Multivariate Submodular Optimization

Richard Santiago [* 1] **F. Bruce Shepherd** [* 2]

Abstract

Submodular functions have found a wealth of new applications in data science and machine learning models in recent years. This has been coupled with many algorithmic advances in the area of submodular optimization: (SO) $\min / \max \, f(S) : S \in \mathcal{F}$, where $\mathcal{F}$ is a given family of feasible sets over a ground set V and $f : 2^V \to \mathbb{R}$ is submodular. Our focus is on a more general class of *multivariate submodular optimization* (MVSO) problems: $\min / \max \, f(S_1, S_2, \ldots, S_k) : S_1 \uplus S_2 \uplus \cdots \uplus S_k \in \mathcal{F}$. Here we use $\uplus$ to denote union of disjoint sets and hence this model is attractive where resources are being allocated across k agents, who share a "joint" multivariate nonnegative objective $f(S_1, S_2, \ldots, S_k)$ that captures some type of submodularity (i.e. diminishing returns) property. We provide some explicit examples and potential applications for this new framework. For maximization, we show that practical algorithms such as accelerated greedy variants and distributed algorithms achieve good approximation guarantees for very general families (such as matroids and p-systems). For arbitrary families, we show that monotone (resp. nonmonotone) MVSO admits an $\alpha(1 - 1/e)$ (resp. $\alpha \cdot 0.385$) approximation whenever monotone (resp. nonmonotone) SO admits an α-approximation over the multilinear formulation. This substantially expands the family of tractable models. On the minimization side we give essentially optimal approximations in terms of the curvature of f.

1. Introduction

Submodularity is a property of set functions with deep theoretical consequences and a wide range of applications.

---

*Equal contribution ¹School of Computer Science, McGill University, Montreal, Canada ²Department of Computer Science, University of British Columbia, Vancouver, Canada. Correspondence to: Richard Santiago <richard.santiagotorres@mail.mcgill.ca>.

Optimizing submodular functions is a central subject in operations research and combinatorial optimization (Lovász, 1983). It appears in many important optimization frameworks including cuts in graphs, set covering problems, plant location problems, certain satisfiability problems, combinatorial auctions, and maximum entropy sampling. In machine learning it has recently been identified and utilized in domains such as viral marketing (Kempe et al., 2003), information gathering (Krause & Guestrin, 2007), image segmentation (Boykov & Jolly, 2001; Kohli et al., 2009; Jegelka & Bilmes, 2011), document summarization (Lin & Bilmes, 2011), and speeding up satisfiability solvers (Streeter & Golovin, 2009).

A set function $f : 2^V \to \mathbb{R}$ is *submodular* if $f(S) + f(T) \geq f(S \cup T) + f(S \cap T)$ for any $S, T \subseteq V$. We call f *monotone* if $f(S) \leq f(T)$ for $S \subseteq T$. Throughout, all functions are nonnegative, and we usually assume $f(\emptyset) = 0$. Our functions are given by a *value oracle*, where for a given set S an algorithm can query the oracle to find its value $f(S)$.

We consider the following broad class of submodular optimization (SO) problems:

$$\text{SO}(\mathcal{F}) \quad \text{Min / Max } f(S) : S \in \mathcal{F}$$

where f is a nonnegative submodular set function on a finite ground set V, and $\mathcal{F} \subseteq 2^V$ is a family of feasible sets. These problems have been well studied for a variety of set families $\mathcal{F}$. We explore the connections between these (single-agent) problems and their more general multivariate incarnations. In the *multivariate (MV)* version, we have k agents and a "joint" multivariate nonnegative objective $f(S_1, S_2, \ldots, S_k)$ that captures some type of submodularity (i.e. diminishing returns) property (see Section 1.1). As before, we are looking for sets $S \in \mathcal{F}$, however, we now have a 2-phase task: the elements of S must also be partitioned amongst the agents. Hence we have set variables S_i and seek to optimize $f(S_1, S_2, \ldots, S_k)$. This leads to the multivariate submodular optimization (MVSO) versions:

$$\begin{aligned} \text{MVSO}(\mathcal{F}) \quad &\text{Min / Max } f(S_1, S_2, \ldots, S_k) : \\ &S_1 \uplus S_2 \uplus \cdots \uplus S_k \in \mathcal{F}. \end{aligned}$$

Our main objective is to study the approximability of the multivariate problems in terms of their single-agent versions. We refer to the *multivariate (MV) gap* as the approximation

factor loss incurred by moving to the multivariate setting. To the best of our knowledge, neither the MVSO($\mathcal{F}$) framework for general families $\mathcal{F}$ nor the notion of MV gap have been considered before.

An important special case of MVSO occurs when the function $f(S_1, \ldots, S_k)$ can be *separated* as $f(S_1, \ldots, S_k) = \sum_{i=1}^{k} f_i(S_i)$ where the f_i are all submodular; in this case we say that f is *separable*. This leads to the class of multi-agent submodular optimization (MASO) problems (see related work section)

$$\text{MASO}(\mathcal{F}) \quad \text{Min / Max } \sum_{i=1}^{k} f_i(S_i) :$$
$$S_1 \uplus S_2 \uplus \cdots \uplus S_k \in \mathcal{F}.$$

1.1. Multivariate Submodular Optimization

We consider functions of several variables which satisfy the following type of submodularity property. A multivariate function $f : 2^{kV} \to \mathbb{R}$ is *k-multi-submodular* if for all pairs of tuples $(S_1, S_2, ..., S_k), (T_1, T_2, ..., T_k) \in 2^{kV}$ we have

$$f(S_1, ..., S_k) + f(T_1, ..., T_k) \geq f(S_1 \cup T_1, S_2 \cup T_2, .., S_k \cup T_k)$$
$$+ f(S_1 \cap T_1, S_2 \cap T_2, ..., S_k \cap T_k).$$

We call f *normalized* if $f(\emptyset, \emptyset, \ldots, \emptyset) = 0$, and *monotone* if $f(S_1, \ldots, S_k) \leq f(T_1, \ldots, T_k)$ for all tuples $(S_1, \ldots, S_k)$ and $(T_1, \ldots, T_k)$ satisfying $S_i \subseteq T_i$ for all $i \in [k]$.

For $k = 1$ a k-multi-submodular function is just a submodular function. In Appendix A of the paper full version we discuss how k-multi-submodular functions can also be characterized (or defined) in terms of diminishing returns.

Two explicit examples of (non-separable) k-multi-submodular functions (see Appendix B in the full version for proofs) are the following.

Example 1. *Consider a multilinear function $h : \mathbb{Z}_+^k \to \mathbb{R}$ given by $h(z) = \sum_{S \subseteq [k]} a_S \prod_{m \in S} z_m$. Let $f : 2^{kV} \to \mathbb{R}$ be a multivariate set function defined as $f(S_1, \ldots, S_k) = h(|S_1|, \ldots, |S_k|)$. Then f is k-multi-submodular if and only if $a_S \leq 0$ for all $S \subseteq [k]$.*

Example 2. *Let $h : \mathbb{Z}_+^k \to \mathbb{R}$ be a quadratic function given by $h(z) = z^T A z$. Let $f : 2^{kV} \to \mathbb{R}$ be a multivariate set function defined as $f(S_1, \ldots, S_k) = h(|S_1|, \ldots, |S_k|)$. Then f is k-multi-submodular if and only if $A = (a_{ij})$ satisfies $a_{ij} + a_{ji} \leq 0$ for all $i, j \in [k]$.*

We believe the above examples are useful for modelling "competition" between agents in many domains. In Section 1.3 we discuss an application to sensor placement problems.

