# Fair Submodular Maximization over a Knapsack Constraint

**Lijun Li**[1] , **Chenyang Xu**[2] , **Liuyi Yang**[2] and **Ruilong Zhang**[3]

[1] Department of Computer Science, City University of Hong Kong, Hong Kong, China
[2] Software Engineering Institute, East China Normal University, Shanghai, China
[3] Department of Mathematics, Technical University of Munich, Munich, Germany
lijunli3-c@my.cityu.edu.hk, cyxu@sei.ecnu.edu.cn, 10225101401@stu.ecnu.edu.cn,
ruilong.zhang@tum.de

## Abstract

We consider fairness in submodular maximization subject to a knapsack constraint, a fundamental problem with various applications in economics, machine learning, and data mining. In the model, we are given a set of ground elements, each associated with a weight and a color, and a monotone submodular function defined over them. The goal is to maximize the submodular function while guaranteeing that the total weight does not exceed a specified budget (the knapsack constraint) and that the number of elements selected for each color falls within a designated range (the fairness constraint).

While there exists some recent literature on this topic, the existence of a non-trivial approximation for the problem – without relaxing either the knapsack or fairness constraints – remains a challenging open question. This paper makes progress in this direction. We demonstrate that when the number of colors is constant, there exists a polynomial-time algorithm that achieves a constant approximation with high probability. Additionally, we show that if either the knapsack or fairness constraint is relaxed only to require expected satisfaction, a tight approximation ratio of $(1-1/e-\epsilon)$ can be obtained in expectation for any $\epsilon > 0$.

## 1 Introduction

Submodular maximization is a fundamental problem in artificial intelligence and computer science, with continuous research since the 1970s [Cornuejols *et al.*, 1977; Nemhauser and Wolsey, 1978]. The problem involves selecting a subset of elements from a given set to maximize a submodular function defined over them. It captures a wide range of tasks across various domains, such as clustering [Anegg *et al.*, 2020; Backurs *et al.*, 2019; Chierichetti *et al.*, 2017], feature selection [Amiridi *et al.*, 2021; Bao *et al.*, 2022; Yu *et al.*, 2016], movie recommendation [Avdiukhin *et al.*, 2019; Dutting *et al.*, 2022; Haba *et al.*, 2020], and so on. However, recent studies [Celis *et al.*, 2018; Halabi *et al.*, 2023] have found that in some data mining applications, traditional submodular optimization algorithms may face a fairness issue.

The elements in the practical dataset often come from different groups (e.g., people of varying ages or genders [Halabi *et al.*, 2023; Halabi *et al.*, 2020; Wang *et al.*, 2021]), but traditional algorithms do not account for this factor, leading to an imbalance in the number of selected elements from each group.

To address this issue, [Celis *et al.*, 2018] introduces a *group fairness* notion into submodular maximization. Specifically, assuming that all given elements are partitioned into several groups, a solution is considered fair if the number of selected elements from each group falls within a specified range. One of their results is incorporating this group fairness notion into the classic cardinality-constrained submodular maximization problem, which aims to select at most $k$ elements that satisfy the fairness constraints and maximize the submodular objective. They employ a well-known *relax-and-round* framework [Calinescu *et al.*, 2011] in submodular maximization that first applies continuous greedy to obtain a fractional solution and then designs a randomized rounding procedure to produce an integral solution. They show that a tight approximation ratio of $(1-1/e)$ can be obtained in expectation by a relax-and-round approach.

In this paper, we explore a generalized version of the above model. Note that in many applications [Kempe *et al.*, 2003; Krause *et al.*, 2008; Lin and Bilmes, 2011], submodular maximization is often subject to a more general knapsack constraint rather than a simple cardinality limit, where each element has an associated weight, and the total weight of the selected elements must not exceed a specified budget. Therefore, we focus on optimizing fair solutions under a knapsack constraint. This general constraint introduces new challenges to fair submodular maximization. Selecting elements requires simultaneously balancing their weights, group memberships, and contributions to the submodular objective.

We notice that a recent study [Cui *et al.*, 2024] also considers this generalized model. They focus on the streaming scenario and demonstrate that if fairness constraints are allowed to be violated by a factor of $1/2$, a two-pass streaming algorithm can achieve a constant approximation. Furthermore, we find that directly extending the algorithm from [Celis *et al.*, 2018] to this general problem returns a $(1 - 1/e)$-approximation fair solution, but at the cost of violating the knapsack constraint. Nevertheless, to the best of our knowledge, whether a non-trivial approximation exists when both

fairness constraints and the knapsack constraint must be strictly satisfied remains an open problem.

## 1.1 Problem Definition and Challenges

The problem of *fair knapsack-constrained submodular maximization* (FKSM) is formally defined as follows. There is a ground element set $E = \{e_1, \ldots, e_n\}$ and a monotone submodular[1] function $f : 2^E \rightarrow \mathbb{R}$ defined over them. Each element $e \in E$ has a weight $w_e$ and an associated color $\delta_e$. Let group $G_i$ represent the set of elements of color $i$, and $\mathcal{G} = \{G_1, \ldots, G_k\}$ be the collection of all groups. Each group $G_i \in \mathcal{G}$ is associated with an interval $(l_i, u_i]$. The goal is to select an element subset $S \subseteq E$ to maximize the value of the function $f$, while ensuring that the total weight of the selected elements does not exceed a given budget $B$, i.e., $w(S) := \sum_{e \in S} w_e \leq B$, and that the number of selected elements from each group $G_i$ lies within the range $(l_i, u_i]$, i.e., $l_i < |S \cap G_i| \leq u_i$. We call the former the knapsack constraint and the latter the fairness constraints.

We refer to a special case of our model where the weight of each element is 1, i.e., the knapsack constraint simplifies to a cardinality constraint, as $\overline{\text{FKSM}}$. Similarly, we refer to another special case of our model where $l_i = -\infty$, i.e., there is no lower bound on the number of elements selected from each group, as $\underline{\text{FKSM}}$. These two special cases represent specific simplifications of the knapsack constraint and the fairness constraints, respectively. Both of them have been extensively studied in the literature [Celis *et al.*, 2018; Gu *et al.*, 2023] and can achieve a $(1-1/e-\epsilon)$-approximation for any $\epsilon > 0$ using the relax-and-round framework. The framework involves two steps: first, applying the continuous greedy algorithm to generate a fractional solution $\{x_e \in [0,1]\}_{e \in E}$ that satisfies the constraints and achieves an approximation ratio of $(1 - 1/e)$; and second, employing randomized pipage rounding to produce a feasible integral solution $\{y_e \in \{0, 1\}\}_{e \in E}$.

