Diverse Approximations for Monotone Submodular Maximization Problems with a Matroid Constraint

Anh Viet Do, **Mingyu Guo**, **Aneta Neumann** and **Frank Neumann**
Optimisation and Logistics, School of Computer and Mathematical Sciences, The University of Adelaide
{vietanh.do,mingyu.guo,aneta.neumann,frank.neumann}@adelaide.edu.au

Abstract

Finding diverse solutions to optimization problems has been of practical interest for several decades, and recently enjoyed increasing attention in research. While submodular optimization has been rigorously studied in many fields, its diverse solutions extension has not. In this study, we consider the most basic variants of submodular optimization, and propose two simple greedy algorithms, which are known to be effective at maximizing monotone submodular functions. These are equipped with parameters that control the trade-off between objective and diversity. Our theoretical contribution shows their approximation guarantees in both objective value and diversity, as functions of their respective parameters. Our experimental investigation with maximum vertex coverage instances demonstrates their empirical differences in terms of objective-diversity trade-offs.

1 Introduction

Optimization research has seen rising interest in diverse solutions problems, where multiple maximally distinct solutions of high quality are sought instead of a single solution [Ingmar *et al.*, 2020; Baste *et al.*, 2022; Hanaka *et al.*, 2021; Fomin *et al.*, 2021; Fomin *et al.*, 2020; Hanaka *et al.*, 2022b; Hanaka *et al.*, 2022a]. This class of problem is motivated by practical issues largely overlooked in traditional optimization. Having diverse solutions gives resilient backups in response to changes in the problems rendering the current solution undesirable. It also gives the users the flexibility to correct for gaps between the problem models and real-world settings, typically caused by estimation errors, or aspects of the problem that cannot be formulated precisely [Schittekat and Sörensen, 2009]. Furthermore, diverse solution sets contain rich information about the problem instance by virtue of being diverse, which helps augment decision making capabilities. While there are methods to enumerate high quality solutions, having too many overwhelms the decision makers [Glover *et al.*, 2000], and a small, diverse subset can be more useful. It is also known that k-best enumeration tends to yield highly similar solutions, motivating the use of diversification mechanisms [Wang *et al.*, 2013; Yuan *et al.*, 2015; Hao *et al.*, 2020].

The diverse solutions problem have been studied as an extension to many important and difficult problems. Some examples of fundamental problems include constraint satisfaction and optimization problems [Hebrard *et al.*, 2005; Petit and Trapp, 2015; Ruffini *et al.*, 2019], SAT and answer set problem [Nadel, 2011; Eiter *et al.*, 2009], and mixed integer programming paradigms [Glover *et al.*, 2000; Danna *et al.*, 2007; Trapp and Konrad, 2015]. More recently, the first provably fixed-parameter tractable algorithms have been proposed for diverse solutions to a number of graph-based vertex problems [Baste *et al.*, 2022], as motivated by the complexity of finding multiple high performing solutions. This inspired subsequent research on other combinatorial structures such as trees, paths [Hanaka *et al.*, 2021; Hanaka *et al.*, 2022b], matching [Fomin *et al.*, 2020], independent sets [Fomin *et al.*, 2021], and linear orders [Arrighi *et al.*, 2021]. Furthermore, general frameworks have been proposed for diverse solutions to any combinatorial problem [Ingmar *et al.*, 2020; Hanaka *et al.*, 2022a]. To address the need to obtain both quality and diversity, multicriteria optimization has been considered, leading to interesting results [Gao *et al.*, 2022]. These are mostly applied to problems with linear objective functions and specific matroid intersection constraints.

In this work, we are interested in diverse solutions problem in the domain of submodular optimization, which has been enjoying widespread interests. It captures the diminishing returns property that arises in many real-world problems in machine learning, signal processing [Tohidi *et al.*, 2020], sensor placement [Krause and Guestrin, 2005], data summarization [Lin and Bilmes, 2011; Mirzasoleiman *et al.*, 2013], influence maximization [Kempe *et al.*, 2015], to name a few. Moreover, its hardness (as it generalizes many fundamental NP-hard combinatorial problems) and well-structuredness (which facilitates meaningful results [Vondrák, 2013]) mean the problem class also sees much attention from theoretical perspectives, leading to interesting insights [Nemhauser *et al.*, 1978; Fisher *et al.*, 1978; Calinescu *et al.*, 2011; Krause and Golovin, 2014; Chekuri *et al.*, 2014]. It is important to distinguish between the diverse solutions extension to submodular optimization and results diversification [Zheng *et al.*, 2017], the latter of which considers diversity as a measure of a solution (i.e. a selection of results) and optimizes it along with a submodular utility function.

Our Contributions We investigate the problem of finding a given number of diverse solutions to maximizing a monotone submodular function over a matroid, with a lower bound on solutions' objective values. Matroids are a type of independence system that can be used to model constraints in many important problems, and have been studied in in submodular optimization literature [Conforti and Cornuéjols, 1984; Lee *et al.*, 2010; Calinescu *et al.*, 2011; Kashaev and Santiago, 2023; Chekuri *et al.*, 2014], even recently appeared in diverse solutions research [Fomin *et al.*, 2021]. Among them, uniform matroids which characterize cardinality constraints, and their extension, partition matroids, are often considered in budgeted optimization (e.g. [Lin and Bilmes, 2010]). We consider the distance-sum measure of diversity, which is often chosen for diverse solutions problems [Hanaka *et al.*, 2021; Baste *et al.*, 2022; Hanaka *et al.*, 2022b; Hanaka *et al.*, 2022a; Gao *et al.*, 2022]. Its sole reliance on the ground set elements' representation in the solution set implies generalizability to other diversity measures such as entropy. Our contributions are as follows:

- We propose two simple greedy algorithms which are suitable to deal with the objective requirement, as greedy algorithms are known to perform well on monotone submodular maximization [Nemhauser *et al.*, 1978; Fisher *et al.*, 1978]. The novelty lies in the additional parameters, which adjust the trade-off between guarantees on objective values and diversity. We position our algorithms as simpler, zeroth-order (in terms of objective and independence oracles) alternatives to general frameworks for diverse solutions in recent literature, which have not been analyzed in submodular optimization context.
- We provide analyses of these algorithms in terms of their objective-diversity guarantees trade-offs. Our results are formulated as functions of their respective parameters, thus giving a general guidance on parameter selection. We also give sharpened bounds for cases with uniform matroids, as motivated by the prevalence of cardinality constraints. From these results, we point out settings that guarantee constant approximation ratios in objective, diversity, or both. Our tightness constructions also indicate certain features of matroids that make them pathological to these algorithms.
- We carry out an experimental investigation with maximum vertex coverage instances subjected to uniform and partition matroid constraints, to observe the algorithms' empirical performances in exhaustive parameter settings. The results indicate that while both algorithms produce nearly optimal solutions with reasonable diversity in many parameter settings, the simpler of the two actually provides better objective-diversity trade-offs across all problem settings. Additionally, these establish an empirical baseline for the diverse solutions problem considered in this work.

2 Preliminaries

In this section, we present the problem and relevant definitions, and give some observations that are helpful in our analyses.