1.2. Our Contributions

Our first contribution is to show that the MV framework can model much more general problems than the separable

multi-agent (i.e. MASO) framework. This is quantitatively captured in the next information theoretic result (see Section 3.1) where we establish a large gap between the two problems:

$$(\mathbf{MV - Min}) \quad \begin{array}{ll} \min & f(S_1, \ldots, S_k) \\ \text{s.t.} & S_1 \uplus \cdots \uplus S_k = V \end{array}$$

$$(\mathbf{MA - Min}) \quad \begin{array}{ll} \min & \sum_{i=1}^{k} f_i(S_i) \\ \text{s.t.} & S_1 \uplus \cdots \uplus S_k = V \end{array}$$

Theorem 3. *The MV-Min problem with a nonnegative monotone k-multi-submodular objective function cannot be approximated to a ratio $o(n/\log n)$ in the value oracle model with polynomial number of queries, whereas its separable version MA-Min has a tight $O(\log n)$-approximation polytime algorithm for nonnegative monotone submodular functions f_i.*

The above result shows that the MV model may also potentially face roadblocks in terms of tractability. Fortunately, we can show that the multivariate problem remains very well-behaved in the maximization setting. Our main result establishes that if the single-agent problem for a family $\mathcal{F}$ admits approximation via its multilinear relaxation (see Section 2.2), then we may extend this to its multivariate version with a constant factor loss.

Theorem 4. *If there is a (polytime) $\alpha(n)$-approximation for monotone SO($\mathcal{F}$) maximization via its multilinear relaxation, then there is a (polytime) $(1 - 1/e) \cdot \alpha(n)$-approximation for monotone MVSO($\mathcal{F}$) maximization. Furthermore, given a downwards closed family $\mathcal{F}$, if there is a (polytime) $\alpha(n)$-approximation for nonmonotone SO($\mathcal{F}$) maximization via its multilinear relaxation, then there is a (polytime) $0.385 \cdot \alpha(n)$-approximation for nonmonotone MVSO($\mathcal{F}$) maximization.*

We note that the multilinear relaxation can be efficiently evaluated for a large class of practical and useful submodular functions (Iyer et al., 2014), thus making these algorithms viable for many real-world machine learning problems.

We remark that the MV gap of $1 - 1/e$ for monotone objectives is tight, in the sense that there are families where this cannot be improved. For instance, $\mathcal{F} = \{V\}$ has a trivial 1-approximation for the single-agent problem, and a $1 - 1/e$ inapproximability factor for the separable multi-agent (i.e. MASO) version (Khot et al., 2005; Mirrokni et al., 2008), and hence also for the more general MVSO problem.

An immediate application of Theorem 4 is that it provides the first constant (and in fact optimal) $(1 - 1/e)$-approximation for the monotone *generalized submodular welfare* problem $\max f(S_1, S_2, \ldots, S_k) : S_1 \uplus \cdots \uplus S_k = V$. This problem generalizes the well-studied submodular welfare problem (Lehmann et al., 2006; Vondrák, 2008; Korula et al., 2018), which captures several allocation problems

and has important applications in combinatorial auctions, Internet advertising, and network routing. The MV objectives can capture much more general interactions among the agents/bidders, where now a bidder's valuation does not only depend on the set S of items that she gets, but also on the items that her strategic partners and competitors get. For instance, in a bandwidth spectrum auction, this could capture a company's interest to maximize compatibility and prevent cross-border interference.

In Section 2 we describe a simple reduction that shows that for some families [1] an (optimal) MV gap of 1 holds. We also discuss how for those families, practical algorithms (such as accelerated greedy variants and distributed algorithms) can be used and lead to good approximation guarantees.

Theorem 5. *Let $\mathcal{F}$ be a matroid, a p-matroid intersection, or a p-system. Then, if there is a (polytime) α-approximation algorithm for monotone (resp. nonmonotone) SO($\mathcal{F}$) maximization, there is a (polytime) α-approximation algorithm for monotone (resp. nonmonotone) MVSO($\mathcal{F}$) maximization.*

On the minimization side our approximation results and MV gaps are larger. This is somewhat expected due to the strong hardness results already existing for single-agent submodular minimization (see Section 1.4). However, we give essentially tight approximations in terms of the objective's curvature. The notion of curvature has been widely used for univariate functions (Conforti & Cornuéjols, 1984; Vondrák, 2010; Iyer et al., 2013; Bai & Bilmes, 2018), since it allows for better approximations and it is linear time computable.

Given a tuple $(S_1, \ldots, S_k) \in 2^{kV}$ and $(i, v) \in [k] \times V$, we denote by $(S_1, \ldots, S_k) + (i, v)$ the new tuple $(S_1, \ldots, S_{i-1}, S_i + v, S_{i+1}, \ldots, S_k)$. Then, it is natural to think of the quantity

$$f_{(S_1,\ldots,S_k)}((i,v)) := f((S_1,\ldots,S_k)+(i,v))-f(S_1,\ldots,S_k)$$

as the marginal gain of assigning element v to agent i in the tuple $(S_1, \ldots, S_k)$. We also use $f((i,v))$ to denote the quantity $f(\emptyset, \ldots, \emptyset, v, \emptyset, \ldots, \emptyset)$ where v appears in the ith component. Then given a normalized monotone k-multi-submodular function $f : 2^{kV} \to \mathbb{R}$ we define its *total curvature* c and its *curvature* $c(S_1, \ldots, S_k)$ *with respect to a tuple* $(S_1, \ldots, S_k) \subseteq V^k$ as

$$c = 1 - \min_{i \in [k], v \in V} \frac{f_{(V,V,\ldots,V)-(i,v)}((i,v))}{f((i,v))} \qquad (1)$$

$$c(S_1, \ldots, S_k) = 1 - \min_{i \in [k], v \in S_i} \frac{f_{(S_1,\ldots,S_k)-(i,v)}((i,v))}{f((i,v))}.$$

---

[1] A family of sets $\mathcal{F}$ is a *p-system* if for all $S \in \mathcal{F}$ and $v \in V$ there exists a set $T \subseteq S$ such that $|T| \leq p$ and $S \setminus T \cup \{v\} \in \mathcal{F}$. A *matroid* is a 1-system. Cardinality and partition constraints are examples of matroids. We refer the reader to Schrijver (2003); Calinescu et al. (2007; 2011) for a comprehensive discussion.

We prove the following result for k-multi-submodular objectives. We note the gap is stronger in the sense that it is relative to the single-agent *modular* problem. [2] See Section 3.1 of the full version for proof details.

Theorem 6. *Let f be a monotone k-multi-submodular function, and let $\mathcal{F}$ be a family that admits a (polytime) β-approximation over modular functions. Denote by $(S_1^*, \ldots, S_k^*)$ an optimal solution to monotone MVSO($\mathcal{F}$) minimization, and by $c(S_1^*, \ldots, S_k^*)$ the curvature of f with respect to $(S_1^*, \ldots, S_k^*)$. Then there is a (polytime) $\frac{\beta \sum_{i \in [k]} |S_i^*|}{1+(\sum_{i \in [k]} |S_i^*|-1)(1-c(S_1^*,\ldots,S_k^*))}$-approximation algorithm for monotone MVSO($\mathcal{F}$) minimization.*

In some situations the above result may lead to approximation factors highly preferable to those obtained for general functions (i.e. objectives with curvature 1). Examples of these include families like $\mathcal{F} = \{V\}$, spanning trees, or perfect matchings, where exact algorithms are available for modular objectives (i.e. $\beta = 1$) and any optimal solution $(S_1^*, \ldots, S_k^*)$ satisfies $\sum_{i \in [k]} |S_i^*| = \Omega(n)$. Thus, we go from polynomial approximation factors (for objectives with curvature 1) to constant or logarithmic factors (for constant or order $1 - \frac{1}{\log n}$ curvature).