However, things become more challenging when both constraints—the knapsack and fairness constraints—are enforced without simplification, as prior methods are no longer applicable. While the first step of continuous greedy can still generate a fractional solution that meets the constraints, the second step of randomized pipage rounding encounters feasibility issues. The main complication with the knapsack constraint lies in the varying weights of the elements. Randomized pipage rounding can only guarantee that the number of selected elements remains unchanged (i.e., $\sum_e y_e = \sum_e x_e$), but cannot strictly enforce the total weight constraint. This limitation explains why the problem becomes simpler in $\overline{\text{FKSM}}$, where all element weights are uniform.

On the other hand, although randomized pipage rounding cannot strictly enforce the total weight constraint, it does ensure that the expected total weight satisfies the constraint: $\mathbb{E}[\sum_e w_e y_e] \leq B$ and the $y$ variables are negatively correlated. Using standard knapsack enumeration tech-

niques [Chekuri *et al.*, 2009], it can be shown that, with high probability (w.h.p.), $\sum_e w_e y_e \leq (1 + \epsilon)B$ for any $\epsilon > 0$. This result implies that we can scale the knapsack size down by a factor of $(1 + \epsilon)$ to $B' = B/(1 + \epsilon)$, run the algorithm on the scaled instance, and obtain an integral solution that w.h.p. satisfies the original knapsack constraint: $\sum_e w_e y_e \leq (1 + \epsilon)B' = B$. This knapsack scaling approach works well for $\overline{\text{FKSM}}$, as the simplified fairness constraints exhibit a down-closed property and slightly reducing the knapsack size only has a minor impact on the optimal objective. However, in the general FKSM problem, reducing the knapsack size even slightly may render the instance infeasible. For example, consider an instance where the knapsack capacity is just sufficient to satisfy the lower bounds in fairness constraints. Any further reduction in capacity would result in the absence of feasible solutions, highlighting a critical challenge in handling both knapsack and fairness constraints simultaneously in the general FKSM setting.

## 1.2 Our Techniques and Results

This work makes progress in finding a non-trivial approximation for FKSM. To overcome the challenges mentioned above, we propose a novel technique: *knapsack truncating* and combine it with the *randomized weighted pipage rounding*. The knapsack truncating technique can reduce an FKSM instance $\mathcal{I}$ to an $\overline{\text{FKSM}}$ instance $\bar{\mathcal{I}}$ (with the same ground elements and objective function) that satisfies the following two desirable properties:

- *Optimality Inheritance:* There exists an optimal solution of $\mathcal{I}$ that remains feasible for $\bar{\mathcal{I}}$.
- *Feasibility Extension:* For any feasible solution $\bar{S}$ of $\bar{\mathcal{I}}$, there always exists a feasible solution $S$ of $\mathcal{I}$ such that $f(S) \geq f(S')/2$.

When the number of groups is constant, the reduction can be performed in polynomial time. Therefore, leveraging this technique and the algorithm for the $\overline{\text{FKSM}}$ problem, we have the following:

**Theorem 1.1.** *Given an FKSM instance with a constant number of groups, there exists a polynomial-time algorithm that achieves an approximation ratio of $\frac{1}{2}\left(1 - \frac{1}{e}\right) - \epsilon$ with probability at least $1 - \frac{1}{e} - \frac{1}{e^2}$ for any $\epsilon > 0$.*

The other technique is a generalization of randomized pipage rounding, referred to as *randomized weighted rounding*. The randomized weighted rounding technique has already been applied to additive objective function cases in many previous works [Aziz *et al.*, 2024; Gandhi *et al.*, 2006]. During this process, the weights of the elements are taken into account to guide the rounding. This method ensures that the total weight remains unchanged before and after rounding. Additionally, for each element $e$, it guarantees that $\mathbb{E}[y_e] = x_e$, where $x_e$ represents the fractional solution, and $y_e$ denotes the rounded solution. We extend this technique to the *submodular* function case and prove that all properties of pipage rounding (e.g., negative correlation, objective concentration) still hold in randomized weighted pipage rounding (Section 4.3), which might be useful for other problems. By integrating this technique with the continuous greedy method, we have the following:

---

[1] A function $f : 2^E \rightarrow \mathbb{R}$ is submodular if for all $A, B \subseteq E$ we have $f(A) + f(B) \geq f(A \cap B) + f(A \cup B)$; or equivalently $f(A \cup e) - f(A) \geq f(B \cup e) - f(B)$ for all $A \subseteq B \subseteq E$ and $e \in E$. The function is monotone if $f(A) \leq f(B)$ for all $A \subseteq B$.

**Theorem 1.2.** *Given an FKSM instance, if the fairness constraints are relaxed to be satisfied in expectation, there exists a polynomial-time algorithm that returns a solution with an expected approximation ratio of $\left(1 - \frac{1}{e}\right) - \epsilon$ for any $\epsilon > 0$.*

The above theorem shows that when the fairness constraints are relaxed to require expected satisfaction, an approximation ratio of $1 - 1/e - \epsilon$ can be achieved in expectation. As mentioned earlier, the method from [Celis *et al.*, 2018] can be easily extended to FKSM. It guarantees a fair solution with an expected approximation ratio of $1 - 1/e - \epsilon$ while ensuring the knapsack constraint is satisfied in expectation (a detailed proof is provided in [Li *et al.*, 2025]). Therefore, we can conclude that if either the knapsack or fairness constraint is relaxed to require only expected satisfaction, a tight approximation ratio of $1 - 1/e - \epsilon$ can be achieved.

### 1.3 Other Related Works

Group fairness in submodular maximization has recently attracted considerable attention. Various research efforts have embraced the group fairness model, suggesting fair submodular maximization algorithms that operate under a range of constraints. These include the matroid constraint, applied in both streaming and offline settings [Halabi *et al.*, 2023; Halabi *et al.*, 2020; Halabi *et al.*, 2024]. Fair submodular maximization subject to a knapsack constraint is recently proposed by [Cui *et al.*, 2024] under the streaming setting. Without fairness, the problem just aims to maximize a monotone submodular function subject to a knapsack constraint, which admits a $(1-\frac{1}{e})$-approximation algorithm [Sviridenko, 2004]. [Kulik *et al.*, 2009] studies submodular maximization subject to a constant number of linear constraints and gives a $(1 - \frac{1}{e} - \epsilon)$ approximation algorithm. Further, if the fairness constraint only has an upper bound, then the problem becomes the submodular maximization subject to a knapsack and partition matroid constraint. A batch of works studies the intersection with $p$-matroid and $q$-knapsacks [Chekuri *et al.*, 2014; Feldman *et al.*, 2023; Gu *et al.*, 2023; Lee *et al.*, 2009; Mirzasoleiman *et al.*, 2016]. The current best algorithm is $\Omega(1/(p + q))$-approximation, which is given by [Gu *et al.*, 2023].