2.1 Problem and Definitions

A multiset is a collection that can contain duplicates (e.g. $\{1,1,2\}$). For a set A, we denote the collection of multisets of elements in A with A^*, and $A^r \subseteq A^*$ contains r-size[1] multisets for some integer r. The problem we investigate is as follows: given integer $r \geq 2$, $\alpha \in [0,1]$, a (f,S,d,r,α)-instance asks for a multiset[2] of solutions in

$$\underset{P\in S^r}{\operatorname{argmax}} \left\{ d(P) : \forall x \in P, f(x) \geq \alpha \max_{y\in S} f(y) \right\}. \quad (1)$$

where the objective function, $f : 2^V \to \mathbb{R}$ is non-negative[3] and non-decreasing submodular, $S = \mathcal{I}$ for some matroid $M = (V, \mathcal{I})$, and d is a diversity measuring function defined over $(2^V)^*$. We do not consider non-increasing f due to trivial instances where achieving any positive diversity[4] necessitates degrading solutions beyond the feasibility limit. As per standard practice, we use "monotone" to mean "non-decreasing" in this paper. We call a multiset P feasible to the (f,S,d,r,α)-instance if $P \in S^r$ and every solution in P is a α-approximation of f over S, which is a solution $x \in S$ such that $f(x) \geq \alpha \max_{y\in S} f(y)$. We also briefly give relevant definitions and assumptions.

Definition 1 *Function $f : 2^V \to \mathbb{R}$ is monotone if $f(x) \leq f(y)$ for all $x \subseteq y \subseteq V$.*

Definition 2 *Function $f : 2^V \to \mathbb{R}$ is submodular if $\forall x, y \subseteq V, f(x) + f(y) \geq f(x \cup y) + f(x \cap y)$ or equivalently $\forall x \subseteq y \subseteq V, v \in V \setminus y, f(x \cup \{v\}) - f(x) \geq f(y \cup \{v\}) - f(y)$.*

For problem (1), we assume w.l.o.g. that $f(\emptyset) = 0$, since a multiset feasible to a (f,S,d,r,α)-instance is also feasible to the $(f+f',S,d,r,\alpha)$-instance for some constant non-negative function f'. We assume for our problem that f is given as a value oracle.

For matroid theory concepts, we adopt terminologies from the well-known text book [Oxley, 2011] on the subject.

Definition 3 *A tuple $M = (V, \mathcal{I} \subseteq 2^V)$ is a matroid if a) $\emptyset \in \mathcal{I}$, b) $\forall x \subseteq y \subseteq V, y \in \mathcal{I} \implies x \in \mathcal{I}$, c) $\forall x, y \in \mathcal{I}, |x| < |y| \implies \exists e \in y \setminus x, x \cup \{e\} \in \mathcal{I}$. The set V is the ground set, and $\mathcal{I}$ is the independence collection. A base of M is a maximal set in $\mathcal{I}$.*

Definition 4 *Given a matroid $M = (V, \mathcal{I})$,*

- *the rank function of M, $r_M : 2^V \to \mathbb{N}$, is defined as $r_M(x) = \max\{|y| : y \in 2^x \cap \mathcal{I}\}$, and the rank of M is $r_M = r_M(V)$,*
- *the closure function of M, $cl_M : 2^V \to 2^V$, is defined as $cl_M(x) = \{v \in V : r_M(x \cup \{v\}) = r_M(x)\}$,*
- *a loop of M is a $v \in V$ such that $\{v\} \notin \mathcal{I}$.*

[1] In this work, we use "r-size" to mean "containing r elements".

[2] Satisfying self-avoiding constraint requires algorithmic treatment beyond this work's scope.

[3] The non-negativity assumption is widely used in literature to ensure proper contexts for multiplicative approximation guarantees, which this work includes. This also applies to diversity w.l.o.g.

[4] Assuming the diversity measure returns 0 on duplicate-only multisets.

To give examples, a K-rank uniform matroid over V admits the independence collection $\mathcal{I} = \{x \subset V : |x| \leq K\}$ which we denote with $\mathcal{U}_{V,K}$. A partition matroid admits the independence collection $\mathcal{I} = \{x \subset V : \forall i = 1, \ldots, k, |x \cap B_i| \leq d_i\}$ for some partitioning $\{B_i\}_{i=1}^k$ of V and their corresponding thresholds $\{d_i\}_{i=1}^k$. In graph theory, a graphic matroid $M = (E, \mathcal{I})$ defined over a undirected graph $G = (V, E)$ is such that $\mathcal{I}$ contains all edge sets x where $G' = (V, x)$ has no cycle. A base of a graphic matroid is a spanning forest in the underlying graph, which itself is an object of much interest. Dual to the graphic matroid, the bond matroid $M^* = (E, \mathcal{I}^*)$ is such that $\mathcal{I}^*$ contains all edge sets x where $G^* = (V, E \setminus x)$ has the same number of connected components as G.

For the problem (1), we assume that M is loop-free and $|V| \geq 1$, implying $r_M > 0$. It is known that rank functions are monotone submodular, and closure functions are monotone, i.e. $x \subseteq y \implies cl_M(x) \subseteq cl_M(y)$ [Oxley, 2011]. We also assume that for a matroid, we are given an independence oracle answering whether a set is independent.

Finally, we consider the distance-sum diversity function, which is the usual choice in literature on diverse solutions problems [Hanaka *et al.*, 2021; Baste *et al.*, 2022; Hanaka *et al.*, 2022b; Hanaka *et al.*, 2022a; Gao *et al.*, 2022]. The function is defined over multisets of solutions as $ss(P) = \sum_{x,y \in P} |x \Delta y|$ where Δ is the symmetric difference between two sets, and its size is the Hamming distance. To be precise, each pairwise distance is counted once in an evaluation of ss.

Under this setting, the problem (1) is equivalent to the dispersion problem over the ground set that is the collection of all α-approximations of f over $\mathcal{I}$. The dispersion problem is known to be NP-hard in the ground set's size, even with known ground sets and metric distance functions [Wang and Kuo, 1988; Erkut, 1990; Ravi *et al.*, 1994; Chandra and Halldórsson, 1996]; for our problem, the collection is neither known nor necessarily small. On the other hand, [Hanaka *et al.*, 2022a] showed that this problem admits a poly-time $\max\{1 - 2/r, 1/2\}$-approximation scheme, predicated on a poly-time top-r enumeration scheme over this collection maximizing ss. We are not aware of such a scheme for α-approximations to submodular maximization over a matroid, and we recognize this as an interesting problem in its own right. That said, it is likely that algorithms resulted from this line of ideas will have significantly larger asymptotic run-times than those of the algorithms we present in this work.

2.2 Some Useful Properties

First, we observe that the value of ss is related to the occurrences of each elements of V in the multiset. Let P be a r-size multiset of subsets of V, and for $i = 1, \ldots, |V|$, $n_i(P) = |\{x \in P : i \in x\}|$, we have

$$ss(P) = \sum_{i \in V} n_i(P)[r - n_i(P)]. \tag{2}$$

This means the function can be decomposed into disjoint subsets of V: given a partitioning $\{V_i\}_{i=1}^k$ of V, we have $ss(P) = \sum_{i=1}^k ss(\{x \cap V_i : x \in P\})$. This property can significantly simplify analyses.

