

Monotone Submodular Maximization over a Matroid via Non-Oblivious Local Search

Yuval Filmus and Justin Ward

November 25, 2012

Abstract

We present an optimal, combinatorial $1 - 1/e$ approximation algorithm for monotone submodular optimization over a matroid constraint. Compared to the continuous greedy algorithm (Calinescu, Chekuri, Pál and Vondrák, 2008), our algorithm is extremely simple and requires no rounding. It consists of the greedy algorithm followed by local search. Both phases are run not on the actual objective function, but on a related auxiliary potential function, which is also monotone submodular.

In our previous work on maximum coverage (Filmus and Ward, 2011), the potential function gives more weight to elements covered multiple times. We generalize this approach from coverage functions to arbitrary monotone submodular functions. When the objective function is a coverage function, both definitions of the potential function coincide.

Our approach generalizes to the case where the monotone submodular function has restricted curvature. For any curvature c , we adapt our algorithm to produce a $(1 - e^{-c})/c$ approximation. This matches results of Vondrák (2008), who has shown that the continuous greedy algorithm produces a $(1 - e^{-c})/c$ approximation when the objective function has curvature c , and proved that achieving any better approximation ratio is impossible in the value oracle model.

1 Introduction

In this paper, we consider the problem of maximizing a monotone submodular function f , subject to a single matroid constraint. Formally, let \mathcal{U} be a set of elements and let $f: 2^{\mathcal{U}} \rightarrow \mathbb{R}$ be a function assigning a value to each subset of \mathcal{U} . We say that f is *submodular* if

$$f(A) + f(B) \geq f(A \cup B) + f(A \cap B)$$

for all $A, B \subseteq \mathcal{U}$. If additionally, $f(A) \leq f(B)$ whenever $A \subseteq B$, we say that f is *monotone submodular*. Submodular functions exhibit (and are, in fact, alternately characterized by) the property of diminishing returns—if f is submodular then $f(A \cup \{x\}) - f(A) \leq f(B \cup \{x\}) - f(B)$ for all $B \subseteq A$. Hence, they are useful for modeling various economic and game-theoretic scenarios, as well as various combinatorial problems. In a general monotone submodular maximization problem, we are given a value oracle for f and a membership oracle for some distinguished collection $\mathcal{I} \subseteq 2^{\mathcal{U}}$ of *feasible sets*, and our goal is to find a member of \mathcal{I} that maximizes the value of f . We assume further that f is *normalized* so that $f(\emptyset) = 0$.

We consider the restricted setting in which the collection \mathcal{I} forms a matroid. Matroids are intimately connected to combinatorial optimization: the problem of optimizing a linear function over a hereditary set system (a set system closed under taking subsets) is solved optimally for all possible functions by the standard greedy algorithm if and only if the set system is a matroid [27, 8].

In the case of a *monotone submodular* objective function, the standard greedy algorithm, which takes at each step the element yielding the largest increase in f while maintaining independence, is (only) a 2-approximation [17]. Recently, Calinescu et al. [5, 28, 6] have developed a $(1 - 1/e)$ -approximation for this problem via the *continuous greedy algorithm*, which is essentially a steepest descent algorithm running in

continuous time (in practice, a suitably discretized version is used), producing a fractional solution. The fractional solution is rounded using pipage rounding [1] or swap rounding [7].

Feige [9] has shown that improving the bound $(1 - 1/e)$ is NP-hard. Nemhauser and Wolsey [24] have shown that any improvement over $(1 - 1/e)$ requires an exponential number of queries in the value oracle setting.

Following Vondrák [29], we also consider the case when f has restricted curvature. We say that f has *restricted curvature* c with respect to \mathcal{I} if for any two non-empty and disjoint independent sets A, B ,

$$f(A \cup B) \geq f(A) + (1 - c)f(B).$$

When $c = 1$, this is a restatement of monotonicity. Vondrák [29] has shown that the continuous greedy algorithm produces a $(1 - e^{-c})/c$ approximation when f has restricted curvature c . Furthermore, he has shown that any improvement over $(1 - e^{-c})/c$ requires an exponential number of queries in the value oracle setting.

Our definition of curvature differs from the usual (non-restricted) definition of curvature in that we only consider sets A and B that are independent, while the general definition requires that the above inequality hold for all (not necessarily independent) sets A and B . Therefore, every function with curvature at most c also has restricted curvature at most c and so our algorithm applies to a *wider* class of functions than those with non-restricted curvature at most c .

1.1 Our contribution

We propose a conceptually simple randomized polynomial time local search algorithm for the problem of monotone submodular maximization over a matroid. Like the continuous greedy algorithm, our algorithm delivers the optimal $(1 - 1/e)$ -approximation. However, unlike the continuous greedy algorithm, our algorithm is entirely combinatorial, in the sense that it deals only with integral solutions to the problem and hence involves no rounding procedure. As such, we believe that the algorithm may serve as a gateway to further improved algorithms in contexts where pipage rounding and swap rounding break down, such as submodular maximization subject to multiple matroid constraints.

Our main result is a $(1 - 1/e)$ approximation algorithm for monotone submodular maximization over a matroid. For a function with curvature c , the algorithm can be generalized to a $(1 - e^{-c})/c$ approximation algorithm, assuming the value of c is known in advance. We also obtain a $(1 - e^{-c})/c - \epsilon$ algorithm which does not require knowledge of c .

Our algorithmic approach is based on *non-oblivious local search*, a technique first proposed by Alimonti [2] and by Khanna, Motwani, Sudan and Vazirani [20]. In classical (or, *oblivious*) local search, the algorithm starts at an arbitrary solution, and proceeds by iteratively making small changes that improve the objective function, until no such improvement can be made. The *locality ratio* of a local search algorithm is $\min f(S)/f(O)$, where S is a solution that is locally-optimal with respect to the small changes considered by the algorithm, O is a global optimum, and f is the objective function. The locality ratio provides a natural, worst-case guarantee on the approximation performance of the local search algorithm.

In many cases, oblivious local search may have a very poor locality ratio, implying that a locally-optimal solution may be of significantly lower quality than the global optimum. For example, for monotone submodular maximization over a matroid, the locality ratio for an algorithm changing a single element at each step is $1/2$ [17]. *Non-oblivious* local search attempts to avoid this problem by making use of a secondary potential function to guide the search. By carefully choosing this auxiliary function, we ensure that poor local optima with respect to the original objective function are no longer local optima with respect to the new potential function.

In previous work [14], we designed an optimal non-oblivious local search algorithm for the restricted case of maximum coverage subject to a matroid constraint. In this problem, we are given a weighted universe of elements, a collection of sets, and a matroid defined on this collection. The goal is to find a collection of sets that is independent in the matroid and covers elements of maximum total weight. The non-oblivious potential function used in [14] gives extra weight to solutions that cover elements multiple times. In the present work,

we extend this approach to general monotone submodular functions. This presents two challenges: defining a non-oblivious potential function without reference to the coverage representation, and analyzing the resulting algorithm.

In order to define the general potential function, we construct a variant of the potential function from [14] which doesn't refer to elements. Instead, the potential function aggregates the information obtained by applying the objective function to all subsets of the input, weighted according to their size. Intuitively, the resulting potential function gives extra weight to solutions that contain a large number of good sub-solutions, or equivalently, remain good solutions on average even when elements are randomly removed. An appropriate setting of the weights defining our potential function yields a function which coincides with the previous definition for coverage functions, but still makes sense for arbitrary monotone submodular functions.

The analysis of the algorithm in [14] is relatively straightforward. For each type of element in the universe of the coverage problem, we must prove a certain inequality among the coefficients defining the potential function. In the general setting, however, we need to construct a proof using only the inequalities given by monotonicity and submodularity. The resulting proof is non-obvious and delicate.

This paper extends and simplifies previous work by the same authors. The paper [15], appearing in FOCS 2012, only discusses the case $c = 1$. The general case is discussed in [16], which appears in ArXiv. The present paper simplifies the work appearing in [16]. This simplification comes at the cost of slightly inferior results. The slightly superior results obtained in [15] and [16] are discussed in §A.2.

An exposition of the ideas of both [14] and [16] can be found in the second author's thesis [31]. In particular, the thesis explains how the auxiliary objective function can be determined by solving a linear program, both in the special case of maximum coverage and in the general case of monotone submodular functions with restricted curvature.

1.2 Related work

Fisher, Nemhauser and Wolsey [25, 17] analyze greedy and local search algorithms for submodular maximization subject to various constraints, including single and multiple matroid constraints, and obtain some of the earliest results in the area, including a $1/(k+1)$ -approximation for monotone submodular maximization subject to k matroid constraints. A recent survey by Goundan and Schulz [19] reviews many results pertaining to the greedy algorithm for submodular maximization.

More recently, Lee, Sviridenko and Vondrák [23] consider the problem of both monotone and non-monotone submodular maximization subject to multiple matroid constraints, attaining a $1/(k+\epsilon)$ -approximation for monotone submodular maximization subject to $k \geq 2$ constraints using local search. Feldman et al. [13] show that a local search algorithm attains the same bound for the related class of k -exchange systems, which includes the intersection of k strongly base orderable matroids, as well as the independent set problem in $(k+1)$ -claw free graphs. Further work by Ward [30] shows that a non-oblivious local search routine attains an improved $2/(k+3) - \epsilon$ approximation for this class of problems.

In the case of unconstrained non-monotone maximization, Feige, Mirrokni and Vondrák [10] give a $2/5$ approximation via a randomized local search algorithm, and give an upper bound of $1/2$ in the value oracle model. Gharan and Vondrák [18] improved the algorithmic result to 0.41 by enhancing the local search algorithm with ideas borrowed from simulated annealing. Feldman, Naor and Schwarz [12] later improved this to 0.42 by using a variant of the continuous greedy algorithm. Buchbinder, Feldman, Naor and Schwartz have recently obtained an optimal $1/2$ approximation algorithm [4].

In the setting of constrained non-monotone submodular maximization, Lee et al. [22] give a $1/(k+2+1/k+\epsilon)$ approximation subject to k matroid constraints and a $1/5 - \epsilon$ approximation for k knapsack constraints. Further work by Lee, Sviridenko and Vondrák [23] improves the approximation ratio in the case of k matroid constraints to $1/(k+1+1/(k-1)+\epsilon)$. Feldman et al. [13] attain this ratio for k -exchange systems. In the case of non-monotone submodular maximization subject to a *single* matroid constraint, Feldman, Naor and Schwarz [11] obtain a $1/e$ approximation by using a version of the continuous greedy algorithm. They additionally unify various applications of the continuous greedy and obtain improved approximations for non-monotone submodular maximization subject to a matroid constraint or $O(1)$ knapsack constraints.

