

# On the Approximation of Submodular Functions

Nikhil R. Devanur\*    Shaddin Dughmi†    Roy Schwartz\*    Ankit Sharma‡  
Mohit Singh\*

April 19, 2013

## Abstract

Submodular functions are a fundamental object of study in combinatorial optimization, economics, machine learning, etc. and exhibit a rich combinatorial structure. Many subclasses of submodular functions have also been well studied and these subclasses widely vary in their complexity. Our motivation is to understand the relative complexity of these classes of functions. Towards this, we consider the question of how well can one class of submodular functions be approximated by another (simpler) class of submodular functions. Such approximations naturally allow algorithms designed for the simpler class to be applied to the bigger class of functions. We prove both upper and lower bounds on such approximations. Our main results are:

- General submodular functions<sup>1</sup> can be approximated by cut functions of directed graphs to a factor of  $n^2/4$ , which is tight.
- General symmetric submodular functions<sup>1</sup> can be approximated by cut functions of undirected graphs to a factor of  $n - 1$ , which is tight up to a constant.
- Budgeted additive functions can be approximated by coverage functions to a factor of  $e/(e - 1)$ , which is tight.

Here  $n$  is the size of the ground set on which the submodular function is defined. We also observe that prior works imply that monotone submodular functions can be approximated by coverage functions with a factor between  $O(\sqrt{n} \log n)$  and  $\Omega(n^{1/3}/\log^2 n)$ .

## 1 Introduction

Submodular optimization problems have been a rich area of research in recent years, motivated by the principle of diminishing marginal returns which is prevalent in real world applications. Such functions are ubiquitous in diverse disciplines, including economics, algorithmic game theory, machine learning, combinatorial optimization and combinatorics. While submodular function can be minimized efficiently, i.e., in polynomial time [16, 23, 18], many natural optimization problems over submodular functions are NP-hard, e.g., Max- $k$ -Coverage [20], Max-Cut and Max-DiCut [14], and Max-Facility-Location [9]. Consequently, many works, specifically in the setting of algorithmic game theory [7, 10, 11, 17], have explored *simpler* subclasses of submodular functions for which

---

\*Microsoft Research, Redmond {nikdev, roysch, mohits}@microsoft.com

†University of Southern California shaddin@usc.edu

‡Carnegie Mellon University ankits@cs.cmu.edu

<sup>1</sup>We additionally assume that the submodular function takes value 0 on the null set and the universe.

the given algorithmic problem can still be well-approximated. Such subclasses of submodular functions have included cut functions of graphs, coverage functions of set systems, budgeted additive functions, matroid rank functions, etc.

Our work is motivated by the question, how complex can a submodular function be? Since this is such a fundamental question, it has been asked in different forms previously. Goemans et al. [13] consider how many queries to a submodular function are sufficient to infer the value of the function, approximately, at every point in the domain. Balcan and Harvey [2] focus on the problem of learning submodular functions in a probabilistic model; are few random queries enough to infer the value at almost all points in the domain? Badanidiyuru et al. [1] ask whether an approximate *sketch* of a submodular function, or more generally a subadditive function, exists (i.e., can the function be represented in polynomial space)? Seshadri and Vondrák [24] consider the testability of submodular functions: how many queries does it take to check whether a function is *close* to being submodular?

We approach this question by noting that not all submodular functions are identically complex and some have been more amenable to optimization than others. Thus, one natural way to characterize the relative complexity of one class of submodular functions w.r.t another, is to ask how well can a function in the first class be approximated by a function in the second. Formally, we ask the following question. Given two classes of submodular functions  $\mathcal{F}$  and  $\mathcal{G}$  (typically  $\mathcal{G} \subset \mathcal{F}$ ), what is the smallest  $\theta$ , such that for every  $f \in \mathcal{F}$ , there exists a  $g \in \mathcal{G}$  such that  $f(S) \leq g(S) \leq \theta \cdot f(S)$  for each  $S \subseteq U$ ? Here class  $\mathcal{G}$  would represent the class of submodular functions which are easier to optimize for some problem and class  $\mathcal{F}$  would represent a bigger class which we want to optimize over. We also note that this concept of approximation is not special to submodular functions and can be asked for any two classes of functions. We focus on submodular functions due to their ubiquitous nature in optimization.

Intuitively, this notion of approximation resembles the long and rich line of work that deals with the algorithmic applications of geometric embeddings, in which the goal is to embed hard metric spaces into simpler ones. Some successful examples include embedding general metrics into normed spaces [5], dimension reduction in a Euclidean space [19] and the probabilistic embedding into ultrametrics [4, 12]. As in the metric case, a natural byproduct of the above approach is that if there exists an  $\alpha$ -approximation algorithm for any submodular function in  $\mathcal{G}$ , then there exists a  $(\theta \cdot \alpha)$ -approximation algorithm for all functions in  $\mathcal{F}$ . As an application of our approach, we show how to obtain an algorithm for the online submodular function maximization problem, for general monotone submodular functions [6]. Previously, results were known for only certain subclasses of submodular functions; see Appendix E for details.

## 1.1 Our Results and Techniques

We start by asking how well a general submodular function  $f : 2^U \rightarrow \mathbb{R}_+$  (with the additional property that  $f(\emptyset) = 0 = f(U)$ ) can be approximated by a function in the canonical simpler subfamily of non-symmetric submodular functions, cut function of a directed graph. We give matching upper and lower bounds for such an approximation (Theorem 1.1). Next, we ask the same question for *symmetric* submodular functions vis-a-vis its canonical simpler subfamily, cut functions of undirected graphs. In this case, we provide nearly matching upper and lower bounds (Theorem 1.2). We then move our attention to two subfamilies, budgeted additive functions and coverage functions, both of which, as already mentioned in the introduction, have received considerable interest in the algorithmic game theory setting. We show tight upper and lower bounds for approximating

Input Class	Output Class	Approximation Upper Bound	Approximation Lower Bound
General Submodular	Cut Functions (directed)	$\frac{n^2}{4}$	$\frac{n^2}{4}$
Symmetric Submodular	Cut functions (undirected)	$n - 1$	$\frac{n}{4}$
Budgeted Additive	Coverage	$\frac{e}{e-1}$	$\frac{e}{e-1}$
Monotone Submodular	Coverage/Budgeted Additive	$O(\sqrt{n} \log n)$ [13]	$\Omega(\frac{n^{1/3}}{\log^2 n})$ [2, 1]

Table 1: Our results are described in the first three rows. The results in the last row are either implicit in the references or follow as a corollary (Appendix G). When the output class is a cut function of a graph, we assume that the input function  $f$  satisfies  $f(\emptyset) = f(U) = 0$ , as every cut function must satisfy this constraint. Here,  $n$  denotes the size of the ground set.

budgeted additive functions with coverage functions (Theorem 1.3). These results are summarized in Table 1. While previous works [13, 2, 1] studied the complexity of submodular functions from different perspectives, they do imply some additional results, both positive and negative, on the approximation of monotone submodular functions by simpler classes of submodular functions (as illustrated in Table 1 and discussed in detail in Appendix G).

Let us now briefly discuss the main techniques that we use to obtain our results. In contrast to previous works [13, 2, 1], *arbitrary* submodular functions, as opposed to *monotone* submodular functions, present different challenges. As an illustration, for approximating a submodular function  $f$  via a cut function of a graph  $G$ , consider the case when there is a non-trivial set  $\emptyset \neq S \neq U$  for which  $f(S) = 0$ . Then the weight of the cut  $(S, \bar{S})$  in  $G$  is forced to be zero. Indeed, *all* sets  $S$  with  $f(S) = 0$  must be in a correspondence with cuts in  $G$  of value zero. Thus, given a submodular function  $f$ , our construction of  $G$  optimizes for the minimizers of the submodular function  $f$ . Surprisingly, this can be shown to give the best possible approximation. For a *symmetric* submodular function  $f$ , we show that it suffices to use the cut function of a *tree* (as opposed to a general undirected graph) utilizing the Gomory-Hu tree representation [15, 21] of  $f$ .

For approximating budgeted additive functions by coverage functions, we first give a randomized construction achieving an approximation factor of  $e/(e-1)$ . We then show that this is the best possible approximation factor as characterized by a linear program. The proof of the lower bound of  $e/(e-1)$  uses linear programming duality and proceeds by presenting a feasible dual solution to the linear program achieving an objective value of  $e/(e-1)$  in the limit. We would like to point out that all our results are algorithmic and the claimed approximations can be found in polynomial time given a value oracle for the submodular function.