We would also like to bound the maximum achievable diversity in various settings. While without the constraint on the values of f, this bound can be computed (over a matroid) exactly and efficiently, e.g. by using the method in [Hanaka *et al.*, 2021], estimating it with a formula can be useful. To this end, we define a function $g : \mathbb{N}^3 \to \mathbb{N}$ with

$$g(a, b, c) = aq(c - q) + m(c - 2q - 1),$$

where $h = \min\{b, a/2\}$, and $m \in [0, a)$, q are integers such that $\lceil c/2 \rceil \lceil h \rceil + \lfloor c/2 \rfloor \lfloor h \rfloor = qa + m$. This function returns the maximum ss values of a c-size multisets of at most b-size subsets of a a-size ground set (Theorem 1). Also, we let $g(0, \cdot, \cdot) = g(\cdot, 0, \cdot) = g(\cdot, \cdot, 1) = 0$. For convenience, let $\delta : \mathbb{N}^2 \to \mathbb{N}$ be defined with $\delta(a, b) = a - 2b - 1$, we have

$$\begin{aligned}\forall x \in P, e \in V \setminus x, & ss(P \setminus \{x\} \cup \{x \cup \{e\}\}) - ss(P) \\ &= |P| - 2n_e(P) - 1 = \delta(|P|, n_e(P)).\end{aligned}$$

This expression exposes the connection between g and the process of adding elements into solutions in P, which is relevant to the algorithms we consider in this work. That is, we can rewrite g using δ: $g(a, b, c) = a \sum_{i=0}^{q-1} \delta(c, i) + m\delta(c, q)$; this simplifies the proof of its monotonicity (see Appendix). Here, we include an inequality which gives an intuitive bound of a result in Section 3.

Lemma 1 *Given integers $a, b, c \geq 1$ and $k \geq 0$, $g(\lceil ka/b \rceil, k, c) \geq kg(a, b, c)/y$.*

To establish an upper bound on diversity, we use the following straightforward observation from the fact that uniform matroid constraints are the least restrictive.

Observation 1 *Given a set V, function f over 2^V, matroids $M = (V, \mathcal{I})$ and $M' = (V, \mathcal{I}')$ where M is uniform and $r_M \geq r_{M'}$, then the optimal value for the $(f, \mathcal{I}', d, r, 0)$-instance cannot exceed that for the $(f, \mathcal{I}, d, r, 0)$-instance with any $r \geq 1$, and real function d over $(2^V)^*$.*

With this, we can use uniform matroids to formulate a simple upper bound, which is also tight for some non-uniform matroids and, surprisingly, any value of the threshold ratio α.

Theorem 1 *The optimal value for a $(f, \mathcal{I}, ss, r, \alpha)$-instance for some matroid $M = (V, \mathcal{I})$, function f over 2^V, integer $r \geq 1$, and $\alpha \in [0, 1]$ is at most $g(|V|, r_M, r)$. Moreover, this bound is tight for any $|V| \geq 1$, $r \geq 1$, $\alpha \in [0, 1]$, and matroid rank $r_M \in [1, |V|]$, even if the matroid is non-uniform.*

In augment-type algorithms like greedy, how the feasible selection pool for a partial solution (i.e. set of elements that can be added without violating constraints) changes over the course of the algorithm influences the guaranteed quality of the final output. This insight was made evident in seminal works on greedy algorithms [Fisher *et al.*, 1978; Nemhauser *et al.*, 1978], and is replicated in subsequent works on submodular optimization under more complex constraints. This is especially important in diverse solutions, as high diversity can be seen as additional restrictions on the selection pool. In the context of matroid constraint, this pool is determined by the partial solution's closure, thus we include an observation connecting closures to the upper bound on diversity.

Lemma 2 *Let $M = (V, \mathcal{I})$ be a matroid (may contain loops), and $x \in \mathcal{I}$, then for all $y \in \mathcal{I}$, $|y \cap cl_M(x)| \leq |x|$. By extension, $|y \cap z| \leq r_M(z)$ for all $z \subseteq V$.*

Lemma 2 lets us sharpen the upper bound on ss values for highly non-uniform matroids.

Lemma 3 *Given a matroid $M = (V, \mathcal{I})$ (may contain loops) and integer $r \geq 1$, then for any $P \in \mathcal{I}^r$,*

$$ss(P) \leq \min_{x \in \mathcal{I}} \{g(|V| - |cl_M(x)|, r_M - \lfloor n_x \rfloor, r) + g(|cl_M(x)|, \lceil n_x \rceil, r)\},$$

where $n_x = \min\{r_M |cl_M(x)|/|V|, |x|\}$. There exists a matroid where equality holds.

3 Greedy Algorithms for Diverse Solutions

We describe two different greedy algorithms to obtain an approximation to the problem (1), by incrementally building solutions. They are greedy in the sense that they select, in each step, the "best" choice out of a selection pool. Here, choice refers to a solution-element pair where the element is added into the solution. The differences between the two algorithms lie in how this pool is defined, and the selection criteria. In both algorithms, the pool is controlled by a parameter, which determines a trade-off between objective values and diversity.

In the following, we claim several worst-case bounds, i.e. for all settings I (each including a problem instance and an algorithm parameter value) in a universe clear from the context, $p(I) \geq q(I)$ for some quantities p and q of the setting (e.g. optimal value, worst-case diversity, etc.) A bound is tight if there is a setting I' where $p(I') = q(I')$. It is nearly tight if instead we have $p(I') = q(I') + \epsilon$ for an arbitrary small $\epsilon > 0$ independent from other factors.

3.1 Diversifying Greedy With Common Elements

The first approach, outlined in Algorithm 1, is a deterministic version of a heuristic for a special case of problem (1), proposed in [Neumann *et al.*, 2021]. The idea is to first have all solutions share common elements selected by the classical greedy algorithm, so as to efficiently obtain some objective value guarantee. Then, in the second phase (starting from line 5), each solution is finalized with added elements that maximize ss, which are precisely those least represented. To be specific, in each iteration, the algorithm looks at all solution-element pairs which maintain independence, and selects a pair based on criteria, the first of which maximizes diversity (line 7). This approach is simple and efficient, but prevents the common elements from contributing to diversity. Here, we formulate the algorithm to take the number of common elements as an input (b), which cannot exceed the rank of the matroid constraint.

We observe that since the image of ss is polynomially bounded in size, there are frequently many equivalent choices in each iteration in the second phase, motivating the use of tie-breaking rules, which are formulated as lexicographical argmin at Line 7. Of note is the second rule, which prioritizes solutions with the fewest remaining choices. The idea is to minimize the shrinkage of the pool among under-represented elements (the inclusion of which incurs large marginal gains in diversity) with a simple heuristic. We show that this tie-breaking rule helps guarantee a non-trivial lower bound of ss value under a general matroid, whereas it makes no difference under a uniform matroid. The other tie-breaking rules aim to improve the minimum objective value whenever possible.