Moreover, having the curvature $c(S_1^*, \ldots, S_k^*)$ can be much more beneficial than having the total curvature c. For instance, for the problem $\min f(S_1, \ldots, S_k) : S_1 \uplus \cdots \uplus S_k = V$ with $f(S_1, \ldots, S_k) = \min\{n, \sum_{i=1}^{k} |S_i|\}$. Here the total curvature of f is 1 (hence leading to an n-approximation in Theorem 6), while the curvature $c(S_1^*, \ldots, S_k^*)$ with respect to any partition $(S_1^*, \ldots, S_k^*)$ is 0 (and thus leading to an exact approximation via Theorem 6).

In Section 3.1 we give evidence that Theorem 6 is essentially tight, even for $\mathcal{F} = \{V\}$ where we show the following curvature dependent information-theoretic lower bound.

Theorem 7. *The monotone MVSO($\mathcal{F}$) minimization problem over $\mathcal{F} = \{V\}$ and objectives f with total curvature c cannot be approximated to a ratio $o(\frac{n/\log n}{1+(\frac{n}{\log n}-1)(1-c)})$ in the value oracle model with polynomial number of queries.*

Finally, we give an approximation in terms of the number of agents k, which may be preferable in settings where k is not too large. See Section 3.2 of the full version for details.

Theorem 8. *If there is a (polytime) $\alpha(n)$-approximation for monotone SO($\mathcal{F}$) minimization based on rounding the convex relaxation, then there is a (polytime) $k\alpha(n)$-approximation for monotone MVSO($\mathcal{F}$) minimization.*

---

[2] A set function $f : 2^V \to \mathbb{R}$ is modular if $f(A) + f(B) = f(A \cup B) + f(A \cap B)$ for all $A, B \subseteq V$. Modular functions can always be expressed in the form $f(S) = w(S) := \sum_{v \in S} w(v)$ for some weight function $w : V \to \mathbb{R}$.

1.3. The Multivariate Model and Applications

Our second objective is to extend the multivariate model and show that in some cases this larger class remains tractable. Specifically, we define the *capacitated multivariate submodular optimization (CMVSO) problem* as follows:

$$\text{CMVSO}(\mathcal{F}) \quad \begin{array}{cl} \max / \min & f(S_1, S_2, \ldots, S_k) \\ \text{s.t.} & S_1 \uplus \cdots \uplus S_k \in \mathcal{F} \\ & S_i \in \mathcal{F}_i, \ \forall i \in [k] \end{array}$$

where we are supplied with subfamilies $\mathcal{F}_i$.

Our results imply that one maintains good approximations even while adding interesting side constraints. For example, for a monotone maximization instance of CMVSO where $\mathcal{F}$ is a p-matroid intersection and the $\mathcal{F}_i$ are all matroids, our results from Section 2 lead to a $(\frac{1}{p+1} - \epsilon)$-approximation algorithm via the multilinear relaxation, or a $1/(p+2)$-approximation via a simple greedy algorithm. We believe that these, combined with other results from Section 2, substantially expand the family of tractable models (both in theory and practice) for maximization.

Many existing applications fit into the CMVSO framework and some of these can be enriched through the added flexibility of the capacitated model. For instance, one may include set bounds on the variables: $L_i \subseteq S_i \subseteq U_i$ for each i, or simple cardinality constraints: $|S_i| \leq b_i$ for each i. A well-studied (Fleischer et al., 2006; Goundan & Schulz, 2007; Calinescu et al., 2011) application of CMVSO in the maximization setting is the Separable Assignment Problem (SAP), which corresponds to the setting where the objective is separable and modular, the $\mathcal{F}_i$ are downward closed (i.e. hereditary) families, and $\mathcal{F} = 2^V$. The following example illustrates CMVSO's potential as a general model.

Example 9 (Sensor Placement with Multivariate Objectives). *Sensor placement and information gathering problems have been popular in the submodularity literature (Krause & Guestrin, 2007; Krause et al., 1973; 2008). Given a set of sensors V and a set of possible locations $\{1, 2, \ldots, k\}$, the goal is to place sensors at some of the locations so as to maximize the "informativeness" gathered. There is usually also a budget constraint restricting the total number of sensors that can be deployed. This application is well suited to a k-multi-submodular objective function $f(S_1, ..., S_k)$ which measures the "informativeness" of placing sensors S_i at location i. A natural mathematical formulation for this is given by*

$$\begin{array}{cl} \max & f(S_1, S_2, ..., S_k) \\ s.t. & S_1 \uplus S_2 \uplus \cdots \uplus S_k \in \mathcal{F} \\ & S_i \in \mathcal{F}_i, \end{array}$$

where $\mathcal{F} := \{S \subseteq V : |S| \leq b\}$ imposes the budget constraint and $\mathcal{F}_i$ gives additional modelling flexibility. For instance, we could impose $\mathcal{F}_i = \{S \subseteq V_i : |S| \leq b_i\}$ to constrain the types and number of sensors that can be placed at location i. Notice that in these cases both $\mathcal{F}$ and the $\mathcal{F}_i$ are matroids and hence the algorithms from Section 2.4 apply. One may form a multivariate objective by defining $f(S_1, S_2, \ldots, S_n) = \sum_i f_i(S_i) - R(S_1, S_2, \ldots, S_n)$ where the f_i's measure the benefit of placing sensors S_i at location i, and $R()$ is a redundancy function. If the f_i's are submodular and $R()$ is k-multi-supermodular (i.e. $-R()$ is k-multi-submodular), then f is k-multi-submodular. In this setting, it is natural to take the f_i's to be coverage functions, where $f_i(S_i)$ measures the coverage of placing sensors S_i at location i. We next propose a family of "redundancy" functions which are k-multi-supermodular.

SUPERMODULAR PENALTY MEASURES VIA QUADRATIC FUNCTIONS. *We denote $\mathbf{S} := (S_1, S_2, \ldots, S_n)$ and $z_{\mathbf{S}} := (|S_1|, |S_2|, \ldots, |S_n|)$. One can show (see Lemma 28 in Appendix B of full version) that if A is a matrix satisfying $a_{ij} + a_{ji} \geq 0$, then $R(\mathbf{S}) := z_{\mathbf{S}}^T A z_{\mathbf{S}}$ is k-multi-supermodular. Then for this particular example one could for instance take redundancy coefficients a_{ij} as $\Theta(\frac{1}{d(i,j)^2})$ where $d(i, j)$ denotes the distance between locations i and j. This can be further extended so that different sensor types contribute different weights to the vector $z_{\mathbf{S}}$, e.g., define $z_{\mathbf{S}}(i) = \sum_{j \in S_i} w(j)$ for an associated sensor weight vector w.*

1.4. Related Work

Submodularity naturally arises in many machine learning applications such as viral marketing (Kempe et al., 2003), information gathering (Krause & Guestrin, 2007), image segmentation (Boykov & Jolly, 2001; Kohli et al., 2009; Jegelka & Bilmes, 2011), document summarization (Lin & Bilmes, 2011), news article recommendation (El-Arini et al., 2009), active learning (Golovin & Krause, 2011), and speeding up SAT solvers (Streeter & Golovin, 2009).

Single Agent Optimization. Minimizing a submodular function can be solved in polytime (Grötschel et al., 2012; Schrijver, 2000; Iwata et al., 2001). Unconstrained maximization, on the other hand, is known to be inapproximable for general submodular functions but admits a polytime constant-factor approximation algorithm when f is nonnegative (Buchbinder et al., 2015; Feige et al., 2011).

