph), and it has value roughly $n/\sqrt{d}$, has a signature S with $\rho(S)$ very close to 1 (because the bulk of the value of the solution comes from a set of 0 cost). The fractional value of the LP is now larger than $n/2$, giving an integrality gap of $\Omega(\sqrt{d})$, which is much larger than $1/\rho(S)$.

6.2 An LP relaxation: recoverable value LP

Recall that in the MFL problem, each facility $i \in \mathcal{F}$ has an opening cost of c_i and each client $j \in \mathcal{J}$ has a revenue u_{ij} for being connected to facility i. We call pair (i, T) of a facility i and a subset T of clients connected to it a *star*. Let x_{iT} be an indicator variable of star (i, T), i.e., that facility i is opened and connected to clients $j \in T$. The revenue of connecting the clients in T to facility i is $r_{iT} = \sum_{j \in T} u_{ij}$. For every star (i, T) we associate a *recoverable value* which is $\hat{v}_{iT} = r_{iT}(1 - \alpha_{iT} - \alpha_{iT} \ln \frac{1}{\alpha_{iT}})$, where $\alpha_{iT} = \frac{c_i}{r_{iT}}$. Then the optimal fractional recoverable value is described by the following LP, called the *recoverable value LP relaxation*.

$$\begin{aligned} \text{maximize} \quad & \sum \hat{v}_{iT} x_{iT} && (3) \\ \text{subject to} \quad & \sum_{i,T: j \in T} x_{iT} \leq 1 && j \in \mathcal{J} \\ & \sum_{T \subseteq \mathcal{J}} x_{iT} \leq 1 && i \in \mathcal{F} \\ & x_{iT} \geq 0 && i \in \mathcal{F}, T \subseteq \mathcal{J} \end{aligned}$$

The first inequality guarantees that each client contributes revenue to at most one facility and the second inequality guarantees that each facility is opened at most once. Every integral solution satisfies these constraints. Hence the value of the LP is at least as large as the recoverable value of the best integer solution (the one maximizing the recoverable value).

Let $\hat{V}_f$ be the optimal fractional recoverable value, namely, the optimal value to the above LP. Let V_f be the fractional true value associated with this solution, namely $\sum (r_{iT} - c_i) x_{iT}$. Typically, LP-relaxations provide upper bounds for maximization problems. In contrast, it is not in general true that V_f provides an upper bound on the true value of the best integer solution. Instead, as Lemma 6.1 will show, $\hat{V}_f$ provides a lower bound.

Our LP has exponentially many variables; however, we can solve it using the ellipsoid method [13]. We solve the separation oracle of the dual linear program using a greedy algorithm. This algorithm exploits concavity of the recoverable values and some other structural properties of the dual. The technique developed here can be of independent interest.

Our randomized rounding procedure is composed of two steps: The first step considers facilities independently. Facilities of 0-cost are always opened. For the remaining facilities, $\alpha_{iT} > 0$. For each such facility i and each star (i, T) let $\beta_{iT} = x_{iT} \ln \frac{1}{\alpha_{iT}}$. Let $\beta_i = \sum_T \beta_{iT}$. We open facility i with probability $\min[\beta_i, 1]$. The first step might open several facilities with overlapping sets of clients. In the second step, we assign any over-demanded client j to the facility to which it contributes the maximum revenue. In the next two section, we prove Theorem 2.11.

6.3 Solving the recoverable value LP

Our LP has exponentially many variables. To show that it is solvable in polynomial time, we consider its dual.

$$\text{minimize} \qquad \sum_i \mu_i + \sum_j \nu_j \tag{4}$$

$$\text{subject to} \qquad \mu_i + \sum_{j \in T} \nu_j \geq \hat{v}_{iT} \qquad i \in \mathcal{F}, T \subset \mathcal{J} \tag{5}$$

$$\mu_i, \nu_j \geq 0 \qquad \forall i \in \mathcal{F}, j \in \mathcal{J}$$

The dual LP has exponentially many constraints but only a polynomial number of variables. Thus, if we find a separation oracle for it, we can solve it using the ellipsoid method [13]. As for the separation oracle for this LP, we need to solve the following problem: given vectors $< \mu_i | i \in \mathcal{F} >$ and $< \nu_j | j \in \mathcal{J} >$, we have to check if any of the constraints $\mu_i + \sum_{j \in T} \nu_j \geq \hat{v}_{iT}$ is violated. Rewriting the constraint, we need to check whether for any $i \in \mathcal{F}$ and $T \in \mathcal{J}$, the following inequality is violated.