Algorithm 1: Greedy with common elements

Input: f, S, b, r // Assuming $b \leq \max_{z \in S} |z|$
Output: $P \in S^r$
1 $x \leftarrow \emptyset, \kappa(\cdot) \leftarrow \{u \in (V \setminus \cdot) : (\cdot \cup \{u\}) \in S\}$;
2 **while** $|x| < b$ *and* $\kappa(x) \neq \emptyset$ **do**
3 | $v \leftarrow \mathrm{argmax}_{u \in \kappa(x)} f(x \cup \{u\}), x \leftarrow x \cup \{v\}$;
4 $P \leftarrow \{x\}^r$; //P contains r duplicates of x
5 $R \leftarrow \{(z, v) : z \in P, v \in \kappa(z), f(z \cup \{v\}) \geq f(z) \wedge n_v(P) < \lceil r/2 \rceil\}$;
6 **while** $R \neq \emptyset$ **do** // below argmin over vectors is done in left-to-right lexicographical order
7 | $(y, v) \leftarrow \mathrm{argmin}_{(z,u) \in R}(n_u(P), |\kappa(z)|, f(z), f(z) - f(z \cup \{u\}))$;
8 | $P \leftarrow P \setminus \{y\} \cup \{y \cup \{v\}\}$;
9 | Update R as in Line 5;

The time complexity of Algorithm 1 is $O(b|V| + r(r_M - b)(|V| - b))$ in both value oracle model and independence oracle model. The algorithm may not return r bases if rr_M is sufficiently large relative to $|V|$. Additionally, having all solutions sharing elements can be undesirable in some applications. Note the condition $n_v(P) < \lceil r/2 \rceil$ at line 5 ensures $ss(P)$ never decreases during the second phase.

Let $\mathcal{A}(f, S, b, r)$ be the collection of possible outputs from Algorithm 1 when run with inputs f, S, b, r. We first show that in uniform constraint case, the algorithm returns a constant diversity for each input configuration.

Theorem 2 *For any monotone submodular f over 2^V, integers $K \geq 1$, $r \geq 2$, $b \in [0, K)$, and let $\alpha = 1 - e^{-b/K}$, $\forall P \in \mathcal{A}(f, \mathcal{U}_{V,K}, b, r)$, $ss(P) = g(|V| - b, K - b, r)$, thus Algorithm 1 is $g(|V| - b, K - b, r)/g(|V|, K, r)$-approximate for the $(f, \mathcal{U}_{V,K}, ss, r, \alpha)$-instance. Moreover, this ratio bound is tight for any $|V| \geq 1$, $r \geq 2$, $K \in [1, |V|]$, and $b \in [0, K)$.*

Due to the monotonicity of g, the diversity guarantee in Theorem 2 decreases with b. Specifically, Lemma 1 implies this ratio bound is at least $1 - b/K$, which is tight in many cases. We also observe this linear relationship frequently in our experimental results (Section 4). This also means by setting b such that b/K is constant, the algorithm guarantees simultaneously constant approximation ratios in both objective values and diversity, independent of $|V|$ and r. Additionally, with $\alpha = 1 - e^{-b/K}$, we have $1 - b/K = 1 + \ln(1 - \alpha)$, giving a direct objective-diversity trade-off curve (in terms of ratios) within $\alpha \in [0, 1 - 1/e]$.

For general matroids, we can infer the objective approximation guarantee from Algorithm 1 as a function of parameter b, by using an important result in [Fisher *et al.*, 1978].

Lemma 4 *Algorithm 1 under a matroid $M = (V, \mathcal{I})$ outputs $\left[1 - (1 - 1/r_M)^{\min\{b,k\}}\right]$-approximations of f over $\mathcal{I}$ where $k = \min\left\{|z| : z \in 2^V \setminus \mathcal{I}\right\} - 1$. Additionally, these solutions are $b/(2r_M)$-approximations.*

Algorithm 2: Greedy with representation limits

Input: f, S, r, l // Assuming $l \in [1, r]$
Output: $P \in S^r$
1 $v^* \leftarrow \text{argmax}_{v \in S} f(\{v\}), P \leftarrow \{\{v^*\}\}^r$;
2 $\kappa(\cdot) \leftarrow \{u \in (V \setminus \cdot) : (\cdot \cup \{u\}) \in S \wedge n_u(P) < l\}$;
3 $R \leftarrow \{(z, v) : z \in P, v \in \kappa(z), f(z \cup \{v\}) \geq f(z)\}$;
4 **while** $R \neq \emptyset$ **do** // below argmin over vectors is done in left-to-right lexicographical order
5 $\quad (y, v) \leftarrow \underset{(z,u) \in R}{\text{argmin}}(|z|, f(z) - f(z \cup \{u\}), f(z), n_u(P))$;
6 $\quad P \leftarrow P \setminus \{y\} \cup \{y \cup \{v\}\}$;
7 $\quad$ Update R as in Line 3;

Theorem 3 *For any monotone f over 2^V, matroid $M = (V, \mathcal{I})$, and integers $r \geq 2$, $b \in [0, r_M)$, $\forall P \in \mathcal{A}(f, \mathcal{U}_{V,K}, b, r)$, $ss(P) \geq g(r_M - b - 1, r_M - b - 1, r) + g(m, 1, r)$, where $m = |V \setminus cl_M(x)| - r_M + b + 1$ and x is the solution obtained in the first phase of the algorithm. Moreover, this bound is tight for any $|V| \geq 1$, $r \geq 2$, matroid rank $r_M \in [1, |V|]$, $b \in [0, s)$ and $m \in [1, |V| - r_M + b + 1]$.*

It is important to note that while the bound in Theorem 3 can be small for any positive choice of b if the closure of the common elements set x is large, sufficiently large ones (e.g. $|cl_M(x)|/|x| > |V|/r_M$) also lower the upper bound on maximum diversity, according to Lemma 3.

3.2 Simultaneous Greedy With Representation Limits

The second approach, outlined in Algorithm 2, is inspired by the SIMULTANEOUSGREEDYS algorithm proposed in [Feldman *et al.*, 2020], which obtains a set of disjoint solutions, in which the best one provides an approximation guarantee. Since for our problem, all solutions need to be sufficiently good, we make crucial changes to adapt the algorithm to the task. Firstly, each element can appear in multiple solutions, the maximum number of which is given as an input (l). This simultaneously expands the selection pool for each solution in each iteration, which helps with quality, and controls the amount of representation in the output each element enjoys, which guarantees some diversity. Secondly, a single element (v^*) is allowed to be included in all solutions, so a non-trivial quality guarantee is possible, as there are instances where excluding an element ensures that the solution is arbitrarily bad. Finally, the selection criteria, especially the first one, enforce building solutions evenly, so the worst one does not fall too far behind. This allows us to derive a non-trivial lower bound on the objective value of every solution in the output.

Compared to Algorithm 1, this algorithm does not maximize diversity directly, but guarantees it indirectly by imposing additional constraints. Since these constraints are on elements' representation, it can be applied to the problem (1) with any diversity measure that can be formulated by elements' representation, such as entropy [Neumann *et al.*, 2021].

The time complexity of Algorithm 2 is $O(rr_M|V|)$ in both value oracle model and independence oracle model. Like Algorithm 1, it may not return r bases if rr_M is sufficiently large and l is sufficiently small. We remark that the inclusion of the initial element v^* in all solutions is meant to deal with pathological instances; this might be avoided with a more complex heuristic. With a view to simplicity, we choose not to pursue this further in this work.

We observe that if $l = r$, Algorithm 2 must return solutions obtainable by the classical greedy algorithm since if there is a $v \in V$ that cannot be added to a solution x due to the new constraint, then $v \in x$. However, one can construct instances where it is guaranteed to achieve ss value of 0, even when restricted to uniform matroids and linear objective functions. Therefore, we only consider cases where $l < r$. We show the extent to which diversity is guaranteed, simply from limiting elements' representations.