We also notice that a closely related work is the deterministic weighted pipage rounding proposed by [Ageev and Sviridenko, 2004] for the max-coverage problem with knapsack constraint. They show that the weighted pipage rounding algorithm returns a $(1 - \frac{1}{e} - \epsilon)$-approximate solution. However, since their rounding method is deterministic, it cannot be directly applied to ensure fairness in our problem. In our technique, a randomization procedure is introduced, which enables us to maintain the fairness constraint in expectation, and demonstrates that the produced random variables are well-concentrated.

### 1.4 Roadmap

Section 2 states some terminology and prior results. In Section 3, we introduce the knapsack truncating technique and present a constant approximation algorithm for FKSM strictly respecting knapsack and fairness constraints when the number of groups is constant. In Section 4, we consider the scenario where fairness constraints are allowed to be satisfied

in expectation and show that a constant approximation can be achieved by a randomized weighted pipage rounding approach. The paper finally concludes in Section 5.

## 2 Preliminaries

In this section, we introduce some terminology and prior results that will be used throughout this paper. A fractional vector $\mathbf{x} = \{x_e \in [0,1]\}_{e \in E}$ is called a fractional solution of an FKSM instance if $\mathbf{x}$ is a point of the *fair knapsack polytope* defined as follows.

**Definition 2.1** (Fair Knapsack Polytope). *A feasible fractional solution is a point in the following polytope $\mathcal{P}_{\mathrm{FK}}$:*

$$\left\{ \mathbf{x} \in [0,1]^E : \sum_{e \in E} w_e x_e \le B; l_i < \sum_{e \in G_i} x_e \le u_i, \forall i \in [k] \right\}$$

In the previous section, we mentioned that continuous greed can return a fractional solution with a constant approximation. However, the given submodular function $f$ cannot directly evaluate the objective value corresponding to a fractional solution. A common approach in submodular maximization is to utilize the concept of multi-linear extension to assess fractional solutions.

**Definition 2.2** (Multilinear Extension). *The multilinear extension of a submodular function $f$ is a function $F$ for $\mathbf{x} = \{x_e \in [0,1]\}_{e \in E}$ where $F(x)$ is equal to*

$$\mathbb{E}_{R \sim \mathcal{D}(\mathbf{x})}[f(R)] = \sum_{R \subseteq E} \left( f(R) \prod_{e \in R} x_e \prod_{e \notin R} (1 - x_e) \right)$$

*where $\mathcal{D}(\mathbf{x})$ represents a probability distribution over elements, where each element is sampled independently with probability $x_e$, and $R$ is a random subset sampled from this distribution.*

It is shown by [Patel *et al.*, 2021] that the problem admits a PTAS when the objective is an additive function (i.e., $f(S) = \sum_{e \in S} f(\{e\})$ for any $S \subseteq E$).

**Lemma 2.3** ([Patel *et al.*, 2021]). *When $f$ is an additive function, there exists an algorithm with running time $\mathrm{poly}(n, k, \frac{1}{\epsilon})$ that returns a solution $S$ such that (i) $\sum_{e \in S} f(\{e\}) \ge (1 - \epsilon) \cdot \mathrm{OPT}_A$; (ii) $l_i < |S \cap G_i| \le u_i$ for all $i \in [k]$, where $\mathrm{OPT}_A$ is the optimal solution under the additive function case.*

The prior work [Calinescu *et al.*, 2011] introduces a continuous greedy framework to address constrained submodular maximization problems. Their analysis shows that, given any constraints, if the corresponding additive version (i.e., replacing the submodular objective with an additive function) admits a $(1 - \epsilon)$-approximation algorithm, then combining that algorithm with continuous greedy yields a $(1 - 1/e - \epsilon)$-approximate fractional solution. Therefore, using Lemma 2.3 as a subroutine, it is shown that the continuous greedy algorithm returns a good fractional solution in polynomial time.

**Lemma 2.4.** *Given any instance of FKSM, there is a polynomial time algorithm that returns a point $x \in \mathcal{P}_{\mathrm{FK}}$ such that $F(x) \ge (1 - \frac{1}{e} - \epsilon) \cdot \mathrm{OPT}$ for any $\epsilon > 0$, where $\mathrm{OPT}$ is the optimal objective value.*

# 3 Knapsack Truncating

In this section, we introduce the knapsack truncating technique and use it to prove the following theorem:

**Theorem 1.1.** *Given an FKSM instance with a constant number of groups, there exists a polynomial-time algorithm that achieves an approximation ratio of $\frac{1}{2}\left(1-\frac{1}{e}\right)-\epsilon$ with probability at least $1-\frac{1}{e}-\frac{1}{e^2}$ for any $\epsilon > 0$.*

Given an FKSM instance, let $S^*$ denote an optimal solution for this instance. Since this section focuses on the setting where the number of element groups is constant, through polynomial-time enumeration, we can assume that we know the following for each group $G_i$:

- The number of elements selected from $G_i$ by $S^*$, i.e., the value of $\gamma_i := |S^* \cap G_i|$.

- The number of elements among the top $\gamma_i$ smallest-weight elements in $G_i$ that are included in $S^*$, i.e., the value of $\beta_i := |S^* \cap L_i(\gamma_i)|$, where $L_i(\gamma_i)$ represents the set of the $\gamma_i$ smallest-weight elements in $G_i$ (with ties broken arbitrarily).

The reduction method is stated in Algorithm 1. From the description, we observe that during the reduction, both the cost of each element and the knapsack size are reduced. For elements in $L_i(\gamma_i)$, their costs are set to 0. For each element not in $L_i(\gamma_i)$, the cost decreases by $\frac{w(L_i(\gamma_i))-w(L_i(\beta_i))}{\gamma_i-\beta_i}$. This reduction amount can be interpreted as the average weight of the $\gamma_i - \beta_i$ largest elements in the subset $L_i(\gamma_i)$. Since $L_i(\gamma_i)$ is defined as the set of the $\gamma_i$ elements in $G_i$ with the smallest weights, we always have each element's new weight in the reduced $\bar{\text{F}}$KSM instance $\bar{w}_e \geq 0$.

**Algorithmic Intuition.** The purpose of knapsack truncating is to transform an FSMK instance into an $\bar{\text{F}}$KSM instance such that it can be addressed efficiently. A very natural idea is to directly set all fairness lower bounds to zero. Clearly, in this case, the *optimality inheritance* property (see its definition in Section 1.2 or the lemma below) would always hold. However, the *feasibility extension* property is unlikely to be satisfied, as it is possible that some weight-heavy elements have already been selected, leaving no room to add more elements to meet the fairness lower bounds.