## 1.2 Related Work

Goemans et al. [13] considered the problem of how well a given monotone submodular function  $f$  can be approximated when only polynomially many value oracle queries are permitted. They presented an approximation of  $O(\sqrt{n} \log n)$ , and an improved guarantee of  $\sqrt{n+1}$  in the case that  $f$  is a matroid rank function. Implicit in this algorithm and relevant to our setting is an approximation of all monotone submodular functions by budgeted additive functions (Appendix G). The current best lower bound for the problem studied by [13] is given by Svitkina and Fleischer [25] and is

$\Omega(\sqrt{n/\log n})$ .

Balcan and Harvey [2] take the learning perspective to the study of the complexity of submodular functions. They study the problem of probabilistically learning a monotone submodular function, given the values the function takes on a polynomial sized sample of its domain. They provide a lower bound of  $\Omega(n^{1/3})$  on the best possible approximation a learning algorithm can give to the submodular function, even when it knows the underlying sampling distribution, and the submodular function to be learned is Lipschitz. Another result with this perspective is by Balcan et al. [3] who show that a symmetric non-monotone submodular function can be approximated to within  $\sqrt{n}$  by the square root of a quadratic function. Furthermore, they show how to learn such submodular functions.

Badanidiyuru et al. [1], motivated by the problem of communicating bidders' valuations in combinatorial auctions, study how well specific classes of set functions can be approximated given the constraint that the approximating function be representable in polynomially many bits. They named such an approximation a *sketch*, proving that coverage functions admit sketches with an arbitrarily good approximation factors. Additionally, for the larger class of monotone subadditive functions, they construct sketches that achieve an approximation of  $\sqrt{n} \cdot \text{polylog}(n)$ . Combining the results of Badanidiyuru et al. [1] and Balcan and Harvey [2], a lower bound of  $\Omega(n^{1/3}/\log^2 n)$  follows for the approximation of monotone submodular functions by budgeted additive functions (Appendix G).

Testing of submodular functions has been studied recently by Seshadri and Vondrák [24] for general monotone submodular functions and for coverage functions by Chakrabarty and Huang [8]. The goal here is to query the function on few domain points and answer whether a function is *close* to being submodular or not. The measure of closeness is the fraction of the domain in which the function needs to be modified so as to make it submodular.

### 1.3 Preliminaries and Formal Statement of Results

Given a ground set  $U$ , a function  $f : 2^U \rightarrow \mathbb{R}_+$  is called *submodular* if for all subsets  $S, T \subseteq U$ , we have  $f(S) + f(T) \geq f(S \cup T) + f(S \cap T)$ . A submodular function  $f$  is called non-negative if  $f(S) \geq 0$  for each  $S \subseteq U$ . In this paper, we only consider non-negative submodular functions. A submodular function  $f$  is called *symmetric* if  $f(S) = f(U \setminus S)$  for each  $S \subseteq U$ , and *monotone* if  $f(S) \leq f(T)$  for each  $S \subseteq T \subseteq U$ . We say that a class of functions  $\mathcal{G}$   $\theta$ -*approximates* another class  $\mathcal{F}$ , if for every  $f \in \mathcal{F}$  there exists a  $g \in \mathcal{G}$  such that  $f(S) \leq g(S) \leq \theta \cdot f(S)$  for any  $S \subseteq U$ . We denote by  $n$  the size of the ground set  $U$ . We now define certain subclasses of submodular functions that we consider in the paper.

**Definition 1** (Coverage function). *A function  $f$  is a coverage function if there exists an auxiliary ground set  $Z$ , a weight function  $w : Z \rightarrow \mathbb{R}_+$  and family of subsets  $\{A_i : A_i \subseteq Z, i \in U\}$  such that  $\forall S \subseteq U, f(S) = \sum_{z \in \cup_{i \in S} A_i} w(z)$ .*

**Definition 2** (Budgeted additive function). *A function  $f$  is a budgeted additive function if there exist non-negative reals  $a_i$  for each  $i \in U$ , and a non-negative real  $B$  such that  $\forall S \subseteq U, f(S) = \min\{B, \sum_{i \in S} a_i\}$ .*

It is well known that coverage functions and budgeted additive functions are monotone submodular functions.

**Definition 3** (Cut function). *A function  $f$  is a directed cut function if there exists a directed graph  $G = (U, A)$  with non-negative arc weights  $w : A \rightarrow \mathbb{R}_+$ , such that  $\forall S \subseteq U$ ,  $f(S) = w(\delta^+(S))$ , where  $\delta^+(S)$  denotes the set of outgoing arcs, with their tails in  $S$  and heads in  $\bar{S}$ , and  $w(F) \triangleq \sum_{a \in F} w(a)$  for any subset  $F \subseteq A$  of arcs.*

Similarly, one can define  $f$  to be the *undirected cut function* of an undirected graph by substituting  $\delta^+(S)$  with  $\delta(S)$ , the set of edges with exactly one endpoint in  $S$ . It is well known that cut functions, whether directed or undirected, are submodular. Furthermore, clearly, undirected cut functions are symmetric.

Let us now formally state our main results:

**Theorem 1.1.** *Let  $f : 2^U \rightarrow \mathbb{R}_+$  be a non-negative submodular function with  $f(\emptyset) = f(U) = 0$ . Then the class of directed cut functions  $(n^2/4)$ -approximates  $f$ . Moreover, there exists a non-negative submodular function  $f : 2^U \rightarrow \mathbb{R}_+$  with  $f(\emptyset) = f(U) = 0$  such that any directed cut function cannot approximate  $f$  within a factor better than  $n^2/4$ .*

**Theorem 1.2.** *Let  $f : 2^U \rightarrow \mathbb{R}_+$  be a non-negative symmetric submodular function with  $f(\emptyset) = 0$ . Then the class of undirected cut functions  $(n - 1)$ -approximates  $f$ . Moreover, there exists a symmetric submodular function  $f : 2^U \rightarrow \mathbb{R}_+$  with  $f(\emptyset) = 0$  such that any undirected cut function cannot approximate  $f$  within a factor better than  $n/4$ .*

**Theorem 1.3.** *Let  $f : 2^U \rightarrow \mathbb{R}_+$  be a budgeted additive function. Then coverage functions  $(e/(e - 1))$ -approximates  $f$ . Moreover, for every fixed  $\varepsilon > 0$ , there exists a budgeted additive function  $f : 2^U \rightarrow \mathbb{R}_+$  such that any coverage function cannot approximate  $f$  within a factor better than  $e/(e - 1) - \varepsilon$ .*

Theorems 1.1, 1.2 and 1.3, are proved in Sections 2, 3 and 4 respectively.

## 2 Approximating General Submodular Functions by Directed Cut Functions of Graphs

In this section we prove Theorem 1.1 which provides a tight approximation of a non-negative submodular function  $f$  using a directed cut function of a graph  $G$ . Before proving the main result of this section, we first state a technical lemma, whose proof we defer to Appendix C.

**Lemma 2.1.** *For every submodular function  $f$ , and any collection of sets  $A_1, A_2, \dots, A_n \subseteq U$ :  $f(\cap_{i=1}^n A_i) \leq \sum_{i=1}^n f(A_i)$ .*

We are now ready to prove Theorem 1.1.

*Proof of Theorem 1.1.*

**Upper Bound:** Given a submodular function  $f$ , we construct a directed graph  $G = (U, A)$  with non-negative weights  $w$  on the arcs such for every  $S \subseteq U$ ,  $f(S) \leq w(\delta^+(S)) \leq n^2/4 \cdot f(S)$ . For every  $(u, v) \in U \times U$  and  $u \neq v$ , introduce a directed arc from  $u$  to  $v$  with weight:  $w_{uv} = f(T_{uv})$  where  $T_{uv} = \operatorname{argmin} \{f(R) : R \subseteq U, u \in R, v \notin R\}$ . We start by proving that:

$$f(S) \leq w(\delta^+(S)) \quad \forall S \subseteq U. \quad (1)$$

If  $S = U$  or  $S = \phi$ , then clearly  $w(\delta^+(S)) = f(S) = 0$  and (1) holds. We now restrict our attention to the case where  $S, \bar{S} \neq \emptyset$ . For any  $u \in S$  note that  $u \in \cap_{v \in \bar{S}} T_{uv}$ , since the definition of  $T_{uv}$  implies that  $u \in T_{uv}$  for all  $v \in \bar{S}$ . Additionally, for any  $w \in \bar{S}$  note that  $w \notin \cap_{v \in \bar{S}} T_{uv}$ , since the definition of  $T_{uv}$  implies that  $w \notin T_{uw}$ . Thus, one can conclude that  $\cup_{u \in S} \cap_{v \in \bar{S}} T_{uv} = S$  and therefore,

$$f(S) \stackrel{(i)}{\leq} \sum_{u \in S} f(\cap_{v \in \bar{S}} T_{uv}) \stackrel{(ii)}{\leq} \sum_{u \in S} \sum_{v \in \bar{S}} w_{uv} = w(\delta^+(S)).$$

Inequality (i) is derived from the fact that  $f$  is submodular and non-negative. Inequality (ii) is derived from Lemma 2.1 and the definition of  $w_{uv}$ . This concludes the proof of (1).