$$\mu_i + \sum_{j \in T} \nu_j - r_{iT}(1 - \alpha_{iT} - \alpha_{iT} \ln \frac{1}{\alpha_{iT}}) \geq 0$$

Substituting α_{iT} by $\frac{c_i}{r_{iT}}$ and r_{iT} by $\sum_{j \in T} u_{ij}$, we get:

$$\mu_i + \sum_{j \in T} (\nu_j - u_{ij}) + c_i \ln(\sum_{j \in T} u_{ij}) \geq c_i(\ln c_i - 1)$$

Thus, for the separation oracle and facility i, it suffices that we show how to find the set T that *maximizes*

$$\sum_{j \in T} (u_{ij} - \nu_j) - c_i \ln(\sum_{j \in T} u_{ij})$$

We design an algorithm that maximizes the above expression. Consider first a related problem. Rather than deal with an expression that involves the difference between linear terms and logarithmic terms, replace the logarithmic term by a constraint. Namely, for some arbitrary value Q where $0 \leq Q \leq \sum_j u_{ij}$, require that $\sum_{j \in T} u_{ij} = Q$. This constraint enforces that $c_i \ln(\sum_{j \in T} u_{ij}) = c_i \ln Q$. Subject to this constraint, maximize $\sum_{j \in T} (u_{ij} - \nu_j)$. This constrained problem might not have a solution at all, because there may not be any subset of clients whose revenue sums to exactly Q. To overcome this problem, we allow for fractional solutions. Namely, a client j can be taken to an extent of $0 \leq \gamma_j \leq 1$, and then it contributes $\gamma_j(u_{ij} - \nu_j)$ to the sum and $\gamma_j u_{ij}$ to the constraint.

It is not difficult to see that the following greedy algorithm provides an optimal solution to the related problem introduced above. Sort all clients in decreasing order of $\frac{u_{ij} - \nu_j}{u_{ij}}$ (breaking ties arbitrarily). Pick items into T in this order until $\sum_{j \in T} u_{ij} \geq Q$. If the value of Q is reached with equality, we have an integer optimal solution. If the last client chosen causes the sum to exceed Q, then this last client needs to be chosen only fractionally (with a fraction that will make the sum equal to Q).

Observe that the only effect of Q on the greedy algorithm above is the stopping time. Hence processing the clients in decreasing order of $\frac{u_{ij} - \nu_j}{u_{ij}}$ is simultaneously optimal for all values of Q. Since the maximum of $\sum_{j \in T} (u_{ij} - \nu_j) - c_i \ln(\sum_{j \in T} u_{ij})$ occurs at some value of Q, the greedy algorithm (upon reaching this value of Q) will have at that point a fractional solution of value at least as high as this maximum (and possibly higher, because the greedy algorithm may have taken one client fractionally, whereas the true set T cannot do that).

To complete the proof, we observe that at the point (value Q) where the greedy algorithm maximizes $\sum_{j\in T}(u_{ij}-\nu_j) - c_i \ln(\sum_{j\in T} u_{ij})$, it must be the case that the greedy solution is integral. Let T' be the set just before the last client is added, and consider the last client k added by the greedy algorithm. The value of the expression to maximize can be expressed as a function of γ_k, the fraction of the last client added. This value is:

$$\sum_{j\in T'}(u_{ij}-\nu_j) + \gamma_k(u_{ik}-\nu_k) - c_i \ln(\gamma_k u_{ik} + \sum_{j\in T} u_{ij})$$

The second derivative of this expression is positive, implying that the maximum is attained when γ_k is either 0 or 1.