Similarly, we use $\mathcal{B}(f, S, r, l)$ to denote the collection of possible outputs from Algorithm 2 when run with inputs f, S, r, l. Given a matroid $M = (V, \mathcal{I})$, for any $P \in \mathcal{B}(f, \mathcal{I}, r, l)$ and $x \in P$, let P_t be P at iteration t ($P_0 = \{\{v^*\}\}^r$), $t_{x,i}$ be the iteration in which the i-th element is added to x (counting v^*), $x^{(i)}$ be x right after that iteration, $V_i = \left\{v \in V : n_v(P_{t_{x,i}-1}) \geq l\right\}$, $W_i = cl_M\left(x^{(i-1)}\right)$ and $U_i = V_i \cup W_i \setminus x^{(i-1)}$. Intuitively, U_i contains elements that cannot be added to x at step i; inspecting this set gives the following guarantee.

Lemma 5 *For any $Y \in (2^V)^*$, if for some $a, q \geq 0$, $\sum_{v \in U_i} n_v(Y) \leq a(i-1)$ for all $i = 1, \ldots, |x|$ and $\sum_{v \in U_{|x|+1}} n_v(Y) \leq a(i-1) + q$, then $\min\left\{a + q/|x| + |Y|, \sum_{y \in Y} |y|\right\} f(x) \geq \sum_{y \in Y} f(y)$.*

We show the lower bound on the ss value as the algorithm progresses.

Lemma 6 *For all $t \geq 0$, $ss(P_t) \geq \lfloor t/l \rfloor l(r-l) + c(r-c)$ where $c \in [0, l)$ such that $c \equiv t \mod l$.*

With Lemma 5 and 6, the following objective-diversity trade-off guarantees can be inferred.

Theorem 4 *Given monotone submodular f over 2^V, integers $r \geq 2$, $l \in [1, r)$ and $K \in [1, |V|]$, then for all $P \in \mathcal{B}(f, \mathcal{U}_{V,K}, r, l)$ and $k \in [1, (r-1)K/l]$,*

$$\min\left\{\frac{r-1}{l} + 1, k\right\} \min_{x \in P} f(x) \geq \max_{|y| \leq k} f(y) \qquad \textit{and}$$

$$ss(P) \geq l(r-l)\lfloor h/l \rfloor + c(r-c),$$

where $h = \min\{r(K-1), l(|V|-1)\}$ and $c \in [0, l)$ such that $c \equiv h \mod l$.[5] *Moreover, the former bound is nearly tight when $K \geq r + l - 1$, and the latter bound is tight for any $|V| \geq 1$, $r \geq 2$, $K \in [1, |V|]$, and $l \in [1, r]$.*

Theorem 5 *Given monotone submodular f over 2^V, matroid $M = (V, \mathcal{I})$, integers $r \geq 2$ and $l \in [1, r)$, then for all $P \in \mathcal{B}(f, \mathcal{I}, r, l)$*

$$\min\left\{\frac{r-1}{l} + 2, r_M\right\} \min_{x \in P} f(x) \geq \max_{y \in \mathcal{I}} f(y) \qquad \textit{and}$$

$$ss(P) \geq l(r-l)(r_M - 1).$$

Moreover, the former bound is nearly tight, and the latter bound is tight for any $|V| \geq 1$, $r \geq 2$, matroid rank $r_M \in [1, |V|]$, and $l \in [1, r]$.

[5]The latter bound holds trivially at $l = r$.

We remark that the tightness cases in the proof of Theorem 5 prevent Algorithm 2 from exercising the last tie-breaking rule, which is the component that lets it improve diversity beyond the lower bound. We suspect that this bound might be overly pessimistic for instances where the image under f of the feasible set is small.

The result suggests that for uniform constraints, setting $l = \max\{\lfloor r(r_M - 1)/(|V| - 1)\rfloor, 1\}$ leads to Algorithm 2 guaranteeing $(1 - 1/|V|)(1 - O(1/r_M))$ approximation ratio in diversity, whereas $l = \lfloor r/2\rfloor$ guarantees $(r_M - 1)/|V|$ approximation ratio for pathological matroid constraint. Additionally, if l/r is constant, then every output solution guarantees a constant approximation ratio in objective value.

Above results only consider extreme values (e.g. optimal f value). On the other hand, by comparing the algorithm's output against an arbitrary solution set, a more nuanced picture emerges which suggests the algorithm can exploit a certain feature in the global structure of f to lessen compromise on diversity (i.e. by lowering parameter l) while maintaining objective guarantees.

Theorem 6 *Given monotone submodular f over 2^V, integers $r \geq 2$, $l \in [1, r)$, $K \in [1, |V|]$ and $Y \in (2^V)^*$ such that $m = \max_{v\in V} n_v(Y)$, then for all $P \in \mathcal{B}(f, \mathcal{U}_{V,K}, r, l)$*

$$\min\left\{\frac{m(r-1)h}{l} + |Y|, \sum_{y\in Y} |y|\right\} \min_{x\in P} f(x) \geq \sum_{y\in Y} f(y),$$

where $h = \max\left\{l \sum_{y\in Y} |y|/[Km(r-1)], 1\right\}$. This bound is nearly tight for all $r \geq 2$, $l \in [1, r)$, size of Y and $m \in [1, |Y|]$.

Corollary 1 *If there is $Y \in (2^V)^k$ for some $k \geq 1$ where $\max_{v\in V} n_v(Y) < l \sum_{y\in Y} |y|/[K(r-1)]$ and $\sum_{y\in Y} f(y)/k \geq \alpha \max_{|y|\leq K} f(y)$, then Algorithm 2 under K-rank uniform constraint returns $\alpha/2$-approximations with parameter l. If there is a set of k disjoint α-approximations, then Algorithm 2 returns $\alpha/2$-approximations at any $l \in [(r-1)/k, r)$.*

Theorem 7 *Given monotone submodular f over 2^V, integers $r \geq 2$, $l \in [1, r)$, matroid $M = (V, \mathcal{I})$ and $Y \in \mathcal{I}^*$ such that $m = \max_{v\in V} n_v(Y)$, then for all $P \in \mathcal{B}(f, \mathcal{I}, r, l)$*

$$\min\left\{\frac{m(r-1)}{l} + 2|Y|, \sum_{y\in Y} |y|\right\} \min_{x\in P} f(x) \geq \sum_{y\in Y} f(y).$$

This bound is nearly tight for all $r \geq 2$, $l \in [1, r)$, size of Y and $m \in [1, |Y|]$.

Corollary 2 *Given a matroid $M = (V, \mathcal{I})$, if there is $Y \in \mathcal{I}^k$ for some $k \geq 1$ where $\max_{v\in V} n_v(Y) \leq lk/(r-1)$ and $\sum_{y\in Y} f(y)/k \geq \alpha \max_{y\in\mathcal{I}} f(y)$, then Algorithm 2 under matroid constraint M returns $\alpha/3$-approximations with parameter l. If there is a set of k disjoint α-approximations, then Algorithm 2 returns $\alpha/3$-approximations at any $l \in [(r-1)/k, r)$ and $\alpha/(2 + 1/k)$-approximations at $l = r - 1$.*

Going further, these bounds can be strictly improved when the number of disjoint optimal solutions exceeds certain thresholds. In particular, we show that in such cases, Algorithm 2 guarantees objective values identical to those from the classical greedy when maximizing monotone submodular functions under the same constraints [Nemhauser *et al.*, 1978; Fisher *et al.*, 1978]. For a function f and a solution set S let $D(f, S, \alpha)$ be the largest number of disjoint non-empty α-approximations of f over S, and for a solution x, let i_x be its size before it stops being improved by the algorithm.