We continue by proving that:

$$w(\delta^+(S)) \leq \frac{n^2}{4} f(S) \quad \forall S \subseteq U. \quad (2)$$

If  $S = U$  or  $S = \phi$ , then clearly  $w(\delta^+(S)) = f(S) = 0$  and (2) holds. We now restrict our attention to the case where  $S, \bar{S} \neq \emptyset$ . Note that for any  $u \in S$  and  $v \in \bar{S}$ , by the definition of  $T_{uv}$ :  $f(T_{uv}) \leq f(S)$ . Thus, one can conclude that:

$$w(\delta^+(S)) \stackrel{(i)}{=} \sum_{u \in S} \sum_{v \in \bar{S}} f(T_{uv}) \stackrel{(ii)}{\leq} \sum_{u \in S} \sum_{v \in \bar{S}} f(S) \stackrel{(iii)}{\leq} \frac{n^2}{4} f(S).$$

Equality (i) is by the definition of weights  $w_{uv}$ . Inequality (ii) is by the definition of  $T_{uv}$ . Inequality (iii) is by the fact that the number of pairs  $(u, v) \in S \times \bar{S}$  is at most  $n^2/4$ . This concludes the proof of (2). Combining both (1) and (2) concludes the proof of the upper bound of the theorem.

**Lower Bound:** Assume that  $n$  is even and fix an arbitrary  $A \subseteq U$  of size  $|A| = n/2$ . Consider the following function  $f$

$$f(S) = \begin{cases} 1 & \text{if } S \cap A \neq \emptyset, \bar{A} \setminus S \neq \emptyset \\ 0 & \text{otherwise} \end{cases}$$

Namely,  $f(S)$  is the indicator function that  $S$  hits  $A$  but does not hit all of  $\bar{A}$ . A simple check shows that  $f$  is submodular. Let  $G = (U, A)$  be a weighted graph with non-negative weights  $w : A \rightarrow \mathbb{R}_+$  on the arcs whose directed cut function satisfies for each set  $S \subseteq U$ ,  $f(S) \leq w(\delta^+(S)) \leq \theta \cdot f(S)$  for some  $\theta$ . We will show that  $\theta \geq \frac{n^2}{4}$  proving the lower bound.

First, we prove that the arcs with non-zero weight must go from  $A$  to  $\bar{A}$ . We consider the following cases.

1. Consider an edge  $(u, v) \in A \times A$ . But  $(u, v) \in \delta^+(U \setminus \{v\})$  and  $w(\delta^+(U \setminus \{v\})) \leq \theta \cdot f(U \setminus \{v\}) = 0$  since  $\bar{A} \setminus (U \setminus \{v\}) = \emptyset$ . Thus,  $w_{(u,v)} = 0$ .
2. Consider an edge  $(u, v) \in \bar{A} \times \bar{A}$ . But  $(u, v) \in \delta^+(\bar{A} \setminus \{v\})$  and  $w(\delta^+(\bar{A} \setminus \{v\})) \leq \theta \cdot f(\bar{A} \setminus \{v\}) = 0$  since  $A \cap (\bar{A} \setminus \{v\}) = \emptyset$ . Thus,  $w_{(u,v)} = 0$ .
3. Consider an edge  $(u, v) \in \bar{A} \times A$ . But  $(u, v) \in \delta^+(\bar{A})$  and  $w(\delta^+(\bar{A})) \leq \theta \cdot f(\bar{A}) = 0$  since  $A \cap \bar{A} = \emptyset$ . Hence,  $w_{(u,v)} = 0$ .



Therefore, all arcs with non-zero weight must go from  $A$  to  $\bar{A}$ . For any  $u \in A$  and  $v \in \bar{A}$  note that  $w_{(u,v)} \geq 1$  since:

$$w_{(u,v)} = w(\delta^+(\{u\} \cup (\bar{A} \setminus \{v\}))) \stackrel{(i)}{\geq} f(\{u\} \cup (\bar{A} \setminus \{v\})) \stackrel{(ii)}{=} 1. \quad (3)$$

Inequality (i) is derived from the fact  $w(\delta^+(S)) \geq f(S)$  for each set  $S$ . Equality (ii) is by the definition of  $f$ . Furthermore, note that:

$$\frac{n^2}{4} = |A| \cdot |\bar{A}| \stackrel{(i)}{\leq} w(\delta^+(A)) \stackrel{(ii)}{\leq} \theta \cdot f(A) \stackrel{(iii)}{=} \theta. \quad (4)$$

Inequality (i) is derived from inequality (3). Inequality (ii) is derived from the fact that  $w(\delta^+(S)) \leq \theta f(S)$  for each set  $S \subseteq U$ . Equality (iii) is by the definition of  $f$ . Note that inequality (4) implies that  $\theta \geq \frac{n^2}{4}$ , thus, concluding the proof of the lower bound of the theorem.  $\blacksquare$

### 3 Approximating Symmetric Submodular Functions by Undirected Cut Functions of Graphs

In this section, we prove Theorem 1.2 which provides upper and lower bounds on the approximation of a symmetric submodular function using an undirected cut function of a graph. For the upper bound, our algorithm uses Gomory-Hu trees of symmetric submodular functions [15, 21]. Given a symmetric non-negative submodular function  $f$ , a tree  $T = (U, E_T)$  is a Gomory-Hu tree if for every edge  $e = (u, v) \in E_T$ :  $f(R_e) = \min \{f(R) : R \subseteq U, u \in R, v \notin R\}$ , where  $R_e$  is one of the two connected components obtained after removing  $e$  from  $T$  (since  $f$  is symmetric, it does not matter which one of the two connected components we choose). In other words, in a Gomory-Hu tree, the cut  $e = (u, v)$  induced in  $T$ , corresponds to a minimum value subset that separates  $u$  and  $v$ . We prove that the cut function of the Gomory-Hu tree of  $f$  is a good approximation.

*Proof of Theorem 1.2.*

**Upper Bound:** Let  $f$  be a symmetric submodular function. We shall construct an undirected tree  $T = (U, E_T)$  with non-negative weights  $w : E \rightarrow \mathbb{R}_+$  on the edges such that for every  $S \subseteq U$ ,  $f(S) \leq w(\delta(S)) \leq (n-1) \cdot f(S)$ . We set  $T$  to be a Gomory-Hu tree of  $f$  and let the weight of any edge  $e = \{u, v\}$  to be  $f(R_e)$  where  $R_e$  is the one of the two connected components obtained after removing edge  $e$ . As mentioned above, the weight of edge  $e = \{u, v\}$  is the minimum of  $f(R)$  over all  $R$  separating  $u$  and  $v$ .

Fix an arbitrary  $S \subseteq U$  and denote by  $\{e_1, \dots, e_k\}$  all the edges crossing the cut that  $S$  defines in  $T$ . Let  $T_1, \dots, T_{k+1}$  denote the partition of  $U$  induced by deleting the edges  $e_1, \dots, e_k$  from  $T$ . Furthermore, denote by  $\{S_1, \dots, S_p\}$  the non-empty sets in  $\{T_i \cap S : 1 \leq i \leq k+1\}$ . Observe that  $S_1, \dots, S_p$  is a partition of  $S$ . Since each  $e_i$ ,  $1 \leq i \leq k$ , has exactly one vertex in  $S$  and the other in  $\bar{S}$ , we can associate  $e_i$  with a *unique* set from  $S_1, \dots, S_p$ , the set containing one of the endpoints of  $e$ . Additionally, let us denote  $F_i$  to be the edges which are associated with set  $S_i$  for each  $1 \leq i \leq p$ . Clearly  $F_1, \dots, F_p$  form a partition of  $\{e_1, \dots, e_k\}$ .