For constrained maximization, the classical work Nemhauser et al. (1978); Nemhauser & Wolsey (1978); Fisher et al. (1978) established an optimal $(1 - 1/e)$-approximation for nonnegative monotone maximization under a cardinality constraint, and a $(1/(k + 1))$-approximation under k matroid constraints. The latter is almost tight since there is an $\Omega(\log(k)/k)$ inapproximability result (Hazan et al., 2006). For nonnegative monotone functions, Vondrák (2008); Calinescu et al.

(2011) give an optimal $(1 - 1/e)$-approximation based on the multilinear extension when $\mathcal{F}$ is a matroid; and Lee et al. (2010) gives a local-search algorithm that achieves a $(1/k - \epsilon)$-approximation (for any fixed $\epsilon > 0$) when $\mathcal{F}$ is a k-matroid intersection. For nonnegative nonmonotone functions, a 0.385-approximation (Buchbinder & Feldman, 2016) is the best factor known for a matroid constraint. In Lee et al. (2009) a $1/(k + O(1))$-approximation is given for k matroid constraints with k fixed, and Gupta et al. (2010) gives a simple "multi-greedy" algorithm that matches the approximation of Lee et al. but is polytime for any k.

For constrained minimization the news is worse (Goel et al., 2009; Svitkina & Fleischer, 2011; Iwata & Nagano, 2009). If $\mathcal{F}$ consists of spanning trees Goel et al. (2009) show a lower bound of $\Omega(n)$, while if $\mathcal{F}$ corresponds to the cardinality constraint $\{S : |S| \geq k\}$ Svitkina & Fleischer (2011) show a lower bound of $\tilde{\Omega}(\sqrt{n})$. There are a few exceptions. The problem can be solved exactly when $\mathcal{F}$ is a ring family (Schrijver, 2000), triple family (Grötschel et al., 2012), or parity family (Goemans & Ramakrishnan, 1995). In the context of NP-Hard problems, there is a 2-approximation (Goel et al., 2009; Iwata & Nagano, 2009) for submodular vertex cover, and an $O(k)$-approximation for k-uniform hitting set.

Multivariate Problems. The notion of k-multi-submodularity already appeared (under the name of multidimensional submodularity) in the work of Fisher et al. (1978), where they consider the multivariate monotone maximization problem with $\mathcal{F} = \{V\}$ as a motivating example for submodular maximization subject to a matroid constraint. They show that for this problem a simple greedy algorithm achieves a $1/2$-approximation. The work of Singh et al. (2012) considers the special case of 2-multi-submodular functions (they call them *simple bisubmodular*). They give constant factor approximations for maximizing monotone 2-multi-submodular functions under cardinality and partition constraints, and provide applications to coupled sensor placement and coupled feature selection problems.

Other different extensions of submodular functions to multivariate settings have been studied. Some of these include bisubmodular functions (Qi, 1988; Ando et al., 1996; Fujishige & Iwata, 2005; Bouchet & Cunningham, 1995), k-submodular functions (Huber & Kolmogorov, 2012; Ward & Živný, 2014; Ohsaka & Yoshida, 2015), or skew bisubmodular functions (Huber et al., 2014; Thapper & Živný, 2016; Thapper & Zivny, 2012).

Finally, as mentioned in the introduction, an important class of (multi-agent submodular optimization) problems arises when $f(S_1, \ldots, S_k) = \sum_{i \in [k]} f_i(S_i)$. These problems have been widely studied when $\mathcal{F} = \{V\}$, both for minimization (Hayrapetyan et al., 2005; Svitkina & Tardos, 2010; Ene & Vondrák, 2014; Chekuri & Ene, 2011) and maximization (Fisher et al., 1978; Lehmann et al., 2006;

Vondrák, 2008), and have also been considered for general families (Goel et al., 2009; Santiago & Shepherd, 2018).

2. Multivariate Submodular Maximization

We describe two different reductions. The first one reduces the capacitated multivariate problem CMVSO to a single-agent SO problem, and it is based on the simple idea of taking k disjoint copies of the original ground set. We use this to establish an (optimal) MV gap of 1 for families such as spanning trees, matroids, and p-systems. The second reduction is based on the multilinear extension of a set function. We show that if the single-agent problem admits approximation via its multilinear relaxation (see Section 2.2), then we may extend this to its multivariate version with a constant factor loss, in the monotone and nonmonotone settings. For the monotone case the MV gap is tight.

2.1. The Lifting Reduction

We describe a generic reduction of CMVSO to a single-agent problem $\max / \min \bar{f}(S) : S \in \mathcal{L}$. The argument is based on the idea of viewing assignments of elements v to agents i in a *multi-agent bipartite graph*. This simple idea (which is equivalent to making k disjoint copies of the ground set) already appeared in the classical work of Fisher et al. (1978), and has since then been widely used (Lehmann et al., 2006; Vondrák, 2008; Calinescu et al., 2011; Singh et al., 2012; Santiago & Shepherd, 2018). We review it briefly here for completeness.

Consider the complete bipartite graph $G = ([k] + V, E)$. Every subset of edges $S \subseteq E$ can be written uniquely as $S = \uplus_{i \in [k]}(\{i\} \times S_i)$ for some sets $S_i \subseteq V$. This allows us to go from a multivariate objective (such as the one in CMVSO) to a univariate objective $\bar{f} : 2^E \to \mathbb{R}$ over the lifted space. Namely, for each set $S \subseteq E$ we define $\bar{f}(S) = f(S_1, S_2, \ldots, S_k)$. The function $\bar{f}$ is well-defined because of the one-to-one correspondence between sets $S \subseteq E$ and tuples $(S_1, \ldots, S_k) \subseteq V^k$.

We consider two families of sets over E that capture the original constraints:

$$\mathcal{F}' := \{S \subseteq E : S_1 \uplus \cdots \uplus S_k \in \mathcal{F}\}$$
$$\mathcal{H} := \{S \subseteq E : S_i \in \mathcal{F}_i, \forall i \in [k]\}.$$

We now have:

$$\begin{array}{lll} \max / \min & f(S_1, S_2, \ldots, S_k) & = \\ \text{s.t.} & S_1 \uplus \cdots \uplus S_k \in \mathcal{F} & \\ & S_i \in \mathcal{F}_i, \forall i \in [k] & \end{array}$$

$$\begin{array}{llll} \max / \min & \bar{f}(S) & = & \max / \min & \bar{f}(S) \\ \text{s.t.} & S \in \mathcal{F}' \cap \mathcal{H} & & \text{s.t.} & S \in \mathcal{L}. \end{array}$$

where in the last step we just let $\mathcal{L} := \mathcal{F}' \cap \mathcal{H}$.

Clearly, this reduction is interesting if our new function $\bar{f}$ and the family of sets $\mathcal{L}$ have properties which allow us to handle them computationally. This depends on the original structure of the function f, and the set families $\mathcal{F}$ and $\mathcal{F}_i$. The following is straightforward.

Claim 10. *If f is a (nonnegative, respectively monotone) k-multi-submodular function, then $\bar{f}$ as defined above is also (nonnegative, respectively monotone) submodular.*

In Section 2.4 we discuss several properties of the families $\mathcal{F}$ and $\mathcal{F}_i$ that are preserved under this reduction, as well as their algorithmic consequences.

2.2. Multilinear Extensions for MV Problems

For a set function $f : \{0,1\}^V \to \mathbb{R}$ we define its *multilinear extension* $f^M : [0,1]^V \to \mathbb{R}$ as

$$f^M(z) = \sum_{S \subseteq V} f(S) \prod_{v \in S} z_v \prod_{v \notin S} (1 - z_v).$$

This extension (introduced in Calinescu et al. (2007)) has several nice properties. One very useful is the following.