Theorem 8 *Given monotone submodular f over 2^V, integers $r \geq 2$, $l \in [1, r)$, and matroid $M = (V, \mathcal{I})$, then for all $P \in \mathcal{B}(f, \mathcal{I}, r, l)$, given $x \in P$ where $|x| > 1$ and $D(f, \mathcal{I}, \alpha) > \lfloor\eta(r-1)/l\rfloor$ for some α, then*

- *$f(x) \geq \alpha \left[1 - (1 - 1/|x|)^{|x|}\right] \max_{y\in\mathcal{I}} f(y)$ if M is uniform and $\eta = |x| - 1$,*
- *$f(x) \geq \alpha \max_{y\in\mathcal{I}} f(y)/2$ if M is non-uniform and $\eta = i_x$.*

If $D(f, \mathcal{I}, \alpha) = \lfloor\eta(r-1)/l\rfloor$, the bound does not necessarily hold in either case.

We include a simple observation relating maximum diversity and the number of disjoint approximations.

Observation 2 *Let $k = D(f, S, \alpha)$ for some function f, solution set $S \subseteq 2^V$, threshold α, then the maximum ss value to a (f, S, r, α)-instance is at most $g(h, s, r)$ where $h = \min\{r(s-1) + k, |V|\}$ and $s = \max_{x\in S} |x|$.*

4 Experimental Investigation

To observe how these algorithms perform on concrete instances, we experiment with the maximum vertex coverage problem: given a graph $G = (V, E)$, find a set $x \in \mathcal{I}$ for some matroid $M = (V, \mathcal{I})$ that maximizes $|x \cup \{v \in V : \exists u \in x, \{u, v\} \in E\}|$, which is monotone submodular. For the benchmark instance, we use the complement of frb30-15-1, frb30-15-2, frb35-17-1, frb40-19-1 from the standard benchmark suite BHOSLIB created using the Model RB [Xu and Li, 2006], containing 450, 450, 595, 760 vertices respectively, and 17827, 17874, 27856, 41314 edges respectively; these are available at [Rossi and Ahmed, 2015].

We use four matroid constraints in the experiments, including 2 uniform matroids and 2 partition matroids. As mentioned, partition matroids admit a independence collections of the form $\mathcal{I} = \{x \subseteq V : \forall i = 1, \ldots, k, |x \cap V_i| \leq b_i\}$ for some partitioning $\{V_i\}_{i=1}^k$ of V and integers $\{b_i\}_{i=1}^k$. These are useful in modeling group-based budget constraints [Cornuejols *et al.*, 1977; Nemhauser *et al.*, 1978; Chekuri and Kumar, 2004; Chekuri and Pál, 2005; Fleischer *et al.*, 2006]. For uniform matroids, we set the ranks to $\{10, 15\}$, and denote them with U10 and U15, respectively (the numbers represent the ranks). For partition matroids, we group consecutive vertices sorted by degrees into 10 partitions, i.e. V_i contains from $|V|(i-1)/10+1$-th to $|V|i/10$-th smallest degree vertices. This is to force the solutions to include a limited number of high-degree vertices, creating scenarios where the greedy algorithm would obtain very different solutions from the ones it would return under uniform constraints. In case 10 does not divide $|V|$, we set $|V_i| = \lfloor|V|/10\rfloor$ for $i = 2, \ldots, 10$. For the first partition

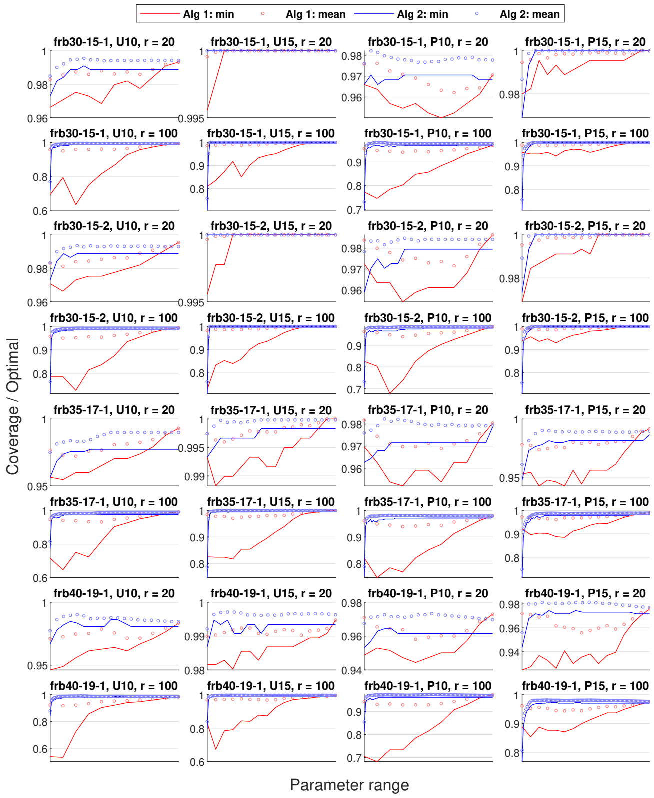

Figure 1: Minimum and mean objective values in the outputs of the algorithms run with all parameter values in respective ranges: $[0, r_M]$ for Algorithm 1, $[1, r]$ for Algorithm 2. Values are normalized against respective known optima.

matroid (denoted P10), we set b_i to 1 for all i, while for the second (denoted P15), we assign 6 to b_1 and 1 to the rest.

For each of the 16 instances, we run with $r \in \{20, 100\}$, Algorithm 1 with all parameter values $b \in [0, r_M]$, and Algorithm 2 with all parameter values $l \in [1, r]$. For both algorithms, the last tie-breaking is done by selecting the first choice (in increasing order of vertex labels). Therefore, there is no randomization, so each algorithm is run once on each instance with each parameter value.

To contextualize the results, we obtain a best known coverage for each instance using the built-in integer linear programming solver in MATLAB. Furthermore, the upper bounds on ss values are given by $\sum_i g(|V_i|, b_i, r)$ since ss can be decomposed by disjoint subsets (in case of uniform matroid, $V_i \leftarrow V$ and $b_i \leftarrow r_M$). Note that this bound applies to all threshold ratio $\alpha \in [0, 1]$, the actual optimal value might very well be much smaller, especially for α close to 1. We choose not to normalize our results against actual optimal values because *a*) solving exactly the problem (1) is prohibitively costly, and *b*) this is exacerbated by the large number of α values for each of which an optimal value needs to be obtained (i.e. the number of distinct minimum objective values from the algorithms on each instance).

The results are shown in Figure 1 and 2, which visualize, for each graph-constraint-parameter combination, the mean and minimum objective values in the output, and the ss value.