There is one more issue that we need to handle. Note that $\gamma = 0$ may correspond to an empty set T which is not a feasible answer for the separation oracle. In this case, if $\mu_i \geq c_i(\ln c_i - 1)$ then necessarily facility i is not involved in a violated constraint. However, if $\mu_i < c_i(\ln c_i - 1)$, it is still possible that facility i is involved in a violated constraint, and moreover, that the corresponding star does not contain the client j that maximizes $\frac{u_{ij}-\nu_j}{u_{ij}}$. In this case we must also check all singleton sets. (The following argument shows that checking singleton sets suffices in this case. The first derivative of the expression is maximized when trying to add client j, and must initially be negative. Assume for the sake of contradiction that it is not a singleton set that maximizes the expression. Then it must be the case the after some clients are added, the derivative changes to be positive for some client. But at that point the derivative must be positive for client j as well. So it must be the case that client j is part of the set of clients that maximizes the expression, and the greedy algorithm could not have made a mistake by adding it first.)

Hence to find a violated constraint in the dual one considers each facility i separately. (This is required because the revenue of a client j depends on the facility.) Then clients are sorted in order of decreasing $\frac{u_{ij}-\nu_j}{u_{ij}}$, the greedy algorithm presented above is run, and after every new client added by the greedy algorithm (obtaining say a set T) one checks whether $\mu_i + \sum_{j\in T}(\nu_j - u_{ij}) + c_i \ln r_{iT} \geq c_i(\ln c_i - 1)$. Additionally, due to boundary conditions, one must check for each j whether $\mu_i + \nu_j - u_{ij} + c_i \ln r_{iT} \geq c_i(\ln c_i - 1)$.

This shows that the dual linear program 4 can be solved in polynomial time using ellipsoid method.

6.4 Rounding the recoverable value LP

In this section we describe how to round the LP to get a feasible solution of true value at least equal to the LP-value. Our randomized rounding procedure is composed of two steps: The first step considers facilities independently. Facilities of 0-cost are always opened. For the remaining facilities, $\alpha_{iT} > 0$. For each such facility i and each star (i,T) let $\beta_{iT} = x_{iT} \ln \frac{1}{\alpha_{iT}}$. Let $\beta_i = \sum_T \beta_{iT}$. We open facility i with probability $\min[\beta_i, 1]$. The first step might open several facilities with overlapping sets of clients. In the second step, we assign any over-demanded client j to the facility to which it contributes the maximum revenue, i.e., assign any over-demanded client j to the facility $i^* = \text{argmax}_{\text{open } i\in\mathcal{F}}\{u_{ij}\}$.

Lemma 6.1. *Consider an optimal fractional solution of LP (3), with fractional recoverable value $\hat{V}_f$. For the MFL problem, our randomized rounding technique achieves an integral solution of expected (true) value at least $\hat{V}_f$.*

Proof. Consider an arbitrary star (i,T) that has fractional value $0 \leq x_{iT} \leq 1$ in the LP solution, and has revenue $r_{iT} = \sum_{j\in T} u_{ij}$ and cost c_i. Let $\alpha_{iT} = \frac{c_i}{r_{iT}}$ be the signature associated with (i,T). Note that in general $0 \leq \alpha_{iT} \leq 1$, and without loss of generality, we may assume that both inequalities are strict. The

value $\hat{V}_f$ of the recoverable value LP can be decomposed to:

$$\hat{V}_f = \sum_{i,T} x_{iT}\hat{V}_{iT} = \sum_{i,T}(x_{iT}r_{iT}(1-\alpha_{iT}) - x_{iT}r_{iT}\alpha_{iT}\ln\frac{1}{\alpha_{iT}})$$

Thus the contribution of star (i,T) to $\hat{V}_f$ $x_{iT}r_{iT}(1-\alpha_{iT}) - x_{iT}r_{iT}\alpha_{iT}\ln\frac{1}{\alpha_{iT}}$. We now show that the contribution of star (i,T) to the rounded solution is at least as large.