But we notice that things change if each group contains at least $\gamma_i$ elements with a weight of 0. In this scenario, the fairness lower bounds are guaranteed to be met because these zero-weight elements can be selected freely. This suggests that to ensure feasibility extension, we must introduce zero weights into the $\bar{\text{F}}$KSM instance—some elements' weights need to be reduced to 0. Intuitively, reducing the weights of elements in $L_i(\gamma_i)$ to zero should have the least impact (as $L_i(\gamma_i)$ is the set of elements with the smallest weights). Therefore, in our algorithm, we choose to let these elements become zero-weight. Once their weights are set to 0, the knapsack budget should be reduced accordingly by $\sum_i w(L_i(\gamma_i))$. This adjustment can be seen as reserving space in the knapsack for elements in $L_i(\gamma_i)$, allowing them to be freely selected later.

However, reducing the budget harms the optimality inheritance property, as the optimal solution may contain many

elements outside $L_i(\gamma_i)$. To ensure that the optimal solution from the original instance still remains feasible under the new budget, we also need to reduce the weights of elements not in $L_i(\gamma_i)$. Since FKSM and $\bar{\text{F}}$KSM are not equivalent, we cannot guarantee that all feasible solutions will satisfy the new budget constraint after weight reduction. Nevertheless, based on the characterization of the optimal solution provided by $\{\gamma_i, \beta_i\}$, we can carefully design the weight reduction strategy to ensure that the optimal solution remains feasible in the new $\bar{\text{F}}$KSM instance.

**Lemma 3.1.** *Given an FKSM instance $\mathcal{I}$, Algorithm 1 reduces it to an $\bar{\text{F}}$KSM instance $\bar{\mathcal{I}}$ (with the same ground elements and objective function) that satisfies the following:*

*(3.1a)* *Optimality Inheritance: The optimal solution $S^*$ of $\mathcal{I}$ remains feasible for $\bar{\mathcal{I}}$.*

*(3.1b)* *Feasibility Extension: For any feasible solution $\bar{S}$ of $\bar{\mathcal{I}}$, there always exists a feasible solution $S$ of $\mathcal{I}$ such that $f(S) \geq f(\bar{S})/2$.*

*Proof.* We begin by proving the first property. We show that $S^*$ satisfies both the fairness and budget constraints of the reduced $\bar{\text{F}}$KSM instance $\bar{\mathcal{I}}$.

For the fairness constraints, according to the description in Algorithm 1, each group $G_i$ in the original instance $\mathcal{I}$ is split into two groups: $L_i(\gamma_i)$ and $G_i \setminus L_i(\gamma_i)$, with corresponding fairness upper bounds of $\beta_i$ and $\gamma_i - \beta_i$, respectively. Due to the definitions of $\gamma_i$ and $\beta_i$, we directly have $|S^* \cap L_i(\gamma_i)| = \beta_i$ and $|S^* \cap (G_i \setminus L_i(\gamma_i))| = \gamma_i - \beta_i$. Therefore, $S^*$ satisfies the fairness constraints of $\bar{\mathcal{I}}$.

For the budget constraint, we have

$$\bar{w}(S^*) = \sum_{i \in [k]} \bar{w}(S^* \cap L_i(\gamma_i)) + \bar{w}(S^* \cap (G_i \setminus L_i(\gamma_i)))$$

$$= \sum_{i \in [k]} w(S^* \cap L_i(\gamma_i)) - w(S^* \cap L_i(\gamma_i))$$

$$+ w(S^* \cap (G_i \setminus L_i(\gamma_i)))$$

$$- (\gamma_i - \beta_i) \cdot \frac{w(L_i(\gamma_i)) - w(L_i(\beta_i))}{\gamma_i - \beta_i}$$
$$\text{(Definitions of } \bar{w}_e, \gamma_i \text{ and } \beta_i\text{)}$$

$$\leq B - \sum_{i \in [k]} w(L_i(\gamma_i))$$

$$+ \sum_{i \in [k]} w(L_i(\beta_i)) - w(S^* \cap L_i(\gamma_i))$$
$$\text{(Feasibility of } S^* \text{ for } \mathcal{I}\text{)}$$

$$\leq \bar{B},$$

where the last inequality follows from the definition of $\bar{B}$, the fact that $L_i(\beta_i)$ is the top $\beta_i$ smallest-weight elements in $G_i$, and that $|S^* \cap L_i(\gamma_i)| = \beta_i$. Thus, $S^*$ remains feasible for $\bar{\mathcal{I}}$.

For the second property, given a feasible solution $\bar{S}$ for $\bar{\mathcal{I}}$, we construct a solution following the procedure described in Algorithm 2. We first prove the feasibility and then analyze its objective value. The algorithm ultimately selects the solution with the larger objective value between $S$ and $L(\gamma)$. Since $L(\gamma)$ is clearly feasible, the following focuses on proving the feasibility of $S$. By the algorithm's construction, $S$

---

**Algorithm 1** Knapsack Truncating

---

**Input:** An FKSM instance characterized by $\mathcal{I} = (f, \{w_e, \delta_e\}_{e \in E}, \{l_i, u_i\}_{i \in [k]}, B)$ and a corresponding parameter set $\{\gamma_i, \beta_i\}_{i \in [k]}$.

**Output:** A reduced $\overline{\text{F}}$KSM instance.

    // Split each group into two groups according to parameter $\gamma_i, \beta_i$ in the reduced instance

1: **for** each group index $i \in [k]$ **do**
2:     **for** each element $e \in G_i$ **do**
3:         **if** $e \notin L_i(\gamma_i)$ **then**
4:             Set its new color $\bar{\delta}_e \leftarrow i + k$ and weight $\bar{w}_e \leftarrow w_e - \frac{w(L_i(\gamma_i)) - w(L_i(\beta_i))}{\gamma_i - \beta_i}$.
5:         **else**
6:             Keep its color $\bar{\delta}_e \leftarrow i$ unchanged and set its new weight $\bar{w}_e \leftarrow 0$.
7:         **end if**
8:     **end for**
9: **end for**
    // Construct the upper limit for each new group
10: **for** each new group index $i \in [2k]$ **do**
11:     **if** $i \leq k$ **then**
12:         Set $\bar{u}_i \leftarrow \beta_i$.
13:     **else**
14:         Set $\bar{u}_i \leftarrow \gamma_i - \beta_i$.
15:     **end if**
16: **end for**
    // Construct the new budget
17: Set the new budget: $\bar{B} \leftarrow B - \sum_{i \in [k]} w(L_i(\gamma_i))$
18: **return** $\bar{\mathcal{I}} = (f, \{\bar{w}_e, \bar{\delta}_e\}_{e \in E}, \{\bar{u}_i\}_{i \in [2k]}, \bar{B})$.

---

**Algorithm 2** Feasibility Extension

---

**Input:** A feasible solution $\bar{S}$ of the reduced $\overline{\text{F}}$KSM instance.

**Output:** A feasible solution $S$ of the original FKSM instance.

1: **for** each group index $i \in [k]$ **do**
2:     Set $S \leftarrow \bar{S} \cap (G_i \setminus L_i(\gamma_i))$.
3:     Add elements from $L_i(\gamma_i)$ to $S$ in increasing order of weight until $|S \cap G_i| = \gamma_i$.
4: **end for**
5: Let $L(\gamma) := \bigcup_{i \in [k]} L_i(\gamma_i)$
6: **return** $\arg\max\{f(S), f(L(\gamma))\}$.