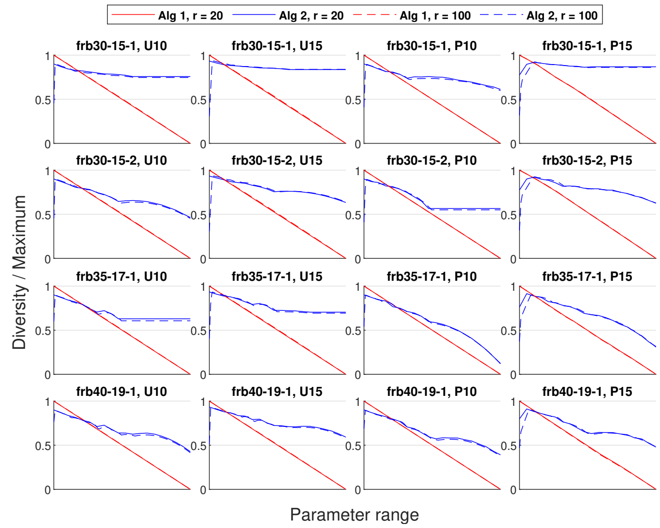

Figure 2: Diversity as ss values of the outputs of the algorithms run with all parameter values in respective ranges: $[0, r_M]$ for Algorithm 1, $[1, r]$ for Algorithm 2. Values are normalized against respective known upper bounds.

We see that the objective values in the outputs are high (within 5% gap of the optimal) for $r = 20$, and predictably degrade for $r = 100$ (about 30%), although the mean values stay within 10% in most settings. Notably, Algorithm 2 produces higher minimum objective values than Algorithm 1 does in most settings, and smaller gaps between mean values and minimum values. More importantly, Algorithm 2 achieves significantly higher ss values in most settings, thus yielding better objective-diversity trade-offs than Algorithm 1. This indicates benefits of limiting common elements by controlling their representation in the output, as they do not contribute to diversity.

Interestingly, increasing the output size r only seems to affect the objective values, as the relative diversity values are virtually the same across all settings. We suspect that this might change for more complex matroids. Incidentally, the impacts of r on objective values from Algorithm 2 seem minimal outside of edge cases (i.e. $l = 1$).

5 Conclusion

The diverse solutions problem is a challenging extension to optimization problems that is of practical and theoretical interests. In this work, we considered the problem of finding diverse solutions to maximizing monotone submodular functions over a matroid. To address the difficulty in finding multiple high-quality solutions, we exploited submodularity with two simple greedy algorithms, equipped with objective-diversity trade-off adjusting parameters. Theoretical guarantees by these algorithms were given in both objective values and diversity, as functions of their respective parameters. Our experimental investigation with maximum vertex coverage instances demonstrates strong empirical performances from these algorithms despite their simplicity.

Acknowledgements

This work was supported by the Australian Research Council through grants DP190103894 and FT200100536.

References

[Arrighi *et al.*, 2021] Emmanuel Arrighi, Henning Fernau, Daniel Lokshtanov, Mateus de Oliveira Oliveira, and Petra Wolf. Diversity in kemeny rank aggregation: A parameterized approach. In *IJCAI*, pages 10–16. International Joint Conferences on Artificial Intelligence Organization, August 2021.

[Baste *et al.*, 2022] Julien Baste, Michael R. Fellows, Lars Jaffke, Tomáš Masařík, Mateus de Oliveira Oliveira, Geevarghese Philip, and Frances A. Rosamond. Diversity of solutions: An exploration through the lens of fixed-parameter tractability theory. *Artificial Intelligence*, 303:103644, February 2022.

[Calinescu *et al.*, 2011] Gruia Calinescu, Chandra Chekuri, Martin Pál, and Jan Vondrák. Maximizing a monotone submodular function subject to a matroid constraint. *SIAM Journal on Computing*, 40(6):1740–1766, January 2011.

[Chandra and Halldórsson, 1996] Barun Chandra and Magnús M. Halldórsson. Facility dispersion and remote subgraphs. In *SWAT*, volume 1097 of *Lecture Notes in Computer Science*, pages 53–65. Springer Berlin Heidelberg, 1996.

[Chekuri and Kumar, 2004] Chandra Chekuri and Amit Kumar. Maximum coverage problem with group budget constraints and applications. In *APPROX-RANDOM*, volume 3122 of *Lecture Notes in Computer Science*, pages 72–83. Springer Berlin Heidelberg, 2004.

[Chekuri and Pál, 2005] Chandra Chekuri and Martin Pál. A recursive greedy algorithm for walks in directed graphs. In *FOCS*, pages 245–253. IEEE, 2005.

[Chekuri *et al.*, 2014] Chandra Chekuri, Jan Vondrák, and Rico Zenklusen. Submodular function maximization via the multilinear relaxation and contention resolution schemes. *SIAM Journal on Computing*, 43(6):1831–1879, January 2014.

[Conforti and Cornuéjols, 1984] Michele Conforti and Gérard Cornuéjols. 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, March 1984.

[Cornuejols *et al.*, 1977] Gerard Cornuejols, Marshall L. Fisher, and George L. Nemhauser. Location of bank accounts to optimize float: An analytic study of exact and approximate algorithms. *Management Science*, 23(8):789–810, April 1977.

[Danna *et al.*, 2007] Emilie Danna, Mary Fenelon, Zonghao Gu, and Roland Wunderling. Generating multiple solutions for mixed integer programming problems. In *IPCO*, pages 280–294. Springer Berlin Heidelberg, 2007.

[Eiter *et al.*, 2009] Thomas Eiter, Esra Erdem, Halit Erdoğan, and Michael Fink. Finding similar or diverse solutions in answer set programming. In *Logic Programming*, pages 342–356. Springer Berlin Heidelberg, 2009.

[Erkut, 1990] Erhan Erkut. The discrete p-dispersion problem. *European Journal of Operational Research*, 46(1):48–60, May 1990.

[Feldman *et al.*, 2020] Moran Feldman, Christopher Harshaw, and Amin Karbasi. Simultaneous greedys: A swiss army knife for constrained submodular maximization. *CoRR*, abs/2009.13998, 2020.

[Fisher *et al.*, 1978] M. L. Fisher, G. L. Nemhauser, and L. A. Wolsey. An analysis of approximations for maximizing submodular set functions—II. In *Mathematical Programming Studies*, pages 73–87. Springer Berlin Heidelberg, 1978.

[Fleischer *et al.*, 2006] Lisa Fleischer, Michel X. Goemans, Vahab S. Mirrokni, and Maxim Sviridenko. Tight approximation algorithms for maximum general assignment problems. In *SODA*, pages 611–620. ACM Press, 2006.

[Fomin *et al.*, 2020] Fedor V. Fomin, Petr A. Golovach, Lars Jaffke, Geevarghese Philip, and Danil Sagunov. Diverse pairs of matchings. In *ISAAC*, pages 26:1–26:12, Dagstuhl, Germany, 2020. Schloss Dagstuhl - Leibniz-Zentrum für Informatik.

[Fomin *et al.*, 2021] Fedor V. Fomin, Petr A. Golovach, Fahad Panolan, Geevarghese Philip, and Saket Saurabh. Diverse collections in matroids and graphs. In *STACS*, pages 31:1–31:14, Dagstuhl, Germany, 2021. Schloss Dagstuhl - Leibniz-Zentrum für Informatik.

[Gao *et al.*, 2022] Jie Gao, Mayank Goswami, C. S. Karthik, Meng-Tsung Tsai, Shih-Yu Tsai, and Hao-Tsung Yang. Obtaining approximately optimal and diverse solutions via dispersion. In *LATIN 2022: Theoretical Informatics*, pages 222–239. Springer International Publishing, 2022.

[Glover *et al.*, 2000] Fred Glover, Arne LøKketangen, and David L. Woodruff. Scatter search to generate diverse MIP solutions. In *Operations Research/Computer Science Interfaces Series*, pages 299–317. Springer US, Boston, MA, 2000.