1.3 Organization of the paper

Some basic definitions are included in §2. The auxiliary objective function we use is defined in §3, and the issue of evaluating it is discussed in §4. The main algorithm is described and analyzed in §5, and some simple extensions are described in §6. Auxiliary objective functions resulting in a better locality ratio are discussed in §A. The definition of the auxiliary objective function is demystified in §B. A combinatorial version of the continuous greedy algorithm appears in §C.

2 Definitions

Notation If B is some Boolean condition, then

$$\llbracket B \rrbracket = \begin{cases} 1 & \text{if } B \text{ is true,} \\ 0 & \text{if } B \text{ is false.} \end{cases}$$

For n a natural number, $[n] = \{1, \dots, n\}$. We use H_k to denote the k th Harmonic number,

$$H_k = \sum_{t=1}^k \frac{1}{t}.$$

It is well-known that $H_k = \log n + \gamma + O(1/n)$, where $\log n$ is the natural logarithm.

For S a set and x an element, $S + x = S \cup \{x\}$ and $S - x = S \setminus \{x\}$. We will use $S + x$ even when $x \in S$, and $S - x$ even when $x \notin S$.

Let \mathcal{U} a set. A *set-function* f on \mathcal{U} is a function $f: 2^{\mathcal{U}} \rightarrow \mathbb{R}$ whose arguments are subsets of \mathcal{U} . For $x \in \mathcal{U}$, we use $f(x) = f(\{x\})$. For $A, B \subseteq \mathcal{U}$, the *marginal* of B with respect to A is

$$f_A(B) = f(A \cup B) - f(A).$$

Properties of set-functions A set-function f is *normalized* if $f(\emptyset) = 0$. It is *monotone* if whenever $A \subseteq B$ then $f(A) \leq f(B)$. It is *submodular* if whenever $A \subseteq B$ and C is disjoint from B , $f_A(C) \geq f_B(C)$. If f is monotone, we need not assume that B and C are disjoint. Submodularity is also characterized by the submodular inequality

$$f(A) + f(B) \geq f(A \cup B) + f(A \cap B).$$

The set-function f has *curvature* c if for all $A \subseteq \mathcal{U}$ and $x \notin A$, $f_A(x) \geq (1 - c)f(x)$. Note that if f has curvature c and $c' \geq c$, then f also has curvature c' . Every normalized monotone function has curvature 1. A normalized function with curvature 0 is linear, that is $f_A(x) = f(x)$.

If \mathcal{I} is a collection of sets, then f has *restricted curvature* c with respect to \mathcal{I} if for all disjoint $A, B \in \mathcal{I}$, $f_A(B) \geq (1 - c)f(B)$. This is a weaker notion than curvature.

Matroids A matroid \mathcal{M} on a ground set \mathcal{U} is a collection of subsets of \mathcal{U} satisfying the following two properties: (1) If $A \in \mathcal{M}$ and $B \subseteq A$ then $B \in \mathcal{M}$; (2) If $A, B \in \mathcal{M}$ and $|A| > |B|$ then $B + x \in \mathcal{M}$ for some $x \in A \setminus B$.

The sets in \mathcal{M} are called *independent sets*. Maximal independent sets are known as *bases*. It turns out that all bases of the matroid have the same size, called the *rank* of the matroid.

One simple example is a *partition matroid*. The universe \mathcal{U} is partitioned into r parts $\mathcal{U}_1, \dots, \mathcal{U}_r$, and a set is independent if it contains at most one element from each part.

If A is an independent set, then the *slice matroid* \mathcal{M}/A , defined on $\mathcal{U} \setminus A$, is given by

$$\mathcal{M}/A = \{B \subseteq \mathcal{U} \setminus A : A \cup B \in \mathcal{M}\}.$$

Our proofs crucially depend on the following elementary result from [3].

Fact (Brualdi's lemma). *Suppose A, B are two bases in a matroid. There is a bijection $\pi: A \rightarrow B$ such that for all $a \in A$, $A - a + \pi(a)$ is a basis. Furthermore, π is the identity on $A \cap B$.*

Monotone submodular maximization An instance of monotone submodular maximization is given by a triple $(\mathcal{M}, \mathcal{U}, f)$, where \mathcal{M} is a matroid on \mathcal{U} , and f is a set-function on \mathcal{U} which is normalized, monotone and submodular.

The *optimum* of the instance is

$$f^* = \max_{O \in \mathcal{M}} f(O).$$

The maximum is always attained at some basis. A set $S \in \mathcal{M}$ is an α -*approximation* if $f(S) \geq \alpha f(O)$. Thus $0 \leq \alpha \leq 1$. (Some authors use α^{-1} instead of α .)

Maximum coverage Maximum coverage is the special case of monotone submodular maximization when f is a *coverage function*. A coverage function is given by the following data. There is a universe \mathbf{V} with non-negative weight function $w: \mathbf{V} \rightarrow \mathbb{R}_{\geq 0}$. The weight function is extended to subsets of \mathbf{V} linearly. Each element in \mathcal{U} is a subset \mathbf{V} , and the function f satisfies $f(A) = w(\bigcup A)$.

3 The auxiliary objective function

Throughout this section as well as §4,5, fix some instance $(\mathcal{M}, \mathcal{U}, f)$ of monotone submodular maximization, and let r be the rank of \mathcal{M} . In this section we define the auxiliary objective function g , assuming that f has curvature c , where $c \in (0, 1]$. Our construction depends on c , but c will not figure out in any of the notations we use to avoid clutter. Careful inspection of our proofs shows that it is enough to assume that f has restricted curvature c with respect to \mathcal{M} .

3.1 Definition

The definition is in stages. Let P be the probability distribution supported on $[0, 1]$ with density $ce^{cx}/(e^c - 1)$. Let μ be a probability distribution on $2^{\mathcal{U}}$ defined as follows: choose p randomly according to P , and then take each $x \in \mathcal{U}$ with probability p independently. For $A \subseteq \mathcal{U}$, let μ_A be the marginal distribution on 2^A . Finally, define $m_k^{(n)} = \mu_A(B)$, where $B \subseteq A$ are arbitrary sets of sizes $|A| = n$ and $|B| = k$.

As it turns out, μ will govern the definition of *marginals* of g . In order to define g itself, we have to define a different measure ν_A for each subset $A \subseteq \mathcal{U}$. This measure is defined by $\nu_A(\emptyset) = 0$ and for non-empty $B \subseteq A$,

$$\nu_A(B) = \mathbb{E}_{p \sim P} [p^{|B|-1} (1-p)^{|A|-|B|}].$$

Note that ν_A is *not* a probability measure. We define

$$g(A) = \sum_{B \subseteq A} f(B) \nu_A(B). \tag{1}$$

Lemma 1. For any $A \subseteq \mathcal{U}$ and $x \in \mathcal{U} \setminus A$,

$$g_A(x) = \mathbb{E}_{B \sim \mu_A} f_B(x). \tag{2}$$

Proof. Substituting (1),

$$\begin{aligned}
g_A(x) &= g(A+x) - g(A) \\
&= \sum_{B \subseteq A} \mathbb{E}_{p \sim P} \left[p^{|B|} (1-p)^{|A|-|B|} \right] f(B+x) \\
&\quad + \mathbb{E}_{p \sim P} \left[p^{|B|-1} (1-p)^{|A|+1-|B|} - p^{|B|-1} (1-p)^{|A|-|B|} \right] f(B) \\
&= \sum_{B \subseteq A} \mathbb{E}_{p \sim P} \left[p^{|B|} (1-p)^{|A|-|B|} \right] [f(B+x) - f(B)] \\
&= \sum_{B \subseteq A} \mu_A(B) f_B(x) = \mathbb{E}_{B \sim \mu_A} f_B(x). \quad \square
\end{aligned}$$

Note The definition of g might look mysterious. However, as we show in §B, any function g satisfying some natural properties can be written in this form for an appropriate distribution P .

3.2 Properties

The following result is immediate.

Lemma 2. *For $x \in \mathcal{U}$ we have $g(x) = f(x)$.*

Proof. Since $f(\emptyset) = g(\emptyset) = 0$, this follows directly from formula (2) with $A = \emptyset$. □

Next, we show that g shares all basic properties of f .

Lemma 3. *The function g is normalized, monotone, submodular and has curvature c .*

Proof. It is clear from the definition that g is normalized. Formula (2) immediately implies that g is monotone. Moreover, suppose that $A_1 \subseteq A_2$ and $x \notin A_2$. The same formula shows that

$$g_{A_2}(x) = \mathbb{E}_{B \sim \mu_{A_2}} f_B(x) \leq \mathbb{E}_{B \sim \mu_{A_2}} f_{B \cap A_1}(x) = \mathbb{E}_{B \sim \mu_{A_1}} f_B(x) = g_{A_1}(x).$$

Thus g is submodular. Finally, for $x \notin A$,

$$g_A(x) = \mathbb{E}_{B \sim \mu_A} f_B(x) \geq (1-c)f(x) = (1-c)g(x).$$

This shows that g has curvature c .

Curiously, for given $|A|$, g has somewhat lower curvature. Indeed, a more accurate estimate is

$$g_A(x) \geq [(1-c) + cm_0^{(|A|)}]g(x) = (1 - c(1 - m_0^{(|A|)}))g(x). \quad \square$$

Finally, we present bounds on the ratio between g and f .