---

satisfies the fairness constraints in $\mathcal{I}$, as elements are added to $S$ until $|S \cap G_i| = \gamma_i$ for each $i \in [k]$. The proof of the budget constraint relies on the fact that elements are added in increasing order of weight (in Line 3 of Algorithm 2). Intuitively, this ensures that the average weight among any subset of these elements is at most $\frac{w(L_i(\gamma_i)) - w(L_i(\beta_i))}{\gamma_i - \beta_i}$. For notational simplicity, let $\kappa_i := |\bar{S} \cap (G_i \setminus L_i(\gamma_i))|$ for each $i \in [k]$, and due to Line 3 in Algorithm 2, we have $|S \cap L_i(\gamma_i)| = \gamma_i - \kappa_i$. Thus, the value of $w(S)$ is equal to

$$\sum_{i \in [k]} w(\bar{S} \cap (G_i \setminus L_i(\gamma_i))) + w(S \cap L_i(\gamma_i))$$
$$= \sum_{i \in [k]} \bar{w}(\bar{S} \cap (G_i \setminus L_i(\gamma_i))) + w(S \cap L_i(\gamma_i))$$

$$+ \kappa_i \cdot \frac{w(L_i(\gamma_i)) - w(L_i(\beta_i))}{\gamma_i - \beta_i}$$
$$\leq B + \sum_{i \in [k]} -w(L_i(\gamma_i)) + w(S \cap L_i(\gamma_i))$$
$$+ \kappa_i \cdot \frac{w(L_i(\gamma_i)) - w(L_i(\beta_i))}{\gamma_i - \beta_i}$$
$$\text{(Feasibility of } \bar{S} \text{ for } \bar{\mathcal{I}})$$
$$\leq B + \sum_{i \in [k]} -w(L_i(\gamma_i)) + w(L_i(\beta_i))$$
$$+ (\gamma_i - \kappa_i - \beta_i) \cdot \frac{w(L_i(\gamma_i)) - w(L_i(\beta_i))}{\gamma_i - \beta_i}$$
$$+ \kappa_i \cdot \frac{w(L_i(\gamma_i)) - w(L_i(\beta_i))}{\gamma_i - \beta_i}$$
$$= B,$$

where the last inequality uses the fact that, by the algorithm's construction, $S \cap L(\gamma_i)$ consists of the $\gamma_i - \kappa_i$ smallest-weight elements in $L(\gamma_i)$.

To prove the objective value guarantee, we observe that $\bar{S}$ can be partitioned into two parts: $\bar{S} \cap L(\gamma)$ and $\bar{S} \setminus L(\gamma)$, which are subsets of $L(\gamma)$ and $S$, respectively. Since $f$ is monotone and submodular, we can complete the proof:

$$f(\bar{S}) \leq f(\bar{S} \cap L(\gamma)) + f(\bar{S} \setminus L(\gamma)) \quad \text{(Submodularity)}$$
$$\leq f(L(\gamma)) + f(S) \quad \text{(Monotonicity)}$$
$$\leq 2 \cdot \max\{f(L(\gamma)), f(S)\}.$$

$\square$

Lemma 3.1 demonstrates a desirable reduction from FKSM to $\overline{\text{FKSM}}$. The last piece in proving Theorem 1.1 is to use an existing algorithm to handle $\overline{\text{FKSM}}$.

**Lemma 3.2** ([Chekuri *et al.*, 2009]). *For the $\overline{\text{FKSM}}$ problem, there exists a polynomial time algorithm that returns an approximation ratio of $1 - \frac{1}{e} - \epsilon$ with probability at least $1 - \frac{1}{e} - \frac{1}{e^2}$ for any $\epsilon > 0$.*

*Proof of Theorem 1.1.* The theorem can be easily proven using the above lemmas. We first reduce the given FKSM instance using Algorithm 1, then apply the algorithm provided in [Chekuri *et al.*, 2009] to the reduced $\overline{\text{FKSM}}$ instance. Finally, we use Algorithm 2 to produce a feasible solution for the original problem. Lemma 3.2 ensures that we can obtain a solution with an approximation ratio of $1 - \frac{1}{e} - \epsilon$ with high probability in the reduced instance, while the two properties from Lemma 3.1 guarantee that the solution constructed by Algorithm 2 will have at most a $1/2$ decrease in its approximation ratio, which completes the proof. $\qquad\square$

## 4 Randomized Weighted Pipage Rounding

In this section, we apply randomized weighted pipage rounding to the submodular function case and aim to establish the following theorem.

**Theorem 1.2.** *Given an FKSM instance, if the fairness constraints are relaxed to be satisfied in expectation, there exists a polynomial-time algorithm that returns a solution with an expected approximation ratio of $\left(1 - \frac{1}{e}\right) - \epsilon$ for any $\epsilon > 0$.*

---

**Algorithm 3** Randomized Weighted Pipage Rounding

**Input:** An instance with $(f, \{w_e, \delta_e\}_{e \in E}, \{l_i, u_i\}_{i \in [k]}, B)$ and a fractional solution $\mathbf{x}$
**Output:** A rounded solution $\mathbf{y}$.
1: Initialize solution $\mathbf{y} \leftarrow \mathbf{x}$.
2: **while** there are two elements $p, q$ with $y_p, y_q \in (0, 1)$ **do**
3:      Let $\delta_1 \leftarrow \min\{y_p, (1 - y_q) \cdot \frac{w_q}{w_p}\}$.
4:      Let $\delta_2 \leftarrow \min\{1 - y_p, y_q \cdot \frac{w_q}{w_p}\}$.
5:      With probability $\frac{\delta_2}{\delta_1 + \delta_2}$:
         Set $y_p \leftarrow y_p - \delta_1$ and $y_q \leftarrow y_q + \delta_1 \cdot \frac{w_p}{w_q}$.
6:      Otherwise:
         Set $y_p \leftarrow y_p + \delta_2$ and $y_q \leftarrow y_q - \delta_2 \cdot \frac{w_p}{w_q}$.
7: **end while**
8: **return** $\mathbf{y}$.

---

The description of the generalized rounding algorithm is provided in Algorithm 3. Note that in the classic pipage rounding [Calinescu *et al.*, 2011] for cardinality constraint, $\delta_1$ and $\delta_2$ are set to $\min\{y_p, (1 - y_q)\}$ and $\min\{1 - y_p, y_q\}$, respectively. And, the coordinates $y_p$ and $y_q$ always change at the same rate. However, in the weighted version, we utilize the element weights to guide the rounding process, ensuring that the knapsack constraint is always satisfied. We begin by showing several properties of weighted pipage rounding.