We claim that for every  $1 \leq i \leq p$ :

$$f(S_i) \leq \sum_{f \in F_i} f(R_f). \quad (5)$$

Recall that  $R_f$  is one of the connected component after removing edge  $f$  from  $T$ . Since  $S_i$  is a subset of a connected component formed after removing all the edges  $\{e_1, \dots, e_k\}$  from  $T$ , it must be contained in one of the components formed after removing edge  $f \in F_i$  from  $T$ . Without loss of generality, we assume that  $R_f \cap S_i = \emptyset$  for each edge  $f \in F_i$ . It is straightforward to see that  $\bigcap_{f \in F_i} \bar{R}_f = S_i$ . Now, we have

$$\sum_{f \in F_i} f(R_f) \stackrel{(i)}{\geq} f(\cup_{e \in F_i} R_f) \stackrel{(ii)}{=} f(\overline{\cup_{f \in F_i} R_f}) = f(\cap_{e \in F_i} \bar{R}_f) \stackrel{(iii)}{=} f(S_i).$$

Inequality (i) is derived from the fact that  $f$  is submodular and non-negative. Equality (ii) is derived from the symmetry of  $f$ . Equality (iii) is derived from the fact that  $\bigcap_{f \in F_i} \bar{R}_f = S_i$ .

We start by proving that:

$$f(S) \leq w(\delta(S)) \quad \forall S \subseteq U. \quad (6)$$

This can be proved as follows:

$$w(\delta(S)) \stackrel{(i)}{=} \sum_{i=1}^k f(R_{e_i}) \stackrel{(ii)}{=} \sum_{i=1}^p \sum_{f \in F_i} f(R_f) \stackrel{(iii)}{\geq} \sum_{i=1}^p f(S_i) \stackrel{(iv)}{\geq} f(\cup_{i=1}^p S_i) = f(S).$$

Equality (i) is by the definition of edge weights in  $T$ . Equality (ii) is by the fact that  $F_1, \dots, F_p$  form a partition of  $\{e_1, \dots, e_k\}$ . Inequality (iii) is derived from inequality (5). Inequality (iv) is derived from the fact that  $f$  is submodular and non-negative. This concludes the proof of (6).

We continue by proving that:

$$w(\delta(S)) \leq (n-1) \cdot f(S) \quad \forall S \subseteq U. \quad (7)$$

Let  $u$  and  $v$  be the endpoints of edge  $e_i$  and without loss of generality assume that  $u \in S$  and  $v \notin S$ . Note that for every  $1 \leq i \leq k$ ,  $f(R_{e_i}) \leq f(S)$  since  $S$  is a candidate set separating  $u$  and  $v$ . Hence, one can conclude that:

$$w(\delta(S)) \stackrel{(i)}{=} \sum_{i=1}^k f(R_{e_i}) \stackrel{(ii)}{\leq} k \cdot f(S) \stackrel{(iii)}{\leq} (n-1) \cdot f(S).$$

Equality (i) is by the definition of edge weights in  $T$ . Inequality (ii) is what we proved above, and inequality (iii) is derived from the fact that  $T$  contains at most  $n-1$  edges, thus,  $k \leq n-1$ . This concludes the proof of (7). Combining both (6) and (7) concludes the proof of the upper bound of the theorem.

**Lower Bound:** Refer to Appendix A. ■

## 4 Approximating Budgeted Additive Functions by Coverage Functions

In this section, we present matching upper and lower bounds for approximating budgeted additive functions by coverage functions (Theorem 1.3)<sup>2</sup>. The following lemma from Chakrabarty and Huang [8] provides the alternate representation of coverage functions used in our proof of the lower bound.

---

<sup>2</sup>It is easy to show that a coverage function can be written exactly as a sum of budgeted additive functions.



**Lemma 4.1.** [8] A function  $f : 2^U \rightarrow \mathbb{R}_+$  is a coverage function if and only if there exist reals  $x_T \geq 0$  for each  $T \subseteq U$  such that  $f(S) = \sum_{T: T \cap S \neq \emptyset} x_T$  for each  $S \subseteq U$ .

*Proof of Theorem 1.3.*

**Upper Bound:** Refer to Appendix B.

**Lower Bound:** We will construct a budgeted additive function which cannot be approximated by coverage functions to factor better than  $\frac{e}{e-1} - \epsilon$  for any  $\epsilon > 0$ . We will consider the family of budgeted additive function  $f_k$ , parameterized by the size of domain  $|U| = n$  they are defined on, where  $n = k^2$  for some integer  $k$ . Under  $f_k$ , all  $n = k^2$  items have value one and the budget is  $k$ . Please note that these also constitute a family of uniform matroid rank functions. Therefore,

$$f_k(S) = \begin{cases} |S| & \text{if } |S| \leq k \\ k & \text{o.w.} \end{cases} \quad (8)$$

Let  $h_k$  be a coverage function that gives the maximum value of  $\beta$  such that  $\forall S \subseteq [n], \beta \cdot f_k(S) \leq h_k(S) \leq f_k(S)$  and  $\alpha_k$  be the value of  $\beta$  as given by  $h_k$ . Observe that here function  $h_k$  is always smaller than the function  $f_k$ . The function  $\frac{h_k}{\beta}$  would give a  $\frac{1}{\beta}$ -approximation for approximating function  $f_k$ . This slight change in notation helps for exposition below. We shall show that as  $k \rightarrow \infty$ ,  $\alpha_k$  tends to a value that is at most  $1 - 1/e$ . This shall prove our claim.

Using Lemma 4.1, we note that  $\alpha_k$  can be characterized by a solution to a linear problem (P) given below. Here, the variables are  $x_T$ , one for each set  $T \subseteq U$ . The dual (D) of this linear program is given alongside. We will construct a dual solution of value approaching  $1 - \frac{1}{e}$  as  $k \rightarrow \infty$ . Since every feasible dual solution is an upper bound on  $\alpha_k$ , the result follows.

$$\begin{array}{l|l} \begin{array}{l} \max \alpha_k \\ \text{subject to} \\ \forall S \subseteq U, \quad \sum_{T \cap S \neq \emptyset} x_T \leq f_k(S) \\ \forall S \subseteq U, \quad \alpha_k f_k(S) - \sum_{T \cap S \neq \emptyset} x_T \leq 0 \\ \forall S \subseteq U, \quad x_S \geq 0 \end{array} & \text{(P)} & \begin{array}{l} \min \sum_{S \subseteq U} f_k(S) \cdot u_S \\ \text{subject to} \\ \forall S \subseteq U, \quad \sum_{T \cap S \neq \emptyset} (u_T - v_T) \geq 0 \\ \sum_{S \subseteq U} f_k(S) \cdot v_S \geq 1 \end{array} & \text{(D)} \end{array}$$

Since,  $f(\cdot)$  is symmetric across sets of the same cardinality, we can assume, without loss of generality, that the optimal dual solution is also symmetric. Specifically, the values of the dual variables  $u_T$  and  $v_T$  shall depend only the cardinality  $|T|$ . Let us write the symmetrized dual program.

$$\min \sum_{j=1}^k j \cdot \binom{n}{j} \cdot u_j + \sum_{j=k+1}^n k \cdot \binom{n}{j} \cdot u_j \quad \text{Symmetrized Dual Program}$$

subject to

$$\forall j \in [n], \quad \sum_{i=1}^n \left( \binom{n}{i} - \binom{n-j}{i} \right) (u_i - v_i) \geq 0 \quad (9)$$

$$\sum_{j=1}^k j \cdot \binom{n}{j} \cdot v_j + \sum_{j=k+1}^n k \cdot \binom{n}{j} \cdot v_j \geq 1 \quad (10)$$

Let  $c_j$  denote the coefficient of  $v_k$  in the equation corresponding to set size  $j$ , i.e.,  $c_j = \binom{n}{k} - \binom{n-j}{k}$ . Further, define  $\Delta c_j = c_{j+1} - c_j$ .

We give the following solution to the dual linear program. Let  $v_k = \frac{1}{\binom{n}{k} \cdot k}$ ,  $u_1 = \Delta c_k \cdot v_k$  and  $u_n = (c_k - k \cdot \Delta c_k) \cdot v_k$ . Rest of the variables are set to zero.

We first show that the above solution is feasible for the dual and has objective value that tends to  $1 - 1/e$  as  $k \rightarrow \infty$ . It is easy to see that with the proposed setting of  $v_k$ , Equation (10) is satisfied. To show that Equation (9) is satisfied, we show that  $\forall j \in [n]$ ,  $j \cdot u_1 + u_n \geq \left( \binom{n}{k} - \binom{n-j}{k} \right) v_k$ . Using our notation, it suffices to show that for all  $j \in [n]$ ,  $(j - k) \cdot \Delta c_k + c_k \geq c_j$ .