The expected cost incurred by star (i,T) in the rounded solution is $\beta_{iT}c_i = x_{iT}r_{iT}\alpha_{iT}\ln\frac{1}{\alpha_{iT}}$. This means that the approximation ratio can be met if the expected revenue from (i,T) is at least $x_{iT}r_{iT}(1-\alpha_{iT})$. That is conditioned on (i,T) being chosen, we want its revenue to be $x_{iT}r_{iT}(1-\alpha_{iT})/\beta_{iT} = r_{iT}(1-\alpha_{iT})/\ln(\frac{1}{\alpha_{iT}})$. To do this, we use an approach inspired by [6, 12]. We show that for every client, there exists a randomized contention resolution rule that guarantees that each client in (i,T) is credited to i with probability at least $(1-\alpha_{iT})/\ln(\frac{1}{\alpha_{iT}})$ conditioned on (i,T) being chosen, which translated back to an unconditional probability of at least $x_{iT}(1-\alpha_{iT})$. Our Step-2 rule (assigning any over-demanded client j to the open star for which it contributes the maximum revenue) is the best contention resolution technique. It follows that it also retains sufficient revenue.

Consider an arbitrary client Z and stars used by the feasible fractional solution that contain Z. To show that a contention resolution rule exists, we use the max-flow min-cut theorem. Consider the following bipartite graph. The left hand side (lhs) contains one vertex (i,T) for each star. The right hand side (rhs) contains a vertex for each collection of stars. An lhs vertex is connected to a rhs vertex if the respective star is a member of the respective collection. The capacity of the edge is set to be infinite. Now, add two special vertices s and t. Vertex s is connected to all lhs vertices, where the capacity of an edge connecting it to vertex i is exactly $x_{iT}(1-\alpha_{iT})$, the probability with which we need to give client Z to star (i,T). All rhs vertices are connected to t, where the capacity of an edge is exactly the probability that the respective collection is chosen by the randomized rounding (note this probability may be zero, for example if the collection contains two stars with the same center). Now if there exists a flow from s to t that saturates all edges going out of s then this gives the desired contention resolution technique. (Namely, the amount of flow from a lhs vertex to a rhs vertex is the joint probability of the rhs collection being chosen and the lhs set getting client Z.) We claim that such a saturated flow always exists. To show this, we need to prove that the cut that separates s from all lhs vertices is a minimum (s,t)-cut in the graph. Observe that every minimum cut will only cut edges incident with s or with t, as other edges have infinite capacity. So it remains to show that for every set $\mathcal{T}$ of lhs vertices, the sum of capacities of edges from s to this set is no larger than the sum of capacities of edges from their neighbors to t. We call the sum of capacities of edges from s to $\mathcal{T}$ the "demand" of $\mathcal{T}$ and the sum of capacities of edges from the neighbors of $\mathcal{T}$ to t the "supply" for $\mathcal{T}$. Observe that the supply for $\mathcal{T}$ is exactly the probability that the rounded solution picks at least one of the stars in $\mathcal{T}$. This probability is complicated to write down due to the dependence in the rounding. Hence, we first consider another rounding in which each star is selected independently with probability $x_{iT}\ln\frac{1}{\alpha_{iT}}$ and then show that our dependent rounding can only improve the situation.

The independent rounding scenario leads to proving the following inequality:

$$\sum_{(i,T)\in\mathcal{T}} x_{iT}(1-\alpha_{iT}) \leq 1 - \prod_{(i,T)\in\mathcal{T}}(1 - x_{iT}\ln\frac{1}{\alpha_{iT}}) \tag{6}$$

Observe that there are two constraints involved. One is that $\sum_{(i,T)\in\mathcal{T}} x_{iT} \leq 1$, which is a constraint in our LP. The other is that for every (i,T), $x_{iT}\ln\frac{1}{\alpha_{iT}} \leq 1$. Note that for the MFL problem we can duplicate facilities without changing the problem. In particular, given two stars (i,T) and (i,T') that refer to the same

facility i, we may create a clone i' of facility i and replace (i, T') by (i', T'). Hence we may think of stars as referring to distinct facilities. If for some facility i (that now has only one star (i, T) associated with it) we have that $\beta_i > 1$, we can simply create $M = \lceil \beta_i \rceil$ clones $i_1, \ldots, i_M$ of facility i, assign weight x_{iT}/M to each clone (i_j, T), and treat clones as independent facilities in the rounding. This new fractional solution has the same LP-value $\hat{V}_f$ as the original fractional solution. If the rounding happens to open multiple clones of the same facility, then we close all but one and assign all clients from each clone to the remaining open facility. This reduces the cost but does not change the revenue. So we can assume for every (i, T), $x_{iT} \ln \frac{1}{\alpha_{iT}} \leq 1$.