### 4.1 Rounding Properties

In this subsection, we discuss the properties of weighted pipage rounding from three perspectives: the knapsack constraint (Lemma 4.1), the fairness constraints (Lemma 4.2), and the objective guarantee (Lemma 4.3).

**Lemma 4.1** (Knapsack Satisfaction). $\sum_{e \in E} w_e y_e \leq B$.

**Lemma 4.2** (Expected Fairness Satisfaction). *For each group $G_i$, $l_i < \mathbb{E}[\sum_{e \in G_i} y_e] \leq u_i$.*

**Lemma 4.3** (Objective Guarantee). *In any iteration, we have $\mathbb{E}[F(\mathbf{y})] \geq F(\mathbf{x})$.*

The first two lemmas can be easily proven by performing some mathematical calculations based on the probability distribution defined in Algorithm 3. Due to space limitations, we defer their detailed proofs to [Li *et al.*, 2025].

In the following, we focus on proving the third lemma. During the proof, we build on an existing result (Lemma 4.4) to generalize the convexity of a submodular function's multilinear extension (Lemma 4.5) and then use this convexity property to complete the proof.

**Lemma 4.4** ([Calinescu *et al.*, 2011]). *Let $F : [0, 1]^E \to \mathbb{R}_{\geq 0}$ be the multilinear extension of a set function $f : 2^E \to \mathbb{R}_{\geq 0}$.*

*(4.4a)* $\frac{\partial^2 F}{\partial x_i^2} = 0$ *holds at any point in $[0, 1]^E$ for any $i \in E$.*

*(4.4b)* *If $f$ is submodular, then $\frac{\partial^2 F}{\partial x_i \partial x_j} \leq 0$ holds at any point in $[0, 1]^E$ for all $i, j \in E$*

**Lemma 4.5** (Convexity of Multilinear Extension). *Let $\mathbf{e}_i$ denote the vector with a value of 1 in the $i$-th dimension and 0 in all other dimensions. If $f$ is a monotone submodular function, then its multilinear extension is convex along the line $(c_i \mathbf{e}_i - c_j \mathbf{e}_j)$ for any $i, j \in E$ and constants $c_i, c_j \geq 0$, i.e., the function $F_{ij}^{\mathbf{a}}(\lambda) := F(\mathbf{a} + c_i \lambda \mathbf{e}_i - c_j \lambda \mathbf{e}_j)$ is a convex function for any $\mathbf{a} \in [0, 1]^E$, $i, j \in E$ and constants $c_i, c_j \geq 0$.*

*Proof.* To show the function $F_{ij}^{\mathbf{a}}(\cdot)$ is a convex function, it is sufficient to show the second derivative of $F_{ij}^{\mathbf{a}}$ is nonnegative. We define the function $\mathbf{g}(\lambda) := \mathbf{a} + c_i \lambda \mathbf{e}_i - c_j \lambda \mathbf{e}_j$. Note that $\mathbf{g}(\lambda)$ is a set of functions. For each $e \in E$, define $g_e(\lambda)$ as the $e$-th function of $\mathbf{g}(\lambda)$, i.e., $g_e(\lambda) = a_e$ for all $e \in E \setminus \{i, j\}$, $g_i(\lambda) = a_i + c_i \lambda$ and $g_j(\lambda) = a_j - c_j \lambda$. So, $F_{ij}^{\mathbf{a}}$ is a composition function: $F_{ij}^{\mathbf{a}} = F(g_1(\lambda), \ldots, g_n(\lambda))$. For simplicity, let $\Delta := \mathbf{a} + c_i \lambda \mathbf{e}_i - c_j \lambda \mathbf{e}_j$. By the chain rule,

$$\frac{\mathrm{d} F_{ij}^{\mathbf{a}}}{\partial \lambda} = \sum_{e=1}^{n} \frac{\partial F}{\partial g_i} \cdot \frac{\partial g_i}{\partial \lambda} = \left( c_i \frac{\partial F}{\partial x_i} - c_j \frac{\partial F}{\partial x_j} \right) \Big|_{\Delta}.$$

Differentiating one more time and by the chain rule again,

$$\frac{\mathrm{d}^2 F_{ij}^{\mathbf{a}}}{\partial \lambda^2} = \left( c_i^2 \frac{\partial^2 F}{\partial x_i^2} - 2 c_i c_j \frac{\partial^2 F}{\partial x_i \partial x_j} + c_j^2 \frac{\partial^2 F}{\partial x_j^2} \right) \Big|_{\Delta}.$$

Since $c_i, c_j \geq 0$ and by Lemma 4.4, we get $\frac{\mathrm{d}^2 F_{ij}^{\mathbf{a}}}{\partial \lambda^2} \geq 0$. Thus, $F(\mathbf{a} + c_i \lambda \mathbf{e}_i - c_j \lambda \mathbf{e}_j)$ is a convex function of $\lambda$. $\qquad\square$

*Proof of Lemma 4.3.* Let $\mathbf{y}^{(t)}$ denote the solution $\mathbf{y}$ in iteration $t$. We prove the lemma inductively by showing that $\mathbb{E}[F(\mathbf{y}^{(t+1)}) \mid \mathbf{y}^{(t)}] \geq F(\mathbf{y}^{(t)})$. Assume that $\mathbf{y}^{(t+1)}$ and $\mathbf{y}^{(t)}$ differ at $p, q$-th coordinates. Let $\mathbf{e}_q$ denote the vector with a value of 1 in the $q$-th dimension and 0 in all other dimensions. By the description of Algorithm 3, $\mathbf{y}^{(t+1)}$ becomes $\mathbf{y}^{(t)} - \delta_1 \cdot \mathbf{e}_p + \delta_1 \cdot \frac{w_q}{w_p} \cdot \mathbf{e}_q$ with probability $\frac{\delta_2}{\delta_1 + \delta_2}$, and $\mathbf{y}^{(t)} + \delta_2 \cdot \mathbf{e}_p - \delta_2 \cdot \frac{w_q}{w_p} \cdot \mathbf{e}_q$ with probability $\frac{\delta_1}{\delta_1 + \delta_2}$. Define function $H(\lambda) := F(\mathbf{y}^{(t)} + \lambda \cdot \mathbf{e}_p - \frac{w_p}{w_q} \cdot \lambda \cdot \mathbf{e}_q)$. Then,

$$
\begin{aligned}
&\mathbb{E}[F(\mathbf{y}^{(t+1)}) \mid \mathbf{y}^{(t)}] \\
&= \frac{\delta_2}{\delta_1 + \delta_2} \cdot H(-\delta_1) + \frac{\delta_1}{\delta_1 + \delta_2} \cdot H(\delta_2) \\
&\geq H\left(-\frac{\delta_2 \cdot \delta_1}{\delta_1 + \delta_2} + \frac{\delta_1 \cdot \delta_2}{\delta_1 + \delta_2}\right) \quad \text{(Lemma 4.5)} \\
&= H(0) = F(\mathbf{y}^{(t)}) \, .
\end{aligned}
$$

$\square$

## 4.2 Constant Approximation with Relaxed Fairness

This subsection proves Theorem 1.2. In Lemma 2.4, we demonstrate that the continuous greedy technique can always yield a fractional solution $\mathbf{x}$ with an approximation ratio of $(1 - 1/e - \epsilon)$. Subsequently, we apply Algorithm 3 to round $\mathbf{x}$ into a solution $\mathbf{y}$. Using Lemma 4.1, Lemma 4.2 and Lemma 4.3, we can guarantee that $\mathbf{y}$ satisfies the knapsack constraint, (expected) fairness constraints, and maintains the objective value.