Proposition 11. *Let $f : 2^V \to \mathbb{R}$ be a submodular function and $f^M : [0,1]^n \to \mathbb{R}$ its multilinear extension. Then f^M is convex along directions $d = \mathbf{e_{v_i}} - \mathbf{e_{v_j}}$ for $i, j \in \{1, 2, \ldots, n\}$, where $\mathbf{e_v}$ denotes the characteristic vector of $\{v\}$.*

This now gives rise to natural relaxations. The *single-agent multilinear extension relaxation* is:

$$\text{(SA-ME)} \quad \max f^M(z) : z \in P^*(\mathcal{F}),$$

and the *multivariate multilinear extension relaxation* is:

$$\text{(MV-ME)} \quad \max \bar{f}^M(z_1, z_2, \ldots, z_k) : z_1 + z_2 + \cdots + z_k \in P^*(\mathcal{F}),$$

where $P^*(\mathcal{F})$ denotes some relaxation of the polytope $conv(\{\chi^S : S \in \mathcal{F}\})$ [3], and $\bar{f}$ the lifted univariate function from the reduction in Section 2.1. Note that $\bar{f}$ is defined over vectors $\bar{z} = (z_1, z_2, \ldots, z_k) \in [0,1]^E$, where we think of $z_i \in \mathbb{R}^n$ as the vector associated to agent i.

The relaxation SA-ME has been widely used (Calinescu et al., 2011; Lee et al., 2009; Feldman et al., 2011; Ene & Nguyen, 2016; Buchbinder & Feldman, 2016) for submodular maximization. The next result discusses its solvability. We note that MV-ME can be reduced to an instance of SA-ME, and thus be approximated to the same factor (see Section 2.2 of full version).

Theorem 12 (Buchbinder & Feldman (2016); Vondrák (2008)). *Let $f : 2^V \to \mathbb{R}_+$ be a nonnegative submodular function and $f^M : [0,1]^V \to \mathbb{R}_+$ its multilinear extension. Let $P \subseteq [0,1]^V$ be any downwards closed polytope that admits a polytime separation oracle, and denote*

$OPT = \max f^M(z) : z \in P$. *Then there is a polytime algorithm (Buchbinder & Feldman, 2016) that finds $z^* \in P$ such that $f^M(z^*) \geq 0.385 \cdot OPT$. Moreover, if f is monotone there is a polytime algorithm (Vondrák, 2008) that finds $z^* \in P$ such that $f^M(z^*) \geq (1 - 1/e)OPT$.*

2.3. A Tight $1 - 1/e$ MV Gap

In this section we prove Theorem 4. The main idea is that we start with an (approximate) optimal solution $z^* = z_1^* + z_2^* + \cdots + z_k^*$ to the MV-ME relaxation and build a new feasible solution $\hat{z} = \hat{z}_1 + \hat{z}_2 + \cdots + \hat{z}_k$ where the $\hat{z}_i$ have supports V_i that are pairwise disjoint. We think of V_i as the set of items associated (or pre-assigned) to agent i. 'Once we have such a pre-assignment we consider the single-agent problem $\max g(S) : S \in \mathcal{F}$ where

$$g(S) = f(S \cap V_1, S \cap V_2, \ldots, S \cap V_k). \tag{2}$$

It is clear that g is nonnegative monotone submodular since f is nonnegative monotone k-multi-submodular. Moreover, for any feasible solution $S \in \mathcal{F}$ for this single-agent problem, we obtain a multivariate solution of the same cost by setting $S_i = S \cap V_i$, since then $g(S) = f(S \cap V_1, S \cap V_2, \ldots, S \cap V_k) = f(S_1, S_2, \ldots, S_k)$.

For a set $S \subseteq V$ and a vector $z \in [0,1]^V$ we denote by $z|_S$ the truncation of z to elements of S. That is, we set $z|_S(v) = z(v)$ for each $v \in S$ and to zero otherwise. Then by definition of g we have that $g^M(z) = \bar{f}^M(z|_{V_1}, z|_{V_2}, \ldots, z|_{V_k})$, where $\bar{f}$ is the lifted function from Section 2.1. Moreover, if the sets V_i are pairwise disjoint, then $\bar{f}^M(z|_{V_1}, z|_{V_2}, \ldots, z|_{V_k}) = \bar{f}^M(z_1, z_2, \ldots, z_k)$. The next result formalizes this observation.

Proposition 13. *Let $z = \sum_{i \in [k]} z_i$ be a feasible solution to MV-ME such that the vectors z_i have pairwise disjoint supports V_i. Then $g^M(z) = \bar{f}^M(z_1, z_2, \ldots, z_k)$.*

We now have all the ingredients to prove our main result for maximization. We note that a gap of $1 - 1/e$ appeared in Santiago & Shepherd (2018) for the case of separable objectives $f(S_1, \ldots, S_k) = \sum_i f_i(S_i)$. That argument uses the component-wise linearity of the multilinear extension, while our proof for non-separable objectives strongly uses the convexity property from Proposition 11.

Theorem 4. *If there is a (polytime) $\alpha(n)$-approximation for monotone SO($\mathcal{F}$) maximization based on rounding SA-ME, then there is a (polytime) $(1 - 1/e) \cdot \alpha(n)$-approximation for monotone MVSO($\mathcal{F}$) maximization. Furthermore, given a downwards closed family $\mathcal{F}$, if there is a (polytime) $\alpha(n)$-approximation for nonmonotone SO($\mathcal{F}$) maximization based on rounding SA-ME, then there is a (polytime) $0.385 \cdot \alpha(n)$-approximation for nonmonotone MVSO($\mathcal{F}$) maximization.*

Proof. We discuss first the case of monotone objectives.

---

[3]$conv(X)$ denotes the convex hull of a set X of vectors, and χ^S denotes the characteristic vector of the set S.

STEP 1. Let $z^* = z_1^* + z_2^* + \cdots + z_k^*$ denote an approximate solution to MV-ME obtained via Theorem 12, and let OPT_{frac} be the value of an optimal solution. We then have that $f^M(z_1^*, z_2^*, \ldots, z_k^*) \geq (1-1/e)OPT_{frac} \geq (1-1/e)OPT_{MV}$.

STEP 2. For an element $v \in V$ let $\mathbf{e_v}$ denote the characteristic vector of $\{v\}$, i.e. the vector in $\mathbb{R}^V$ which has value 1 in the v-th component and zero elsewhere. Then by Proposition 11 we have that the function

$$h(t) = \bar{f}^M(z_1^*, z_2^*, \ldots, z_{i-1}^*, z_i^* + t\mathbf{e_v}, z_{i+1}^*, \ldots, z_{i'-1}^*,$$
$$z_{i'}^* - t\mathbf{e_v}, z_{i'+1}^*, \ldots, z_k^*)$$

is convex for any $v \in V$ and $i \neq i' \in [k]$. In particular, given any $v \in V$ such that there exist $i \neq i' \in [k]$ with $z_i^*(v), z_{i'}^*(v) > 0$, there is always a choice so that increasing one component and decreasing the other by the same amount does not decrease the objective value.

Let $v \in V$ be such that there exist $i \neq i' \in [k]$ with $z_i^*(v), z_{i'}^*(v) > 0$. Then, we either set $z_i^*(v) = z_i^*(v) + z_{i'}^*(v)$ and $z_{i'}^*(v) = 0$, or $z_{i'}^*(v) = z_i^*(v) + z_{i'}^*(v)$ and $z_i^*(v) = 0$, whichever does not decrease the objective value. We repeat until the vectors z_i^* have pairwise disjoint support. Let us denote these new vectors by $\hat{z}_i$ and let $\hat{z} = \sum_{i \in [k]} \hat{z}_i$. Then notice that the vector $z^* = \sum_{i \in [k]} z_i^*$ remains invariant after performing each of the above updates (i.e. $\hat{z} = z^*$), and hence the new vectors $\hat{z}_i$ remain a feasible solution.