**Claim 4.2.**  $c_j$  is an increasing function of  $j$  and  $\Delta c_j = c_{j+1} - c_j$  is a decreasing function of  $j$ .

*Proof.*  $\Delta c_j = c_{j+1} - c_j = \left( \binom{n}{k} - \binom{n-j-1}{k} \right) - \left( \binom{n}{k} - \binom{n-j}{k} \right) = \binom{n-j}{k} - \binom{n-j-1}{k} = \binom{n-j-1}{k-1}$ . ■

**Claim 4.3.** For all  $j \in [n]$ ,  $(j - k) \cdot \Delta c_k + c_k \geq c_j$ .

*Proof.* • For  $j = k + i$  such that  $0 \leq i \leq n - k$ : LHS =  $i \cdot \Delta c_k + c_k \geq c_{k+i}$  = RHS

• For  $j = k - i$  with  $0 \leq i \leq k$ : LHS =  $c_k - i \cdot \Delta c_k \geq c_k - \sum_{q=k-i}^{k-1} \Delta c_q = c_{k-i} \cdot v_k =$  RHS where the second inequality in both the cases follows because  $\Delta c_j$  is a decreasing function in  $j$ . ■

Let us now bound the value of the dual objective function. Look at the value that the dual objective function attains with this setting of variables. The function value is  $n \cdot u_1 + k \cdot u_n = (n \cdot \Delta c_k + k \cdot (c_k - k \cdot \Delta c_k)) \cdot v_k$ . Since  $n = k^2$ , the dual objective value is equal to

$$k \cdot c_k \cdot v_k = 1 - \frac{\binom{k^2-k}{k}}{\binom{k^2}{k}}.$$

This quantity tends to  $1 - 1/e$  as  $k \rightarrow \infty$ . ■

## 5 Future Directions

We mention here a couple of main research questions that are left open by the present work. The first is how well a non-negative monotone submodular function can be approximated by the sum of matroid rank functions. Dughmi et al. [11] show that the Hessian matrix for a matroid rank sum has to be negative semi-definite, and it is easy to come with a budgeted additive function that does not obey this property. Hence, we cannot hope for the best approximation factor for a submodular function by a matroid rank sum to be 1; in fact we can show that the approximation factor cannot be better than some constant bounded away from 1. In terms of positive results, a  $O(\sqrt{n})$  factor approximation follows from [13] and a  $O(\frac{\max_{e \in U} f(e)}{\min_{e \in U} f(e)})$  follows from a result in Section 44.6(B) in [22].

The second is approximating a non-negative symmetric submodular function by a hypergraph cut function (in this paper, we only considered graph cut functions). The lower bound example in the paper for graph cut functions can be extended to show that a  $r$ -regular hypergraph cannot approximate to a factor better than  $O(\frac{n}{r})$ . In terms of positive results, we know no better than the ones mentioned in this paper.

## References

- [1] Ashwinkumar Badanidiyuru, Shahar Dobzinski, Hu Fu, Robert Kleinberg, Noam Nisan, and Tim Roughgarden. Sketching valuation functions. In *SODA*, pages 1025–1035, 2012.
- [2] Maria-Florina Balcan and Nicholas J. A. Harvey. Learning submodular functions. In *STOC*, pages 793–802, 2011.
- [3] Maria-Florina Balcan, Nicholas J.A. Harvey, and Satoru Iwata. Learning symmetric non-monotone submodular functions. In *Discrete Optimization in Machine Learning (DISCML)*, 2012.
- [4] Yair Bartal. Probabilistic approximations of metric spaces and its algorithmic applications. In *FOCS*, pages 184–193, 1996.
- [5] J. Bourgain. On lipschitz embedding of finite metric spaces in Hilbert space. *Israel Journal of Mathematics*, 52(1):46–52, March 1985.
- [6] Niv Buchbinder, Joseph Naor, R. Ravi, and Mohit Singh. Approximation algorithms for online weighted rank function maximization under matroid constraints. In *ICALP (1)*, pages 145–156, 2012.
- [7] David Buchfuhrer, Michael Schapira, and Yaron Singer. Computation and incentives in combinatorial public projects. In *ACM EC*, pages 33–42, 2010.
- [8] Deeparnab Chakrabarty and Zhiyi Huang. Testing coverage functions. In *ICALP (1)*, pages 170–181, 2012.
- [9] G. Cornuejols, M. L. Fisher, and G. L. Nemhauser. Location of bank accounts to optimize float: an analytic study of exact and approximate algorithms. *Management Sciences*, 23:789–810, 1977.

- [10] Shaddin Dughmi. A truthful randomized mechanism for combinatorial public projects via convex optimization. In *ACM EC*, pages 263–272, 2011.
- [11] Shaddin Dughmi, Tim Roughgarden, and Qiqi Yan. From convex optimization to randomized mechanisms: toward optimal combinatorial auctions. In *STOC*, pages 149–158, 2011.
- [12] Jittat Fakcharoenphol, Satish Rao, and Kunal Talwar. A tight bound on approximating arbitrary metrics by tree metrics. *J. Comput. Syst. Sci.*, 69(3):485–497, 2004.
- [13] Michel X. Goemans, Nicholas J. A. Harvey, Satoru Iwata, and Vahab S. Mirrokni. Approximating submodular functions everywhere. In *SODA*, pages 535–544, 2009.
- [14] Michel X. Goemans and David P. Williamson. Improved approximation algorithms for maximum cut and satisfiability problems using semidefinite programming. *J. ACM*, 42(6):1115–1145, 1995.
- [15] Ralph E Gomory and Tien Chung Hu. Multi-terminal network flows. *Journal of the Society for Industrial & Applied Mathematics*, 9(4):551–570, 1961.
- [16] M. Grtschel, L. Lovsz, and A. Schrijver. The ellipsoid method and its consequences in combinatorial optimization. *Combinatorica*, 1:169–197, 1981.
- [17] Martin Hoefer and Thomas Kesselheim. Secondary spectrum auctions for symmetric and submodular bidders. In *ACM EC*, pages 657–671, 2012.
- [18] Satoru Iwata, Lisa Fleischer, and Satoru Fujishige. A combinatorial strongly polynomial algorithm for minimizing submodular functions. *J. ACM*, 48(4):761–777, 2001.
- [19] W. Johnson and J. Lindenstauss. Extensions of Lipschitz maps into a Hilbert space. *Contemporary Mathematics*, 1984.
- [20] G. L. Nemhauser, L. A. Wolsey, and M. L. Fisher. An analysis of approximations for maximizing submodular set functions - I. *Mathematical Programming*, 14:265–294, December 1978.
- [21] M. Queyranne. A gomory-hu tree for symmetric submodular functions. unpublished manuscript, Faculty of Commerce, University of British Columbia, 1993.
- [22] A. Schrijver. *Combinatorial Optimization - Polyhedra and Efficiency*. Springer, 2003.
- [23] Alexander Schrijver. A combinatorial algorithm minimizing submodular functions in strongly polynomial time. *J. Comb. Theory, Ser. B*, 80(2):346–355, 2000.
- [24] C. Seshadhri and Jan Vondrák. Is submodularity testable? In *ICS*, pages 195–210, 2011.
- [25] Zoya Svitkina and Lisa Fleischer. Submodular approximation: Sampling-based algorithms and lower bounds. *SIAM J. Comput.*, 40(6):1715–1737, 2011.

## A Lower Bound of Theorem 1.2

*Lower Bound of Theorem 1.2.* Consider the following symmetric submodular function  $f$ :

$$f(S) = \begin{cases} 1 & \text{if } S \neq \emptyset, U \\ 0 & \text{otherwise} \end{cases}$$

Let  $G = (U, E)$  be an edge weighted graph with non-negative weights  $w : E \rightarrow \mathbb{R}_+$  on the edges whose cut function satisfies  $f(S) \leq w(\delta(S)) \leq \theta \cdot f(S)$  for each set  $S \subseteq U$  for some  $\theta$ . We will show that  $\theta \geq \frac{n}{4}$ .