[Hanaka *et al.*, 2021] Tesshu Hanaka, Yasuaki Kobayashi, Kazuhiro Kurita, and Yota Otachi. Finding diverse trees, paths, and more. *Proceedings of the AAAI Conference on Artificial Intelligence*, 35(5):3778–3786, May 2021.

[Hanaka *et al.*, 2022a] Tesshu Hanaka, Masashi Kiyomi, Yasuaki Kobayashi, Yusuke Kobayashi, Kazuhiro Kurita, and Yota Otachi. A framework to design approximation algorithms for finding diverse solutions in combinatorial problems. *CoRR*, abs/2201.08940, 2022.

[Hanaka *et al.*, 2022b] Tesshu Hanaka, Yasuaki Kobayashi, Kazuhiro Kurita, See Woo Lee, and Yota Otachi. Computing diverse shortest paths efficiently: A theoretical and experimental study. *Proceedings of the AAAI Conference on Artificial Intelligence*, 36(4):3758–3766, June 2022.

[Hao *et al.*, 2020] Fei Hao, Zheng Pei, and Laurence T. Yang. Diversified top-k maximal clique detection in social internet of things. *Future Generation Computer Systems*, 107:408–417, June 2020.

[Hebrard *et al.*, 2005] Emmanuel Hebrard, Brahim Hnich, Barry O'Sullivan, and Toby Walsh. Finding diverse and similar solutions in constraint programming. In *AAAI*, pages 372–377. AAAI Press / The MIT Press, 2005.

[Ingmar *et al.*, 2020] Linnea Ingmar, Maria Garcia de la Banda, Peter J. Stuckey, and Guido Tack. Modelling diversity of solutions. *Proceedings of the AAAI Conference on Artificial Intelligence*, 34(02):1528–1535, April 2020.

[Kashaev and Santiago, 2023] Danish Kashaev and Richard Santiago. A simple optimal contention resolution scheme for uniform matroids. *Theoretical Computer Science*, 940(Part A):81–96, 2023.

[Kempe *et al.*, 2015] David Kempe, Jon Kleinberg, and Éva Tardos. Maximizing the spread of influence through a social network. *Theory of Computing*, 11(4):105–147, 2015.

[Krause and Golovin, 2014] Andreas Krause and Daniel Golovin. Submodular function maximization. In *Tractability*, pages 71–104. Cambridge University Press, 2014.

[Krause and Guestrin, 2005] Andreas Krause and Carlos Guestrin. UAI. In *Proceedings of the Twenty-First Conference on Uncertainty in Artificial Intelligence*, UAI'05, pages 324–331, Arlington, Virginia, USA, 2005. AUAI Press.

[Lee *et al.*, 2010] Jon Lee, Vahab S. Mirrokni, Viswanath Nagarajan, and Maxim Sviridenko. Maximizing nonmonotone submodular functions under matroid or knapsack constraints. *SIAM Journal on Discrete Mathematics*, 23(4):2053–2078, January 2010.

[Lin and Bilmes, 2010] Hui Lin and Jeff Bilmes. Multi-document summarization via budgeted maximization of submodular functions. In *HLT-NAACL*, pages 912–920, Los Angeles, California, June 2010. Association for Computational Linguistics.

[Lin and Bilmes, 2011] Hui Lin and Jeff Bilmes. A class of submodular functions for document summarization. In *ACL*, HLT '11, pages 510–520, USA, 2011. Association for Computational Linguistics.

[Mirzasoleiman *et al.*, 2013] Baharan Mirzasoleiman, Amin Karbasi, Rik Sarkar, and Andreas Krause. Distributed submodular maximization: Identifying representative elements in massive data. In *NIPS*, NIPS'13, pages 2049–2057, Red Hook, NY, USA, 2013. Curran Associates Inc.

[Nadel, 2011] Alexander Nadel. Generating diverse solutions in SAT. In *SAT*, volume 6695 of *Lecture Notes in Computer Science*, pages 287–301. Springer Berlin Heidelberg, 2011.

[Nemhauser *et al.*, 1978] G. L. Nemhauser, L. A. Wolsey, and M. L. Fisher. An analysis of approximations for maximizing submodular set functions—I. *Mathematical Programming*, 14(1):265–294, December 1978.

[Neumann *et al.*, 2021] Aneta Neumann, Jakob Bossek, and Frank Neumann. Diversifying greedy sampling and evolutionary diversity optimisation for constrained monotone submodular functions. In *GECCO*. ACM, June 2021.

[Oxley, 2011] James Oxley. *Matroid Theory*. Oxford University Press, February 2011.

[Petit and Trapp, 2015] Thierry Petit and Andrew C. Trapp. Finding diverse solutions of high quality to constraint optimization problems. In *IJCAI*, IJCAI'15, pages 260—-266. AAAI Press, 2015.

[Ravi *et al.*, 1994] S. S. Ravi, Daniel J. Rosenkrantz, and Giri Kumar Tayi. Heuristic and special case algorithms for dispersion problems. *Operations Research*, 42(2):299–310, April 1994.

[Rossi and Ahmed, 2015] Ryan A. Rossi and Nesreen K. Ahmed. The network data repository with interactive graph analytics and visualization. In *AAAI*, 2015.

[Ruffini *et al.*, 2019] Manon Ruffini, Jelena Vucinic, Simon de Givry, George Katsirelos, Sophie Barbe, and Thomas Schiex. Guaranteed diversity & quality for the weighted CSP. In *ICTAI*, pages 18–25. IEEE, November 2019.

[Schittekat and Sörensen, 2009] Patrick Schittekat and Kenneth Sörensen. Supporting 3pl decisions in the automotive industry by generating diverse solutions to a large-scale location-routing problem. *Operations Research*, 57(5):1058–1067, 2009.

[Tohidi *et al.*, 2020] Ehsan Tohidi, Rouhollah Amiri, Mario Coutino, David Gesbert, Geert Leus, and Amin Karbasi. Submodularity in action: From machine learning to signal processing applications. *IEEE Signal Processing Magazine*, 37(5):120–133, September 2020.

[Trapp and Konrad, 2015] Andrew C. Trapp and Renata A. Konrad. Finding diverse optima and near-optima to binary integer programs. *IIE Transactions*, 47(11):1300–1312, April 2015.

[Vondrák, 2013] Jan Vondrák. Symmetry and approximability of submodular maximization problems. *SIAM Journal on Computing*, 42(1):265–304, January 2013.

[Wang and Kuo, 1988] D.W. Wang and Yue-Sun Kuo. A study on two geometric location problems. *Information Processing Letters*, 28(6):281–286, August 1988.

[Wang *et al.*, 2013] Jia Wang, James Cheng, and Ada Wai-Chee Fu. Redundancy-aware maximal cliques. In *Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining*, KDD '13, pages 122–130. ACM, August 2013.

[Xu and Li, 2006] Ke Xu and Wei Li. Many hard examples in exact phase transitions. *Theoretical Computer Science*, 355(3):291–302, April 2006.

[Yuan *et al.*, 2015] Long Yuan, Lu Qin, Xuemin Lin, Lijun Chang, and Wenjie Zhang. Diversified top-k clique search. *The VLDB Journal*, 25(2):171–196, October 2015.

[Zheng *et al.*, 2017] Kaiping Zheng, Hongzhi Wang, Zhixin Qi, Jianzhong Li, and Hong Gao. A survey of query result diversification. *Knowledge and Information Systems*, 51(1):1–36, September 2017.