STEP 3. In the last step we use the function g defined in (2), with sets V_i corresponding to the supports of the $\hat{z}_i$. Given our α-approximation rounding assumption for SA-ME, we can round $\hat{z}$ to find a set $\hat{S}$ such that $g(\hat{S}) \geq \alpha g^M(\hat{z})$. Then, by setting $\hat{S}_i = \hat{S} \cap V_i$ we obtain a multivariate solution satisfying

$$f(\hat{S}_1, \ldots, \hat{S}_k) = g(\hat{S}) \geq \alpha g^M(\hat{z}) = \alpha f^M(\hat{z}_1, \ldots, \hat{z}_k) \geq$$
$$\alpha f^M(z_1^*, \ldots, z_k^*) \geq \alpha(1-1/e)OPT_{MV},$$

where the second equality follows from Proposition 13. This completes the monotone proof.

For the nonmonotone case the argument is very similar. Here we restrict our attention to downwards closed families, since then we can get a 0.385-approximation at STEP 1 via Theorem 12. We then apply STEP 2 and 3 in the same fashion as we did for monotone objectives. This leads to a $0.385 \cdot \alpha(n)$-approximation for the multivariate problem. $\qquad\square$

2.4. Invariance Under the Lifting Reduction

In Section 2.3 we established a MV gap of $1-1/e$ for monotone objectives and of 0.385 for nonmonotone objectives and downwards closed families based on the multilinear formulations. In this section we describe several families

Table 1. Invariant properties under the lifting reduction (Santiago & Shepherd, 2018)

	Multivariate problem	Single-agent (i.e. reduced) problem
1	$(V, \mathcal{F})$ a p-system	$(E, \mathcal{F}')$ a p-system
2	$\mathcal{F}$ = bases of a p-system	$\mathcal{F}'$ = bases of a p-system
3	$(V, \mathcal{F})$ a matroid	$(E, \mathcal{F}')$ a matroid
4	$(V, \mathcal{F})$ a p-matroid intersection	$(E, \mathcal{F}')$ a p-matroid intersection
5	$(V, \mathcal{F}_i)$ a matroid for all $i \in [k]$	$(E, \mathcal{H})$ a matroid
6	$\mathcal{F}_i$ a ring family for all $i \in [k]$	$\mathcal{H}$ a ring family

with an (optimal) MV gap of 1. Examples of such family classes include spanning trees, matroids, and p-systems.

We saw in Section 2.1 that if the original function f is k-multi-submodular then the lifted function $\bar{f}$ is submodular. We now discuss some properties of the original families $\mathcal{F}_i$ and $\mathcal{F}$ that are also preserved under the lifting reduction; these were already proved in Santiago & Shepherd (2018). It is shown there, for instance, that if $\mathcal{F}$ induces a matroid (or more generally a p-system) over the ground set V, then so does the family $\mathcal{F}'$ over the lifted space E. We summarize some of these results in Table 1, and discuss next some of the algorithmic consequences.

In the setting of MVSO this invariance allows us to leverage several results from the single-agent to the multivariate setting. These are based on the following result, which uses the fact that the size of the lifted space E is nk.

Theorem 14. *Let $\mathcal{F}$ be a matroid, a p-matroid intersection, or a p-system. If there is a (polytime) $\alpha(n)$-approximation algorithm for monotone (resp. nonmonotone) $SO(\mathcal{F})$ maximization (resp. minimization), then there is a (polytime) $\alpha(nk)$-approximation algorithm for monotone (resp. nonmonotone) $MVSO(\mathcal{F})$ maximization (resp. minimization).*

For both monotone and nonmonotone maximization the approximation factors $\alpha(n)$ for the family classes described in Theorem 14 are independent of (the size of the ground set) n. Hence, we immediately get that $\alpha(nk) = \alpha(n)$ for those cases, and thus approximation factors for the corresponding multivariate and single-agent problems are the same. In our MV gap terminology this implies an MV gap of 1 for such problems. This proves Theorem 5.

In the setting of CMVSO the results described on entries 5 and 6 of Table 1 provide additional modelling flexibility. This allows us to maintain good approximations while combining several constraints. For example, for a monotone maximization instance of CMVSO where $\mathcal{F}$ corresponds to a p-matroid intersection and the $\mathcal{F}_i$ are all matroids, the above invariance results lead to a $(\frac{1}{p+1} - \epsilon)$-approximation.

The results from this section also imply that algorithms that behave very well in practice (such as accelerated greedy variants (Mirzasoleiman et al., 2016a) and distributed algorithms (Mirzasoleiman et al., 2016b)) for the corresponding single-agent problems, can also be used for the more gen-

eral multivariate setting while preserving the same approximation guarantees. We believe this makes the CMVSO framework a good candidate for potential applications in large-scale machine learning problems.

3. Multivariate Submodular Minimization

In this section we give (almost tight) curvature dependent approximations for minimization. Due to space limit, we omit the discussion and proof of Theorem 6 to Section 3.1 of the full version.

3.1. An $o(\frac{n}{\log n})$ Lower Bound Hardness for $\mathcal{F} = \{V\}$

In this section we focus on the special case where $\mathcal{F} = \{V\}$, and show that the curvature dependent approximation factors obtained in Theorem 6 are essentially tight (see Section 3.3 of the full version for proofs). We follow a technique from Goemans et al. (2009); Feige et al. (2011); Svitkina & Fleischer (2011) and build two multivariate submodular functions that are hard to distinguish with high probability for any (even randomized) algorithm.

Assume that $k = n$, and let $\mathcal{R} := (R_1, R_2, \ldots, R_n) \subseteq V^n$ be a random partition of V. Notice that $\sum_{i=1}^{n} |R_i| = n$. Let $\beta = \omega(\log n)$ and such that β is integer. Consider the two nonnegative monotone n-multi-submodular functions $f_1, f_2 : 2^{nV} \to \mathbb{R}_+$ given by:

$$f_1(S_1, \ldots, S_n) = \min\{n, \sum_{i=1}^{n} |S_i|\}, \qquad (3)$$

$$f_2(S_1, \ldots, S_n) = \min\{f_1(S_1, \ldots, S_n), \beta + \sum_{i=1}^{n} |S_i \cap \bar{R}_i|\}$$

where $\bar{R}_i = V - R_i$. We then have the following (see Section 3.3 of full version for proof details).

Lemma 15. *Any algorithm that makes a polynomial number of oracle calls has probability $n^{-\omega(1)}$ of distinguishing the functions f_1 and f_2 above.*

We now prove our (curvature independent) lower bound.

Theorem 16. *The monotone MVSO($\mathcal{F}$) minimization problem over $\mathcal{F} = \{V\}$ cannot be approximated to a ratio $o(n/\log n)$ in the value oracle model with polynomially many queries.*

Proof. Assume there is a polytime algorithm achieving an approximation factor of $\alpha = o(n/\log n)$. Choose $\beta = \omega(\log n)$ such that $\alpha\beta < n$. Consider the output of the algorithm when f_2 is given as input. The optimal solution in this case is the partition $\mathcal{R} = (R_1, \ldots, R_k)$, with $f_2(R_1, \ldots, R_k) = \beta$. So the algorithm produces a feasible solution (i.e. a partition) $(S_1^*, \ldots, S_k^*)$ satisfying $f_2(S_1^*, \ldots, S_k^*) \leq \alpha\beta < n$. However, since f_1 takes value

exactly n over any partition, there is no feasible solution $(S_1, \ldots, S_n)$ such that $f_1(S_1, \ldots, S_n) < n$. This means that if the input is the function f_1 then the algorithm produces a different answer, thus distinguishing between f_1 and f_2, contradicting Lemma 15. $\square$

The above result contrasts with the known $O(\log n)$ approximation (Svitkina & Tardos, 2010) for the case where the multivariate objective is separable, that is $f(S_1, \ldots, S_k) = \sum_i f_i(S_i)$ with f_i monotone submodular. These two facts combined now prove Theorem 3.