Lemma 4. *For $S \subseteq \mathcal{U}$,*

$$f(S) \leq g(S) \leq \frac{ce^c}{e^c - 1} H_{|S|} f(S).$$

Proof. Formula (2) implies that for $x \notin A$,

$$g_A(x) = \mathbb{E}_{B \sim \mu_A} f_B(x) \geq f_A(x).$$

This easily implies that $g(S) \geq f(S)$. For the other direction, formula (1) implies that

$$g(S) \leq \sum_{T \subseteq S} \nu_S(T) f(T).$$

We proceed to estimate the sum:

$$\begin{aligned}
\sum_{T \subseteq S} \nu_S(T) &= \sum_{k=1}^{|S|} \binom{|S|}{k} \mathbb{E}_{p \sim P} [p^{k-1} (1-p)^{|S|-k}] \\
&= \frac{c}{e^c - 1} \sum_{k=1}^{|S|} \binom{|S|}{k} \int_0^1 p^{k-1} (1-p)^{|S|-k} e^{cp} dp \\
&\leq \frac{ce^c}{e^c - 1} \sum_{k=1}^{|S|} \binom{|S|}{k} \int_0^1 p^{k-1} (1-p)^{|S|-k} dp \\
&= \frac{ce^c}{e^c - 1} \sum_{k=1}^{|S|} \frac{\binom{|S|}{k}}{|S| \binom{|S|-1}{k-1}} \\
&= \frac{ce^c}{e^c - 1} \sum_{k=1}^{|S|} \frac{1}{k},
\end{aligned}$$

using the Beta integral. □

Lemma 5. For $S \subseteq \mathcal{U}$ and $x \in \mathcal{U} \setminus S$,

$$\frac{c}{e^c - 1} \cdot \frac{f(x)}{|S| + 1} \leq g_S(x) \leq f(x).$$

Moreover,

$$\frac{c}{e^c - 1} \cdot \frac{1}{|S| + 1} \leq \mu_S(\emptyset) \leq \frac{ce^c}{e^c - 1} \cdot \frac{1}{|S| + 1}.$$

Proof. Formula (2) shows that

$$g_S(x) = \mathbb{E}_{T \sim \mu_S} f_T(x) \leq f(x).$$

On the other hand,

$$g_S(x) \geq \mu_S(\emptyset) f(x).$$

We can estimate $\mu_S(\emptyset)$ directly from its formula, using the monotonicity of e^{cp} in p :

$$\begin{aligned}
\mu_S(\emptyset) &= \frac{c}{e^c - 1} \int_0^1 (1-p)^{|S|} e^{cp} dp \\
&= \frac{ce^{cp^*}}{e^c - 1} \int_0^1 (1-p)^{|S|} dp = \frac{ce^{cp^*}}{e^c - 1} \cdot \frac{1}{|S| + 1},
\end{aligned}$$

for some $p^* \in [0, 1]$. □

The factors $c/(e^c - 1)$ and $ce^c/(e^c - 1)$ are bounded, as the following simple result shows.

Lemma 6. The function $c/(e^c - 1)$ is decreasing, the function $ce^c/(e^c - 1)$ is increasing. Consequently, for $c \in [0, 1]$,

$$\begin{aligned}
\frac{1}{e - 1} &\leq \frac{c}{e^c - 1} \leq 1, \\
1 &\leq \frac{ce^c}{e^c - 1} \leq \frac{e}{e - 1}.
\end{aligned}$$

4 Evaluating g

Having defined g , we present two ways to evaluate g (approximately or exactly). In the set coverage case, which we cover first, g can be evaluated exactly, and (unsurprisingly) for $c = 1$, we get the same function as in [14]. In general, we can evaluate g by sampling, as we explain next.

4.1 Set coverage

When f is a coverage function, we can compute g explicitly in terms of the *histogram* of the input. Let us recall the setting: there is some ground set \mathbf{V} , and every element $\mathbf{a} \in \mathbf{V}$ has a non-negative weight $w(\mathbf{a})$. Each element $x \in \mathcal{U}$ is some subset of \mathbf{V} .

For a set $S \subseteq \mathcal{U}$, let $h(\mathbf{a}; S)$ be the number of times that \mathbf{a} appears in sets in S .

Lemma 7. For $S \subseteq \mathcal{U}$ we have

$$g(S) = \sum_{\mathbf{a} \in \bigcup S} \alpha_{h(\mathbf{a}; S)} w(\mathbf{a}),$$

where the sequence α_n is given by

$$\alpha_n = \mathbb{E}_{p \sim P} \left[\frac{1 - (1 - p)^n}{p} \right].$$

Proof. It is immediate from the definition of $g(S)$ that there are some constants $\alpha_k^{(m)}$ such that

$$g(S) = \sum_{\mathbf{a} \in \mathbf{V}} \alpha_{h(\mathbf{a}; S)}^{(|S|)} w(\mathbf{a}).$$

In order to see that $\alpha_k^{(m)}$ doesn't depend on m , fix S and let x be a new element corresponding to the empty set. Since $f_T(x) = 0$ for all T , formula (2) shows that $g_S(x) = 0$, that is, $g(S + x) = g(S)$. This implies that

$$\sum_{\mathbf{a} \in \mathbf{V}} \alpha_{h(\mathbf{a}; S)}^{(|S|+1)} w(\mathbf{a}) = \sum_{\mathbf{a} \in \mathbf{V}} \alpha_{h(\mathbf{a}; S)}^{(|S|)} w(\mathbf{a}).$$

In other words, $\alpha_k^{(|S|+1)} = \alpha_k^{(|S|)}$ for all $k \leq |S|$. This shows that the sequences $\alpha^{(|S|)}$ extend to a universal sequence α .

Finally, in order to calculate α_n , consider a set S consisting of n copies of the set $\{\mathbf{a}\}$, where $w(\mathbf{a}) = 1$. On the one hand, $g(S) = \alpha_n$, and on the other,

$$g(S) = \sum_{T \subseteq S} f(T) \nu_S(T) = \sum_{T \subseteq S} \nu_S(T).$$

Evaluate the sum using the formula for $\nu_S(T)$:

$$\begin{aligned} \alpha_n &= \sum_{T \subseteq S} \nu_S(T) = \sum_{k=1}^n \binom{n}{k} \mathbb{E}_{p \sim P} [p^{k-1} (1-p)^{n-k}] \\ &= \mathbb{E}_{p \sim P} \left[\frac{1 - (1-p)^n}{p} \right]. \quad \square \end{aligned}$$

In order to evaluate the sequence α_n , it will be useful to define the difference sequence $\beta_n = \alpha_{n+1} - \alpha_n$.

Lemma 8. The β sequence is given by

$$\beta_n = \mathbb{E}_{p \sim P} [(1-p)^n] = m_0^{(n)}.$$

It is also given by the recurrence

$$\beta_0 = 1, \quad \beta_n = c^{-1}n\beta_{n-1} - \frac{1}{e^c - 1}.$$

The α sequence is given by the recurrence

$$\alpha_0 = 0, \quad \alpha_1 = 1, \quad \alpha_{n+1} = (c^{-1}n + 1)\alpha_n - c^{-1}n\alpha_{n-1} - \frac{1}{e^c - 1}.$$

Proof. The formula for β follows from the identity

$$\frac{1 - (1-p)^{n+1}}{p} - \frac{1 - (1-p)^n}{p} = (1-p)^n.$$

The recurrence for β follows from Lemma 10, proved below in §4.2. The recurrence for α follows by substituting the definition of β in its recurrence formula. \square

When $c = 1$, these are the same recurrences (up to scaling) that appear in [14]. Finally, we deduce from Lemma 5 some bounds on β_n and α_n .

Lemma 9. For $n \geq 0$,

$$\begin{aligned} \frac{c}{e^c - 1} H_n \leq \alpha_n \leq \frac{ce^c}{e^c - 1} H_n, \\ \frac{c}{e^c - 1} \cdot \frac{1}{n+1} \leq \beta_n \leq \frac{ce^c}{e^c - 1} \cdot \frac{1}{n+1}. \end{aligned}$$

4.2 Evaluating μ and ν

In order to evaluate μ on subsets of \mathcal{U} , it is enough to calculate $m_k^{(n)}$ for all relevant n and k . The following lemma gives a recurrence facilitating this computation.

Lemma 10. For $n > 0$ and $0 \leq k \leq n$,

$$cm_k^{(n)} = (n-k)m_k^{(n-1)} - km_{k-1}^{(n-1)} + \begin{cases} -c/(e^c - 1) & \text{if } k = 0, \\ 0 & \text{if } 0 < k < n, \\ ce^c/(e^c - 1) & \text{if } k = n. \end{cases}$$

The base case is $m_0^{(0)} = 1$.

Proof. The proof is based on a simple integration by parts:

$$\begin{aligned} cm_k^{(n)} &= \frac{c}{e^c - 1} \int_0^1 p^k (1-p)^{n-k} ce^{cp} dp \\ &= \frac{c}{e^c - 1} [p^k (1-p)^{n-k} e^{cp}]_0^1 \\ &\quad - \frac{c}{e^c - 1} \int_0^1 [kp^{k-1} (1-p)^{n-k} - (n-k)p^k (1-p)^{n-k-1}] e^{cp} dp \\ &= \frac{\llbracket k = n \rrbracket ce^c - \llbracket k = 0 \rrbracket c}{e^c - 1} + (n-k)m_k^{(n-1)} - km_{k-1}^{(n-1)}. \end{aligned} \quad \square$$

The measures ν and μ are related by $\nu_S(T) = m_{|T|-1}^{(|S|-1)}$. However, ν is not a probability measure. In order to determine the measure of the entire space, consider the function $f(S) = \llbracket S \neq \emptyset \rrbracket$. Lemma 7 shows that the required measure is $\alpha_{|S|}$ (indeed, this is how we evaluated α_n in the first place). It is therefore natural to define the probability measure $\nu_S^* = \nu_S / \alpha_{|S|}$. This measure satisfies

$$g(T) = \alpha_{|S|} \mathbb{E}_{T \sim \nu_S^*} f(T). \quad (3)$$

4.3 Sampling g

We show how to estimate $g(S)$ by sampling. The main tool is Hoeffding's bound, which we presently recall.

Fact (Hoeffding's bound). *Let X_1, \dots, X_N be i.i.d. non-negative random variables bounded by B , and let \bar{X} be their average. Suppose that $\mathbb{E} \bar{X} \geq \rho B$. For any $\epsilon > 0$,*

$$\Pr[(1 - \epsilon)\mathbb{E} \bar{X} \leq \bar{X} \leq (1 + \epsilon)\mathbb{E} \bar{X}] \geq 1 - 2 \exp\left(-\frac{2(\epsilon\rho)^2}{N}\right). \quad (4)$$

This has the following implication.

Lemma 11. *There is some universal constant C_1 such that the following holds.*

Let S be a set of size n . Choose $M, \epsilon > 0$. Let $\tilde{g}(S)$ be an estimate of $g(S)$ obtained by taking N samples of $\alpha_n f(T)$, where $T \sim \nu_S^$ and*

$$N = C_1 \epsilon^{-2} (\log n)^2 \log M.$$

Then

$$\Pr[(1 - \epsilon)g(S) \leq \tilde{g}(S) \leq (1 + \epsilon)g(S)] \geq 1 - \frac{1}{M}.$$

Proof. Each sample $\alpha_n f(T)$ is bounded by $\alpha_n f(S)$. On the other hand, Lemma 4 shows that $g(S) \geq f(S)$. This shows that we can take $\rho = 1/\alpha_n$ in (4). Finally, Lemma 9 shows that $\alpha_n = O(\log n)$. \square

5 Main algorithm

We describe the algorithm from the point of view of general monotone submodular functions. In the maximum coverage case, we can replace sampling with direct evaluation. Otherwise, we will use the notation $\tilde{g}^{[N]}(S)$ to mean the average of N samples of $\alpha_{|S|} f(T)$, where $T \sim \nu_S^*$. The value of N will be a parameter of the algorithm, which we set later. Another parameter of the algorithm is $\epsilon \in (0, 1/5]$.