To see that inequality (6) holds introduce a new variable $x_0 = 1 - \sum_{(i,T)\in\mathcal{T}} x_{iT}$ insuring that $\sum x_{iT} = 1$ and a corresponding $\alpha_0 = 1$. The inequality becomes

$$\sum x_{iT}\alpha_{iT} \geq \prod (\alpha_{iT})^{x_{iT}} = \prod e^{x_{iT}\ln(\alpha_{iT})} \geq \prod (1 - x_{iT} \ln \frac{1}{\alpha_{iT}})$$

which holds by the arithmetic-geometric mean inequality.

We now argue that in the dependent rounding scenario, the supply for any set T can only be larger than in the independent rounding scenario. To see this, fix a facility i and consider what happens when we round it according to the dependent rule. We are interested in the event that at least one of the stars containing i is selected. Let $E(i, T)$ be the event that star (i, T) is selected. The events $E(i, T)$ have the same probability in the independent and dependent rounding scenarios, but in the dependent rounding scenario, the events are disjoint. Hence, the probability that at least one event $E(i, T)$ happens is higher in the dependent rounding scenario. As the rounding decision regarding facility i is independent of the rounding decision regarding facility i', this observation can be applied to all facilities separately and hence the supply for any set $\mathcal{T}$ is larger in the dependent rounding scenario than the independent rounding scenario. □

Acknowledgements

The work of the first author was supported in part by The Israel Science Foundation (grant No. 873/08). We would like to acknowledge the anonymous referees for the their useful comments and suggestions.

References

[1] A. Ageev and M. Sviridenko. An 0.828-approximation algorithm for uncapacitated facility location problem. *Discrete Applied Mathematics*, 93:289–296, 1999.

[2] A. Blum and J. Spencer. Coloring random and semi-random k-colorable graphs. *Journal of Algorithms*, 19(2):204–234, 1995.

[3] G. Cornuejols, G. Nemhauser, and L. Wolsey. Locations of bank accounts to optimize float: an analytic study of exact and approximate algorithms. *Management Science*, 23:789–810, 1977.

[4] G. Cornuejols, G. Nemhauser, and L. Wolsey. The uncapacitated facility location problem. In *Discrete Location Theory*, pages 119–171. Wiley, 1990.

[5] U. Feige. A threshold of ln for approximating set cover. *Journal of ACM*, 45(4):634–652, 1998.

[6] U. Feige. On maximizing welfare when utility functions are subadditive. In *Proceedings of the 38th Annual ACM Symposium on Theory of Computing (STOC)*, pages 41–50, 2006.

[7] U. Feige, N. Immorlica, V. Mirrokni, and H. Nazerzadeh. A combinatorial allocation mechanism with penalties for banner advertising. *Proceedings of the 17th International World Wide Web Conference (WWW)*, 2008.

[8] U. Feige, N. Immorlica, V. Mirrokni, and H. Nazerzadeh. PASS Approximation: a framework for analyzing and designing heuristics. In *Proceedings of 19th International Workshop Approximation Algorithms for Combinatorial Optimization (APPROX)*, 2009.

[9] U. Feige and J. Killian. Heuristics for semirandom graph problems. *Journal of Computer and System Sciences*, 63:639–671, 2001.

[10] U. Feige, L. Lovász, and P. Tetali. Approximating min-sum set cover. In *Proceedings of 5th International Workshop Approximation Algorithms for Combinatorial Optimization (APPROX)*, pages 94–107, 2002.

[11] U. Feige, V. Mirrokni, and J. Vondrak. Maximizing non-monotone submodular functions. In *Proceedings of the 48th annual IEEE symposium on Foundations of Computer Science (FOCS)*, pages 461–471, 2007.