However, it is worth noting that the proof does not conclude here. The solution $\mathbf{y}$ is not guaranteed to be integral. Due to the different rates at which $y_e$ increases or decreases during the rounding process, $\sum_e y_e$ may no longer be an integer. In fact, $\mathbf{y}$ is a *nearly* integral solution, where "nearly" means that at most one element has a fractional $y_e \in (0, 1)$, while the remaining components are integers.

**Lemma 4.6.** *The solution $\mathbf{y}$ returned by Algorithm 3 is a nearly integral solution. Let $\mathbf{z} := \{z_e = \lfloor y_e \rfloor\}_{e \in E}$. We have integral solution $\mathbf{z}$ satisfies the knapsack constraint and (expected) fairness constraint.*

*Proof.* It is easy to see that $\mathbf{y}$ is a nearly integral solution because if there were at least two fractional components, Algorithm 3 would continue iterating in the while loop. Since $z_e$ is the floor of each $y_e$, it follows that $z_e \leq y_e$. By Lemma 4.1, we can verify the satisfaction of the knapsack constraint: $\sum_{e \in E} w_e z_e \leq \sum_{e \in E} w_e y_e \leq B$.

For the fairness constraints, the continuous greedy process ensures that the $\ell_1$ norm of the fractional solution within any group is an integer, i.e., $\sum_{e \in G_i} x_e \geq l_i + 1$. By Lemma 4.2, the expected $\ell_1$ norm of $\mathbf{y}$ also satisfies $\mathbb{E}[\sum_{e \in G_i} y_e] \in [l_i + 1, u_i]$. Furthermore, since $\mathbf{y}$ is nearly integral, the rounding process decreases the $\ell_1$ norm by less than 1. Hence, for each group $G_i$, we have:

$$
\sum_{e \in G_i} y_e - 1 < \sum_{e \in G_i} z_e \leq \sum_{e \in G_i} y_e \Rightarrow \mathbb{E}\left[\sum_{e \in G_i} z_e\right] \in (l_i, u_i].
$$

$\square$

*Proof of Theorem 1.2.* Using the above lemmas, we obtain an integer solution $\mathbf{z}$ that satisfies the knapsack constraint and the expected fairness constraints. The last step is to analyze the objective value of $\mathbf{z}$. According to the knapsack enumeration technique applied in [Chekuri *et al.*, 2009], we can always ensure, through a polynomial-time enumeration process, that all elements selected via the relax-and-round method contribute only a small constant $\eta$ to the objective function value. Namely, removing any single element reduces the objective value by at most a factor of $(1 + \eta)$. Therefore, $F(\mathbf{z})$ differs from $F(\mathbf{y})$ by at most a factor of $(1 + \eta)$. Furthermore, Lemma 4.3 implies that the expected approximation ratio of $\mathbf{z}$ is $1 - 1/e - \epsilon$. $\square$

## 4.3 Further Discussion

In this subsection, we discuss some additional properties of the randomized weighted pipage rounding, which we believe to be of independent interest. Notably, we observe that the *negative correlation* and *objective concentration* properties, previously established in the classic pipage rounding, still hold in the weighted version.

**Lemma 4.7** (Negative Correlation)**.** *The random variables in $\{y_e\}_{e \in E}$ are negatively correlated, which means that for any $S \subseteq E$: $\mathbb{E}\left[\prod_{e \in S} y_e\right] \leq \prod_{e \in S} \mathbb{E}[y_e]$.*

**Lemma 4.8** (Objective Concentration)**.** *If the objective function $f$ has a marginal value at most 1, for any $\delta \in (0, 1)$, we have $\Pr[F(\mathbf{y}) \leq (1 - \delta)F(\mathbf{x})] \leq \exp(-\delta^2 \cdot F(\mathbf{x})/2)$.*

We remark that when generalized to the weighted pipage rounding, many prior properties still hold, but the original proofs are no longer applicable. Since the algorithm now moves the fractional solution along the line $\mathbf{e}_p - \mathbf{e}_q \cdot \frac{w_p}{w_q}$ rather than the specific direction $\mathbf{e}_p - \mathbf{e}_q$, all relevant proofs need to be extended. For example, when proving the objective concentration lemma, it is necessary to establish a more general property of the concave pessimistic estimator (see [Li *et al.*, 2025] for details).

## 5 Conclusion

In this paper, we consider the problem of fair submodular maximization under a knapsack constraint. We introduce two novel techniques: knapsack truncating and randomized weighted pipage rounding, and apply them to derive fair solutions with good approximations. Several directions for future research remain open. For example, when the number of groups is non-constant, it still remains an open question if a non-trivial approximation can be achieved. Investigating whether our proposed techniques can be further extended to this more general setting, or if entirely new techniques are required to resolve this open problem, is an interesting direction for future work.

## Acknowledgements

## Contribution Statement

All authors in the paper have equal contributions and are corresponding authors, and thus, it is ordered alphabetically.

## References

[Ageev and Sviridenko, 2004] Alexander A. Ageev and Maxim Sviridenko. Pipage rounding: A new method of constructing algorithms with proven performance guarantee. *J. Comb. Optim.*, 8(3):307–328, 2004.

[Amiridi *et al.*, 2021] Magda Amiridi, Nikos Kargas, and Nicholas D. Sidiropoulos. Information-theoretic feature selection via tensor decomposition and submodularity. *IEEE Trans. Signal Process.*, 69:6195–6205, 2021.

[Anegg *et al.*, 2020] Georg Anegg, Haris Angelidakis, Adam Kurpisz, and Rico Zenklusen. A technique for obtaining true approximations for k-center with covering constraints. In *IPCO*, volume 12125 of *Lecture Notes in Computer Science*, pages 52–65. Springer, 2020.

[Avdiukhin *et al.*, 2019] Dmitrii Avdiukhin, Slobodan Mitrovic, Grigory Yaroslavtsev, and Samson Zhou. Adversarially robust submodular maximization under knapsack constraints. In *KDD*, pages 148–156. ACM, 2019.