Recall r is the rank of the matroid. The algorithm has two phases: greedy phase and local search phase. The greedy phases successively chooses sets S_1, \dots, S_r in such a way that $S = \{S_1, \dots, S_r\}$ is a basis. We choose S_k according to the formula

$$S_k = \operatorname{argmax}_{x: \{S_1, \dots, S_{k-1}, x\} \in \mathcal{M}} \tilde{g}^{[N]}(\{S_1, \dots, S_{k-1}, x\}).$$

In the local search phase, we continually try to improve the set S , until it is impossible to improve it by more than $1 + 3\epsilon$. At each step, we attempt to find $i \in [r]$ and $x \in \mathcal{U}$ such that $S - S_i + x$ is a basis and

$$\tilde{g}^{[N]}(S - S_i + x) > (1 + 3\epsilon)\tilde{g}^{[N]}(S).$$

If such i, x are found, then we replace S_i with x , and go to the next step. Otherwise, we stop and output S .

We will set N so that Lemma 11 applies, with the same ϵ and some M which will be so large that with high probability, the bad event alluded to in the lemma never happens. For the rest of the analysis (until §5.4), we will make this assumption, and later on we will choose M (and so N) to make the assumption valid with high probability. We call this the *sampling assumption*.

5.1 Greedy phase

We evaluate the greedy phase from the perspective of maximizing g . This is needed so that we can bound the number of local search iterations.

Lemma 12. *The set S produced by the greedy phase satisfies*

$$(2 + 3r\epsilon)g(S) \geq \max_{O \in \mathcal{M}} g(O).$$

Proof. Let O be a set at which the maximum is attained. According to Brualdi's lemma, we can write $O = \{O_1, \dots, O_r\}$ so that $\{S_1, \dots, S_{k-1}, O_k\}$ is independent for all $k \in [r]$. The sampling assumption implies that

$$(1 + 3\epsilon)g(\{S_1, \dots, S_k\}) \geq g(\{S_1, \dots, S_{k-1}, O_k\}).$$

(We used $(1 + \epsilon)/(1 - \epsilon) \leq 1 + 3\epsilon$ for $\epsilon \leq 1/5$.) In particular,

$$3\epsilon g(S) + g_{\{S_1, \dots, S_{k-1}\}}(S_k) \geq g_{\{S_1, \dots, S_{k-1}\}}(O_k).$$

Summing this over all k , we obtain

$$(1 + 3r\epsilon)g(S) \geq \sum_{k=1}^r g_{\{S_1, \dots, S_{k-1}\}}(O_k) \geq \sum_{k=1}^r g_S(O_k) \geq g_S(O) \geq g(O) - g(S). \quad \square$$

5.2 Local optimality

The sampling assumption, along with the bound $\epsilon \leq 1/5$, allows us to relate the working of local search to actual (rather than sampled) values of g .

Lemma 13. *If $\tilde{g}^{[N]}(A) > (1 + 3\epsilon)\tilde{g}^{[N]}(B)$ then*

$$g(A) > (1 + \epsilon/3)g(B).$$

If $\tilde{g}^{[N]}(A) \leq (1 + 3\epsilon)\tilde{g}^{[N]}(B)$ then

$$g(A) \leq (1 + 7\epsilon)g(B).$$

Proof. If $\tilde{g}^{[N]}(A) > (1 + 3\epsilon)\tilde{g}^{[N]}(B)$ then

$$(1 + \epsilon)g(A) > (1 + 3\epsilon)(1 - \epsilon)g(B).$$

If $\tilde{g}^{[N]}(A) \leq (1 + 3\epsilon)\tilde{g}^{[N]}(B)$ then

$$(1 - \epsilon)g(A) \leq (1 + 3\epsilon)(1 + \epsilon)g(B).$$

The lemma follows from simple arithmetic. □

At the end of the algorithm, we reach an approximate local optimum S . Let O be a basis maximizing $f(O)$ (this is an optimum for f rather than g). According to Brualdi's lemma, we can write $O = \{O_1, \dots, O_r\}$ so that $S - S_i + O_i$ is a base for all $i \in [r]$.

Lemma 14. *The sets S, O satisfy*

$$7r\epsilon g(S) + \sum_{i=1}^r [g(S) - g(S - S_i + O_i)] \geq 0.$$

Proof. The termination condition of the algorithm guarantees that for all $i \in [r]$,

$$(1 + 7\epsilon)g(S) \geq g(S - S_i + O_i).$$

Summing this over all $i \in [r]$, the lemma follows. □

5.3 Locality ratio

In this section, S and O keep their meaning from §5.2. Furthermore, we assume (without loss of generality) that if $O_i \in S$ for some $i \in [r]$, then in fact $O_i = S_i$. The goal of this section is to prove the following lemma, which is the crux of the entire argument. The proof relies on three technical lemmas, proved below.

Lemma 15. *The sets S, O satisfy*

$$\frac{ce^c}{e^c - 1} f(S) \geq f(O) + \sum_{i=1}^r [g(S) - g(S - S_i + O_i)].$$

Proof. We have

$$\begin{aligned} \sum_{i=1}^r g_{S-S_i}(S_i) &\geq \sum_{i=1}^r [g(S) - g(S - S_i + O_i)] + \mathbb{E}_{T \sim \mu_S} \sum_{i=1}^r f_{T-O_i}(O_i) \\ &\geq \sum_{i=1}^r [g(S) - g(S - S_i + O_i)] + f(O) - c \mathbb{E}_{T \sim \mu_S} f(T), \end{aligned}$$

where the first inequality follows from Lemma 16, and the second inequality follows from Lemma 17. Since $f(\emptyset) = 0$, Lemma 18 implies that

$$\frac{ce^c}{e^c - 1} f(S) = \sum_{i=1}^r g_{S-S_i}(S_i) + c \mathbb{E}_{T \sim \mu_S} f(T) \geq f(O) + \sum_{i=1}^r [g(S) - g(S - S_i + O_i)]. \quad \square$$

We proceed to prove the technical lemmas.

Lemma 16. *For all $i \in [r]$,*

$$g_{S-S_i}(S_i) \geq g(S) - g(S - S_i + O_i) + \mathbb{E}_{T \sim \mu_S} f_{T-O_i}(O_i).$$

Proof. We consider two cases: $O_i \notin S$ and $O_i = S_i$. If $O_i \notin S$ then the submodularity of g implies

$$\begin{aligned} g_{S-S_i}(S_i) &\geq g_{S-S_i+O_i}(S_i) \\ &= g(S + O_i) - g(S - S_i + O_i) \\ &= g_S(O_i) + g(S) - g(S - S_i + O_i) \\ &= g(S) - g(S - S_i + O_i) + \mathbb{E}_{T \sim \mu_S} f_T(O_i). \end{aligned}$$

Since $S_i \notin S$, $T - O_i = T$ for all $T \subseteq S$.

When $O_i = S_i$, the desired inequality is in fact tight:

$$g_{S-S_i}(S_i) = \mathbb{E}_{T \sim \mu_{S-S_i}} f_T(S_i) = \mathbb{E}_{T \sim \mu_S} f_{T-S_i}(S_i) = \mathbb{E}_{T \sim \mu_S} f_{T-O_i}(O_i).$$

The calculation relies on the fact that μ_{S-S_i} is the marginal of μ_S on 2^{S-S_i} . □

Lemma 17. *For any $T \subseteq S$,*

$$\sum_{i=1}^r f_{T-O_i}(O_i) \geq f(O) - cf(T).$$

Proof. Let $X = T \cap O$, $T' = T \setminus X$ and $O' = O \setminus X$. Furthermore, let $I(X) \subseteq [r]$ be the set of indices such that $O_i \in X$. We separate the sum on the left-hand side into two parts. The first part is

$$\sum_{i \notin I(X)} f_{T-O_i}(O_i) = \sum_{i \notin I(X)} f_T(O_i) \geq f_T(O').$$

Using $T \cup O' = O \cup T'$ and $O \cap T' = \emptyset$, we get

$$\begin{aligned} f_T(O') &= f(T \cup O') - f(T) \\ &= f(O \cup T') - f(T) \\ &\geq f(O) + (1-c)f(T') - f(T). \end{aligned}$$

The second part is

$$\sum_{i \in I(X)} f_{T-O_i}(O_i) \geq \sum_{i \in I(X)} (1-c)f(O_i) \geq (1-c)f(X).$$

Putting both parts together, we deduce

$$\begin{aligned} \sum_{i=1}^r f_{T-O_i}(O_i) &\geq f(O) + (1-c)[f(T') + f(X)] - f(T) \\ &\geq f(O) + (1-c)f(T) - f(T) \\ &= f(O) - cf(T). \end{aligned} \quad \square$$

The following lemma is the only place in which we use the specific definition of μ_S , in the form of Lemma 10.

Lemma 18. *We have*

$$\sum_{i=1}^r g_{S-S_i}(S_i) + c \mathbb{E}_{T \sim \mu_S} f(T) = \frac{ce^c}{e^c-1} f(S) - \frac{c}{e^c-1} f(\emptyset). \quad (5)$$

Proof. Recall

$$g_{S-S_i}(S_i) = \mathbb{E}_{U \sim \mu_{S-S_i}} [f(U + S_i) - f(U)].$$

Each subset $T \subseteq S$ appears on the right-hand side of this expression: if $S_i \in T$ then it appears with positive coefficient $\mu_{S-S_i}(T - S_i)$, and if $S_i \notin T$ then it appears with negative coefficient $\mu_{S-S_i}(T)$. Therefore the coefficient of $f(T)$ in the left-hand side of (5) is

$$c\mu_S(T) + \sum_{S_i \in T} \mu_{S-S_i}(T - S_i) - \sum_{S_i \notin T} \mu_{S-S_i}(T) = cm_{|T|}^{(r)} + |T|m_{|T|-1}^{(r-1)} - (r - |T|)m_{|T|}^{(r-1)}.$$

According to Lemma 10, the right-hand side vanishes unless $T = \emptyset$ or $T = S$. Substituting the values obtained in these two cases, the lemma follows. \square

5.4 Parameters and analysis

Combining Lemma 14 and Lemma 15, we deduce the approximation ratio of the algorithm.