[12] U. Feige and J. Vondrák. Approximation algorithms for allocation problems: Improving the factor of 1 - 1/e. In *Proceedings of 47th Annual IEEE Symposium on Foundations of Computer Science (FOCS)*, pages 667–676, 2006.

[13] M. Grötschel, L. Lovasz, and A. Schrijver. *Geometric Algorithms and Combinatorial Optimization (Algorithms and Combinatorics)*. Springer-Verlag, 1988.

[14] J. Hastad. Clique is hard to approximate within $n^{1-\varepsilon}$. *Acta Mathematica*, 182, 105-142, 1999.

[15] D. Kempe, J. Kleinberg, and Éva Tardos. Maximizing the spread of influence through a social network. In *KDD '03: Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 137–146, New York, NY, USA, 2003. ACM.

[16] J. M. Kleinberg, C. H. Papadimitriou, and P. Raghavan. Segmentation problems. *J. ACM*, 51(2):263–280, 2004.

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

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

[19] D. Spielman and S. Teng. Smoothed analysis: Why the simplex algorithm usually takes polynomial time. *Journal of the ACM*, 51:385–463, 2004.

[20] D. Zuckerman. Linear Degree Extractors and the Inapproximability of Max Clique and Chromatic Number. *Theory of Computing* 3(1): 103–128, 2007.

[21] U. Zwick. Outward Rotations: a tool for rounding solutions of semidefinite programming relaxations, with applications to MAX CUT and other problems. *STOC*, 679–687, 1999.

A Special cases of SMFL

In this section we list some important special cases of submodular maximum facility location (SMFL). For completeness, we first restate the definition of SMFL.

Problem A.1. Submodular Maximum Facility Location (SMFL). *Consider a set N of n facilities and a set function $f : 2^N \to R^+$. For any subset $S \subset N$, $f(S) = R(S) - c(S)$, where R is a non-negative non-decreasing submodular set function corresponding to the revenue, and $c(S) = \sum_{i \in S} c_i$ is a linear cost function. As a result, set function f is a non-monotone submodular function and the goal is to find a subset S that maximizes $f(S)$.* [6] *We assume a value oracle for the revenue function R and a description for the cost c (this is of polynomial size) are given.*

The most important special case, and the one we use to illustrate our results, is the maximum facility location (MFL) in which the revenue is defined by a matching.

Problem A.2. Maximum Facility Location (MFL). *A set $\mathcal{F}$ of m facilities is given. For every facility i, there is an opening cost of c_i. There is also a set $\mathcal{J}$ of n clients. The revenue of connecting client j to facility i is $u_{ij} \geq 0$ (this may be interpreted as a client revenue minus a connection cost). Every client can connect to at most one open facility (or none). The goal in MFL is to open some facilities and connect clients to them so as to maximize the total revenue from the connected clients minus the total cost of the opened facilities.*

For comparison with some previous work [16], we shall discuss also the following problem that [16] call the variable catalog segmentation problem.

Problem A.3. Catalog Segmentation Problem. *A company has a collection of products and a collection of potential clients. Clients have various levels of interest associated with each type of product. The company wishes to produce several types of catalogs, each type containing a subset of the products (the number of products in a catalog may be limited by considerations such as weight), and mail to every potential client at most one catalog (presumably, of a type that would be of interest to the client). Assuming that producing a catalog-type has unit cost, and that for each type i and client j there is a expected revenue of u_{ij} from mailing a catalog of type i to client j, which catalogs should the company produce in order to maximize its expected profit (expected benefit minus production cost)? If all potential types of catalogs can be listed beforehand and all values u_{ij} are known, then this is a special case of MFL, with the catalogs serving as facilities. (In [16] it is assumed that all types of catalogs cost the same to produce, and we follow this assumption in our presentation. More generally, we may associate a cost c_i for producing the catalog of type i, and then the problem becomes equivalent to MFL.)*

The MFL problem is also referred to as *uncapacitated facility location* (see [1] for example). More generally, each facility may be limited in the number of clients it can serve. This problem is called the *capacitated maximum facility location problem (CMFL).*

Problem A.4. Capacitated Maximum Facility Location (CMFL) *This problem is identical to MFL except that each facility i additionally has a capacity constraint k_i, meaning that in a feasible solution, the number of clients connected to facility i is at most k_i.*

Although not entirely obvious, we show in Section A.1 that CMFL is a special case of SMFL.