[Aziz *et al.*, 2024] Haris Aziz, Xinhang Lu, Mashbat Suzuki, Jeremy Vollen, and Toby Walsh. Fair lotteries for participatory budgeting. In *AAAI*, pages 9469–9476. AAAI Press, 2024.

[Backurs *et al.*, 2019] Arturs Backurs, Piotr Indyk, Krzysztof Onak, Baruch Schieber, Ali Vakilian, and Tal Wagner. Scalable fair clustering. In *ICML*, volume 97 of *Proceedings of Machine Learning Research*, pages 405–413. PMLR, 2019.

[Bao *et al.*, 2022] Wei-Xuan Bao, Jun-Yi Hang, and Min-Ling Zhang. Submodular feature selection for partial label learning. In *KDD*, pages 26–34. ACM, 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 J. Comput.*, 40(6):1740–1766, 2011.

[Celis *et al.*, 2018] L. Elisa Celis, Lingxiao Huang, and Nisheeth K. Vishnoi. Multiwinner voting with fairness constraints. In *IJCAI*, pages 144–151. ijcai.org, 2018.

[Chekuri *et al.*, 2009] Chandra Chekuri, Jan Vondrak, and Rico Zenklusen. Dependent randomized rounding for matroid polytopes and applications, 2009.

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

[Chierichetti *et al.*, 2017] Flavio Chierichetti, Ravi Kumar, Silvio Lattanzi, and Sergei Vassilvitskii. Fair clustering through fairlets. In *NIPS*, pages 5029–5037, 2017.

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

[Cui *et al.*, 2024] Shuang Cui, Kai Han, Shaojie Tang, Feng Li, and Jun Luo. Fairness in streaming submodular maximization subject to a knapsack constraint. In *KDD*, pages 514–525. ACM, 2024.

[Dutting *et al.*, 2022] Paul Dutting, Federico Fusco, Silvio Lattanzi, Ashkan Norouzi-Fard, and Morteza Zadimoghaddam. Deletion robust submodular maximization over matroids. In *ICML*, volume 162 of *Proceedings of Machine Learning Research*, pages 5671–5693. PMLR, 2022.

[Feldman *et al.*, 2023] Moran Feldman, Christopher Harshaw, and Amin Karbasi. How do you want your greedy: Simultaneous or repeated? *J. Mach. Learn. Res.*, 24:72:1–72:87, 2023.

[Gandhi *et al.*, 2006] Rajiv Gandhi, Samir Khuller, Srinivasan Parthasarathy, and Aravind Srinivasan. Dependent rounding and its applications to approximation algorithms. *J. ACM*, 53(3):324–360, 2006.

[Gu *et al.*, 2023] Yu-Ran Gu, Chao Bian, and Chao Qian. Submodular maximization under the intersection of matroid and knapsack constraints. In *AAAI*, pages 3959–3967. AAAI Press, 2023.

[Haba *et al.*, 2020] Ran Haba, Ehsan Kazemi, Moran Feldman, and Amin Karbasi. Streaming submodular maximization under a k-set system constraint. In *ICML*, volume 119 of *Proceedings of Machine Learning Research*, pages 3939–3949. PMLR, 2020.

[Halabi *et al.*, 2020] Marwa El Halabi, Slobodan Mitrovic, Ashkan Norouzi-Fard, Jakab Tardos, and Jakub Tarnawski. Fairness in streaming submodular maximization: Algorithms and hardness. In *NeurIPS*, 2020.

[Halabi *et al.*, 2023] Marwa El Halabi, Federico Fusco, Ashkan Norouzi-Fard, Jakab Tardos, and Jakub Tarnawski. Fairness in streaming submodular maximization over a matroid constraint. In *ICML*, volume 202 of *Proceedings of Machine Learning Research*, pages 9150–9171. PMLR, 2023.

[Halabi *et al.*, 2024] Marwa El Halabi, Jakub Tarnawski, Ashkan Norouzi-Fard, and Thuy-Duong Vuong. Fairness in submodular maximization over a matroid constraint. In *AISTATS*, volume 238 of *Proceedings of Machine Learning Research*, pages 1027–1035. PMLR, 2024.

[Kempe *et al.*, 2003] David Kempe, Jon Kleinberg, and Éva Tardos. Maximizing the spread of influence through a social network. In *Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 137–146, 2003.

[Krause *et al.*, 2008] Andreas Krause, Ajit Singh, and Carlos Guestrin. Near-optimal sensor placements in gaussian processes: Theory, efficient algorithms and empirical studies. *Journal of Machine Learning Research*, 9(2), 2008.

[Kulik *et al.*, 2009] Ariel Kulik, Hadas Shachnai, and Tami Tamir. Maximizing submodular set functions subject to multiple linear constraints. In *SODA*, pages 545–554. SIAM, 2009.

[Lee *et al.*, 2009] Jon Lee, Vahab S. Mirrokni, Viswanath Nagarajan, and Maxim Sviridenko. Non-monotone submodular maximization under matroid and knapsack constraints. In *STOC*, pages 323–332. ACM, 2009.

[Li *et al.*, 2025] Lijun Li, Chenyang Xu, Liuyi Yang, and Ruilong Zhang. Fair submodular maximization over a knapsack constraint. *arXiv*, abs/2505.12126, 2025.

[Lin and Bilmes, 2011] Hui Lin and Jeff Bilmes. A class of submodular functions for document summarization. In Dekang Lin, Yuji Matsumoto, and Rada Mihalcea, editors, *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 510–520, Portland, Oregon, USA, June 2011. Association for Computational Linguistics.

[Mirzasoleiman *et al.*, 2016] Baharan Mirzasoleiman, Ashwinkumar Badanidiyuru, and Amin Karbasi. Fast constrained submodular maximization: Personalized data summarization. In *ICML*, volume 48 of *JMLR Workshop and Conference Proceedings*, pages 1358–1367. JMLR.org, 2016.

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

[Patel *et al.*, 2021] Deval Patel, Arindam Khan, and Anand Louis. Group fairness for knapsack problems. In *AAMAS*, pages 1001–1009. ACM, 2021.

[Sviridenko, 2004] Maxim Sviridenko. A note on maximizing a submodular set function subject to a knapsack constraint. *Oper. Res. Lett.*, 32(1):41–43, 2004.

[Wang *et al.*, 2021] Yanhao Wang, Francesco Fabbri, and Michael Mathioudakis. Fair and representative subset selection from data streams. In *WWW*, pages 1340–1350. ACM / IW3C2, 2021.

[Yu *et al.*, 2016] Baosheng Yu, Meng Fang, Dacheng Tao, and Jie Yin. Submodular asymmetric feature selection in cascade object detection. In *AAAI*, pages 1387–1393. AAAI Press, 2016.