Lemma 19. *Let S denote the output of the algorithm, and let O be a basis maximizing $f(O)$. Then*

$$\frac{f(S)}{f(O)} \geq \frac{1 - e^{-c}}{c} - O(r \log r\epsilon).$$

Proof. Combining Lemma 14 and Lemma 15, we deduce

$$\frac{c}{1 - e^{-c}} f(S) \geq f(O) - 7r\epsilon g(O) \geq (1 - O(r \log r\epsilon))f(O),$$

where the second inequality follows from Lemma 4. \square

The final piece of the puzzle is the number of iterations needed for local search to converge.

Lemma 20. *The number of steps T in the local search phase is bounded by*

$$T \leq \epsilon^{-1} + 9r.$$

Proof. Suppose there are T steps in all, and the versions of S encountered during the algorithm are S^0, \dots, S^T , where S^0 is the result of the greedy phase. According to Lemma 12,

$$(2 + 3r\epsilon)g(S^0) \geq g(S^T).$$

On the other hand, Lemma 13 implies that for $t \in [T]$,

$$g(S^{t+1}) \geq (1 + \epsilon/3)g(S^t).$$

Therefore

$$2 + 3r\epsilon \geq (1 + \epsilon/3)^T \geq 1 + (T/3)\epsilon. \quad \square$$

We can finally put everything together, discharging the sampling assumption.

Theorem 1. *There are some universal constants C_2, C_3 such that the following holds.*

Given $\eta \in (0, 1]$, choose ϵ and N as follows:

$$\epsilon = \frac{\eta}{C_2 r \log r}, \quad N = C_3 \eta^{-2} r^2 (\log r)^4 \log(\eta^{-1} |\mathcal{U}|).$$

Then with probability at least $1 - 1/|\mathcal{U}|$, the algorithm has an approximation ratio of $(1 - e^{-c})/c - \eta$, and makes $\tilde{O}(\eta^{-3} r^4 |\mathcal{U}|)$ oracle calls to f .

Proof. Choose $C_2 \leq 1/5$ so that Lemma 19 gives the bound $(1 - e^{-c})/c - \eta$. Each step of the greedy phase requires at most $|\mathcal{U}|$ samplings of g , and each step of the local search phase requires at most $r|\mathcal{U}|$ samplings. Therefore Lemma 20 implies that the total number of samplings G is at most

$$G = O(\epsilon^{-1} r |\mathcal{U}|) = O(\eta^{-1} r^2 \log r |\mathcal{U}|).$$

Take $M = |\mathcal{U}|G$ in Lemma 11 to choose C_3 so that the sampling assumption fails with probability at most $1/|\mathcal{U}|$. □

6 Extensions

The algorithm presented in §5 produces a $(1 - e^{-c})/c - \eta$ approximation for any $\eta > 0$, and it requires knowledge of c . In this section we show how to produce a clean $(1 - e^{-c})/c$ approximation, and how to dispense with the knowledge of c . Unfortunately, we are unable to combine both improvements for technical reasons.

It will be useful to define the function

$$\rho(c) = \frac{1 - e^{-c}}{c},$$

which gives the optimal approximation ratio.

6.1 Clean approximation

In this section, we assume c is known, and our goal is to obtain a $\rho(c)$ approximation algorithm. The idea, taken from Khuller et al. [21] and Calinescu et al. [5], is to “guess” a “high-weight” set from the optimal solution, and run the main algorithm on the marginal instance.

Fact. Let $I = (\mathcal{M}, \mathcal{U}, f)$ be an instance of monotone submodular maximization over a matroid with optimal solution attained at O . Given $A \in \mathcal{M}$, the derived instance is given by

$$I/A = (\mathcal{M}/A, \mathcal{U} \setminus A, f_A).$$

This is another instance of monotone submodular maximization.

Lemma 21. Suppose $A \subseteq O$ and $B \subseteq \mathcal{U} \setminus A$ satisfy $f(A) \geq (1 - \theta_A)f(O)$ and $f_A(B) \geq (1 - \theta_B)f_A(O \setminus A)$. Then

$$f(A \cup B) \geq (1 - \theta_A \theta_B)f(O).$$

Proof. We have

$$\begin{aligned} f(A \cup B) &= f_A(B) + f(A) \geq (1 - \theta_B)f_A(O \setminus A) + f(A) \\ &= (1 - \theta_B)f(O) + \theta_B f(A) \\ &\geq (1 - \theta_B)f(O) + \theta_B(1 - \theta_A)f(O) = (1 - \theta_A \theta_B)f(O). \quad \square \end{aligned}$$

Putting things together, we get the improved algorithm. We loop over all elements $x \in \mathcal{U}$. For each element x , we apply the main algorithm with $\eta = (1 - \rho(c))/r$ to get a set S_x . Finally, we output the set $S_x + x$ maximizing $f(S_x + x)$.

Theorem 2. With probability at least $1 - 1/|\mathcal{U}|$, the improved algorithm has an approximation ratio of $\rho(c)$, and makes $\tilde{O}_c(r^7|\mathcal{U}|^2)$ oracle calls to g .

Proof. Let O be an optimal solution. Since

$$\sum_{i=1}^r f(O_i) \geq f(O),$$

there is some set $x = O_i$ such that $f(O_i) \geq f(O)/r$. Take $A = \{x\}$ and $B = S_x$ in Lemma 21. Substituting $\theta_A = 1 - 1/r$ and $\theta_B = 1 - \rho(c) + \eta$, we deduce that the resulting approximation ratio is

$$\begin{aligned} 1 - \left(1 - \frac{1}{r}\right) \left(1 - \rho(c) + \frac{1 - \rho(c)}{r}\right) &= 1 - \left(1 - \frac{1}{r}\right) \left(1 + \frac{1}{r}\right) (1 - \rho(c)) \\ &\geq 1 - (1 - \rho(c)) = \rho(c). \quad \square \end{aligned}$$

6.2 Unknown curvature

In this section, we remove the assumption that c is known, but retain a parameter η . The key observation is that if a function has curvature c then it also has curvature c' whenever $c' \geq c$. This, combined with the continuity of ρ , allows us to “guess” an approximate value of c .

Given η , consider the following oblivious algorithm. Define the set C of curvature approximations by

$$C = \{k\eta : 1 \leq k \leq \lfloor \eta^{-1} \rfloor\} \cup \{1\}.$$

For each $c \in C$, we run the main algorithm with that setting of c to get a set S_c . Finally, we output the set S_c maximizing $f(S_c)$.

Theorem 3. Suppose f has curvature d . With probability at least $1 - 1/|\mathcal{U}|$, the oblivious algorithm has an approximation ratio of $\rho(d) - 2\eta$, and makes $\tilde{O}(\eta^{-4}r^4|\mathcal{U}|)$ oracle calls to g .

Proof. From the definition of C it is clear that there is some $c \in C$ satisfying $d \leq c \leq d + \eta$. Since f has curvature c , the set S_c is a $\rho(c) - \eta$ approximate optimum. Elementary calculus shows that on $[0, 1]$, $\rho' \geq -1/2$, and so $\rho(d) \leq \rho(c) + \eta$. \square

References

- [1] Alexander A. Ageev and Maxim I. Sviridenko. Pipage rounding: A new method of constructing algorithms with proven performance guarantee. *J. of Combinatorial Optimization*, 8(3):307–328, September 2004.
- [2] Paola Alimonti. New local search approximation techniques for maximum generalized satisfiability problems. In *CIAC: Proc. of the 2nd Italian Conf. on Algorithms and Complexity*, pages 40–53, 1994.
- [3] Richard A. Brualdi. Comments on bases in dependence structures. *Bull. of the Austral. Math. Soc.*, 1(02):161–167, 1969.
- [4] Niv Buchbinder, Moran Feldman, Joseph (Seffi) Naor, and Roy Schwartz. A tight linear time $(1/2)$ -approximation for unconstrained submodular maximization. In *FOCS*, 2012.
- [5] Gruia Calinescu, Chandra Chekuri, Martin Pál, and Jan Vondrák. Maximizing a submodular set function subject to a matroid constraint (Extended abstract). In *IPCO*, pages 182–196, 2007.
- [6] Gruia Calinescu, Chandra Chekuri, Martin Pál, and Jan Vondrák. Maximizing a monotone submodular function subject to a matroid constraint. *SIAM J. Comput.*, 40(6):1740–1766, 2011.
- [7] Chandra Chekuri, Jan Vondrak, and Rico Zenklusen. Dependent randomized rounding via exchange properties of combinatorial structures. *Proceedings of the 2010 IEEE 51st Annual Symposium on Foundations of Computer Science*, pages 575–584, 2010.
- [8] Jack Edmonds. Matroids and the greedy algorithm. *Math. Programming*, 1(1):127–136, 1971.
- [9] Uriel Feige. A threshold of $\ln n$ for approximating set cover. *J. ACM*, 45:634–652, July 1998.
- [10] Uriel Feige, Vahab S. Mirrokni, and Jan Vondrak. Maximizing non-monotone submodular functions. In *FOCS*, pages 461–471, 2007.
- [11] Moran Feldman, Joseph Naor, and Roy Schwartz. A unified continuous greedy algorithm for submodular maximization. In *FOCS*, pages 570–579, 2011.
- [12] Moran Feldman, Joseph (Seffi) Naor, and Roy Schwartz. Nonmonotone submodular maximization via a structural continuous greedy algorithm. In *ICALP*, pages 342–353, 2011.
- [13] Moran Feldman, Joseph (Seffi) Naor, Roy Schwartz, and Justin Ward. Improved approximations for k -exchange systems. In *ESA*, pages 784–798, 2011.
- [14] Yuval Filmus and Justin Ward. The power of local search: Maximum coverage over a matroid. In *STACS*, pages 601–612, 2012.
- [15] Yuval Filmus and Justin Ward. A tight combinatorial algorithm for submodular maximization subject to a matroid constraint. In *FOCS*, 2012.
- [16] Yuval Filmus and Justin Ward. A tight combinatorial algorithm for submodular maximization subject to a matroid constraint. *Preprint*, 2012. arXiv:1204.4526.
- [17] Marshall L. Fisher, George L. Nemhauser, and Leonard A. Wolsey. An analysis of approximations for maximizing submodular set functions—II. In *Polyhedral Combinatorics*, pages 73–87. Springer Berlin Heidelberg, 1978.
- [18] Shayan Oveis Gharan and Jan Vondrák. Submodular maximization by simulated annealing. In *SODA*, pages 1098–1116, 2011.
- [19] Pranava R. Goundan and Andreas S. Schulz. Revisiting the greedy approach to submodular set function maximization. (manuscript), 2007.