We use a construction from Iyer et al. (2013) to explicitly introduce the effect of curvature into the lower bound. Their work is for univariate functions, but it can be naturally extended to the multivariate setting. We modify the functions f_1, f_2 from (3) as follows:

$$f_i^c(S_1, \ldots, S_k) = c \cdot f_i(S_1, \ldots, S_k) + (1-c) \sum_{i=1}^{n} |S_i|, \ i = 1, 2.$$

Both f_1^c and f_2^c have total curvature c. Moreover, since $f_1(S_1, \ldots, S_k) = f_2(S_1, \ldots, S_k)$ if and only if $f_1^c(S_1, \ldots, S_k) = f_2^c(S_1, \ldots, S_k)$, by Lemma 15 it follows that any algorithm that makes polynomially many queries cannot distinguish between f_1^c and f_2^c with high probability. In addition, the gap between the optimal solutions for these two functions is given by

$$\frac{OPT_1}{OPT_2} = \frac{cn + (1-c)n}{c\beta + (1-c)n} = \frac{n}{c\beta + (1-c)n}$$

$$= \frac{n}{\beta + (n-\beta)(1-c)} = \frac{n/\beta}{1 + (n/\beta - 1)(1-c)}.$$

Then, since $\beta = \omega(\log n)$, the lower bound follows.

Theorem 7. *The monotone MVSO($\mathcal{F}$) minimization problem over $\mathcal{F} = \{V\}$ and objectives f with total curvature c cannot be approximated to a ratio $o(\frac{n/\log n}{1 + (\frac{n}{\log n} - 1)(1-c)})$ in the value oracle model with polynomial number of queries.*

4. Conclusions

We introduce a new class of multivariate submodular optimization problems, and give information theoretic evidence that this class encodes much more than the separable versions arising in multi-agent objectives. We provide some explicit examples and potential applications.

For maximization, we show that practical algorithms such as accelerated greedy variants and distributed algorithms achieve good approximation guarantees under very general constraints. Moreover, for arbitrary families we show MV gaps of $1 - 1/e$ and 0.385 for the monotone and non-monotone problems respectively. For minimization, we give (essentially tight) approximation factors with respect to the curvature of the multivariate objective function.

Acknowledgements

References

Ando, K., Fujishige, S., and Naitoh, T. A characterization of bisubmodular functions. *Discrete Mathematics*, 148 (1-3):299–303, 1996.

Bai, W. and Bilmes, J. A. Greed is still good: Maximizing monotone submodular+ supermodular functions. *arXiv preprint arXiv:1801.07413*, 2018.

Bouchet, A. and Cunningham, W. H. Delta-matroids, jump systems, and bisubmodular polyhedra. *SIAM Journal on Discrete Mathematics*, 8(1):17–32, 1995.

Boykov, Y. Y. and Jolly, M.-P. Interactive graph cuts for optimal boundary & region segmentation of objects in nd images. In *Computer Vision, 2001. ICCV 2001. Proceedings. Eighth IEEE International Conference on*, volume 1, pp. 105–112. IEEE, 2001.

Buchbinder, N. and Feldman, M. Constrained submodular maximization via a non-symmetric technique. *arXiv preprint arXiv:1611.03253*, 2016.

Buchbinder, N., Feldman, M., Seffi, J., and Schwartz, R. A tight linear time (1/2)-approximation for unconstrained submodular maximization. *SIAM Journal on Computing*, 44(5):1384–1402, 2015.

Calinescu, G., Chekuri, C., Pál, M., and Vondrák, J. Maximizing a submodular set function subject to a matroid constraint. In *Integer programming and combinatorial optimization*, pp. 182–196. Springer, 2007.

Calinescu, G., Chekuri, C., Pál, M., and Vondrák, J. Maximizing a monotone submodular function subject to a matroid constraint. *SIAM Journal on Computing*, 40(6): 1740–1766, 2011.

Chekuri, C. and Ene, A. Submodular cost allocation problem and applications. In *International Colloquium on Automata, Languages, and Programming*, pp. 354–366. Springer, 2011. Extended version: arXiv preprint arXiv:1105.2040.

Conforti, M. and Cornuéjols, G. Submodular set functions, matroids and the greedy algorithm: tight worst-case bounds and some generalizations of the rado-edmonds theorem. *Discrete applied mathematics*, 7(3):251–274, 1984.

El-Arini, K., Veda, G., Shahaf, D., and Guestrin, C. Turning down the noise in the blogosphere. In *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 289–298. ACM, 2009.

Ene, A. and Nguyen, H. L. Constrained submodular maximization: Beyond 1/e. In *Foundations of Computer Science (FOCS), 2016 IEEE 57th Annual Symposium on*, pp. 248–257. IEEE, 2016.

Ene, A. and Vondrák, J. Hardness of submodular cost allocation: Lattice matching and a simplex coloring conjecture. *Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques (APPROX/RANDOM 2014)*, 28:144–159, 2014.

Feige, U., Mirrokni, V. S., and Vondrak, J. Maximizing non-monotone submodular functions. *SIAM Journal on Computing*, 40(4):1133–1153, 2011.

Feldman, M., Naor, J., and Schwartz, R. A unified continuous greedy algorithm for submodular maximization. In *Foundations of Computer Science (FOCS), 2011 IEEE 52nd Annual Symposium on*, pp. 570–579. IEEE, 2011.

Fisher, M. L., Nemhauser, G. L., and Wolsey, L. A. *An analysis of approximations for maximizing submodular set functions-II*. Springer, 1978.

Fleischer, L., Goemans, M. X., Mirrokni, V. S., and Sviridenko, M. Tight approximation algorithms for maximum general assignment problems. In *Proceedings of the seventeenth annual ACM-SIAM symposium on Discrete algorithm*, pp. 611–620. Society for Industrial and Applied Mathematics, 2006.

Fujishige, S. and Iwata, S. Bisubmodular function minimization. *SIAM Journal on Discrete Mathematics*, 19(4): 1065–1073, 2005.

Goel, G., Karande, C., Tripathi, P., and Wang, L. Approximability of combinatorial problems with multi-agent submodular cost functions. In *Foundations of Computer Science, 2009. FOCS'09. 50th Annual IEEE Symposium on*, pp. 755–764. IEEE, 2009.

Goemans, M. X. and Ramakrishnan, V. Minimizing submodular functions over families of sets. *Combinatorica*, 15(4):499–513, 1995.

Goemans, M. X., Harvey, N. J., Iwata, S., and Mirrokni, V. Approximating submodular functions everywhere. In *Proceedings of the twentieth annual ACM-SIAM symposium on Discrete algorithms*, pp. 535–544. Society for Industrial and Applied Mathematics, 2009.

Golovin, D. and Krause, A. Adaptive submodularity: Theory and applications in active learning and stochastic optimization. *Journal of Artificial Intelligence Research*, 42:427–486, 2011.

Goundan, P. R. and Schulz, A. S. Revisiting the greedy approach to submodular set function maximization. *Optimization online*, pp. 1–25, 2007.

Grötschel, M., Lovász, L., and Schrijver, A. *Geometric algorithms and combinatorial optimization*, volume 2. Springer Science & Business Media, 2012.