For any vertex  $v \in U$ ,  $1 = f(\{v\}) \leq w(\delta(\{v\}))$ . Thus, the total weight of edges in  $G$  is at least  $\frac{1}{2} \sum_{v \in U} w(\delta(\{v\})) \geq \frac{n}{2}$ . Every undirected graph has a non-trivial cut that contains at least half the total weight of edges in the graph, thus, there exists a cut  $S \subseteq U$ ,  $S \neq \emptyset, U$ , where  $w(\delta(S)) \geq \frac{n}{4}$ . The existence of such a cut can be shown by picking a cut at random where each vertex is in  $S$  with probability  $\frac{1}{2}$  independently. The expected weight of the cut will be exactly half the total weight of all edges. Now, we have  $\frac{n}{4} \leq w(\delta(S)) \leq \theta \cdot f(S) = \theta$ , concluding the proof of the lower bound of the theorem. ■

## B Upper Bound of Theorem 1.3

*Upper Bound of Theorem 1.3.* Consider any budgeted additive function  $f(\cdot)$  over some domain  $U$ , with budget  $B$  and the values of the elements be denoted by  $v_1, v_2, \dots, v_n$  where  $n = |U|$ . Without loss of generality, we assume all these values to be integers. Take an auxiliary ground set  $G$  of size  $B$ . For each  $i \in U$ , construct a set  $A_i \subseteq G$ , formed by choosing  $v_i$  points (with replacement) at random from  $G$ . Consider function  $g : 2^U \rightarrow \mathbb{Z}$ , defined as  $g(S) = |\cup_{i \in S} A_i|$  for all  $S \subseteq U$ .

By definition,  $g(\cdot)$  is a coverage function. Furthermore, it is easy to see that for all  $S \subseteq U$ ,  $g(S) \leq f(S)$ . We now show that  $\mathbb{E}[g(S)] \geq (1 - 1/e) \cdot f(S)$ , where the expectation is taken over the randomness of the procedure described to construct  $g(S)$ . Note that  $g'(\cdot) = \mathbb{E}[g(\cdot)]$  is a coverage function. Consider any set  $S \subseteq U$ . Let  $f(S) = V$ , i.e.,  $\sum_{i \in S} v_i = V$ . Consider the case when  $V < B$ . Consider any point in auxiliary ground set  $G$ . The probability that this point is not covered by any of the sets  $A_i$  for  $i \in S$  is at most  $(1 - 1/B)^V$ . Hence, the expected value of  $|\cup_{i \in S} A_i|$  is at least  $B \cdot (1 - (1 - 1/B)^V) \geq B \cdot (1 - e^{-V/B}) \geq (1 - 1/e) \cdot V$ . Here we use the inequality  $1 - e^{-x} \geq (1 - 1/e) \cdot x$ . Hence,  $|\cup_{i \in S} A_i| \geq (1 - 1/e) \cdot f(S)$ . The proof for the case when  $V = B$  is similar. Thus for each set  $S \subseteq U$ , we have

$$(1 - \frac{1}{e})f(S) \leq g'(S) \leq f(S).$$

Thus, we obtain the function  $\frac{e}{e-1}g'(\cdot)$  approximates  $f$  within a factor of  $\frac{e}{e-1}$ . ■

## C Proof of Lemma 2.1

*Proof of Lemma 2.1.* The following inequalities are derived from the definition of submodularity:

$$\begin{aligned}
f(A_1) + f(A_2) &\geq f(A_1 \cap A_2) + f(A_1 \cup A_2) \\
f(A_3) + f(A_1 \cap A_2) &\geq f(A_1 \cap A_2 \cap A_3) + f((A_1 \cap A_2) \cup A_3) \\
&\vdots \\
f(A_n) + f(\cap_{i=1}^{n-1} A_i) &\geq f(\cap_{i=1}^n A_i) + f((\cap_{i=1}^{n-1} A_i) \cup A_n)
\end{aligned}$$

Summing up the above inequalities and canceling common terms on the two sides and using the fact that  $f$  is non-negative we obtain that  $\sum_{i=1}^n f(A_i) \geq f(\cap_{i=1}^n A_i)$ . ■

## D Uniform Submodular and Matroid Rank Functions

**Definition 4** (Uniform Submodular Function). *A submodular function is said to be uniform if the value it takes on a set depends only on the cardinality of the set.*

**Lemma D.1.** *Any non-negative, integer-valued, monotone, uniform, submodular function is 1-approximated by a sum of uniform matroid rank functions, and hence by a sum of budgeted additive functions.*

*Proof.* Consider an integer-valued, non-negative, monotone, uniform, submodular function  $f(\cdot)$  over the universe  $[n]$  and let  $f_k$  be the value  $f$  takes for sets  $S$  of cardinality  $k$ . Consider uniform matroid rank functions  $g_1, g_2, \dots, g_n$  where

$$g_i(S) = \begin{cases} |S| & |S| \leq i \\ i & |S| > i \end{cases} \quad (11)$$

We claim that there exist a set of  $\alpha_i$ 's, such that  $\alpha_i \geq 0$  for all  $i \in [n]$  such that

$$\forall j \in [n], f_j = \sum_{i=1}^k \alpha_i \cdot j + \sum_{i=k+1}^n \alpha_i \cdot i \quad (12)$$

It is easy to see that the above claim implies that  $f(S) = \sum_i \alpha_i \cdot g_i(S)$  for all  $S \subseteq [n]$ .

Now, we prove the claim. If, for every  $j \in [n-1]$ , we subtract equation  $j$  from  $j+1$ , we get the following set of equations

$$\forall j \in [n-1], f_{j+1} - f_j = \sum_{i=j+1}^n \alpha_i \quad (13)$$

From here, we can see that the following assignments to  $\alpha_i$  is a valid solution to the above equations. Set  $\alpha_1 = 2 \cdot f_2 - f_1$ ,  $\alpha_n = f_n - f_{n-1}$  and for  $i \notin \{1, n\}$ , set  $\alpha_i = 2 \cdot f_i - f_{i+1} - f_{i-1}$ . All the  $\alpha_i$ 's are positive since  $f$  is monotone and  $f$  is submodular.

Finally, it is easy to see that every uniform matroid rank function is also a budgeted additive function with all elements having value one, and the budget equal to the rank of the matroid. ■

## E Application to Online Submodular Function Maximization

We consider the problem of online submodular function maximization as studied by Buchbinder et. al [6]. We are given a universe  $U$  and matroid  $\mathcal{M} = (U, \mathcal{I})$ . In an online manner, at each step for  $1 \leq i \leq m$ , we are given a monotone submodular function  $f_i : 2^U \rightarrow \mathbb{R}_+$ . The goal is to maintain an independent set  $F_i \in \mathcal{I}$  at any step  $i$  such that  $F_i \subseteq F_{i+1}$ . The objective value to maximize is  $\sum_{i=1}^m f_i(F_i)$ . As in the notion of competitive analysis, any algorithm is compared to the best offline optimum  $\max_{O \in \mathcal{I}} \sum_{i=1}^m f_i(O)$ .

Buchbinder et. al [6] give a  $O(\log^2 n \log m \log f_{ratio})$ -competitive algorithm when each of the submodular function is weighted matroid rank function where  $f_{ratio} = \frac{\max_{i,a} f_i(\{a\})}{\min_{i,a: f_i(\{a\}) > 0} f_i(\{a\})}$ . In particular, the result applies when each of the functions  $f_i$  is a coverage function.

Using the fact every monotone submodular function can be approximated by a coverage function to a factor of  $O(\sqrt{n} \log n)$ , we directly obtain the following corollary.

**Corollary E.1.** *There is a  $O(\sqrt{n} \log^3 n \log m f_{ratio})$ -competitive online algorithm for the online submodular function maximization problem when each of the submodular functions is an arbitrary monotone submodular function.*

## F Approximating Monotone Submodular Functions by Coverage Functions and by Budgeted Additive Functions

The two main results of the section are the following.

**Theorem F.1.** *Coverage functions can approximate every non-negative monotone submodular function to within a factor  $O(\sqrt{n} \log n)$ . Additionally, the class of coverage functions cannot approximate every non-negative monotone submodular function to a factor within  $o\left(\frac{n^{1/3}}{\log^2 n}\right)$ .*

**Theorem F.2.** *The class of sum of budgeted additive functions can approximate every non-negative monotone submodular function to a factor within  $(\sqrt{n} \log n)$ . Additionally, the class of sum of budgeted additive functions cannot approximate every non-negative monotone submodular function to a factor within  $o\left(\frac{n^{1/3}}{\log^2 n}\right)$ .*

### F.1 Upper Bound

For the upper-bound, we show that budgeted additive functions can  $\sqrt{n} \log(n)$ -approximate the class of non-negative, monotone, submodular functions. Then we use Theorem 1.3 to infer that the coverage functions too can give approximately the same guarantee.