- [20] Sanjeev Khanna, Rajeev Motwani, Madhu Sudan, and Umesh Vazirani. On syntactic versus computational views of approximability. *SIAM J. Comput.*, 28(1):164–191, 1999.
- [21] Samir Khuller, Anna Moss, and Joseph (Seffi) Naor. The budgeted maximum coverage problem. *Inf. Process. Lett.*, 70:39–45, April 1999.
- [22] Jon Lee, Vahab S. Mirrokni, Viswanath Nagarajan, and Maxim Sviridenko. Non-monotone submodular maximization under matroid and knapsack constraints. In *STOC*, pages 323–332, 2009.
- [23] Jon Lee, Maxim Sviridenko, and Jan Vondrák. Submodular maximization over multiple matroids via generalized exchange properties. *Math. of Oper. Res.*, 35(4):795–806, November 2010.
- [24] George L. Nemhauser and Leonard A. Wolsey. Best algorithms for approximating the maximum of a submodular set function. *Math. of Oper. Res.*, 3(3):177–188, 1978.
- [25] George L. Nemhauser, Leonard A. Wolsey, and Marshall L. Fisher. An analysis of approximations for maximizing submodular set functions—I. *Math. Programming*, 14(1):265–294, 1978.
- [26] Henri Padé. Sur la représentation approchée d’une fonction par des fractions rationnelles. *Annales scientifiques de l’École Normale Supérieure, Ser. 3*, 9:3–93 (supplement), 1892.
- [27] Richard Rado. Note on independence functions. *Proc. of the London Math. Soc.*, s3-7(1):300–320, 1957.
- [28] Jan Vondrák. Optimal approximation for the submodular welfare problem in the value oracle model. In *STOC*, pages 67–74, 2008.
- [29] Jan Vondrák. Submodularity and curvature: the optimal algorithm. In S. Iwata, editor, *RIMS Kokyuroku Bessatsu*, volume B23, Kyoto, 2010.
- [30] Justin Ward. A $(k+3)/2$ -approximation algorithm for monotone submodular k -set packing and general k -exchange systems. In *STACS*, pages 42–53, 2012.
- [31] Justin Ward. *Oblivious and Non-Oblivious Local Search for Combinatorial Optimization*. PhD thesis, University of Toronto, 2012.

A Tighter algorithms

The locality ratio of our algorithm for a function of curvature c is $(1 - e^{-c})/c$. Feige’s result [9] shows that for $c = 1$, this ratio is optimal, in the sense that for general r , the approximation ratio of any polynomial time algorithm is at most $(1 - e^{-c})/c + o(1)$, unless $P = NP$. For general c , we have Vondrák’s result [29] showing that $(1 - e^{-c})/c$ can’t be beaten in the value oracle model without exponentially many value queries. Conversely, as we show in §6.1 (following earlier work [21, 5]), it is always possible to “boost” the approximation ratio by a C/r amount for any constant C .

In this section we discuss the best achievable locality ratio that our method is able to produce. In the case of maximum coverage, we are able to find the best achievable ratio, and moreover this leads to better algorithms for restricted instances of maximum coverage, where every element appears at most R times. In the general case, we can improve on $(1 - e^{-c})/c$, but the locality ratio that we get is not optimal.

A.1 Maximum coverage

In this section we only consider the case $c = 1$.

As we show in §4.1, when f is a coverage function, we can define g as follows:

$$g(S) = \sum_{\mathbf{a} \in \mathbf{V}} \alpha_{h(\mathbf{a}, S)} w(\mathbf{a}),$$

where α_n is given by the recurrence

$$\alpha_0 = 0, \quad \alpha_1 = 1, \quad \alpha_{n+1} = (n+1)\alpha_n - n\alpha_{n-1} - \frac{1}{e-1}.$$

The sequence α is the only infinite sequence that gives the optimal approximation ratio $1 - 1/e$. However, for any given rank, there are other sequences that work. In particular, consider the following sequence:

$$\alpha_0^{(E)} = 0, \quad \alpha_1^{(E)} = 1, \quad \alpha_{n+1}^{(E)} = (n+1)\alpha_n^{(E)} - n\alpha_{n-1}^{(E)} - \frac{1}{E-1}.$$

Using the sequence $\alpha^{(E)}$ in place of α , we get a new function which we christen $g^{(E)}$. The corresponding sequence $\beta^{(E)}$ defined by $\beta_n^{(E)} = \alpha_{n+1}^{(E)} - \alpha_n^{(E)}$ satisfies the recurrence

$$\beta_0^{(E)} = 1, \quad \beta_n^{(E)} = n\beta_{n-1}^{(E)} - \frac{1}{E-1}.$$

The following lemma shows what values of E we should consider.

Lemma 22. *The sequence $\alpha_0^{(E)}, \dots, \alpha_{r-1}^{(E)}$ is non-decreasing if and only if*

$$E \geq \sum_{k=0}^{r-2} \frac{1}{k!}.$$

The sequence $\beta_0^{(E)}, \dots, \beta_r^{(E)}$ is non-increasing if and only if

$$E \leq \sum_{k=0}^{r-1} \frac{1}{k!} + \frac{1}{(r-1)(r-1)!}.$$

Proof. Lemma 2 in [14] gives the explicit formula

$$\beta_i^{(E)} = \frac{1}{E-1} i! \left(E - \sum_{k=0}^i \frac{1}{k!} \right).$$

Since $\alpha_{i+1}^{(E)} \geq \alpha_i^{(E)}$ if and only if $\beta_i^{(E)} \geq 0$, this immediately implies the first item.

The proof of Lemma 3 in [14] gives the formula

$$\beta_{i+1}^{(E)} - \beta_i^{(E)} = \frac{1}{E-1} i \cdot i! \left(E - \sum_{k=0}^i \frac{1}{k!} - \frac{1}{i \cdot i!} \right).$$

This immediately implies the second item.

Both formulas we have quoted can be easily proved by induction. □

As a consequence, we get the following theorem, which results from careful analysis of the proof of Theorem 5 in [14].

Theorem 4. *Suppose $R \geq 2$ is an integer and E satisfies*

$$\sum_{k=0}^{R-2} \frac{1}{k!} \leq E \leq \sum_{k=0}^{R-1} \frac{1}{k!} + \frac{1}{(R-1)(R-1)!}.$$

Let f be a coverage function corresponding to a set system in which every element is covered at most R times. Then Lemma 15 holds for $g^{(E)}$ with $E/(E-1)$ instead of $ce^c/(e^c-1)$.

If \mathcal{M} is a partition matroid, then the condition on the set system can be weakened to: every element is covered by sets belonging to at most R different partitions.

Proof. Say that an element of \mathbf{V} is of type (l, s, g) if it belongs to $l + s$ of the sets S_i , to $g + s$ of the sets O_i , and to s of the intersections $S_i \cap O_i$. By assumption, all elements have type (l, s, g) for $0 \leq l + s + g \leq R$. Denote by $F(l, s, g)$ the total weight of elements of type (l, s, g) . Using this notation, Lemma 15 reads

$$\frac{E}{E-1} \sum_{l+s \geq 1} F(l, s, g) \geq \sum_{g+s \geq 1} F(l, s, g) + \sum_{l+s+g \leq R} [l\beta_{l+s-1}^{(E)} - g\beta_{l+s}^{(E)}] F(l, s, g).$$

Comparing coefficients of $F(l, s, g) \geq 0$, we need to show that for integers $l, s, g \geq 0$ satisfying $1 \leq l+s+g \leq R$,

$$l\beta_{l+s-1}^{(E)} - g\beta_{l+s}^{(E)} \leq \begin{cases} 1/(E-1) & l+s, g+s \geq 1, \\ E/(E-1) & l \geq 1, g=s=0, \\ -1 & g \geq 1, l=s=0. \end{cases} \quad (6)$$

The condition on E implies that $\beta_0^{(E)}, \dots, \beta_{R-2}^{(E)} \geq 0$ and that $\beta_0^{(E)} \geq \dots \geq \beta_R^{(E)}$ is non-increasing.

When $l = s = 0$, $\beta_0^{(E)} = 1$ implies $-g\beta_0^{(E)} \leq -1$, and so (6) holds. When $g = s = 0$, the fact that $1 = \beta_1^{(E)} \geq \beta_l^{(E)}$ implies that

$$l\beta_{l-1}^{(E)} = \beta_l^{(E)} + \frac{1}{E-1} \leq \frac{E}{E-1},$$

and again (6) holds. When $g = 0$ and $s \geq 1$, the inequalities $\beta_l^{(E)} \geq \beta_{l+s-1}^{(E)}$ and $\beta_{l+1}^{(E)} \geq \beta_l^{(E)}$ (note $l+1 \leq l+s \leq R$) imply that

$$l\beta_{l+s-1}^{(E)} \leq l\beta_l^{(E)} = \beta_{l+1}^{(E)} + \frac{1}{E-1} - \beta_l^{(E)} \leq \frac{1}{E-1},$$

verifying (6). Finally, suppose $g \geq 1$ and $l+s \geq 1$. Since $\beta_{l+s-1}^{(E)} \geq 0$ due to $g \geq 1$, and furthermore $\beta_{l+s}^{(E)} \geq 0$ when $g \geq 2$,

$$l\beta_{l+s-1}^{(E)} - g\beta_{l+s}^{(E)} \leq l\beta_{l+s-1}^{(E)} - \beta_{l+s}^{(E)} = \frac{1}{E-1} - s\beta_{l+s-1}^{(E)} \leq \frac{1}{E-1},$$

completing the proof of (6). \square

Since $1 - 1/E$ is an increasing function of E , the best choice for E given R is

$$E^{(R)} = \sum_{k=0}^{R-1} \frac{1}{k!} + \frac{1}{(R-1)(R-1)!} = e + \frac{1}{(R+2)!} + O\left(\frac{1}{(R+3)!}\right).$$

This is the optimal choice of parameters, as the following theorem implies.