Gupta, A., Roth, A., Schoenebeck, G., and Talwar, K. Constrained non-monotone submodular maximization: Offline and secretary algorithms. In *International Workshop on Internet and Network Economics*, pp. 246–257. Springer, 2010.

Hayrapetyan, A., Swamy, C., and Tardos, É. Network design for information networks. In *Proceedings of the sixteenth annual ACM-SIAM symposium on Discrete algorithms*, pp. 933–942. Society for Industrial and Applied Mathematics, 2005.

Hazan, E., Safra, S., and Schwartz, O. On the complexity of approximating k-set packing. *computational complexity*, 15(1):20–39, 2006.

Huber, A. and Kolmogorov, V. Towards minimizing k-submodular functions. In *International Symposium on Combinatorial Optimization*, pp. 451–462. Springer, 2012.

Huber, A., Krokhin, A., and Powell, R. Skew bisubmodularity and valued csps. *SIAM Journal on Computing*, 43 (3):1064–1084, 2014.

Iwata, S. and Nagano, K. Submodular function minimization under covering constraints. In *Foundations of Computer Science, 2009. FOCS'09. 50th Annual IEEE Symposium on*, pp. 671–680. IEEE, 2009.

Iwata, S., Fleischer, L., and Fujishige, S. A combinatorial strongly polynomial algorithm for minimizing submodular functions. *Journal of the ACM (JACM)*, 48(4): 761–777, 2001.

Iyer, R., Jegelka, S., and Bilmes, J. Monotone closure of relaxed constraints in submodular optimization: Connections between minimization and maximization: Extended version. 2014.

Iyer, R. K., Jegelka, S., and Bilmes, J. A. Curvature and optimal algorithms for learning and minimizing submodular functions. In *Advances in Neural Information Processing Systems*, pp. 2742–2750, 2013.

Jegelka, S. and Bilmes, J. Submodularity beyond submodular energies: coupling edges in graph cuts. In *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*, pp. 1897–1904. IEEE, 2011.

Kempe, D., Kleinberg, J., and Tardos, É. Maximizing the spread of influence through a social network. In *Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 137–146. ACM, 2003.

Khot, S., Lipton, R. J., Markakis, E., and Mehta, A. Inapproximability results for combinatorial auctions with submodular utility functions. In *International Workshop on Internet and Network Economics*, pp. 92–101. Springer, 2005.

Kohli, P., Kumar, M. P., and Torr, P. H. P^3 & beyond: Move making algorithms for solving higher order functions. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 31(9):1645–1656, 2009.

Korula, N., Mirrokni, V., and Zadimoghaddam, M. Online submodular welfare maximization: Greedy beats 1/2 in random order. *SIAM Journal on Computing*, 47(3):1056–1086, 2018.

Krause, A. and Guestrin, C. Near-optimal observation selection using submodular functions. In *AAAI*, volume 7, pp. 1650–1654, 2007.

Krause, A., Leskovec, J., Guestrin, C., VanBriesen, J., and Faloutsos, C. Efficient sensor placement optimization for securing large water distribution networks. *Journal of Water Resources Planning and Management*, 134(6): 516–526, November 2008.

Krause, K., Goodwin, M., and Smith, R. *Optimal software test planning through automated network analysis*. TRW Systems Group, 1973.

Lee, J., Mirrokni, V. S., Nagarajan, V., and Sviridenko, M. Non-monotone submodular maximization under matroid and knapsack constraints. In *Proceedings of the forty-first annual ACM symposium on Theory of computing*, pp. 323–332. ACM, 2009.

Lee, J., Sviridenko, M., and Vondrák, J. Submodular maximization over multiple matroids via generalized exchange properties. *Mathematics of Operations Research*, 35(4): 795–806, 2010.

Lehmann, B., Lehmann, D., and Nisan, N. Combinatorial auctions with decreasing marginal utilities. *Games and Economic Behavior*, 55(2):270–296, 2006. URL http://EconPapers.repec.org/RePEc:eee:gamebe:v:55:y:2006:i:2:p:270-296.

Lin, H. and Bilmes, J. A class of submodular functions for document summarization. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies-Volume 1*, pp. 510–520. Association for Computational Linguistics, 2011.

Lovász, L. Submodular functions and convexity. In *Mathematical Programming The State of the Art*, pp. 235–257. Springer, 1983.

Mirrokni, V., Schapira, M., and Vondrák, J. Tight information-theoretic lower bounds for welfare maximization in combinatorial auctions. In *Proceedings of the 9th ACM conference on Electronic commerce*, pp. 70–77. ACM, 2008.

Mirzasoleiman, B., Badanidiyuru, A., and Karbasi, A. Fast constrained submodular maximization: Personalized data summarization. In *ICML*, pp. 1358–1367, 2016a.

Mirzasoleiman, B., Karbasi, A., Sarkar, R., and Krause, A. Distributed submodular maximization. *The Journal of Machine Learning Research*, 17(1):8330–8373, 2016b.

Nemhauser, G. L. and Wolsey, L. A. Best algorithms for approximating the maximum of a submodular set function. *Mathematics of operations research*, 3(3):177–188, 1978.

Nemhauser, G. L., Wolsey, L. A., and Fisher, M. L. An analysis of approximations for maximizing submodular set functions - i. *Mathematical Programming*, 14(1): 265–294, 1978.

Ohsaka, N. and Yoshida, Y. Monotone k-submodular function maximization with size constraints. In *Advances in Neural Information Processing Systems*, pp. 694–702, 2015.

Qi, L. Directed submodularity, ditroids and directed submodular flows. *Mathematical Programming*, 42(1-3): 579–599, 1988.

Santiago, R. and Shepherd, F. B. Multi-Agent Submodular Optimization. *Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques (APPROX/RANDOM 2018)*, 116:23:1–23:20, 2018.

Schrijver, A. A combinatorial algorithm minimizing submodular functions in strongly polynomial time. *Journal of Combinatorial Theory, Series B*, 80(2):346–355, 2000.

Schrijver, A. *Combinatorial optimization: polyhedra and efficiency*, volume 24. Springer, 2003.

Singh, A., Guillory, A., and Bilmes, J. On bisubmodular maximization. In *Artificial Intelligence and Statistics*, pp. 1055–1063, 2012.

Streeter, M. and Golovin, D. An online algorithm for maximizing submodular functions. In *Advances in Neural Information Processing Systems*, pp. 1577–1584, 2009.

Svitkina, Z. and Fleischer, L. Submodular approximation: Sampling-based algorithms and lower bounds. *SIAM Journal on Computing*, 40(6):1715–1737, 2011.

Svitkina, Z. and Tardos, É. Facility location with hierarchical facility costs. *ACM Transactions on Algorithms (TALG)*, 6(2):37, 2010.

Thapper, J. and Zivny, S. The power of linear programming for valued csps. In *Foundations of Computer Science (FOCS), 2012 IEEE 53rd Annual Symposium on*, pp. 669–678. IEEE, 2012.

Thapper, J. and Živnỳ, S. The complexity of finite-valued csps. *Journal of the ACM (JACM)*, 63(4):37, 2016.

Vondrák, J. Optimal approximation for the submodular welfare problem in the value oracle model. In *Proceedings of the fortieth annual ACM symposium on Theory of computing*, pp. 67–74. ACM, 2008.

Vondrák, J. Submodularity and curvature: The optimal algorithm (combinatorial optimization and discrete algorithms). 2010.

Ward, J. and Živnỳ, S. Maximizing bisubmodular and k-submodular functions. In *Proceedings of the Twenty-Fifth Annual ACM-SIAM Symposium on Discrete Algorithms*, pp. 1468–1481. SIAM, 2014.