**Lemma F.3.** *The class of sum of budgeted additive functions can approximate every non-negative monotone submodular function to within a factor  $(\sqrt{n} \log n)$ .*

The following corollary follows easily from Lemma G.3 and Theorem 1.3

**Corollary F.4.** *The class of sum of coverage functions can approximate every non-negative monotone submodular function to factor  $O(\sqrt{n} \log n)$ .*



The proof of Lemma G.3 follows from Lemmas G.5 and G.6. Lemma G.5 [13] gives a particular function that  $\sqrt{n} \log(n)$ -approximates a general monotone, non-negative, sub-modular function, and Lemma G.6 implies that this approximating function can be written as a sum of budgeted additive functions.

**Lemma F.5.** [13] *For every monotone submodular function  $f : 2^U \rightarrow R_+$ , there exists positive reals  $a_e$  for each  $e \in E$  such that  $g : 2^U \rightarrow R_+$  defined as  $g(S) = \sqrt{\sum_{e \in S} a_e}$ , approximates  $f$  within factor  $\sqrt{n} \log(n)$ .*

**Lemma F.6.** *Every submodular function  $f : 2^{[n]} \rightarrow \mathbb{Z}^+$  of the form  $f(S) = g(\sum_{i \in S} a_i)$ , where  $a_i \in \mathbb{Z}^+$  and  $g$  is a non-negative, monotone, concave and integer valued on integral inputs, can be written as a sum of budgeted additive functions.*

*Proof.* Let  $m = \sum_{i=1}^n a_i$ . Consider the function  $h$ , over the domain  $[m]$ , defined as  $h(S) = g(|S|)$  for all  $S \subseteq [m]$ . Since  $g(\cdot)$  is a non-negative, monotone, concave function, it is easy to verify that  $h(\cdot)$  is a non-negative, monotone, submodular function. Construct  $n$  mutually disjoint sets  $A_i \subseteq [m]$  such that  $|A_i| = a_i$ . Clearly,  $\forall S \subseteq [n], f(S) = h(\cup_{i \in S} A_i)$ .

From Lemma D.1, we know that  $h(\cdot)$  can be expressed as  $\sum_{i=1}^m \alpha_i \cdot t_i(\cdot)$  where each  $t_i$  is a uniform matroid rank function with rank  $i$ , over the domain  $[m]$  and each  $\alpha_i \geq 0$ .

This implies that for all  $S \subseteq [n], f(S) = \sum_{i=1}^m \alpha_i \cdot t_i(\cup_{i \in S} A_i)$ . Now, for every  $i \in [m]$ , construct the budget additive function  $t'_i$ , defined as  $\forall S \subseteq [n], t'_i(S) = \min\{\sum_{i \in S} v_{ij}, B_i\}$ , where for  $j \in [n]$ , the value  $v_{ij} = a_j$  and the budget  $B_i$  is  $i$ . Since  $t_i$  is a uniform matroid rank function of rank  $i$  and  $A_i$ 's are mutually disjoint, we have for all  $i \in [m]$ ,

$$\forall S \subseteq [n], t'_i(S) = t_i(\cup_{i \in S} A_i) \tag{14}$$

Therefore, we get for all sets  $S \subseteq [n], f(S) = \sum_{i=1}^m t'_i(S)$ . ■

## F.2 Lower Bound

For the lower bound, we first show that sum of coverage functions cannot approximate the class of monotone, submodular functions well, and then use Theorem 1.3, to infer that, therefore, even the class of sum of budgeted additive functions cannot approximate a monotone submodular function well.

**Lemma F.7.** *The class of sum of coverage functions cannot approximate every non-negative monotone submodular function to a factor within  $o\left(\frac{n^{1/3}}{\log^2 n}\right)$ .*

An easy corollary of Lemma G.7 that follows from Theorem 1.3 is the following.

**Corollary F.8.** *The class of sum of budgeted additive functions cannot approximate every non-negative monotone submodular function to a factor within  $o\left(\frac{n^{1/3}}{\log^2 n}\right)$ .*

We now present the proof of Lemma G.7. We will need to use results from [1] and [2], for which we first present a definition.

**Definition 5.** *A  $\beta$ -sketch of a function  $f : 2^U \rightarrow \mathbb{R}$  is a polynomially sized (in  $|U|$  and  $1/(1-\beta)$ ) representable function  $g$  such that  $\forall S \subseteq U, \beta \cdot f(S) \leq g(S) \leq f(S)$ .*

The following result is from [1].

**Lemma F.9.** [1] *Coverage functions allow from arbitrary well sketches i.e., for any  $\epsilon > 0$ , there exists a  $1 - \epsilon$  sketch.*

The following result is from [2]. It gives a ‘large’ family of matroid rank functions, such that any two functions in the class have at least one point where the values that they take differ by a ‘significant’ factor.

**Lemma F.10.** [2] *For any  $k = 2^{o(n^{1/3})}$ , there exists a family of sets  $\mathcal{A} \subseteq 2^{[n]}$  with  $|\mathcal{A}| = k$  and a family of matroids  $\mathcal{M} = \{M_B | B \subseteq \mathcal{A}\}$  such that for all  $B \subseteq \mathcal{A}$ , it is the case that*

$$\forall S \in \mathcal{A}, r_{M_B}(S) = \begin{cases} 8 \log k & \text{if } S \in B \\ n^{1/3} & \text{if } S \notin B \end{cases} \quad (15)$$

where  $r_{M_B}$  is the rank function of the matroid  $M_B$

of Lemma G.7. Let the class of matroid rank functions on the domain of size  $n$  be  $\alpha$ -approximable by coverage functions, for some  $\alpha$ . That is, for a domain  $[n]$ , for all matroid rank function  $r$ , there exists a coverage function  $g$  such that  $\forall S \subseteq [n], r(S) \leq g(S) \leq \alpha \cdot r(S)$ .

By Lemma G.9, for every  $\epsilon > 0$  and every coverage function  $g$ , there exists a polynomially sized (polynomial in  $n$  and  $1/\epsilon$ ) representable function  $h$  such that  $\forall S \subseteq [n], (1 - \epsilon) \cdot g(S) \leq h(S) \leq g(S)$ . Hence, for all  $\epsilon > 0$  and for all matroid rank functions  $r$ , there exists a polynomial sized representable function  $h$  such that  $\forall S \subseteq [n], r(S) \leq h(S)/(1 - \epsilon) \leq \frac{\alpha}{1 - \epsilon} \cdot r(S)$ . For any given  $\epsilon > 0$ , there are only  $2^{O(n, 1/\epsilon)}$  many different  $h$  functions.

From Lemma G.10, for  $k = 2^{\log^2(n)}$ , there exists family of sets  $\mathcal{A} \subseteq 2^{[n]}$  with  $|\mathcal{A}| = k$ , and a  $2^k$  sized matroid family  $\mathcal{M}_B$  such that for all sets  $A \in \mathcal{A}$  and  $\forall B \subseteq \mathcal{A}$ ,

$$\forall S \in \mathcal{A}, r_{M_B}(S) = \begin{cases} 8 \log^2 n & \text{if } S \in B \\ n^{1/3} & \text{if } S \notin B \end{cases} \quad (16)$$

Now while the number of different  $g'$  functions are  $2^{O(n, 1/\epsilon)}$ , the number of different matroid rank functions in this family is  $2^{n^{\log(n)}}$ . Hence, by pigeon-hole principle, there must be two matroids  $B$  and  $B'$  ( $B \neq B'$ ) such that the best coverage functions  $g$  and  $g'$  approximating  $B$  and  $B'$  respectively, have the same best polysized representation  $h$ . But since, for every set  $S \in B \Delta B'$ ,  $r_{M_B}$  and  $r_{M_{B'}}$ , differ by a factor of  $\Omega(n^{1/3}/\log^2(n))$ , therefore,  $h$  cannot approximate at least one of these two to a factor better  $\Omega(n^{1/3}/\log^2(n))$ . Since the value of  $g$  and  $g'$  at any point in the domain is off from that of  $h$  by at most  $1 - \epsilon$ , and hence it follows that  $\alpha = \Omega(n^{1/3}/\log^2(n))$ . ■

## G Approximating Monotone Submodular Functions by Coverage Functions and by Budgeted Additive Functions

The two main results of the section are the following.

**Theorem G.1.** *Coverage functions can approximate every non-negative monotone submodular function to within a factor  $O(\sqrt{n} \log n)$ . Additionally, the class of coverage functions cannot approximate every non-negative monotone submodular function to a factor within  $o\left(\frac{n^{1/3}}{\log^2 n}\right)$ .*

**Theorem G.2.** *The class of sum of budgeted additive functions can approximate every non-negative monotone submodular function to a factor within  $(\sqrt{n} \log n)$ . Additionally, the class of sum of budgeted additive functions cannot approximate every non-negative monotone submodular function to a factor within  $o\left(\frac{n^{1/3}}{\log^2 n}\right)$ .*

### G.1 Upper Bound

For the upper-bound, we show that budgeted additive functions can  $\sqrt{n} \log(n)$ -approximate the class of non-negative, monotone, submodular functions. Then we use Theorem 1.3 to infer that the coverage functions too can give approximately the same guarantee.