Theorem 5. *For every setting of the sequence α , the locality ratio of the corresponding auxiliary local search algorithm on matroids of rank r is at most $1 - 1/E^{(r)}$.*

Proof. Let r be given. Construct the following set system. The universe \mathbf{V} consists of elements $\mathfrak{r}(i, A)$ for every $A \subset [r]$ and $i \notin A$, and $\mathfrak{r}(i)$ for every $i \in [r]$. Their weights are given by

$$w(\mathfrak{r}(i, A)) = (r-1)(r-1-|A|)!, \quad w(\mathfrak{r}(i)) = 1.$$

The sets S_i, O_i , where $i \in [r]$, are defined as

$$\begin{aligned} S_i &= \{\mathfrak{r}(j, A) : A \subseteq [r] - j \text{ and } i \in A\} \cup \{\mathfrak{r}(j) : j \in [r]\}, \\ O_i &= \{\mathfrak{r}(i, A) : A \subseteq [r] - i\} + \mathfrak{r}(i). \end{aligned}$$

For T a set of subsets of \mathbf{V} , let $w_k(T)$ be the total weight of elements appearing in T exactly k times. Elementary calculations show that for all $i, k \in [r]$, $w_k(S) = w_k(S_{-i} + O_i)$. Therefore in the partition matroid with r partitions $\{S_i, O_i\}$, the set S is locally optimal for any choice of α . Further calculations show that $w(S)/w(O) = 1 - 1/E^{(R)}$. \square

Example When $r = 2$, the set system is particularly simple, consisting of six elements a, b, c, d, e, f of unit weight. The sets are

$$\begin{aligned} O_1 &= \{a, b, c\} & O_2 &= \{d, e, f\}, \\ S_1 &= \{c, d, e\} & S_2 &= \{b, c, d\}. \end{aligned}$$

Since $w(O) = 6$ while $w(S) = 4$, the locality gap is $2/3$. In both S and the hybrids $\{S_1, O_2\}, \{O_1, S_2\}$ there are two elements covered once and two elements covered twice.

A.2 General case

The analysis in the general case involves Padé approximants to e^x (though see comment below).

Fact. Let $a, b \geq 0$ be integers. The (a, b) Padé approximant to e^x is $R_{a,b} = P_{a,b}/Q_{a,b}$, where

$$P_{a,b}(x) = \sum_{t=0}^a \frac{x^t (a+b-t)! a!}{(a+b)! t! (a-t)!}, \quad Q_{a,b}(x) = \sum_{t=0}^b \frac{(-x)^t (a+b-t)! b!}{(a+b)! t! (b-t)!}.$$

The error in the approximation is given by

$$Q_{a,b}(x)e^x - P_{a,b}(x) = (-1)^b \frac{x^{a+b+1}}{(a+b)!} \int_0^1 (1-t)^a t^b e^{xt} dt. \quad (7)$$

Proof. See, for example, Padé's thesis [26, §66]. □

The crucial identity used in the proof of Lemma 15 is the recurrence of Lemma 10.

Lemma 23. Let $c > 0$ and $E \neq 1$. Suppose the coefficients $\check{m}_k^{(n)}$ satisfy the following recurrence, for all $n > 0$ and $k \in \{0, \dots, n\}$, and furthermore $\check{m}_0^{(0)} = 1$:

$$c\check{m}_k^{(n)} = (n-k)\check{m}_k^{(n-1)} - k\check{m}_{k-1}^{(n-1)} + \begin{cases} -c/(E-1) & \text{if } k=0, \\ 0 & \text{if } 0 < k < n, \\ cE/(E-1) & \text{if } k=n. \end{cases} \quad (8)$$

Then $\check{m}_k^{(n)}$ is given by the following formula:

$$\check{m}_k^{(n)} = \frac{c^{-n} n!}{E-1} (-1)^k [Q_{n-k,k}(c)E - P_{n-k,k}(c)]. \quad (9)$$

Conversely, if we define $\check{m}_k^{(n)}$ according to the formula, then the recurrence is satisfied.

Moreover, if \check{m} satisfies either of the equivalent conditions then for $n \geq 1$ and $k \in \{0, \dots, n-1\}$,

$$\check{m}_k^{(n-1)} = \check{m}_k^{(n)} + \check{m}_{k+1}^{(n)}. \quad (10)$$

Proof. Suppose $c > 0$ is rational, and let \check{m} satisfy the recurrence. It is easy to prove by induction that there are rational coefficients $A_k^{(n)}(c), B_k^{(n)}(c)$ such that

$$\check{m}_k^{(n)} = \frac{A_k^{(n)}(c)E - B_k^{(n)}(c)}{E-1}.$$

When $E = e^c$, the coefficients $m_k^{(n)}$ satisfy the recurrence, hence

$$\frac{A_k^{(n)}(c)e^c - B_k^{(n)}(c)}{e^c - 1} = m_k^{(n)} = \frac{c}{e^c - 1} \int_0^1 p^k (1-p)^{n-k} e^{cp} dp. \quad (11)$$

A glance at formula (7) shows that one solution to (11) is given by

$$A_k^{(n)}(c) = c^{-n} n! (-1)^k Q_{n-k,k}(c), \quad B_k^{(n)}(c) = c^{-n} n! (-1)^k P_{n-k,k}(c). \quad (12)$$

On the other hand, the uniqueness of the solution (over the rationals) follows from the irrationality of e^c (which itself follows from the transcendentality of e).

For general c , the values $A_k^{(n)}(c), B_k^{(n)}(c)$ still exist, though they are not necessarily rational. As functions of c , they are continuous in $(0, \infty)$, hence uniquely determined by their values on rational c . We conclude that (12) is valid for all $c > 0$.

Now suppose \check{m} is given by the explicit solution, for some $c > 0$ and $E \neq 1$. We can easily construct coefficients \check{M} satisfying the recurrence for the same c and E . These are given by the explicit solution. Hence $\check{M} = \check{m}$, and we conclude that \check{m} satisfies the recurrence.

For the remaining claim, let $c > 0$ be rational, $n \geq 1$ and $k \in \{0, \dots, n-1\}$. First, notice that (10) holds for m , since for $X \sim \mu_{[n]}$,

$$\begin{aligned} \Pr[X \cap [n-1] = [k-1]] &= \Pr[X = [k-1]] + \Pr[X = [k-1] + n] \\ &= \Pr[X = [k-1]] + \Pr[X = [k]]. \end{aligned}$$

As before, there are rational coefficients $G_k^{(n)}(c), H_k^{(n)}(c)$ satisfying

$$\check{m}_k^{(n-1)} - (\check{m}_k^{(n)} + \check{m}_{k+1}^{(n)}) = \frac{G_k^{(n)}(c)E - H_k^{(n)}(c)}{E-1}. \quad (13)$$

For $E = e^c$, the left-hand side of (13) vanishes. As before, this implies that $G_k^{(n)}(c) = H_k^{(n)}(c) = 0$, and so the left-hand side of (13) vanishes for arbitrary E and rational $c > 0$. The general claim follows by continuity. \square

Note The paper [16] proves a similar result for the γ sequences, which are related to the \check{m} sequences by

$$\binom{n}{k} \check{m}_k^{(n)} = \frac{c}{E-1} \frac{\gamma_{k+1}^{(n+1)}}{n+1}.$$

Careful inspection of the proof of Lemma 15 gives the following result.

Theorem 6. *Let $r > 0$ be an integer. Suppose \check{m} satisfies recurrence (8) for some E with $\check{m}_0^{(0)} = 1$, and furthermore $\check{m}^{(r)} \geq 0$. Then Lemma 15 holds for the corresponding function \check{g} , with $cE/(E-1)$ replacing $ce^c/(e^c-1)$.*

Moreover, when $c \in (0, 1]$, the condition $\check{m}^{(r)} \geq 0$ is satisfied if and only if

$$\frac{P_{r-L,L}(c)}{Q_{r-L,L}(c)} \leq E \leq \frac{P_{r-U,U}(c)}{Q_{r-U,U}(c)}, \quad L = 2 \left\lfloor \frac{r+1}{4} \right\rfloor, \quad U = 2 \left\lfloor \frac{r-1}{4} \right\rfloor + 1.$$

Proof. For the first part, note that identity (10) together with $\check{m}^{(r)} \geq 0$ implies that we can define probability distributions $\check{\mu}_A$ on 2^A for sets of size $|A| \leq r$, with the following property: if $B \subseteq A$ then the restriction of $\check{\mu}_A$ to B follows the same law as $\check{\mu}_B$. This is enough to imply that \check{g} is submodular as long as all sets involved have size at most $r+1$, as a simple analysis of Lemma 3 shows. The only other property needed in the proof of Lemma 15 is recurrence (8).

For the second part, note that $\check{m}^{(n)}$ satisfies the following recurrence, for $k \in \{0, \dots, n\}$:

$$(n-k)\check{m}_{k+1}^{(n)} = (2k-n+c)\check{m}_k^{(n)} + k\check{m}_{k-1}^{(n)} + \begin{cases} c/(E-1) & \text{if } k=0, \\ 0 & \text{if } 0 < k < n, \\ -cE/(E-1) & \text{if } k=n. \end{cases} \quad (14)$$

This follows from (8) by substituting $\check{m}_{k-1}^{(n-1)} = \check{m}_{k-1}^{(n)} + \check{m}_k^{(n)}$ and $\check{m}_k^{(n-1)} = \check{m}_k^{(n)} + \check{m}_{k+1}^{(n)}$.

Suppose $c \in (0, 1)$. If $E = R_{r-t,t}(c)$ then formula (9) shows that $\check{m}_t^{(r)} = 0$. Suppose furthermore that $\check{m}^{(r)} \geq 0$. Taking $n = r$ and $k = t + 1$ in (14), we deduce that $2(t + 1) - r + c \geq 0$. Taking $k = t - 1$, we deduce that $2(t - 1) - r + c \leq 0$. Therefore

$$\frac{r-c}{2} - 1 \leq t \leq \frac{r-c}{2} + 1.$$

Since $c \neq 1$, there are exactly two integers in that range. These must be L and U since

$$r - c - 2 \leq r - 2 \leq 4 \left\lfloor \frac{r+1}{4} \right\rfloor, 4 \left\lfloor \frac{r-1}{4} \right\rfloor + 2 \leq r + 1 \leq r - c + 2.$$

On the other hand, since $Q_{a,b}(c) > 0$, formula (9) shows that $\check{m}^{(r)} \geq 0$ is equivalent to

$$E \geq R_{r-2k,2k}(c), \quad 2k \in \{0, \dots, n\}, \quad E \leq R_{r-2k-1,2k+1}(c), \quad 2k+1 \in \{1, \dots, n\}. \quad (15)$$

Since $m^{(r)} > 0$, the setting $E = e^c$ satisfies (15) strictly, hence $R_{r-k,k}(c) < e^c$ for even k and $R_{r-k,k}(c) > e^c$ for odd k . Therefore the following two settings satisfy (15):

$$E_e = \max_{k \text{ even}} R_{r-k,k}(c), \quad E_o = \min_{k \text{ odd}} R_{r-k,k}(c).$$

The corresponding indices k_e, k_o necessarily satisfy $\{k_e, k_o\} = \{L, U\}$. This finishes the proof when $c \neq 1$. The case $c = 1$ follows by continuity; when r is odd, there are three values of k that give rise to a value of E satisfying (15). \square

An alternative proof of the second part follows the method of Lemma C.1 in [16].