The CMFL has many important applications, including, for example, the allocation of banner advertisements in online advertising.

[6] Note that function f can be possibly negative and therefore the result of Feige et al. [11] does not apply.

Problem A.5. Banner Ad Allocation. *In an instance of this problem, there is a set $\mathcal{F}$ of m advertisers, and a set $\mathcal{J}$ of n ad opportunities (or* ads*, for short). Each advertiser $i \in \mathcal{F}$ is interested in a subset of ads $T_i \subseteq \mathcal{J}$, and is associated with a bid (willingness to pay) value b_i, and a desired number of ads d_i. We should find a subset of advertisers and assign ads to them. We are also given a* penalty parameter*, α as follows: if we assign a set $X_i \subset T_i$ to advertiser i where $|X_i| \leq d_i$, the net profit that we collect from advertiser i is $|X_i|b_i - \alpha b_i(d_i - |X_i|) = (1+\alpha)b_i|X_i| - \alpha b_i d_i$. Our goal is to accept and serve a set of advertisers $\mathcal{W}$, and assign ads to them to maximize the total revenue.*

The banner ad allocation problem is a special case of CMFL. Each bid i can be thought of as a facility of capacity d_i, opening cost $\alpha b_i d_i$ and revenue $(1+\alpha)b_i$ per item (and 0 for items that the bidder is not interested in).

Another special case of SMFL, which is not a special case of CMFL is the following problem.

Problem A.6. Influence Maximization. *The goal in this problem is to choose an initial set of people such that their adoption of a new product or technology spreads over a social network. For various random influence dynamics, the expected revenue obtained from the spread of influence has been proved to be a non-decreasing submodular function of the set of initial people [18, 15]. Assuming that there exists a cost c_i for motivating person i to adopt the new product, the problem is defined as finding a set of initial people such that the revenue obtained from spread of influence from these people minus the cost incurred to motivate these people is maximized.*

Other special cases of SMFL include a variety of optimization problems such as set buying and optimal sensor installation for outbreak detection [17].

A.1 Submodularity of CMFL

In this section we prove the following proposition.

Proposition A.7. *The revenue $R(.)$ of CMFL is a submodular function. Namely, for every two sets S and T of facilities, $R(S) + R(T) \geq R(S \cap T) + R(S \cup T)$. Equivalently, for every facility i and every two sets of facilities S and T with $S \subset T$, $M(i|S) \geq M(i|T)$.*

Before proving Proposition A.7, let us remove a possible misunderstanding regarding the statement of the proposition. The marginal revenue of a facility does not refer only to the revenue from clients served by the facility. It takes into account also the revenue change to other facilities. The following simple example illustrates this distinction.

Example. There are two clients, a_1 and a_2, and three facilities 1, 2, and 3. We set the revenues as follows. $u_{11} = 2$, $u_{12} = 0$, $u_{21} = 3$, $u_{22} = 2$, $u_{31} = 0$, $u_{32} = 3$. All capacities are 1. Let $S = \{1\}$ and $T = \{1, 3\}$. Then $M(2|S) = 2$ (client a_2 will be allocated to facility 2), and $M(2|T) = 1$ (client a_1 transfers from facility 1 to facility 2), even though facility 2 shows a revenue of 3 in this latter case.

If there are no capacity constraints, the proof of Proposition A.7 is straightforward. Given a set of open facilities, every client connects to the facility that offers maximum revenue to that client. As more facilities open, the revenue of each client is a nondecreasing function. Hence the possible benefit of a new facility cannot increase if more facilities are open.

When there are capacity constraints, the situation becomes more complicated. Clients can no longer greedily choose which facility to connect to. Rather, a matching problem needs to be solved. As a consequence, the revenue of a client might decrease when a new facility is opened. This is illustrated in the following example.