**Lemma G.3.** *The class of sum of budgeted additive functions can approximate every non-negative monotone submodular function to within a factor  $(\sqrt{n} \log n)$ .*

The following corollary follows easily from Lemma G.3 and Theorem 1.3

**Corollary G.4.** *The class of sum of coverage functions can approximate every non-negative monotone submodular function to factor  $O(\sqrt{n} \log n)$ .*

The proof of Lemma G.3 follows from Lemmas G.5 and G.6. Lemma G.5 [13] gives a particular function that  $\sqrt{n} \log(n)$ -approximates a general monotone, non-negative, sub-modular function, and Lemma G.6 implies that this approximating function can be written as a sum of budgeted additive functions.

**Lemma G.5.** [13] *For every monotone submodular function  $f : 2^U \rightarrow R_+$ , there exists positive reals  $a_e$  for each  $e \in E$  such that  $g : 2^U \rightarrow R_+$  defined as  $g(S) = \sqrt{\sum_{e \in S} a_e}$ , approximates  $f$  within factor  $\sqrt{n} \log(n)$ .*

**Lemma G.6.** *Every submodular function  $f : 2^{[n]} \rightarrow \mathbb{Z}^+$  of the form  $f(S) = g(\sum_{i \in S} a_i)$ , where  $a_i \in \mathbb{Z}^+$  and  $g$  is a non-negative, monotone, concave and integer valued on integral inputs, can be written as a sum of budgeted additive functions.*

*Proof.* Let  $m = \sum_{i=1}^n a_i$ . Consider the function  $h$ , over the domain  $[m]$ , defined as  $h(S) = g(|S|)$  for all  $S \subseteq [m]$ . Since  $g(\cdot)$  is a non-negative, monotone, concave function, it is easy to verify that  $h(\cdot)$  is a non-negative, monotone, submodular function. Construct  $n$  mutually disjoint sets  $A_i \subseteq [m]$  such that  $|A_i| = a_i$ . Clearly,  $\forall S \subseteq [n], f(S) = h(\cup_{i \in S} A_i)$ .

From Lemma D.1, we know that  $h(\cdot)$  can be expressed as  $\sum_{i=1}^m \alpha_i \cdot t_i(\cdot)$  where each  $t_i$  is a uniform matroid rank function with rank  $i$ , over the domain  $[m]$  and each  $\alpha_i \geq 0$ .

This implies that for all  $S \subseteq [n], f(S) = \sum_{i=1}^m \alpha_i \cdot t_i(\cup_{i \in S} A_i)$ . Now, for every  $i \in [m]$ , construct the budget additive function  $t'_i$ , defined as  $\forall S \subseteq [n], t'_i(S) = \min\{\sum_{i \in S} v_{ij}, B_i\}$ , where for  $j \in [n]$ ,

the value  $v_{ij} = a_j$  and the budget  $B_i$  is  $i$ . Since  $t_i$  is a uniform matroid rank function of rank  $i$  and  $A_i$ 's are mutually disjoint, we have for all  $i \in [m]$ ,

$$\forall S \subseteq [n], t'_i(S) = t_i(\cup_{i \in S} A_i) \quad (17)$$

Therefore, we get for all sets  $S \subseteq [n]$ ,  $f(S) = \sum_{i=1}^m t'_i(S)$ . ■

## G.2 Lower Bound

For the lower bound, we first show that sum of coverage functions cannot approximate the class of monotone, submodular functions well, and then use Theorem 1.3, to infer that, therefore, even the class of sum of budgeted additive functions cannot approximate a monotone submodular function well.

**Lemma G.7.** *The class of sum of coverage functions cannot approximate every non-negative monotone submodular function to a factor within  $o\left(\frac{n^{1/3}}{\log^2 n}\right)$ .*

An easy corollary of Lemma G.7 that follows from Theorem 1.3 is the following.

**Corollary G.8.** *The class of sum of budgeted additive functions cannot approximate every non-negative monotone submodular function to a factor within  $o\left(\frac{n^{1/3}}{\log^2 n}\right)$ .*

We now present the proof of Lemma G.7. We will need to use results from [1] and [2], for which we first present a definition.

**Definition 6.** *A  $\beta$ -sketch of a function  $f : 2^U \rightarrow \mathbb{R}$  is a polynomially sized (in  $|U|$  and  $1/(1-\beta)$ ) representable function  $g$  such that  $\forall S \subseteq U$ ,  $\beta \cdot f(S) \leq g(S) \leq f(S)$ .*

The following result is from [1].

**Lemma G.9.** *[1] Coverage functions allow from arbitrary well sketches i.e., for any  $\epsilon > 0$ , there exists a  $1 - \epsilon$  sketch.*

The following result is from [2]. It gives a ‘large’ family of matroid rank functions, such that any two functions in the class have at least one point where the values that they take differ by a ‘significant’ factor.

**Lemma G.10.** *[2] For any  $k = 2^{o(n^{1/3})}$ , there exists a family of sets  $\mathcal{A} \subseteq 2^{[n]}$  with  $|\mathcal{A}| = k$  and a family of matroids  $\mathcal{M} = \{M_B | B \subseteq \mathcal{A}\}$  such that for all  $B \subseteq \mathcal{A}$ , it is the case that*

$$\forall S \in \mathcal{A}, r_{M_B}(S) = \begin{cases} 8 \log k & \text{if } S \in B \\ n^{1/3} & \text{if } S \notin B \end{cases} \quad (18)$$

where  $r_{M_B}$  is the rank function of the matroid  $M_B$

of Lemma G.7. Let the class of matroid rank functions on the domain of size  $n$  be  $\alpha$ -approximable by coverage functions, for some  $\alpha$ . That is, for a domain  $[n]$ , for all matroid rank function  $r$ , there exists a coverage function  $g$  such that  $\forall S \subseteq [n]$ ,  $r(S) \leq g(S) \leq \alpha \cdot r(S)$ .

By Lemma G.9, for every  $\epsilon > 0$  and every coverage function  $g$ , there exists a polynomially sized (polynomial in  $n$  and  $1/\epsilon$ ) representable function  $h$  such that  $\forall S \subseteq [n]$ ,  $(1 - \epsilon) \cdot g(S) \leq h(S) \leq g(S)$ .

Hence, for all  $\epsilon > 0$  and for all matroid rank functions  $r$ , there exists a polynomial sized representable function  $h$  such that  $\forall S \subseteq [n], r(S) \leq h(S)/(1 - \epsilon) \leq \frac{\alpha}{1 - \epsilon} \cdot r(S)$ . For any given  $\epsilon > 0$ , there are only  $2^{O(n, 1/\epsilon)}$  many different  $h$  functions.

From Lemma G.10, for  $k = 2^{\log^2(n)}$ , there exists family of sets  $\mathcal{A} \subseteq 2^{[n]}$  with  $|\mathcal{A}| = k$ , and a  $2^k$  sized matroid family  $\mathcal{M}_B$  such that for all sets  $A \in \mathcal{A}$  and  $\forall B \subseteq \mathcal{A}$ ,

$$\forall S \in \mathcal{A}, r_{M_B}(S) = \begin{cases} 8 \log^2 n & \text{if } S \in B \\ n^{1/3} & \text{if } S \notin B \end{cases} \quad (19)$$

Now while the number of different  $g'$  functions are  $2^{O(n, 1/\epsilon)}$ , the number of different matroid rank functions in this family is  $2^{n^{\log(n)}}$ . Hence, by pigeon-hole principle, there must be two matroids  $B$  and  $B'$  ( $B \neq B'$ ) such that the best coverage functions  $g$  and  $g'$  approximating  $B$  and  $B'$  respectively, have the same best polysized representation  $h$ . But since, for every set  $S \in B \Delta B'$ ,  $r_{M_B}$  and  $r_{M_{B'}}$ , differ by a factor of  $\Omega(n^{1/3}/\log^2(n))$ , therefore,  $h$  cannot approximate at least one of these two to a factor better  $\Omega(n^{1/3}/\log^2(n))$ . Since the value of  $g$  and  $g'$  at any point in the domain is off from that of  $h$  by at most  $1 - \epsilon$ , and hence it follows that  $\alpha = \Omega(n^{1/3}/\log^2(n))$ . ■