Note Virtually the only property of the Padé approximants that we use is $Q_{a,b}(c) > 0$. This can be proved directly by induction on n using recurrence (8), the case $k = n$ requiring special treatment.

Unfortunately, while the optimal choice of $\check{m}^{(n)}$, calculated by solving an appropriate linear program (see [31]), seems to satisfy recurrence (8), the choice $E = R_{r-U,U}(c)$ implied by the theorem is sub-optimal for some values of r .

B The form of g

The definition of g might seem mysterious. In this section, we show that up to the specific distribution P used, the definition of g follows from several natural axioms. We then discuss the distribution P .

Theorem 7. *Let \mathcal{U} be a countable set. Given a set-function f defined on some finite subset of \mathcal{U} , define another set-function g on the same domain by*

$$g(A) = \sum_{B \subseteq A} \check{n}_{|B|}^{(|A|)} f(B),$$

where the coefficients \check{n} do not depend on f .

Suppose that g satisfies the following properties:

1. g is normalized, and if f is normalized then $g(x) = f(x)$.
2. If f is monotone then so is g .
3. If $f(B+x) = f(B)$ for all $B \subseteq A$ then $g(A+x) = g(A)$.

4. If $f(B+x+y)+f(B) = f(B+x)+f(B+y)$ for all $B \subseteq A$ then $g(A+x+y)+g(A) = g(A+x)+g(A+y)$.

Then there exists a probability distribution P on $[0, 1]$ such that

$$g_A(x) = \mathbb{E}_{p \sim P} \mathbb{E}_{B \sim \text{Ber}(A,p)} f_B(x),$$

where $\text{Ber}(A, p)$ is a random subset of A chosen by independently selecting each element $y \in A$ with probability p .

Proof. It will be profitable to consider the marginal of g instead of g itself:

$$g_A(x) = \sum_{B \subseteq A} \check{n}_{|B|+1}^{(|A|+1)} f(B+x) + (\check{n}_{|B|}^{(|A|+1)} - \check{n}_{|B|}^{(|A|)}) f(B).$$

Property 3 implies that $\check{n}_{|B|+1}^{(|A|+1)} = \check{n}_{|B|}^{(|A|)} - \check{n}_{|B|}^{(|A|+1)}$. Denote the common value of these two quantities by $\check{m}_{|B|}^{(|A|)}$, so that

$$g_A(x) = \sum_{B \subseteq A} \check{m}_{|B|}^{(|A|)} f_B(x).$$

Property 2 implies that $\check{m} \geq 0$.

Taking marginals again, we have

$$g_A(x) - g_{A+y}(x) = \sum_{B \subseteq A} (\check{m}_{|B|}^{(|A|)} - \check{m}_{|B|}^{(|A|+1)}) f_B(x) - \check{m}_{|B|+1}^{(|A|+1)} f_{B+y}(x).$$

Property 4 implies that

$$\check{m}_{|B|}^{(A)} = \check{m}_{|B|}^{(|A|+1)} + \check{m}_{|B|+1}^{(|A|+1)}. \quad (16)$$

Induction shows that for non-empty A ,

$$\sum_{B \subseteq A} \check{m}_{|B|}^{(|A|)} = \check{m}_0^{(0)}.$$

Property 1 implies that $\check{m}_0^{(0)} = 1$. Hence for each A we can define a probability distribution $\check{\mu}_A$ on 2^A such that

$$g_A(x) = \mathbb{E}_{B \sim \check{\mu}_A} f_B(x).$$

Identity (16) implies that if $A_1 \subseteq A_2$ and $B \sim \check{\mu}_{A_2}$ then $B \cap A_1 \sim \check{\mu}_{A_1}$. Therefore we can define a probability distribution $\check{\mu}$ on $2^{\mathcal{U}}$ with the property that $B \sim \check{\mu}$ implies $B \cap A \sim \check{\mu}_A$.

We can think of the probability distribution $\check{\mu}$ as a collection of indicator variables, one for each $x \in \mathcal{U}$. Since $\Pr_{B \sim \check{\mu}}[B \cap A = C]$ depends only on $|A|$ and $|C|$, these random variables are exchangeable. De Finetti's theorem now shows that there is a probability distribution P such that $\check{\mu} = \text{Ber}(\mathcal{U}, p)$, where $p \sim P$. \square

If the probability distribution $\check{\mu}$ mentioned in the theorem is known, then the distribution P can be extracted from it. Indeed, let $\mathcal{U} = \mathbb{N}$, and suppose $K \sim \check{\mu}$. Then

$$\Pr[P < p] = \lim_{t \rightarrow \infty} \Pr[|K \cap [t]| < pt].$$

In [15, 16] we calculate the distribution $\check{\mu}$ explicitly for given c , and notice that

$$\Pr[|K \cap [t]| = s] \propto e^{cs/t},$$

where the proportionality sign hides an approximation. This implies that the corresponding probability distribution is given by the density $ce^{cp}/(e^c - 1)$.

Note One crucial property of the distributions $\check{\mu}_A$ is the restriction property, which shows that if we restrict a random variable $\check{\mu}_A$ to a subset $C \subseteq A$, then we get a random variable distributed $\check{\mu}_C$. This property follows from (16), a property of the coefficients \check{m} . During the course of the proof, we actually prove a similar property for the coefficients \check{n} . Furthermore, $\check{m} \geq 0$ implies that $\check{n} \geq 0$, except perhaps for the coefficients $\check{n}_0^{(t)}$. Moreover,

$$\sum_{B \subseteq A} \check{n}_{|B|}^{(|A|)} = \check{n}_0^{(0)}.$$

On the other hand, for our function g it is the case that

$$\sum_{\emptyset \neq B \subseteq A} \check{n}_{|B|}^{(|A|)} = \alpha_{|A|},$$

which is unbounded.

There are two ways to define the quantity $\check{n}_0^{(t)}$ in such a way that we obtain a measure $\check{\nu}$ on $2^{\mathcal{U}}$. One way is to define $\check{n}_0^{(t)} = -\alpha_t$, in which case $\check{\nu}$ is a signed measure and $\check{\nu}(\{\emptyset\}) = -\infty$. The other way is to define $\check{n}_0^{(t)} = \infty$, in which case $\check{\nu}$ is an infinite measure and $\check{\nu}(\{\emptyset\})$. In both cases, the formula

$$g(A) = \sum_{B \subseteq A} \check{n}_{|B|}^{(|A|)} f(B) = \int_A f(B) d\check{\nu}(B)$$

holds only for normalized f .

C Combinatorial version of the continuous greedy algorithm

In this section we only consider the case $c = 1$.

Calinescu, Chekuri, Pál and Vondrák [6] developed the continuous greedy algorithm, which is an optimal $1 - 1/e$ approximation algorithm for the problem of monotone submodular maximization subject to a matroid constraint. At first glance, the continuous greedy algorithm is not combinatorial. Indeed, it is composed of two phases: the continuous greedy phase, and a rounding phase. The running time of the rounding phase has bad dependence on $|\mathcal{U}|$ due to the fact that the output of the continuous greedy phase can be an arbitrary vector in the base polytope of the matroid. However, a small variation of this algorithm, in which swap rounding [7] is performed at every step, alleviates this problem.

The continuous greedy algorithm makes use of the *continuous relaxation* of the given monotone submodular function f . For $v \in [0, 1]^{\mathcal{U}}$, let $R(v)$ be a random subset of \mathcal{U} which contains each element $x \in \mathcal{U}$ with probability $v(x)$. We define

$$f^*(v) = \mathbb{E}_{T \sim R(v)} f(T).$$

For a set T , its *indicator function* 1_T is defined by $1_T(x) = \mathbb{1}[x \in T]$. Note that $f^*(1_T) = f(T)$.

The combinatorial continuous greedy algorithm proceeds as follows. Aiming at an approximation ratio of $1 - 1/e - \eta$, let $\delta = \eta/3r$; for simplicity, suppose $1/\delta$ is an integer. For each $t \in [1/\delta]$, we will compute a base B_t of the matroid. The starting point (which is not a base) is $B_0 = \emptyset$, and we compute B_{t+1} from B_t as follows:

- For each $x \in \mathcal{U}$, estimate

$$\omega(x) = \mathbb{E}_{T \sim R(t\delta 1_{B_t})} [f_T(x)]$$

by averaging over $N = \frac{10}{\delta^2}(1 + \log |\mathcal{U}|)$ samples.

- Using the greedy algorithm, find a base C_t maximizing

$$\sum_{x \in C_t} \omega(x).$$

- If $t = 0$ then set $B_{t+1} := C_t$ and continue to the next iteration.
- Otherwise, let $B := B_t$ and $C := C_t$. While $B \neq C$:
 - Find $b \in B$ and $c \in C$ such that both $B' := B - b + c$ and $C' := C - c + b$ are bases.
 - Estimate $\tau_1 = f^*(t\delta 1_{B'} + \delta 1_C)$ and $\tau_2 = f^*(t\delta 1_B + \delta 1_{C'})$ by averaging over N samples.
 - If $\tau_1 > \tau_2$ then replace B with B' , else replace C with C' .
- Set $B_{t+1} := B$.

The output of the algorithm is the base $B_{1/\delta}$.

The inner loop in the algorithm is a deterministic version of the procedure MergeBases from [7]. The method of the proof of Lemma 3.5 in [6] extends to show that with high probability,

$$f^*((t+1)\delta B_{t+1}) \geq f^*(t\delta B_t + \delta C_t) - \epsilon,$$

where ϵ is some small error that we do not explicitly estimate here. The proof of Lemma 3.3 in the same paper then implies that with high probability, the algorithm gives a $1 - 1/e - \eta - o(1)$ approximation.

Since the inner loop runs at most r times, the number of times we need to evaluate f is at most

$$\delta^{-1} \cdot N \cdot (2|\mathcal{U}| + 2r) = \tilde{O}(\delta^{-3}|\mathcal{U}|) = \tilde{O}(\eta^{-3}r^3|\mathcal{U}|).$$

In comparison, Theorem 1 shows that our algorithm requires $\tilde{O}(\eta^{-3}r^4|\mathcal{U}|)$ evaluations of f .