Example. There are two clients a_1 and a_2 and two facilities 1 and 2, each of capacity 1. The revenues are $u_{11} = 3$, $u_{12} = 2$, $u_{21} = 2$, $u_{22} = 0$. Opening facility 1, the revenue of a_1 is 3. Then, opening facility 2, a_1 is transferred and its revenue decreases to 2, whereas the revenue of a_2 increases from 0 to 2.

Proof. [Proposition A.7] Let S and T be two arbitrary sets of facilities. Consider the following bipartite multi-graph (we will allow parallel edges). The set V_1 of left hand side vertices contains one vertex for every facility in $S \cup T$. The set V_2 of right hand side vertices contains one vertex for every client. The graph contains two types of edges, white and black. The white edges corresponds to an optimal (or arbitrary, this does not matter for the proof) legal assignment of clients to the facilities in $S \cup T$ (each client is assigned to at most one facility, the number of clients served by a facility does not exceed the facility's capacity). Thus a white edge connects facility i to client j iff i serves client j. The black edges correspond in a similar way to an optimal (or arbitrary) legal assignment of clients to facilities in $S \cap T$. Hence every right hand side vertex is connected to at most one white edge and at most one black edge, and every left hand side vertex of capacity k_i is connected to at most k_i edges of each color. To prove submodularity of the revenue function, it suffices to show that all edges can be recolored in red and blue in a way corresponding to legal assignments of clients to facilities in S and in T respectively. Namely, every client is connected to at most one red edge and at most one blue edge, every facility in $S \setminus T$ is connected only to red edges, every facility in $T \setminus S$ is connected only to blue edges, and for every facility $i \in S \cap T$, neither the number of red edges not the number of blue edges connected to it exceed its capacity k_i.

We now show an algorithm that obtains a legal red/blue coloring from a legal white/black coloring. Observe that all edges incident with $S \setminus T$ are initially colored white, and likewise with edges incident with $T \setminus S$. The algorithm will proceed in steps. In every step, the algorithm will take a maximal alternating white/black path or an alternating white/black cycle and color all its edges red/blue in an alternating way. We show that this can be done in a way that eventually gives a legal red/blue coloring.

Given an alternating white/black cycle, color its white edges red and its black edges blue. All clients on the cycle receive one red edge and one blue edge and hence become legally colored. All facilities on the cycle must belong to $S \cap T$ and hence are allowed to have both red and blue edges. Moreover, the cycles result in the same number of red and blue edges in a facility, and hence capacity constraints cannot be exceeded in any color.

Given a maximal alternating white/black path (maximal in the sense that it cannot be extended in either direction), check whether one of its endpoints lies in $S \setminus T$. If yes, the edge incident with it must have been colored white. Color all white edge red and all black edges blue. Else (if no endpoint lies in $S \setminus T$), color all white edge blue and all black edge red. All clients on the path receive at most one red edge and one blue edge, and all their edges are exhausted (because the path is maximal and initially they had at most one white edge and at most one black edge). Hence they are legally colored. Vertices in $S \setminus T$ can appear only as endpoints of the path (as they are not incident with black edges). Hence they are legally colored. To see that the same applies to vertices in $T \setminus S$, one needs to consider the case that (at least) one of the endpoints lies in $T \setminus S$. The only source of trouble in this case might have been if the other endpoint lies in $S \setminus T$ (because then all white edges are colored red, and this is illegal for vertices in $T \setminus S$). However, a simple parity argument shows that this cannot happen without having a black edge enter a vertex in $T \setminus S$, but there are no such black edges. Last, for vertices in $S \cap T$ we need to check that capacity constraints are not exceeded in $S \cap T$. Note that as long that such a vertex has both white and black edges incident with it, it receives the same number of red and blue edges (because paths are maximal). Hence if W, B denote the numbers of white and black edges it started with, and r, b denote the numbers of red and blue edges it ended with (hence $W + B = r + b$), we have that $\min[r, b] \geq \min[W, B]$ implying that $\max[r, b] \leq \max[W, B]$, and no capacity constraint can be exceeded. □